

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER SYSTEMS

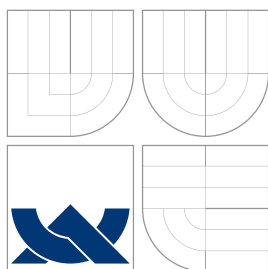
## POUŽITÍ TELEFONU JAKO BEZDRÁTOVÉ MYŠI

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

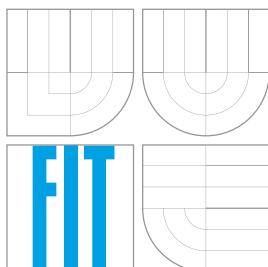
AUTOR PRÁCE  
AUTHOR

MARTIN ELIÁŠ

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER SYSTEMS

# POUŽITÍ TELEFONU JAKO BEZDRÁTOVÉ MYŠI

USING CELL PHONE AS WIRELESS MOUSE

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

MARTIN ELIÁŠ

VEDOUCÍ PRÁCE  
SUPERVISOR

Ing. PETR POSPÍCHAL

BRNO 2011

## Abstrakt

Tato bakalářská práce se zabývá návrhem a implementací aplikace, která umožňuje použít mobilní zařízení s operačním systémem Google Android jako bezdrátový touchpad a klávesnici pro ovládání počítače s operačním systémem Linux. Aplikace funguje na standardní síťové architektuře klient-server. Mobilní zařízení se do počítačové sítě připojuje pomocí bezdrátového síťového rozhraní WiFi. Pro vývoj jednotlivých částí byly použity programovací jazyky Java a C++. Práce se dále zabývá testováním této aplikace a možnostmi jejího dalšího vývoje.

## Abstract

This bachelor's thesis describes the design and implementation of an application that allows to use a mobile device running the Google Android operating system as a wireless touchpad and keyboard to control a computer running the Linux operating system. It works on the standard client-server network model. The mobile device connects to the computer network via a wireless WiFi network interface. For the development of the individual components were used Java and C++ programming languages. The thesis also deals with the testing of the application and possibilities for its further development.

## Klíčová slova

RemotePad, Android, Linux, Java, C++, touchpad, klávesnice, ovládání počítače, bezdrátová komunikace, WiFi

## Keywords

RemotePad, Android, Linux, Java, C++, touchpad, keyboard, computer controlling, wireless communication, WiFi

## Citace

Martin Eliáš: Použití telefonu jako bezdrátové myši, bakalářská práce, Brno, FIT VUT v Brně, 2011

# Použití telefonu jako bezdrátové myši

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Petra Pospíchala.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Martin Eliáš  
17. května 2011

## Poděkování

Chtěl bych poděkovat panu Ing. Petru Pospíchalovi za odbornou pomoc a psychickou podporu při vzniku této bakalářské práce.

© Martin Eliáš, 2011.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

# Obsah

<b>1 Úvod</b>	<b>3</b>
1.1 Obsah kapitol . . . . .	3
<b>2 Mobilní operační systém</b>	<b>4</b>
2.1 Google Android . . . . .	4
2.1.1 Historie . . . . .	4
2.1.2 Rozšíření na trhu . . . . .	4
2.1.3 Architektura . . . . .	5
2.1.4 Vývoj aplikací . . . . .	6
2.1.5 Verze systému . . . . .	7
<b>3 Komunikace</b>	<b>11</b>
3.1 Možnosti bezdrátového propojení mobilního přístroje a PC . . . . .	11
3.1.1 Bluetooth . . . . .	11
3.1.2 Wi-Fi . . . . .	13
3.2 Volba technologie bezdrátové komunikace . . . . .	14
3.3 Model ISO/OSI . . . . .	14
3.4 IP adresa . . . . .	15
3.5 IP Broadcast . . . . .	16
3.6 Architektura server-klient . . . . .	16
<b>4 Zadání práce</b>	<b>17</b>
4.1 Návrh funkcí aplikace . . . . .	17
4.2 Kompatibilita s operačními systémy . . . . .	17
<b>5 Uživatelské rozhraní a vzhled</b>	<b>18</b>
5.1 Uživatelské rozhraní - software . . . . .	18
5.1.1 Rozhraní v PC . . . . .	18
5.1.2 Rozhraní v mobilním přístroji . . . . .	19
5.2 Uživatelské rozhraní - hardware . . . . .	20
5.2.1 Rezistivní technologie dotykových displejů . . . . .	20
5.2.2 Kapacitní technologie dotykových displejů . . . . .	20
<b>6 Implementace</b>	<b>21</b>
6.1 Vývojové nástroje . . . . .	21
6.1.1 Android SDK . . . . .	21
6.1.2 Editor programového kódu Eclipse IDE . . . . .	22
6.1.3 Editor programového kódu Qt Creator IDE . . . . .	22
6.1.4 Použité programovací jazyky . . . . .	22
6.2 Aplikace v mobilním přístroji . . . . .	23
6.3 Aplikace v PC . . . . .	24

<b>7</b>	<b>Testování</b>	<b>26</b>
7.1	Metody testování . . . . .	26
7.2	Výsledky testování . . . . .	27
<b>8</b>	<b>Závěr</b>	<b>28</b>
8.1	Postup práce . . . . .	28
8.2	Dosažené výsledky . . . . .	28
8.3	Možnosti využití . . . . .	29
8.4	Možnosti budoucího rozšíření . . . . .	29
<b>A</b>	<b>Obsah CD</b>	<b>32</b>
<b>B</b>	<b>Uživatelská dokumentace</b>	<b>33</b>
B.1	O aplikaci . . . . .	33
B.2	Požadavky . . . . .	33
B.3	Instalace . . . . .	33
B.4	Ovládání . . . . .	33
B.4.1	Server - Linux . . . . .	33
B.4.2	Klient - Android . . . . .	34

# Kapitola 1

## Úvod

Neustálý a intenzivní vývoj technologií v oblasti osobních počítačů nezpůsobuje pouze rostoucí výkon těchto zařízení, ale má také dopad na jejich rozměry a využití zdrojů energie. Asi bychom si dnes dokázali jen stěží představit mít doma počítač pokrývající plochu několika desítek metrů čtverečních, vážící desítky tun a s příkonem v řádu stovek kilowatt, jako tomu bylo u počítače ENIAC<sup>1</sup> ve 40. letech minulého století. Díky rostoucímu výkonu, klesající pořizovací ceně a stále menším rozměrům se postupem času počítače staly, ať už na poli práce či zábavy, nedílnou součástí života každého člověka.

Na základě trendů poslední doby a nároků na přenositelnost osobních počítačů začaly vznikat takzvané kapesní počítače. Ve spojení s telekomunikačními technologiemi jsou tato zařízení nazývána chytrými telefony. Dokonce i přes kompaktní rozměry mnoho z nich svým výpočetním výkonem převyšuje dnes už starší, ale stále běžně používané stolní počítače. Účelem těchto mobilních přístrojů je pomoci uživateli s organizací osobního i pracovního života, poskytnout hledané informace nebo také zprostředkovat zábavu. Dalo by se říct, že jde o určitou formu osobního pomocníka. Cílem této práce bude vytvořit aplikaci pro mobilní platformu, která díky technologii dotykového displeje a bezdrátového spojení s počítačem funguje jako touchpad.

### 1.1 Obsah kapitol

Teoretický úvod o mobilních operačních systémech a o systému Google Android se nachází ve 2. kapitole. 3. kapitola pojednává o problematice síťové komunikace mobilního zařízení s počítačem a o použitelných bezdrátových technologiích. 4. kapitola definuje zadání práce a kompatibilitu s operačními systémy. V 5. kapitole je popsán návrh softwarového a hardwarového uživatelského rozhraní včetně popisu technologií dotykových displejů. V kapitole 6. si přiblížíme použité vývojové nástroje, prostředí a programovací jazyky spolu s postupem implementace aplikace na obou platformách (PC, mobilní přístroj). Na testování a analýzu výsledků je zaměřena kapitola č. 7. Postup práce, popis dosažených výsledků, možnosti využití aplikace a případná budoucí rozšíření obsahuje závěrečná kapitola č. 8.

---

<sup>1</sup><http://en.wikipedia.org/wiki/ENIAC>

## Kapitola 2

# Mobilní operační systém

Stejně jako stolní počítače, i mobilní zařízení mohou mít své operační systémy. V principu jsou těm klasickým podobné. Jsou však znatelně jednodušší a přizpůsobené hardwarové platformě, nad kterou běží. Mobilní operační systémy výrobci nejčastěji instalují do chytrých telefonů, tabletů a PDA. Existuje řada různých výrobců takových systémů. Mezi nejznámější systémy patří Symbian, Microsoft Windows Mobile, Microsoft Windows Phone, Apple iOS, RIM BlackBerry OS nebo Google Android.

## 2.1 Google Android

### 2.1.1 Historie

Americká společnost Google, sídlící v Silicon Valley v Kalifornii, je v dnešní době proslulá nejen díky internetovému vyhledávači. Tento gigant je tvořen několika divizemi zabývajícími se tvorbou různých produktů ve světě IT, přičemž jedna z nich poskytuje mobilní operační systém pro chytré telefony a tablety. Systém založený na linuxovém jádře nese název Android vycházející ze jména firmy Android Inc., která tuto softwarovou platformu původně vyvíjela, ale v roce 2005 byla Googlem převzata formou akvizice. 5. listopadu 2007 Google celou platformu předal sdružení Open Handset Alliance (OHA)<sup>1</sup>, které sám založil a je také jeho členem. Původně OHA tvořilo 34 různých hardwarových, softwarových výrobců a telekomunikačních společností, které se tímto podílejí na prosazování otevřených standardů ve světě mobilních zařízení. Od 23. září roku 2008, kdy byla oficiálně představena první verze systému Android, jsou všechny součásti této platformy veřejně dostupné pod licencí Apache<sup>2</sup> zdarma. Jedná se tedy o poměrně mladý open source operační systém.

### 2.1.2 Rozšíření na trhu

Operační systém Android zažil během svého krátkého působení na trhu velmi rychlý vzestup v oblasti chytrých telefonů. Výzkumná společnost Canalys odhadovala, že k druhému čtvrtletí roku 2009 měl Android podíl 2,8% na světovém trhu mezi chytrými telefony.[1] V únoru 2010 společnost ComScore řekla, že platforma Android má 9% podíl na trhu s chytrými telefony ve Spojených státech. Za první čtvrtletí roku 2010 překonal v USA prodej Androidu konkurenční platformu Apple iOS. Ze zprávy NPD group vycházelo, že zařízení s platformou Android v této době činilo 25% všech prodaných chytrých telefonů v USA. Celosvětově to podle Googlu znamenalo 100 000 nově aktivovaných mobilních telefonů s Androidem denně.[2] V prosinci 2010 se Google nechal slyšet, že se toto číslo zvýšilo dokonce na 300 000.[5]

---

<sup>1</sup><http://www.openhandsetalliance.com/>

<sup>2</sup><http://www.apache.org/licenses/>



Co stojí za takovým úspěchem? Jednoduchý fakt, že Android OS je kompletně open source. Výrobci mobilních zařízení zdarma stáhnou programový kód, který můžou kompletně přizpůsobit podle svých představ. Výsledkem pak jsou jejich vlastní aplikace usnadňující přístup většinou k internetovým službám či různé grafické a funkční uživatelské nastavení jako HTC Sense, Samsung TouchWiz apod., které zpříjemňují práci s mobilním zařízením. Mezi těmi, kdo systém Android do svých zařízení instalují, je řada předních světových výrobců mobilních telefonů, tabletů a netbooků jako například Acer, Dell, HTC Corp., Huawei, Lenovo, LG Group, Motorola, Samsung Group nebo Sony Ericsson. V budoucnu by se OS Android mohl rozšířit třeba do televizních přijímačů nebo hudebních přehrávačů. Za úspěchem v neposlední řadě stojí také jméno jedné z ekonomicky nejsilnějších IT společností na světě - Google.

