

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**  
Fakulta elektrotechnická

**BAKALÁŘSKÁ PRÁCE**

**2014**

**Jiří Pils**



# **ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

Fakulta elektrotechnická

Katedra telekomunikační techniky

## **Projektování síťové infrastruktury pro technologii RTLS v pásmu 2,45 GHz**

**květen 2014**

**Bakalant: Jiří Pils**

**Vedoucí práce: Ing. Jaromír Hrad, Ph.D.**

## Čestné prohlášení

Prohlašuji, že jsem zadanou bakalářskou práci zpracoval sám s přispěním vedoucího práce a konzultanta a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé bakalářské práce nebo její části se souhlasem katedry.

Datum: 23. května 2014

.....  
podpis bakalanta

České vysoké učení technické v Praze  
Fakulta elektrotechnická

katedra telekomunikační techniky

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Pils Jiří**

Studijní program: Komunikace, multimédia a elektronika  
Obor: Síťové a informační technologie

Název tématu: **Projektování síťové infrastruktury pro technologii RTLS  
v pásmu 2,45 GHz**

Pokyny pro vypracování:

Seznamte se s funkcí dostupného systému RTLS a ve shodě s jeho specifikacemi navrhnete, realizujete a nakonfigurujete síťovou infrastrukturu pro pokrytí zadané oblasti (část interiéru budovy) signálem. Provedte analýzu s cílem identifikovat prostředky, které jsou nutné pro účely lokalizace. Následně na platformě Android vytvořte lokalizační aplikaci (klienta pro uživatelské zařízení) a dohledový systém (aplikaci pro lokalizaci objektu či uživatele). Aplikace navrhnete pro použití ve spojení se síťovým serverem, který bude sloužit pro uložení mapových podkladů a databáze lokalizačních údajů. Provedte měření s cílem zhodnotit možné kolísání úrovně signálu; podle výsledků pak vytvořené aplikace optimalizujte.

Seznam odborné literatury:

- [1] Malik, A.: RTLS for Dummies. Wiley Publishing, Inc. 2009. ISBN 978-0-470-39868-5.
- [2] Dokumentace Ekahau RTLS - <http://www.ekahau.com/> [online]

Vedoucí: Ing. Jaromír Hrad, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

  
prof. Ing. Boris Šimák, CSc.  
vedoucí katedry



  
prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 9. 12. 2013

## **Anotace:**

V této práci se zabývám možností lokalizace uvnitř budovy pomocí existující bezdrátové sítě standardu Wi-Fi na principu RTLS. Cílem práce bylo navrhnout a nakonfigurovat bezdrátovou síť s přístupem k síťovému serveru, navrhnout a následně vytvořit aplikaci pro OS Android, která je schopná lokalizace uživatele případně více uživatelů či klientských zařízení v prostorách budovy pokryté vytvořenou sítí. Síťový server slouží jako úložiště pro potřebné databáze, mapové podklady a další nezbytné soubory. Dále jsem vyhodnotil naměřená data s důrazem na kolísání úrovně signálu a provedl jejich vyhodnocení.

## **Klíčová slova:**

Bezdrátová síť, FTP, lokalizace v reálném čase, lokalizační systém, mobilní aplikace, OS Android, síťový server, Wi-Fi

## **Annotation:**

In this task, I'm looking at possibility of localization inside the buildings using existing wireless network running on Wi-Fi standard which will fulfill RTLS principles. Main objective of this task was to design and configure wireless network with access to network server and thereafter create an application that will run on OS Android platform. This application will be able to locate number of users or parts of equipment in part of building covered by signal from created wireless network. The network server serves as a remote storage for databases, map materials and other files needed for flawless function. At the end of the task, I've made a measurement of fluctuation of signal strength and made its evaluation.

## **Keywords:**

FTP, Localization System, Mobile Application, Network Server, OS Android, Real Time Localization, Wireless Network, Wi-Fi

# Obsah

<b>Seznam použitých zkratek .....</b>	<b>7</b>
<b>1. Úvod .....</b>	<b>8</b>
<b>2. Bezdrátové sítě .....</b>	<b>9</b>
2.1 Počítačové sítě dle velikosti .....	9
2.2 Bezdrátový standard Wi-Fi .....	10
2.3 Bezdrátové LAN sítě a základní myšlenka návrhu .....	11
<b>3. Návrh bezdrátové sítě, analýza pokrytí a konfigurace přístupových bodů ....</b>	<b>13</b>
3.1 Návrh bezdrátové sítě.....	13
3.2 Simulace pokrytí pomocí software Ekahau Site Survey a konfigurace AP .....	14
<b>4. Lokalizační systémy.....</b>	<b>18</b>
4.1 Zkratka RTLS a součásti RTLS systému .....	18
4.2 Metody, techniky a algoritmy potřebné pro získání dat.....	20
4.3 Techniky využívající metody trilaterace .....	21
4.4 Techniky využívající metody triangulace .....	22
4.5 Techniky využívající metody „nejbližšího souseda“ .....	23
4.6 Lokalizační systém Ekahau .....	25
<b>5. Tvorba aplikace pro OS Android .....</b>	<b>28</b>
5.1 Specifikace OS Android, jeho vývoj a znaky .....	28
5.1.1 Historie OS Android .....	28
5.1.2 Struktura OS Android .....	28
5.2 Programování aplikací pro OS Android.....	29
5.3 Návrh aplikace lokalizačního systému .....	30
5.4 Tvorba aplikace pro OS Android .....	31
5.5 Vývoj Administrátorské aplikace.....	32
5.6 Popis tříd Administrátorské aplikace .....	35
5.6.1 Třída AboutApp.java .....	36
5.6.2 Třída Configuration.java.....	36
5.6.3 Třída DB_Scan.java.....	36
5.6.4 Třída DB_Supervision.java .....	36
5.6.5 Třída DBHandler_Scan.java .....	37
5.6.6 Třída DBHandler_Supervision.java.....	37
5.6.7 Třída DownloadDB.java .....	37
5.6.8 Třída FTPManager.java .....	37

5.6.9	Třída Global.java .....	38
5.6.10	Třída IOFile.java.....	38
5.6.11	Třída Language.java .....	38
5.6.12	Třída Localization.java .....	38
5.6.13	Třída Map.java.....	38
5.6.14	Třída Multiscan.java .....	38
5.6.15	Třída NetworkValues.java .....	38
5.6.16	Třída RTLS.java .....	39
5.6.17	Třída Scan.java .....	39
5.6.18	Třída Settings.java .....	39
5.6.19	Třída Supervision.java .....	39
5.6.20	Třída SupervisionDBManager.java .....	39
5.6.21	Třída Survey.java.....	39
5.6.22	Třída UploadDB.java.....	39
5.7	Podmínky pro správné použití Administrátorské aplikace .....	40
5.8	Vývoj Uživatelské aplikace.....	40
5.9	Popis tříd Uživatelské aplikace .....	41
5.9.1	Třída DownloadLocList.java .....	41
5.9.2	Třída DownloadMap.java .....	41
5.9.3	Třída HandleTrasnfer.java .....	41
5.10	Podmínky pro správné využití Uživatelské aplikace .....	42
5.11	Vývoj aplikace Lokalizační služba .....	42
5.12	Popis tříd Lokalizační služby .....	42
5.12.1	Třída ClientService.java .....	42
5.13	Podmínky pro správné použití Lokalizační služby .....	43
<b>6.</b>	<b>FTP server .....</b>	<b>44</b>
6.1	Co je to FTP .....	44
6.2	Použitý FTP server, korektní nastavení .....	45
<b>7.</b>	<b>Zhodnocení kolísání úrovní signálu .....</b>	<b>45</b>
<b>8.</b>	<b>Závěr .....</b>	<b>46</b>
	<b>Seznam použité literatury .....</b>	<b>48</b>
<b>9.</b>	<b>Seznam příloh .....</b>	<b>49</b>
9.1	Návod pro vytvoření aplikace v prostředí Eclipse .....	49
9.2	Návody na použití aplikací.....	52



## **Seznam použitých zkratek**

*AP - Access Point (Přístupový bod)*

*DHCP - Dynamic Host Configuration Protocol (Protokol pro dynamickou konfiguraci uživatelského zařízení)*

*FTP - File Transfer Protocol (Protokol pro přenos souborů)*

*GPS - Global Positioning System (Globální polohovací systém)*

*HTTP - Hypertext Transfer Protocol*

*IEEE - Institute of Electrical and Electronics Engineers (Institut pro elektrotechnické a elektronické inženýrství)*

*ITU - International Telecommunication Union (Mezinárodní telekomunikační unie)*

*(W)LAN - (Wireless) Local Area Network ((Bezdrátová) místní síť)*

*LTE - Long Term Evolution*

*MAC - Media Access Control*

*(W)MAN - (Wireless) Metropolitan Area Network ((Bezdrátová) městská síť)*

*NAT - Network Address Translation (Překlad síťových adres)*

*OS - Operating System (Operační systém)*

*(W)PAN - (Wireless) Personal Area Network ((Bezdrátová) osobní síť)*

*PoE - Power Over Ethernet (Napájení po Ethernetu)*

*RFID - Radio Frequency Identification*

*RSSI - Received Signal Strength Indicator (Indikátor přijaté síly signálu)*

*RTLS - Real-Time Locating System (Lokalizační systém v reálném čase)*

*SQL - Structured Query Language*

*SW - Software*

*UMTS - Universal Mobile Telecommunication System (Univerzální mobilní telekomunikační systém)*

*(W)WAN - (Wireless) Wide Area Network ((Bezdrátová) rozlehlá síť)*

*Wi-Fi- Wireless Fidelity*

*XML- Extensible Markup Language*

# 1. Úvod

Vzhledem k aktuálnímu trendu ve světě, používání tzv. chytrých telefonů a tabletů k lokalizaci uživatele ať již při cestě autem, nebo procházení se po městě nebo navigaci v přírodě pomocí GPS senzorů, jsem se rozhodl pro pojetí lokalizace a možné navigace uživatele v částečně odlišném směru, a to lokalizace pomocí bezdrátové Wi-Fi sítě.

Důvodem k mému rozhodnutí je to, že jak Wi-Fi sítě, tak mobilní zařízení využívající operační systém Android, jsou dnes všudypřítomné. Operační systém Android jsem si vybral proto, že vývoj aplikací pro tuto platformu probíhá v opensource prostředí, které využívá programovací jazyk Java, a veškeré informace potřebné pro vývoj aplikací pro Android jsou volně dostupné na internetu a dá se v nich rychle a snadno orientovat.

Pro samotnou lokalizaci je nezbytná již výše zmíněná Wi-Fi síť. Proto ji bude potřeba navrhnout korektním způsobem, nakonfigurovat všechny přístupové body a případně optimalizovat. Protože ale tato oblast lokalizace není úplně neprobádané území, rozhodl jsem se nejprve v kooperaci s mým kolegou Martinem Cihlářem, na jehož práci budu místy odkazovat, prostudovat již existující RTLS systém od společnosti Ekahau a na funkcích a přednostech tohoto systému budu následně stavět.

Další nezbytnou součástí celé práce je síťový server, na který se budou ukládat veškerá potřebná data. Zvolil jsem si server FTP, protože jeho implementace do lokalizační aplikace samotné je nejjednodušší.

V úvodu práce nejdříve projdu a vysvětlím principy a funkce, bez kterých by aplikace ani samotná lokalizace nemohla fungovat a dále se budu zabývat zvolenými postupy, vysvětlím veškeré funkce aplikace a v závěru zhodnotím získané výsledky a zaměřím se na možné problémy způsobené například kolísáním úrovně signálu od jednotlivých AP nebo rozdílností Wi-Fi adaptérů použitých ve smartphonech a tabletech lišících se výrobcem nebo dokonce modelovou řadou výrobku.

## 2. Bezdrátové sítě

### 2.1 Počítačové sítě dle velikosti

Komunikace mezi různými zařízeními (PC, tiskárny, AP, smartphony a tablety) může probíhat buď pomocí kabelů, nebo pomocí prvků bezdrátové sítě, které dokáží elektrický signál z kabelu změnit na elektromagnetickou vlnu. Tento převod se uskutečňuje na fyzické vrstvě těchto síťových prvků. Propojením síťových prvků může vzniknout složitá a nepřehledná síť. Proto je nutné tuto infrastrukturu rozdělit na menší části, které jsou snáze spravovatelné a lépe a přehledněji uspořádané. Bezdrátových technologií pro komunikaci je celá řada. Proto se zde nebudu zabývat sítěmi standardů GSM/UMTS ani LTE a jim podobnými, ani technologií WiMAX, ani technologií Bluetooth, ale sítěmi, které souží ke komunikaci osobních počítačů, tiskáren případně dalších zařízení včetně smartphonů a tabletů.

Počítačové sítě se dělí rozdělit podle své geografické rozlehlosti na:

- 1) Síť PAN (Personal Area Network) – Síť o nejmenší rozloze, nejčastěji v domácnostech, v těchto sítích nebývá kladen důraz na přenosovou rychlost (ta dosahuje řádově několik Mbit/s).
- 2) Síť LAN (Local Area Network) – Síť propojující koncové uzly typu PC, server, tiskárna. Tato síť bývá v soukromé správě a působí na malém území. V současné době rozmachu počítačových sítí je síť PAN a LAN často to samé.
- 3) Síť MAN (Metropolitan Area Network) – Síť která je pod správou města nebo její části město využívá. Může sdružovat některé LAN sítě.
- 4) Síť WAN (Wide Area Network) – Síť, která je svou rozlohou největší. V těchto sítích je kladen důraz na rychlost a bezpečnost komunikace. Tyto sítě nemívají zpravidla jednoho správce. Typickým příkladem takovéto sítě je internet.

Tento výčet zahrnuje pouze sítě, jejímž médiem je propojovací kabel. Pro mou práci jsou však důležitější bezdrátové sítě. Jejich dělení prakticky vychází z dělení uvedeného výše. Pro pořádek tedy uvedu také typy bezdrátových sítí a zmíním příklady využití a nejčastěji používaná zařízení v těchto sítích:

- 1) WPAN (Wireless Personal Area Network) - Tato síť bývá tvořena zařízeními osobní potřeby (osobní počítače s bezdrátovým adaptérem, notebooky, bezdrátové tiskárny, telefony vybavené bezdrátovým adaptérem, tablety, fotoaparáty,...). Vzdálenosti mezi zařízeními jsou obvykle menší než 20 metrů.
- 2) WLAN (Wireless Local Area Network) – Stejně jako LAN, i WLAN se často prolíná s WPAN. Tento typ sítě je v dnešní době nasazován všude, kde je preferováno bezdrátové řešení, zejména v podobě bezdrátového přístupu k internetu. Využití proto nachází například v kavárnách, knihovnách, v MHD, firmách a samozřejmě v domácnostech.
- 3) WMAN (Wireless Metropolitan Area Network) – Tato síť se využívá ve větších městech. Vzniká za účelem rozvoje městských částí. U nás se ale metropolitní sítě prakticky nevyužívají a jsou nahrazeny LAN/WLAN sítěmi.
- 4) WWAN (Wireless Wide Area Network) - Bezdrátové WAN sítě jsou tvořeny hlavně technologií WiMAX nebo mobilní technologie LTE, UMTS či GSM.

## 2.2 Bezdrátový standard Wi-Fi

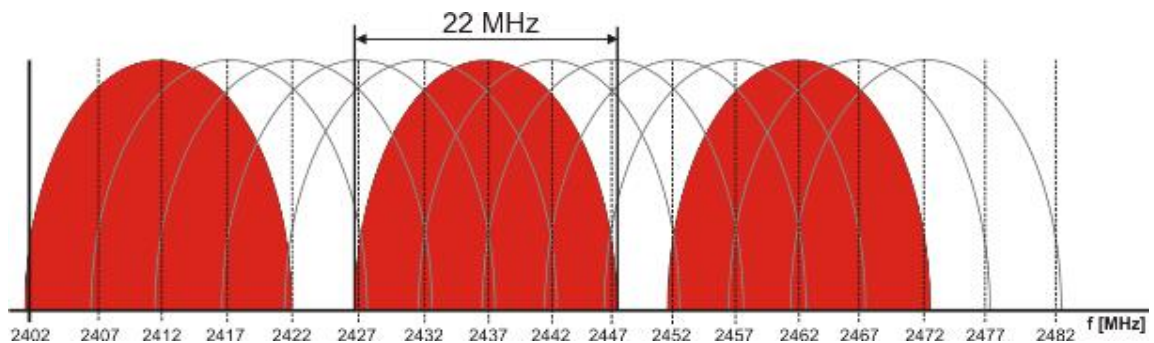
Tato technologie není nic nového, má počátky již v roce 1942, kdy si hudební skladatel George Antheil a herečka Hedy Lamarr nechali patentovat princip rozprostřeného spektra. Mezi 60. a 80. rokem se tato technologie používala výhradně pro vojenské účely, od 80. roku bylo možné i civilní využití. Zkratka Wi-Fi, vytvořená skupinou Wireless Ethernet Compatibility Alliance, neměla původně nic znamenat, ale později se z ní stalo spojení *wireless fidelity* (bezdrátová věrnost) - analogicky ke zkratce *hi-fi* (*high fidelity*, vysoká věrnost).

Technologie Wi-Fi je standardizovaná organizací *Institute of Electrical and Electronics Engineers* (IEEE), která, kromě jiných, vydává standardy pro jednotlivé vývojové stupně Wi-Fi technologie.

Komunikace prostřednictvím Wi-Fi je provozována v bezlicenčním frekvenčním pásmu. Tyto pásma se označují ISM (*Industrial, scientific and medical*) a slouží pro volné použití v průmyslu, vědě a lékařství. O regulaci těchto pásem se stará organizace ITU-R (*International Telecommunication Union - Radiotelecommunications*).

Vzhledem k tomu, že technologie Wi-Fi je provozována v pásmech 2,45 GHz a 5 GHz, jsem si musel vybrat, které pásmo pro svou práci využiji. Bohužel, pásmo 2,45 GHz je v dnešní době velmi zarušené. Jednak je to díky obrovskému množství bezdrátových sítí v tomto pásmu a jednak také díky vyššímu povolenému vyzářenému výkonu 100 mW a s tím souvisejícímu vyššímu dosahu. Pásmo na 5 GHz má výkon omezený na 25 mW. Vyšší frekvenci je dosažena větší přenosová rychlost, nižším výkonem je omezený dosah. Protože ale většina koncových zařízení stále ještě podporuje pouze nižší frekvenční pásmo, bylo tedy rozhodnutí nakonec celkem snadné a pro práci jsem si zvolil pásmo 2,45 GHz. Z toho důvodu se dále nebudu zabývat vyšším frekvenčním pásmem, nicméně řešení práce je možné aplikovat i v pásmu 5 GHz.

Aby nedocházelo ke vzájemnému rušení sítí, bylo nutné rozdělit celé frekvenční pásmo do několika kanálů (*Obr. 2.1*)[11]:



*Obr. 2.1: Uspořádání kanálů dle IEEE 802.11b*

Toto řešení umožňuje dostatečný odstup sítí vysílajících alespoň 3 kanály od sebe. Nicméně tato podmínka je v praxi až na výjimečné případy nesplnitelná z důvodu zarušení pásma. Celkový počet kanálů se může lišit pro jednotlivé země, kde se síť provozuje. V České republice lze kanálů využít 13.

