

VŠB – Technická univerzita Ostrava Fakulta  
Elektrotechniky a informatiky  
Katedra Informatiky

Softwarové rozhraní inteligentního domu  
Software Interface for Intelligent Household

Ostrava 2011

Bc. Jan Plucar

ZDE PROSIM VLOZTE ZADANI

**Prohlášení studenta:**

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal. V Ostravě dne 1.5.2011

Bc. Jan Plucar

### **Poděkování:**

Rád bych poděkoval vedoucímu mé diplomové práce, panu Ing. Michalu Krumníkovi za trpělivost a pomoc při řešení problémů spojených s diplomovou prací.

## **Abstrakt**

Práce se zabývá návrhem vhodného softwaru pro systém ovládání spotřebičů v domácnosti s použitím embedded zařízení. Projekt je navržen tak, aby co nejvíc ulehčil život fyzicky postiženým lidem. Cílem je návrh a implementace řešení, které by bylo levné, robustní a rozšiřitelné. Tato diplomová práce navazuje na práci Bc. Patrika Dubce, který navrhnul pro dané řešení hardware. V této práci je pak softwarová analýza projektu, návrh prezentační vrstvy a logické vrstvy. Práce je zakončena nasazením a vyhodnocením uživatelské spokojenosti.

Klíčová slova: embedded, Windows embedded standard 7, windows služba, webové služba, softwarová analýza

## **Abstract**

The paper describes the design of suitable software for controlling home appliances using embedded devices. The project is designed to make life much easier for physically disabled people. The aim is to design and implement a solution that would be cheap, robust and scalable. This thesis builds on the work of Patrick Bachelor Dubeč, who designed the hardware for the solution. In this thesis, we work with the software project analysis, design of presentation layer and logic layer. The work ends with the deployment and evaluation of user satisfaction.

Keywords: Embedded, Windows Embedded Standard 7, Windows services, Web services, software analysis

## **Seznam použitých symbolů a zkratk**

OS - operační systém

WES7 - Windows Embedded Standard 7

XPE - Windows Xp Embedded

HW - hardware

SW - software

UP - unified process

1	Úvod .....	9
2	Návrh systému pro handicapované osoby .....	10
2.1	Systémy pro osoby s pohybovým handicapem .....	10
2.1.1	Systémy pro rozpoznání řeči .....	10
2.1.2	Systémy pro sledování pohybu očí .....	10
2.2	Systémy pro osoby se zrakovým handicapem .....	11
2.2.1	Návrh systému pro barvoslepé a slabozraké .....	11
2.2.2	Čtečky .....	11
2.3	Systémy pro osoby se sluchovým handicapem .....	11
3	Analýza .....	12
3.1	Zadání klienta .....	12
3.2	Zaměření projektu .....	12
3.3	Identifikace stakeholderů .....	13
3.4	Analýza požadavků .....	14
3.5	Uživatelské scénáře (user stories): .....	15
3.6	Identifikace případů užití .....	16
3.7	Funkční požadavky .....	17
3.8	Nefunkční požadavky .....	18
3.9	Analýza nákladů .....	18
3.9.1	Sestava HW .....	18
4	Návrh .....	23
4.1	Návrh fyzické architektury .....	23
4.2	Návrh softwarové architektury .....	24
4.2.1	Vícevrstvá architektura, návrhový vzor MVC .....	26
4.2.2	Návrhový vzor fasáda .....	27
5	Implementace .....	28

5.1	Návrh uživatelského rozhraní.....	28
5.1.1	Style a DataTemplate .....	29
5.1.2	Grafický návrh pro mobilní zařízení a server .....	31
5.2	Konkrétní implementace Model-View-Controller návrhového vzoru .....	33
5.3	Zjednodušení přístupu k funkcím - návrhový vzor fasáda .....	34
5.4	Vyjímky, varování, informativní zprávy .....	35
5.5	Služba realizující časové události.....	36
6	Nasazení v reálném provozu .....	37
6.1	Nasazení .....	37
6.1.1	Diagram nasazení .....	38
6.1.2	Postup při nasazení .....	38
6.1.3	Postup v případě pádu systému .....	39
6.1.4	Technické aspekty nasazení .....	39
6.1.5	Připomínky klienta .....	40
7	Shrnutí výsledků nasazení a přínosů pro cílovou skupinu .....	42
7.1	Jiná řešení na trhu v porovnání s naším řešením .....	45
7.1.1	Insight home .....	45
7.1.2	Haidy .....	47
8	Závěr.....	50



# 1 Úvod

Cílem této práce je navrhnout a naimplementovat aplikaci, sloužící k dálkovému ovládní inteligentního domu. Svým diplomovým projektem budu doplňovat práci Bc. Patrika Dubce, který má za úkol nalézt a připravit vhodný hardware pro toto řešení. Hlavním přínosem práce by pak mělo být usnadnění života tělesně postiženým lidem, pro které je tento projekt primárně určen. Řešení by mělo nabízet velkou míru modularity, robustnost a snadnou ovladatelnost. Budeme se zabývat otázkou integrace stávajících komponent domácnosti do systému, rešerší systémů stávajících.

Po celou dobu realizace projektu se budeme řídit metodikou UP (Unified Software Development Process - unifikovaný proces vývoje softwaru), které nám poskytuje rámec pro uchopení celé problematiky vhodným způsobem.

Tato metodika je založena na interaktivním vývoji softwaru, což pro nás vlastně znamená funkční dekompozici problému na menší části a jejich postupný vývoj. Dílčí části se budou postupně spojovat, až nakonec vznikne kompletní řešení, které bude otestováno. Výhodou iterativního vývoje je odhalení velkého množství chyb již v průběhu vývoje softwaru.

Nyní bych rád shrnul obsah této diplomové práce. V prvním kapitole se podíváme na průzkum trhu se stávajícími řešeními. Ukážeme si jejich použitelnost, náklady na integraci a pokusíme se na těchto řešeních najít chyby, kterých bychom se měli vyvarovat. Poté provedeme analýzu problému a vytvoříme si seznam požadavků (funkčních i nefunkčních). Na základě těchto požadavků navrhne architekturu systému a zvolíme vhodnou platformu, pro kterou vytvoříme funkční obraz operačního systému Windows Embedded (postup vytvoření obrazu bude detailněji popsán v práci Bc. Patrika Dubce). V kapitole implementace si ukážeme konkrétní postupy při překlopení funkčních požadavků na software. Popíšeme použití zajímavých programátorských praktik, jako je používání návrhových vzorů. Jednou z nejdůležitějších kapitol pak bude kapitola věnující se nasazení projektu do reálného provozu a následného vyhodnocení výsledků.