### 2.1.3 Architektura

Android OS je postaven na pěti základních komponentách:

- Aplikace
- Aplikační framework
- Knihovny
- Android runtime
- Linuxový kernel

**Aplikace** zajišťují základní funkčnost mobilního zařízení. Jedná se například o telefon, kalendář, emailový klient, SMS program, mapy, internetový prohlížeč nebo kontakty. Všechny aplikace jsou psány v jazyce Java.

**Aplikační framework** je softwarová struktura, která pomáhá vývojářům při programování aplikací. Android nabízí možnost vytvářet extrémně bohaté a inovativní aplikace. Vývojáři mají možnost využívat výhod hardware přístroje, přistupovat k lokálním informacím, spouštět služby na pozadí, nastavovat alarmy, přidávat oznámení do stavového řádku a další. Jejich aplikace mají plný přístup ke stejným API, která používají jádrové aplikace systému. Aplikační architektura je navržena pro zjednodušení opakovaného používání komponent. Kterákoliv aplikace může zveřejnit své možnosti a jakákoliv jiná aplikace může těchto možností využít. Takový mechanismus dovoluje komponenty nahrazovat uživatelem.

Android zahrnuje sadu C/C++ **knihoven**, které jsou používány různými součástmi systému. Tyto možnosti jsou vývojářům k dispozici skrze aplikační framework. Některé z těchto knihoven jsou vyjmenovány níže.

- Systémová knihovna C - implementace standardní knihovny jazyka C (libc) odvozené z BSD a vyladěné pro vestavěná linuxově založená zařízení.
- Knihovny médií - založené na PacketVideo OpenCORE<sup>3</sup> systému, které podporují přehrávání a záznam mnoha populárních audio a video formátů i zobrazení obrázků (např.: MPEG4, H.264, MP3, AAC, AMR, JPG nebo PNG).
- Povrchový manažer - řídí přístup k podsystému displeje a bezproblémově skládá vrstvy 2D a 3D grafiky aplikací.
- LibWebCore - moderní webový prohlížeč engine, který pohání prohlížeč systému Android a vestavěný webový pohled.
- SGL - základní prvek grafického 2D enginu.

---

<sup>3</sup><http://www.packetvideo.com/>

- 3D knihovny - implementace založená na OpenGL ES<sup>4</sup> verze 1.0. Knihovny používají buď hardwarovou 3D akceleraci (pokud je dostupná), nebo vysoce optimalizovaný softwarový 3D rastrovací nástroj obsažený v systému.
- FreeType - rastrové a vektorové zobrazení písma.
- SQLite - mocný a jednoduchý engine relační databáze dostupný všem aplikacím.

**Android runtime** - Android obsahuje soubor knihoven jádra, které poskytují většinu funkcionality jádrových knihoven programovacího jazyka Java.

**Linuxový kernel** se chová jako abstraktní vrstva mezi hardware a software přístroje. Android se opírá o Linux verze 2.6 pro systémové služby jádra jako je bezpečnost, řízení paměti, řízení procesů, obsluha sítě a další.

Systém Android využívá výše zmíněných vlastností a dovoluje například současný běh více aplikací, jde tedy o určitou formu multi-taskingu. Každá aplikace běží jako samostatný proces pod vlastním názvem ve vlastním Java virtuálním stroji. To vede ke zvýšení bezpečnosti systému a izolaci aplikací. Pokud běží aplikace na pozadí, operační systém se stará o její uspaní či automatické ukončení v případě nedostatku systémových zdrojů. Popis životního cyklu aktivity aplikace můžeme vidět na obrázku 2.1 na konci kapitoly. API také dovoluje přístup k síťovému připojení či periferním zařízením jako je například vestavěný GPS<sup>5</sup> modul. V rámci aplikace je pak možné tyto prostředky využívat, ovšem pouze za předpokladu schválení takového přístupu při instalaci dané aplikace uživatelem mobilního zařízení.

#### 2.1.4 Vývoj aplikací

První ohlasy na vývoj aplikací pro systém Android nebyly úplně příznivé. Objevovaly se stížnosti na nedostatky v dokumentaci, neadekvátní strukturu řízení kvality, chybějící systém sledování chyb a další. Přesto týden po oznámení platformy Android spatřily světlo světa první aplikace pro tento systém. První veřejně dostupnou aplikací byla hra Snake. Vývojáři aplikací mohou pro testování využít softwarový emulátor nebo jakékoliv reálné zařízení s nainstalovaným Androidem. Pro pokročilé vývojáře jsou pak určeny především zařízení s označením "Pure Google Experience". Tyto přístroje nejsou blokovány pro žádného telefonního operátora a obsahují pouze čistý operační systém Android bez jakékoliv grafické či funkční nástavby. Navíc je možné do nich instalovat verzi systému Android dle vlastní vůle. Aktuálně jsou na trhu dostupná dvě taková zařízení, mobilní telefon Nexus One vyrobený společností HTC a Nexus S od Samsungu. Nakonec lze takto využít i libovolné brandované zařízení. Na internetu je možné získat návody pro jejich odblokování, takový postup se nazývá rooting. V podstatě jde o získání práv uživatele root. Problém je, že taková změna v mobilním zařízení často znamená konec platnosti záruky tohoto výrobku.

Pokud bychom chtěli do mobilního zařízení se systémem Android nainstalovat aplikaci, máme několik možností. Stáhnutí z internetu, odesláním z jiného zařízení, například pomocí bezdrátové technologie bluetooth, zkopírováním z počítače nebo nejjednodušeji prostřednictvím aplikace Android Market. Android Market je aplikace vyvinutá Googlem, která je předinstalována do většiny Androidích zařízení. Prostřednictvím připojení k internetu nabízí uživatelům aplikace vyvinuté jinými uživateli. Některé aplikace je možné instalovat zdarma, za některé je ovšem nutné zaplatit. Kolik bude stát záleží pouze na jejím autorovi. Za každou staženou aplikaci utrží autor 70% z její ceny. V prosinci 2010 bylo v Android Marketu dostupných 200 000 her, aplikací a widgetů přičemž počet všech stáhnutí přesáhl hranici 2,5 miliardy.[19] Aplikace mají koncovku .apk a v operačním systému Android jsou uloženy ve složce /data/app, která je z bezpečnostních důvodů přístupná pouze uživateli root.

<sup>4</sup>[http://en.wikipedia.org/wiki/OpenGL\\_ES](http://en.wikipedia.org/wiki/OpenGL_ES)

<sup>5</sup><http://www.gps.gov/>

Dalším, kdo do Android Marketu přispívá svými výtvoři je pochopitelně i samotný Google. Rozšiřuje tím své internetové služby také do mobilních zařízení. Jedná se například o aplikace Google Maps, Google Goggles, Google Translate, Google Shopper a jiné. V srpnu 2010 byla spuštěna aplikace Google Voice Actions, která umožňuje uživatelům systému Android vyhledávat, psát zprávy, emaily nebo zahajovat hovory pomocí hlasu. V Marketu se objevují také aplikace třetích stran.

Google také nezapomněl na správnou motivaci pro vývojáře. Android Developer Challenge soutěže se účastní tisíce vývojářů a jejich úkolem je naprogramovat inovativní, kvalitní aplikace. Nejdůležitějším faktorem při hodnocení v soutěži jsou běžní uživatelé. V prvním kole vyberou svým hodnocením soutěžních aplikací dostupných v Android Marketu 200 aplikací (20 v každé z 10 kategorií), které postupují do kola druhého. Hodnocení uživateli v něm probíhá stejně jako v předchozím kole, nicméně toto se projeví jako 40% celkového skóre. Zbýlých 60% tvoří rozhodnutí týmu rozhodčích, které vybere Google z členů sdružení Open Handset Alliance, mobilních expertů a ze svých řad. Hodnoceny jsou vždy první 3 pozice v každé kategorii, navíc jsou vybráni i 3 celkoví vítězové napříč všemi kategoriemi. K dnešnímu datu proběhla Android Developer Challenge dvakrát a za odměny v obou těchto soutěžích dohromady utratil Google neuvěřitelných 10 milionů dolarů.

Asi nejlepší možností, jak začít programovat aplikace pro Android je s pomocí SDK neboli Software Development Kit<sup>6</sup>. Je to podobné jako u jiných platform. Android SDK je dostupný na oficiálním webu pro vývojáře a obsahuje dokumentaci, knihovny, debugger, tutoriály, vzorové kódy a softwarový emulátor zařízení. Na výběr jsou různé verze SDK, které jsou kompatibilní se všemi dnes běžnými operačními systémy osobních počítačů (Windows XP a vyšší, Linux, Mac OS X). Oficiálně podporované vývojové prostředí pro tvorbu kódu se nazývá Eclipse IDE, aktuálně ve verzi 3.4 a 3.5, do kterého je nutné nainstalovat plugin s vývojářskými nástroji pro Android. V podstatě lze ale psát kód v jakémkoliv jiném textovém editoru a následně program přeložit pomocí vývojových nástrojů Java. Výhodou je, že i v nejnovější verzi SDK je možné spustit v emulátoru také starší verze systému Android (konkrétně od verze 1.5 výše), což je možné využít při řešení problému s kompatibilitou aplikací mezi verzemi. V srpnu 2008 vyšla betaverze 0.9 Android SDK a o měsíc později 1. vydání verze 1.0. Aktuálně nejnovější je revize číslo 8 z prosince roku 2010.

Android OS nepoužívá standardy jako JAVA SE nebo JAVA ME, aby znemožnil kompatibilitu takto napsaných aplikací s aplikacemi pro Android. To může být ovšem jednou z překážek při programování. Android používá syntax jazyka Java, avšak neposkytuje knihovny celých tříd a API svázané s platformami Java SE a Java ME. Naštěstí existuje na trhu mnoho nástrojů, které umožňují aplikaci napsanou v J2ME konvertovat pro systém Android. Dalším problémem je nekompatibilita mezi různými verzemi systému, obzvláště různá podpora rozlišení displejů mezi verzemi 1.5 a 1.6, což zdůraznili účastníci soutěže Android Developer Challenge. Díky rychlému rozšíření Androidu je také velice obtížné napsat aplikaci fungující na širokém spektru zařízení s různými hardwarovými schopnostmi. K datu 4. ledna 2011 už sice 87% všech zařízení běží na operačním systému Android 2.1 a vyšším, ale více jak 12% má stále nainstalovanou verzi 1.5 nebo 1.6.[11] Kompletní statistiku je možné vidět v tabulce 2.1 níže.

Jednodušší možností pro tvorbu aplikací je Google App Inventor. Je to webová aplikace určená pro programátory začátečníky nebo i pro lidi, kteří o programování nic nevědí. Metodou "táhni a pusť" zde uživatelé pracují s vizuálními objekty, spojují je dohromady a to jim umožňuje vytvářet jednodušší aplikace, které například pro svá zařízení zrovna potřebují. Přístupný veřejnosti je Google App Inventor od 15. prosince 2010.