Technologie Wi-Fi se dělí, jak jsem již zmínil, dle IEEE na několik standardů, které se liší hlavně rychlostí a pásmem, ve kterém vysílají. Přehled těchto standardů, včetně použitých modulací, je uveden v následující tabulce:

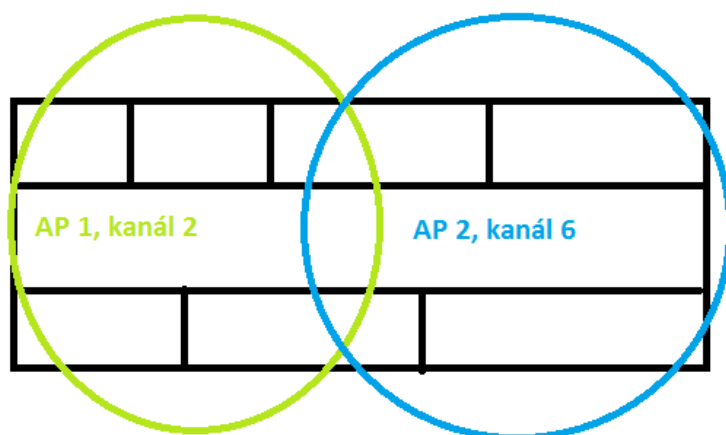
Wi-Fi standardy podle IEEE				
Standard	Rok vydání	Přenosová modulace	Pásmo [GHz]	Maximální rychlost [Mbit/s]
IEEE 802.11	1997	DSSS, FHSS	2,4	2
IEEE 802.11a	1999	OFDM	5	54
IEEE 802.11b	1999	DSSS	2,4	11
IEEE 802.11g	2003	OFDM	2,4	54
IEEE 802.11n	2009	MIMO OFDM	2,4; 5	600
IEEE 802.11ac	2013	MU-MIMO OFDM	5	1000
IEEE 802.11ad	2014	OFDM	2,4; 5; 60	7000

Tab. 2.1: Wi-Fi standardy dle IEEE

### 2.3 Bezdrátové LAN sítě a základní myšlenka návrhu

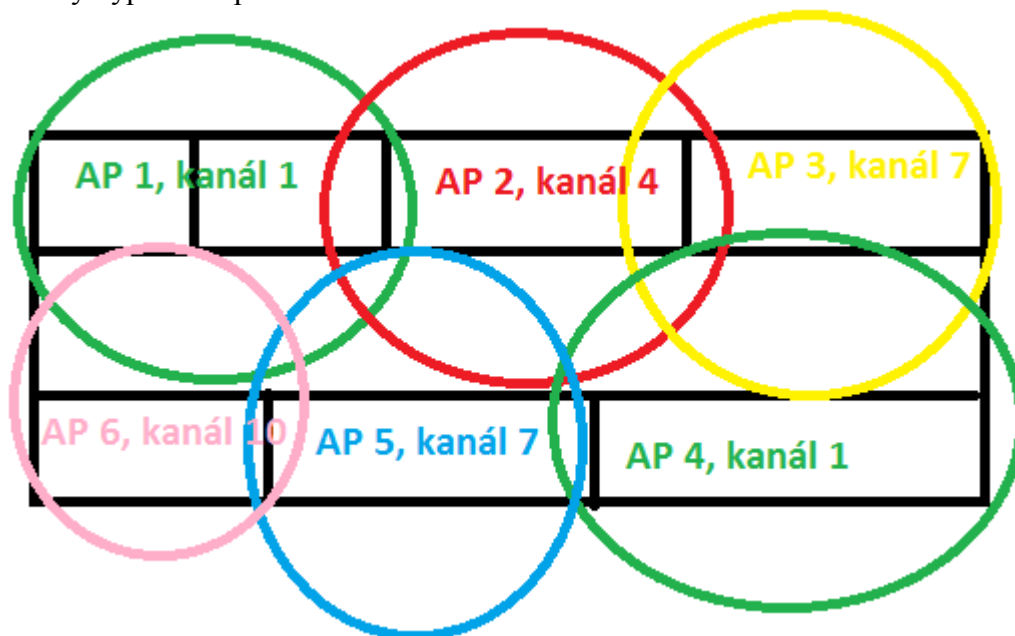
Vlastnosti bezdrátové lokální sítě WLAN jsem popsal již výše. Nyní bych se rád na tento typ bezdrátové sítě podíval z hlediska návrhového a funkčního. K vytvoření bezdrátové sítě potřebujeme zařízení, kterému se říká přístupový bod (AP), případně Wi-Fi router.

Dnes jsou asi nejvíce rozšířená standardy IEEE 802.11g a IEEE 802.11n. Oba standardy dosahují pro potřebu řešení dostatečných přenosových rychlostí. Při návrhu sítě je třeba, abychom vzali v úvahu rozlohu prostoru k pokrytí a případné stínění zdi případně jinými předměty. S tím následně souvisí počet AP, které je třeba využít. Dále také velice záleží na využití budované sítě. Proto se například počet AP odlišuje v případě pokrytí patra budovy sítí, která má za úkol pouze připojení koncových zařízení a sítí, která má za úkol lokalizovat uživatele. Druhá zmíněná využije mnohem vyšší počet všude viditelných AP, než první, u které stačí AP zpravidla 1 nebo 2. Příklad takovýto způsobů pokrytí je uveden na následujících obrázcích (Obr. 2.2 a Obr. 2.3):



Obr. 2.2: Příklad návrhu sítě pro jiné účely než lokalizační

Také musíme brát v úvahu dostatečný rozestup kanálů, na kterém vysílají jednotlivá AP. Pro uvedený obrázek by tedy případně 3. AP muselo vysílat na kanálu dostatečně vzdáleném od kanálu 2 i 6, minimálně o 3 kanály. Nejbližší použitelný kanál by tedy byl kanál 9. Díky tomu je příklad návrhu pro lokalizační potřeby o něco složitější a mohl by vypadat například takto:

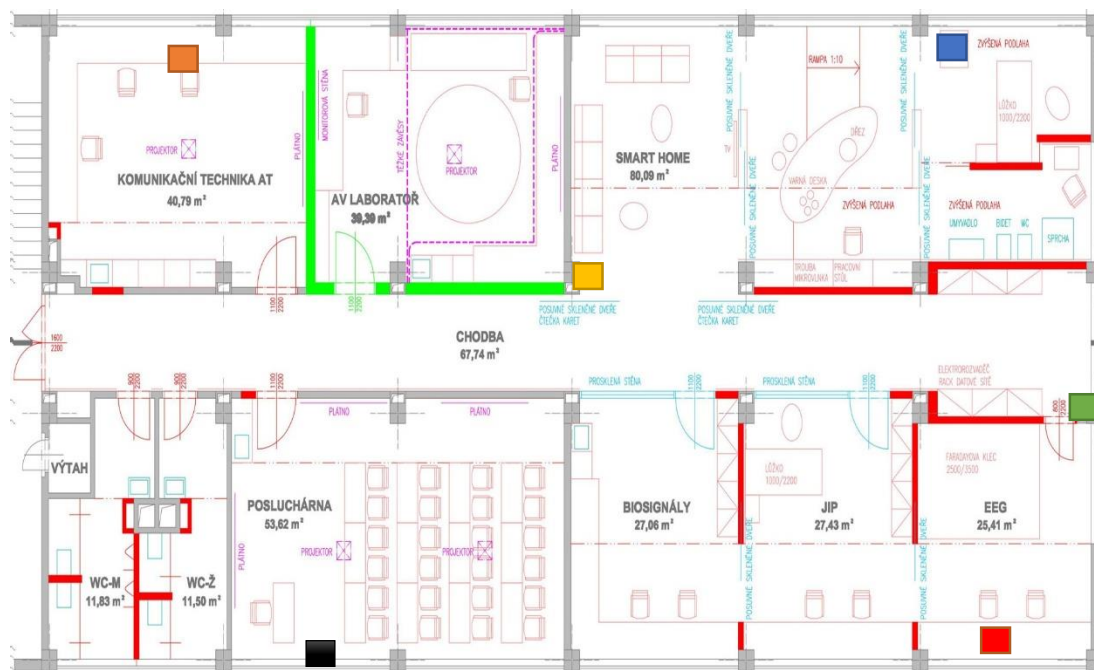


*Obr. 2.3: Příklad návrhu sítě pro lokalizační účely*

### 3. Návrh bezdrátové sítě, analýza pokrytí a konfigurace přístupových bodů

#### 3.1 Návrh bezdrátové sítě

Při návrhu bezdrátové sítě jsem se v první řadě řídil rozlohou a tvarováním prostoru, ve kterém se síť bude nacházet. Na *Obr. 3.1* je mapa tohoto prostoru. Jde o část 3. patra v prostorách Elektrotechnické fakulty, blok A. Vzhledem k různorodému využití prostor a tudíž i různým útlumům jednotlivých místností jsem se nakonec rozhodl využít 6 AP napájených dle standardu IEEE 802.3af (PoE). K tomuto názoru jsem se přiklonil hlavně z toho důvodu, že AP budou připojeny k síťovému FTP serveru přes router, který tento standard také podporuje, a je proto zbytečné, aby od AP vedl ještě další kabel do zásuvky. Kam jsem se AP rozhodl po důkladném zvážení umístit je také zobrazeno na *Obr. 3.1*.



*Obr. 3.1: Mapa prostoru pokrývaného signálem*

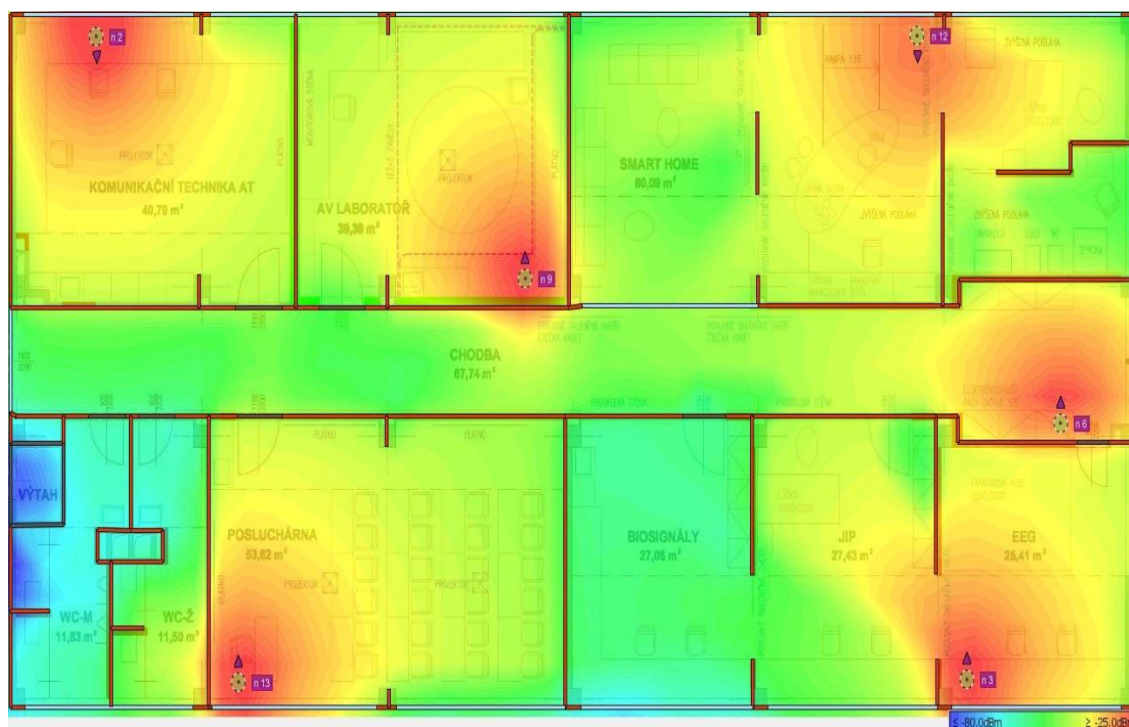
### Seznam používaných AP:

- Oranžové AP 1, Mikrotik RB751U-2HnD, Wi-Fi IEEE 802.11n, 2,45GHz, kanál 2
- Žluté AP 2, Mikrotik RB751U-2HnD, Wi-Fi IEEE 802.11n, 2,45GHz, kanál 9
- Modré AP 3, Mikrotik RB751U-2HnD, Wi-Fi IEEE 802.11n, 2,45GHz, kanál 12
- Černé AP 4, Mikrotik RB751U-2HnD, Wi-Fi IEEE 802.11n, 2,45GHz, kanál 13
- Červené AP 5, Mikrotik RB751U-2HnD, Wi-Fi IEEE 802.11n, 2,45GHz, kanál 3
- Zelené AP 6, Mikrotik RB751U-2HnD, Wi-Fi IEEE 802.11n, 2,45GHz, kanál 6

Jak je ze seznamu vidět, snažil jsem se, aby pokud možno nedocházelo k interferenci na blízkých kanálech. K těmto AP bylo ještě potřeba zakoupit PoE lištu: POE-PAN12, 12 - portový pasivní POE injektor panel a napájecí adaptér k této PoE liště: Napájecí adaptér spínaný 24 V, 5A 120W - Zd24V5A (s kabelem síťovým k počítači, 1.8m).

## 3.2 Simulace pokrytí pomocí software Ekahau Site Survey a konfigurace AP

Schéma rozložení přístupových bodů jsem, s pomocí kolegy Cihláře vložil do programu *Ekahau Site Survey* a nechal ho vytvořit simulaci. Její znázornění je na *Obr. 3.2*. Podrobnější popis systému Ekahau obsažen v následující kapitole.

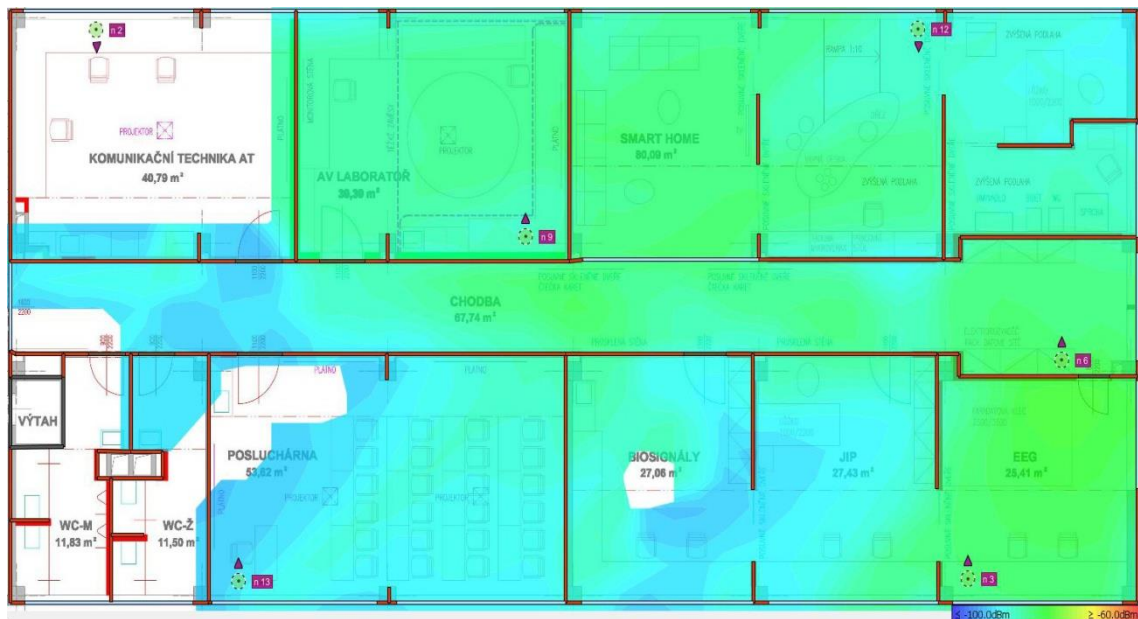


Obr. 3.2: Simulace rozložení sil signálů



Simulace na obrázku zobrazuje rozvržení síly signálu od jednotlivých přístupových bodů a stupnice síly signálu je nastavena v rozmezí od modré (-80dBm) do červené (-20dBm) barvy. Nejhorší pokrytí je na toaletách a ve výtahu, což je zobrazeno modrou barvou. Nicméně ve výtahu osoby lokalizovat nutně nemusíme, tudíž nám i takové pokrytí vyhovuje.

Také bylo potřeba zjistit vzájemnou interferenci vysílacích kanálů. Tato simulace je zobrazena na *Obr. 3.3*. Čím nižší síla je signálu, tím lepší jsou podmínky pro naši síť. V zelených oblastech nám části kanálů interferují, překrývá se asi 5 MHz z 20 MHz kanálu, což je dostačující, hlavně pokud vezmeme v potaz umístění sítí v okolí, kterých je kolem spousta a vysílají téměř na všech kanálech. Právě díky těmto sítím byly vybrány kanály, na kterých naše AP vysílají a také samotné umístění těchto AP. Požadavkem na tuto síť bylo totiž také to, aby pokud možno neovlivňovala okolní síť.



*Obr. 3.3: Simulace interference kanálů*

Pro naši práci bylo ale ze všeho nejdůležitější to, aby pokud možno každá část prostoru byla pokryta alespoň třemi (nejlépe však čtyřmi) AP. I k tomu můžeme SW od firmy Ekahau využít a simulace je zobrazena na *Obr. 3.4*. Červená barva, dle stupnice, znamená alespoň 6 viditelných AP a modrá je 1 AP a méně (tedy spíše 1). Nicméně i přesto, že toalety tedy mají být pokryty jen 1 či 2 AP jsme se nakonec rozhodli tuto infrastrukturu realizovat a vyzkoušet.



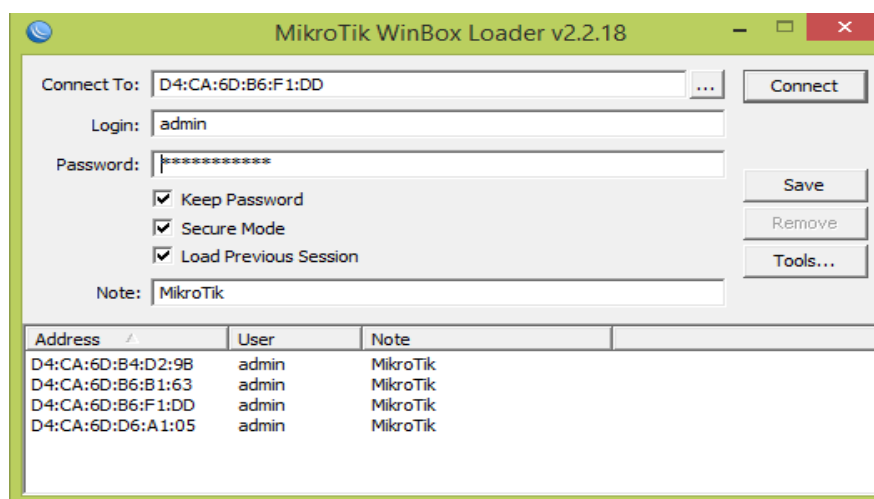
Obr. 3.4: Simulace počtu AP viditelných pro zařízení

Konfigurování AP probíhalo v grafickém uživatelském rozhraní (GUI) pomocí programu WinBox od MikroTiku Obr. 3.5. Tento program je velmi intuitivní. Na následujícím obrázku je okno, ve kterém si pomocí tlačítka „...“ vybereme připojené zařízení, které se hlásí svou MAC adresou. Díky tomu není třeba DHCP server pro získání IP adres, případně jinak IP adresu nejdříve nastavovat. Dále je potřeba zadat přihlašovací údaje:

Login: admin

Password: (prázdné) – po nastavení heslo RTLSAndroid

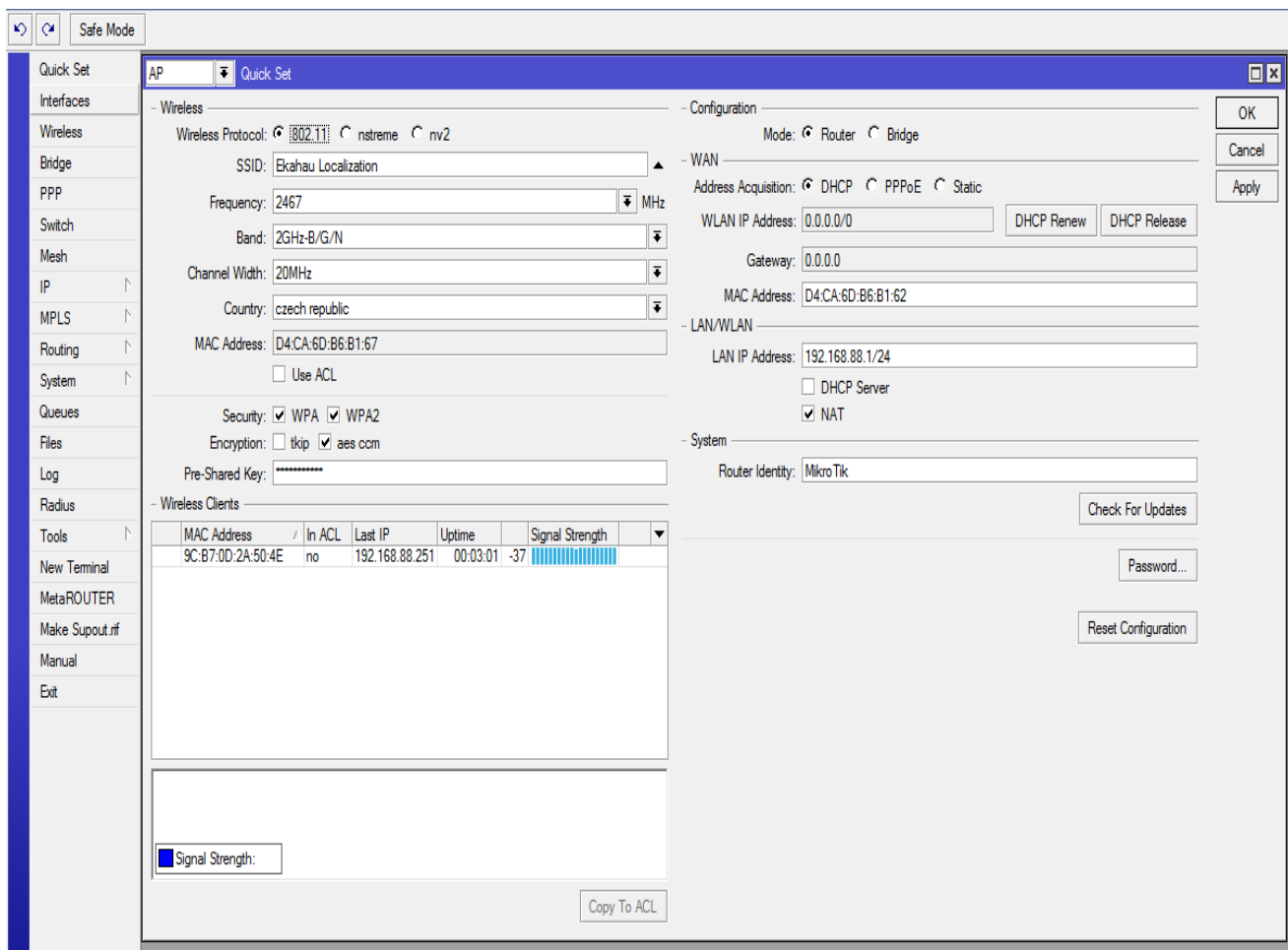
Přihlašovací údaje je možné uložit mezi nejpoužívanější konfigurace.



