

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

APLIKACE PRO ZOBRAZENÍ MODELU BEZDRÁTOVÉ SÍTĚ

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

AUTOR PRÁCE  
AUTHOR

Bc. PETR ŽOLDOŠ

BRNO 2011



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**  
**ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

## **APLIKACE PRO ZOBRAZENÍ MODELU BEZDRÁTOVÉ SÍTĚ**

APPLICATION FOR DISPLAY WIRELESS NETWORK MODEL

**DIPLOMOVÁ PRÁCE**  
MASTER'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**Bc. PETR ŽOLDOŠ**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**Ing. PAVEL VAJSAR**

BRNO 2011



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Diplomová práce

magisterský navazující studijní obor  
Telekomunikační a informační technika

**Student:** Bc. Petr Žoldoš

**ID:** 77727

**Ročník:** 2

**Akademický rok:** 2010/2011

**NÁZEV TÉMATU:**

**Aplikace pro zobrazení modelu bezdrátové sítě**

## POKYNY PRO VYPRACOVÁNÍ:

V rámci diplomové práce bude nutné seznámit se s bezdrátovými sítěmi a technologiemi Adobe Flex / Air. Na základě zjištěných informací navrhnout XML strukturu, která bude sloužit jako zdroj dat pro zobrazení modelu sítě v aplikaci a model databáze kompatibilní s XML modelem. V prostředí technologií Adobe Flex / Air navrhnout a vytvořit aplikaci, která bude zobrazovat model bezdrátové sítě na základě navrženého XML modelu, nebo na základě informací obsažených v databázi. Z aplikace bude patrná pozice jednotlivých uzlů spolu s informacemi o uzlech. Aplikace bude schopna zobrazit model sítě ve volném prostoru za použití mapového podkladu, tak i model sítě uvnitř objektu (budovy).

## DOPORUČENÁ LITERATURA:

- [1] Tapper, J. and Labriola, M. and Boles, M. and Talbot, J. Adobe Flex 3: training from the source
- [2] Li, Yingshu, Thai, My T., Wu, Weili, Wireless Sensor Networks and Applications, Signals and Communication Technology, 2008, p. 444, ISBN: 978-0-387-49591-0

**Termín zadání:** 7.2.2011

**Termín odevzdání:** 26.5.2011

**Vedoucí práce:** Ing. Pavel Vajsar

**prof. Ing. Kamil Vrba, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Cieľom diplomovej práce bolo, na základe nadobudnutých vedomostí o technológiách Adobe Flex SDK a tiež Google Maps API, vybudovať aplikáciu, ktorá umožní tvoriť, generovať a upravovať grafický model bezdrôtovej siete. Rozmiestnenie a vlastnosti jednotlivých prvkov sú sledované na mapovom podklade alebo v plánoch budovy. Dáta sú čerpané z užívateľom vyplnených formulárov, z externého súboru alebo v pravidelných intervaloch z pripojeného databázového systému.

Písomná časť približuje použité technológie. Spolu so vzorovými príkladmi zdrojového kódu popisuje vývoj aplikácie a použité riešenia. Súčasťou sú grafické ukážky užívateľského rozhrania.

## **KLÍČOVÁ SLOVA**

aplikácia, bezdrôtová sieť, model, mapa, budova, Adobe Flex SDK, Google Maps API, XML, MXML, ActionScript, databázový systém, PostgreSQL, XML

## **ABSTRACT**

The first step of Master's thesis was to gain knowledge about technologies Adobe Flex SDK and Google Maps API. Knowledge was used to develop an application, which let users create, generate and modify graphical wireless network model. Position and characteristics of each single unit are monitored either in a map interface or in building plans. Dates are gathered from forms filled by current user, from external file or periodically from connected database system.

Theoretical part enlightens technologies that were used. It describes program development and solutions, which were made, along with examples of the source code. Included are printscreens of graphical user interface as well as description of how does it all work.

## **KEYWORDS**

application, wireless network, model, map, building, Adobe Flex SDK, Google Maps API, XML, MXML, ActionScript, database system, PostgreSQL, XML

ŽOLDOŠ, Petr *Aplikácia pre zobrazenie modelu bezdrôtovej siete*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2011. 65 s. Vedoucí práce byl Ing. Pavel Vajsar

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Aplikácia pre zobrazenie modelu bezdrôtovej siete“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Brno .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Touto cestou by som sa rád poďakoval predovšetkým svojej rodine, ktorá mi pomáha a podporuje ma v každom mojom rozhodnutí. Je pre mňa oporou a vážim si všetkého, čo pre mňa moji najbližší robia. Som vďaka nim šťastný a prajem si aby som rovnaké šťastie v živote prinášal aj ja im.

Vďaka patrí aj vedúcemu práce, môjmu kolegovi a dobrému priateľovi Pavlovi Vajsarovi za ochotu a pomoc a to nie len pri riešení diplomovej práce.

# OBSAH

<b>1</b>	<b>Úvod</b>	<b>11</b>
<b>2</b>	<b>Bezdrôtové siete</b>	<b>12</b>
2.1	Úvod do bezdrôtových sietí . . . . .	12
2.2	Nároky bezdrôtových zariadení . . . . .	13
2.3	Oblasti využitia . . . . .	14
<b>3</b>	<b>Adobe Flex</b>	<b>15</b>
3.1	Flex framework . . . . .	15
3.2	Adobe Flash Builder . . . . .	16
<b>4</b>	<b>Adobe AIR</b>	<b>18</b>
<b>5</b>	<b>Geografické vyjadrenie polohy</b>	<b>19</b>
5.1	Google Maps SDK . . . . .	19
5.1.1	Registrácia Google Maps API . . . . .	20
5.1.2	Inštalácia a implementácia . . . . .	21
<b>6</b>	<b>Budovanie aplikácie</b>	<b>22</b>
6.1	Stavy . . . . .	23
6.2	Moduly . . . . .	23
6.3	XML štruktúra . . . . .	24
6.4	HTTPService . . . . .	26
6.5	Ťahaj a pusť . . . . .	26
<b>7</b>	<b>Modul máp</b>	<b>27</b>
7.1	Komponent Map . . . . .	28
7.2	Použitie HttpService . . . . .	28
7.3	Ikony uzlov . . . . .	29
7.4	Funkcie modulu mapy . . . . .	29
7.4.1	Inicializácia . . . . .	30
7.4.2	Nové jednotky . . . . .	31
7.4.3	Vizualizácia jednotiek . . . . .	31
7.4.4	Import . . . . .	32
7.4.5	Export . . . . .	33
7.5	Užívateľské rozhranie . . . . .	34
7.5.1	Tvorba nových jednotiek . . . . .	34
7.5.2	Import . . . . .	35

7.5.3	Zoznam jednotiek . . . . .	36
7.5.4	Ukladanie vytvorených dát . . . . .	36
7.5.5	Mazanie . . . . .	36
7.5.6	Ďalšie vlastnosti . . . . .	37
<b>8</b>	<b>Modul budovy</b>	<b>38</b>
8.1	Komponent Accordion . . . . .	38
8.2	Funkcie modulu budovy . . . . .	40
8.2.1	Tvorba a vizualizácia jednotiek . . . . .	40
8.2.2	Určovanie polohy jednotiek . . . . .	41
8.2.3	Mazanie jednotiek . . . . .	42
8.2.4	Zmena pohľadu . . . . .	42
8.3	Spojenie s databázovým systémom . . . . .	43
8.3.1	Dotaz SELECT . . . . .	44
8.3.2	Dotaz INSERT . . . . .	45
8.4	Užívateľské rozhranie . . . . .	45
8.4.1	Tvorba nových jednotiek . . . . .	46
8.4.2	Import . . . . .	47
8.4.3	Zoznam jednotiek . . . . .	48
8.4.4	Pohľad z inej perspektívy . . . . .	48
8.4.5	Výpočet odchýlky polohy . . . . .	50
8.4.6	Mazanie . . . . .	51
8.4.7	Ďalšie vlastnosti . . . . .	51
<b>9</b>	<b>Verzia desktopového programu</b>	<b>52</b>
<b>10</b>	<b>Nasadenie aplikácie v praxi</b>	<b>55</b>
10.1	Potenciál ďalšieho využitia . . . . .	55
10.2	Možnosti rozšírenia . . . . .	55
<b>11</b>	<b>Záver</b>	<b>57</b>
	<b>Literatura</b>	<b>58</b>
	<b>Seznam symbolů, veličin a zkratek</b>	<b>60</b>
	<b>Seznam příloh</b>	<b>61</b>
<b>A</b>	<b>Prvá príloha</b>	<b>62</b>
A.1	XML štruktúra pre model siete na širšom území . . . . .	62
A.2	XML štruktúra pre model siete v budove . . . . .	63



A.3 CSV štruktúra . . . . .	64
<b>B Druhá príloha</b>	<b>65</b>
B.1 Priložené médium . . . . .	65

# SEZNAM OBRÁZKŮ

3.1	Spôsob komunikácie Flex aplikácie so serverom. . . . .	16
7.1	Základné užívateľské rozhranie . . . . .	27
7.2	Možnosti užívateľského rozhrania . . . . .	34
7.3	Formulár pre vloženie novej bezdrôtovej jednotky . . . . .	35
7.4	Formulár pre vloženie XML štruktúry . . . . .	35
7.5	Zoznam vytvorených/importovaných prvkov . . . . .	36
8.1	Základné užívateľské rozhranie modulu . . . . .	38
8.2	Formulár pre vytvorenie nového objektu bezdrôtovej siete . . . . .	47
8.3	Maximalizovaná tabuľka s testovacími databázovými záznamami . . .	49
8.4	Komponent pohľadu na poschodie spredu . . . . .	49
8.5	Komponent pohľadu na poschodie spredu . . . . .	50
9.1	Inštalácia AIR aplikácie – krok číslo 1 . . . . .	52
9.2	Inštalácia AIR aplikácie – krok číslo 2 . . . . .	52
9.3	Inštalácia AIR aplikácie – krok číslo 3 . . . . .	52
9.4	Inštalácia AIR aplikácie – krok číslo 4 . . . . .	53
9.5	Inštalácia AIR aplikácie – krok číslo 5 . . . . .	53
9.6	Úvodná obrazovka v prostredí AIR . . . . .	54
9.7	Modul mapy v prostredí AIR . . . . .	54
9.8	Modul budovy v prostredí AIR . . . . .	54
A.1	Preklad exportovaných dát formátu CSV . . . . .	64

# 1 ÚVOD

Káblové rozvody už dávno nepredstavujú jedinou možnosť prenosu dát medzi elektronickými zariadeniami. S pokrokovejšími technológiami sú kladené nároky na mobilitu predovšetkým koncových zariadení. Signál je možné šíriť na veľké obmedzené vzdialenosti nehmotným bezplatným médiom - vonkajším prírodným prostredím.

Bezdrôtové siete sa zbavili nutnosti vedenia nekonečne dlhých káblov, ktoré sa stávajú s narastajúcim počtom elektronických zariadení značne nepraktickými. Objavenie možnosti ovládať zariadenia nie len cez káblové rozvody otvorilo cestu k efektívnejšiemu a pohodlnejšiemu užívaniu moderných elektronických prístrojov. Na druhej strane však potreba monitorovania pripojených koncových staníc do danej siete zostáva relevantná.

Druhov bezdrôtových technológií je mnoho, spôsobov využitia ešte viac. Ako príklad je dobré zmieniť už takmer všade prítomné Wifi siete, siete mobilných operátorov, Bluetooth, či senzorové siete.

Podstatným faktorom a úskalím bezdrôtových sietí je dosah. Signály nie je možné šíriť na nekonečné vzdialenosti, vo väčšine prípadov to ale nie je ani žiadané. Pri návrhu siete musíme mať predstavu o ploche, resp. priestore, v ktorom budú uzly operovať. V prípade desiatok, stoviek až tisícov jednotiek je administrácia siete bez vhodných softvérových nástrojov nezvládnuteľná.

Praktickú časť diplomovej práce tvorí budovanie systému, ktorý poskytuje komfortný prehľad o stave, vlastnostiach a fyzickej polohe každého prvku siete. Systém rieši rozloženie siete na rozsiahlej ploche, k čomu sú využívané reálne mapové podklady Google Maps, a rozmiestnenie sieťových prvkov vo vnútri uzatvoreného priestoru, napríklad vo viacposchodovej budove.

Bola navrhnutá XML štruktúra, v ktorej sú obsiahnuté jednoznačné údaje o každom bezdrôtovom prijímači, či vysielačej stanici v sieti. Štruktúra je použitá jednak pri spracovávaní vstupných dát a jednak pri exporte dát výstupných.

Aplikácia je napojená na školský databázový systém za účelom dynamického sledovania pohybu bezdrôtových, hlavne senzorových jednotiek v objekte budovy.

Písomná časť rozoberá tématiku bezdrôtových sietí, technológiu Adobe Flex SDK<sup>1</sup> slúžiacu k vývoju softvérového modelovacieho nástroja, implementáciu ďalších technológií (Google Maps SDK, jazyk ActionScript, jazyk MXML, XML). Okrem toho sú priblížené postupy a riešenia záležitostí týkajúcich sa zdrojového kódu programovanej aplikácie. V neposlednej rade bola pozornosť venovaná názorným grafickým ukážkam a popisu ovládania výsledného produktu.

---

<sup>1</sup>SDK (*Software Development Kit*) – sada nástrojov pre vývoj softvérových produktov

## 2 BEZDRÔTOVÉ SIETE

### 2.1 Úvod do bezdrôtových sietí

Bezdrôtová jednotka je typicky vybavená rádiostanicou, či iným bezdrôtovým komunikačným zariadením schopným daný signál prijímať a vysieľať, procesorovej jednotky, pamäte a zdrojom energie, ktorým sú väčšinou batérie. V prípade sietí slúžiacich na účely pozorovacie sú uzly vybavené snímačom - senzorom. Zariadenia sietí určené k pripojeniu užívateľov disponujú obrazovým výstupom a ovládacou konzolou.

Centrálna stanica je špeciálna komponenta. K behu celého systému je nevyhnutná. V závislosti na účelu siete sprostredkováva signál, spracováva, vyhodnocuje a preposiela dáta, zabezpečuje pripojenie, komunikuje s operátorom, atď.. Centrálnych staníc môže byť v sieti tiež viacero.

Uzly v sieti komunikujú rôznymi spôsobmi. Nie vždy je prijateľné, aby bola každá stanica priamo spojená s tou centrálnou. Niekedy je to naopak nevyhnutné. V prípade senzorových sietí, pri predstave tisícov senzorov, je scenár počítajúci s individuálnou komunikáciou s každým uzlom zvlášť nereálny.

Senzorové systémy pozostávajú z kolekcie priestorovo distribuovaných autonómnych uzlov určených k sledovaniu fyzických alebo environmentálnych podmienok. Prvky navzájom spolupracujú a tvoria tak jednoducho realizovateľný systém, nenáročný na údržbu a náklady. Počiatok vývoja bol motivovaný armádnymi aplikáciami, prevažne kvôli pozorovaniu bojového pola. V súčasnej dobe tvoria senzorové systémy bežnú súčasť priemyselných objektov či domácností. Bezdrôtová senzorová sieť predstavuje tzv. Ad-hoc sieť. Uzly komunikujú vzájomne medzi sebou. Nie je nutné aby existovala cesta od centrálnej stanice ku každému jednému uzlu zvlášť. Namiesto toho sú uzly implementované väčšinou do mriežkovej topológie - čiastočnej, či úplnej. Smerovacie algoritmy protokolov typu bod-bod prikazujú uzlom plniť rolu preposielateľa dát, ktoré putujú jednou z viacerých možných ciest až do základnej stanice. Tu sú spracovávané a vyhodnocované. V prípade zlyhania uzlu, bude cesta, na ktorej uzol leží vylúčená, automaticky sa vyhľadá ďalšia možná a smerovacie tabuľky sa aktualizujú. Prenos dát pokračuje prakticky bez obmedzenia ďalej pôvodným smerom ale inou cestou. Týmto spôsobom sa pri výpadku jedného prvku predíde zlyhaniu celej siete. Je nutno dodať, že čím viac spojení existuje, tým je systém energeticky náročnejší. Optimálnym riešením je zabezpečiť viacero ciest, tak aby existovala varianta v prípade výpadku nejakého uzla a zároveň v sieti nefigurovala prílišná redundancia [1].

## 2.2 Nároky bezdrôtových zariadení

Bezdrôtové elektronické zariadenia sú navrhované za účelom realizácie určitých úkonov, ktoré potenciál danej siete podpora alebo využijú. Senzory môžu byť implementované v náročných človekom neznesiteľných environmentálnych podmienkach, zariadenia slúžiace k pripojeniu na internet sú zase citlivé na stabilitu signálu. Podľa zmyslu implementácie je nutné hľadať vyhovujúce vlastnosti a vážiť kompromis.