### 2.1.5 Verze systému

- 23. 09. 2008 - 1.0

---

<sup>6</sup><http://developer.android.com/sdk/index.html>

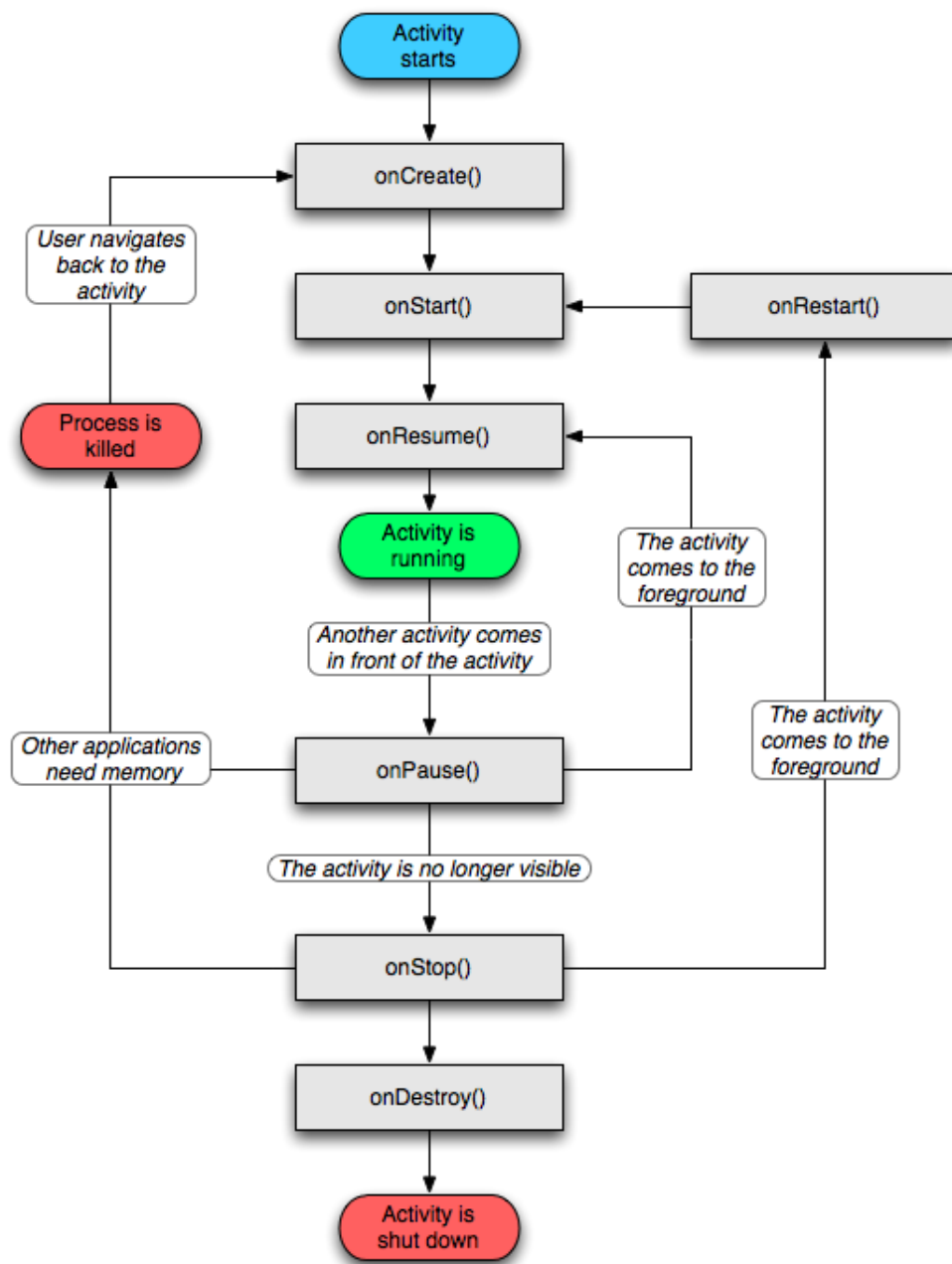
Platforma	API úroveň	Rozšíření
Android 1.5	3	4,7%
Android 1.6	4	7,9%
Android 2.1	7	35,2%
Android 2.2	8	51,8%
Android 2.3	9	0,4%

Tabulka 2.1: Výskyt OS Android podle verze ke 4.1.2011

- 09. 02. 2009 - **1.1** (pouze pro T-Mobile G1)
  - Změny API
  - Rozšíření Google map
  - Možnost skrýt softwarovou klávesnici
- 30. 04. 2009 - **1.5 Cupcake**
  - Nahrávání, přehrávání videa
  - Upload videa na Youtube, obrázky na Picasa
  - Predikce textu
  - Podpora Bluetooth A2DP
  - Animované přechody
  - Rozšíření funkce kopírovat&vložit
- 15. 09. 2009 - **1.6 Donut**
  - Nový Android Market
  - Aktualizace vyhledávání hlasem
  - Podpora WVGA rozlišení displeje
  - Podpora Bluetooth A2DP
- 26. 10. 2009 - **2.0/2.1 Eclair**
  - Podpora různých velikostí a rozlišení displeje
  - Podpora HTML5
  - Nové Google Mapy (v.3.1.2)
  - Podpora Microsoft Exchange
  - Podpora přisvětlovací LED
  - Podpora Bluetooth 2.1
  - Animované tapety na ploše
- 20. 05. 2010 - **2.2 Froyo**
  - Podpora Adobe Flash 10.1
  - Optimalizace správy paměti a rychlosti systému
  - Možnost instalovat aplikace na paměťovou kartu
  - Možnost vytvořit WiFi hotspot
  - Sdílení připojení k internetu s počítačem pomocí kabelu
  - Podpora OpenGL ES 2.0

- 06. 12. 2010 - **2.3 Gingerbread**
  - Změna vzhledu uživatelského rozhraní
  - Podpora většího displeje telefonu
  - Podpora SIP protokolu pro VoIP telefonii
  - Rozšíření funkce kopírovat&vložit do celého systému
  - Vylepšená podpora pro vývojáře aplikací
  - Podpora více fotoaparátů
  - Download manažer
  - Vylepšena úspora energie
  - Změna souborového systému na ext4
- březen 2011 - **3.0/3.1 Honeycomb**
  - Podpora tabletů
  - Nové uživatelské rozhraní
  - Upravený multi-tasking
  - Přístup ke službě Google eBooks
  - Nové Google Mapy (v.5)
  - Podpora videochatu (Google Talk)
- oznámeno/nevydáno - **Ice Cream Sandwich**

Další použitá literatura v této kapitole: [\[12\]](#), [\[13\]](#), [\[14\]](#)



Obrázek 2.1: Životní cyklus aktivity aplikace

## Kapitola 3

# Komunikace

### 3.1 Možnosti bezdrátového propojení mobilního přístroje a PC

Pokud nepočítáme dnes už spíše ustupující komunikaci přes infračervený port, tak nejrozšířenějšími způsoby bezdrátové komunikace mezi mobilním zařízením a počítačem jsou Bluetooth a WiFi.

#### 3.1.1 Bluetooth

Technologie Bluetooth vznikla jako produkt konsorcia SIG (Special Interest Group), kterou založily počátkem roku 1998 společnosti IBM, Intel, Lucent, Microsoft, Motorola, Nokia a Toshiba. V roce 1999 byla vydána specifikace Bluetooth 1.0. O rok později se objevil na trhu mimo jiné první mobilní telefon s funkcí Bluetooth. Do konsorcia SIG stále přibývají noví a noví členové, v roce 2010 přesáhl jejich počet 13 000.

Standard IEEE 802.15.1 definuje technologii Bluetooth, která pracuje v bezlicenčním pásmu na frekvenci 2,4 GHz. Konkrétní rozsah je 2,400 GHz - 2,4835 GHz a jednotlivé kanály jsou od sebe vzdáleny 1 MHz, z čehož vyplývá, že počet kanálů dostupných pro zařízení Bluetooth je 79. Za účelem potlačení interference s dalšími signály, používá metodu frekvenčních skoků (FHSS) s rychlostí 1600 skoků za sekundu (po 625 mikrosekundách). Frekvence se mění po každém přenosu a příjmu a tím je zajištěna větší kvalita spojení. Modulace signálu je prováděna pomocí Gaussovské modulace s frekvenčním klíčováním (GFSK).

Standard Bluetooth rozlišuje dva stavy:

- **Master**
- **Slave**

Do stavu master se dostává to zařízení, které je v daném prostoru aktivováno jako první. Ostatní, která se dostanou do jeho dosahu, získávají stav slave a jejich komunikace je řízena zařízením typu master. Ten pak rovněž řídí frekvenční skoky, sestavuje komunikaci mezi ostatními čipy a přiděluje komunikační kanály. Když zařízení ve stavu master zachytí aktivitu ve svém okolí, začne vysílat na prvních 16 frekvencích tzv. page (v případě, že zná adresu druhého zařízení), nebo inquiry (pokud druhé zařízení nezná). Jestliže nedostane odpověď, začne vysílat na dalších 16 frekvencích. Tento postup se v případě neúspěchu opakuje, avšak nejpozději do 2,56 sekund od začátku tohoto vysílání se obě zařízení domluví na určité frekvenci pro komunikaci. K jednomu zařízení typu master je možné připojit až 7 zařízení typu slave.

Způsob komunikace mezi zařízeními určují tzv. profily. Tyto profily zajišťují jejich slučitelnost. Aby dvě zařízení mohly mezi sebou komunikovat, musí obě podporovat stejný

profil. Součástí každého Bluetooth modulu by měl být manažer, který v systému vytváří tyto profily. Zde jsou vyjmenovány některé používané profily.<sup>1</sup>

- **A2DP** - přenos audia ve stereo kvalitě
- **BIP** - práce s obrázky (fotoaparát)
- **BPP** - komunikace s tiskárnou
- **DID** - identifikace právě připojeného zařízení
- **DUN** - připojení jiného zařízení (notebooku) k internetu přes mobilní telefon
- **FTP** - přístup k adresářům a složkám jiného zařízení
- **HFP** - připojení k hands-free sadě
- **HID** - připojení periferií pro ovládání počítače (myš, klávesnice)
- **HSP** - podobné jako HFP, ale méně možností ovládání telefonu
- **LAP** - sestavení počítačové sítě
- **OPP** - posílání souborů nebo vizitek
- **SAP** - přístup k SIM kartě telefonu (automobilové hands-free)
- **SDP** - dotaz jiného zařízení na podporu profilů
- **SPP** - emulace sériového portu, nutný pro fungování dalších profilů
- **VDP** - přenos videa v reálném čase

Pro definování oblasti používá Bluetooth daného zařízení takzvané profily. Ty zajišťují vzájemnou slučitelnost zařízení na nejvyšší softwarové úrovni. Aby zařízení mohla komunikovat, musí podporovat obě komunikující strany stejný profil.

Maximální dosah Bluetooth se liší podle třídy (Class), do které zařízení spadá. Tyto třídy jsou definovány podle maximálního výstupního výkonu. Rozdělení můžeme najít v tabulce 3.1.

Výkonová třída	Maximální výkon	Maximální dosah
Class 1	100 mW	100 m
Class 2	2,5 mW	50 m
Class 3	1 mW	10 m

Tabulka 3.1: Dosah Bluetooth podle výkonových tříd

Aktuálně je vyvinuto několik specifikací, některé z nich můžeme vidět v tabulce 3.2 níže, kde jsou porovnány jejich přenosové rychlosti. Většina dnes dostupných mobilních zařízení jako mobilní telefony, notebooky nebo televize obsahuje specifikaci Bluetooth 2.0 + EDR případně 2.1 + EDR. Některé novinky však dostávají mnohokrát rychlejší verzi 3.0.