Obr. 3.5: Přihlašování do AP

Po přihlášení dostaneme ke konfiguraci základních i velmi pokročilých funkcí AP, které ale nebylo potřeba využít.

Na dalším obrázku (Obr. 3.6) je konfigurace základních parametrů AP.



Obr. 3.6: Konfigurace základních parametrů AP

Je potřeba nastavit následující:

- Protokol bezdrátového vysílání na 802.11 (tedy Wi-Fi)
- SSID: Ekahau Localization
- Zabezpečení WPA/WPA2 aes ccm s heslem RTLSAndroid
- Vysílací frekvence 2 GHz, standard IEEE 802.11 b/g/n
- Země použití Česká republika
- Kanál vysílání (2, 3, 6, 9, 12, 13)
- Šířka pásma 20 MHz
- DHCP server na AP v dostatečném rozsahu

Dále je možné na záložce Interfaces->Wireless->Advanced Settings->Tx power možné nastavit stabilní vysílací výkon AP (budu používat hodnotu 15 dBm), případně výkon nastavit na automatický, což ale později může znehodnotit survey a následnou lokalizaci, tím, že AP automaticky reguluje signál tak, aby klientské zařízení zůstala co nejdéle připojená k AP a aby komunikace byla co nejkvalitnější.

Nyní je potřeba AP rozmístit po prostoru a provést simulaci pokrytí pomocí softwaru Ekahau Site Survey.

## 4. Lokalizační systémy

### 4.1 Zkratka RTLS a součásti RTLS systému

Zkratka RTLS vyjadřuje anglické označení pro *Real-Time Locating System*. Mezi nejznámější RTLS systémy patří například GPS. Princip GPS je založen na několika satelitech, které pokrývají každé místo na Zemi. Pro úspěšné určení polohy jsou třeba nejméně tři satelity a pomocí GPS přijímače se zobrazí vlastní poloha na mapě v GPS přijímači. GPS však funguje pouze v místech s přímou viditelností satelitů a v budovách či prostorách pod zemí je GPS vysoce nepřesná, až nefunkční. Pro vyřešení problému lokalizace uvnitř budov byly vyvinuty způsoby a možnosti, jak využít stávajících technologií, díky nimž bude lokalizace v budovách možná. Příklady technologií, které se dají využít, jsou například Bluetooth, RFID, Wi-Fi, ZigBee, digitální TV, ultrazvuk či infračervené záření.

Abychom mohli říci, jak přesná je lokalizace, je nejprve potřeba tuto přesnost definovat pomocí tzv. míry přesnosti a stupně lokalizace. Ta uvádí odlišnost odhadu pozice určené RTLS technologií a pozice, na které se nacházíme. Stupeň lokalizace pak udává úroveň preciznosti RTLS technologie. Stupně lokalizace jsou uvedeny zde [5] :

- a) *lokalizace na základě zóny*: Tag se dostane do části budovy (například zaměstnanec přešel z bloku B do bloku C)
- b) *lokalizace na úrovni pokoje*: Poloha tagu udává pokoj, ve kterém se nachází
- c) *lokalizace na úrovni části pokoje*: Tag nám říká, že je například u umyvadla v koupelně
- d) *lokalizace na úrovni průchodu*: Poloha tagu je udávána pomocí pohybu kolem přijímače
- e) *lokalizace vztažená k něčemu*: Poloha tagu je odhadnutá pomocí ostatních tagů nebo vysílačů
- f) *lokalizace precizní*: Poloha tagu je dána souřadnicemi, případně přímo hodnotou v jednotkách vzdálenosti (tag je na pozici  $x, y$ )

Z tohoto vyplývá, že přesnost RTLS technologie je daná použitým stupněm lokalizace. Proto například nemůžeme určit polohu zařízení s přesností na metry, když lokalizační systém funguje na základě „lokalizace vztažené k něčemu“.

RTLS technologii je vhodné využít zejména v oblastech a místech, kde je třeba neustálý dohled nad osobami nebo zařízeními. Takovým příkladem mohou být nemocnice, továrny nebo také vězení. RTLS technologie se zde mohou zhostit řady úkolů. Mohou samozřejmě sledovat pohyb a tím i stav uživatelů. Na tuto funkci je možné implementovat další, například využití zobrazovacího zařízení, jež upozorní a oznámí polohu uživatele dozorců.

Základním cílem RTLS technologií je tedy lokalizovat osoby a předměty. Způsob lokalizace se dá rozdělit na dva typy:

- 1) Lokalizace na vyžádání - osoba chce být lokalizována na mapě ve svém zařízení
- 2) Lokalizace v reálném čase - není nutné zasílat požadavek, protože daná osoba či předmět je neustále pod dohledem.

K úspěšné lokalizaci je potřeba zařízení v angličtině označovaný jako „tag“ nebo *identifikátor*, což je první ze tří důležitých částí RTLS systému. Je to zařízení vybavené bezdrátovým adaptérem, které je lokalizováno systémem. Proto, abychom mohli někoho nebo něco lokalizovat, musíme daný subjekt tímto zařízením opatřit. Technologii využitou pro bezdrátové tagy můžeme rozdělit do tří skupin [5]:

- 1) Pasivní tagy - nemají vestavěnou baterii, takže samy o sobě nemohou generovat radiové vysílání. Mezi tato zařízení patří například RFID tagy. Aby mohly být lokalizovány, musí být nejprve vystaveny rádiovému vysílání. RFID tagy jsou daleko levnější a menší, než jiné typy. Disponují velmi krátkým dosahem.
- 2) Napůl pasivní tagy - od pasivních liší pouze tím, že obsahují baterii. Tu však nevyužívají pro generování signálu, ale například pro zjišťování teploty okolí a udržování této hodnoty v paměti.
- 3) Aktivní tagy - obsahují baterii, která je využívána pro generování signálu. Po určitých intervalech se bude tag sám hlásit a tím udávat svoji polohu.

Téměř každý tag je proprietárním systémem firmy, která ho vyrábí. S nástupem a rozvojem chytrých telefonů, tabletů a dalších mobilních zařízení včetně notebooků a netbooků, se ale nyní rozvíjí možnosti lokalizace i těchto zařízení. Stačí tedy na tato zařízení nainstalovat lokalizačního klienta a tím vznikne víceúčelový tag. Díky tomu, že jsou tato zařízení již běžně k dispozici, není tedy potřeba nakupovat zbytečně další tagy pro lokalizaci. Je ale samozřejmě nutné vzít v potaz situaci, pro jaký účel chceme tagy nasadit. Není například vhodné chtít návštěvníky obchodního centra značkovat náramky a tabletem s mapou, aby věděli, kde se nachází, ale je možné využít jejich stávajícího zařízení pro volitelnou instalaci aplikace a tím jim dopřát možnost lokalizace, pokud budou chtít.

Dalším základním prvkem RTLS technologií je *dohledový software* [5]. Jeho úkolem je poskytovat vizuální znázornění polohy dané osoby nebo zařízení. Tento software může být obsluhován z dohledového centra pověřenou osobou nebo může být součástí chytrého telefonu, či tabletu v podobě mobilní aplikace, jež využívá lokalizovanou osobu.

Posledním ze základních prvků RTLS je *lokalizační engine* [5]. Jeho úkolem je určit výslednou polohu z informací, které jsou získávány přímo z tagu. Lokalizační engine je software, který je schopen určit výslednou polohu pomocí matematických postupů a instrukcí. Zpravidla se instaluje na servery organizace, která má na starost dohled, a kam je směrována veškerá komunikace pro lokalizaci určená a odkud je také možno zasílat výsledky po celé síťové infrastruktuře. Je ale samozřejmě možné nainstalovat tento engine na zařízení, v němž běží i dohledový software.

RTLS technologie je tedy složena ze tří prvků: tag, dohledový software a lokalizační stroj. To vše je implementováno do WLAN sítě organizace, která požaduje lokalizaci osob či věcí.

## 4.2 Metody, techniky a algoritmy potřebné pro získání dat

V této podkapitole se zaměříme na poslední prvek RTLS technologií, jímž je lokalizační engine a matematické algoritmy, které lze využít pro výpočet pozice [5, str. 40].

Celý proces získání informací o poloze tagu je rozdělen na dvě dílčí části [5, str. 38]. V první části je potřeba využít vybrané techniky pro získání potřebných dat, které definují vzájemný vztah mezi tagem a přístupovým bodem WLAN sítě. Mezi potřebná data patří zejména doba trvání, za kterou signál dorazí z vysílače do přijímače, tedy z tagu do AP případně obráceně. Další veličinou je vzdálenost mezi těmito dvěma zařízeními, která lze určit právě z doby trvání šíření signálu. Poslední často využívanou veličinou je síla signálu v místě tagu, udávaná většinou v jednotce dBm, případně v mW. Tyto tři veličiny jsou na sobě závislé. Jejich závislosti se dají popsat vztahy pro výpočet vzdálenosti  $D$  a pro ztráty šířením  $L_p$ .

- 1) Vzdálenost  $D$  při šíření elektromagnetické vlny prostorem za čas  $t$ , při rychlosti šíření vlny  $c$  ( $c = 3 * 10^8$  m/s; rychlost světla):

$$D = c * t \quad [m] \quad (1)$$

- 2) Empirický model pro ztráty (útlum) šířením  $L_p$  vlny v závislosti na vzdálenosti  $d$  od vysílače [6, (8.2)]:

$$L_p(d) = L_0(d_0) + 10 * n * \log\left(\frac{d}{d_0}\right) \quad [dB] \quad (2)$$

kde ztráty volným prostorem  $L_{FSL}$  jsou [6, (3.8), upraveno]:

$$L_{FSL} = L_0(d_0) = 10 * \log\left(\frac{4 \pi d}{\lambda}\right)^2 \quad [dB] \quad (3)$$

$d_0$  - referenční vzdálenost pro pikobuňku (prostor budovy) je 1 metr

$n$  - útlumový činitel, v budově na přímou viditelnost 1,6 - 1,8 a na nepřímou 4 - 6

$\lambda$  - vlnová délka elektromagnetické vlny

Druhá část procesu má za úkol zpracování získaných hodnot pomocí vybraného matematického algoritmu. Díky němu získáme výsledné souřadnice určující polohu lokalizovaného tagu. Následující výpis obsahuje principy nejčastěji využívaných technik [5, str. 38] pro získávání dat.

## 4.3 Techniky využívající metody trilaterace

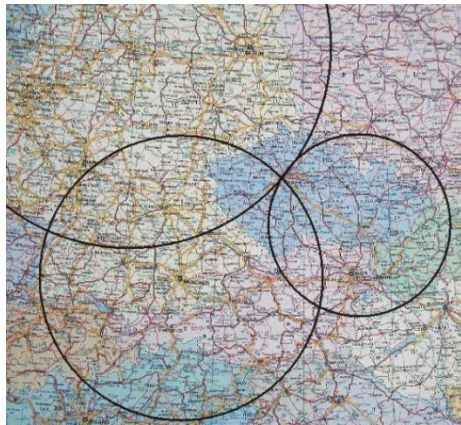
### Trilaterace

Základem tohoto algoritmu je průnik 3 kružnic, díky kterému zjistíme požadovanou pozici. Pokud tedy máme alespoň tři AP a jeden tag, pak pomocí techniky TOA, TDOA, případně RSSI, stanovíme jednotlivé vzdálenosti mezi ním a AP. Tři vzdálenosti vztažené k daným AP, definují kružnici, kterou musíme danému AP opsat. Po opsání kružnic všem AP získáme požadovaný průsečík (Obr. 4.1). Dále potřebujeme znát přesnou polohu všech AP, abychom mohli pro výpočet pozice použít následující vztah (7):

$$r_i^2 = (x - x_i)^2 + (y - y_i)^2, \quad i = 1, 2, 3 \dots N \quad (7)$$

$r$  - je poloměr kružnice (vzdálenost od AP),  $N$  - počet AP

Algoritmus nelze využít pro nižší počet AP, než 3.



Obr. 4.1: Model trilaterace [13]

#### 1) Technika TOA (Time of Arrival)

- technika založená na měření doby, za kterou signál urazí vzdálenost mezi vysílačem (tag) a přijímačem (AP). Doba je počítána od určitého časového okamžiku  $t_0$ , který je stejný pro obě účastnické strany. Obě strany spolu musí být velmi kvalitně sesynchronizovány. Protože je tato technika velmi náchylná ke zpoždění signálu, nastává problém tehdy, když se v dráze signálu objeví jakákoliv překážka. Stačí jen zlomek vteřiny, kdy signál změnil dráhu šíření, aby překážku překonal, a při prodloužení časového intervalu o 100 nanosekund změni vypočtenou dráhu o 30 metrů. Výsledná vzdálenost je, po změření časového intervalu, v lokalizačním enginu vypočtena dle vztahu (1) a slouží pro výpočet pozice pomocí metody *trilaterace*.

#### 2) Technika TDOA (Time Distance of Arrival)

- technika závislá na měření časových okamžiků (podobně jako TOA). Je založená na tom, že tag vyšle signál, který obdrží všechny AP, ale v různých časových okamžicích. V lokalizačním enginu poté dojde ke zjištění rozdílů mezi těmito okamžiky a podle známých souřadnic pozic jednotlivých AP se určí výsledná poloha tagu. I zde se využívá *trilaterace*, ale nepracuje se s průniky kružnic, ale hyperbol. Při použití této techniky je třeba sesynchronizovat pouze AP, z jejichž časů se stanovují jednotlivá zpoždění, proto není nutné využívat počáteční časový okamžik  $t_0$  a tím synchronizaci i pro tag.

## 4.4 Techniky využívající metody triangulace

### Triangulace

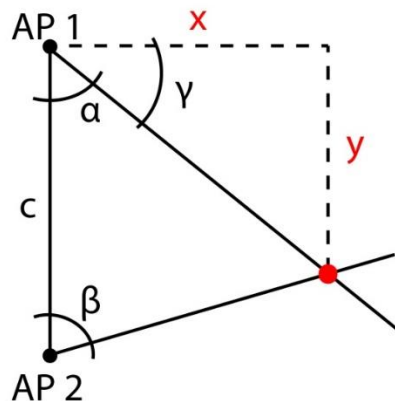
Principem této metody je výpočet polohy na základě úhlů dopadu signálu, které získáme pomocí AOA. Pro výpočet postačují pouze dva úhly, tedy dvě AP, které jsou dány odklonem od referenčního směru přístupových bodů. K výpočtu použijeme vztah (8) [7]:

$$x = \frac{c \times \tan(\alpha) \times \tan(\beta)}{\tan(\alpha) + \tan(\beta)} \quad (8)$$

Díky němu získáme velikost kolmice na spojnici dvou AP. Z tohoto pak jednu souřadnici, například X. Druhá souřadnice se vypočítá dle vztahu (9), kde výsledná pozice je vztažena k AP 1:

$$y = \tan(|90^\circ - \alpha|) * x \quad (9)$$

Princip triangulace je na Obr. 4.2.



Obr. 4.2: Model triangulace

#### 1) Technika AOA (Angle of Arrival)

- technika, která určuje směr signálu, ze kterého signál přichází. AP zde má definován určitý referenční směr vysílání a přijímání signálu a díky jeho citlivé směrové anténě poté dokáže určit úhel příchodu signálu. Tento je určen v lokalizačním enginu a slouží jako vstupní veličina pro algoritmus *triangulace*. Jelikož se nám ale jedná o lokalizaci uvnitř budov, je téměř jisté, že se signál šíří více cestami pomocí odrazů. Pokud tedy tag a AP nejsou v přímé viditelnosti, může rozdíl úhlu řádově jednotek stupňů ve výsledku způsobit chybu v řádu jednotek až desítek metrů. Částečné zlepšení lze provést zvýšením počtu směrových antén případně jejich kvality, nicméně stále je tato technika pro využití uvnitř budov velmi nepřesná.



## 4.5 Techniky využívající metody „nejbližšího souseda“

### Nejbližší soused

Tento algoritmus získává výsledné souřadnice díky databázi dat. Tato databáze obsahuje soubor dat, které vyplývají z metody s anglickým označením *Location Fingerprinting*, kterou popíší později a kterou jsme s kolegou Cihlářem také nakonec využili pro navrhovaný systém.

V tomto algoritmu [8] se prozatím spokojíme s tvrzením, že máme k dispozici vektor  $P$ , který obsahuje takový počet  $N$  hodnot RSSI, jež je v daném okamžiku k dispozici od přístupových bodů. Dále obsahuje souřadnice  $x$  a  $y$ , které popisují pozici měření vektoru. Takový vektor může být popsán takto:

$$P = (\vartheta_1, \dots, \vartheta_N, x, y) \quad (10)$$

Nejdříve musí algoritmus seřadit hodnoty RSSI ve vektoru  $P$  sestupně. Nyní vytvoříme aktuálně naměřený vektor  $R$ , který obsahuje  $M$  hodnot RSSI z dostupných AP, kterých musí být stejné množství, jako u vektoru  $P$ :  $M = N$ .

$$R = (\varphi_1, \dots, \varphi_N) \quad (10)$$

Do vektoru  $R$  však nezapisujeme souřadnice  $x$ ,  $y$ . Pokud bude vektor  $R$  kratší (z pohledu RSSI hodnot) než  $P$ , dosadíme na zbylá místa hodnotu  $-100$  dBm respektive  $0$  mW a seřadíme ho sestupně. Pokud bude delší, nejprve provedeme sestupné řazení, a poté ho zkrátíme na stejnou délku, jakou má vektor  $P$ . Nyní jsou k dispozici dva stejně velké a stejně seřazené vektory  $P$ ,  $R$ . Pro ně vypočítáme Spearmanovu vzdálenost pomocí vztahu (11):

$$D = \sqrt{\sum_{k=1}^n (\vartheta_k - \varphi_k)^2} \quad (11)$$

$n$  - počet prvků ve vektorech  
 $k$  - pozice hodnoty ve vektoru.

Číslo  $D$  udává rozdílnost těchto dvou vektorů. Čím je menší, tím jsou vektory podobnější. Pokud tedy  $D=0$ , pak jsou vektory stejné. Po porovnání všech vektorů s aktuálním a jejich následným seřazením vzestupně, získáme vektor s nejmenším číslem  $D$ . To je tzv. *nejbližší soused* pro aktuálně naměřený vektor a tudíž vektor obsahující nejvíce pravděpodobnou polohu tagu. Pokud by se stalo, že máme větší počet shodných vektorů, je vhodné využít dodatečný filtr pro zpracovávání vektorů, ještě předtím, než budou použity pro výpočet své odlišnosti.

Metoda nejbližšího souseda NN (*nearest neighbor*) je základem pro další algoritmy a to konkrétně algoritmy kNN (*k nearest neighbors*) a WkNN (*weighted k nearest neighbors*).

Metoda kNN spočívá na principu výpočtu výsledné pozice z určitého počtu souřadnic z databáze. Jednoduše řečeno, pokud je  $k = 5$ , tak z pěti nejmenších rozdílových vektorů a tedy pěti hodnot  $x$  a  $y$ , vypočítáme průměr (12), který značí výslednou polohu [8, (2)].

$$(x, y) = \frac{1}{K} \times \sum_{i=1}^K (x_i, y_i) \quad (12)$$

$K$  - počet vektorů, ze kterých se určují souřadnice

$(x, y)$  - výsledné souřadnice

$(x_i, y_i)$  - souřadnice na dané pozici ve vektoru

Metoda WkNN je odlišná tím, že se počítá „váha“, s níž se poté jednotlivé souřadnice váhují. Ona váha je převrácená hodnota druhé mocniny vypočítané vzdálenosti (11) dvou vektorů. Výsledné souřadnice se poté určí dle vztahů (13), (14).

$$(x, y) = \sum_{i=1}^K (x_i, y_i) \times w_i \quad (13)$$

$$w_i = \frac{1}{D_i^2} / \sum_{j=1}^K \frac{1}{D_j^2} \quad (14)$$

$w_i$  - váha

$D$  - Spearmanova vzdálenost

### 1) **Technika RSSI** (Received Signal Strength Indicator)

- každé AP je definováno svým maximálním vysílacím výkonem. Pro AP využívající Wi-Fi standard je v Evropě maximální povolený výkon 100 mW [4]. Cílem je tedy změřit hodnotu síly signálu od AP. Protože síla signálu díky útlumu se vzdáleností klesá, můžeme takto získat vrstevnicový popis okolí AP, který popisuje místa se stejnou úrovní signálu. Hodnoty RSSI, jež vyjadřují velikost síly signálu v daném místě, můžeme změřit spektrálním analyzátozem, nebo můžeme využít mobilní, či počítačové programy. Tato technika tedy nevyužívá hodnoty získané pomocí AP, ale hodnoty získané z tagu. Ani tato technika však není dokonalá. Chyba je způsobena například a hlavně bezdrátovým adaptérem v tagu, který pochází často od různých výrobců a je tedy i jinak kalibrován. Proto je možné na stejném místě změřit několik různých hodnot RSSI. Hodnoty RSSI se také mohou lišit i v závislosti na denní době. V pracovní době je budova zaplněna větším počtem lidí a přístrojů, které způsobují rušení a útlum signálu. Protože tyto překážky nemusí být statické, dochází proto k větším odchylkám. Další chybu v hodnotě naměřené úrovně signálu způsobuje kolísání úrovně signálu vysílaného z AP. Toto kolísání může způsobovat několik faktorů, například anténa nebo firmware AP či jiné a objevuje se i v případech, kdy je AP nakonfigurováno na vysílání stále stejné úrovně signálu. Data získaná pomocí této techniky lze uložit do databáze a využít v metodě nejbližšího souseda.