- Veľkosť – S rozvojom nanotechnologických výrobných procesov sa rozmery zariadení znižujú až na veľkosť zrnka prachu. Samozrejme nie vždy je miniaturizácia žiadaná.
- Cena – Obmedzenie investícií popri obmedzení veľkosti znamená zníženie výpočtovej rýchlosti, využiteľnej šírky pásma, energetickej, či pamäťovej kapacity.
- Energetická náročnosť – Spotreba patrí k rozhodným vlastnostiam. Existuje rozdiel v prístupe užívateľov z hľadiska pozornosti, ktorú venujú zariadeniam. Bezdrôtové senzory musia pracovať samostatne, bez dozoru, dlhodobo. Osobné počítače a telefóny do istej miery využívajú určitú pravidelnú starostlivosť a tak ich dobíjanie nepredstavuje príliš veľký problém. Bezdrôtové zariadenia s rôznym účelom v tomto smere prakticky nie je možné porovnávať. Nie len prístup k nim je rozdielny. Iná je aj tolerancia veľkosti batérií a dáta, s ktorými pracujú (veľkosť, typ, náročnosť na spracovanie, požiadavky na kvalitu prenosu).
- Pokrytie – V zásade ide o pokrytie čo najväčšej plochy čo najkvalitnejším signálom s minimálnymi nákladmi viazanými na počet a rozmiestnenie staníc.
- Podpora protokolov – Štandardy závisia na oblasti použitia. Príklady: ISA100, IEEE 1451, ZigBee, 802.15.4, EnOcean, IETF RPL, BlueTooth, Wifi. O využití komunikačných protokolov v aplikáciách sa zatiaľ neuvažuje, preto by ich hlbší rozbor nebol účelný.
- Bezpečnosť – Zabezpečovacie mechanizmy, ako kódovanie vysielacieho kanálu, kvalita prenosu alebo šifrovanie spracovávaných dát, musia byť pri prenose citlivých informácií premyslené.

Do úvahy ďalej musí byť braná flexibilita uzlu a reakcia na výpadky. Výsledným kompromisom je teda voľba medzi úrovňou kvality pokrytia a prenosu, výškou energetickej náročnosti, spôsobe implementácie a hranicou investičných nákladov.

## 2.3 Oblasti využitia

Informačné systémy nám sprostredkovávajú nekonečné možnosti. Bezdrôtové siete sú toho príkladom. Či už sa jedná o zariadenia osobné, priemyselné, biomedicínske, armádne, poľnohospodárske, táto technológia sa uplatní v akejkoľvek smere ľudskej potreby.

- Senzorové siete

V oblasti bezpečnosti sa používajú senzory reagujúce na určitý jav (požiar, zosuv pôdy, prítomnosť chemických látok v ovzduší, výška vodnej hladiny, miera vlhkosti, teplota). Senzor na základe podmienok okolitého prostredia reaguje a v prípade zistenia výskytu nežiadúceho javu odosiela varovanie alebo zahajuje kroky k potlačeniu daného stavu (spustenie trysiek požiarneho systému, otvorenie okien, a iné).

V armáde senzorové systémy zisťujú nie len prítomnosť a pohyb nepriateľa v bojovom poli.

Environmentálne monitorovanie je väčšinou dočasné ale priebežné, určené k sledovaniu prírodných javov, života živočíšnej ríše.

V priemysle pomáhajú senzory udržiavať priemyselné stroje a zariadenia v potrebnom stave, znižujú tým náklady a poskytujú možnosť použitia v miestach neprístupných, nebezpečných alebo ľudskému organizmu neznesiteľných. Výroba je tak efektívnejšia, menej nákladná a menej náchylná k ľudským chybám.

Sledovanie dopravy senzorovými systémami je často rozšírené o polohovací systém GPS<sup>1</sup>, ktorý môže byť súčasťou dopravných prostriedkov firiem, prepravných spoločností, armád, súkromných vozidiel sledovaných bezpečnostnými agentúrami.

- Počítačové siete

V dnešnej dobe je každému známa sieť typu WLAN, ktorá umožňuje bezdrôtové pripojenie k digitálnemu svetu pre prácu aj zábavu.

Technológia Bluetooth slúžiaca k priamemu pripojeniu koncových užívateľov medzi sebou za účelom výmeny dát.

---

<sup>1</sup>GPS (*Global Positioning System*) – globálny polohovací systém

## 3 ADOBE FLEX

Nástroj použitý k vývoju aplikácie pochádza od spoločnosti Adobe Systems, ktorá ponúka širokú, pomerne populárnu paletu produktov. Ako príklad môžeme uviesť grafický editor Adobe Photoshop, či prehliadač digitálnych dokumentov Adobe Acrobat Reader.

Produkt Adobe Flex SDK je určený k budovaniu robustných, predovšetkým internetových klientských aplikácií. Je voľne šíriteľný, dostupný na stránkach výrobcu<sup>1</sup> a jeho zdrojový kód je otvorený. Pozostáva z Flex kompilátora a softvérovej štruktúry Flex framework. Aplikácia pre generovanie modelu bezdrôtovej siete je tvorená v prostredí Adobe Flash Builder 4 alebo tiež Adobe Flex Builder. V našom prípade bol prekladač implementovaný ako doplnok vývojového prostredia Eclipse.

Informácie v tejto kapitole sú čerpané z oficiálnych stránok spoločnosti Adobe Systems Incorporated[3].

### 3.1 Flex framework

Internetové aplikácie, zvané tiež RIA (*Rich Internet Applications*) programované v tomto jazyku sú podporované prakticky všetkými hlavnými webovými prehliadačmi, je však nutná inštalácia prehrávača Adobe Flash Player. To nie je veľký problém. Mnohé štatistiky ukazujú, že Adobe Flash Player je inštalovaný na viac ako 98% počítačov pripojených k internetovej sieti a na viac ako 800 miliónoch mobilných zariadení<sup>23</sup>.

Flex pozostáva z dvoch subtechnológií - MXML<sup>4</sup> a ActionScript 3.0. Jazyk MXML definuje vlastnosti a správanie užívateľského rozhrania pomocou výrazov vychádzajúcich zo značkovacieho jazyka XML. Objektovo orientovaný jazyk ActionScript slúži na tvorbu klientskej logiky.

Pri preklade zdrojového kódu je vytvorený tzv. SWF<sup>5</sup> formát. Referencia na neho je vložená do súboru HTML uloženom na serveri. Klient odošle požiadavku na server, ktorý po získaní dát a služieb z databázy vygeneruje HTML kód. Webový prehliadač HTML kód preloží a odošle dotaz na získanie SWF súboru. Ten je stiahnutý do

---

<sup>1</sup><http://www.adobe.com/cfusion/entitlement/index.cfm?e=flex4sdk>

<sup>2</sup><http://www.adobe.com/devnet/swf.html>

<sup>3</sup><http://www.pcpro.co.uk/blogs/2009/02/20/99-percent-flash-player-penetration>

<sup>4</sup>MXML - nemá oficiálny preklad. Pravdepodobne označuje *Macromedia eXtensible Markup Language*, tj. rozšíriteľný značkovací jazyk so špecifickými značkami definovanými skupinou Macromedia Studios

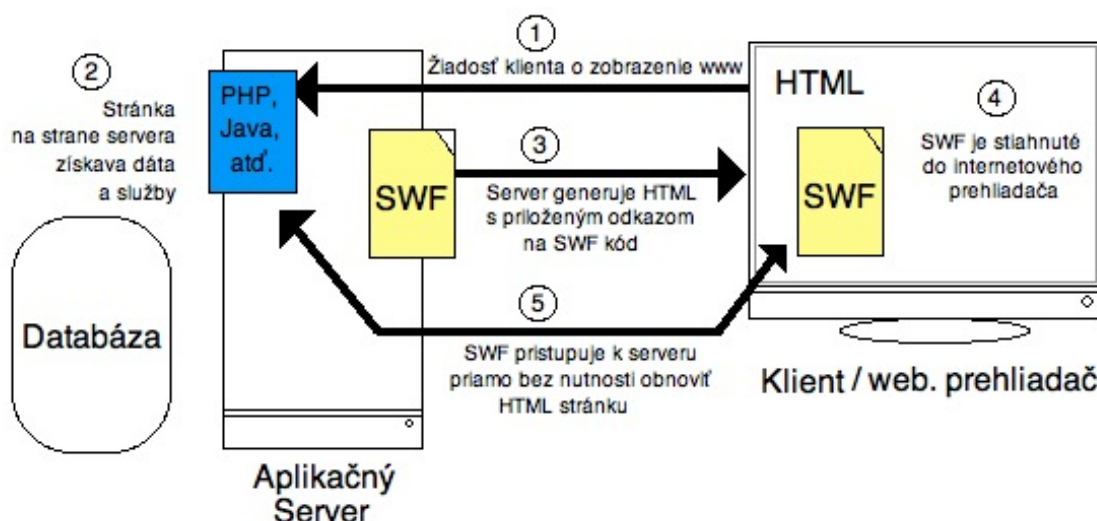
<sup>5</sup>SWF (*Small Web Format*) – obsahuje informácie o vektorovej grafike, texte, videu, zvuku, je podporovaný softvérom Adobe Flash Player a Adobe Air

pamäte a poskytnutý programu Flash Player. SWF súbor je vďaka kompilátoru Flash Builder do HTML kódu vložený automaticky JavaScript kódom:

```
swfobjectembedSWF ("applicationFile.swf","flashContent",
    "100\\%", "100\\%", swfVersionStr, xiSwfUrlStr, flashvars,
    params, attributes);
```

Príkaz je volaný automaticky a nebude mu ďalej venovaná pozornosť. Je vhodné všimnúť si, že technológie HTML, CSS, JavaScript a Flex dokážu vzájomne spolupracovať.

Akonáhle je SWF uložený do pamäte na strane klienta, klient komunikuje s aplikáčnym serverom priamo, bez nutnosti opakovaného obnovovania HTML stránky. Situácia je graficky znázornená na obrázku 3.1.



Obr. 3.1: Spôsob komunikácie Flex aplikácie so serverom.

Touto logikou sú dáta načítané dynamicky na podnet udalostí vyvolaných koncovým užívateľom. Výhodou je interaktivita, rýchlosť, redukcia počtu odosielaných žiadostí smerom k serveru a množstvo dát nutných k opakovanému prenosu.

Flex disponuje vlastnou knižnicou s viac ako sto rozšíriteľnými komponentami užívateľského rozhrania. V práci je využívaná najnovšia verzia Flex 4.

## 3.2 Adobe Flash Builder

Adobe Flash Builder 4 urýchľuje a zjednodušuje vývoj Flex projektov. Spolupracujúc s vývojovým prostredím Eclipse poskytuje možnosť prehľadného komponovania kódu s pomocou nápovedy, interaktívneho ladenia a vizuálneho návrhu programu



vo zvláštnom návrhovom režime. Obsahuje Flex framework spolu s kompilátormi a ladiacimi algoritmami. Inštalácia možná samostatne alebo ako doplnok programátorského prostredia Eclipse. Dostupný je v dvoch edíciách - Standard a Premium.

## 4 ADOBE AIR

Vďaka technológiám Adobe AIR (*Adobe Integrated Runtime*) nemusí byť užívateľ odkázaný iba na internetové prehliadače. Toto integrované prostredie umožňuje vývojárom tvoriť samostatné klientské aplikácie, ktorých cieľová platforma nie je obmedzená len na konkrétny operačný systém alebo zariadenie.

Adobe AIR nahradzuje prehliadač a ponúka rovnaké vývojové a implementačné prostriedky ako v prípade sady nástrojov FLEX. Znamená to, že zdrojový kód jazyka FLEX je možné jednoducho použiť pri tvorbe desktopového AIR projektu. Výsledkom je inšalačný súbor, do ktorého je exportovaný zdrojový kód. Prekladač Flash Builder dáva na výber vytvorenie inšalačného balíčku vo formáte AIR alebo natívneho inšalátoru pre daný operačný systém. V oboch prípadoch je nutné použiť existujúci certifikát alebo vytvoriť nový s menom vydavateľa a ďalšími informáciami udávajúcimi vlastníctvo a pôvod. K aplikácii je ďalej pridané heslo a čas vzniku. Po potvrdení je vytvorený inšalačný súbor digitálne podpísanej aplikácie.

Výhody AIR[5]:

- inšalovaná aplikácia je rýchlejšia
- možnosť aktualizácie softvérových produktov
- inšalovateľné a použiteľné na rôznych platformách a zariadeniach (tablety, smartphony, PDA, operačné systémy iOS, Android a ďalšie)
- aplikácia môže pracovať na pozadí a informovať o udalostiach upozorneniami
- možnosť práce offline, ukladanie dát lokálne
- aplikácia má voľnejší prístup k lokálnemu súborovému systému
- upraviteľné grafické rozhranie

Jednou z nevýhod je, že je nutné, aby bola v operačnom systéme zariadenia táto technológia inšalovaná. Je dodávaná zdarma napríklad automaticky s inšaláciou Adobe Reader alebo je k dispozícii samostatne, na oficiálnych stránkach Adobe.

Ďalším záporným bodom je nevyhnutnosť inšalácie samotnej aplikácie. Zatiaľčo softvér prístupný cez webový prehliadač nevyžaduje inšaláciu, pracuje vzdialene na serveri, desktopová AIR aplikácia musí byť v konečnom kroku vývoja zabalená do inšalačného súboru a digitálne podpísaná. Inšalačný proces je ale bezproblémový, vyžaduje len potvrdenie a výber cieľového umiestnenia.

## 5 GEOGRAFICKÉ VYJADRENIE POLOHY

Povrch Zeme je pokrytý imaginárnou sieťou čiar poludníkov a rovnobežiek. Rovnobežky sú horizontálne orientované úsečky, kružnice, s rovnakou vzdialenosťou jednej od druhej. Každý stupeň predstavuje približne 111 kilometrov, čiže rovnobežky sú od seba 111 kilometrov ďaleko. Presnú vzdialenosť je ťažké určiť, nakoľko Zem nie je, vplyvom otáčania sa okolo svojej osy, presného guľového tvaru. Podľa rovnobežiek sú posudzované súradnice šírky. Číselné hodnoty sa pohybujú v rozmedzí -90 až 90 stupňov, pričom nultou rovnobežkou je rovník.

Poludníky sú vertikálne orientované úsečky, pólkružnice. Spájajú body severného a južného pólu. Ich vzájomná vzdialenosť nie je konštantná, smerom k rovníku sa zväčšuje. Od nultého poludníka na západ hodnota súradnice dĺžky klesá až na -180 stupňov. Smerom na východ naopak stúpa k hodnote +180 stupňov.

Každý jeden bod na povrchu zemskej plochy leží v priesečníku týchto dvoch krieviek. Priesečník je daný súradnicami šírky a dĺžky. Môžu byť vyjadrené reálnymi číslami, ktoré predstavujú stupne, minúty a sekundy v desiatkovej číselnej sústave. Formát stupňov, minút a sekúnd je najpoužívannejším. Formát jedného typu je samozrejme prevediteľný na formát typu druhého. Podrobnosti boli získané z [11].

Aplikácia dokáže pri modelovaní bezdrôtovej siete určiť ktorýkoľvek bod na Zemi a vyjadriť tak polohu bezdrôtových zariadení s vysokou presnosťou. Používa k tomu nástroj *Google Maps SDK* a jeho súradnicový systém.

V zdrojovom kóde aj v ovládacích prvkoch programu sa používajú anglické výrazy *latitude*, pre súradnicu šírky, a *longitude* pre súradnicu dĺžky. Google Maps SDK pri práci s polohou bezdrôtových jednotiek počíta so súradnicami vyjadrenými desatinnými číslami (reálnymi číslami) namiesto počítania so stupňami, minútami a sekundami. Je to praktickejšie a z vývojárskeho pohľadu jednoduchšie.

### 5.1 Google Maps SDK

Významnú pomoc pri programovaní aplikácií, ktoré pracujú s digitálnymi mapovými podkladmi firmy Google, predstavuje adresa [code.google.com](http://code.google.com). Adresa je oficiálnou vývojovou stránkou spoločnosti, prístupná komukoľvek. Obsahuje dokumentácie, príklady, manuály, typy, zoznamy príkazov a mnohé iné. Nachádzajú sa tu API<sup>1</sup> pre takmer všetky produkty súčasného zábavného internetového priemyslu, napr. Google Maps, Picasso, YouTube, GoogleApps, GoogleEarth, či nástroje pre programovanie pre operačný systém Android a iné [6]. Google Code poskytuje podporu

---

<sup>1</sup>API (*Application Programming Interface*) – Rozhranie pre programovanie softvérových aplikácií

viacerým programovacím jazykom. Pri práci s mapami majú informácie na tejto internetovej adrese neoceniteľný prínos a v práci sú námety riešení nachádzané často práve tu ([7]).

Pre implementáciu technológie mapového podkladu do ktorejkoľvek aplikácie je nutné uskutočniť určité kroky. Sú nimi:

- Registrácia Google Maps API – získanie unikátneho kľúča
- Inštalácia Google Maps API pre Flex SDK
- Implementácia Google Maps API do zdrojového kódu

### 5.1.1 Registrácia Google Maps API

Sú dané presné podmienky pre pripojenie máp Googlu k bežnej www doméne. Jednou z nich je získanie unikátneho API kľúča – *Google Maps API key*[8]. Ten je platný v rámci jedinej domény, či zložky. API kľúč jednoznačne identifikuje server, z ktorého je aplikácia spúšťaná. Pre každú IP adresu musí existovať unikátny textový reťazec, bez ktorého je aplikácia používajúca technológiu Google Maps síce spustiteľná ale nepoužiteľná. Ak je kľúč vygenerovaný a použitý správne užívateľa absolútne neovplyvňuje. V opačnom prípade sa na mieste mapovej plochy zobrazí varovanie o probléme s chybným API kľúčom.