## **2 Návrh systému pro handicapované osoby**

Každá skupina postižených má své specifické požadavky. Při návrhu systému je třeba položit si následující otázky:

1. Jaký typ postižení mohou uživatelé mít a jako to ovlivňuje použití softwaru?
2. Jaké problémy mají postižení se současným softwarem?
3. Jaké podpůrné technologie mohou tito lidé použít?
4. Jaké technologie a nástroje může vývojář využít?
5. Jaké vedlejší efekty může mít úprava softwaru?

### **2.1 Systémy pro osoby s pohybovým handicapem**

#### **2.1.1 Systémy pro rozpoznání řeči**

Účelem těchto systémů je rozpoznání mluvené řeči (případně jiného zvukového zdroje). Tyto systémy dovolují lidem, kteří se nemohou pohybovat, ovládat technologie bez větších omezení. První použití tohoto systému bylo v roce 1952 [1]. V současné době se začínají tyto nástroje integrovat do běžných systémů (Windows Vista, Windows 7). Systémy pro rozpoznání řeči jsou často implementovány pomocí samoučících neuronových sítí [2]. Nevýhodou této implementace může být závislost na daném vzorku. Tento druh systému je, díky integraci v běžných systémech, často vylepšován.

#### **2.1.2 Systémy pro sledování pohybu očí**

Tyto systémy sledují pohyb očí a převádí jej na pohyb myši. Existují dva druhy zjištění pohybu očí [3]. První metoda zjišťuje pohyby ve svalech, které ovládají oči. Tato metoda je velice neefektivní. Druhou metodou je rozpoznávání pomocí kamery a analýzy obrazu.

Tyto systémy nejsou tak často vylepšovány jako systémy pro rozpoznání řeči, ale i tak mohou najít uplatnění v běžné sféře například pro analýzu reklamy [4].

## **2.2 Systémy pro osoby se zrakovým handicapem**

### **2.2.1 Návrh systému pro barvoslepe a slabozraké**

Zrakovým handicapem nemusí být myšlena pouze slepota. Uživatelé mohou být barvoslepi nebo slabozrací. Při návrhu uživatelského rozhraní je nutné přihlídnout i k těmto skupinám. Je třeba zvolit správné barvy, kontrast mezi podkladem a barvou textu. Důležitá je také velikost písma.

Nejčastější barvoslepostí je Deuteranomaly [9] (6% muži, 0.4% ženy).

### **2.2.2 Čtečky**

Nejrozšířenějším nástrojem pro pomoc slepým (nebo jinak zrakově handicapovaným) osobám jsou čtečky textu. Tyto nástroje postupně čtou texty na obrazovce. Stejně jako systémy pro rozpoznání řeči jsou i čtečky textu integrovány do běžných systémů. Existují i čtečky, které pouze načtou obsah, ale informují uživatele o objektech na obrazovce [6].

## **2.3 Systémy pro osoby se sluchovým handicapem**

Komunikace s počítačem pro hluché osoby je mnohem jednodušší než pro slepé. Je však nutné vyřešit například komunikaci pomocí VOIP nástrojů. Pro titulky lze využít opět nástrojů pro rozpoznání řeči. Existují také nástroje pro interpretaci znakové řeči.

Při návrhu uživatelského rozhraní je také nutné brát v potaz, že jakékoliv zvukové oznámení (zvonící telefon) nebo alarm musí být doprovázen grafickou reprezentací.

### 3 Analýza

V této kapitole se budeme soustředit na analýzu systému jako celku. Analýzu jsem prováděl na základě dobrých praktik RUP, a tak svou strukturou připomíná artefakt RUP - "Vision dokument". Proces analýzy je jeden z nejdůležitějších kroků vývoje softwaru, z důvodů nalezení veškeré potřebné funkcionality, nalezení časově náročných problémů a potenciálních chyb.

#### 3.1 Zadání klienta

Chtěla bych si nechat vytvořit systém, který by mi umožnil řídit a ovládat některá zařízení v domě, kde bydlím. Díky mé částečné fyzické indispozici mám problémy s některými běžnými úkony. Proto bych si přála, aby mi výsledný produkt usnadnil alespoň některé z těchto činností. Problémy mi činí například dostatečně rychle reagovat na zvonění lidí u dveří. Často se mi tak stává, že než stihnu dojít ke dveřím, tak návštěva odejde. Obecně mi činí zvýšenou námahu, když musím vstávat. Ocenila bych tak například možnost ovládání světel v domě, zatáhnutí žaluzií u oken. Systém by mohl automaticky ovládat topení podle teploty v místnosti a udržovat tak nastavenou teplotu. Občas se mi stane, že potřebuji pomoc druhé osoby, systém by tedy měl umět oznámit někomu v domě, že potřebuji pomoc.

K ovládání bych chtěla používat dálkový ovladač, který bych mohla mít neustále u sebe. Ovladač se mi musí pohodlně ovládat (veliká tlačítka). Pro zbytek rodiny by mělo být přístupné nějaké centrální ovládání. Zároveň celé řešení nesmí být příliš drahé.

#### 3.2 Zaměření projektu

Pro	Osoby s fyzickým postižením
Proč	Současná řešení jsou příliš drahá a v mnoha případech neberou ohled na osoby s postižením.
Název	Diplomový projekt
Přínos	Největším přínosem je cena systému, která se pohybuje na hranici 20% ceny konkurenčních řešení.