Pro zabezpečení spojení používá technologie Bluetooth několik mechanismů. Na spojové vrstvě jsou to 4 kódy:

- Veřejná adresa o délce 48 bitů (jedinečná pro každé zařízení)

<sup>1</sup>[http://en.wikipedia.org/wiki/Bluetooth\\_profile](http://en.wikipedia.org/wiki/Bluetooth_profile)



- 2 tajné klíče o délce 128 bitů
- náhodné číslo o délce 128 bitů (různé pro každou novou operaci)

Dále se před prvním spojením musí obě zařízení spárovat. To se provádí tak, že na jednom ze zařízení vytvoří uživatel číslo PIN, které následně zadá i na druhém zařízení. K bezpečnosti přenosu přes Bluetooth také přispívají frekvenční skoky a malý dosah signálu, což velice komplikuje odposlech takové komunikace.

Specifikace	Přenosová rychlost
Bluetooth 1.2	1 Mbit/s
Bluetooth 2.0 + EDR	3 Mbit/s
Bluetooth 3.0 + HS	24 Mbit/s
Bluetooth 4.0	24 Mbit/s

Tabulka 3.2: Přenosové rychlosti standardů Bluetooth

### 3.1.2 Wi-Fi

V počítačových sítích existuje několik bezdrátových standardů 802.11 (přehled v tabulce 3.3), které se souhrnně označují zkratkou Wi-Fi. Tento název původně neměl znamenat nic, ale časem se z mu začalo přezdívat anglický spojením "wireless fidelity" (bezdrátová věrnost) což je analogií k Hi-Fi známé z audio systémů. Původním cílem Wi-Fi byla možnost vzájemného propojení přenosných zařízení a připojení do lokálních sítí LAN. Časem se tato technologie začala využívat převážně pro přístup do sítě internet. Dnes jsou Wi-Fi adaptéry samozřejmou součástí každého přenosného počítače i chytrého telefonu.

Wi-Fi technologie	Frekvence	Maximální propustnost pásma
802.11a	5 GHz	54 Mbit/s
802.11b	2,4 GHz	11 Mbit/s
802.11g	2,4 GHz	54 Mbit/s
802.11n	2,4 nebo 5 GHz	450 Mbit/s

Tabulka 3.3: Generace Wi-Fi technologií

Bezdrátovou síť lze vybudovat různými způsoby v závislosti na tom, k čemu budeme tuto síť využívat. Vždy hraje důležitou roli identifikátor sítě neboli SSID. Je to řetězec ASCII znaků o maximální délce 32, který je vyslán jako broadcast. Každé zařízení s Wi-Fi adaptérem má tak možnost zobrazit seznam dostupných sítí v daném prostoru a k některé se připojit. V praxi se nejčastěji setkáváme se dvěma druhy struktury sítě:

- **Ad-hoc** - spojení klientů mezi sebou nezávisle na zprostředkovali (peer-to-peer spojení), každý má v síti stejnou roli a všichni připojení klienti musí být v přímém rádiovém dosahu. Tento typ není tak častý, využívá se příležitostně například pro přenos souborů nebo při LAN-párty.
- **Infrastruktura** - je to nejčastější typ struktury sítě. Základem je jeden nebo více přístupových bodů, které vysílají své SSID. SSID mohou být i stejné a je pouze na klientovi, ke kterému přístupovému bodu se připojí. To se hodí například pokud chceme poskytnout připojení do jedné sítě na větším prostoru. Klient se pak při pohybu automaticky přepojuje z jednoho přístupového bodu na druhý.

Jedním z největších úskalí u bezdrátových sítí Wi-Fi je jejich bezpečnost. Zásadním problémem je skutečnost, že většina bezdrátových síťových prvků se prodává bez nastaveného zabezpečení, aby bylo možné je vybalit, zapojit a vše bylo okamžitě funkční. Většina neznalých uživatelů se nastavení zabezpečení dále nevěnuje a riziko napadení sítě je vysoké. Signál Wi-Fi se často šíří i mimo oblast, pro kterou je určen, proto je taková síť snadno napadnutelná. Díky směrovým anténám je možné napadnout bezdrátovou síť i z poměrně velké vzdálenosti. Existuje tedy několik druhů zabezpečení, primitivní a snadno prolomitelné i složité. Podrobně popsané jsou níže.

- **Skrytí SSID** - nejjednodušší zabezpečení. SSID se nepřenáší pomocí broadcastu. Při připojování klienta je však toto SSID přenášeno v otevřené podobě a je snadno odchytilitelné.
- **Filtrování MAC adres** - na přístupovém bodu sítě se nastaví seznam povolených MAC adres síťových zařízení, které se mohou připojit. Útočník se může připojit, pokud se vydává za jiného, už připojeného klienta.
- **WEP** - starší protokol, který symetricky šifruje komunikaci v síti pomocí statických WEP klíčů. Tyto klíče jsou ručně nastaveny na obou bezdrátových stanicích. Pro získání klíčů existují specializované programy, díky kterým je snadné se do sítě nabourat.
- **WPA** - protokol, který kvůli zpětné kompatibilitě šifruje komunikaci také pomocí WEP klíčů, ty jsou ale bezpečným způsobem dynamicky měněny. K tomuto účelu slouží doprovodný program zvaný suplikant.
- **WPA2** - novější protokol, který používá šifrování typu AES. Je bezpečnější než předchozí, nicméně více náročný na výpočetní výkon. Proto není možné použít toto zabezpečení na starších zařízeních.
- **802.1x** - podobně jako u WPA je zde využíván doprovodný program suplikant, kterému přístupový bod zprostředkuje komunikaci další stranou (nejčastěji RADIUS server), která ověří připojovaného uživatele. Pomocí tohoto protokolu lze odstranit některé nedostatky při používání WEP klíčů.

## 3.2 Volba technologie bezdrátové komunikace

Technologie Bluetooth má oproti Wi-Fi menší dosah, což není zase tak zásadní pro funkci této aplikace. Se spotřebou energie je na tom zase Bluetooth lépe. U mobilních zařízení s operačním systémem Android je samozřejmostí vybavení oběma technologiemi, nicméně ne každý počítač je vybaven Bluetooth modulem, zatímco síťovým adaptérem oplývá naprostá většina dnešních počítačů.

Na základě myšlenek zmíněných v této kapitole jsem se rozhodl, že pro bezdrátovou komunikaci mezi počítačem a mobilním zařízením použiji technologii Wi-Fi. Aplikace bude pracovat na klasickém principu server-klient a bude tedy složena ze dvou částí. Aplikace na počítači bude představovat serverovou část a aplikace v mobilním zařízení Android se bude chovat jako klient. Pro správný běh aplikace bude nutné, aby byla zařízení připojena do stejné lokální počítačové sítě LAN typu infrastruktura, protože OS Android nativně nepodporuje WiFi síť typu Ad-hoc. Síťová komunikace bude probíhat pomocí schránek neboli tzv. socketů.

## 3.3 Model ISO/OSI

ISO/OSI je referenční model, který vypracovala organizace ISO (International Standards Organization) jako hlavní část snahy o standardizaci počítačových sítí nazvané OSI (Open

Systems Interconnect). V roce 1984 jej přijala jako mezinárodní normu ISO 7498. Referenční model ISO/OSI se používá jako názorný příklad řešení komunikace v počítačových sítích pomocí vrstvého modelu, ve kterém jsou jednotlivé vrstvy nezávislé a snadno nahraditelné. Mnoho dnes běžných protokolů vychází právě ze struktury tohoto referenčního modelu. Jedním z nejpoužívanějších v praxi je TCP/IP<sup>2</sup> (Transmission Control Protocol/Internet Protocol). ISO/OSI model je tvořen sedmi vrstvami. Každá z nich vykonává skupinu jasně definovaných funkcí a protokolů potřebných pro komunikaci. Každá vrstva využívá pro svou činnost služby své sousední nižší vrstvy a své služby pak poskytuje sousední vrstvě vyšší. Jednotlivé vrstvy modelu ISO/OSI je možno vidět na obrázku 3.1.

Aplikační vrstva
Prezentační vrstva
Relační vrstva
Transportní vrstva
Síťová vrstva
Linková vrstva
Fyzická vrstva

Obrázek 3.1: Referenční model ISO/OSI

### 3.4 IP adresa

IP adresa je číslo, které jednoznačně identifikuje síťové rozhraní v počítačové síti, která používá IP protokol (internet protocol). V současné době je nejrozšířenější verze IPv4, která používá 32-bitové adresy zapsané dekadicky do 4 oktetů (osmicí bitů), například 192.168.0.1. Také z důvodu nedostatečného počtu takových IP adres se čím dál více prosazuje novější verze IPv6, která používá 128-bitové adresy. Tato adresa se zapisuje jako osm skupin po čtyřech hexadecimálních číslicích. Pro efektivnější využití počtu IPv4 adres byly tyto adresy rozděleny do tříd podle toho, jaká část adresy určuje síť a jaká určuje stanici v síti. Hranici mezi adresou sítě a počítačem v síti určuje maska podsítě. Jedná se o 32-bitovou hodnotu zapisovanou stejně jako IPv4 adresa. V binárním tvaru obsahuje jedničky tam, kde se v adrese nachází síť, a nuly tam, kde se v adrese nachází počítač. V tabulce 3.4 je k dispozici přehled tříd adres IPv4.

Třída	Začátek (bin)	1. bajt	Stand. maska	Bitů sítě	Bitů stanice
A	0	0-127	255.0.0.0	7	24
B	10	128-191	255.255.0.0	14	16
C	110	192-223	255.255.255.0	21	8
D	1110	224-239	Multicast		
E	1111	240-255	Vyhrazeno jako rezerva		

Tabulka 3.4: Rozdělení IPv4 adres do tříd

<sup>2</sup><http://cs.wikipedia.org/wiki/TCP/IP>

### 3.5 IP Broadcast

Veškerá data, která jsou mezi síťovými rozhraními přenášena, mají podobu IP datagramů neboli paketů. Jeden typ paketu se označuje jako broadcast. Pokud je takový paket v počítačové síti vyslán, tak jej přijmou všechna aktuálně připojená síťová rozhraní. Rozsah IP adres použitý v síti představuje IP adresu sítě, rozsah IP adres použitelný pro připojená zařízení a IP adresu broadcastu. Například v síti 192.168.0.0 s maskou 255.255.255.0 jsou k dispozici následující IP adresy:

- **IP adresa sítě** 192.168.0.0
- **IP adresy pro připojená zařízení** 192.168.0.1 – 192.168.0.254
- **IP adresa broadcastu** 192.168.0.255

Odchozí paket, který je adresován na IP adresu broadcastu, je v ethernetové síti odeslán v rámci, jehož cílová MAC adresa je nastavena na FF:FF:FF:FF:FF:FF. Takový rámec je přijat všemi síťovými kartami v dané podsíti.