K získaniu kľúča je nutné prihlásiť sa na účet Google, resp. registrovať nový. Pred odoslaním žiadosti o kľúč treba zadať URL adresu www stránky, pre ktorú je kľúč vytváraný, tj. adresa, kde budú služby Google Maps používané. Nasledujúcim krokom je odsúhlasiť licenčné podmienky.

Počet zobrazení stránky, teda zobrazení mapy, je neobmedzený, obmedzený je ale počet žiadostí o tzv. geokód. Geokódovanie je proces hľadania geografickej polohy na základe geografických informácií ako adresa ulice, poštové smerovacie číslo, atď., kedy je využitý algoritmus prevodu textového reťazca (adresy) na súradnice GPS. Je povolených maximálne 2500 žiadostí prijatých z jednej IP adresy za deň. Ak je tento limit dosiahnutý alebo ak je frekvencia odosielaných žiadostí príliš vysoká, Google Maps API geokóder vráti chybový status. Ak takéto správanie pokračuje, môže dôjsť k zablokovaniu danej IP adresy[9]. Google Maps API nepodporuje inzerciu a reklamy, služby www stránky musia byť volne prístupné, je zakázané meniť logá a atribúty mapových podkladov Google. Kompletné licenčné podmienky je možné nájsť na [8] a zoznam často kladených otázok, kde sú položené a zodpovedané praktické dotazy, na [9].

Vygenerovaný kľúč bude uschovaný a pripojený k danému Google účtu.

### 5.1.2 Inštalácia a implementácia

Ďalšou nutnosťou je získanie Google Maps API pre Flash SDK<sup>2</sup>. Je to knižnica rozhrania zabalená do súboru zip. Kópia obsahu archívu musí existovať v adresári knižníc vyvíjanej aplikácie.

Ďalej je v zdrojovom kóde bezpodmienečne potrebné nastaviť vlastnosť **sensor** a hlavne API kód.

Hodnota parametra **sensor** nastavená na hodnotu **true** vyjadruje, že bude služba počítat s využitím polohovacieho zariadenia GPS. Táto vlastnosť je nastavená na **false**:

```
<maps:Map sensor = "false"/>
```

Implementácia API kľúča je realizovaná nasledujúcim MXML kódom:

```
<maps:Map key = "ABQIAAAAAAt15scvA..." />
```

---

<sup>2</sup><http://maps.googleapis.com/maps/flash/release/sdk.zip>

## 6 BUDOVANIE APLIKÁCIE

Nároky, ktoré kladú užívatelia súčasnej elektronickej doby, sa neustále zvyšujú. Softvér musí byť okrem iného dostatočne intuitívny, univerzálny a tiež multiplatformný.

Rovnaký zámer má aj aplikácia programovaná v tejto práci. Užívateľ k nej má prístup cez internetové rozhranie webového prehliadača, kedy je využitá podpora multimediálneho prehrávača Adobe Flash Player, alebo ju môže spustiť ako samostatný program, na ktoromkoľvek počítači, či mobilnom zariadení prakticky kdekoľvek. Túto možnosť sprostredkováva technológia Adobe AIR. Vo výsledku je použitie produktu flexibilné v maximálnej možnej miere.

Štruktúra aplikácie je modulárna. Je rozdelená do dvoch hlavných častí, samostatných modulov. Moduly pracujú nezávisle na sebe a je možná ich voľná implementácia do ďalších projektov. Daný modul je volaný až počas behu programu a to len v prípade potreby, inak s ním aplikácia nepracuje. To kladie menšie nároky na výpočtovú náročnosť a systémové prostriedky.

V zdrojovom kóde projektu sú Flash Builderom preddefinované mxml značky :

```
<?xml version="1.0" encoding="utf-8"?>
<s:Application xmlns:fx="http://ns.adobe.com/mxml/2009"
               xmlns:s="library://ns.adobe.com/flex/spark"
               xmlns:mx="library://ns.adobe.com/flex/mx">
  <fx:Declarations>
    <!--Place non-visual elements
          (e.g., services, value objects) here -->
  </fx:Declarations>
</s:Application>
```

V prvej značke je definovaná verzia XML a kódovanie písma.

Nasledujúca párová značka `<s:Application>`, v prípade AIR aplikácie je to `<s:WindowedApplication>`, vymedzuje začiatok a koniec programu, môže byť použitá maximálne raz. Na tomto mieste sú definované xmlns knižnice *fx*, *spark* a *mx*. Každá obsahuje rôzne komponenty, ich symboly stoja vždy pred názvom komponenty. Rozhodovanie, ktorú knižnicu použiť, môžeme riešiť za pomoci interaktívnej nápovedy Flash Buildera<sup>1</sup>. Nápoveda pomáha aj pri volení samotnej komponenty, či metódy, funkcie, vlastnosti.

Miesto pre definíciu nevizuálnych prvkov, tj. premenných, ich vlastností a služieb, je vyhradené elementom `<fx:Declarations>`, resp. `</fx:Declarations>`.

---

<sup>1</sup>klávesová skratka nápovedy - *Ctrl + medzerník*

Produkty vyvinuté v Adobe Flex nepracujú úplne rovnako ako bežné www stránky. Na server je uložený už vopred skompilovaný zdrojový kód, ktorý väčšinou tvorí iba jeden primárny súbor. Užívateľ sa nepohybuje prostredníctvom odkazov na iné stránky, ale k www adrese pristupuje prakticky ako k bežnému programu. Navigácia v rozsiahlejšom obsahu musí byť riešená rôznymi komponentami alebo stavmi.

## 6.1 Stavy

Stavy, teda *States*, sa používajú k prepínaniu pohľadov. V rôznych stavoch môže byť rôzny obsah, či rozdielne grafické rozhranie. Dá sa povedať, že stavy reprezentujú odkazy, sú však efektívnejšie. Nie je nutné riešiť predávanie dát z jednej adresy na inú, vlastnosti komponent všetkých stavov je možné definovať na jednom mieste v zdrojovom kóde jediného súboru - nastavením stavu priamo u vlastnosti, ktorá platí práve daný prvok.

Aplikácia pre generovanie modelu bezdrôtovej siete pracuje s tromi hlavnými stavmi:

- Starter - primárny pohľad zobrazený práve po spustení aplikácie, sú z neho volané ďalšie stavy
- MapState - zobrazenie modulu modelovania siete na mapovom podklade
- BuildingState - zobrazenie modulu modelovania siete v plánoch budovy

Deklarácia:

```
<s:states>
  <s:State name="Starter"/>
  <s:State name="MapState"/>
  <s:State name="BuildingState"/>
</s:states>
```

Komponenty sú do stavov vkladané a odoberané vlastnosťami `includeIn=""` a `excludeFrom=""`. Ak je jedna komponenta potrebná vo viacerých stavoch, rozdielne vlastnosti sú pre každý stav odlíšené bodkovou notáciou (`width.Starter="80"`).

## 6.2 Moduly

Princíp modulárneho vývoja spočíva v tom, že sa časť zdrojového kódu skompiluje samostatne. Výsledkom je SWF súbor, ktorý je volaný hlavnou aplikáciou. Tento súbor nie je samostatne spustiteľným objektom. Po načítaní však pracuje nezávisle na hlavnom programe. Chyba modulu neohrozí chod celého systému.

Aplikáciu tvoria dva moduly. Jeden modul slúži na modelovanie siete na širokom aplikáciou neobmedzenom území. Jednotky siete sú rozmiestňované na mape, takže je možné sledovať sieť rozloženú kdekoľvek na svete. Vďaka druhému modulu je užívateľ schopný modelovať a monitorovať sieť vo vnútri uzatvoreného objektu. Objektom môže byť viacposchodová budova, ku každému poschodiu potom aplikácia pristupuje ako k osobitnej pracovnej ploche.

Modul je možné vybrať na úvodnej obrazovke hneď po spustení projektu. Medzi modulami je možné sa kedykoľvek prepínať prostredníctvom tlačítka. Aktuálny modul je odhlásený, aby sa šetrili systémové prostriedky v čase, kedy nie je využívaný, a je načítaný modul druhý. Je nutné pamätať, že prepnutím modulu sú stratené všetky neexportované dáta.

Používané moduly sú definované *mx* komponentom *ModuleLoader*:

```
<mx:ModuleLoader id="ModuleID"
                url.MapState="mapModule.swf"
                url.BuildingState="buildingModule.swf"/>
```

Parameter *url* ukazuje na názov a umiestnenie SWF súboru. Pre každý stav je pripravený zvláštny modul.

Funkcie pre volanie a rušenie modulu, kde vstupným parametrom je id komponenty *ModuleLoader*:

```
public function showModule(mLoader:ModuleLoader):void{
    mLoader.loadModule();
}
public function removeModule(mLoader:ModuleLoader):void{
    mLoader.unloadModule();
}
```

Vývoj samotného modulu sa od vývoja klasickej aplikácie líši v zásade iba zmenou značky `<s:Application>` na `<mx:Module>`.

## 6.3 XML štruktúra

Jazyk XML bol vytvorený hlavne za účelom prenosu informácií medzi nehomogénnymi systémami v štandardizovanom formáte.

V tejto práci predstavuje jazyk XML, resp. obsah, ktorý nesie, bránu medzi reálnym svetom a digitálnym grafickým výstupom. Štruktúra je definovaná tak, aby vyjadrila hustotu a vlastnosti siete výstižne a komplexne. Dáta, ktoré nesie, definujú rozloženie bezdrôtovej siete, jej konkrétne stanice a ich parametre<sup>2</sup>.

---

<sup>2</sup>Príklad návrhu štruktúry dát, s ktorými aplikácia pracuje je uvedený v prílohách



Je nutné dodať, že nakoľko je aplikácia modulárna, každý modul pristupuje k modelovanej sieti z rôznych pohľadov, pričom využíva rozdielne pozičné systémy. Taktiež môžu byť rôzne aj popisované vlastnosti charakterizujúce členské jednotky vzhľadom ku konečným požiadavkom a účelu využitia modulu. V každom prípade bol na túto skutočnosť braný ohľad a nie je nevyhnutné používať rôzne štruktúry pre každý modul zvlášť. Údaje pre všetky moduly je možné niesť v jedinom XML reťazci. Podmienkou správnej práce aplikácie však je, aby XML obsahovalo minimálne tie parametre, ktoré sú určené pre modul, s ktorým chceme pracovať. Nasleduje konkrétnejší popis navrhutej štruktúry:

- element **network** je hraničnou značkou vymedzujúcou začiatok a koniec XML štruktúry, tj. začiatok a koniec množiny všetkých jednotiek v sieti
- element **node** zahrňuje informácie o vlastnostiach konkrétneho uzla

Parametre uzlu rovnaké pre oba moduly:

- **name** - názov bezdrôtového uzlu
- **nodeID** - jedinečný identifikátor bezdrôtového uzlu
- **serialNum** - sériové číslo bezdrôtového uzlu
- **macAdd** - mac adresa hardwaru zariadenia
- **type** - typ bezdrôtového zariadenia, na základe ktorého aplikácia rozlíši, či sa jedná o prijímaciu alebo vysielačnú stanicu

Parametre uzlu používané modulom, ktorý modeluje sieť na rozsiahlom území:

- **latitude** - GPS súradnica v jednotkách stupňov, na základe ktorej je nájdená presná horizontálna poloha zariadenia
- **longitude** - GPS súradnica v jednotkách stupňov, na základe ktorej je nájdená presná vertikálna poloha zariadenia
- **scope** - hodnota dosahu signálu bezdrôtového uzla v metroch

Parametre uzlu používané modulom modelujúcim sieť v uzatvorenom objekte:

- **positionX** - horizontálna vzdialenosť od počiatku súradnicovej sústavy plánu udávaná v metroch
- **positionY** - vertikálna vzdialenosť od počiatku súradnicovej sústavy plánu udávaná v metroch
- **positionZ** (nepovinné) - výška umiestnenia senzora v miestnosti pre náhľad z bočnej strany budovy, udávaná v metroch
- **floorNum** - číslo poschodia budovy, na ktorom sa daná jednotka nachádza
- **mobility** - zariadenie je buď mobilné alebo statické
- **timeStmp** - čas a dátum poslednej zmeny polohy zariadenia

Návrh XML bol ovplyvnený štruktúrou dát databázového systému. Za účelom zachovania prehľadnosti bola snaha o štandardizáciu názvov a parametrov aby sa pracovalo s rovnakými vlastnosťami, či už boli dáta získané z databázy alebo z XML súboru.

## 6.4 HTTPService

Jedným zo spôsobov, akým získavať dáta z externých súborov počas behu programu, je služba *HTTPService*. Vďaka parametru `url` triedy *HTTPService*, je možné poslať HTTP požiadavky na konkrétnu adresu. Adresou môže byť umiestnenie súboru na disku lokálneho počítača alebo server kdekoľvek na svete.

Odoslanie požiadavky sa koná prostredníctvom metódy `send()`, odpoveďou je často napríklad XML štruktúra. Kvôli spracovaniu výsledku dotazu musí byť v tejto triedy definovaná reakcia na udalosť `result`. Reakciou je zavolanie funkcie typu *Event Handler*, ktorá prevedie XML štruktúru na pole objektov. Aplikácia takúto štruktúru spracuje a ďalej využije.

Trieda *HTTPService* nemusí dáta len získavať. Pracuje sa s ňou rovnako ako s klasickými HTTP dotazmi (GET, POST, atď). Vhodne zvoleným parametrom `url` je teda možné dáta aj poslať.

Je nutné brať do úvahy, že takáto komunikácia je otvorená, bez akýchkoľvek zabezpečovacích metód, takže pre predávanie citlivých dát by bolo vhodnejšie zvoliť iný spôsob.

## 6.5 Ťahaj a puš'

Metóda, ktorej význam nie je nutné predstavovať, sa často aj v češtine, či slovenčine označuje *Drag&Drop*. Jedná sa o interakciu užívateľa s aplikáciou prostredníctvom práce s myšou.

V aplikácií slúži metóda k pohodlnejšej a praktickejšej správe rozmiestnenia prvkov modelu siete.

Rozlišujú sa viaceré úkony, ktoré spadajú do skupiny udalostí - *DragEvent*:

- *Drag\_START* – bolo kliknuté na objekt a bol zahájený jeho presun (ťahanie)
- *Drag\_OVER* – presúvaný objekt sa pohybuje cez miesto iného objektu
- *Drag\_ENTER* – presúvaný objekt je na mieste iného objektu, ktorý ho očakáva
- *Drag\_EXIT* – presúvaný objekt opustil hranice daného objektu
- *Drag\_DROP* – objekt bol uvoľnený
- *Drag\_COMPLETE* – všetky operácie presunu boli úspešne skončené

Tieto úkony sú priradené konkrétnym objektom ako tzv. *EventListeners*<sup>3</sup>, to jest funkcie čakajúce na konkrétnu činnosť užívateľa.

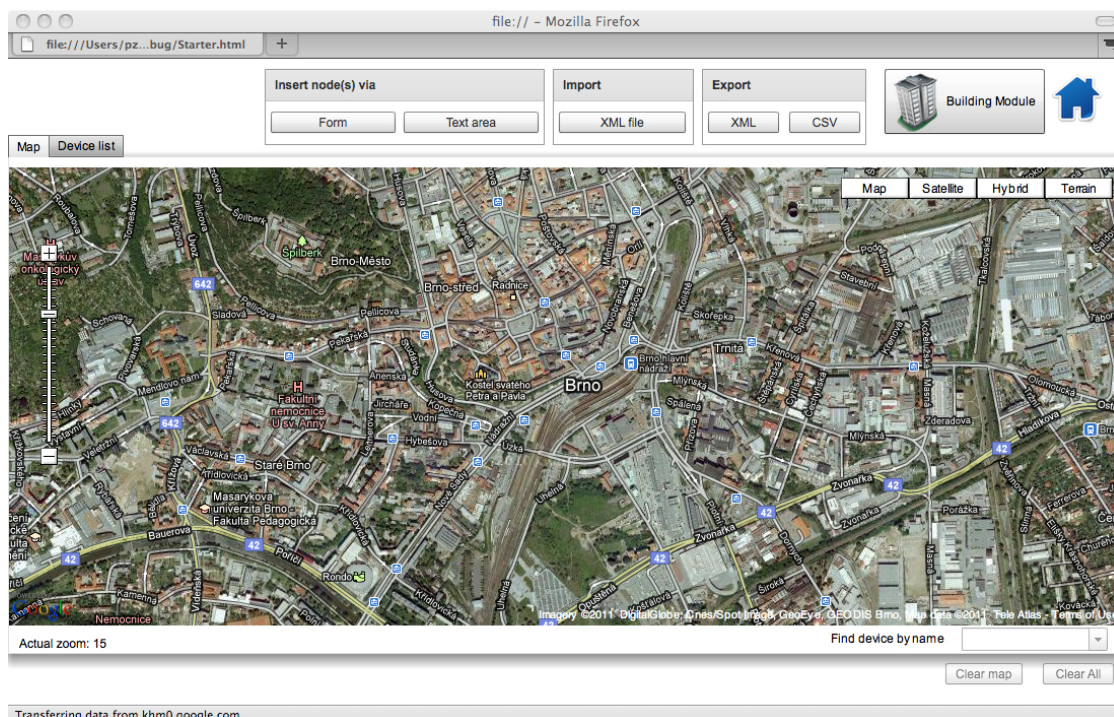
---

<sup>3</sup>Funkciám *EventListeners* bude väčšia pozornosť venovaná v ďalších kapitolách.