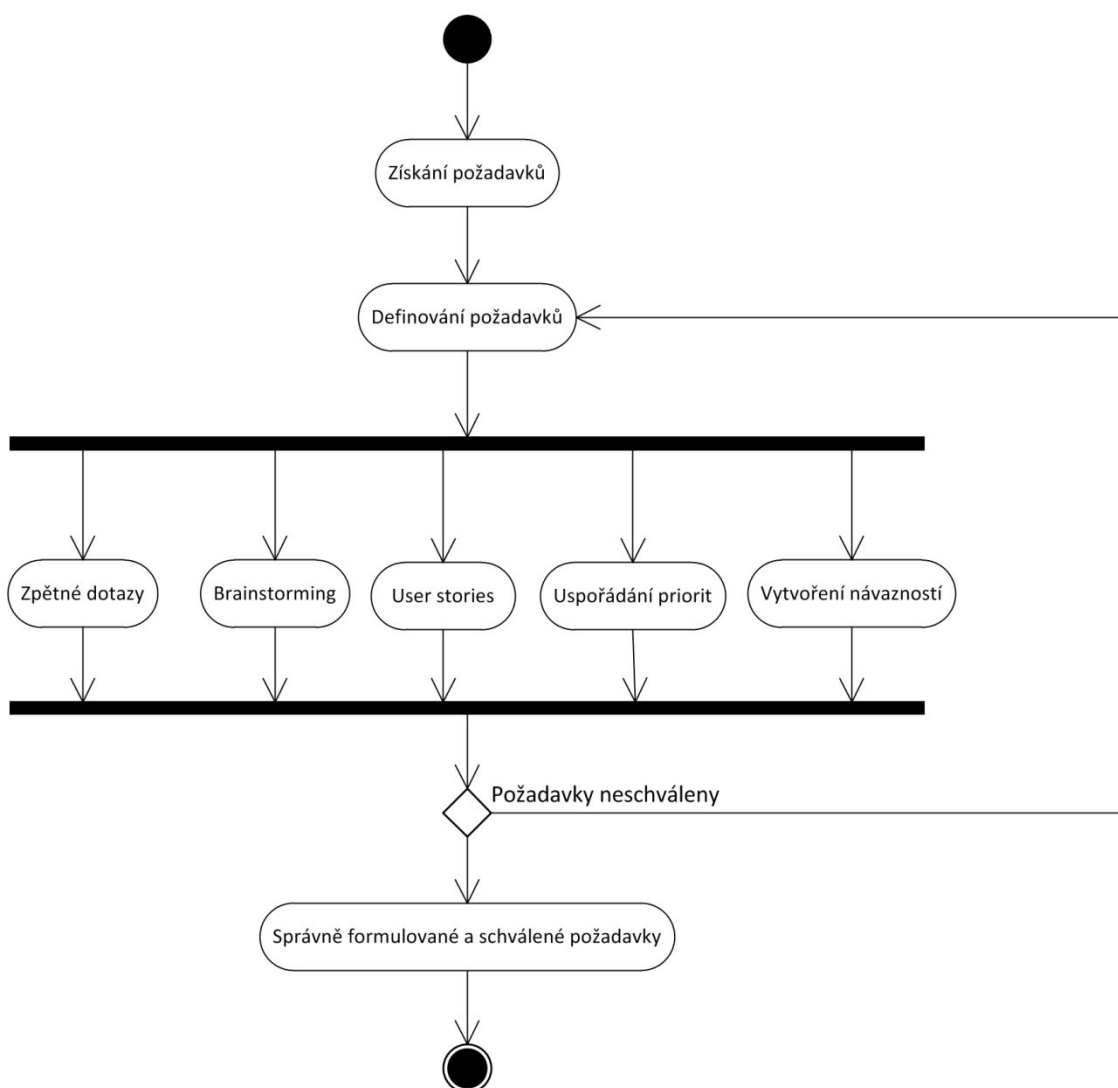
### 3.3 Identifikace stakeholderů

V následující tabulce jsou zmíněni všichni nalezení stakeholdři, které jsem získal analýzou zadání a dotazováním na první schůzce s klientem. Důležitost nalezení stakeholderů plyne z možnosti získání nových zdrojů informací. V našem případě jsme například našli stakeholdera "Technik", který nebyl na první pohled zjevný. Tento stakeholder může ovlivnit zejména seznam nefunkčních požadavků.

Jméno	Popis	Odpovědnost
Uživatel	Osoba s fyzickým postižením, která bude systém ovládat.	
Programátor	Osoba vzdělaná v oboru informačních technologií.	Programátor zajistí návrh a vytvoření systému, vyhodnotí zprávu testera a v průběhu vývoje bude průběžně opravovat chyby, které s vývojem systému vyvstanou.
Technik	Osoba s oprávněním provádění práce na zařízeních spadajících do kategorie "slabo proud"	Technik zajistí zapojení systému v místě instalace.
Tester	Osoba s fyzickým postižením, která bude systém po dobu několika týdnů testovat.	Tester zajistí otestování softwaru formou používání a formou A/B testů.

### 3.4 Analýza požadavků

Specifikace požadavků je jedna z nejdůležitějších aktivit procesu analýzy. Je to první možnost vývojového týmu pro nalezení možných chyb systému. Ač se může zdát, že požadavky slouží jen ve fázi analýzy pro uvedení a zjednodušení zadání, tak tomu tak není. Požadavky, zejména pak funkční požadavky jasně definují hranici systému. Co systém umí a co ne, takže se v průběhu vytváření softwaru testuje, jestli dané komponenty odpovídají požadavkům (akceptační testování viz. V-model testování). Požadavkům jsme přiřadili prioritu a návaznost (dependency), což nám pomohlo k vytvoření plánu iterací vývoje.



Obr.1 - tvorba požadavků

### 3.5 Uživatelské scénáře (user stories):

Uživatelské scénáře jsou získávány od klíčových stakeholderů, kteří popíší pro ně důležité funkce krátkým "příběhem". Analytik pak tyto scénáře přepíše do jednotného formátu, který je používán v celé analýze.

Jeden z nejčastěji používaných formátů:

Jako <uživatel> chci <provést akci> tak, abych <získal nějaký konkrétní výstup>.

**Aktér:** vlastník uživatelského scénáře. Měl by to být specifický stakeholder/uživatel systému.

**Akce:** co chce vlastně stakeholder/uživatel udělat. Akce může být buď povinná nebo volitelná, což se rozlišuje použitím klíčového slova chci nebo musím.

**Konkrétní výstup:** výsledek po provedení akce.

Mnou získané uživatelské scénáře:

US1. Jako Uživatel chci zjistit kdo zvoní u dveří tak, aby jsem k získání této informace nemusel fyzicky přijít ke dveřím.

US2. Jako Uživatel musím být schopen oznámit lidem v domě, že potřebuji pomoc tak, aby jsem měl jistotu, že mé volání o pomoc slyší osoby v celém domě.