### 3.6 Architektura server-klient

Standardní schéma síťové komunikace mezi dvěma procesy se nazývá model server-klient. Server i klient jsou aplikační procesy, které komunikují přes síťové rozhraní. Základní činností klienta je posílat požadavky na zpracování. Server čeká na příchozí požadavky, přijímá je, zpracovává a posílá odpověď. Příkladem může být WWW server (například Apache) a WWW klient (prohlížeč). Prohlížeč odešle požadavek na webovou stránku. Server požadavek přijme, vyhledá příslušnou stránku na lokálním serveru a pošle zpět klientovi odpověď. Důležité je, že zpracování probíhá výhradně na straně serveru. Klient pouze předává požadavek a případně zobrazuje odpověď. U modelu server-klient obvykle inicializuje komunikaci klient. Server často běží v nekonečné smyčce, kde čeká na dotaz a odpovídá.

Další použitá literatura v této kapitole: [10], [21], [7], [4], [9], [17], [18], [6], [3]

## Kapitola 4

# Zadání práce

Jak již bylo zmíněno v úvodu, cílem této práce je vytvořit aplikaci pro mobilní zařízení s operačním systémem Android, která umožní ovládání kurzoru myši na bezdrátově připojeném osobním počítači. Jak se bude aplikace chovat v určitých situacích a jaké bude mít funkce je popsáno v následujících řádcích.

### 4.1 Návrh funkcí aplikace

Aplikace spuštěná na počítači je jednoduchá a neobsahuje žádná složitá nastavení, pouze zadání hesla pro spárování a jeho potvrzení. Heslo je posíláno po síti šifrovaně, aby jej nebylo možné odchytit a zneužít. Po spuštění aplikace na mobilním zařízení je zkontrolováno, zda-li je zařízení připojeno do sítě Wi-Fi. Pokud není, aplikace navrhne možnost zapnutí Wi-Fi adaptéru. Stiskem tlačítka "Vyhledat PC" začne aplikace vyhledávat připojená zařízení v síti se zapnutou serverou aplikací. Výběrem některého z vyhledaných zařízení se aplikace nejprve dotáže na heslo pro spárování a následně se pokusí spojit se vzdálenou aplikací na serveru. V případě neúspěchu zahlásí, že se k vybranému zařízení nelze připojit. To může nastat například v případě neshody hesla. Pokud se spojení podaří, zobrazí se plocha a dvě tlačítka jako na touchpadu v dolní části displeje. Aplikace funguje jako myš připojeného počítače. Pro zadávání znaků slouží malá ikona v dolní části displeje mezi velkými tlačítky touchpadu, která po stisknutí vyvolá softwarovou klávesnici. Tu je možné zrušit stisknutím hardwarového tlačítka "Zpět". Pro aktivaci výběru, kdy je potřeba provést stisk levého tlačítka a pohyb myši zároveň, je nutné odháčknout tzv. checkbox, který se nachází, podobně jako ikona pro vyvolání klávesnice, ve spodní části displeje mezi tlačítky touchpadu. Vypnutí této funkce se provede opětovným stiskem checkboxu. Pozadí aplikace na mobilním zařízení má tmavou barvu pro nižší spotřebu energie.

### 4.2 Kompatibilita s operačními systémy

Serverová aplikace je schopna pracovat na běžně rozšířených verzích systému Linux. Klientská aplikace je kompatibilní se systémem Android 2.1 (API 7) nebo vyšším.

## Kapitola 5

# Uživatelské rozhraní a vzhled

### 5.1 Uživatelské rozhraní - software

Aplikace pro mobilní zařízení i PC vychází z jednotného vzhledu. Pozadí má černou barvu a v horní části okna se zobrazuje červené logo.

#### 5.1.1 Rozhraní v PC

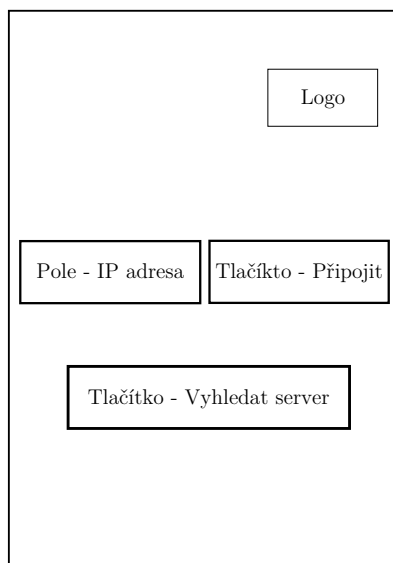
Program je tvořen jediným oknem, které obsahuje 4 tlačítka. První tlačítko slouží ke zjištění IP adresy lokálního síťového rozhraní. Aplikace vypisuje IP adresu přímo do hlavního okna. IP adresa k dispozici proto, aby ji uživatel nemusel ručně zjišťovat. Uživatel ji pak může snadno opsat do mobilního zařízení a připojit se k počítači. Druhé tlačítko nastavuje heslo, které uživatel vyplní. To je pak použito při spárování obou aplikací. Následuje tlačítko, které zobrazí v internetovém prohlížeči lokální soubor s nápovědou. Poslední tlačítko aplikaci ukončuje. V horní liště záměrně chybí tlačítka pro minimalizaci a zvětšení/zmenšení okna. Tlačítko s křížkem aplikaci neukončuje, ale minimalizuje ji do systémové lišty. Odtud může být aplikace znovu vyvolána do popředí nebo ukončena. Finální podoba aplikace je znázorněna na obrázku 5.1 níže.



Obrázek 5.1: Schéma aplikace v PC

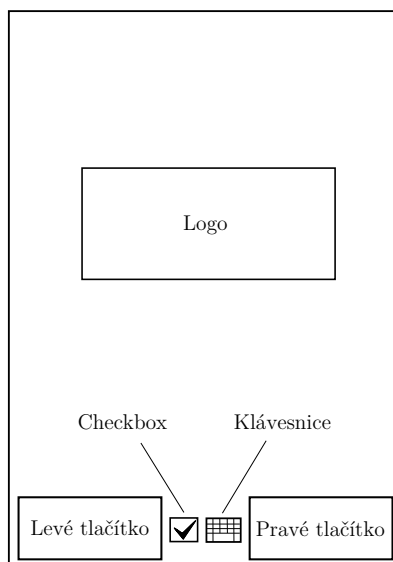
### 5.1.2 Rozhraní v mobilním přístroji

Po spuštění aplikace se objeví základní obrazovka, kde má uživatel 2 možnosti, jak se k danému počítači připojit. První z nich je pokusit se vyhledat počítač v síti, do které je mobilní přístroj připojen. Druhou možností je ruční opsání IP adresy z okna aplikace v PC. Vzhled základní obrazovky je možno vidět na obrázku 5.2 níže.



Obrázek 5.2: Schéma základní obrazovky aplikace v mobilním přístroji

Pokud připojení k počítači proběhne úspěšně, zobrazí se obrazovka touchpadu. Obrazovka je roztáhlá přes celý displej, takže zakrývá i statusovou lištu operačního systému Android. Uprostřed svítí, s velmi nízkou intenzitou, velké logo aplikace. V dolní části jsou vedle sebe umístěna 2 tlačítka, která nahrazují funkci tlačítek na reálném touchpadu. Mezi nimi je malá ikona, která při stisknutí vyvolá softwarovou klávesnici pro zadávání textu na připojeném počítači, a tzv. checkbox, který je určený pro simulaci dlouhého stisku levého tlačítka myši. Rozložení prvků na této obrazovce je k dispozici na obrázku 5.3 níže.



Obrázek 5.3: Schéma obrazovky touchpadu v mobilním přístroji

Rozložení tlačítek a natočení pozadí se řídí tím, je-li zařízení natočeno na výšku nebo na šířku. Při stisknutí kontextového tlačítka MENU je vyvoláno menu s možností zobrazit

nápovědu nebo ukončit aplikaci. Nápověda se zobrazuje ve vyskakovacím okně a ukončuje se stisknutím tlačítka OK.

## 5.2 Uživatelské rozhraní - hardware

Vzhledem k jednoduchému vzhledu aplikace nebude potřeba zvláště vysokého hardwarového výkonu mobilního zařízení. Důležité je, aby byla přítomná technologie WiFi. Aplikace musí být spolehlivě ovladatelná na zařízeních se dvěma nejčastěji používanými technologiemi dotykových displejů, rezistivní a kapacitní.

### 5.2.1 Rezistivní technologie dotykových displejů

Rezistivní neboli odporový panel dotykové obrazovky se skládá z několika vrstev, z nichž nejdůležitější jsou dvě tenké elektricky vodivé vrstvy, odděleny úzkou mezerou. Když objekt, stlačí místo na vnějším povrchu panelu, dvě kovové vrstvy se v tomto místě spojí a panel se pak chová jako pár napěťových děličů s připojenými výstupy. To vede ke změně v elektrický proud, který je registrován jako událost doteku a poslán k řadiči pro zpracování. Takový typ displeje je možné najít u levnějších přístrojů. Nevýhodou je, že při ovládání takového přístroje je nutné na displej více tláčit, aby byl pohyb zaznamenán. Naopak výhodou je, že při ovládání je možné použít nejen prst, ale jakýkoliv předmět, například stylus.

### 5.2.2 Kapacitní technologie dotykových displejů

Kapacitní dotykový panel se skládá z izolantu, jako je sklo potažené transparentním vodičem, kterým je nejčastěji indium tin oxid (ITO). Vzhledem k tomu, že lidské tělo je vodič, dotýkání se povrchu obrazovky má za následek narušování elektrostatického pole obrazovky, což je měřitelné jako změna v kapacitním odporu. K určení místa dotyku mohou být použité různé technologie. Lokace tohoto místa je pak poslána k řadiči pro zpracování. Výhodou je, že při ovládání není nutné silně tláčit na displej jako je tomu u rezistivní technologie, ale na druhou stranu je k ovládání nutné použít bříška prstů. Na nehty, stylus a jiné předměty dotykový displej nereaguje.

Další použítá literatura v této kapitole: [16]



## Kapitola 6

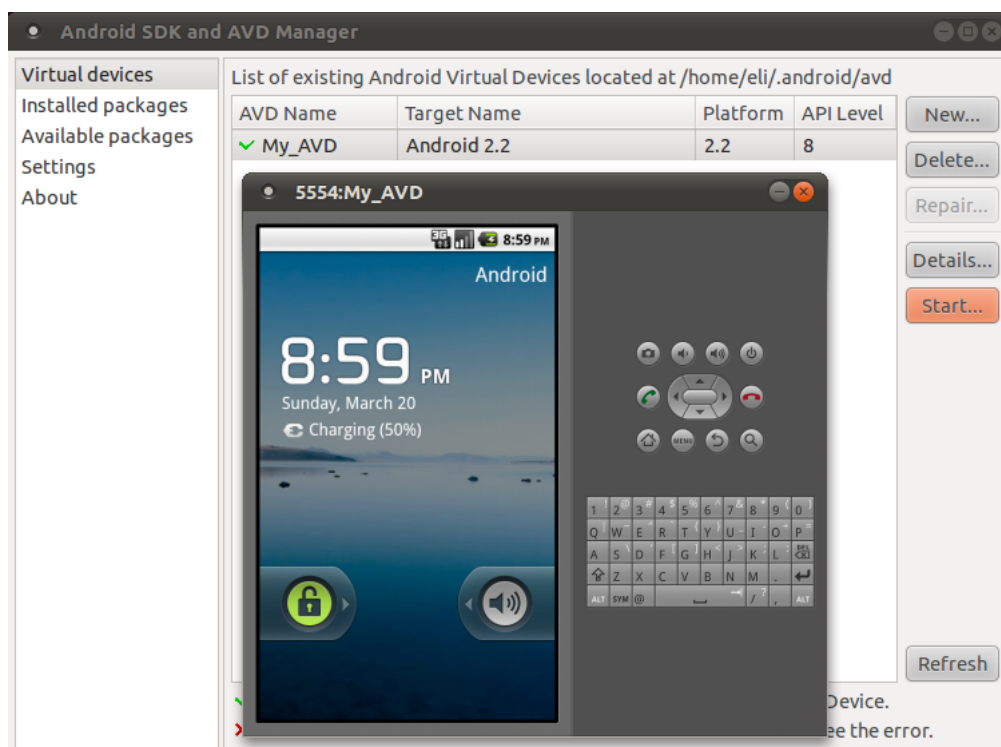
# Implementace

### 6.1 Vývojové nástroje

Pro tvorbu této aplikace byla zvolena následující vývojová prostředí a programovací jazyky.