## 4.6 Lokalizační systém Ekahau

Zde bude představen balíček [9] lokalizačního systému od firmy Ekahau. Tento balíček obsahuje softwarové i hardwarové prostředky, díky nimž může být na konkrétním případu objasněna funkce lokalizačních systémů. Lokalizace v tomto systému funguje na principu RTLS. Hlavní uplatnění tohoto řešení je mířeno zejména do nemocničních prostor. Zde může být využito pro lokalizaci a správu nemocničních zařízení, což napomáhá k jejich hledání nebo zabezpečení proti krádeži. Samozřejmě tato technologie může být také využita pro monitorování pohybu pacientů, případně zaměstnanců. U náramkových tagů, jež jsou vybaveny pohotovostním tlačítkem, je dokonce možné přivolat rychlou pomoc. Lokalizační technologie však nemusí nutně zastávat jen funkci monitorování pozice osob či předmětů. Namísto tagů se mohou využít různé bezdrátové senzory, například senzory teploty nebo vlhkosti.

Ekahau RTLS technologie je rozdělena na pět nezbytných součástí:

### 1) Software Ekahau Site Survey

- slouží pro návrh a optimalizaci WLAN sítě. Díky němu můžeme navrhnout bezdrátovou síť, která bude splňovat požadavky systému Ekahau, případně ho můžeme využít i pro návrh WLAN pro jiné využití. Pro kvalitní návrh jsou potřeba dobré mapové a jiné podklady prostoru, ve kterém bude síť umístěna. Po vložení této mapy do softwaru je potřeba do ní zadat měřítko a dále možné zdroje stínění (zdi, okna, dveře), u kterých následně definujeme parametry, jako tloušťka a materiál. Tyto informace slouží k výpočtu útlumu, který je potřeba pro simulaci pokrytí prostoru. Na takto „popsanou“ mapu je dále třeba přidat na námi zvolená umístění jednotlivá AP, kterým také můžeme nadefinovat specifické vlastnosti (standard IEEE 802.11 na kterém vysílají, kanál vysílání, vyzařovaný výkon a mnoho dalšího). Díky všem těmto informacím můžeme najít slabá místa sítě (případně překryvy kanálů) nebo pravděpodobná slepá, signálem nepokrytá místa.

### 2) Tvorba site survey

- Jde o jakýsi „průzkum“ objektu s WLAN sítí (nyní již fyzicky nainstalovanou). Tento „průzkum“ se provádí velice jednoduchým způsobem. Administrátor (případně kdokoliv jiný) si vezme notebook s nainstalovaným SW a připravenou mapou a jen kliká do mapy, kudy jde a jakým směrem a tím na mapu přidává body. Mezi těmito body jsou v reálném čase zaznamenávána data o síle signálu z vybraných AP, která jsou poté interpolována a kromě možnosti dodatečné optimalizace sítě na základě reálně naměřených hodnot tvoří také celkem kvalitní základ pro následnou lokalizaci. Kvalita velmi záleží na pečlivosti, s jakou byl „průzkum“ vytvořen. Systém Ekahau je dle nedostatečných dostupných materiálů a našeho testování (hlavně způsob sběru dat pro lokalizaci) založen nejspíše na principu *fingerprintingu*, ke kterému je doplněna částečná predikce. Po dokončení site survey je ještě potřeba definovat otevřené prostory (chodby, místnosti), která se v prostoru nacházejí. Díky tomu může program dopočítat další pravděpodobné hodnoty potřebné k lokalizaci.

### 3) Lokalizační engine RTLS Controller

- Do tohoto enginu je načtena proměřená mapa z předchozího bodu a zde se aktivuje. RTLS Controller je systémová služba běžící na pozadí. Tuto službu lze nakonfigurovat buď konfiguračním SW, nebo pomocí webového rozhraní. Služba může (a měla by být) spuštěna na síťovém serveru. Je však možné i využít jakýkoliv jiný dostupný počítač.

### 4) Ekahau Vision

- Opět služba, běžící na pozadí, stejně jako RTLS Controller. Má na starosti dohled a monitoring připojených uživatelů s tagy. Může být spuštěna na stejném stroji jako RTLS Controller a je také přístupná přes webové rozhraní. Jejím hlavním úkolem je zobrazení zjištěných tagů na mapě, která se aktivovala v RTLS Controlleru, v „reálném“ čase. Tyto zobrazené body můžeme seskupovat, vytvářet k nim záznamy, případně na tag, který to podporuje, zaslat textovou zprávu.

### 5) Tagy Ekahau

- K dispozici jsou 3 základní druhy tagů, pracující na IEEE 802.11b/g/n:
  - a) Poziční tag T301W
    - o pro osoby, *Obr. 4.3*. Může být dovybaven náramkem pro aplikaci na zápěstí pacienta. Data o síle signálu zasílá buď po stisknutí tlačítka, nebo v pravidelných intervalech. Tlačítko také vykonává funkci žádost o asistenci nemocničního personálu, pokud je potřeba. Citlivost lokalizovaného místa je na úrovni pokoje či místa v pokoji.



*Obr. 4.3: T301W [15]*

#### b) A4 Asset

- o pro lokalizaci zařízení, *Obr. 4.4*. Je o něco rozměrnější, než předchozí, díky vestavěným bateriím s životností kolem 5 let. Na přední straně jsou 2 tlačítka a infračervený senzor, sloužící k velmi přesné lokalizaci. Na zadní straně je umístěn senzor, který se aktivuje při oddálení od podložky. Ten slouží jako poplachový spínač.



*Obr. 4.4: A4 Asset [15]*

c) B4 Badge

- Obr. 4.5, umožňuje přijímání textových zpráv.



Obr. 4.5: B4 Badge [15]

d) Tag na platformě Windows, Windows Phone

- Speciální aplikaci *Positioning Client* který si mohou uživatelé výše zmíněných operačních systémů nainstalovat a mohou být zařazeni na seznam tagů.

Systém Ekahau je velice vydařený, nicméně ani po dlouhém úsilí se nám ho s kolegou Cihlářem nepodařilo korektně nastavit a použít. Hlavním nedostatkem tohoto systému je kompatibilita pouze s některými bezdrátovými adaptéry v počítačích, které jsou buď potřeba k aktivaci tagů nebo pro monitoring aplikace *Positioning Client* na platformě Windows. Také velmi omezená funkčnost na OS Windows 8 byla zarážející, a proto jsme se pokusili z tohoto systému využít jeho užitečné funkce a implementovat je do našeho vlastního řešení, které bude popsáno v následujících kapitolách.

## 5. Tvorba aplikace pro OS Android

### 5.1 Specifikace OS Android, jeho vývoj a znaky

#### 5.1.1 Historie OS Android

Operační systém Android vznikl na bázi linuxového jádra a byl primárně navržen pro zařízení s dotykovou obrazovkou (hlavně pro smartphony a tablety, ale dnes je možné ho najít i v hodinkách, brýlích, televizích, fotoaparátech a ve spoustě dalších zařízeních). Nejdříve byl vyvíjen firmou Android, Inc., kterou částečně financovala společnost Google. Ta také v roce 2005 firmu koupila a vývoj Androidu si vzala na starost. Android je od roku 2007 Open Source projekt, tedy projekt, do jehož vývoje může zasahovat kdokoliv, případně tento systém může kdokoliv libovolně upravovat a vyvíjet dle svých představ.

Od roku 2010, kdy byla vydána první více využívaná verze OS Android, konkrétně 2.2 Froyo, vzniklo mnoho novějších verzí, jejichž shrnutí je v následující tabulce (Obr. 5.1):

Version	Codename	API	Distribution
2.2	Froyo	8	1.0%
2.3.3 - 2.3.7	Gingerbread	10	16.2%
3.2	Honeycomb	13	0.1%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	13.4%
4.1.x	Jelly Bean	16	33.5%
4.2.x		17	18.8%
4.3		18	8.5%
4.4	KitKat	19	8.5%

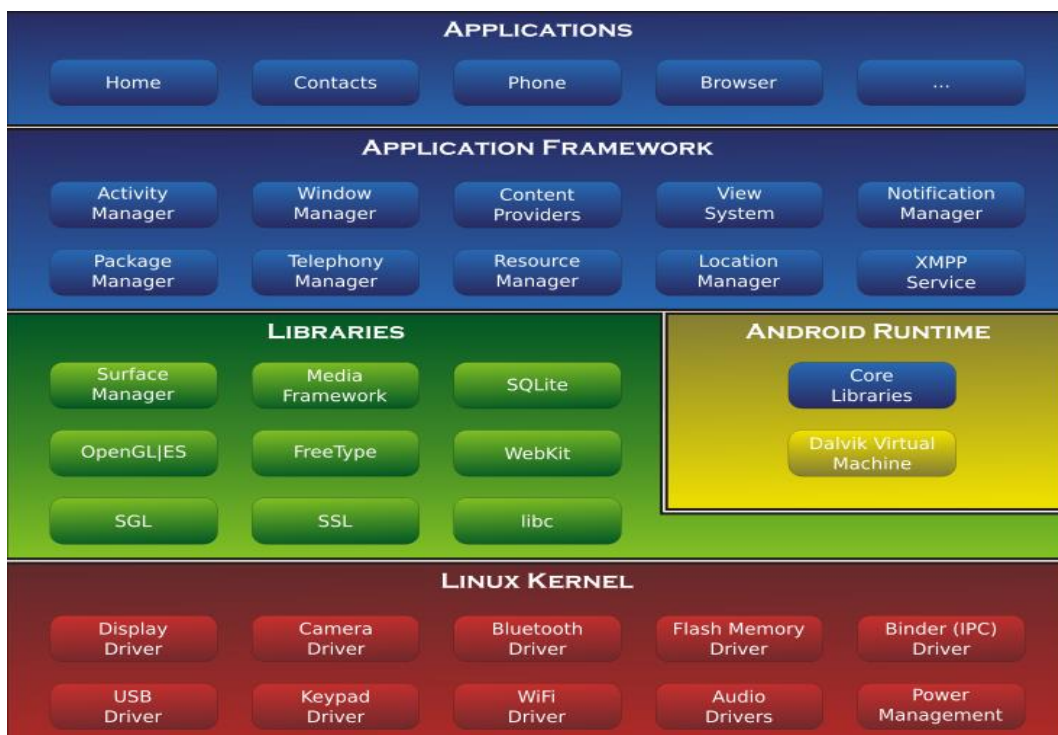
*Data collected during a 7-day period ending on May 1, 2014.  
Any versions with less than 0.1% distribution are not shown.*

Obr. 5.1: Vývoj OS Android, jeho verze

Dle nejnovějších informací a průzkumů je OS Android aktuálně nejpoužívanější OS na mobilních zařízeních. I to, jak jsem již předeslal dříve, je důvodem, proč byl k tvorbě aplikace vybrán tento OS.

#### 5.1.2 Struktura OS Android

Jak již bylo zmíněno výše, OS Android je založen na linuxovém jádře. Na tom jsou dále postaveny bloky servisního SW, který poskytuje přístup k jádru a hardware, knihovny a rozhraní pro programování aplikací (API). Tyto součásti jsou napsány v programovacím jazyku C. V OS Android se dále nachází aplikační software, běžící na aplikačním frameworku. Poslední nezbytnou součástí OS Android je tzv. Dalvik Virtual Machine, který zajišťuje překlad z kódu aplikací psaných v jazyku Java. Graficky vypadá struktura OS Android takto:



Obr. 5.2: Struktura OS Android

## 5.2 Programování aplikací pro OS Android

Pro vývoj aplikací pro OS Android je potřeba stáhnout si z oficiálních stránek vývojářů pro Android <http://developer.android.com/index.html> balík aplikací ADT (Android Developer Tools), který obsahuje jednak vývojové prostředí Eclipse IDE (modifikované přímo pro Android) a také SDK (Software Development Kit), což je jakýsi správce balíků knihoven a ovladačů nutných pro vývoj aplikace pro určitou verzi OS Android.

Po stažení ADT je potřeba nejdříve spustit SDK, který nám umožní vybrat si námi požadovanou verzi knihoven pro Android a případně stáhne chybějící ovladače jako Google USB driver případně aktualizace již stažených knihoven. Ještě, než ale aplikaci začnu popisovat, je potřeba se zamyslet nad tím, na jakých principech bude fungovat a jak bude postavena. Samozřejmě, že toto lze později v Eclipse kdykoliv změnit, nicméně je potřeba nějaký základní návrh, od kterého se vývoj odrazí. Poté již můžeme spustit Eclipse a pustit se do navrhování aplikace.

### 5.3 Návrh aplikace lokalizačního systému

Hlavním smyslem navrhované aplikace bude lokalizace osob (případně zařízení) v prostorách budovy. To tedy znamená, že požadovaná přesnost lokalizace je nejméně na úrovni pokoje, případně lepší, pokud to podmínky dovolí. Navrhovaný lokalizační systém by tedy měl mít následující vlastnosti:

- a) Systém nepotřebuje žádné speciální tagy, je tedy možné využít dostupná zařízení s OS Android
- b) Systém je pokud možno co nejpřesnější a nejrychlejší
- c) Použití lokalizační aplikace je pokud možno intuitivní jak pro administrátory, tak pro koncové uživatele
- d) Komunikace se síťovým serverem musí být co nejrychlejší a nejefektivnější, aby byly výsledky dostupné pokud možno v reálném čase
- e) Aplikace by měla co nejméně zatěžovat mobilní zařízení (a to jak spotřebou baterie, tak výpočetní kapacitou a kapacitou v úložišti dat)
- f) Využití v jakýchkoliv prostorách (nejen nemocnice, vězení a sklady, ale i další jako kancelářské budovy, nákupní centra a podobně)

Celý lokalizační systém musí samozřejmě vycházet z výše zmíněných principů lokalizace a pro jeho provoz budou potřeba všechny 3 součásti (tag, lokalizační engine a dohledový software). Vzhledem k tomuto bude výsledná aplikace rozdělena do určitých částí. Jak, bude ještě vysvětleno později.

Protože mohu při návrhu aplikace vycházet z již představeného SW Ekahau, rád bych při lokalizaci využil metodu *fingerprinting*. Ta spočívá v tom, že administrátor nasbírá v terénu (prostor pro lokalizaci) dostatečné množství dat o silách signálu v konkrétních místech, které se poté uloží do databáze a ta se uloží na server. Odtud poté bude přístupná všem zařízením, která se nechávají lokalizovat. Tato databáze obsahuje samozřejmě také minimálně souřadnice X a Y, které jsou nutné pro zobrazení bodu na mapě.

Podívejme se nyní na návrh samotné aplikace. Nejdříve tedy musíme vytvořit aplikaci, která bude schopná měřit sílu signálu od různých AP vysílající stejný název sítě a ukládat tyto hodnoty společně se souřadnicemi do databáze. Tato aplikace bude sloužit administrátorovi, který bude mít za úkol provést *site survey*. Tato databáze se poté uloží na FTP server, který bude dostupný ze stejné sítě, jakou administrátor skenoval. *Administrátorská aplikace*, jak tuto budu dále nazývat, bude i dohledový systém, který bude zobrazovat aktuálně připojené uživatele, kteří se nechávají lokalizovat.

Další aplikace, kterou je potřeba vytvořit, je aplikace pro uživatele. Ta bude mít za úkol stažení databáze s podklady pro výpočty z FTP serveru. Bude zobrazovat mapu s aktuální polohou uživatele, která se bude v daných intervalech obnovovat. Také bude zasílat informace o poloze uživatele do dohledového systému. Tuto aplikaci budu nazývat *Uživatelská*.



V případě, že bychom chtěli lokalizovat zařízení, je potřeba využít aplikaci, která vyžaduje pokud možno minimální zásah administrátora. Proto vytvořím „aplikaci na pozadí“, která poběží neustále, pokud bude zařízení s Androidem v provozu. Stále ale bude potřeba, aby ji administrátor nejprve spustil. Tato aplikace se bude nazývat *Lokalizační služba*.

Dále je potřeba aplikace pro vlastní výpočty, tedy lokalizační engine. Z důvodu jednoduchosti řešení jsem se rozhodl, po konzultaci s kolegou Cihlářem, který mi s vývojem částí aplikací pomáhal, že tento engine bude zabudovaný již přímo ve výše zmíněných aplikacích a nepoběží proto na serveru. Je to z toho důvodu, že by bylo nutné vyvíjet ještě aplikaci speciálně pro server, která by musela být zabezpečena, aby neohrožovala provoz serveru.

## 5.4 Tvorba aplikace pro OS Android

Základním prvkem všech aplikací pro OS Android je „Aktivita“ případně „Servis“. Vývojové prostředí Eclipse nám nabídne vytvoření aktivity ihned po zvolení vytvoření nové aplikace. Vytvářením nové aplikace a aktivity nás provede jednoduchý průvodce, který se nás vyptá na vše potřebné, jako na název aplikace a název projektu, pro jakou verzi OS Android bude minimálně aplikace funkční, pro jakou verzi bude vytvářené a na jaké verzi ji budeme kompilovat. Dále nám nabídne uložení aplikace do pracovní složky (Workspace) a také ikonky, které budou použity po instalaci aplikace v zařízení. Dále máme možnost vybrat si jeden ze tří základních typů aktivit a pojmenovat vybraný. Jak průvodce vypadá, je na obrázcích *Obr. 9.1 – Obr. 9.4* v příloze.

Jakmile je průvodce za námi, Eclipse vytvoří projekt dle námi zadaných specifikací a můžeme se pustit do vlastního tvoření. Nyní je potřeba pohrát si s hlavní aktivitou. Ta má samozřejmě i grafickou nadstavbu (layout), kterou můžeme upravovat buď pomocí XML souboru, nebo přímo v grafickém editoru. Zde je možné přidávat tlačítka, okna pro zobrazení obsahů různých typů a podobně. Jakmile jsme hotovi a spokojeni s grafickým návrhem, přijde na řadu programování funkcí aplikace. Zde již vše funguje jako při programování jakéhokoliv jiného programu. Vytváříme tedy jednotlivé třídy, metody, podmínky a proměnné dle libosti a naší potřeby.