## 7 MODUL MÁP

Užívateľské rozhranie modulu je definované v mxml súbore *mapModule.mxml* a funkcie jazyka ActionScript v *MapActionScript.as*.

Úmyslom rozloženia komponent je ponúknuť čo najväčší priestor mapovému podkladu, keďže ten bude pracovným priestorom celého nástroja. Ukážka užívateľského rozhrania je na obrázku 7.1.



Obr. 7.1: Základné užívateľské rozhranie

Vo vertikálnom strede je mapa rozťahnutá na celú dĺžku. Nasledujúcim kódom<sup>1</sup> je vložená do záložky *Map* mx komponentu *TabNavigator*:

```
<mx:TabNavigator>
    <s:NavigatorContent label="Map">
        <maps:Map/>
    </s:NavigatorContent>
    <s:NavigatorContent label="Device list">
        <mx:DataGrid/>
    </s:NavigatorContent>
</mx:TabNavigator>
```

<sup>1</sup>Definícia mxml elementov nie je vždy uvedená so všetkými vlastnosťami a volaním funkcií, či ďalších subkomponentov. Účelom je ukázať hlavne konkrétne popisované riešenie.

V jednom okamžiku je možné zobraziť grafické rozmiestnenie uzlov na mapovom podklade alebo zoznam vložených zariadení v tabuľke *DataGrid* (<mx:DataGrid>). Ďalšie prvky rozhrania reagujú na udalosti vyvolané užívateľom. Sú nimi tlačítka, ktoré volajú funkcie, a roletové menu. Samotná mapa je podobne citlivá na rôzne podnety – kliknutie, prechádzanie kurzorom myši, drag&drop.

## 7.1 Komponent Map

Framework *Google Maps API SDK* rieši rôzne úkony vlastnou cestou a vlastnými príkazmi. Napríklad pre samotné vykreslenie elementu používa svoje unikátne objekty a vlastnosti. Spolu s definovaním knižnice *maps* ich je potrebné nastaviť:

```
<maps:Map id="map"
    xmlns:maps="com.google.maps.*"
    key="ABQIAA...R52Swo15Z6o4Q"
    mapevent_mappreinitialize="onMapPreinitialize(event)"
    mapevent_mapready="onMapReady(event)"
    sensor="false"/>
```

Z ukážky je patrné, že vygenerovaný kľúč sa použije vo vlastnosti `key`. Vlastnosť `sensor` rozhoduje o tom, či bude aplikácia používaná zariadením s GPS lokátorom. V tomto prípade sa o tejto variante neuvažuje, preto je jej priradená hodnota `false`. Identifikátor `id` slúži na to, aby sa na tento prvok mohli iné komponenty a funkcie odvolať – skrz textový reťazec typu `String`. Ďalej sú volané tzv. inicializačné funkcie, ktoré počas alebo po vzniku prvku volajú ďalšie metódy, určujú iné vlastnosti (priblíženie, typ mapy, stred mapy a iné) a vkladajú ďalšie kontrolné prvky, napríklad pre priblíženie, zmenu typu mapy. Funkcie budú ozrejmené v časti *Funkcie*, str. 29.

## 7.2 Použitie HttpService

Služba je definovaná v časti deklarácií. Okrem identifikátora a adresy dát je podstatná funkcia `myHttpService_resultHandler(event)`, ktorá riadi udalosť spracovania súboru.

```
<s:HTTPService id="myHttpService"
    url="XML_structure.xml"
    result="myHttpService_resultHandler(event)"/>
```

Spracovaním súboru sa v tomto prípade myslí konvertovanie XML štruktúry do globálnej premennej pola objektov `XMLnodes`. Skrátené to znamená, že je odoslaný

požiadavok na službu *HTTPService*, ktorý skrz parameter `result` spustí preklad algoritmu funkcie, ktorá vloží zoznam uzlov z XML do pola.

## 7.3 Ikony uzlov

Grafické ikony zobrazujúce polohu bezdrôtových uzlov sú externými PNG<sup>2</sup> obrázkami. Sú vložené prostredníctvom tried *Class* deklarovaných v ActionScript kóde zvlášť pre každú ikonu:

```
<fx:Script>
    [Embed(source="/WNModelingTool/images/wireless_icon.png")]
    private var markerIcon1:Class;
</fx:Script>
```

Triedy sú potom použité ako vlastnosť objektu mapového podkladu.

## 7.4 Funkcie modulu mapy

Aplikácie budované jazykom Flex sú riadené udalosťami. Ich správanie ovplyvňuje koncový užívateľ.

Funkcie, ktoré čakajú na danú udalosť sú označované ako *Event Listeners*, funkcie udalosťou spúšťané *Event Handlers*. Je určená postupnosť krokov, ktorá je spracovaná na základe vonkajšieho podnetu. Vonkajším podnetom môže byť iná funkcia, udalosť a v prípade *Event Handleru Event Listener*. *Event Listener* čaká na jednu konkrétnu udalosť, ktorá je viazaná k jednému objektu. Po vyskytnutí sa tejto udalosti je volaná funkcia *Event Handler*, ktorej vstupným parametrom je práve daná udalosť. Príklad definície nasleduje:

```
map.addEventListener(MapMouseEvent.CLICK, onMapClick);
```

Objektu `map` je priradený "naslúchač udalostí". Užívateľ ho kliknutím na plochu mapového podkladu aktivuje a ten v tomto prípade spustí celý sled úkonov. Funkcia `onMapClick` totiž na základe vstupného parametru zistí GPS súradnice kliknutého bodu<sup>3</sup> a zavolá funkciu `createPopUpForm()`:

```
private function onMapClick (event:MapMouseEvent):void{
    createPopUpForm();
    pop.onMapClickCoordinates = event.latLng;
}
```

---

<sup>2</sup>PNG (*Portable Network Graphics*) - grafický formát určený pre bezstratovú kompresiu rastrovej grafiky

<sup>3</sup>Stupne súradníc sú priradené objektu *pop* a ďalej spracovávané.

`createPopUpForm` vytvorí objekt, ktorému je priradený *Event Listener* čakajúci na ďalšiu reakciu užívateľa. Takto sa aplikácia, bez akejkoľvek obnovy www stránky, dynamicky mení a prispôsobuje okamžitým požiadavkom a konečnému účelu.

### 7.4.1 Inicializácia

Objekt mapového podkladu bol deklarovaný v *mapModule.mxml*. Objekt volá inicializačné funkcie, prostredníctvom ktorých nastavíme niektoré vlastnosti tak, ako ich chceme pri spustení:

```
private function onMapPreinitialize(event:MapEvent):void {
    var myMapOptions:MapOptions = new MapOptions();

    myMapOptions.scrollWheelZoom = true;
    myMapOptions.continuousZoom = true;
    myMapOptions.zoom = 15;
    myMapOptions.center = new LatLng(49.190399,16.610813)
    myMapOptions.mapType = MapType.SATELLITE_MAP_TYPE;

    this.map.setInitOptions(myMapOptions);
}
```

Funkcia definuje nový objekt `myMapOptions` typu `MapOptions`<sup>4</sup>. Následne sú priradené hodnoty jednotlivým parametrom:

1. povolenie priblíženia / oddialenia mapy kolečkom myši
2. povolenie plynulého približovania / odd'ľovania
3. nastavenie úrovne priblíženia
4. nastavenie súradníc stredu zobrazenia funkciou `LatLng`
5. nastavenie typu mapy

Posledný príkaz tieto vlastnosti priradí objektu `map`.

Na vznik mapy bezprostredne reaguje funkcia `onMapReady(event)`. Vkladá kontrolné prvky a nastavuje reakcie na udalosti (klik, pohyb kurzoru myši, zmena úrovne priblíženia, atď.).

---

<sup>4</sup>Takmer každý prvok, vlastnosť či funkcia, ktorá nie je obsiahnutá v základnom balíčku Flex SDK potrebuje pre svoje účely vlastnú knižnicu (v tomto prípade knižnicu *com.google.maps.MapOptions*). Práca by sa však menovaním každej, v nej použitej, stala neprehľadnou. Používaním prostredia Flash Builder je vývojár na chýbajúcu knižnicu upozornený a v prípade používania nápovery, sú kódy pre import knižníc dopĺňané automaticky.

### 7.4.2 Nové jednotky

Vytváranie nových jednotiek je nutné realizovať ako graficky, tak dátovo, to znamená založiť nový záznam do skupiny bezdrôtových prvkov siete.

Aby sme nový uzol vytvorili, musíme zadať informácie popisujúce zariadenie. Informácie sú zadávané do formulára, kvôli ktorému bola vytvorená samostatná komponenta `newNodeForm`. Tento prvok je uložený v súbore `newNodeOnMapForm.mxml` a je používaný metódou `PopUpManagera`. Je to metóda, ktorá formulár zobrazí v popredí aplikácie:

```
PopUpManager.createPopUp(this, newNodeOnMapForm, true);
```

Formulár (ukážka na obrázku 7.3 v kapitola *Užívateľské rozhranie* 7.5) obsahuje editačné polia pre vloženie názvu, ID, sériového čísla, MAC adresy, súradníc a iných detailov. Editačné polia sú ošetrované a nie je napríklad možné zadať znaky tam, kde sú očakávané numerické hodnoty. Formulár čaká na stlačenie tlačítka *Confirm*, ktorým sa aplikácia vnorí do algoritmu spracovania udalosti `popFormHandler()`.

Za podmienky vyplnenia všetkých požadovaných položiek je deklarovaný nový objekt a hodnoty sa do neho uložia vo forme parametrov. Objekt je ďalej použitý ako vstupný parameter funkcie starajúcej sa o vloženie novej jednotky do pola objektov a tiež je použitý ako vstupný parameter pre vykreslenie uzlu. Nakoniec je okno formulára zmazané:

```
var obj:Object = new Object();
    obj.name = pop.nodeNameF.text;

addNodeToArray(obj)
createMarker(obj);
PopUpManager.removePopUp(pop);
```

### 7.4.3 Vizualizácia jednotiek

O vykreslenie bezdrôtového prvku sa stará funkcia `createMarker(obj:Object)`. Vstupným parametrom je objekt, ktorý nesie informácie o danej stanici. Algoritmus rozlíši o aký typ zariadenia sa jedná - prijímač alebo vysielateľ, ďalej vytvorí obrazový objekt určujúci umiestnenie a vlastnosti prvku a patrične ho zobrazí na mape metódou `addOverlay()`:

```
var myMarker:Marker = new Marker(
    new LatLng(obj.latitude, obj.longitude),
    new MarkerOptions({}) );
map.addOverlay(myMarker);
```

Zároveň je volaná funkcia `createPolygon()`, ktorá vracia objekt typu `Polygon`. Je to grafický útvar, kruh symbolizujúci dosah signálu bezdrôtového zariadenia. Funkcii sú predávané súradnice uzla a číselná reálna hodnota predstavujúca vzdialenosť v metroch, čiže polomer kruhu.

`Polygon` je vykreslený, stred mapy umiestnený v bode nového uzlu:

```
map.addOverlay(myPolygon);  
map.setCenter(new LatLng(nodeObj.latitude, nodeObj.longitude));
```

Možný výsledok tohoto úsilia je patrný z ukážky v kapitole *Užívateľské rozhranie* 7.5 na obrázku 7.2.

Pri práci s virtuálnym uzlom je nutné obslúžiť udalosti s ním spojené – zmena polohy pretiahnutím uzla (`Drag&Drop`), klik, zameranie kurzora myši na uzol a iné. Tieto udalosti sú sledované funkciami *Event Listeners*.

#### 7.4.4 Import

Aplikácia spracováva textové reťazce XML štruktúry. Bolo by najvhodnejšie, použiť súbor so zoznamom staníc uložený na lokálnom počítači. Bohužiaľ, nie je možné zistiť cestu k tomuto súboru. Je možné ho vyhľadať, použiť zdroj ako objekt s obmedzenými vlastnosťami, ale nie je možné získať cestu k nemu a ďalej s ním pracovať podľa našich potrieb. Je to totiž považované za možné riziko narušenia zabezpečenia a povolené je to iba v prípade desktopových AIR aplikácií [4]. K súboru je implicitne definovaná cesta s názvom súboru. Ak chceme teda konkrétny súbor použiť, musíme ho premenovať a umiestniť do adresára *xml\_import*.

Tento postup je nepraktický a tak povediac strnulý. Bola vyvinutá alternatíva, ktorá tento problém obchádza. Bol vytvorený nový komponent `newXMLForm`, ktorý disponuje prvkom *TextArea*. Je do neho možné vložiť akokoľvek dlhý textový reťazec. Užívateľ teda môže textovú XML štruktúru jednoducho prekopírovať. Formulár je používaný rovnakým spôsobom ako formulár pre vloženie nového uzla – prostredníctvom *PopUpManagera*.

Potvrdením program zaháji preklad procedúry `popXmlFormHandler()`. Po overení vyplnenia textovej oblasti nasledujú kroky prevodu XML objektu na pole:

```
var xmlLi:XMLList = new XMLList();  
    xmlLi = XMLList(popXml.xmlTextAreaF.text);  
var xmlDoc:XMLDocument = new XMLDocument(xmlLi);  
var sDec:SimpleXMLDecoder = new SimpleXMLDecoder();  
var xmlObj:Object = sDec.decodeXML(xmlDoc);  
var arrayX:Array = ArrayUtil.toArray(xmlObj);
```



```

for each (var objNet:Object in arrayX){
    for (var i:int=0; i<objNet.network.node.length; i++){
        addNodeToArray(objNet.network.node[i]);
        createMarker (objNet.network.node[i]);
    }
}

```

Ako je z ukážky vidieť, na to aby sme z premennej dostali prvok iného typu, sú nutné určité medzikroky – zmena datového typu na `XMLDocument`, ktorý je za pomoci objektu `XML` dekodera konvertovaný na premennú typu `Object`. Objekt je ďalej vložený do pola.

Tým sa však vytvorí pole o obsahu jediného prvku – celého objektu. Preto je nutné prejsť celým polom a jednotlivé položky doslova vyčleniť, takto uložiť do pola a zobrazíť na mape.

### 7.4.5 Export

Aplikácia ukladá záznamy o bezdrôtových prvkoch siete do spoločného pola. Záznamy, tj. vlastnosti a parametre všetkých jednotiek sú užívateľovi k dispozícii v prehľadnej tabuľke. Export týchto dát je možný hneď v dvoch formátoch: *XML* a *CSV*.

Formáty majú spoločné, že predstavujú pevnú známu štruktúru. Zatiaľ čo *XML* organizuje dáta do značiek, dáta v *CSV* sú delené do obyčajných riadkov. *CSV* je formát, ktorý sa používa v tabuľkových programoch typu Microsoft Excel. Čo sa týka štruktúry, riadky tabuľky sú riadkami textového reťazca *CSV* súboru. Dáta v jednotlivých stĺpcoch sú od seba rozlíšené medzerou.

Možnosť uloženia modelu siete do *CSV* ocenia užívatelia v situácií, kedy potrebujú z výstupných dát spracovať napríklad grafy, či použiť hodnoty pri ďalších, napríklad štatistických výpočtoch.

O export sa starajú funkcie `generateXMLfile()`, `generateCSVfile()`. Obe získavajú dáta skrz premennú `nodes`, ktorou je definované pole uzlov. Dáta upravia do požadovanej formy `exportStruc` typu *String* a pomocou triedy *FileReference* ich vo vybranom formáte uložia na konkrétne miesto pevného disku počítača (tip cieľového umiestnenie v tvare textového reťazca je druhým vstupným parametrom funkcie `save`):

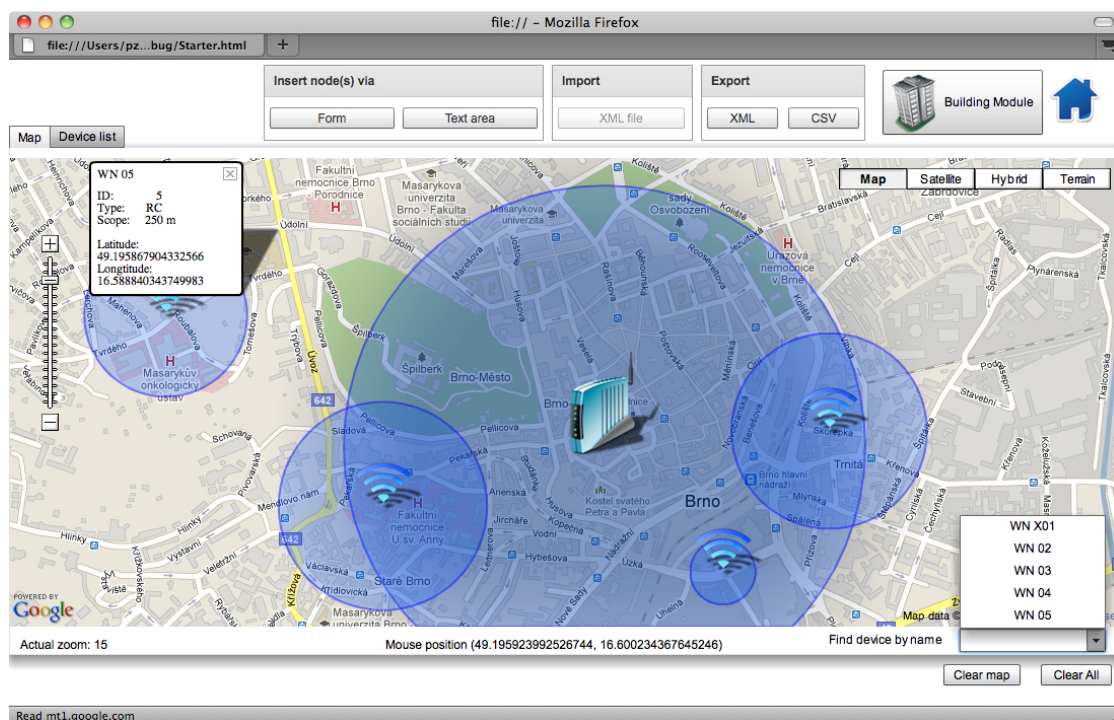
```

var file:FileReference = new FileReference();
file.save(exportStruc, "/address");

```

## 7.5 Uživatelské rozhranie

Ovládacie prvky rozhrania (obr.7.2) sú rozmiestnené nad a pod mapovým podkladom do panelov podľa účelu. Pri prechode kurzorom myši ponad nimi je daný panel jemne zvýraznený.