#### 6.1.1 Android SDK

Android SDK je nástroj pro vývojáře aplikací pro Android. Umožňuje vytvářet a spouštět virtuální zařízení tzv. AVD<sup>1</sup>. Ty slouží jako emulátory, ve kterých je možné vyvíjenou aplikaci spouštět a testovat. Na výběr jsou různé verze OS Android a také různé vlastnosti virtuálních zařízení, kterými může vývojář simulovat reálné zařízení s OS Android. Okno této aplikace a spuštěné virtuální zařízení je možné vidět na obrázku 6.1.



Obrázek 6.1: Android SDK a virtuální zařízení

<sup>1</sup><http://developer.android.com/guide/developing/devices/index.html>

### 6.1.2 Editor programového kódu Eclipse IDE

Pro implementaci klientské aplikace, tedy aplikace pro zařízení se systémem Android, bylo vybráno integrované vývojové prostředí Eclipse IDE, které je dostupné na webu zdarma jako open source. V základu podporuje toto prostředí pouze standardní jazyk Java, jeho základní výhodou je však možnost rozšíření pomocí zásuvných modulů neboli pluginů. K vývoji aplikací pro OS Android je k dispozici zásuvný modul Android ADT<sup>2</sup>. Tento plugin umožňuje vývojářům zakládat nové Android projekty, spouštět je ve virtuálním nebo připojeném reálném Android zařízení přímo z okna aplikace Eclipse IDE. Mimo jiné také podporuje export podepsaných (i nepodepsaných) aplikací do souboru ".apk", který je pak možné dále distribuovat například do oficiálního Android Marketu. Vývoj aplikací pro Android v prostředí Eclipse IDE se zásuvným modulem Android ADT je tudíž snazší a efektivnější než jakákoliv jiná varianta.



Obrázek 6.2: Logo - Eclipse IDE

### 6.1.3 Editor programového kódu Qt Creator IDE

Qt Creator IDE je jedno z nejpopulárnějších multi-platformních integrovaných vývojových prostředí. Původní Qt toolkit byl vytvořen v roce 1999 společností Trolltech. Od roku 2008 jej však vyvíjí společnost Nokia. Toto prostředí poskytuje mimo rozsáhlou knihovnu programovacího jazyka C++ i editor pro tvorbu kódu v C++ a JavaScriptu, překladač, debugger a také vestavěný nástroj pro tvorbu grafického uživatelského rozhraní. Díky zmíněným výhodám a bezplatnému užívání bylo toto vývojové prostředí uznáno za vhodné pro implementaci serverové aplikace.



Obrázek 6.3: Logo - QT Creator

### 6.1.4 Použité programovací jazyky

Jak již vyplývá z předchozích odstavců, při implementaci aplikace pro tuto bakalářskou práci byly použity objektově orientované programovací jazyky Java a C++.

#### Java

Programovací jazyk Java<sup>3</sup>, který představila 23. května 1995 společnost Sun Microsystems, je dnes jedním z nejpoužívanějších programovacích jazyků na světě. Podle některých zdrojů

<sup>2</sup><http://developer.android.com/sdk/eclipse-adt.html>

<sup>3</sup><http://www.java.com/en/>

je to dokonce ten nejpoužívanější. Díky své přenositelnosti je používán pro programy, které pracují na nejrozličnějších systémech počínaje čipovými kartami, přes mobilní telefony a různá zabudovaná zařízení, aplikace pro desktopové počítače až po rozsáhlé distribuované systémy pracující na velkém množství spolupracujících počítačů rozprostřené po celém světě. V roce 2007 uvolnila společnost Sun zdrojové kódy jazyka Java a tak je od té doby dále vyvíjena jako open source. Hlavní nevýhodou aplikací napsaných v tomto programovacím jazyce je znatelně pomalejší start oproti aplikacím napsaných v jiných jazycích, které provádějí tzv. statickou kompilaci (například C++). Další nevýhodou projevující se hlavně u jednodušších programů je větší paměťová náročnost při běhu, která je způsobena nutností mít v paměti celé běhové prostředí. V návrhu jazyka Java je vidět snaha znemožnit programátorovi psát problematické konstrukce známé především z jazyka C. Jedním ze systémů používajících pro vývoj aplikací syntax programovacího jazyka Java je právě Android.

## C++

Za vývojem programovacího jazyka C++ stojí především člověk jménem Bjarne Stroustrup<sup>4</sup>. C++ podporuje několik programovacích stylů jako je procedurální programování, objektově orientované programování a generické programování. Jazyk C++ tudíž není jazykem čistě objektovým. V současné době patří C++ podobně jako Java mezi nejrozšířenější programovací jazyky na světě. Mimo jiné je tento jazyk hlavním stavebním kamenem při tvorbě aplikací v základním vývojovém prostředí Qt. Přestože vznikl již v roce 1983, první oficiální norma C++ byla přijata až v roce 1998. Jazyk C, ze kterého C++ vychází, je až na několik jasně definovaných výjimek podmnožinou C++, nicméně považovat C++ za pouhé rozšíření jazyka C by bylo chybou, protože není s jazykem C zcela kompatibilní. Důkazem je, že ne všechny programy napsané v jazyce C lze přeložit překladači pro C++.

## 6.2 Aplikace v mobilním přístroji

Celá aplikace včetně uživatelského rozhraní byla vyvíjena v prostředí Eclipse IDE. Samotný kód je napsán v jazyce Java.

Každá Android aplikace je tvořena jednou nebo více tzv. aktivit, kdy každá z nich má své uživatelské rozhraní. Při přechodu mezi aktivitami je potřeba případně přenést i data, která chceme použít, protože jedna aktivita nemá přístup k proměnným aktivitu druhé. V této klientské aplikaci se vyskytují aktivity tři a každá z nich zastává určitou funkci.

Hlavní aktivita se spouští rovnou při startu aplikace. Během inicializace se pokusí načíst IP adresu počítače ze souboru, který předtím tato aplikace v mobilním zařízení vytvořila při posledním úspěšném připojení. Tuto IP adresu potom předvyplní do příslušného políčka, aby měl uživatel méně práce při opakovaném připojení ke stejnému počítači. Možnosti jak dále pokračovat jsou dvě. První z nich je pokus o přímé připojení k počítači. Aplikace si nejprve vyžádá heslo pro spárování, které zakóduje jednosměrnou šifrou MD5<sup>5</sup> a následně spustí novou aktivitu určenou pro komunikaci se vzdáleným počítačem, které toto heslo společně se zadanou IP adresou odešle. Druhou možností je automatické vyhledání počítače v síti se zapnutou serverovou aplikací. Pokud uživatel zvolí tuto možnost, je spuštěna aktivita, která se pokusí takové počítače v síti vyhledat. U obou možností navíc aplikace kontroluje dostupnost WiFi rozhraní. Když není zapnuto, aplikace nabídne možnost přejít do systémového nastavení a zapnout jej.

Aktivita pro vyhledání počítače v síti musí nejprve zjistit broadcast adresu sítě, do níž je mobilní zařízení připojeno. To dělá na základě informací získaných ze systémové služby, která ukládá informace o aktuálním připojení WiFi rozhraní. Broadcast adresa je spočítána z přidělené IP adresy a masky sítě. Na tuto adresu je odeslán paket, který dostanou všechna

<sup>4</sup><http://www2.research.att.com/~bs/homepage.html>

<sup>5</sup>[http://cs.wikipedia.org/wiki/Message-Digest\\_algorithm](http://cs.wikipedia.org/wiki/Message-Digest_algorithm)

připojená zařízení. Pokud aplikace obdrží od některého z nich odpověď, znamená to, že na tomto zařízení je spuštěna serverová aplikace a je připravena pro připojení klienta. Součástí této odpovědi je IP adresa vzdáleného počítače, kterou aplikace zobrazí uživateli. Když existuje dostupných zařízení více, aplikace zobrazí všechny a uživatel si může vybrat, ke kterému z nich se chce připojit. Jakmile uživatel některou IP adresu zvolí, aplikace si vyžádá heslo pro spárování a spustí novou aktivitu pro komunikaci se vzdáleným počítačem. Zároveň je vybraná IP adresa a zadané heslo odesláno této nové aktivitě.

Aktivita sloužící pro komunikaci se vzdáleným počítačem obdrží od předchozí aktivitiy IP adresu, ke které se má připojit a zakódované heslo, které použije pro autorizaci. Při úspěšném připojení k serverové aplikaci pomocí TCP<sup>6</sup> soketu uloží použitou IP adresu do souboru a na server odešle heslo pro ověření. V případě, že heslo souhlasí, je aplikace připravena přijímat příkazy od uživatele. Součástí uživatelského rozhraní je plocha touchpadu, levé a pravé tlačítko myši, checkbox pro simulaci dlouhého stisku levého tlačítka myši a ikona pro vyvolání softwarové klávesnice. Na každou akci, kterou uživatel na této obrazovce provede, reaguje aplikace tím, že ji okamžitě odešle na vzdálený server pro vykonání. Stiskne-li uživatel například levé tlačítko myši, aplikace tuto akci zaznamená a na server odešle paket s obsahem "LMB#". O něco složitější je zaznamenání dotyku na ploše touchpadu, který se má chovat jako stisk levého tlačítka myši. Zde je nutné ověřit dobu trvání dotyku a rozdíl polohy jeho začátku a konce. Pokud je dotyk dostatečně krátký a začíná a končí na relativně stejném místě, je vyhodnocen jako stisk levého tlačítka myši. Při pohybu prstem po ploše touchpadu se zaznamenávají souřadnice. Aplikace spočítá rozdíl mezi posledními a aktuálními souřadnicemi a ten odešle na vzdálený počítač, který podle velikosti a znamének těchto čísel určí, jak se kurzor myši na obrazovce počítače pohne. Stisk šipek a klávesy backspace je, stejně jako u tlačítek myši, zaznamenán a odeslán serverové aplikaci. Komplikovanější je případ ostatních kláves na virtuální klávesnici. Pro ně je v rozhraní této aktivity vytvořeno skryté textové pole, do kterého je text z klávesnice zadáván. Aplikace při změně textu v textovém poli porovná původní a nový text, který je o jeden znak delší. Tento poslední znak následně odešle vzdálenému počítači.

Při ukončení aplikace je odeslána zpráva vzdálenému počítači, která jej o této skutečnosti informuje. Stejnou zprávu zasílá aplikace také v případě, že uživatel otočením mobilního zařízení změní rozložení prvků na displeji z typu portrait na typ landscape nebo naopak. Po této rotaci se však aplikace pokusí znovu sama připojit.

## 6.3 Aplikace v PC

Grafické uživatelské rozhraní bylo navrženo v programu Qt Designer, což je jedna ze součástí prostředí Qt Creator. Samotný kód celé aplikace byl vytvořen v jazyku C++ v prostředí Qt Creator IDE.