Aby aplikaci bylo možné na zařízení nainstalovat, je potřeba si na konci vývoje (nejlépe ale již průběžně při vývoji) zaznamenat, jaké funkce telefonu bude aplikace využívat. Například jde o tyto:

- a) Čtení/zápis do interní paměti
- b) Čtení/zápis na paměťovou kartu
- c) Přístup k bezdrátovému adaptéru
- d) Měnit stav bezdrátového adaptéru
- e) Přístup k internetu

Veškeré tyto funkce musí být shrnuty v témže XML souboru, v jakém je definován grafický layout, tedy *AndroidManifest.xml*. Příklad této definice je na *Obr. 5.3*.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.rtls"
4     android:versionCode="1"
5     android:versionName="1.0" >
6
7     <uses-sdk
8         android:minSdkVersion="11"
9         android:targetSdkVersion="19" />
10
11     <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
12     <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
13     <uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />
14     <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
15     <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
16     <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
17     <uses-permission android:name="android.permission.WRITE_INTERNAL_STORAGE" />
18     <uses-permission android:name="android.permission.READ_INTERNAL_STORAGE" />
19     <uses-permission android:name="android.permission.WAKE_LOCK" />
20     <uses-permission android:name="android.permission.INTERNET" />
21     <action android:name="android.net.wifi.WIFI_STATE_CHANGED"/>
22     <action android:name="android.net.wifi.SCAN_RESULTS"/>
23
24     <application
25         android:allowBackup="true"

```

Obr. 5.3: AndroidManifest.xml – definice oprávnění aplikace

## 5.5 Vývoj Administrátorské aplikace

Jak zhruba by měla Administrátorská aplikace vypadat, jsem pověděl již dříve. Nyní bych se tedy rád podíval na to, jakým způsobem vznikala, na jaké potíže jsme s kolegou Cihlářem při programování narazili, jaké má omezení a konečné funkce. Všechny aplikace byly vytvářeny a testovány na telefonu Samsung Galaxy S2 GT-I9100.

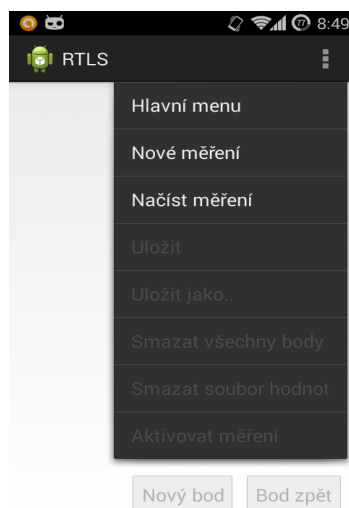
Nejdříve tedy vytvoříme projekt. Nazveme ho RTLS a stejně tak i hlavní aktivitu (RTLS.java). Nyní je potřeba si uvědomit, jak chceme, aby vypadalo hlavní menu aplikace a jaké všechny funkce by měla aplikace mít. Musíme proto vytvořit tlačítko Nastavení, které bude sloužit pro nastavování parametrů aplikace – změna jazyka (čeština/angličtina), konfigurace pro připojení na FTP server, případně další. Dále bude potřeba tlačítko, díky kterému budeme mít možnost provést Site Survey. To nazveme Proměření prostoru. Dále jistě potřebujeme dohledovou část – tlačítko Dohled. A na závěr, pro otestování funkčnosti provedeného survey administrátorem, bude potřeba také tlačítko Spustit lokalizaci. Obrázek hlavního menu je v příloze na Obr. 9.6. Při spuštění aplikace se vytvoří v paměti telefonu adresářová struktura na Obr. 5.4.

rtls	2014-05-13	11:49	drwxrwx---
Config	2014-05-14	10:35	drwxrwx---
Supervision	2014-05-13	15:10	drwxrwx---
client	2014-05-13	14:09	drwxrwx---
databases	2014-05-14	16:36	drwxrwx---
localization	2014-05-14	16:36	drwxrwx---
maps	2014-05-14	10:51	drwxrwx---
scale	2014-05-13	11:46	drwxrwx---
surveys	2014-05-14	20:03	drwxrwx---

Obr. 5.4: Adresářová struktura v paměti telefonu

Nyní, když již známe rozložení hlavního menu, můžeme se pustit do vytváření kódu, který bude jednotlivá tlačítka obsluhovat. Vzhledem k tomu, že velkou část Administrátorské aplikace popsal ve své práci kolega Cihlář, představím ji jen v rychlosti a spíše se zaměřím na změny a doplnění, které byly potřeba provést pro souhrn aplikace s FTP serverem. V práci kolegy totiž aplikace ještě neumí se serverem komunikovat a funguje pouze lokálně.

Srdcem aplikace je třída nazvaná *Survey.java*. Ta se spustí po kliknutí na tlačítko Proměření prostoru. Samozřejmě, že i tato třída má grafický layout. Nechybí zde rolovací menu, které se objeví po stisknutí funkčního tlačítka telefonu, případně ve vyšších verzích OS Android je možné toto menu vyvolat pomocí softwarového tlačítka v horní části aplikace (*Obr. 5.5*). Dále zde je komponenta *ImageView*, do které se načítá administrátorem zvolená mapa prostoru a tlačítka *Nový bod* a *Bod zpět*. Díky rolovacímu menu máme možnost vytvořit nový survey, případně načíst a upravit starší. Po zvolení jedné z těchto možností se aktivují výše zmíněná dvě tlačítka. Tlačítko *Nový bod* zajišťuje přidání nového bodu na mapu. Poloha bodu záleží na administrátorovi, nicméně pro správnou funkci aplikace je vhodné zvolit, pokud možno co nejpřesněji, aktuální pozici. Po umístění bodu následuje 16s dlouhý interval, kdy se skenuje okolí dle zadaného SSID sítě a zjišťují se síly signálu. Tento interval je dlouhý 16s z toho důvodu, že bezdrátový adaptér v zařízení je kalibrován na obnovení přijatých výsledků každých 6s. Abychom tedy naměřili více hodnot a poté z nich průměrem určili hodnotu nejvíce pravděpodobnou, zvolili jsme interval delší, než 3 obnovení hodnot. Po dokončení skenu, který zajišťují třídy *Scan.java* a *Multiscan.java*, máme možnost buď poslední umístěný bod z mapy smazat, nebo pokračovat umístěním dalšího bodu. Když je administrátor se survey spokojený, má možnost si ho pomocí rolovacího menu *Uložit* nebo *Uložit jako*. Tyto tlačítka se chovají stejně jako v kterékoliv jiné aplikaci. Volba *uložit* uloží survey pod názvem, který si administrátor zadal při vytváření nového survey, druhá volba mu umožní název změnit. Další možnosti rolovacího menu jsou *Smazat všechny body* a *Smazat soubor hodnot*. Jak jejich názvy říkají, první smaže všechny body na načtené mapě, druhý smaže databázi naměřených údajů. Další tlačítko v rolovacím menu je *Aktivovat měření*. Tímto dá administrátor pokyn aplikaci, že je survey připravený pro lokalizaci a název survey se spolu s názvem mapy a SSID uloží do speciálního souboru ve složce */rtls/localization/*. Tohoto souboru poté využívají ostatní části aplikace. Všechny úkony se soubory má v popisu práce třída *IOFile.java*. Poslední volbou v rolovacím menu je návrat do hlavního menu.



Obr. 5.5: Rolovací menu

Pod tlačítkem Spustit lokalizaci v hlavním menu se skrývá třída *Localization.java*. Ta má také svůj grafický layout, který obsahuje opět rolovací menu, *ImageView* pro načtení mapy a tzv. „*CheckBox*“ s názvem *Spustit lokalizování*, které je až do načtení mapy deaktivované. Rolovací menu opět obsahuje možnost vrátit se do hlavní nabídky a dále volbu Načíst mapu. Tato spustí pomocí *IOFile.java* načtení mapy, jejíž název obsahuje dříve vytvořený soubor s aktivovaným survey, a k ní přidruženou databázi. Poté se aktivuje tlačítko *Spustit lokalizování*, které, po zapnutí, každé 3 vteřiny obnoví polohu uživatele dle právě naskenovaných hodnot síly signálu.

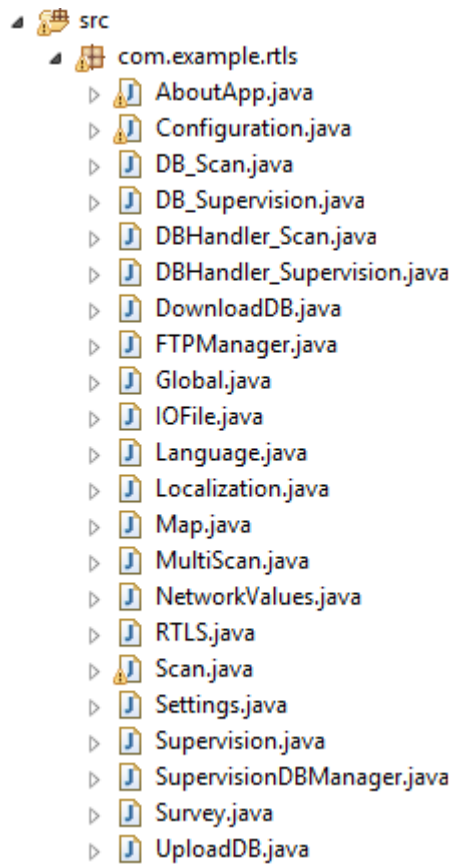
Tlačítko Nastavení v hlavním menu nám umožní podívat se do třídy s názvem *Settings.java*. Ta obsahuje další 4 tlačítka, konkrétně Konfigurace, které slouží výhradně k vývojářským účelům a není možné ho zmáčkнуть. Dále Jazyk, které umožní přepínání mezi angličtinou a češtinou v celé aplikaci. Další tlačítko je tlačítko s názvem Nastavení sítě. To zavolá třídu *ServerValues.java*. Ta nám umožní zadat nastavení nutné k přihlášení na FTP server (IP, port, uživatelské jméno a heslo) a SSID sítě, která se má skenovat. Poslední zmíněné je zde proto, že skenování probíhá pouze, když je zařízení odpojeno od sítě, aby bylo co nejpřesnější. Samozřejmostí jsou tlačítka na smazání hodnot a jejich uložení do souboru *configuration.dat* v */rtls/config/*, který je volně editovatelný například v PC pomocí textového editoru, případně přímo v telefonu a lze tedy údaje o FTP serveru změnit i bez nutnosti zapínání aplikace samotné. Posledním tlačítkem v Nastavení je tlačítko O aplikaci.

V aplikaci doplněné o souhrn s FTP serverem se v hlavním menu nachází ještě tlačítko Dohled. To spustí třídu *Supervision.java*. V grafickém layoutu má tato třída opět *ImageView* pro načtení mapy, tlačítko *Spustit sledování* a rolovací menu. To obsahuje kromě možnosti vrátit se do hlavní nabídky možnost Načíst mapu. Ta opět zajistí načtení mapy dle výběru administrátora ze složky */rtls/supervision/maps/*. Tyto mapy, společně s databázemi hodnot se automaticky synchronizují při spuštění třídy *Supervision.java* s FTP serverem pomocí *SupervisionDBHandler.java*. Po načtení mapy se aktivuje tlačítko *Spustit sledování*, které každých 6 sekund obnoví databáze z FTP a zobrazí polohy uživatelů na mapě. Překážkou pro aplikaci není ani více různých map.

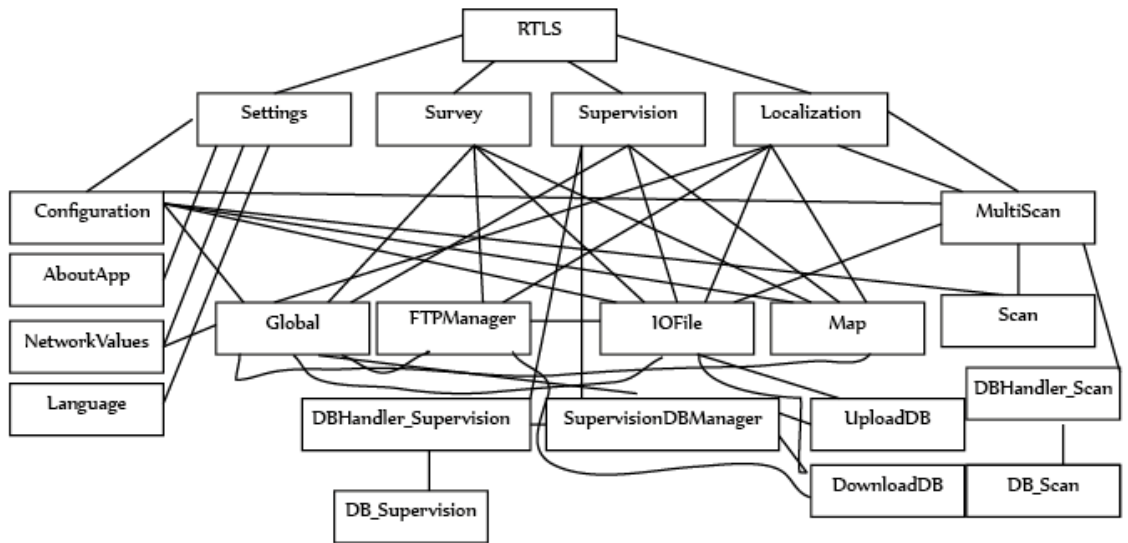
Na mapě se vždy zobrazí jen uživatelé, kteří jsou na stejné mapě. Tedy pokud máme k dispozici ve složkách mapu A s 5 uživateli (5 databází) a zároveň mapu B s 8 uživateli (8 databází), po načtení mapy A se na ní zobrazí pouze 5 teček. Tyto tečky je možné rozlišit pomocí MAC adres, které se u nich zobrazují, takže administrátor ví, kde je které zařízení.

## 5.6 Popis tříd Administrátorské aplikace

Na následujícím obrázku (*Obr. 5.6*) je souhrn tříd aplikace, které se využívají. Na dalším obrázku (*Obr. 5.7*) jsou vidět jejich vzájemné vztahy. V předchozím popisu aplikace jsem již některé třídy a jejich funkci zmínil, nyní je podrobím pozornější prohlídce.



*Obr. 5.6: Třídy Administrátorské aplikace*



Obr. 5.7: Vzájemné vztahy tříd Administrátorské aplikace

### 5.6.1 Třída AboutApp.java

Tato třída má za úkol zobrazit v grafickém layoutu pouze informace o aplikaci. Nenacházejí se zde žádné speciální funkce.

### 5.6.2 Třída Configuration.java

Tato třída se volá po stisknutí tlačítka Nastavení->Konfigurace. Slouží pouze pro účely vývojáře, z pohledu administrátora není důležitá. Nicméně obsahuje testovací funkce pro načtení mapy a budu ji využívat pro závěrečné měření kolísání signálu. Také umožňuje zobrazení aktuálních naměřených hodnot, aktualizovaných každých několik vteřin.

### 5.6.3 Třída DB\_Scan.java

V původní verzi aplikace bez FTP serveru se nazývala *Database.java*. Tato třída má za úkol získávání hodnot z databáze typu SQLite určené pro skenování okolí a jejich ukládání do ní. Jsou zde metody (getter a setter), které toto umožňují. Dále jsou zde definovány proměnné, které se do databáze mohou uložit a to konkrétně proměnná typu Double (Level 1-4, které obsahují síly signálů), dále proměnná ID typu Long (číslování řádku databáze), proměnné typu String (obsahující MAC 1-4, název – název sítě a MapName – název mapy) a naposledy souřadnice X a Y v proměnných typu Float.

### 5.6.4 Třída DB\_Supervision.java

Obdobně jako *DB\_Scan.java*, i tato třída má za úkol ukládání hodnot do databáze a jejich načítání. Jediný rozdíl je v proměnných a použití výsledné databáze. Tato obsahuje proměnné typu String (MapName, název, MAC 1-4 a MacDev, což je MAC adresa zařízení, které databázi vytvořilo). Déle proměnnou ID (opět Long) a pozici zařízení X a Y, opět Float jako v přechodí třídě.

### 5.6.5 Třída DBHandler\_Scan.java

Tato třída má za úkol vytvoření vlastní databáze s hodnotami skenu. Jejich ukládání a načítání se provádí přes *DB\_Scan.java*, ale funkce, které toto zajišťují, se volají zde. Třída nám dále definuje sloupce databáze. Obsahuje funkce jako *pridejZaznam()*, která přidá řádek s hodnotami do databáze nebo *getMapName()*, která získá název mapy dle zadaných specifikací pomocí SQL Query (dotaz v jazyku SQL, pomocí kterého jsou databáze ovládány). Všechny využití databáze jsou typu SQLite. Třída nám tedy umožní:

- a) Vytvořit databázi dle našich potřeb
- b) Přidat záznam (1 řádek) s hodnotami
- c) Získat řádek hodnot dle ID
- d) Získat souřadnici X dle síly signálu
- e) Získat souřadnici Y dle síly signálu
- f) Získat všechny řádky se záznamy do seznamu typu ArrayList
- g) Získat počet řádků
- h) Updatovat řádek
- i) Smazat řádek
- j) Vypočítat průměrnou sílu signálu v daném místě dle uložených hodnot a vrátit nejbližší souřadnice X a Y (pomocí WkNN metody)
- k) Získat název mapy a SSID
- l) Smazat všechny záznamy

Také zde definujeme umístění databáze, případně její název.

### 5.6.6 Třída DBHandler\_Supervision.java

Opět velmi podobná předchozí třídě. Obsahuje pouze navíc funkce, která dokáže získat MAC adresu zařízení a neobsahuje funkci pro výpočet průměrné síly signálu.

### 5.6.7 Třída DownloadDB.java

Od předchozích tříd se liší tím, že je typu *AsyncTask*, tedy asynchronní úloha. Používáme ji proto, že operace, které vyžadují práci po síti, není možné spustit v hlavní úloze. Třída typu *AsyncTask* se skládá ze 3 základních metod a to *onPreExecute* (ta provádí úlohy před spuštěním hlavního kódu, například zobrazí *progressBar*), *doInBackground* (obsahuje hlavní kód a funkce) a *onPostExecute* (volá se po dokončení hlavního kódu v *doInBackground*, provádí například ukončení *progressBaru*). Tato třída, jak již z názvu vyplývá, má za úkol stáhnout z FTP serveru konkrétní databázi definovanou jejím názvem, což je vstupní proměnná této úlohy. Přihlásí se na FTP server, najde v daném adresáři databázi, uloží ji do vnitřního úložiště zařízení a z FTP se odhlásí.

### 5.6.8 Třída FTPManager.java

Opět třída *AsyncTask*. Tato má za úkol při spuštění *Survey.java* provést aktualizaci databází, měření a mapových podkladů na FTP serveru a v telefonu. Dále provádí synchronizaci aktivních měření dle SSID připojené sítě a tyto informace ukládá jak do adresáře na FTP, tak do telefonu, aby vždy byly k dispozici aktuální aktivní měření.

### 5.6.9 Třída Global.java

Na první pohled neobsáhlá třída, bez grafického layoutu, nicméně velmi důležitá. Do této třídy se ukládají všechny proměnné, které jsou potřeba v různých částech aplikace. Je to takové skladiště informací. Obsahuje například jména načtených map nebo údaje ze souboru *Configuration.dat* pro přihlášení k FTP.

### 5.6.10 Třída IOFile.java

Tato třída je ze všech nejobsáhlejší. Obsahuje lehce přes 900 řádků kódu a je používaná snad ve všech ostatních třídách. Mimo jiné nám umožňuje:

- a) Definici velkého množství proměnných, které nejsou potřeba globálně, ale některé jiné třídy je mohou využívat
- b) Volání *SaveDialogů* a *LoadDialogů*
- c) Ukládání a načítání souborů, jejich čtení a zápis, mazání souborů
- d) Načítání mapy
- e) Zobrazování bodů na mapě
- f) Ukládání hodnot ze souboru do databáze a načítání těchto hodnot zpět
- g) Upload a download databází na FTP
- h) Ověření velikosti mapy, kterou chce administrátor načíst

K podrobnějšímu prostudování je potřeba ponořit se přímo do kódu aplikace.

### 5.6.11 Třída Language.java

Jak již název vypovídá, stará se o změnu jazyka v aplikaci.

### 5.6.12 Třída Localization.java

Tato třída má za úkol volání synchronizace s FTP server (*FTPManager.java*), konkrétně synchronizaci aktivních survey. Dále přes třídu *IOFile.java* načítá mapu a pomocí vlastní funkce zobrazuje na mapě body v časovém intervalu 6s dle hodnot vytvořených třídou *Multiscan.java*.

### 5.6.13 Třída Map.java

Zde se provádí veškerá práce s mapu. Její inicializování, centrování, zoom, posouvání a podobné funkce.

### 5.6.14 Třída Multiscan.java

Velmi důležitá třída. Je to opět *AsyncTask*. Zajišťuje několikanásobné volání třídy *Scan.java*, kde získává potřebná data, ze kterých poté počítá kompletní síly signálu pomocí metody *makeData()*. Tyto výsledky poté uloží do souboru, který dále zpracovávají další třídy dle potřeby.

### 5.6.15 Třída NetworkValues.java

Umožňuje zadat nastavení nutné pro přihlášení na FTP server a SSID skenované sítě. Vložené hodnoty jsou patřičně ošetřeny, například taky, aby nebylo možné do IP adresy zadat písmena nebo jiné znaky než čísla a tečky. Tyto hodnoty jsou poté uloženy do souboru *configuration.dat*



### 5.6.16 Třída RTLS.java

Tato třída obsahuje hlavní aktivitu celého programu a tudíž i hlavní menu. Zajišťuje vytvoření správné adresářové struktury pro ukládání souborů a obsluhuje jednotlivá tlačítka v hlavním menu. Také při startu aplikace načte do globálních proměnných nastavení ze souboru `configuration.dat`, pokud soubor existuje. Pokud ne, spustí varování, které administrátora donutí tyto informace nejdříve zadat.

### 5.6.17 Třída Scan.java

Třída provádí vlastní skenování okolí. Výsledky ukládá do polí, která jsou později seřazena, aby mohla sloužit pro výpočty prováděné v *Multiscan.java*. Ze třídy *Multiscan.java* je volána 8x, vždy po 2 vteřinách. Samotný sken se provádí pomocí bezdrátového adaptéru zařízení. Ten odchyťává tzv. Beacons vysílané z AP, a z nich vypočítává sílu signálu.

### 5.6.18 Třída Settings.java

Obsluhuje tlačítka v menu Nastavení a spouští k nim přidružené třídy.

### 5.6.19 Třída Supervision.java

Má za úkol inicializovat stažení podkladů nutných pro dohledový systém z FTP serveru pomocí třídy *SupervisionDBManager.java*. Stejně jako *Localization.java* obsahuje časovač, který každých 6s obnoví zobrazené body. Předtím samozřejmě stáhne aktuální databáze.

### 5.6.20 Třída SupervisionDBManager.java

AsyncTask třída zajišťující stažení databázi a map nutných pro dohledovou část aplikace. Spouští se ze třídy *Supervision.java* při spuštění a pak každých 6s po zapnutí tlačítka. Mapy a databáze ukládá do složek v `/rtls/supervision/` dle typu souboru. Databáze jsou stahovány pouze tehdy, pokud na FTP serveru není tzv. „příznak“ souboru, který značí, že tato databáze ještě není kompletně nahrána na FTP.

### 5.6.21 Třída Survey.java

Opět jedna z velmi důležitých tříd. Tato obsluhuje grafický layout nutný pro úspěšné proměření prostoru. Obsluhuje tedy tlačítka pro umístění bodů na mapu, rolovací menu pro výběr požadované funkce a dále volá metody z *IOFile.java* pro ukládání souborů, načítání mapy a podobné. Také zobrazuje body na mapě a volá *Multiscan.java*. Součástí je i dialogové okno zajišťující případné uložení souboru, když se administrátor vrací do hlavního menu.

### 5.6.22 Třída UploadDB.java

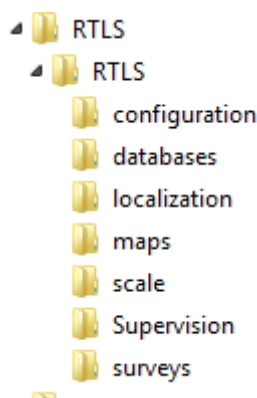
Poslední ze seznamu tříd. Tato je opět AsyncTask a má jediný úkol. Tím je uložení databáze a k ní připojený Survey soubor na FTP server. Tyto soubory poté smaže z paměti telefonu.

Je potřeba zmínit, že FTP soubor je pro funkci aplikace nezbytný. Soubory s informacemi jsou uloženy v paměti telefonu pouze po nezbytně nutnou dobu a následně jsou odeslány na FTP server a z telefonu smazány!

## 5.7 Podmínky pro správné použití Administrátorské aplikace

Pro správný chod Administrátorské aplikace je nutné mít v aktuálně měřené síti připojený FTP server s adresářovou strukturou dle *Obr. 5.8*. Dále je nutné, aby mapa, která se v aplikaci načítá, byla typu „.jpg“ a měla max. rozlišení 2000 x 1000 px (tedy 2MPx). Mapa ve větším rozlišení způsobuje pády aplikace. Dále je potřeba přístroj s bezdrátovým adaptérem a OS Android alespoň 3.0, nejlépe však OS Android 4.1. Pro optimální funkci je také vhodné nastavit použitá AP na vyzařování konstantního výkonu, pokud to podporují. Aplikace pro instalaci a svůj chod vyžaduje následující oprávnění:

- a) android.permission.ACCESS\_WIFI\_STATE – Přístup ke stavu bezdrátového adaptéru
- b) android.permission.CHANGE\_WIFI\_STATE – Měnit stav bezdrátového adaptéru
- c) android.permission.ACCESS\_NETWORK\_STATE – Získat stav připojené sítě
- d) android.permission.CHANGE\_NETWORK\_STATE – Měnit stav připojené sítě
- e) android.permission.WRITE\_EXTERNAL\_STORAGE – Zapisovat do externího úložiště
- f) android.permission.READ\_EXTERNAL\_STORAGE – Číst z externího úložiště
- g) android.permission.WRITE\_INTERNAL\_STORAGE – Zapisovat do interního úložiště
- h) android.permission.READ\_INTERNAL\_STORAGE – Číst z interního úložiště
- i) android.permission.WAKE\_LOCK – Používat zámek bezdrátového adaptéru
- j) android.permission.INTERNET – Svolení s přístupem na internet



*Obr. 5.8: Adresářová struktura na FTP serveru*

## 5.8 Vývoj Uživatelské aplikace

Při vývoji Uživatelské aplikace jsem vycházel z již použitého kódu v aplikaci Administrátorské. Třídy, funkce a metody jsou velmi podobné, hlavně v části lokalizační, protože lokalizace v Administrátorské aplikaci má prakticky stejné potřeby jako lokalizace v Uživatelské. Funkcí, kterou je potřeba doplnit, je zaslání databáze s poslední lokalizovanou polohou na FTP server, kde bude k dispozici v dohledové části. Protože by tato aplikace měla být k dispozici uživatelům, je tedy o mnoho jednodušší na použití a samozřejmě má také jak českou, tak anglickou jazykovou mutaci.

Po spuštění aplikace se ze souboru `configuration.dat`, který je uložený v telefonu v `/rpls/config/`, načte konfigurace nutná k přihlášení na FTP. Následně se zobrazí hlavní menu obsahující 3 tlačítka: Spustit lokalizaci, Jazyk aplikace, Konfigurace sítě. V případě, že soubor `configuration.dat` v telefonu není, spustí se místo hlavního menu aktivita spustitelná pod tlačítkem Konfigurace sítě a umožní nám konfiguraci zadat ručně. Tlačítko Jazyk aplikace nám umožňuje přepínat z ČJ do AJ. Po stisknutí tlačítka Spustit lokalizaci se spustí stahování potřebných souborů z FTP. Mezi nimi je soubor aktivního měření pro danou síť, k němu přidružená databáze a mapa. Po stažení se tato mapa ihned načte a po stisknutí tlačítka Spustit lokalizaci začne aplikace každých 6s vykreslovat na mapu bod aktuální pozice. V tomto intervalu také zasílá na FTP server poslední známou pozici uživatele.

## 5.9 Popis tříd Uživatelské aplikace

Tato aplikace obsahuje celkem 18 tříd, z nichž větší část velmi podobná třídám v Administrátorské aplikaci. Jejich vzájemné vztahy jsou také obdobné jako v Administrátorské. Třídy se stejnou funkcí již nebudu popisovat a zaměřím se na třídy doplněné.

### 5.9.1 Třída `DownloadLocList.java`

V této třídě, která se volá po stisknutí tlačítka Spustit lokalizaci v hlavním menu, se stahují soubory aktivního survey z FTP. Je tedy typu `AsyncTask`. Dále se zde čte stažený soubor, ze kterého zjistíme název aktivní databáze a mapy, kterou má aplikace dále stáhnout. Tyto názvy se předávají do dalších tříd `AsyncTask`, konkrétně `DownloadDB.java` a `DownloadMap.java`. Ty se volají v části `onPostExecute`.

### 5.9.2 Třída `DownloadMap.java`

Jak již bylo naznačeno, tato třída má za úkol stáhnout z FTP korektní mapu a zajistit její načtení do komponenty `ImageView`.

### 5.9.3 Třída `HandleTransfer.java`

Tato třída má za úkol zajišťovat správný průběh nahrávání souborů na FTP. Konkrétně jde o nahrávání databáze s poslední známou polohou uživatele, která se využívá v dohledové části Administrátorské aplikace. Protože se tyto databáze nahrávají na FTP v libovolném čase (každých 6s od spuštění lokalizace) a jejich přenos na FTP chvíli trvá a zároveň si je dohledová aplikace každých 6s stahuje a používá z nich informace, bylo třeba zajistit, aby se nejdříve tyto soubory na FTP nahrály celé, než je bude dohledová část stahovat a otevírat. Pokud by se toto neošetřilo, docházelo by k neustálým pádům aplikace. Právě toto ošetření má na starost třída `HandleTransfer.java`. Před započítím samotného uploadu souboru na FTP server, vytvoří na serveru tzv. „příznak“, který se jmenuje stejně jako nahrávaný soubor a má příponu `.p`. Tento příznak značí, že se původní soubor začne nahrávat. Po dokončení nahrávání z FTP serveru zmizí. Tím dá dohledové části znamení, že si databázi může stáhnout.