Obr. 7.2: Možnosti užívateľského rozhrania

### 7.5.1 Tvorba nových jednotiek

Panel *Insert node(s) via* disponuje tlačítkami pre vytvorenie nového uzla alebo vloženie celej siete uzlov prostredníctvom textu jazyka XML. Kliknutím na tlačítko *Form* sa zobrazí formulár pre vyplnenie potrebných údajov. Aplikácia kontroluje správnosť vložených údajov, ako aj vyplnenie všetkých položiek. Nový prvok siete sa dá vytvoriť ešte jedným spôsobom – kliknutím na plochu mapy. Získame totožný formulár avšak s predvyplnenými hodnotami polohy bodu, na ktorý bolo kliknuté. Ukážka vyplneného formulára je na obrázku 7.3.

Potvrdenie vykonáme stiskom tlačítka *Confirm*. Tým je vytvorený nový záznam a uzol je vykreslený do mapy v bode súradníc *Latitude* a *Longitude*. Okolo jednotky sa zobrazí kruh symbolizujúci dosah vysielaného, resp. prijímaného signálu, ktorého polomer má veľkosť *Scope*, tj. vzdialenosť 150 m.

Takto možno vytvoriť ľubovoľný počet uzlov a budovať tak model komplexnej bezdrôtovej siete manuálne.

Obr. 7.3: Formulár pre vloženie novej bezdrôtovej jednotky

Tlačítkom *Text area*, dosiahneme podobný výsledok. Formulár (viz. obr. 7.4) však obsahuje jedinú položku – textové pole pre vloženie aj celej siete. Program textový XML reťazec spracuje a uzly patrične vykreslí.

Obr. 7.4: Formulár pre vloženie XML štruktúry

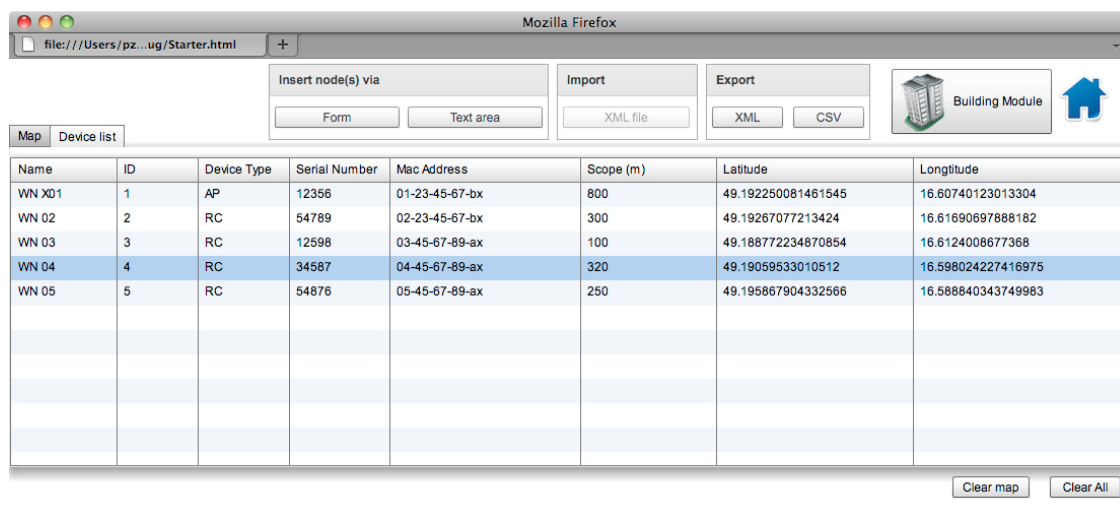
## 7.5.2 Import

Tlačítkom panela *Import* spracujeme externý súbor uložený na lokálnom disku a na základe informácií v ňom obsiahnutých sa zobrazí sieť. Keďže nie je možné získať cestu k nemu<sup>5</sup>, aplikácia pracuje so súborom, ku ktorému je cesta pevne daná.

<sup>5</sup>Dôvod, prečo je tomu tak bol vysvetlený v sekcii Funkcie v časti 7.4.4

### 7.5.3 Zoznam jednotiek

Prvky, ktoré naplnili pole bezdrôtových uzlov, je možné zobraziť v prehľadnej tabuľke – obr.7.5. Nachádza sa pod záložkou *Device List*. V tabuľke je výpis kompletných vlastností všetkých importovaných alebo vytvorených jednotiek. Prvky je možné abecedne zoradovať podľa jednotlivých vlastností – kliknutím na hlavičku stĺpca. Poradie stĺpcov je možné meniť pretiahnutím na uvažované miesto.



Name	ID	Device Type	Serial Number	Mac Address	Scope (m)	Latitude	Longitude
WN X01	1	AP	12356	01-23-45-67-bx	800	49.192250081461545	16.60740123013304
WN 02	2	RC	54789	02-23-45-67-bx	300	49.19267077213424	16.61690697888182
WN 03	3	RC	12598	03-45-67-89-ax	100	49.188772234870854	16.6124008677368
WN 04	4	RC	34587	04-45-67-89-ax	320	49.19059533010512	16.598024227416975
WN 05	5	RC	54876	05-45-67-89-ax	250	49.195867904332566	16.588840343749983

Obr. 7.5: Zoznam vytvorených/importovaných prvkov

### 7.5.4 Ukladanie vytvorených dát

Panel *Export* dáva na výber vytvorenie nového súboru v dvoch formátoch – XML a v tabuľkovom formáte CSV. Použijú sa všetky dáta z pola bezdrôtovej siete a vytvorí sa súbor patričného typu s obsahom požadovanej štruktúry.

Obsah exportovaných súborov modelu z obrázkov, 7.2 a 7.5 je uvedený v prílohách pre obidva formáty

### 7.5.5 Mazanie

Tlačítkom *Clear map* vymažeme stanice a to iba vizuálne, kvôli prehľadnosti. Môžeme doplniť ďalšie jednotky a opäť všetky zobrazíť.

Tlačítkom *Clear All* vymažeme ako mapovú plochu, tak aj záznamy v poli. Prídeme tak prakticky o všetky neuložené dáta.

### 7.5.6 Ďalšie vlastnosti

Modifikácia rozloženia modelu bezdrôtovej siete je realizovaná intuitívne - pretiahnutím grafického objektu symbolizujúceho bezdrôtové elektronické zariadenie na cieľové miesto. Následkom je okamžitá zmena vizuálneho umiestnenia a taktiež aktualizácia súradnicových dát daného objektu v zozname siete.

Kliknutím na ikonu vytvoreného, či importovaného uzla je zobrazené okno s detailnejšími informáciami. Na obrázku 7.2 v úvode podkapitoly je bližšie vidieť, aké možnosti pracovná plocha ponúka.

Typ mapového podkladu a úroveň priblíženia možno kedykoľvek meniť, mapa je posúvateľná. O úrovni aktuálneho priblíženia informuje poznámka *Actual zoom* v ľavom dolnom rohu.

Poznámka *Mouse position* v strede rozhrania je viditeľná len v prípade, že sa kurzor pohybuje po ploche mapy a čísla v zátvorkách sa dynamicky menia podľa toho, na aký bod ukazuje. Po kliknutí alebo ťahaní niektorého z uzlov na iné miesto sa zobrazí text *Marker position* a v zátvorkách budú súradnice uzla, namiesto súradníc kurzoru.

Roletové menu v pravom dolnom rohu je aktívne len v prípade, že pole uzlov obsahuje aspoň jeden záznam, vid'. obr. 7.2. Slúži na vyhľadanie uzla na základe mena. Po výbere z menu sa mapa posunie tak, aby hľadaný uzol ležal v jej presnom strede.

Vlastnosti ovládacích tlačítok sa menia v závislosti na predchádzajúcich udalostiach. Ak pole siete neobsahuje žiadny prvok, tj. model siete neexistuje, tlačítka pre export a mazanie a vyhľadávacie menu sú neaktívne. Sprístupnia sa v prípade, že model obsahuje aspoň jedno zariadenie. Ďalej ak tlačítkom *Clear map* z povrchu mapy vymažeme všetky uzly, toto tlačítko sa zmení na *Show nodes* pripravené uzly kedykoľvek znovu zobrazíť.

Takýmto spôsobom je aplikácia prehľadnejšia (počet ovládacích prvkov klesá na minimum) a tiež je užívateľ intuitívne vedený k jej správne mu užívaniu.

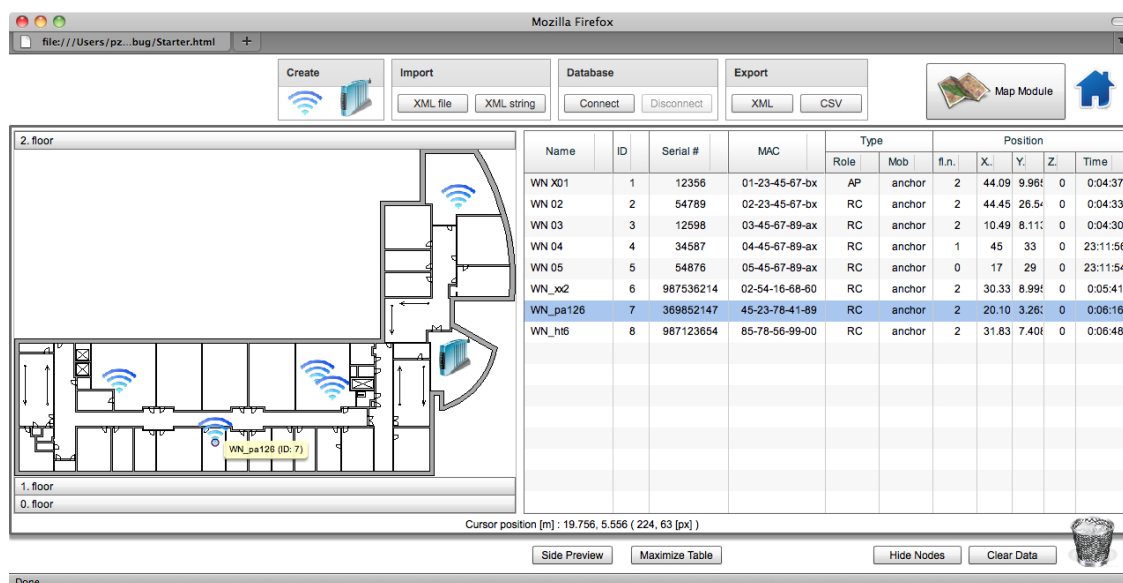
## 8 MODUL BUDOVY

V predchádzajúcom popisovanom module bol vývoj založený na podpore portálu Google Maps. Pri modelovaní siete, ktorá sa nachádza v uzatvorenom priestore nie je možné počítať s využitím podobnej technológie. Každý obytný dom, sklad, budova je unikátna a neexistuje databáza plánov všetkých objektov na svete.

K práci s rozložením jednotiek siete je nutné mať podklad, vzhľadom ku ktorému je posudzovaná ich poloha. Podkladmi, sú v prípade budov pôdorysy, plány - priemy objekty na vodorovnú rovinu pri pohľade zhora. Každá budova je špecifická, málokedy sa pôdorysy viacerých budov zhodujú, preto je nevyhnutné mať ich pripravené, aby s nimi mohla aplikácia pracovať.

Ďalšou otázkou je, ako riešiť trojrozmerný priestor v rámci viacposchodovej budovy. Užívateľ musí mať predstavu a prehľad nie len o horizontálnom rozmiestnení uzlov siete ale aj o poschodí, v ktorom sa nachádzajú.

Grafické rozhranie modulu je rozdelené na hlavné dve časti. Jedna časť predstavuje budovu, resp. pôdorysy jej všetkých poschodí, druhou časťou je tabuľka, ktorá prehľadne zhromažďuje informácie o jednotkách v budove 8.1.



Name	ID	Serial #	MAC	Type	Role	Mob	fl.n.	X	Y	Z	Time
WN_X01	1	12356	01-23-45-67-bx	AP	anchor	2	44.09	9.96	0	0:04:37	
WN_02	2	54789	02-23-45-67-bx	RC	anchor	2	44.45	26.5	0	0:04:33	
WN_03	3	12598	03-45-67-89-ax	RC	anchor	2	10.49	8.11	0	0:04:30	
WN_04	4	34587	04-45-67-89-ax	RC	anchor	1	45	33	0	23:11:56	
WN_05	5	54876	05-45-67-89-ax	RC	anchor	0	17	29	0	23:11:54	
WN_x02	6	987536214	02-54-16-68-60	RC	anchor	2	30.33	8.99	0	0:05:41	
WN_pa126	7	369852147	45-23-78-41-89	RC	anchor	2	20.10	3.26	0	0:06:16	
WN_h06	8	987123654	65-78-56-99-00	RC	anchor	2	31.83	7.40	0	0:06:48	

Obr. 8.1: Základné užívateľské rozhranie modulu

### 8.1 Komponent Accordion

Dôležitým komponentom je *mx* komponent *Accordion*. Sú do neho hierarchicky vsadené pôdorysy poschodí smerom od prízemí. Pri testovaní aplikácie boli využité

plány Fakulty telekomunikačnej a informačnej technológie, Ústavu telekomunikácií. Obrázky sú uložené do dátového typu *Image* vo formáte PNG.

Budova má počet poschodí zhodný s počtom objektov v danom poli a podľa toho má komponent *Accordion* záložiek. Každá záložka predstavuje jedno poschodie. Uzol teda musí niesť informáciu o čísle podlažia, v ktorom sa práve vyskytuje.

Modul bol vyvíjaný s vedomím, že nemôže byť určený len pre jediný konkrétny objekt. Nie je vopred jasné v akej konkrétnej budove bude využívaný a koľko podlaží bude vlastne mať. Bolo by značne nepraktické dopĺňať každé poschodie zvlášť a tomu prispôbovať celý zdrojový kód. Riešením je vkladanie pôdorysov do pola objektov, odkiaľ sú cyklicky spracovávané. Je nutné definovať iba cesty k obrazovým podkladom poschodí a inicializovať komponent *Accordion*.

Inicializačná funkcia `accordionInit(event)` je volaná hneď po vytvorení modulu:

```
<mx:Module creationComplete="accordionInit(event)">
```

Funkcia uloží pôdorysy formátu png do pola obrázkov *Image* `allFloorsArray`, odkiaľ sú cyklom `for` spracovávané. Vytvorí sa nový obrazový typ a sú mu priradené vlastnosti, ktoré bude užívateľ ďalej pri práci potrebovať, hlavne číslo poschodia `flNum` a spôsob reakcie pôdorysu na podnety (*EventListeners*). Takto pripravený obrázkový objekt je priradený novej záložke *NavigatorContent* `newNavCont` komponenty *Accordion* `acc`. Skrátená ukážka inicializačnej funkcie nasleduje:

```
protected function accordionInit(event:FlexEvent):void{
    var newFloor:Image = new Image;

    newFloor.source = "../images/fekt_plan_resized.png";
    allFloorsArray.addItem(newFloor);

    for each(var img:Image in allFloorsArray){
        flNum++;
        img.id='fl'+flNum.toString();
        img.addEventListener(DragEvent.DRAG_DROP, dragDropHandler);

        var newNavCont:NavigatorContent = new NavigatorContent;
        newNavCont.addElement(img);
        acc.addElement(newNavCont);
    }
}
```

## 8.2 Funkcie modulu budovy

Modul pre prácu so sieťou vo vnútri budovy má pracovať obdobne ako modul pracujúci s mapovými podkladmi. Niektoré metódy boli použité v oboch moduloch a upravené podľa potrieb a cieľov zadania. Popisovať dopodrobna rovnaké alebo podobné postupy by bolo bezúčelné. Budú teda spomenuté len v skratke.

- Vkladanie nových prvkov je riešené prostredníctvom upraveného a doplneného *pop-up* formulára alebo okna pre vloženie textového reťazca formátu XML
- Pre import dát z externého súboru bola využitá služba *HttpService*, ktorá vstupnú XML štruktúru prevedie na interné pole objektov *ArrayCollection*
- funkcie pre export sú prakticky identické

### 8.2.1 Tvorba a vizualizácia jednotiek

Po získaní potrebných údajov importovaných bezdrôtových jednotiek sa začne sled príkazov a zreťazených funkcií. Každý vložený prvok nesie informácie o svojich vlastnostiach, napríklad názov, identifikačné číslo, číslo poschodia, sériové číslo, atď..