### 3.6 Identifikace případů užití

<b>Případ užití přiřazení zařízení</b>
ID: 1
Stručný popis: Uživatel vybere generické zařízení ze seznamu volných Quido výstupů a přiřadí na něj zařízení.
Hlavní aktéři: Uživatel, Technik, Programátor
Vstupní podmínky: Hlavní aktér je ověřen vůči systému.
Hlavní scénář: <ol style="list-style-type: none"><li>1. Uživatel spustí program ovládání domácnosti na serveru</li><li>2. Systém zobrazí nabídku funkcí</li><li>3. Uživatel vybere z nabídky funkci nastavení</li><li>4. Systém zobrazí seznam volných portů Quida</li><li>5. Uživatel zvolí jeden z portů, který si pojmenuje tak, aby svým jménem demonstrovalo funkčnost připojeného zařízení.</li><li>6. Systém uloží změny a informuje uživatele o možnosti přidání kategorie</li><li>7. Uživatel zvolí možnost přidání kategorie:<ol style="list-style-type: none"><li>a. Uživatel nezvolil kategorii a zařízení bylo přidáno do kategorie "všechna zařízení"</li><li>b. Uživatel zvolil kategorii a zařízení bylo přidáno do dané kategorie</li></ol></li><li>8. Systém uloží změny a informuje uživatele</li></ol>
Výstupní podmínky: Systém zapíše do databáze informace o novém zařízení.
Alternativní scénáře:

<b>Případ užití vytvoření časové akce</b>
ID: 2
Stručný popis: Uživatel založí v systému novou událost, která se spustí v předem daný čas a dny.
Hlavní aktéři: Uživatel, Technik, Programátor
Vstupní podmínky: Hlavní aktér je ověřen vůči systému.
Hlavní scénář: <ol style="list-style-type: none"><li>1. Systém zobrazí webovou stránku</li><li>2. Uživatel vybere zadání nové události</li><li>3. Systém se zeptá kdy se má akce vykonávat</li><li>4. Uživatel zadá start a konec události</li><li>5. Systém vyzve uživatele, zda chce událost opakovat<ol style="list-style-type: none"><li>a. Uživatel zadá opakování události formou výčtu dnů</li><li>b. Uživatel nechce událost opakovat</li></ol></li><li>6. Systém uloží novou</li></ol>
Výstupní podmínky: Systém zapíše do databáze informace o nové události.
Alternativní scénáře:



### 3.7 Funkční požadavky

- FR1. Systém bude obsahovat komponenty pro připojení do lokální wifi sítě.
- FR2. Systém bude udržovat spojení s hardwarem a zjišťovat stav HW.
- FR3. Systém bude ovládat světla v celé domácnosti.
- FR4. Systém bude ovládat kameru připojenou před dveřmi.
- FR5. Systém bude ovládat signalizační přístroj.
- FR6. Systém bude umět zjistit teplotu v místnosti.
- FR7. Systém bude reagovat na příkazy dálkového ovládání.
- FR8. Systém bude reagovat na příkazy serveru přijaté přes centrální konzoli.
- FR9. Systém bude umět regulovat teplotu.
- FR10. Systém bude obsahovat plánovač úloh.
- FR11. Systém bude umět otevřít dveře.

Název	Priorita (1-5, 1 vysoká, 5 nízká)	Závislost
FR1	2	-
FR2	1	-
FR3	3	FR2
FR4	3	FR2
FR5	3	FR2
FR6	4	FR2
FR7	1	FR1, FR2
FR8	1	FR2
FR9	4	FR2
FR10	2	FR2
FR11	3	FR2

### **3.8 Nefunkční požadavky**

NFR1. Řešení bude využívat pro svůj běh embedded zařízení s nízkým výkonem, ale s malými provozními náklady.

NFR2. Řešení bude udržovat malou odezvu systému.

NFR3. Řešení by nemělo klienta zatěžovat mnoha ovládacími prvky, v nejlepším případě by měl být k ovládní použít mobilní telefon.

NFR4. Systém bude přístupný z celého domu.

### **3.9 Analýza nákladů**

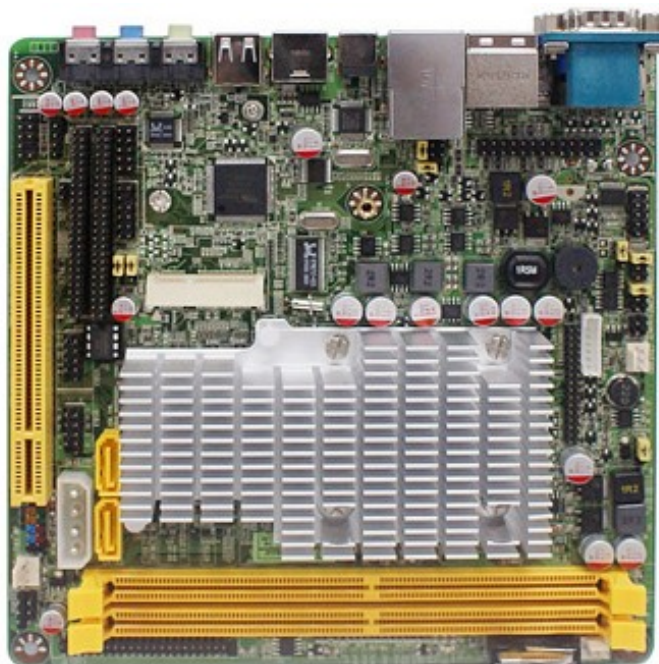
V této sekci se budu zabývat výpočtem celkových nákladů na hardware, instalaci zařízení a nasazení systému. Kdyby se tato analýza tvořila na míru klientovi, je tato sekce důležitá proto, aby se klient mohl rozhodnout, zda jsou finanční nároky projektu přijatelné. My budeme automaticky počítat s tím, že nám klient cenu schválil.

#### **3.9.1 Sestava HW**

Na základě doporučení jsem provedl kalkulaci cen nad dvěma možnými hardwarovými sestavami i. Ceny se liší hlavně v použití jiné základní desky, zdroje a počítačové skříně.