Všechny ostatní třídy této aplikace jsou funkčně téměř shodné s aplikací Administrátorskou a pro detailnější informace o rozdílech je potřeba podívat se přímo do kódu aplikace. Pokud bych je vysvětloval zde, byla by práce zbytečně obsáhlá.

## 5.10 Podmínky pro správné využití Uživatelské aplikace

Aplikace je vytvořena pro OS Android 3.0 a vyšší, ideálně však OS Android 4.1 a výše. Zařízení musí mít ve výbavě bezdrátový adaptér. Aplikace by měla být k dispozici uživatelům, je potřeba, aby měli kromě aplikace samotné při instalaci k dispozici i údaje nutné pro přihlášení na FTP server. Aplikace pro instalaci a svůj chod vyžaduje následující oprávnění:

- a) `android.permission.ACCESS_WIFI_STATE` – Přístup ke stavu bezdrátového adaptéru
- b) `android.permission.CHANGE_WIFI_STATE` – Měnit stav bezdrátového adaptéru
- c) `android.permission.ACCESS_NETWORK_STATE` – Získat stav připojené sítě
- d) `android.permission.CHANGE_NETWORK_STATE` – Měnit stav připojené sítě
- e) `android.permission.WRITE_EXTERNAL_STORAGE` – Zapisovat do externího úložiště
- f) `android.permission.READ_EXTERNAL_STORAGE` – Číst z externího úložiště
- g) `android.permission.WRITE_INTERNAL_STORAGE` – Zapisovat do interního úložiště
- h) `android.permission.READ_INTERNAL_STORAGE` – Číst z interního úložiště
- i) `android.permission.WAKE_LOCK` – Používat zámek bezdrátového adaptéru
- j) `android.permission.INTERNET` – Svolení s přístupem na internet

## 5.11 Vývoj aplikace Lokalizační služba

Aplikace Lokalizační servis by měla běžet na zařízeních, které nebudou primárně obsluhovány člověkem, respektive by měla lokalizovat nějaké vybavení (např. nemocniční lůžko). Vzhledem k tomu, že vybavení samotné nepotřebuje znát svou pozici, není zde potřeba načítání mapy z FTP serveru, a tudíž není ani potřeba mapu zobrazovat. Připojení k FTP je ale stále potřeba pro stažení správné databáze, ze které se počítá poloha zařízení. FTP server je zde také potřebný pro ukládání dat poslední známé pozice pro dohledový systém. Stejně jako u Uživatelské aplikace, i zde je většina tříd shodných s Administrátorskou aplikací a proto se zaměřím primárně na ještě nezmiňované třídy.

Hlavní menu této aplikace je tvořeno dvěma tlačítky, konkrétně `Spustit lokalizační službu` a `Zastavit lokalizační službu`. Aplikaci je možné opět přepnout z češtiny do angličtiny pomocí rolovacího menu. Po stisknutí `Spustit lokalizační službu` se spustí stahování souboru aktivního měření a následně stažení správné databáze pro výpočet polohy. Po zjištění polohy se na FTP server uloží databáze s údaji o poloze. Tato aplikace je funkčně nejjednodušší. Služba poběží od stisku prvního tlačítka, dokud se nestiskne tlačítko `Zastavit lokalizační službu`, a to i na pozadí.

## 5.12 Popis tříd Lokalizační služby

### 5.12.1 Třída `ClientService.java`

Tato třída zajišťuje běh časovače, který každých 12s spustí skenování okolí pomocí `Multiscan.java`. Dále zajišťuje ukládání souborů a volání třídy s uploadem databáze na FTP server (`UploadDB.java`).

Zbylé třídy jsou buď shodné s Administrátorskou, nebo Uživatelskou aplikací.

### 5.13 Podmínky pro správné použití Lokalizační služby

Pro správné použití je potřeba zařízení s OS Android alespoň 3.0 a vyšší, nejlépe však s OS Android 4.1 a vyšší. Zařízení musí obsahovat bezdrátový adaptér. Je potřeba korektní nastavení k připojení na FTP server. Aplikace pro instalaci a svůj chod vyžaduje následující oprávnění:

- a) android.permission.ACCESS\_WIFI\_STATE – Přístup ke stavu bezdrátového adaptéru
- b) android.permission.CHANGE\_WIFI\_STATE – Měnit stav bezdrátového adaptéru
- c) android.permission.ACCESS\_NETWORK\_STATE – Získat stav připojené sítě
- d) android.permission.CHANGE\_NETWORK\_STATE – Měnit stav připojené sítě
- e) android.permission.WRITE\_EXTERNAL\_STORAGE – Zapisovat do externího úložiště
- f) android.permission.READ\_EXTERNAL\_STORAGE – Číst z externího úložiště
- g) android.permission.WRITE\_INTERNAL\_STORAGE – Zapisovat do interního úložiště
- h) android.permission.READ\_INTERNAL\_STORAGE – Číst z interního úložiště
- i) android.permission.WAKE\_LOCK – Používat zámek bezdrátového adaptéru
- j) android.permission.INTERNET – Svolení s přístupem na internet

## 6. FTP server

### 6.1 Co je to FTP

FTP neboli File Transfer Protocol, slouží pro přenos souborů mezi počítači (nejčastěji ve vztahu klient-server) v počítačové síti, případně na internetu. Využívá protokol TCP a je nezávislý na využívané platformě. Jeho základy vznikly v roce 1985 (RFC 959) a rozšířen byl v roce 1997 (RFC 2228). V pozdějším vývoji byla zahrnuta podpora IPv6, zabezpečení přenosu a možnost pasivního režimu.

FTP využívá porty 20 a 21. Na portu 20 se přenáší data, port 21 slouží výhradně pro naslouchání a navazování komunikace. Aby bylo možné přenášet data mezi klientem a serverem, je potřeba, aby se klient na server přihlásil. Přihlášení je možné buď pod anonymním účtem (tj. účet bez hesla, který má velmi omezené pravomoci, nejčastěji může soubory pouze číst, případně stahovat) nebo plnohodnotným účtem uživatele (přihlášení pomocí jména a hesla, po přihlášení na server může se soubory dělat vše, co má povoleno administrátorem). Pro úkony se používají příkazy řídicího jazyka, např.:

- a) USER – zadání uživatelského jména
- b) PASS – zadání hesla
- c) CWD – změna adresáře
- d) QUIT – ukončení spojení
- e) TYPE – typ dat (například text nebo binární soubory)
- f) DELE – smazání souboru
- g) LIST – získání seznamu souborů

Hlavní nevýhodou tohoto protokolu pro přenos dat je jeho nízké zabezpečení. Hesla a soubory jsou ve standardním protokolu zasilána jako prostý text, takže kdokoli, kdo může komunikaci „naslouchat“ si je může zjistit a použít. Tento nedostatek odstraňují pozdější rozšíření (například FTPS). Další nevýhodou je nemožnost sloučit více souborů do jednoho balíku a ten přenášet. To zvyšuje časovou náročnost na přenos a zvyšuje i vytížení serveru.

Výhodou je naopak širší možnosti nastavení a použití, než HTTP server a serverová část je oproti HTTP mnohem jednodušší a tudíž i jeho použití a nastavení je snazší.

Jak již bylo zmíněno, FTP umožňuje dva režimy komunikace. Aktivní a pasivní režim. V pasivním režimu, který je dnes nejvíce využíván, navazuje spojení pro přenos dat klient. V aktivním režimu naopak datovou komunikaci navazuje server a klient naslouchá. V aktivním režimu se mohou vyskytnout potíže, když je klient v privátní síti a jeho IP je překládána pomocí NAT.

## 6.2 Použitý FTP server, korektní nastavení

Pro testování vytvořených aplikací jsem využil freeware FTP serveru s názvem Xlight FTP Server, který je zdarma k dispozici ke stažení na internetu. Server to není nejideálnější, nicméně jeho nastavení bylo rychlé a pro účely prvotního testování byl dostatečně vhodný. Pro reálné využití však doporučuji běžně využívané FTP servery.

Bohužel se při reálném testování aplikace ukázalo, že Xlight FTP server způsoboval chyby v přenesených souborech. Také doba přenosu souborů byla neúměrně vysoká jejich velikosti. Proto jsem se po testování rozhodl využít trial verzi Cerberus FTP Serveru. Jeho nastavení je o něco složitější, než u Xlight FTP, nicméně rychlosti jsou mnohem vyšší a hlavně umožňuje zobrazování podrobných logů (zpráv) o komunikaci, takže veškeré chyby a nesrovnalosti podrobně zachytí. Díky tomu se mi tyto chyby dařilo celkem úspěšně odstranit.

Pro správnou funkci je potřeba vytvořit nejméně jednoho uživatele s přístupem ke složkám používaných aplikacemi. Tyto složky musí být předem vytvořeny dle struktury na *Obr. 5.8* zmíněném v podkapitole 5.7. Vytvořený uživatel musí mít heslo obsahující alespoň 1 znak a musí mít práva do vytvořených složek zapisovat, číst z nich a mazat soubory. FTP server je možné provozovat na jakémkoliv portu a IP adrese, je tedy možné využít i FTP přístupný přes internet. Nicméně, pro dostatečnou rychlost přenosu souboru důrazně doporučuji využít FTP server připojený ve stejné síti, v nichž se necháváme lokalizovat. Konfiguraci nutnou k připojení na FTP je vhodné dát k dispozici všem případným uživatelům.

## 7. Zhodnocení kolísání úrovní signálu

Protože se během testování aplikace objevilo, pro výpočty polohy zařízení, dosti výrazné kolísání úrovní naměřeného signálu, je potřeba toto vyhodnotit a vyvodit důsledky na lokalizační systém.

Vlastní měření kolísání probíhalo 4x1 hodinu na 2 zařízeních ve 4 režimech:

- 1) Zapnuté vysílání stabilního signálu z AP, zařízení připojené k síti
- 2) Zapnuté vysílání stabilního signálu z AP, zařízení nepřipojené k síti
- 3) Vypnuté vysílání stabilního signálu z AP, zařízení připojené k síti
- 4) Vypnuté vysílání stabilního signálu z AP, zařízení nepřipojené k síti

Měřil jsem na telefonech Samsung Galaxy S2 GT-I9100 a Prestigio PAP4322 Duo. Hodnoty síly signálu se zaznamenávaly každých 8 vteřin. Oba telefony byly na stejném místě po celou dobu měření. Naměřené výsledky byly exportovány do MS Excel. Výsledné grafy jsou, vzhledem ke své velikosti, umístěny na CD přiloženém k práci ve složkách odpovídající názvům měření spolu s naměřenými hodnotami, protože není možné je přehledně vytisknout na list A4.

Z výsledků měření vyplývá, že signál kolísá nezávisle na čase, naprosto náhodně. Proto nám toto kolísání, bohužel, vnáší chybu do lokalizačního systému. Získaná výsledná poloha tedy není tak přesná a dle testů provedených kolegou Cihlářem se pohybuje v řádu několika metrů. Kolísání bude potřeba vhodnou úpravou aplikace eliminovat nebo alespoň minimalizovat v další fázi vývoje. Nepřesnější z měření bylo měření č. 1 s hodnotami v dBm.

Další problém, který se vyskytl během testování, byl odlišný výrobce bezdrátového adaptéru v zařízeních a jeho kalibrace. Dle měření na stejném místě všemi použitými přístroji se naměřené hodnoty rozcházejí a jsou znázorněny opět v grafech na příloženém CD (složka pro měření č. 1, poslední 3 listy sešitu Excel).

## 8. Závěr

Díky vypracování této práce jsem získal více informací o chování bezdrátových sítí, naučil jsem se pohybovat v rozhraní pro programování aplikací na OS Android a získal znalosti a praktické zkušenosti s vytvářením FTP serveru a jeho používáním. Vývoj aplikace ve mně probudil velký zájem ve zdokonalení aplikace stávající a vytváření dalších.

Při seznamování se s funkcí RTLS systému Ekahau jsem narazil na několik problémů, mezi nimi hlavně kompatibilita pouze s některými zařízeními a proto nebylo vůbec jednoduché ho zprovoznit. Z tohoto důvodu jsme s kolegou Cihlářem pro řešení dalšího postupu vybrali z tohoto systému pouze některé funkce a vlastnosti, konkrétně metody pro lokalizaci a pomocí Ekahau Site Survey jsme vytvořili simulaci pokrytí prostoru signálem. Díky tomu bylo možné kvalitně navrhnout a sestavit lokalizační bezdrátovou síť, která splňuje většinu potřebných parametrů pro návrh sítě.