Hlavní vlákno aplikace nejprve vytvoří a inicializuje celé uživatelské rozhraní, alokuje místo ve sdíleném paměťovém segmentu pro heslo, které je určeno ke spárování obou aplikací, a pokusí se zjistit lokální IP adresu. To se provádí pomocí třídy QNetworkInterface. Pokud uživatel zadá heslo pro spárování, tak je nejprve zakódováno pomocí jednosměrné šifry MD5 a poté uloženo do sdíleného paměťového segmentu. Heslo musí být dostupné hlavnímu procesu, který jej ukládá po zadání uživatelem, a taky jednomu z dalších procesů, který se stará o síťovou komunikaci. Procesy mezi sebou komunikují pomocí rour.

Další objemnou částí aplikace je síťová komunikace, která funguje pomocí soketů neboli tzv. schránek. Nejprve se ověří dostupnost komunikačního síťového portu vytvořením TCP soketu a následně se pokusí na něj navázat. Další soket, tentokrát typu UDP<sup>7</sup>, je vytvořen pro příjem broadcast paketu. Proces se dostane do fáze, kdy vyčkává pomocí příkazu

<sup>6</sup>[http://cs.wikipedia.org/wiki/Transmission\\_Control\\_Protocol](http://cs.wikipedia.org/wiki/Transmission_Control_Protocol)

<sup>7</sup>[http://cs.wikipedia.org/wiki/User\\_Datagram\\_Protocol](http://cs.wikipedia.org/wiki/User_Datagram_Protocol)

"select", dokud nezachytí broadcast paket, požadavek na přímé spojení od klienta, nebo požadavek na ukončení práce od rodičovského procesu. Broadcast paket vysílá klient do sítě v době, kdy vyhledává dostupný server. Na tento paket server klientovi odpoví, a ten tak získá hledanou IP adresu. Po připojení klienta je nejprve ověřeno párovací heslo. To server přijme již v zašifrované podobě a pouze jej porovná se zápisem ve sdílené paměti. Klientovi pak odešle zprávu o správnosti zadaného hesla. Pokud ověření hesla proběhlo v pořádku, přechází proces do nekonečného cyklu, na jehož začátku se nachází další příkaz "select". Tam opět vyčkává na požadavek ukončení od rodičovského procesu nebo na klientský požadavek pro provedení určité akce. Přijatý požadavek od klienta je zpracován a požadovaná akce je na počítači provedena. Tento cyklus proces opouští, pokud se vzdálený klient odpojí.

Simulování akcí souvisejících s myší a klávesnicí probíhá pomocí funkce `XOpenDisplay`, která umožňuje připojení k X serveru. Funkce, která přímo vykonává pohyb kurzoru myši, se jmenuje `XWarpPointer`. Stisk tlačítek zase zajišťuje funkce `XTestFakeButtonEvent`. Simulování stisknutých kláves na klávesnici je o něco složitější. Nejprve je nutné znak, šipku, enter nebo backspace získaný od klienta převést na konstantu, která jej reprezentuje v knihovně `X11/keysym.h`. Například pro písmeno malé "a" existuje konstanta `"XK_a"`. U některých znaků je nutné dávat pozor na rozdíl mezi tím, jakým způsobem se píše na anglické a české klávesnici. Pro tyto znaky je potřeba zapnout jeden z modifikátorů ("Shift" nebo "Alt Gr") a přiřadit jim konstantu odpovídající české ekvivalentní klávese. Jako příklad uvedu zavináč ("@"), u kterého se musí jako modifikátor zvolit "Alt Gr" a klávesa, která se bude posílat, odpovídá na české klávesnici písmenu "v". Na základě konstanty odpovídající požadované klávese se vytvoří událost `XKeyEvent`, která se odešle příkazem `XSendEvent` do X serveru k vykonání.

Další použitá literatura v této kapitole: [20], [15], [8]

# Kapitola 7

## Testování

### 7.1 Metody testování

Stejně jako návrh a implementace počítačové aplikace je velmi důležité i její testování. Je dobré zjistit, jak se bude aplikace chovat na různých zařízeních, v různých situacích, a především jaké pocity vyvolá u svých budoucích uživatelů. Na základě zjištěných nedostatků a připomínek ze strany testujících uživatelů je vhodné aplikaci optimalizovat a doladit do přijatelné podoby.

Testování probíhalo ve třech fázích. První z nich se odehrávala v průběhu implementace po každé dokončené dílčí části. Serverová aplikace byla během vývoje spouštěna na linuxové distribuci operačního systému Ubuntu 10.10 Maverick Meerkat, zatímco klientská aplikace byla testována na mobilním telefonu HTC Desire HD s operačním systémem Android ve verzi 2.2 Froyo.

Druhá fáze testování následovala po dokončení implementace. Pro vytvoření počítačové sítě byly postupně použity tři WiFi routery od různých výrobců. Snažil jsem se také zjistit, jak se bude aplikace chovat v počítačové síti s běžným provozem v porovnání se sítí bez datových přenosů. Serverová aplikace byla po celou dobu této fáze spuštěna na operačním systému Ubuntu 11.04 Natty Narwhal, což je aktuálně nejnovější verze. Klientskou aplikaci jsem nainstaloval a testoval na virtuálním zařízení Android ve verzi 2.2 Froyo a poté na mobilních zařízeních z různých cenových kategorií. Rozdíly mezi nimi byly pochopitelně především ve výkonu a velikosti displeje, ale také v použité technologii dotykové vrstvy. Tento vzorek zařízení zahrnoval dnes nejběžnější verze systému Android. Zde je jejich výčet:

- Huawei U8100 - Android 2.1 Eclair
- Prestigio MultiPad PMP3084B - Android 2.1 Eclair
- HTC Wildfire - Android 2.2 Froyo
- HTC Desire HD - Android 2.2 Froyo
- HTC Desire HD - Android 2.3 Gingerbread

Po provedených změnách v aplikaci, které vycházely ze získaných poznatků z druhé fáze, jsem nakonec nechal funkčnost aplikace otestovat na vzorku osmi lidí s různou technickou zdatností. Zkoušeli obě dostupné možnosti připojení ke vzdálenému počítači, jejich úspěšnost, funkci touchpadu a také zadávání znaků pomocí virtuální klávesnice. Kromě testování jsem se zajímal také o jejich osobní názory na aplikaci jako celek, které jsem společně s ostatními připomínkami zpracoval a použil při finálních opravách chyb v implementaci.

## 7.2 Výsledky testování

Díky testování jsem v implementaci opravil značné množství chyb a zjistil mnoho užitečných informací, o kterých se rozepíšu v následujících řádcích. Úpravy kódu během první fáze, tedy v průběhu vývoje, zde rozebírat nebudu, raději se zaměřím na podstatné změny a zjištění, které přinesla druhá a třetí fáze testování.

Při testování serverové aplikace na novější verzi systému Ubuntu nastal pouze jeden problém, přestal fungovat algoritmus pro získání IP adresy lokálního počítače. Pro vyřešení jsem nakonec využil vestavěných funkcí knihovny Qt.

Jak jsem předpokládal před testováním, typ a výrobce WiFi routerů neměly na fungování aplikace žádný vliv. Co už ovšem nějakou roli hrálo, byla zatíženost sítě. V síti, kde byl zapojený pouze testovaný server a klient, fungovala aplikace podle předpokladů výtečně. Pokud se ale na síti vyskytl provoz v podobě objemného datového přenosu, začalo se objevovat nepravidelné zpoždění mezi provedením příkazu na mobilním zařízení a vykonáním příslušné akce na počítači. Nicméně ovladatelnost aplikace byla stále přijatelná, takže jsem tento problém dále nezkoumal.

Během testování klientské aplikace v Android emulátoru jsem zjistil, že virtuální zařízení Android neumí emulovat WiFi rozhraní. Přesto se funkčnost aplikace dala alespoň z části otestovat díky tomu, že emulátor je virtuálně připojen do stejné sítě jako počítač, na kterém je spuštěn. Jedinou funkcí, která zde nefungovala, bylo automatické vyhledání počítače v síti. To je způsobeno tím, že aplikace získává broadcast adresu sítě z parametrů aktuálně připojeného WiFi rozhraní, které, jak jsem již zmiňoval, zde chybí. Ale vzhledem k tomu, že reálná zařízení takový problém neměla, tak jsem se tím více nezabýval.

Určité změny byly nutné po zjištění, že u mobilních zařízení s menším displejem a rozlišením se některé grafické prvky na obrazovce překrývají. Proto jsem hlavní obrazovku aplikace umístil do boxu, který při přesáhnutí rozlišení displeje grafickými prvky aplikace umožňuje posouvání aktuálně viditelného výřezu. Všechny prvky pak sice nejdou vidět najednou, ale zůstanou vždy čitelné a nepřekrývají se. Aplikace fungovala správně na zařízeních s odpořovými i kapacitními dotykovými displeji.

Jak je již zmíněno výše, v poslední fázi mě zajímaly ohlasy testujících uživatelů na hotový produkt této práce. Komentáře byly převážně pozitivní až na jednu nejčastěji se opakující výtku. Tou byla nekompatibilita serverové aplikace s operačním systémem Microsoft Windows. Na základě toho jsem se rozhodl zmíněný nedostatek uvést jako možnost budoucího rozšíření této aplikace. Potencionální uživatelé naopak ocenili především možnost automatického vyhledání počítače v síti a také přehlednost grafického uživatelského rozhraní u obou aplikací.

# Kapitola 8

## Závěr

### 8.1 Postup práce

Cílem této práce bylo vytvořit bezdrátový touchpad v podobě mobilního zařízení s dotykovým displejem. Práce na tomto projektu začala návrhem aplikace jako celku, jak vlastně bude takový program fungovat. Uvědomil jsem si, že pro vývoj budu potřebovat znalosti z několika různých odvětví informačních technologií a programování. Bylo třeba vytvořit dvě aplikace, přičemž každá z nich musí mít své grafické uživatelské rozhraní. Aplikace pracují na odlišných platformách a navíc spolu komunikují přes počítačovou síť. Když byl návrh hotový, začal jsem pracovat na tvorbě uživatelských rozhraní, které jsem následnou implementací přivedl do funkční podoby.

### 8.2 Dosažené výsledky

Základním požadavkem na aplikaci byla uživatelská přívětivost. V tomto ohledu jsem musel dbát především na přehlednost a přidání funkcí, které udělají ovládání aplikace rychlejší, jednodušší a příjemnější. Během návrhu aplikace jsem se rozhodl k funkci bezdrátového touchpadu přidat také možnost zadávání znaků z virtuální klávesnice mobilního zařízení do počítače. Dále jsem kladl důraz na to, aby měl uživatel snazší možnost připojení ke vzdálenému počítači, než je přímé zadání jeho IP adresy. Proto vznikla funkce automatického vyhledání počítače v síti. Určité úsilí jsem věnoval také bezpečnosti, tu by mělo zajistit spárování obou aplikací pomocí hesla, které navíc využívá jednosměrnou šifru. Jedním z problémů dnešních chytrých telefonů s velkými displeji a výkonným hardwarem je vysoká spotřeba elektrické energie. Z toho důvodu má pozadí aplikace černou barvu, která by měla spotřebu energie mírně snížit.