Funkciou `assignValuesToNewlyCreatedNode(Object)` je mu priradený komponent grafického rozhrania (ikona uzlu) so všetkým, čo k správne mu zobrazeniu patrí: symbol bezdrôtového uzlu, jeho veľkosť, vypočítané súradnice polohy  $x$  a  $y$ , informačné okno zobrazujúce sa po zameraní kurzoru myši.

Ďalej je volaná funkcia `createNewNode(Object)`, ktorej vstupným parametrom je objekt práve vytváraného bezdrôtového uzla. Táto metóda je veľmi dôležitá, pretože definuje správanie uzlu, tj. ako má reagovať na činnosť užívateľa:

- pri zacielení kurzora na daný prvok je vykreslený presný bod jeho polohy a je zvýraznený v súhrnnej tabuľke
- po dokončení ťahania ikony metódou *drag and drop* je rozhodnuté, či je ťahaný objekt určený k mazaniu alebo k zmene umiestnenia. V prvom prípade sa volá funkcia `destroyNode(Object)`, ktorá odstráni ikonu z užívateľského rozhrania a vymaže prvok z poľa všetkých uzlov. V prípade druhom sú získané nové súradnice, čas zmeny polohy (funkciou `getTimeStmp()`, ktorá získa aktuálny systémový čas upravený do požadovaného formátu). Zmeny sú uložené do vlastností daného uzlu a následne sa aktualizuje zoznam prvkov v tabuľke.

Nakoniec je grafický symbol uzla priradený ako potomok objektu plánu budovy `dropTrgt`, tj. vykreslený do patričného poschodia, podľa čísla poschodia:

```
dropTrgt.addChild(Object.icon);
```

Posledný krok vytvorenia nového bezdrôtového uzla predstavuje skupina príkazov `addNewNodeToArrayCol(Object)`. Táto vytvorený objekt, priradí do dátovej jednotky typu *ArrayCollection*: `nodes.addItem(Object);`



### 8.2.2 Určovanie polohy jednotiek

Určovanie polohy zariadení vo vnútri budovy svetovými súradnicami ako v prípade modulu mapy by bolo nezmyselné. Pokiaľ aplikácia disponuje plánom budovy, je najjednoduchšie z hľadiska posudzovania vzdialeností použiť hodnoty v pixloch. Tie ale nedávajú užívateľovi veľmi jasnú predstavu o reálnej polohe vzhľadom k okolitým predmetom. Ideálnym prípadom je, ak je známa mierka sprostredkovaného obrázku. Potom je možné získať z hodnôt pixlov hodnoty v akýchkoľvek dĺžkových jednotkách metrickej sústavy a naopak.

Aplikácia k presnému určeniu umiestnenia využíva jedine jednotky pixlov, preto je nevyhnutné vstupné a následne výstupné hodnoty prepočítavať. Slúžia k tomu funkcie `pxToM(Number)` a `mToPx(Number)`. Vstupnými parametrami sú číselné hodnoty v jednotkách, ktoré je potrebné previesť na opačné (metre na pixle alebo pixle na metre):

```
private function pxToM(valPx:Number):Number{
    var cm:Number;
    var resultRatio:Number;

    cm = (valPx/72)*2.54;
    resultRatio = cm * 2.5;

    return resultRatio;
}
```

Za predpokladu, že jeden palec sa rovná 72 pixlov, pričom palec je 2,54 cm[12], sú vstupné hodnoty pixlov prepočítané na hodnotu v centimetroch. Následne je podľa platnej mierky získaná reálna vzdialenosť. V tomto konkrétnom príklade je mierka 1:250<sup>1</sup>. Výstupnou hodnotou je výsledok.

Ďalšou komplikáciou je začiatok súradnicovej sústavy obrázkov. Zatiaľ čo v bežnej praxi je za začiatok dvojrozmernej sústavy grafov, či obrazov braný ľavý dolný roh, v praxi programátorskej je to často inak - je ním ľavý ale horný roh. Problém nastane v momente, keď užívateľ potrebuje mať predstavu o umiestnení objektu vzhľadom k určitému vzťažnému bodu (vzdialenosť od klasického súradnicového začiatku, vzdialenosť od stien budovy, atď.). Riešenie je jednoduché. Pri práci so súradnicou *y* stačí použiť jednoduchú operáciu odčítania danej súradnice od celkovej výšky obrazového objektu.

Horšia situácia nastáva, ak je tvar objektu iný, zložitejší ako tvar štvorca alebo obdĺžnika. To je prípad fakulty telekomunikácií, ktorá je členitá - má viac hrán,

---

<sup>1</sup>Mierka 1:250 znamená, že 1 centimeter na mape predstavuje 250 centimetrov v skutočnosti, tj. 2,5 metra

takže viac potenciálnych možných začiatkov súradnicovej sústavy. Je to patrné z obrázku užívateľského rozhrania, kde sa s ňou pracuje - 8.1. Túto prekážku môže rieši funkcia `getDistanceFromWall{Number, Number}`, ktorá na základe súradníc uzlu zistí počet metrov od najbližších stien vo smere x aj y s presnosťou na tri desatinné miesta. Je to bohužiaľ len čiastkové riešenie, ktoré je použiteľné iba v prípade tohoto špecifického tvaru a veľkosti pôdorysu. Pre akúkoľvek inú budovu je nutné funkciu upraviť, preto nie je v konečnej verzii použitá a je uprednostnená verzia, ktorá prepočítava súradnicový systém tak, aby bod `[0, 0]` ležal v ľavom spodnom rohu.

### 8.2.3 Mazanie jednotiek

Prvky modelu siete je možné skryť, vymazať naraz alebo po jednom. V prvom prípade funkcia `wipeOffNodesFromThePlan()` odoberie ikonu uzla objektu, ktorý predstavuje poschodie, v ktorom sa uzol nachádza a to pre každý prvok v poli bezdrôtových jednotiek:

```
for each (var obj:Object in nodes)
    dropTrgt.removeChild(obj.icon)
```

V druhom prípade funkcia `clearNodesInArrayCol()` navyše vymaže celé pole bezdrôtových jednotiek.

```
nodes = null;
```

Pre mazanie jednotlivo, pretiahnutím symbolu bezdrôtového zariadenia na ikonu smetného koša, bolo nutné nastaviť naslúchače udalostí tak, aby objekt koša prijal toto pretiahnutie - metóda `rcBin_dragEnterHandler(DragEvent)`, a aby ťahaný uzol na túto udalosť reagoval - metóda `dragCompleteHandler()`. Ak prvok zistí, že sa nachádza nad ikonou odpadkového koša, je volaná funkcia `destroyNode(Object)`, ktorá tento prvok odstráni z plánu budovy, ako aj z poľa sieťových prvkov. Ak prvok nie je určený k mazaniu, znamená to, že bol ťahaný so zámerom zmeny polohy a je konaná tomu zodpovedajúca postupnosť krokov.

### 8.2.4 Zmena pohľadu

Funkcia zmeny pohľadu je nad rámec pôvodného zadania. Túto možnosť ocení užívateľ v situácií, kedy potrebuje získať prehľad o výškových rozdieloch umiestnenia jednotiek na poschodí. Druhým pohľadom, je teda pohľad z boku (spredu) budovy<sup>2</sup>.

---

<sup>2</sup>Spôsob písania programového kódu tejto časti je podobný tomu, ktorý zobrazuje objekt budovy zvrchu a nie je nutné ho rozoberať do prílišných podrobností.

Vytvorením alebo importovaním novej jednotky vznikne zároveň kópia jej ikony. Kópia je implementovaná do ďalšieho komponentu *Accordion*, ktorý je zobrazený na požiadanie.

Význam nového *Accordion* je obdobný ako toho už popisovaného, ale jeho možnosti sú užšie, pretože slúži len pre jednoduché zobrazenie rozloženia siete daného poschodia z inej perspektívy. Jeho vlastnosti (výška, šírka, počet záložiek) sú odvodené od toho pôvodného aby sa zachoval jednotný vzhľad. Tieto nastavenia má na starosti inicializačná funkcia `acc_sideViewInit()`.

Vždy pri vzniku novej jednotky siete sa vytvára kópia jeho symbolu funkciou `cloneNodeForSidePreview(Object)`. Tá ho následne ukladá do poľa všetkých kópií `nodesClonedForSpreview`. Nový prvok poľa nesie na mieste súradnice Y hodnotu súradnice Z (*positionZ*), tj. výšky, ktorá je jedným z parametrov objektu bezdrôtovej jednotky. Podľa hodnôt týchto súradníc je ikona vykreslená do prázdneho obrázka patričného poschodia príkazom `addChild(Image)`.

Zobrazenie musí byť aktuálne, musí reagovať na zmeny polohy, import nových prvkov a mazanie. Funkcia `updateSidePreview(Object)` sa stará o udalosti spojené s premiestňovaním. Je vykonaná pri každej zmene. Pokiaľ je prvok siete mazaný, má sa aj kópia jeho symbolu v bočnom zobrazení. Ak je presúvaný na iné miesto poschodia, aktualizuje sa jeho poloha v oboch pohľadoch.

## 8.3 Spojenie s databázovým systémom

Okrem už popísanej možnosti importu externého XML súboru je modul schopný získavať externé dáta z databázového systému. Nakoľko je Adobe Flex jazykom pre programovanie klientskej časti aplikácií, musí byť pre spojenie s databázou použitá serverová technológia. Technológia schopná priamej komunikácie so serverom je prostredníkom slúžiacim k preposielaniu dát medzi serverom, na ktorom beží databázový systém a aplikáciou, ktorá tieto dáta spracováva.

Je požadované aby aplikácia slúžila pre účely fakulty, čomu je prispôbovaná aj v tomto smere. Na školskom servery<sup>3</sup>, na ktorom sú zoskupované dáta bezdrôtových jednotiek v budove, je databáza PostgreSQL. Je to *open source* objektovo-relačný databázový systém. Prostredníctvom serverovej aplikácie modul využíva a pravidelne spracováva údaje. Aktuálnosť modelu siete zostáva aj po zmene záznamov v databázových tabuľkách zaručená.

Technológiou hrajúcu rolu prostredníka je JAVA. Jednoduchá JAVA aplikácia pracuje na strane serveru Apache Tomcat (147.229.144.201:8080) a komunikuje s databázou SQL príkazmi. Pri žiadosti o poskytnutie záznamov je odoslaný vhodný

---

<sup>3</sup>Adresa servera: 147.229.144.201

dotaz (napríklad `SELECT`), ktorý vyberie dáta. Tie sú spracované do XML štruktúry a odoslané cez HTTP.

### 8.3.1 Dotaz `SELECT`

Prostredníctvom služby HTTP je získané XML z url adresy, ktorá nesie SQL dotaz<sup>4</sup>:

```
<s:HTTPService id="HttpSrv_postgre_SELECT"
  url='http://147.229.144.201:8080/WNModelingToolServer/
    QueryServlet?query=SELECT/..'
  result="HttpSrv_postgre_SELECT_resultHandler(event)"/>
```

Databázový príkaz `SELECT` vracia záznamy z *VIEW*<sup>5</sup> tabuľky `ActiveNodes`, ktorá sprostredkováva:

- identifikačné čísla – atribút `NodeID`
- súradnice umiestnenia – atribúty `PoistionX`, `PoistionY` a `PoistionZ`
- časovú značku – `TimeStamp`
- typ zariadenia – `Type`

Aplikácia dokáže pracovať s väčšou množinou informácií, ktoré zatiaľ v databáze nefigurujú. Tie sú v programe dopĺňané implicitnými hodnotami, aby sa zachoval jednotný dátový model.

Žiadosť o prevedenie služby je na podnet užívateľa odoslaná príkazom

```
HttpSrv_postgre_SELECT.send();
```

#### Časovač

Nakoľko nie je veľmi účelné prijímať dáta jednorázovo, ale pravidelne, musí byť zavedený mechanizmu, ktorý bude činnosť opakovať. Globálna premenná `timer` je časovačom s pevne daným intervalom:

```
private var pollInterval:int = 10000;
private var timer:Timer = new Timer(pollInterval);
```

Spustenie a ukončenie opakovania v časovom intervale 10000 ms je realizované príkazmi `start()` a `stop()`. Časovaču je v požadovaný okamžik priradený *EventListener*, ktorý volá funkciu `refreshDBdata()` vždy po vypršaní nastaveného intervalu

```
timer.addEventListener(TimerEvent.TIMER, function refreshDBdata);
```

<sup>4</sup>Nakoľko je SQL dotaz príliš dlhý, jeho kompletný formát je uvedený ako príloha v textovom dokumente na priloženom médiu (viď. B.1)

<sup>5</sup>Tabuľka *VIEW* (pohľad) slúži na spojenie požadovaných atribútov rôznych databázových tabuliek do jednej tabuľky.

### 8.3.2 Dotaz INSERT

Rovnakým princípom, bez použitia časovača, sú nové dáta nahrávané na server. Rozdielom je použitie príkazu INSERT<sup>6</sup>. Dáta sú vkladané vždy jednorázovo do tabuľky **Measurements**. Tabuľka obsahuje výsledky meraní a jej účelom je zaznamenávať rozdiely medzi aktuálnou polohou, ktorá bola fyzicky nameraná, a polohou, ktorú určuje samotné bezdrôtové zariadenie s automatickým polohovacím systémom.

Aplikácia teda zobrazí bezdrôtové jednotky z databázového systému a poskytne možnosť porovnania s hodnotami, ktoré zadá užívateľ. Po potvrdení sú tieto hodnoty odoslané spolu s vypočítaným rozdielom a absolútnou odchýlkou do databázovej tabuľky ako nový záznam.

Atribútmi tabuľky sú:

- **NodeID** – identifikačné číslo (integer)
- **PosX** – súradnica  $X$  určená zariadením (real)
- **PosY** – súradnica  $Y$  určená zariadením (real)
- **PosZ** – súradnica  $Z$  určená zariadením (real)
- **RealX** – súradnica  $X$  určená/nameraná užívateľom (real)
- **RealY** – súradnica  $Y$  určená/nameraná užívateľom (real)
- **RealZ** – súradnica  $Z$  určená/nameraná užívateľom (real)
- **FailX** – rozdiel hodnôt  $PosX$  a  $RealX$  (real)
- **FailY** – rozdiel hodnôt  $PosY$  a  $RealY$  (real)
- **FailZ** – rozdiel hodnôt  $PosZ$  a  $RealZ$  (real)
- **FailAbs** – hodnota výsledku výpočtu absolútnej odchýlky (real)
- **TimeStamp** – časový údaj (text)
- **Note** – poznámka (text)

## 8.4 Uživatelské rozhranie

Čo sa týka vizuálneho prejavu užívateľského rozhrania, bolo zámerom vytvoriť ho tak, aby sa v čo najväčšej možnej miere podobal mapovému modulu. Grafickej komponente symbolizujúcej budovu je venovaný centrálny priestor, ktorý zdieľa s tabuľkou. Toto riešenie sa odlišuje od rozmiestnenia komponentov v predchádzajúcom prípade. Užívateľ má takto prehľad súčasne o priestorovom rozmiestnení modelu siete a tiež o vlastnostiach, počte, čase poslednej zmeny a tak ďalej.

V predchádzajúcej ukážke na obrázku 8.1 vidieť, ako aplikácia vyzerá počas modelovania, či monitorovania. Grafický komponent so záložkami obsahuje pôdorysy

---

<sup>6</sup>Kompletný formát je uvedený ako príloha v textovom dokumente na priloženom médiu (viď. B.1)

budovy. Každá záložka symbolizuje jedno poschodie, celkovo tak komponent zosobňuje objekt celej stavby.

Kliknutím na záložku sa vysunie mapa daného poschodia, ktorá je pracovnou plochou nástroja. Reaguje na pohyb kurzora a užívateľ je textom v spodnej časti informovaný o jeho polohe. Hodnoty polohy sú prepočítavané na jednotky metrov tak, aby zodpovedali skutočným vzdialenostiam od posudzovaného bodu, v tomto prípade je ním začiatok súradnicovej sústavy - ľavý dolný roh plánu.

Ak kurzor myši ukazuje na ikonu bezdrôtovej jednotky, objaví sa presný bod reálnej polohy, popis, ktorý zariadenie identifikuje a tiež je zvýraznená patričná položka zoznamu vpravo.

Zoznam prvkov modelu taktiež reaguje na podnety užívateľa. Po kliknutí na uzol, ktorý sa nenachádza na práve zobrazenom poschodí, odošle impulz komponente *Accordion*, ktorá pohľad na toto poschodie plynule otvorí.

### 8.4.1 Tvorba nových jednotiek

Vytvorenie úplne novej jednotky sa uskutoční po pretiahnutí symbolu z panela *Create* na plán poschodia. Toto je jediné miesto, na ktoré je v tomto štádiu možné ikonu pretiahnuť, takže užívateľ nemá čo pokaziť.