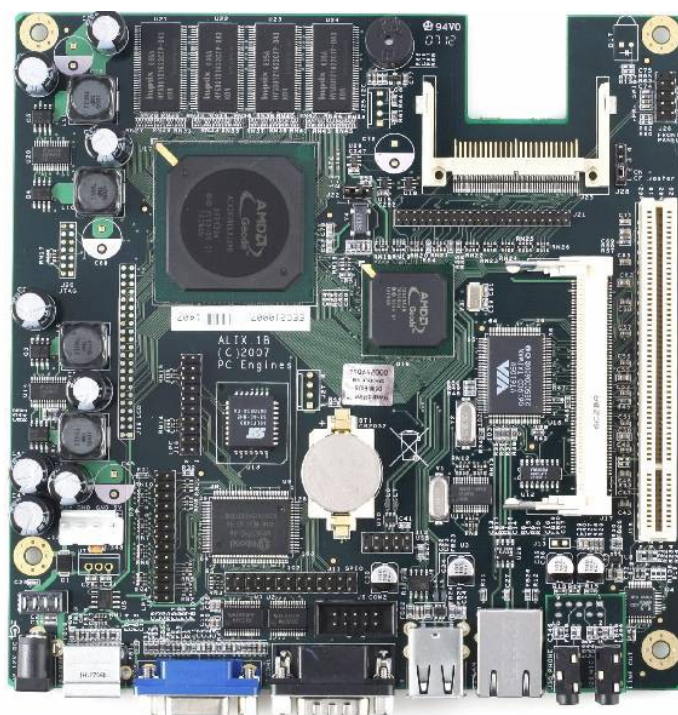
Dříve než však uvedu obě kalkulace, tak Vám ve zkratce představím navržené HW řešení:

## 1. Základní deska (server)



Obr.2 - MB Jetway NF96

Jetway NF96	
Procesor	Intel Atom D525 (2x 1.8 GHz, 1Mb cache)
Podporované OS	Windows Embedded 7
Operační Paměť	2x DDR2 800/667 SDRAM
Disková uložení	2x konektor Serial ATA2 3 Gb/s
USB konektory	5x USB 2.0/1.1
Video výstup	VGA
Síťová rozhraní	1x Gigabit Ethernet LAN RJ-45, 1x (Mini PCI) Wi-Fi (802.11 b/g/n)
Konektivita	1x RS-422/485 sériová linka COM1, 1x paralelní port, 1x LVDS
Rozměry	170mm x 170mm
Cena	2300 Kč



Obr.3 - MB Alix 1D

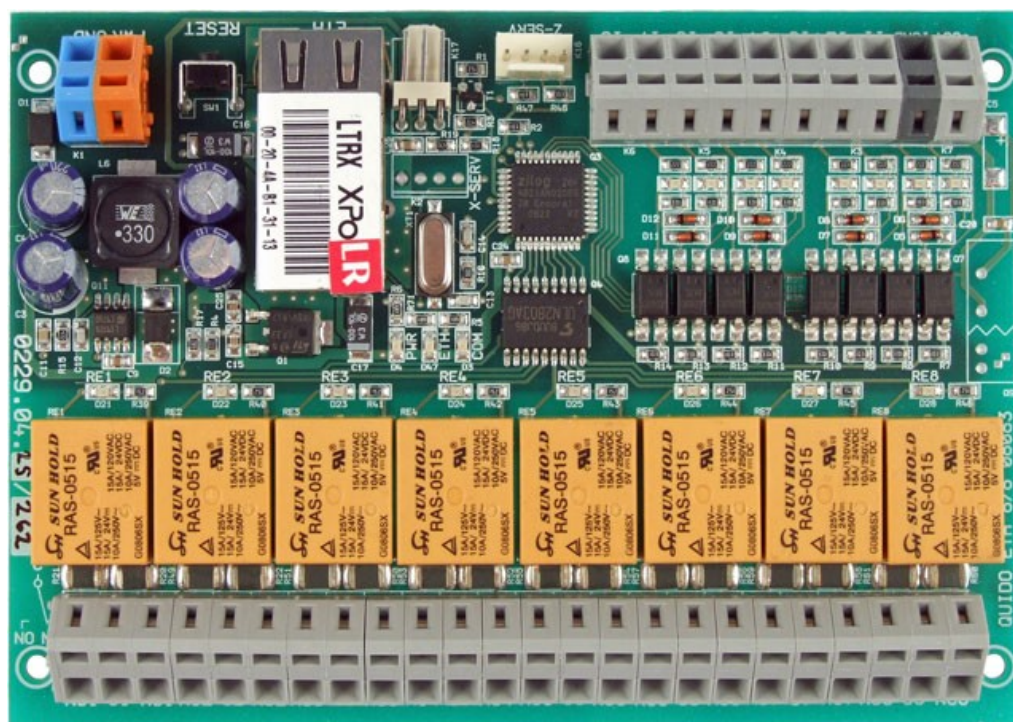
Alix 1D	
Procesor	500 MHz AMD Geode LX
Podporované OS	Win CE6.0
Operační Paměť	256MB DDR
Disková uložení	1x CompactFlash slot, 1x 44 pin ATA konektor
USB konektory	4x USB 2.0
Video výstup	VGA (1024x768)
Síťová rozhraní	1x Ethernet, RJ-45, 1x Mini PCI
Konektivita	2x RS485 sériová linka(COM5, COM6)
Rozměry	185mm x 185mm
Cena	3000 Kč

## 2. Quido ETH 8/8

Quido ETH 8/8 má 8 digitálních vstupů, 8 výstupních relé a teploměr. Komunikuje přes Ethernet. Vstupy jsou pro připojení napětí nebo pro kontakt, výstupy jsou osazeny relé s přepínacím kontaktem.

### Vlastnosti I/O modulu:

- Ovládání a dohled přes Ethernet (10/100 Ethernet)
- Možnosti přístupu: Spinel, TCP a UDP datové spojení
- 8 galvanicky oddělených logických vstupů pro napětí nebo pro kontakt.
- 8 výstupů s přepínacím kontaktem relé.
- Možnost počítání změn na vstupech.
- Rozsah pracovních teplot od -20 do +70 °C.
- Napájení z externího zdroje 8 - 30V.



Obr.4 - Quido ETH 8/8

Představený hardware je klíčový pro naše řešení a také pro lepší výběr konfigurace sestavy, která bude instalována u klienta. Kompletní a jmenovitý výběr komponent i s jejich popisem a odůvodněním proč byly použity je popsán v diplomové práci Bc. Patrika Dubce.