Díky bezdrátovým počítačovým sítím a aplikaci RemotePad, která vznikla jako produkt této bakalářské práce, může uživatel pomocí svého Android zařízení virtuálně ovládat myš a klávesnici počítače z libovolné vzdálenosti. Jediným omezením je dosah signálu WiFi. Vzhledem k tomu, že při síťové komunikaci dochází ke zpoždění, není tato aplikace určena pro každodenní dlouhodobou práci s počítačem. V tomto ohledu se standardní myši s klávesnicí rovnat nemůže. V určitých situacích však může být ovládání počítače díky této aplikaci pohodlnější.

Aplikace nemá problém s různými typy dotykových displejů a funguje spolehlivě v systému Android ve verzi 2.2 Froyo a 2.3 Gingerbread. Ve verzi 2.1 Eclair si není schopná poradit s funkcí automatického vyhledání počítače v síti, která způsobuje pád celé aplikace. Zařízení se staršími verzemi systému Android jsem při vývoji k dispozici neměl. Výsledný produkt splňuje zadání práce obstojně, dokonce ho v určitém ohledu rozšiřuje. Za toto rozšíření lze považovat přidání funkce klávesnice. Ve srovnání s podobnými dostupnými aplikacemi je funkčnost mnou vytvořené aplikace srovnatelná. Až na funkci automatického



vyhledání počítače v síti, kterou ostatní aplikace buď nepodporují, nebo ji autor zpřístupní jen v placené verzi.

### 8.3 Možnosti využití

Uživatel má možnost například pohodlně ovládat vzdálený počítač připojený k externí velkoplošné obrazovce nebo projektoru při sledování filmů, prohlížení fotografií nebo internetových stránek, aniž by se musel k počítači přiblížit. Další možnost využití může být také při prezentování. Když se prezentující pohybuje ve větší vzdálenosti od počítače připojeného k projekčnímu nebo zobrazovacímu zařízení, může pomocí této aplikace posouvat jednotlivé stránky prezentace, ovládat pohyb kurzoru myši, případně zadávat znaky z klávesnice.

### 8.4 Možnosti budoucího rozšíření

Aktuálně není možné z virtuální klávesnice mobilního zařízení posílat na ovládaný počítač znaky s diakritikou. Jednou z možností rozšíření je tedy implementovat podporu diakritiky. Vhodná by byla také výše zmíněná kompatibilita s nejrozšířenějším operačním systémem mezi běžnými uživateli, Microsoft Windows. Problémem je, že zasílání událostí, které simulují stisk klávesy nebo pohyb kurzoru myši, je nutné pro každý operační systém implementovat jiným způsobem, přestože byla serverová aplikace vyvíjena v multi-platformním vývojovém prostředí Qt Creator IDE. Dalším rozšířením by mohla být podpora bezdrátové technologie Bluetooth pro propojení mobilního zařízení a počítače. Uživatel by tak měl na výběr mezi právě zmíněnou technologií Bluetooth a původně implementovanou WiFi.

# Literatura

- [1] AppleInsider: Canals: iPhone outsold all Windows Mobile phones in Q2 2009. 2009-08-21.
- [2] Arthur, C.: Eric Schmidt's dog whistle to mobile developers: abandon Windows Phone [online]. <http://www.guardian.co.uk/technology/2010/jun/25/android-schmidt-mobile-platform>, 2010-06-25 [cit. 2011-01-10].
- [3] Matoušek, P.: Studijní opora: Kapitola 2 - Programování sítí TCP/IP. 2009.
- [4] Nigmatullin, T.: Bakalářská práce: Realizace a zabezpečení domácí ad hoc Wi-fi sítě. 2010.
- [5] Savov, V.: Andy Rubin: over 300,000 Android phones activated daily [online]. <http://www.engadget.com/2010/12/09/andy-rubin-over-300-000-android-phones-activated-daily/>, 2010-12-09 [cit. 2011-01-10].
- [6] WWW stránky: Broadcast [online]. <http://cs.wikipedia.org/wiki/Broadcast>, 2010-06-23 [cit. 2011-05-16].
- [7] WWW stránky: Bluetooth 4.0 gets 100m reach and encryption [online]. <http://www.nordichardware.com/news/83-science/40506-bluetooth-40-gets-100m-reach-and-encryption.html>, 2010-07-08 [cit. 2011-01-13].
- [8] WWW stránky: C++ [online]. <http://cs.wikipedia.org/wiki/C%2B%2B>, 2010-12-06 [cit. 2011-05-09].
- [9] WWW stránky: Wi-Fi [online]. <http://cs.wikipedia.org/wiki/Wi-Fi>, 2010-12-11 [cit. 2011-01-15].
- [10] WWW stránky: The Official Bluetooth Technology Info Site [online]. <http://www.bluetooth.com/English/Pages/Default.aspx>, 2010 [cit. 2011-01-13].
- [11] WWW stránky: Platform Versions:Current distribution [online]. <http://developer.android.com/resources/dashboard/platform-versions.html>, 2011-01-04 [cit. 2011-01-12].
- [12] WWW stránky: What is Android? [online]. <http://developer.android.com/guide/basics/what-is-android.html>, 2011-01-05 [cit. 2011-01-09].
- [13] WWW stránky: Android (operating system) [online]. [http://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)](http://en.wikipedia.org/wiki/Android_(operating_system)), 2011-01-09 [cit. 2011-01-09].
- [14] WWW stránky: Comparison of Android devices [online]. [http://en.wikipedia.org/wiki/List\\_of\\_Android\\_devices](http://en.wikipedia.org/wiki/List_of_Android_devices), 2011-01-11 [cit. 2011-01-11].

- [15] WWW stránky: Java [online].  
[http://cs.wikipedia.org/wiki/Java\\_\(programovací\\_jazyk\)](http://cs.wikipedia.org/wiki/Java_(programovací_jazyk)), 2011-03-13 [cit. 2011-05-09].
- [16] WWW stránky: Dotyková obrazovka [online].  
[http://cs.wikipedia.org/wiki/Dotyková\\_obrazovka](http://cs.wikipedia.org/wiki/Dotyková_obrazovka), 2011-03-15 [cit. 2011-03-17].
- [17] WWW stránky: Model ISO/OSI [online].  
[http://cs.wikipedia.org/wiki/Referenční\\_model\\_ISO/OSI](http://cs.wikipedia.org/wiki/Referenční_model_ISO/OSI), 2011-04-30 [cit. 2011-05-16].
- [18] WWW stránky: IP adresa [online]. [http://cs.wikipedia.org/wiki/IP\\_adresa](http://cs.wikipedia.org/wiki/IP_adresa), 2011-05-12 [cit. 2011-05-16].
- [19] WWW stránky: AppStats [online]. <http://www.androlib.com/appstats.aspx>, 2011 [cit. 2011-01-12].
- [20] WWW stránky: QT Creator IDE [online].  
<http://qt.nokia.com/products/developer-tools>, 2011 [cit. 2011-05-09].
- [21] Čánský, J.: Semestrální práce: Mobilní komunikace Bluetooth. 2006.

# Příloha A

## Obsah CD

- Android emulátor
- Návod na instalaci ve formátu txt
- Spustitelné aplikace
- Technická zpráva ve formátu pdf
- Uživatelská dokumentace ve formátu pdf
- Zdrojové kódy

## Příloha B

# Uživatelská dokumentace

### B.1 O aplikaci

Pomocí aplikace RemotePad je možné bezdrátově ovládat počítač pomocí dotykového displeje zařízení s operačním systémem Android. Aplikace zastává dvě funkce a to funkci myši/touchpadu a funkci klávesnice.

### B.2 Požadavky

Aplikace zastávající serverovou část (ovládaný počítač) je kompatibilní s operačním systémem Linux s českou lokalizací a klávesnicí a rozhraním X11. Aplikace zastávající klientskou část je kompatibilní s operačním systémem Android ve verzi 2.1 (API 7) nebo vyšší. Ovládaný počítač musí být připojen do stejné počítačové sítě jako klient, který se do této sítě připojuje pomocí bezdrátové technologie WiFi. Kvůli chybějícímu rozhraní WiFi, nelze v Android emulátoru využít funkci "automatické vyhledání počítače v síti". Aplikace komunikuje na síťových portech 8110 a 8111, je tedy nutné, aby byly tyto porty dostupné.

### B.3 Instalace

Obě aplikace se nachází na DVD přiloženém k technické zprávě bakalářské práce. Serverovou aplikaci je potřeba pouze zkopírovat na lokální disk systému Linux. Klientská aplikace je ve formátu "apk" a je nutné ji do zařízení se systémem Android zkopírovat a poté nainstalovat. Detailní popis instalace aplikací je možné najít v souboru "README.txt", který se nachází v kořenovém adresáři na DVD přiloženém k technické zprávě.

### B.4 Ovládání

#### B.4.1 Server - Linux

Když se po spuštění programu ve statusové liště objeví text "OK - port is available", a v oblasti "Local IP address" je zobrazena lokální IP adresa, tak je vše připraveno k použití a je možné se zkusit připojit k této serverové aplikaci z aplikace RemotePad na zařízení s operačním systémem Android. Pokud se ve statusové liště však objeví text "Bind error, restart please", tak je nutné aplikaci vypnout a zkusit zapnout znovu. Tento problém je způsoben zablokovaným TCP portem číslo 8111, který potřebuje určitou dobu pro uvolnění po předchozím používání.

Jestliže je potřeba znát IP adresu tohoto počítače, je ji možno získat z oblasti označené "Local IP address". V případě, že je aplikace zapnuta a počítač je přepojován do jiné sítě, je možno stisknout tlačítko "Refresh", které obnoví zobrazenou lokální IP adresu na aktuální.

Z bezpečnostních důvodů je doporučeno nastavit heslo. To je možné docílit vepsáním vymyšleného hesla do rámečku "Password" a následným stiskem tlačítka "Save Password".

Program je možné minimalizovat do systémové lišty několika způsoby. Kliknutím na křížek v horním rohu okna aplikace, dvojklikem na ikonu aplikace v systémové liště nebo kliknutím pravým tlačítkem myši na ikonu aplikace v systémové liště a následným výběrem v menu.

Maximalizovat program je možné dvojklikem na ikonu aplikace v systémové liště nebo kliknutím pravým tlačítkem myši na ikonu aplikace v systémové liště a následným výběrem v menu.

Nápověda se zobrazí ve výchozím webovém prohlížeči po stisku tlačítka "Help" a program skončí stiskem tlačítka "Exit".

#### **B.4.2 Klient - Android**

Před tím, než bude možné se připojit k ovládanému počítači, je potřeba ujistit se, zda-li je serverová aplikace RemotePad na tomto počítači spuštěná.

Pro připojení k ovládanému počítači jsou dvě možnosti. Je možné napsat IP adresu vzdáleného počítače ručně a kliknout na tlačítko "Connect" nebo kliknout na "Find server" a aplikace se pokusí vyhledat server v síti automaticky.

Pro úspěšné připojení je nutné znát heslo, které bylo zadáno v serverové aplikaci.

Jakmile je klientská aplikace připojena k ovládanému počítači, je možné použít dotykový displej Android zařízení jako touchpad. Červená tlačítka nahrazující levé a pravé tlačítko myši jsou umístěna ve spodní části obrazovky.

Pro trvalé stisknutí levého tlačítka myši je určen checkbox ve spodní části obrazovky.

Virtuální klávesnici je možné vyvolat kliknutím na ikonu klávesnice ve spodní části obrazovky. Zadávaní znaků s diakritikou nefunguje, tato funkčnost je uvedena v možnostech budoucího rozšíření.

Nápověda je zobrazena po stisku kontextového tlačítka "MENU" a následném výběru možnosti "Help".