Vyobrazený formulár, ktorého ukážka je na obrázku 8.2, má už niektoré položky predvyplnené, niektoré nastavené nemenne:

- *Floor num.*, teda číslo poschodia sa nastaví podľa toho, do ktorej záložky bola ikona ťahaná
- *Position X* a *Position Y* sú hodnoty súradníc bodu požadovaného umiestnenia
- *Time Stamp* zachytáva systémový čas, kvôli zázname o vytvorení
- *Type* je buď *Receiver* (prijímač) alebo *Access Point* (prístupový bod, smerovač, vysielateľ) podľa zvolenej ikony panela *Create*
- *Mobility* - bezdrôtové zariadenie, je mobilné - *mobile* alebo statické - *anchor*

Pre vysvetlenie položky *Mobility*: mobilné uzly sú schopné informovať o zmenách svojej polohy. Ak chceme aby model zosobňoval skutočnosť, tieto informácie musí aplikácia spracovávať tak, ako sú prijímané. V tom prípade nie je žiadané, aby sa s nimi ešte explicitne manipulovalo. Preto je aplikácia navrhnutá tak, aby bolo možné upravovať polohu iba pevných uzlov, ktoré sa manuálne umiestnia do statickej pozície a záznamy o ňom sa budú ďalej udržiavať prostredníctvom aplikácie.

V rámci prevencie chýb je užívateľovi dovolené vkladať len očakávané znaky, napríklad v prípade súradníc X, Y, Z, sériového, či identifikačného čísla, iba numerické.

Tlačítko *Confirm* spustí príkazy vedúce k overeniu úplnosti a správnosti zadaných dát. Pokiaľ je všetko v poriadku, nový prvok je zakreslený do plánu poscho-

Obr. 8.2: Formulár pre vytvorenie nového objektu bezdrôtovej siete

dia a v tabuľke vznikne nový riadok – objekt sa považuje za vytvorený.

## 8.4.2 Import

Importovať model bezdrôtovej siete sa dá hneď tromi spôsobmi. Vo všetkých zohráva nepostrádateľnú úlohu jazyk XML:

### XML súbor

Opäť je kameňom úrazu idea prevencie bezpečnostných rizík technológie Flex, ktorá bola popísaná v kapitole *Modul máp* v časti *Import* (podkapitola 8.2).

V zdrojovom kóde je teda pevne nastavená relatívna cesta k súboru, ktorý je po stlačení tlačítka *XML file* spracovaný. Výsledok operácie sa prejaví ako v pláne (vykreslením jednotiek), tak novými záznamami v tabuľke vedľa neho.

### XML text

Predchádzajúcu skutočnosť kompenzuje tlačítko *XML string*. Logickým predpokladom je, že jediná vec, pre ktorú je externý súbor potrebný, je jeho obsah.

Stlačením tlačítka získa užívateľ možnosť obsah XML súboru - XML štruktúru, jednoducho skopírovať do textového poľa. Výsledok úkonu je rovnaký, ako pri načítaní súboru.

## Databáza

Nové možnosti spôsobu využitia aplikácie prináša panel *Database*. Ovládacie prvky v ňom nastavujú a rušia pravidelné získavanie dát uložených na vzdialenom servery.

Ak je databáza napĺňaná aktuálnymi údajmi, program ich v minimálnych intervaloch zobrazuje<sup>7</sup>. Je tak možné sledovať pohyb každej mobilnej jednotky, ktorá s databázou komunikuje.

O intenzite čítania záznamov databázových tabuliek informuje znak |, ktorého počet sa s každým načítaním inkrementuje. Po prvom pripojení je vpisovaný do hlavičky panela.

### 8.4.3 Zoznam jednotiek

Strediskom komplexných informácií o členských prvkoch modelu sieťových zariadení je tabuľka.

Významným komfortom, ktorý ponúka je schopnosť prispôsobenia, napríklad šírky stĺpcov, ich poradia, abecedné zorad'ovanie záznamov.

V užívateľskom rozhraní má svoje miesto, ktoré je obmedzené inými komponentami. V prípade požiadavkov na rozšírenie sledovaných parametrov bezdrôtových objektov nastáva problém so šírkou stĺpcov, ktorá nemusí postačovať. Rozhranie bolo navrhnuté tak, aby si ešte pri zobrazení na 13 palcovom monitore zachovalo svoju prehľadnosť. So zväčšujúcou sa šírkou okna aplikácie potom narastá šírka tabuľky a priestor pre údaje v nej.

V každom prípade nie je na škodu, mať možnosť maximalizácie. Tlačítko *Maximize Table* vyhradí tabuľke celú šírku okna (obr. 8.3) a zvyšuje tak úroveň pohodlia pri práci s modulom.

### 8.4.4 Pohľad z inej perspektívy

Okrem prehľadu o horizontálnom rozložení siete a poschodiach, v ktorých členské prvky operujú, môže byť prínosné vedieť o ich vzájomných výškových rozdieloch na konkrétnom poschodí, tj. v akej výške od podlahy sú objekty inštalované. Aj takúto schopnosť užívateľ vďaka tomuto modulu má.

Tlačítko *Side Preview* otvorí novú perspektívu - perspektívu pohľadu na budovu spredu. Ukážka na obrázku 8.4.

---

<sup>7</sup>Zobrazeniu dát predchádza ich spracovanie na strane JAVA aplikácie do XML štruktúry, prijatie tejto štruktúry klientskou Flex aplikáciou a opätovné prispôsobenie dát dátovým štruktúram využitým pri konečnej implementácii (podrobenjšie bolo popísané v časti *Spojenie s databázovým systémom* 8.3).



Mozilla Firefox

file:///Users/pz...bug/Starter.html

Create Import Database Export

XML file XML string Connect Disconnect XML CSV

Map Module

Name	ID	Serial #	MAC	Type		f.n.	Position			Time
				Role	Mob		X[m]	Y[m]	Z[m]	
UfromDB_1	1	123456789	12-34-56-78-90	RC	anchor	2	10	10	1	11:53:29
UfromDB_2	2	123456789	12-34-56-78-90	RC	anchor	2	20	5	1	11:53:38
UfromDB_3	3	123456789	12-34-56-78-90	RC	anchor	2	5	30	1	12:37:04
UfromDB_4	100	123456789	12-34-56-78-90	RC	mobile	2	12.5	17.5	1	11:38:32
UfromDB_5	101	123456789	12-34-56-78-90	RC	mobile	2	7.5	15	1	11:59:57
UfromDB_6	102	123456789	12-34-56-78-90	RC	mobile	2	15	23.75	1	12:02:27
UfromDB_7	103	123456789	12-34-56-78-90	RC	mobile	2	32.5	15	1	12:04:21
UfromDB_8	201	123456789	12-34-56-78-90	RC	mobile	2	14.469727319291476	15.67568977795024	1	0:04:13
UfromDB_9	202	123456789	12-34-56-78-90	RC	mobile	2	12.314056887959348	17.831360209282366	1	0:49:05

Side Preview Restore Table Hide Nodes Clear Data

Transferring data from 147.229.144.201...

Obr. 8.3: Maximalizovaná tabuľka s testovacími databázovými záznamami

Mozilla Firefox

file:///Users/pz...bug/Starter.html

Create Import Database Export

XML file XML string Connect Disconnect XML CSV

Map Module

2. floor

1. floor

0. floor

Name	ID	Serial #	MAC
WN_X01	1	12356	01-23-45-67-bx
WN_02	2	54789	02-23-45-67-bx
WN_03	3	12598	03-45-67-89-ax
WN_04	4	34587	04-45-67-89-ax
WN_05	5	54876	05-45-67-89-ax
WN_TSTx1	6	365412987	02-11-44-88-00
WN_TSTx2	7	986532147	11-15-73-77-44
WN_TSTx3	8	985214763	12-33-45-69-77

Hide Preview Maximize Table Hide Nodes Clear Data

Done

Obr. 8.4: Komponent pohľadu na poschodie spredu

Táto funkcionálnosť je akousi nadstavbou k pôvodnému zadaniu. Užívateľ vidí v akej vzájomnej pozícii sú časti siete z pohľadu, ktorý je pre človeka prirodzenejší ako ten zvrchu.

Náhľad je aktualizovaný pri každej zmene v hlavnom pláne (presun, vznik objektov, ich mazanie). Hodnota súradnice X, tj. horizontálnej polohy, zostáva. K vyjadreniu výškovej úrovne slúži parameter uzla, súradnica Z. Užívateľ jej hodnotu získava z tabuľky – stĺpec  $Z[m]$ .

### 8.4.5 Výpočet odchýlky polohy

Po kliknutí na ktorýkoľvek grafický symbol bezdrôtovej jednotky sa zobrazí nové okno. Ukážka je na obrázku 8.5. Toto okno slúži na zmenu ktorejkoľvek súradnice.

Compare coordinates			<input checked="" type="checkbox"/> DB record
Name: WN 02    ID: 2    Time: 16:07:29			
original X[m]: 45	original Y[m]: 25	original Z[m]: 0.8	
new X[m]: <input type="text" value="42"/>	new Y[m]: <input type="text" value="22"/>	new Z[m]: <input type="text" value="0"/>	
difference [m]: 3	difference [m]: 3	difference [m]: 0.8	
Abs Err: 4.317406628984581			
<input type="button" value="Cancel"/>		note: <input type="text" value="wn 02 pos is wrong"/> <input type="button" value="Send Data To DB"/> <input type="button" value="Apply Changes"/>	

Obr. 8.5: Komponent pohľadu na poschodie spredu

Pri vkladaní nových hodnôt, je automaticky prepočítavaný rozdiel oproti pôvodným súradniciam a taktiež je počítaná absolútna chyba.

Užívateľ má ďalej na výber, či tieto hodnoty zapísať do databázového systému alebo sa majú zmeny premietnuť len do modelu aplikácie. Možné je samozrejme aj oboje.

Odchýlka sa počíta hlavne z toho dôvodu, že bezdrôtové zariadenia, ktoré majú vlastný polohovací systém, odosielať údaje o svojej pozícii. Tie však nemusia byť správne. Táto funkcionálnosť slúži na porovnanie fyzicky nameraných súradníc s tými, ktoré sú získavané zo zariadenia.

Textové pole *note* a tlačítko *Send Data To DB* sú sprístupnené až po zaškrtnutí políčka *DB record* v ľavom rohu hlavičky panela. Takto je užívateľ nútený zamyslieť sa, či vyplnené údaje chce skutočne odoslať na server, čo je nenávratný proces. Tento spôsob predchádza chybám z nepozornosti (napríklad mylnému stlačeniu tlačítka).

Kliknutím na tlačítko *Send Data To DB* sú všetky údaje (identifikačné číslo, pôvodné aj nové hodnoty súradníc, vypočítané rozdiely a odchýlka, čas potvrdenia, prípadná krátka textová poznámka) odoslané do tabuľky databázového systému.

Potom je možné okno zrušiť alebo aplikovať zmeny aj v modele, kedy sa zmení vizuálna poloha jednotky v pláne a je tiež aktualizovaná tabuľka.

#### 8.4.6 Mazanie

Nie len vytváranie a editácia patria k úkonom efektívneho modelovania sieťových infraštruktúr. Rušenie aktívnych objektov je k dispozícii dvomi spôsobmi – hromadne a jednotlivo.

Ovládací prvok *Clear Data* odstráni kompletne všetky dáta.

V druhom prípade sú jednotlivé nevyhovujúce prvky za účelom vymazania postupne vyhadzované do koša. Pretiahnutie ktorejkoľvek ikony elektronického zariadenia na symbol odpadkového koša v pravom dolnom rohu je považované za snahu o odstránenie. Je pravdepodobné, čo sa stane po uvoľnení myši v tejto situácii: subjekt je vymazaný, o čom nás ešte uistí okno potvrdzujúce operáciu.

Pokiaľ je v pláne príliš veľa, prvkov pridávanie nových môže viesť k dezorientácii. Pomôcť by mohlo zmazanie objektov z plánov poschodí. Toto sa udeje tlačítkom *Hide Nodes*, symboly zmiznú, dáta zostanú nedotknuté. Užívateľ potom môže pokračovať v pridávaní nových objektov a kedykoľvek všetky opäť obnoviť tým istým tlačítkom.

#### 8.4.7 Ďalšie vlastnosti

Nesprávne vstupné dáta môžu vyvolať nepredvídateľné správanie nástroja, čomu je nutno predchádzať. Pri spracovávaní importovaných hodnôt sa dbá na korektnosť údajov ovplyvňujúcich hlavne pozíciu objektu. Pokiaľ súradnicové hodnoty importovaného uzla nezodpovedajú jednoznačne danému rozsahu – šírka a dĺžka budovy, ikona tohoto uzla je umiestnená do počiatku osy plánu. Okrem toho je do koloniek *X* a *Y* doplnený namiesto číselných hodnôt znak '-', čo dáva užívateľovi znamenie, že je vstupná hodnota pre tento plán nepoužiteľná. K získaniu platných súradníc stačí toto zariadenie presunúť.

Presun sieťových komponent je opäť riešený metódou *Drag and Drop*, kedy sa aktualizujú zároveň obidva pohľady aj tabuľkový záznam.

Výsledky práce je možné exportovať v rôznych formátoch.

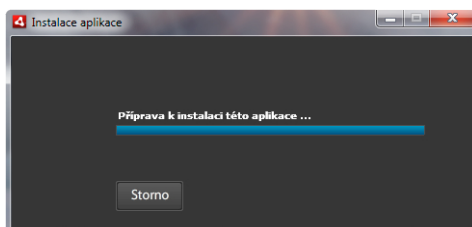
Správanie ovládacích komponent je odvodené od toho v mapovom module. Navyiac je pridaná nápoveda vo forme krátkeho popisu, ktorý sa zobrazí vždy po ustálení kurzora nad patričnou ovládacou jednotkou grafického rozhrania.

## 9 VERZIA DESKTOPOVÉHO PROGRAMU

Uživatel má možnost používat nástroj přímo z prostředí operačního systému svého počítače, mobilního telefonu, tabletu a iných zariadení.

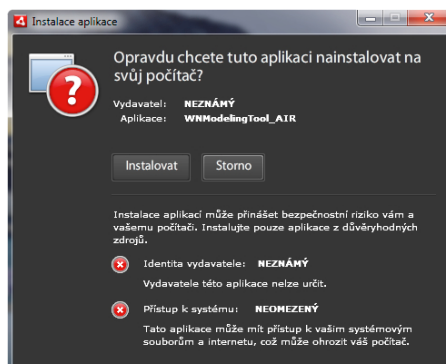
Aplikácia je plnohodnotným programom. Je nutné prejsť inštalačným procesom:

1. Po spustení inštalačného balíčka je zahájená příprava k inštalaci



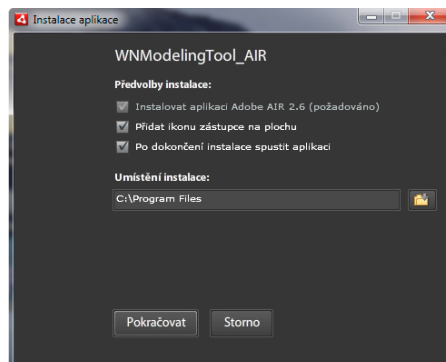
Obr. 9.1: Inštalácia AIR aplikácie – krok číslo 1

2. Následne sa zobrazí výzva pre potvrdenie inštalácie



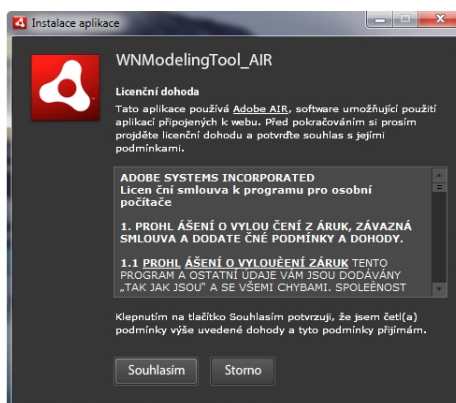
Obr. 9.2: Inštalácia AIR aplikácie – krok číslo 2

3. Možnosti inštalácie a výber cieľového umiestnenia



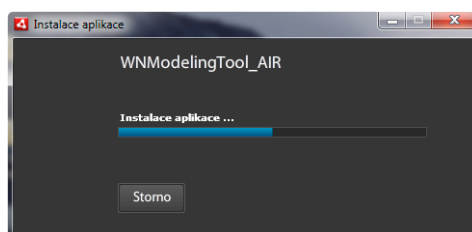
Obr. 9.3: Inštalácia AIR aplikácie – krok číslo 3

#### 4. Uzatvorenie dohody s Adobe AIR



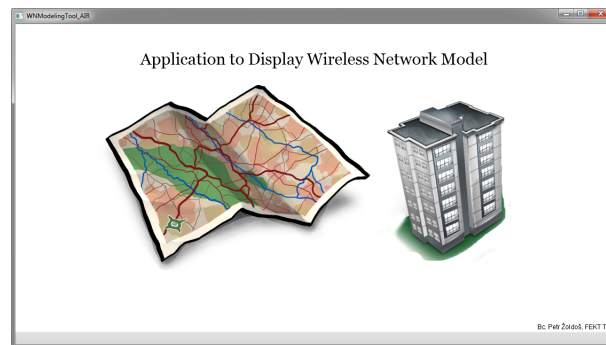
Obr. 9.4: Inštalácia AIR aplikácie – krok číslo 4

#### 5. Posledným krokom tohoto je už samotná inštalácia

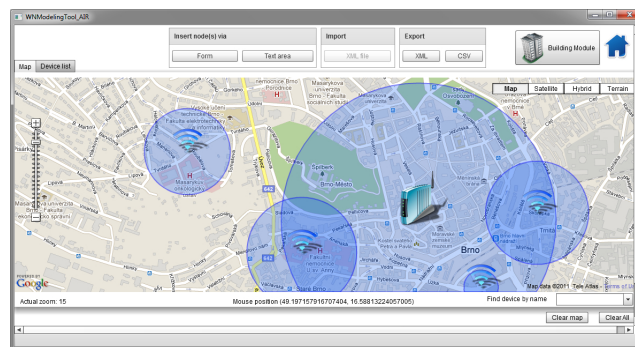


Obr. 9.5: Inštalácia AIR aplikácie – krok číslo 5

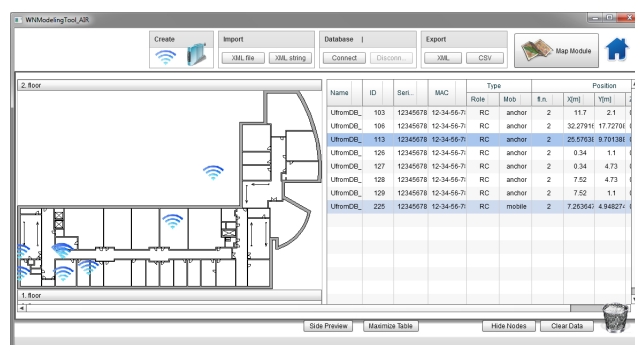
Po spustení nainštalovaného programu sa, tak ako v prípade webovej aplikácie, zobrazí úvodná stránka pre výber modulu 9.6 a užívateľ môže začať pracovať s nástrojom rovnako, akoby ho ovládal z prostredia prehliadača (viz. obr. 9.7, 9.8).