### 3. Sestava JetWay NF 96

Název	Počet kusů v konfiguraci	Cena za kus	Cena celkem
<b>Návrh hardware</b>			
JetWay NF 96	1	2300 Kč	2300 Kč
Dotykový panel	1	12000 Kč	12000 Kč
PC komponenty	1	2000 Kč	2000 Kč
Quido ETH 8/8	1	5 280 Kč	5280 Kč
Wifi Router	1	560 Kč	560 Kč
Ip kamera	1	4749 Kč	4749 Kč
Elektrický zámek	1	1560 Kč	2560 Kč
Boxy	2	900 Kč	1800 Kč
<b>Instalace</b>			
Prvotní zapojení			10 000 Kč
Test zařízení			3000 Kč
<b>Náklady na mobilní telefon</b>			
HTC Mozart			6638 Kč
HTC 7 trophy			5999 Kč
<b>Náklady celkem</b>			
			<b>55 886 Kč</b>

### 4. Sestava Alix1D

Název	Počet kusů v konfiguraci	Cena za kus	Cena celkem
<b>Návrh hardware</b>			
Alix 1D	1	3000 Kč	3000 Kč
CF karta	1	250 Kč	250 Kč
Dotykový panel	1	12000 Kč	12000 Kč
Quido ETH 4/4	1	3150 Kč	3150 Kč
Wifi Router	1	560 Kč	560 Kč
Boxy	2	900 Kč	1800 Kč
<b>Instalace</b>			
Prvotní zapojení			5000 Kč
Test zařízení			3000 Kč
<b>Náklady na mobilní telefon</b>			
HTC Mozart			6638 Kč
HTC 7 trophy			5999 Kč
<b>Náklady celkem</b>			
			<b>34 759 Kč</b>

## 4 Návrh

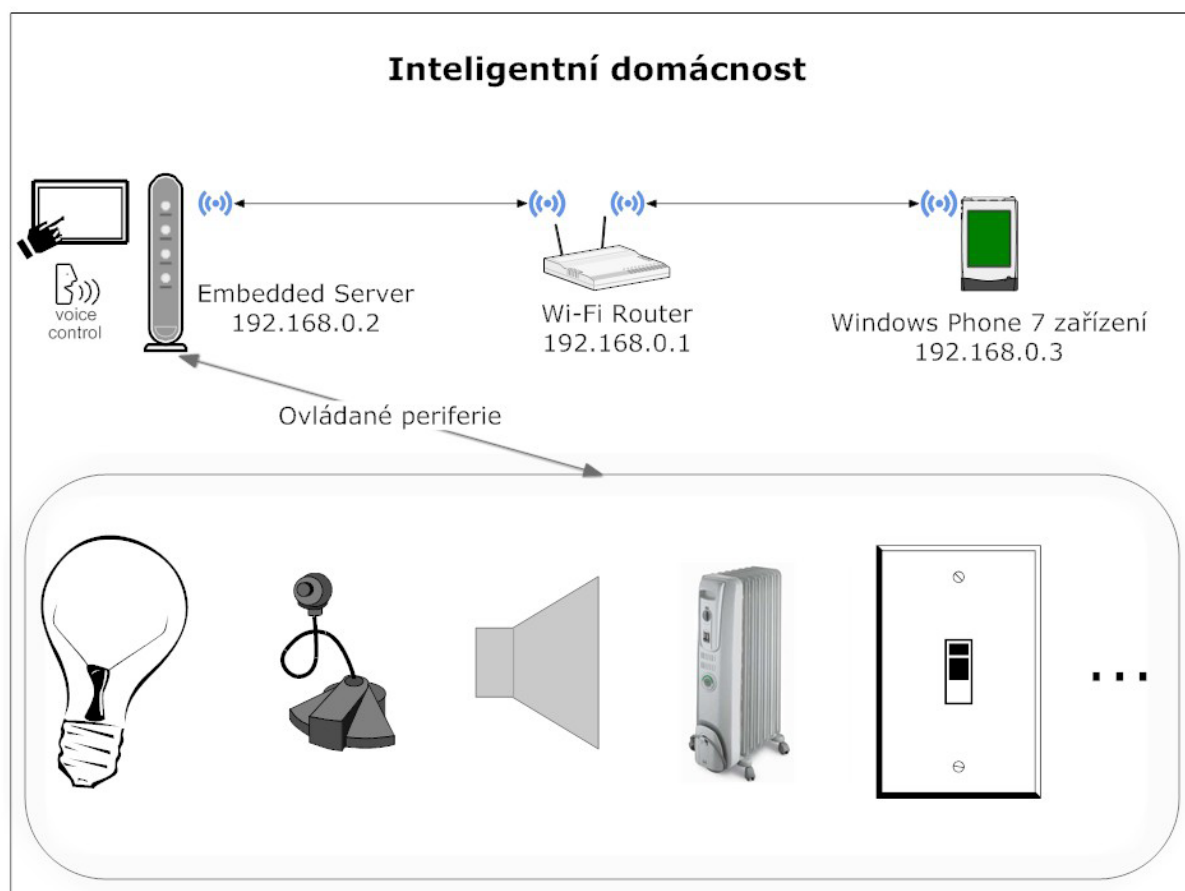
V kapitole návrhu se budeme věnovat přechodu z analytického modelu na návrhový. To znamená transformaci obecně popsaných problémů nalezených v analýze na konkrétní návrh řešení nad cílovou platformou.

### 4.1 Návrh fyzické architektury

Z analýzy víme, že řešení tohoto projektu nebude zabývat vývojem triviálního stand-alone systému. Musíme navrhnout architekturu, která nám podpoří všechny požadavky, funkční i nefunkční a zároveň nebude složitá na pozdější realizaci.

Na obrázku 5 je návrh konečné architektury projektu. Skládá se z několika klíčových částí:

- Embedded server: zařízení fungující jako centrální řídicí středisko. Zde bude nainstalována aplikace, která bude komunikovat s elektronickými zařízeními. Metody pro komunikaci s hardwarem budou zpřístupněny v rámci lokální sítě pomocí webových/wcf služeb.
- Mobilní zařízení: prostřednictvím směrovače se připojuje do lokální sítě, kde funguje jako dálkové ovládání pro hardware.