Po konfiguraci sítě a provedení analýzy pokrytí jsem se pustil do vývoje aplikace pro OS Android, která má za úkol lokalizovat uživatele či zařízení, zobrazit jeho polohu na mapě v zařízení a zároveň tuto polohu sdělovat dohledovému systému. Při vývoji vznikly 3 aplikace. Administrátorská s dohledovým systémem, možností proměření prostoru a vlastní lokalizací, Uživatelská s lokalizačním systémem a Lokalizační služba, která má za úkol lokalizaci zařízení bez zobrazování polohy. Všechny aplikace zasílají data na předem nakonfigurovaný FTP server, odkud si je také v případě potřeby mohou stáhnout. Využil jsem trial verze FTP serveru Cerberus FTP Server. Na tomto serveru jsou také veškeré potřebné mapové podklady, které si aplikace aktivně stahují na základě připojení (případně v konfiguračním souboru napsané) sítě. Dále je pro úspěšné využití aplikace potřeba, aby zařízení „vidělo“ nejméně 3, lépe však 4 AP.

Dále jsem změřil kolísání úrovně signálu. Měření probíhalo paralelně ve dvou různých zařízeních ve 4 režimech. Každé měření probíhalo zhruba hodinu a zařízení byla stále na stejném místě. Nejlepších výsledků dosáhlo měření při vysílání stabilní úrovně signálu z AP se zařízením připojeným k síti, se silou signálu uvedenou v dBm, pro obě zařízení. I přes to ale aplikace využívá při měření převod dBm na mW a při zjišťování síly signálu je odpojena od sítě. Výsledky v samotné lokalizaci byly v tomto módu při použitém algoritmu WkNN nejpřesnější (viz. práce kolegy Cihláře). Bohužel, aplikace zatím nejspíše není připravena pro reálné nasazení. Je to hlavně z toho důvodu, že bezdrátové adaptéry různých zařízení jsou různě kalibrovány. Ve většině případů tedy



například Samsung Galaxy S2 naměří v jednom místě sílu signálu -43dBm, Prestigio PAP4322 Duo -55dBm a LG Optimus L7II P710 -35dBm. Z tohoto důvodu není pro lokalizaci možné využít databázi naměřených sil signálu z jakéhokoliv z těchto telefonů pro lokalizování ostatních. Bohužel jsme k testům neměli k dispozici zařízení stejná (například 2x Samsung Galaxy S2) a proto nemohu odhadnout, jak se liší síla signálu pro stejné modely zařízení. Budeme-li ale uvažovat, že se neliší řádově o více než 5dBm, pak lze aplikaci využít pro lokalizování zařízení stejného typu a modelu, jako bylo zařízení, s nímž se provádělo měření. Polohu lze dle měření kolegy s 85% úspěšností určit na úroveň místnosti s přesností 2-3 metry.

Ve vývoji aplikace bych rád i nadále pokračoval. Je potřeba vylepšit algoritmus pro výpočet polohy (pokus o eliminaci kolísání síly signálu matematickou cestou a to i pro použití zařízení různých modelů a výrobců tak, aby mohly být k lokalizaci využity pokud možno všechna dostupná zařízení), komunikaci aplikací s FTP serverem, případně tento nahradit serverem jiného typu, nejlépe databázového. Dále bych rád přidal do dohledové části možnost přepínat mezi jednotlivými mapami (například mapy různých podlaží). Také bych rád doplnil lokalizační systém o navigační, ve kterém by si uživatel na mapu umístil cílový bod, a aplikace by mu ukazovala, jakým směrem jít, a jak je bod zhruba vzdálený.

Aplikace jsou umístěná na CD přiloženém k práci. Nachází se ve složce Aplikace a dále ve složkách dle názvů jednotlivých aplikací. Spolu s nimi se ve složce Aplikace nachází také textový soubor ReadMe.txt, ve kterém jsou shrnuty, kromě jiného, známé pády aplikace, které se zatím nepodařilo odstranit. Na tomto CD se také nachází výše zmíněné výsledky měření kolísání signálu. Tabulky a grafy v sešitech MS Excel jsou umístěné ve složce Měření.

## Seznam použité literatury

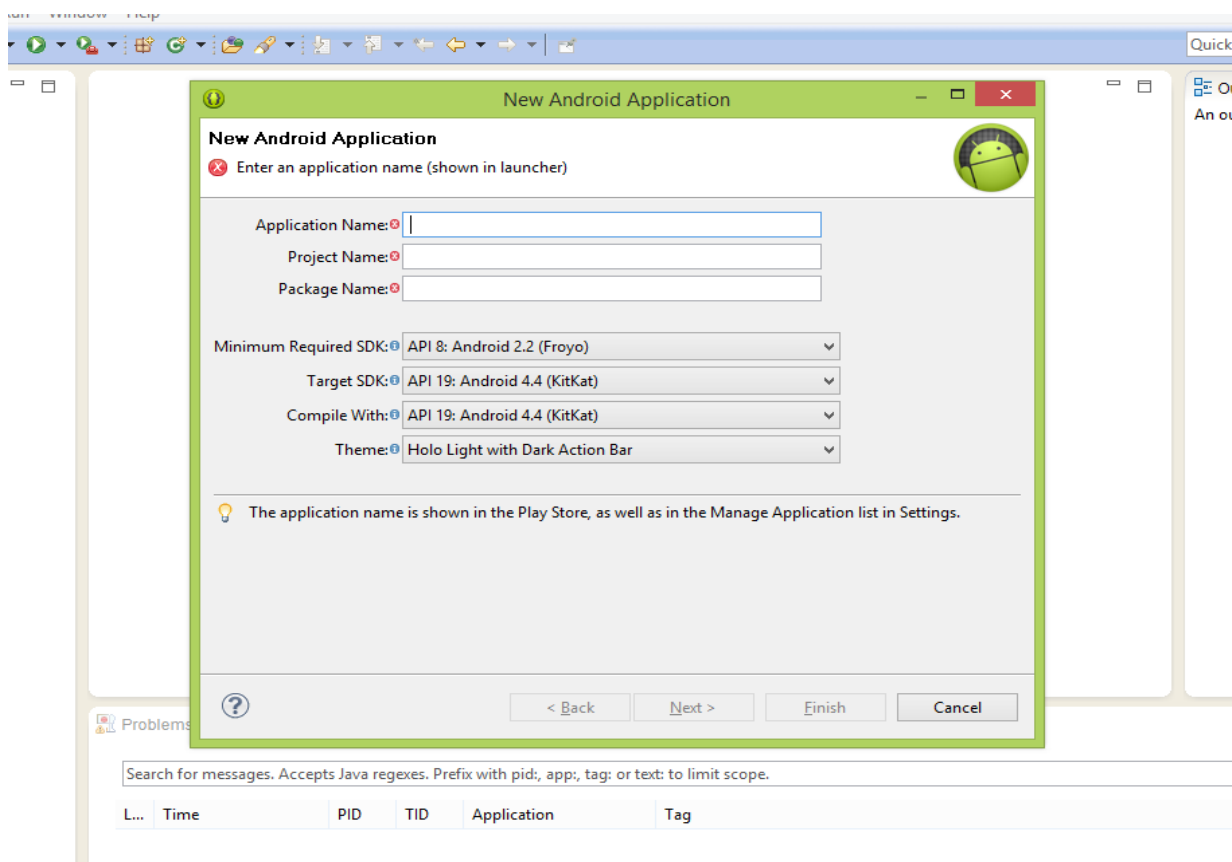
- [1] CHANDRA, P. - M. DOBKIN, D. Basics of Wireless Local Area Networks. V: *Wireless networking: Know It All*. Burlington: Elsevier/Newnes, ©2008, str. 47 - 95. ISBN 0750685824.
- [2] RECOMMENDATION SM. 1056-1. *Limitation of radiation from industrial, scientific and medical (ISM) equipment*. ITU-R, 1997 - 2007. Dostupné z: [http://www.itu.int/dms\\_pubrec/itu-r/rec/sm/R-REC-SM.1056-1-200704-I!!PDF-E.pdf](http://www.itu.int/dms_pubrec/itu-r/rec/sm/R-REC-SM.1056-1-200704-I!!PDF-E.pdf)
- [3] LEE, B. Gi - CHOI, S. *Broadband wireless access and local networks: Mobile WiMax and WiFi*. London: Artech House, ©2008, str. 36. ISBN 1596932937.
- [4] VO-R/12/09.2010-12. *Využívání rádiových kmitočtů a k provozování zařízení pro širokopásmový přenos dat v pásmech 2,4 GHz až 66 GHz*. Praha: Český telekomunikační úřad, 2010. [online]. [cit. 2014-04-11]. Dostupné z: [http://www.ctu.cz/cs/download/oop/rok\\_2010/vo-r\\_12-09\\_2010-12.pdf](http://www.ctu.cz/cs/download/oop/rok_2010/vo-r_12-09_2010-12.pdf)
- [5] MALIK, A. *Rtls for dummies(r)*. 1st ed. Indianapolis, Wiley Pub., Inc., 2009. ISBN 04-703-9868-X.
- [6] Prof. Ing. MAZÁNEK, M., CSc. - Prof. Ing. PECHAČ, P., Ph.D. - Prof. Ing. Vrba, J., CSc. *Základy antén, šíření vln a mikrovlnné techniky*. Vyd. 1. Praha: Vydavatelství ČVUT, 2008. ISBN 8001039978.
- [7] A-level Physics (Advancing Physics)/ Radar and Triangulation. *WIKIBOOKS* [online]. 2012 [cit. 2014-04-11]. Dostupné z: [http://en.wikibooks.org/wiki/A-level\\_Physics\\_\(Advancing\\_Physics\)/Radar\\_and\\_Triangulation](http://en.wikibooks.org/wiki/A-level_Physics_(Advancing_Physics)/Radar_and_Triangulation)
- [8] KHODAYARI, S. - MALEKI, M. - HAMED, E., *A RSS-based fingerprinting method for positioning based on historical data, Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, 2010 International Symposium on, str. 306, 310, 11-14 July 2010. [cit. 2014-04-11]. Dostupné z: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5589221&isnumber=5587938>
- [9] *Dokumentace Ekahau*. [online]. [cit. 2014-04-14]. Dostupné z: <http://www.ekahau.com/>
- [10] Dashboards. *Developer Android* [online]. [cit. 2014-05-14]. Dostupné z: <http://developer.android.com/about/dashboards/index.html>
- [11] KOCUR, Z. A ŠAFRÁNEK, M.. *Fyzická vrstva Wi-Fi* [online]. 2008 [cit. 2014-05-15]. Dostupné z: <http://access.feld.cvut.cz/view.php?cislocianku=2008050006>
- [12] Android (operating system). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-2014 [cit. 2014-05-17]. Dostupné z: [http://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)](http://en.wikipedia.org/wiki/Android_(operating_system))

- [13] Jak funguje GPS?. *Svět Hardware* [online]. 2006 [cit. 2014-05-17]. Dostupné z: <http://www.svethardware.cz/jak-funguje-gps/21826-5>

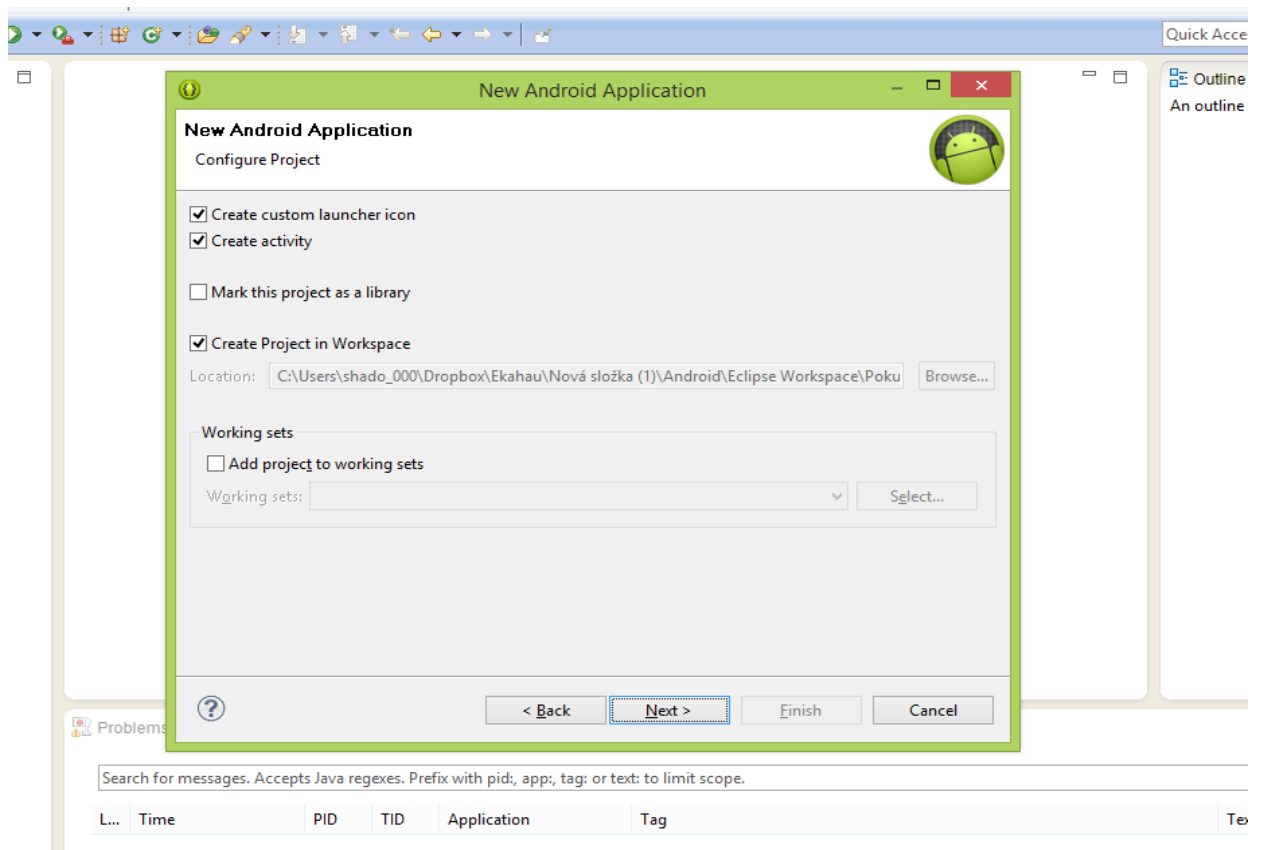
## 9. Seznam příloh

### 9.1 Návod pro vytvoření aplikace v prostředí Eclipse

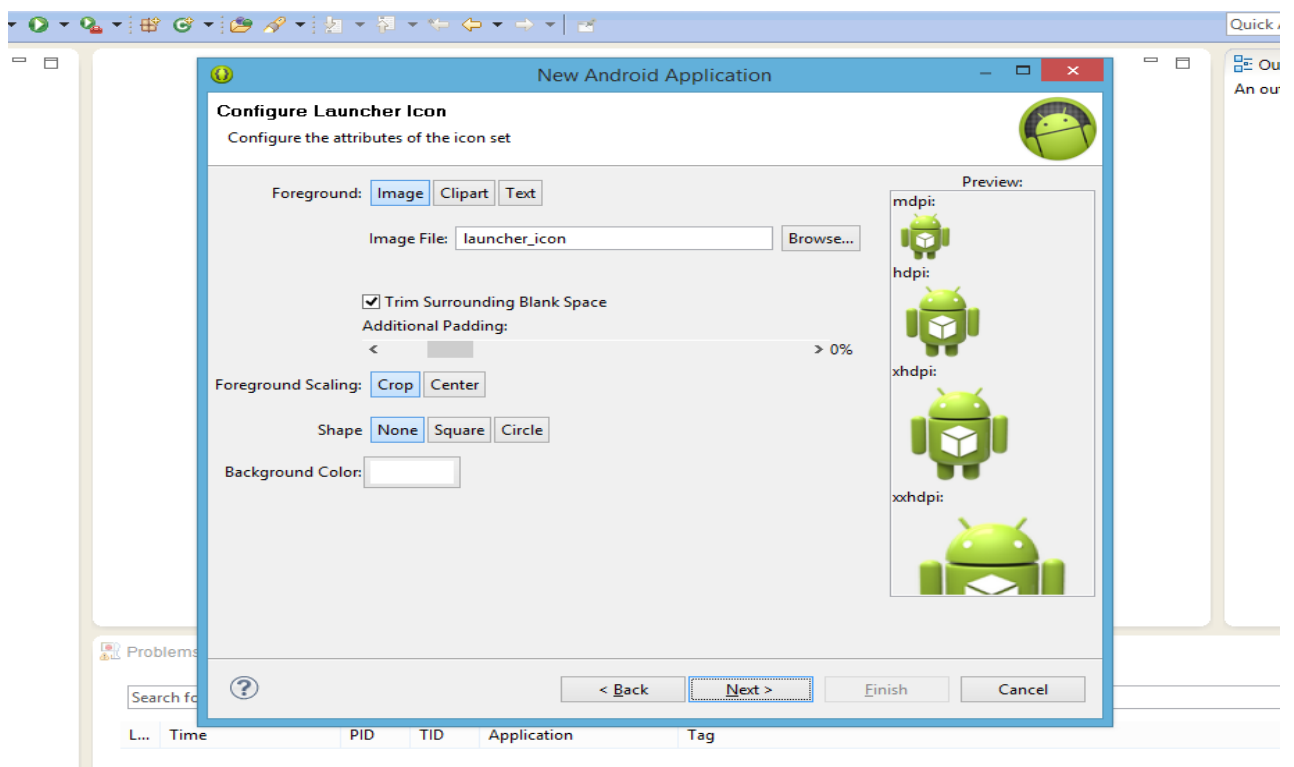
- 1) Nejprve spustíme program SDK Manager, který stáhne požadované doplňky, knihovny a ovladače
- 2) Spustíme Eclipse a nastavíme si pracovní složku (Workspace)
- 3) Stiskneme tlačítko New a vybereme New Android Application Project
- 4) Projekt pojmenujeme (*Obr. 9.1*), stiskneme Next
- 5) V další části průvodce (*Obr. 9.2*) vybereme požadované možnosti a stiskneme Next
- 6) Dále vybereme ikonku launcheru, která se bude zobrazovat v zařízení po nainstalování (*Obr. 9.3*), stiskneme Next
- 7) Vybereme a vytvoříme hlavní aktivitu (*Obr. 9.4*) a pojmenujeme jí (*Obr. 9.5*). Stiskneme Finish
- 8) Projekt je vytvořen, můžeme začít programovat



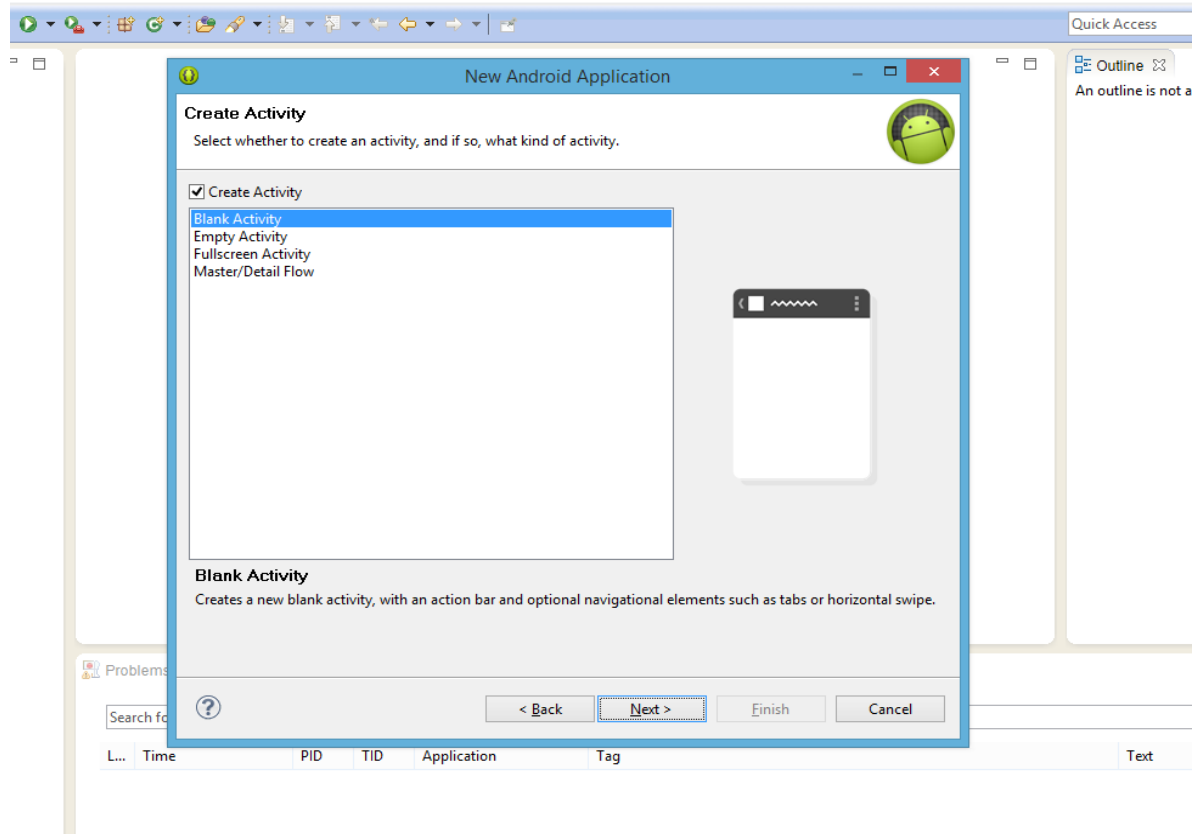
*Obr. 9.1: Vytvoření nového projektu, pojmenování*



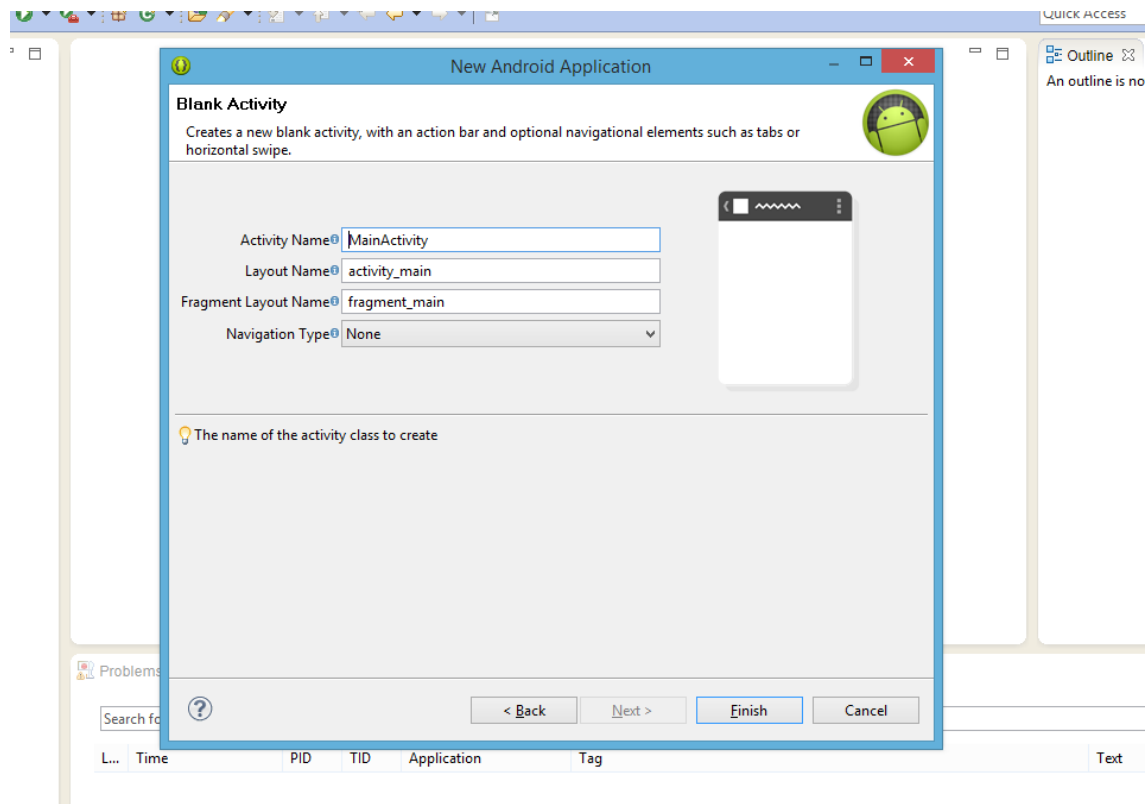
Obr. 9.2: Umístění projektu do Workspace



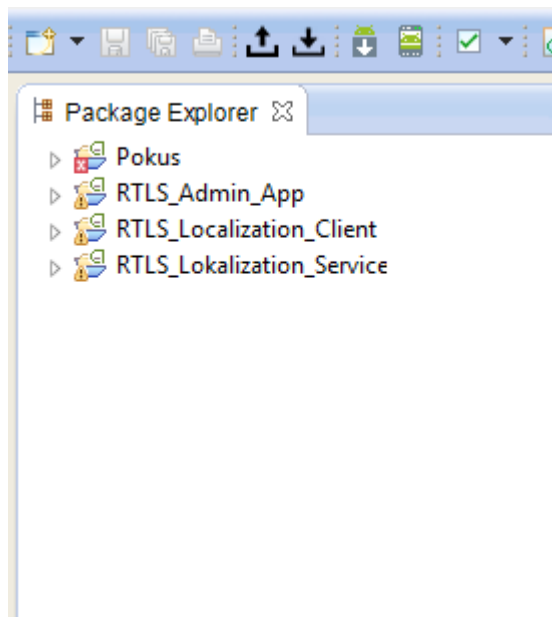
Obr. 9.3: Výběr ikony launcheru



Obr. 9.4: Vytvoření a výběr hlavní aktivity



Obr. 9.5: Pojmenování hlavní aktivity



Obr. 9.6: Projekt po vytvoření

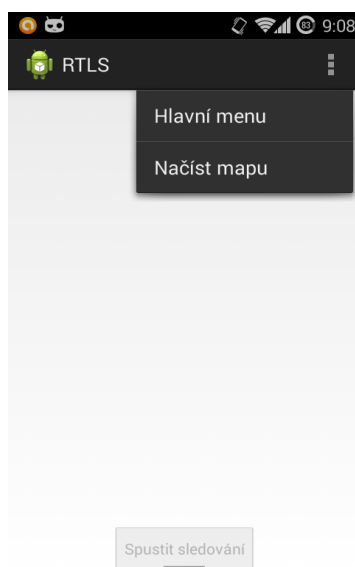
## 9.2 Návody na použití aplikací

### Návod na použití Administrátorské aplikace:

1. Nainstalujeme aplikaci z instalačního balíčku RTLS.apk
2. Spustíme aplikaci do hlavního menu (Obr. 9.7)
3. Vybereme z nabídky požadovanou volbu (při prvním spuštění se zobrazí nastavení sítě (Obr. 9.22)
  - 3.1. Po stisknutí tlačítka Dohled se stáhnou podklady z FTP, načteme mapu pomocí rolovacího menu a stiskneme tlačítko Spustit (Obr. 9.8 – Obr. 9.10)
  - 3.2. Po stisknutí tlačítka Proměření prostoru se stáhnou podklady z FTP, a dále vybíráme z nabídky v menu, co potřebujeme.
    - 3.2.1. Nové měření nám umožní načíst mapu, pojmenovat si měření, měření provést, následně uložit a aktivovat (Obr. 9.11 – Obr. 9.16)
    - 3.2.2. Načíst měření nám umožní načíst a případně doplnit některé z předchozích měření uložených na FTP. Doplnění bodů probíhá stejně jako při vytváření nového měření
    - 3.2.3. Volba Aktivovat měření zajistí, že provedené měření bude použito pro lokalizaci
    - 3.2.4. Funkce ostatních voleb vyplývá z jejich názvů
  - 3.3. Tlačítko Spustit lokalizaci spustí okno lokalizace, umožňující ověření funkce s provedenými měřeními. Proces spuštění je na Obr. 9.17 – Obr. 9.19.
  - 3.4. Tlačítko Nastavení nás pošle do dalšího menu s těmito možnostmi (Obr. 9.20):
    - 3.4.1. Volba Jazyk přepíná mezi češtinou a angličtinou (Obr. 9.21)
    - 3.4.2. Volba Nastavení sítě obsahuje nastavení k připojení na FTP server a skenované SSID (Obr. 9.22)
    - 3.4.3. Volba O aplikaci zobrazí informace o aplikaci



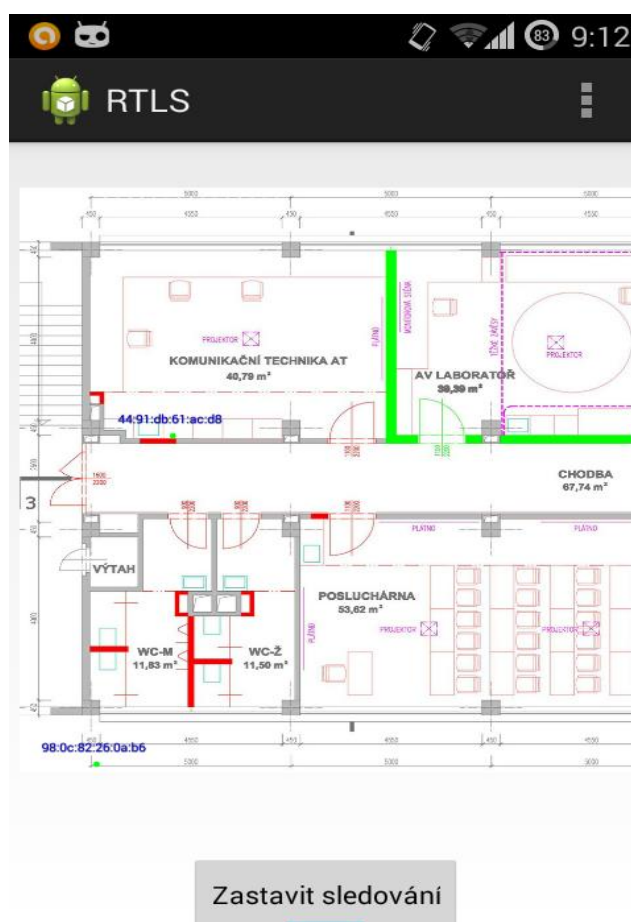
Obr. 9.7: Hlavní menu



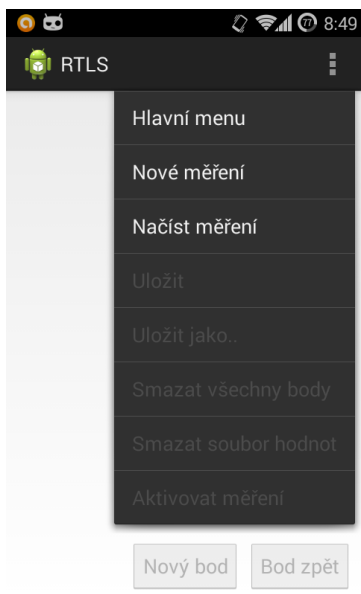
Obr. 9.8: Menu Dohledu



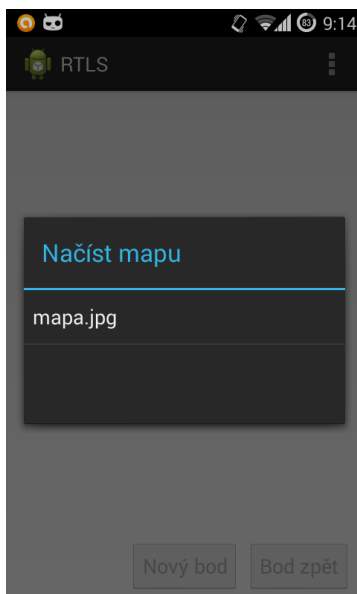
Obr. 9.9: Dohled s mapou



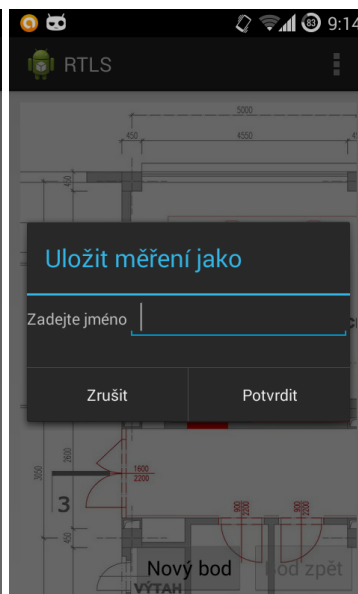
Obr. 9.10: Dohled po spuštění sledování



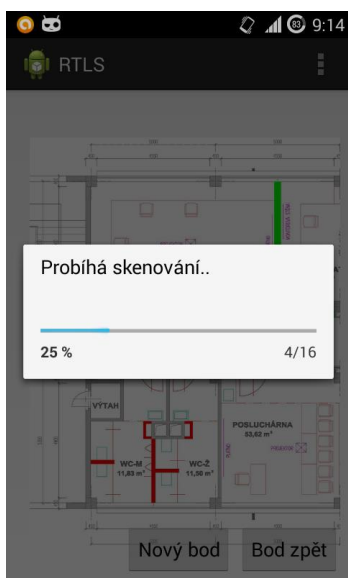
Obr. 9.11: Menu měření



Obr. 9.12: Načtení mapy



Obr. 9.13: Zadání jména



Obr. 9.14: Skenování

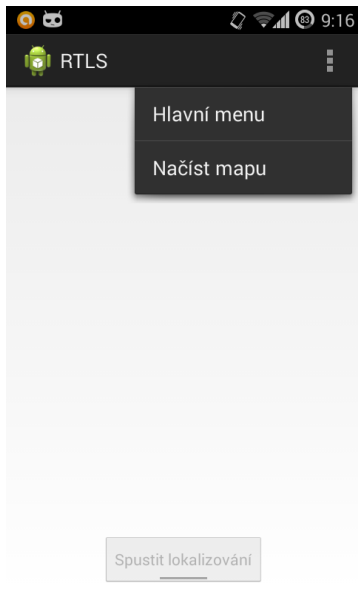


Obr. 9.15: Body na mapě

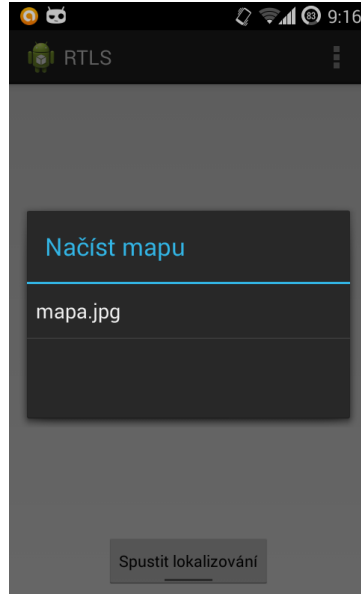


Obr. 9.16: Přístupné volby

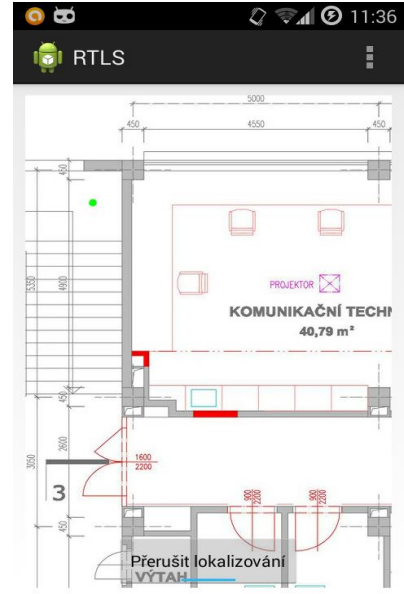




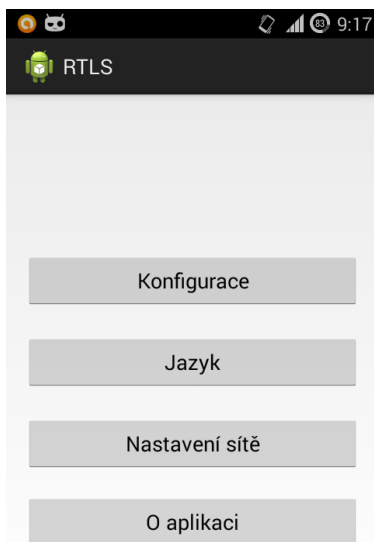
Obr. 9.17: Menu lokalizace



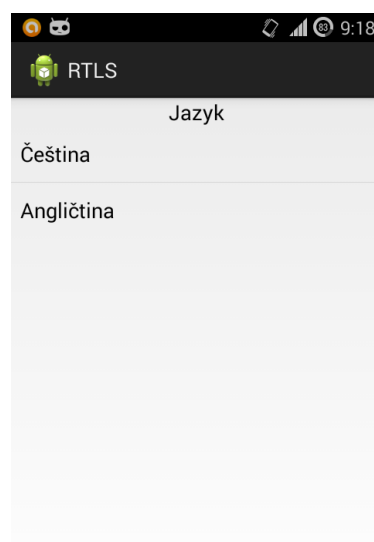
Obr. 9.18: Načtení mapy



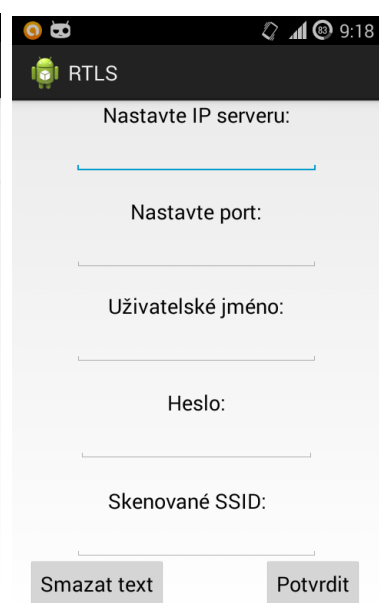
Obr. 9.19 Spuštění lokalizace



Obr. 9.20: Menu nastavení



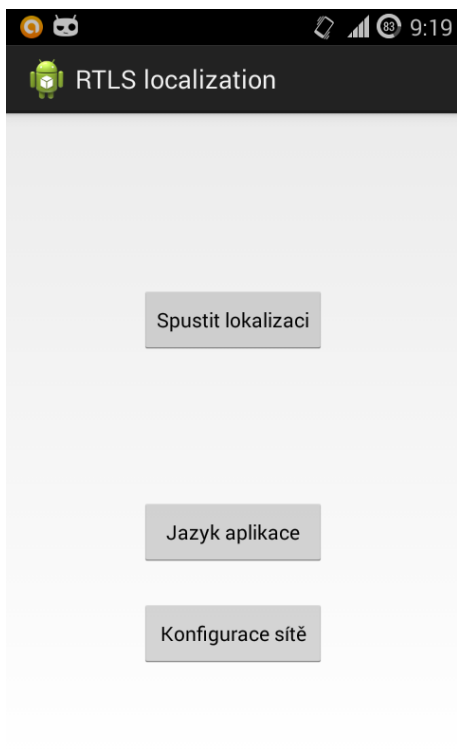
Obr. 9.21: Výběr jazyka



Obr. 9.22: Konfigurace FTP

## Návod na použití Uživatelské aplikace:

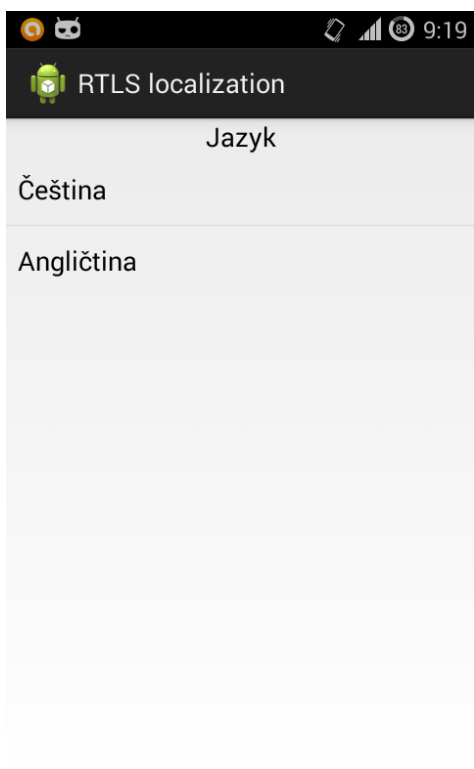
1. Nainstalujeme aplikaci z instalačního balíčku RTLSLocalization.apk
2. Spustíme aplikaci do hlavního menu (*Obr. 9.23*)
3. Vybereme z nabídky požadovanou volbu (při prvním spuštění se zobrazí nastavení sítě (*Obr. 9.26*)
  - 3.1. Tlačítko Spustit lokalizaci stáhne podklady z FTP, zobrazí mapu a po stisknutí tlačítka začne ukazovat pozici uživatele (*Obr. 9.24*)
  - 3.2. Tlačítko Jazyk aplikace umožní přepínat mezi češtinou a angličtinou (*Obr. 9.25*)
  - 3.3. Tlačítko Konfigurace sítě umožní nastavit hodnoty pro připojení na FTP a skenované SSID (*Obr. 9.26*)



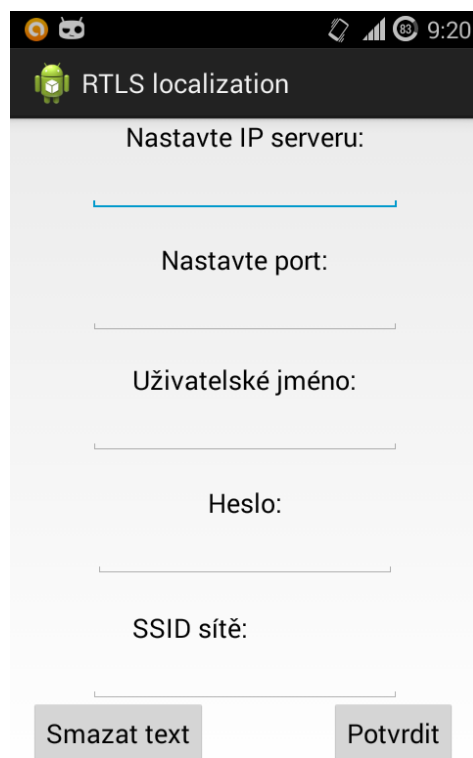
*Obr. 9.23: Hlavní menu*



*Obr. 9.24: Zobrazení pozice*



*Obr. 9.25: Volba jazyka*



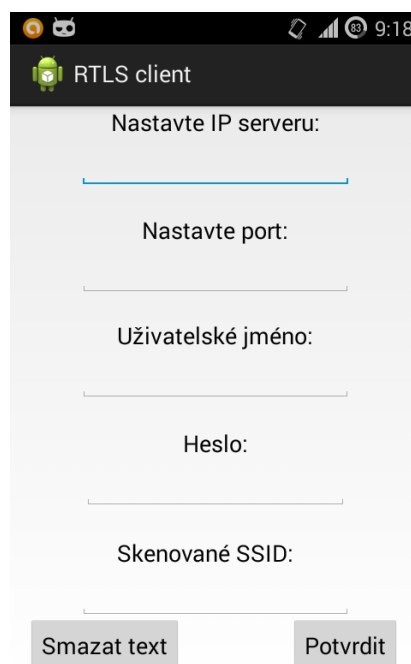
*Obr. 9.26: Konfigurace FTP*

## Návod na použití aplikace Lokalizační služba:

1. Nainstalujeme aplikaci z instalačního balíčku RTLScient.apk
2. Spustíme aplikaci do hlavního menu (*Obr. 9.27*)
3. Vybereme z nabídky požadovanou volbu (při prvním spuštění se zobrazí nastavení sítě (*Obr. 9.28*))
  - 3.1. Tlačítko Spustit lokalizačního klienta spustí službu na pozadí, která bude na FTP zasílat informace o poloze zařízení, dokud se nestiskne Zastavit lokalizačního klienta
  - 3.2. Tlačítko Zastavit lokalizačního klienta zastaví službu na pozadí
  - 3.3. Tlačítko Jazyk aplikace umožňuje přepínání mezi češtinou a angličtinou
  - 3.4. Tlačítko Konfigurace sítě umožní nastavit hodnoty pro připojení na FTP a skenované SSID (*Obr. 9.28*)



*Obr. 9.27: Menu lokalizační služby*



*Obr. 9.28: Konfigurace FTP*