Obr. 9.6: Úvodná obrazovka v prostředí AIR



Obr. 9.7: Modul mapy v prostředí AIR



Obr. 9.8: Modul budovy v prostředí AIR

## 10 NASADENIE APLIKÁCIE V PRAXI

Produkt tejto práce bol vyvíjaný so zámerom a myšlienkou jej ďalšieho využitia v praxi, konkrétne pre potreby telekomunikačného ústavu Fakulty elektrotechniky a komunikačných technológií.

Výsledný program je aplikovaný a používaný za účelom monitorovania bezdrôtových senzorov, umiestnených vo viacerých učebniach. Hlavným použitým nástrojom je druhý modul, ktorý rieši otázku správy modelu siete v uzatvorenom objekte.

Statické senzory sú do aplikácie implementované manuálne. Tie senzory, ktoré sú v zmysle pohybu slobodnejšie, komunikujú s databázovým serverom, ktorému oznamujú súradnice svojej aktuálnej pozície. Tie na strane klienta odchyťávajú práve aplikácia pre generovanie modelu bezdrôtovej siete, čím sprostredkuje predstavu o robustnosti senzorovej siete digitálnou efektívnou formou.

### 10.1 Potenciál ďalšieho využitia

Hoci bolo k projektu pristupované s vedomím, že jeho využitie nájde konkrétne uplatnenie, primárnym požiadavkom nebolo splniť len tento cieľ. Zadanie kládlo ďalšie nároky na veľkosť, funkčnosť a univerzálnosť. Vďaka tomu nie je spôsob te-  
rajšieho využitia jediným možným a produkt tak má potenciál uplatnenia aj v iných oblastiach:

- sledovanie výskytu pripojených užívateľov v lokálnej sieti WLAN v budove alebo na širokom území
- pomoc pri návrhu, či úprave rozsiahlejšej siete tak, aby bolo zabezpečené dostatočné, nie však nadbytočné pokrytie signálu
- sledovanie vlastností zabezpečovacieho systému
- mapovanie polôh vysielačov mobilného operátora
- vyhodnocovanie štatistických údajov o hustote a počte obyvateľov využívajúcich konkrétnu službu na určitom území
- monitorovanie tlakových senzorov rozmiestnených na strechách priemyselných budov za účelom prevencie poškodenia vplyvom prírodných podmienok a ďalších.

### 10.2 Možnosti rozšírenia

Prípadný ďalší rozvoj tohoto produktu by mohol spočívať v pridávaní ďalších a nových funkcionalít ako:

- zavedenie načítania ďalšieho nezávislého modulu samotným užívateľom

- dynamické načítanie plánov poschodí užívateľom
- ukladanie dát každého načítania databázových záznamov do pamäte za účelom pozorovania trajektórie pohybu mobilných senzorov
- ponuka priblíženia plánových podkladov

alebo v zdokonaľovaní tých stávajúcich:

- efektívnejšie napojenie na databázový systém
- úprava dát priamo v tabuľkách bezdrôtových prvkov
- spresnenie zobrazenia druhej perspektívy - zavedenie viditeľných os s reálnymi pomerom jednotiek
- spresnenie zobrazenia dosahu signálu
- aplikácia rovnakých reakcií na udalosti v oboch zobrazeniach (zmena polohy, mazanie, atď.)
- presun uzlov naprieč poschodiami

Nech sa zdajú byť predložené návrhy akokoľvek jednoduché, za ich realizáciou stoja ďalšie s nimi prichádzajúce otázky riešenia a komplikácie, ktoré sú mimo rozsah požiadavkov zadania.



## 11 ZÁVER

V rámci diplomovej práce bol vybudovaný praktický nástroj pre správu modelu bezdrôtovej siete. Tento produkt umožňuje modelovať rozloženie sieťových jednotiek, sledovať ich reálnu polohu a vlastnosti. Výsledok sa dá uložiť v dvoch rôznych formátoch a ďalej s ním pracovať.

Bola navrhnutá XML štruktúra nesúca údaje vedúce k jednoznačnej identifikácii každého prvku, určeniu jeho polohy a vlastností. Táto štruktúra je využívaná ako možný súbor vstupných a výstupných dát.

Silnou stránkou projektu je multiplatformnosť. Produkt bol testovaný v najnovších verziách internetových prehliadačov Mozilla Firefox, Google Chrome, Microsoft Explorer, Safari. Testovanie desktopovej AIR verzie prebiehalo na operačných systémoch Mac OS X 10.6.7 (Snow Leopard) a Microsoft Windows 7. Využitie môže najásť aj v operačných systémoch mobilných zariadení ako Google Android, či iOS.

Aplikácia komunikuje so školským databázovým systémom. Na základe záznamov v ňom generuje model senzorovej siete rozmiestnenej na ústave telekomunikácií a tiež napomáha k ďalším meraniam a testovaniam zariadení.

Aplikácia je modulárna, čo zvyšuje jej stabilitu a spoľahlivosť a znižuje systémové nároky. Je odolná voči nesprávnemu ovládaniu (napr. zadávanie neplatných vstupných hodnôt) a užívateľa vedie a kontroluje pri jeho činnosti.

Softvér bol vyvíjaný sadou nástrojov Adobe FLEX a Google Maps SDK na operačnom systéme Mac OS X Snow Leopard. Grafické rozhranie bolo prispôsobené zobrazeniu na 13 palcovom monitore s rozlíšením 1280x800, kedy je zobrazenie ideálne.

Praktický výsledok je volne prístupný na URL adrese telekomunikačného ústavu <<http://wsnbase.utko.feec.vutbr.cz/WNModelingTool/Starter.html>> alebo na priloženom DVD médiu vo verzii desktopového programu.

Písomná časť zoznámila s použitými technológiami, postupmi a riešeniami. Boli predstavené dôležité časti zdrojového kódu, ukážky grafického rozhrania a bol vysvetlený spôsob používania tohoto softvérového modelovacieho nástroja.

Práca sa zúčastnila študentskej súťaže EEICT student. Dostala sa do finálového kola, kde sa stretla s pozitívnymi reakciami a veľkým záujmom poroty. Vyvolala údiv nad rozsiahlosťou celého projektu a prebehla diskusia o možnosti využitií aj na iných fakultách VUT. Pri konečnom hodnotení však bohužiaľ doplatila na chybné zaradenie recenzentom do súťažnej kategórie, ktoré nebolo možné nijak ovplyvniť. Verím, že v inom prípade by sa projekt napriek veľkej konkurencii dokázal umiestniť medzi tromi najlepšími.

## LITERATURA

- [1] Li, Yingshu, Thai, My T., Wu, Weili, *Wireless Sensor Networks and Applications, Signals and Communication Technology*, 2008, p. 444, ISBN: 978-0-387-49591-0
- [2] VAJSAR, P.; MORÁVEK, P. Solution for wireless sensor network monitoring. In *6th International Conference on Teleinformatics*. 2011. s. 1-4. ISBN: 978-80-214-4231- 3.
- [3] *Adobe* [online].Adobe Systems Incorporated, ©2010, 07-14-2009 [cit. 2010-11-02]. Open source software | Flex | Adobe. Dostupné z WWW: <<http://www.adobe.com/products/flex/overview/>>.
- [4] *Adobe Help* [online].Adobe Systems Incorporated, ©2010, Mon Dec 6 2010, 07:55 AM -08:00 [cit. 2010-12-12]. Flash.net.FileRefernce – ActionScript 3.0 Reference for the Adobe Flash Platform. Dostupné z WWW: <[http://help.adobe.com/en\\_US/FlashPlatform/reference/actionscript/3/flash/net/FileReference.html](http://help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/flash/net/FileReference.html)>.
- [5] Adobe Systems Incorporated. *Adobe* [online]. 2011 [cit. 2011-04-26]. Browser vs. application. Dostupné z WWW: <<http://www.adobe.com/products/air/comparison/>>.
- [6] *Google code* [online].Google, ©2010 [cit. 2010-11-15]. Site Directory – Google Code. Dostupné z WWW: <<http://code.google.com/intl/cs-CZ/more/>>.
- [7] *Google code* [online].Google, ©2010 [cit. 2010-11-16]. Google Maps API for Flash – Google Code. Dostupné z WWW: <<http://code.google.com/apis/maps/documentation/flash/>>.
- [8] *Google code* [online].Google, ©2010 [cit. 2010-11-15]. Sign Up for the Google Maps API – Google Maps API Family – Google Code. Dostupné z WWW: <<http://code.google.com/apis/maps/signup.html>>.
- [9] *Google code* [online].Google, ©2010 [cit. 2010-11-16]. FAQ – Google Maps API Family – Google Code. Dostupné z WWW: <<http://code.google.com/apis/maps/faq.html>>.
- [10] *Google code* [online].Google, ©2010 [cit. 2010-11-23]. Google Maps API ActionScript Reference – Google Maps API for Flash – Google Code. Dostupné z WWW: <<http://code.google.com/apis/maps/documentation/flash/reference.html>>.

- [11] Matt Rosenberg, About.com Guide – *Latitude and Longitude: Discover the Secrets of Parallels and Meridians*[online]. 2010, dostupné z URL: <<http://geography.about.com/cs/latitudelongitude/a/latlong.htm>>.
- [12] Adobe Systems Incorporated. *Adobe Flex 4 : About style value formats* [online]. 2010 [cit. 2011-04-03]. About style value formats. Dostupné z WWW: <[http://help.adobe.com/en\\_US/flex/using/WS2db454920e96a9e51e63e3d11c0bf62883-7ff7.html](http://help.adobe.com/en_US/flex/using/WS2db454920e96a9e51e63e3d11c0bf62883-7ff7.html)>.

## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

- RIA robustné internetové aplikácie – Rich Internet Applications
- AIR integrovaný prevádzkový *runtime* systém –Adobe Integrated Runtime
- SDK sada nástrojov pre vývoj softvérových aplikácií – Software Development Kit
- API rozhranie pre vývoj softvérových aplikácií – Application Programming Interface
- SWF súbor informácií o vektorovej grafike, texte, videu, zvuku – Small Web Format
- MXML jazyk založený na technológií XML – Macromedia eXtensible Markup Language
- XML rozšíriteľný značkovací jazyk – eXtensible Markup Language
- CSV čiarkou oddelené hodnoty (formát pre podporu tabuliek) – Comma-separated values
- HTML číslicové zpracování signálů – HyperText Markup Language
- CSS kaskádové štýly – Cascading Style Sheets
- WLAN lokálna bezdrôtová sieť – Wireless Local Area Network
- GPS globálny polohovací systém – Global Positioning System
- PNG grafický formát pre bezstratovú kompresiu – Portable Network Graphics

# SEZNAM PŘÍLOH

<b>A Prvá příloha</b>	<b>62</b>
A.1 XML štruktúra pre model siete na širšom území . . . . .	62
A.2 XML štruktúra pre model siete v budove . . . . .	63
A.3 CSV štruktúra . . . . .	64
<b>B Druhá príloha</b>	<b>65</b>
B.1 Priložené médium . . . . .	65

## A PRVÁ PRÍLOHA

### A.1 XML štruktúra pre model siete na širšom území

```
<network>
  <node type='AP' nodeID='1' name='WN X01' serialNum='12356'
    macAdd='01-23-45-67-bx' latitude='49.192250081461545'
    longitude='16.60740123013304'
    scope='800' />
  <node type='RC' nodeID='2' name='WN 02' serialNum='54789'
    macAdd='02-23-45-67-bx' latitude='49.19267077213424'
    longitude='16.61690697888182' scope='300' />
  <node type='RC' nodeID='3' name='WN 03' serialNum='12598'
    macAdd='03-45-67-89-ax' latitude='49.188772234870854'
    longitude='16.6124008677368' scope='100' />
  <node type='RC' nodeID='4' name='WN 04' serialNum='34587'
    macAdd='04-45-67-89-ax' latitude='49.19059533010512'
    longitude='16.598024227416975' scope='320' />
  <node type='RC' nodeID='5' name='WN 05' serialNum='54876'
    macAdd='05-45-67-89-ax' latitude='49.195867904332566'
    longitude='16.588840343749983' scope='250' />
</network>
```

## A.2 XML štruktúra pre model siete v budove

```
<network>
  <node name="WN X01" nodeID="1" serialNum="12356"
    macAdd="01-23-45-67-bx" type="AP" mobility="anchor"
    positionX="48" positionY="20" floorNum="2"
    timeStmp="Thu Mar 10 23:11:54 2011"/>
  <node name="WN 02" nodeID="2" serialNum="54789"
    macAdd="02-23-45-67-bx" type="RC" mobility="anchor"
    positionX="45" positionY="25" floorNum="2"
    timeStmp="Thu Mar 10 23:11:56 2011"/>
  <node name="WN 03" nodeID="3" serialNum="12598"
    macAdd="03-45-67-89-ax" type="RC" mobility="anchor"
    positionX="17" positionY="29" floorNum="2"
    timeStmp="Thu Mar 10 23:12:00 2011"/>
  <node name="WN 04" nodeID="4" serialNum="34587"
    macAdd="04-45-67-89-ax" type="RC" mobility="anchor"
    positionX="19" positionY="16" floorNum="1"
    timeStmp="Fri May 06 16:49:05 2011"/>
  <node name="WN 05" nodeID="5" serialNum="54876"
    macAdd="05-45-67-89-ax" type="RC" mobility="anchor"
    positionX="17" positionY="29" floorNum="0"
    timeStmp="Thu Mar 10 23:11:54 2011"/>
</network>
```

## A.3 CSV štruktúra

Príklad exportovanej CSV štruktúry pre obidva modely siete je na priloženom CD. Preklad obsahu CSV formátu v programe OpenOffice (alternatívne Microsoft Excel):

	A	B	C	D	E	F	G	H	I	J
1	network									
2	node									
3	name	nodeID	serialNum	macAdd	type	mobility	positionX	positionY	floorNum	timeStmp
4	WN X01	1	12356	01-23-45-67-bx	AP	anchor	48	20	2	Thu Mar 10 23:11:54 2011
5	WN 02	2	54789	02-23-45-67-bx	RC	anchor	45	25	2	Thu Mar 10 23:11:56 2011
6	WN 03	3	12598	03-45-67-89-ax	RC	anchor	17	29	2	Thu Mar 10 23:12:00 2011
7	WN 04	4	34587	04-45-67-89-ax	RC	anchor	19	16	1	Fri May 06 16:49:05 2011
8	WN 05	5	54876	05-45-67-89-ax	RC	anchor	17	29	0	Thu Mar 10 23:11:54 2011
9										

Obr. A.1: Preklad exporotvaných dát formátu CSV



## B DRUHÁ PRÍLOHA

### B.1 Priložené médium

Na priloženom DVD nosiči sa nachádza:

- písomná časť diplomovej práce v elektronickej podobe vo formáte PDF
- zdrojové kódy projektov Flex a Air
  - WNModelingTool
  - WNModelinTool\_AIR
- zdrojové kódy JAVA aplikácie
  - DbConnection\_JAVAapp
- súbor s obsahom navrhnutej jednotnej XML štruktúry pre obidva moduly používaný aj pri testovaní projektu
  - Map\_and\_building\_nodes\_import.xml
- textový súbor nesúci použité SQL dotazy pre komunikáciu s databázovým systémom
  - Pouzite\_SQL\_dotazy.txt
- súbory s obsahom exportovanej siete vo formátoch XML a CSV
  - Export\_modul\_mapy.xml
  - Export\_modul\_mapy.csv
  - Export\_modul\_budovy.xml
  - Export\_modul\_budovy.csv
- inštalačné súbory aplikácie vo verzii desktopového programu pre platformy
  - Adobe AIR: WSN Tool.air
  - MacOS: WSN Tool.dmg
- digitálny certifikát, ktorým je podpísaný desktopový projekt
  - WSN Tool Certificate.p12
- článok do súťaže EEICT student
  - EEICT\_clanok.pdf