Obr.5 - architektura systému



## 4.2 Návrh softwarové architektury

Z analýzy a z návrhu fyzické architektury je patrné, že se návrh softwarové architektury rozdělí na dvě části:

- Architektura embedded serveru
- Architektura mobilního zařízení

### **Architektura Embedded serveru:**

Nejdříve si musíme ujasnit, jaké úkoly bude server vykonávat a jakým způsobem je bude vykonávat. Požadovaná funkcionality serveru:

1. Server bude udržovat informace o připojených periferiích
2. Server bude ovládat periferie
3. Uživatel bude komunikovat se serverem prostřednictvím dotykového LCD panelu
4. Server bude reagovat na příkazy mobilního zařízení, sloužícího jako dálkové ovládání

Z uvedených požadavků plyne několik limitací. Zaprvé si musíme uvědomit, že server spadá do kategorie embedded zařízení a bude tedy používat speciální operační systém. Na základě požadavků byl nejprve vybrán operační systém Windows XP embedded, ten byl však v průběhu vývoje nahrazen operačním systémem Windows embedded standard 7 (WES7). Mezi další limitace patří hardwarové specifikace samotného embedded zařízení. Jedna ze specifikací HW navrhovala použít základnu Alix 1D, která obsahuje pouze 256Mb paměti. Tato paměť není navíc rozšiřitelná.

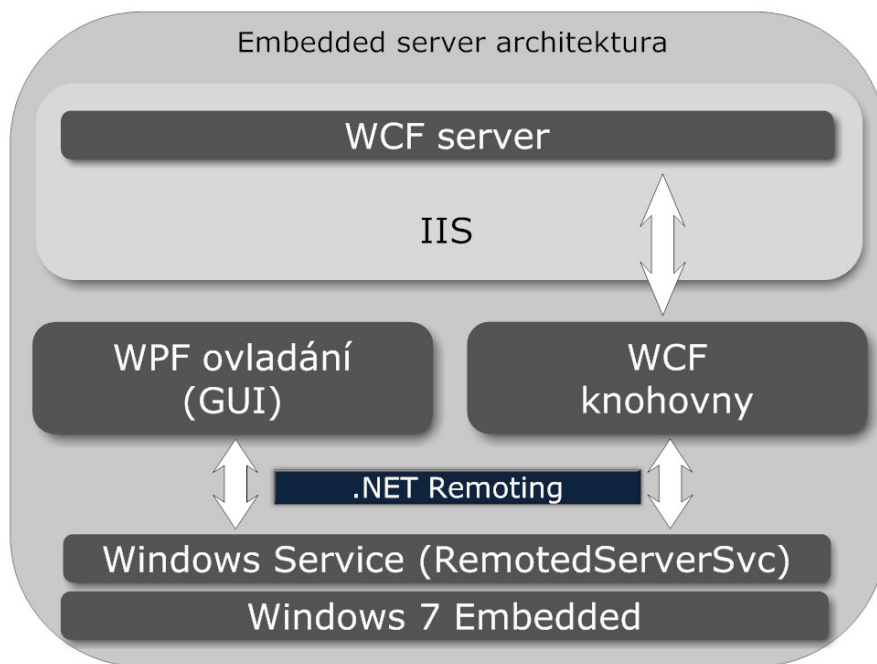
Po zvážení všech těchto faktorů byla navržena architektura viz. obrázek č. 6. Nejdůležitější součástí architektury tvoří:

- IIS - WCF server: zde je hostovaná WCF služba, tedy spíše služby, které budou na venek propagovat potřebné metody.
- Windows Service: integruje obsluhu veškerých periferií.

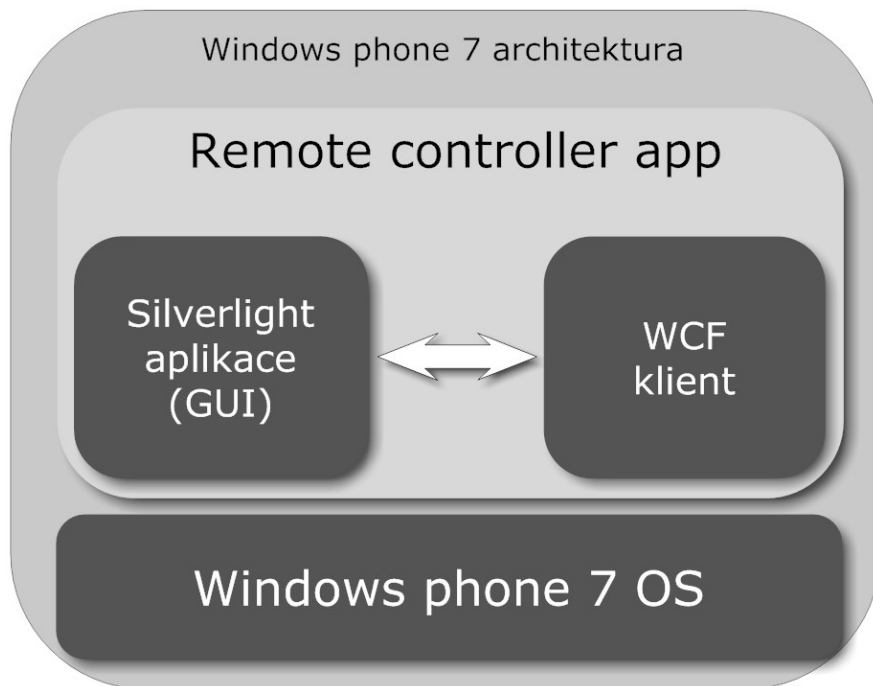
### **Architektura mobilního zařízení:**

Mobilní zařízení s operačním systémem Windows Phone 7. Specifikace tohoto systému je podrobně popsána v diplomové práci Bc. Patrika Dubce.





Obr.6 - architektura serveru



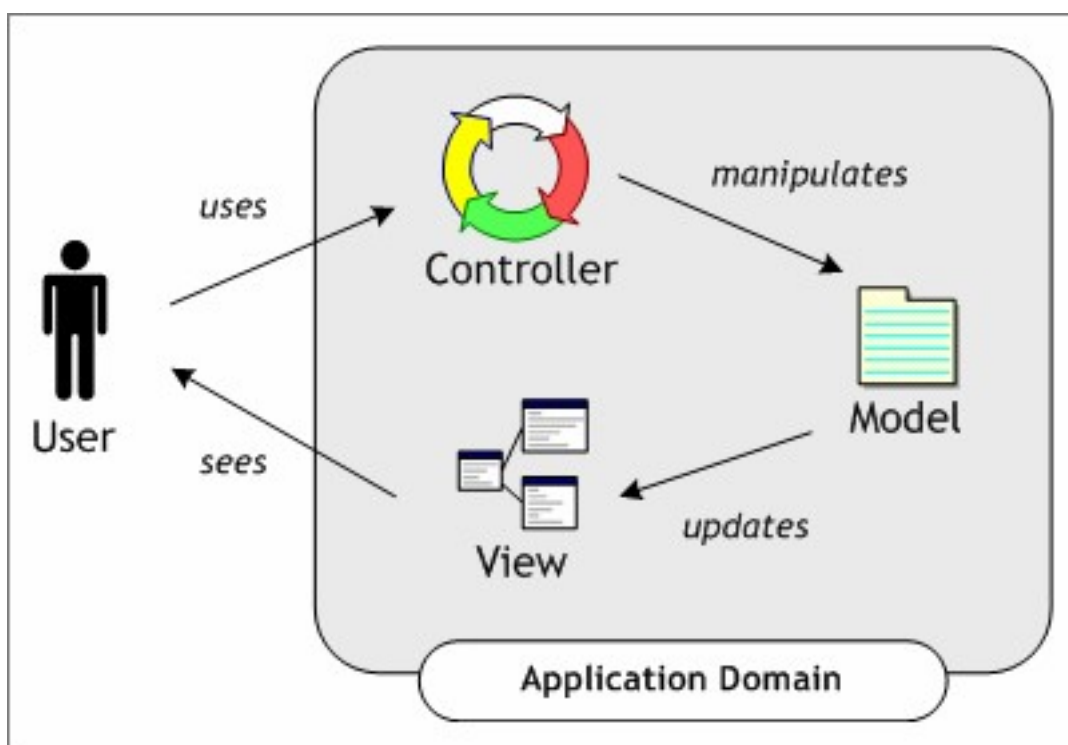
Obr.7 - architektura MS

### 4.2.1 Vícevrstvá architektura, návrhový vzor MVC

V tomto projektu budeme využívat vícevrstvou architekturu, která bude vytvářena podle návrhového vzoru MVC (někdy taky označován jako Model2). Ten slouží pro snadné zobrazování dat klientovi.

Vzor se skládá ze tří částí:

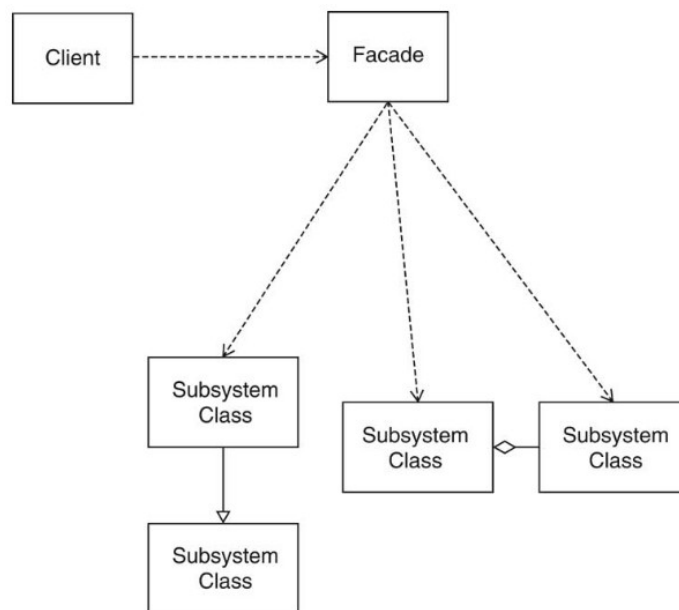
- Model - slouží jako vzor pro data.
- View - část určující zobrazení dat.
- Controller – výkonná část přenášející data do zobrazovací části pomocí modelů.



Obr.8 - návrhový vzor MVC

#### 4.2.2 Návrhový vzor fasáda

Tento systém je vyvíjen jako otevřené řešení, a proto musí být jeho implementace jednoduše pochopitelná a srozumitelná, aby jí mohli využívat i jiní programátoři. Z analýzy víme, že budeme přistupovat k různému hardwaru, jako je například kamera, světlo, siréna. Veškerou funkcionalitu těchto tříd můžeme promítnout do jednoho společného rozhraní. Tím se využívaný kód nesmírně zjednoduší.



Obr.9 - návrhový vzor fasáda

## 5 Implementace

### 5.1 Návrh uživatelského rozhraní

Pro návrh uživatelského rozhraní jsem využil WPF - windows presentation foundation.

#### Výhody využití Windows Presentation Foundation:

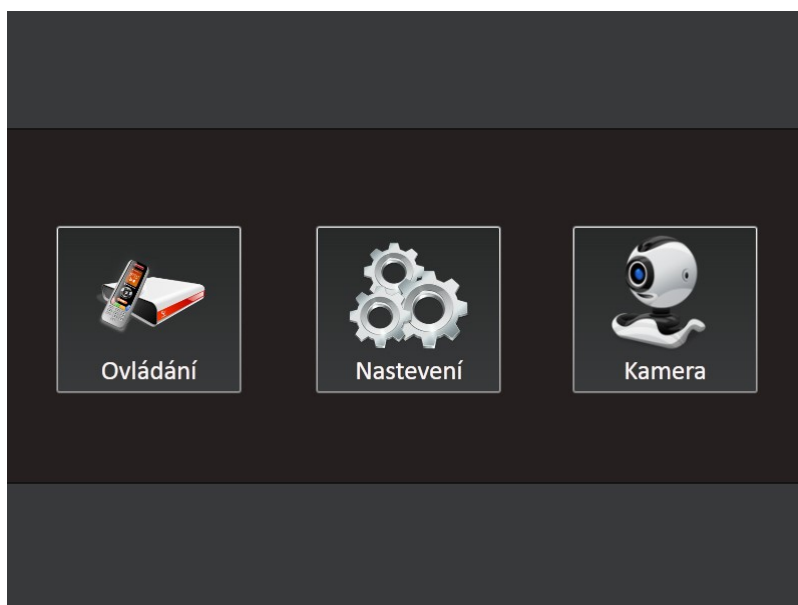
- Uživatelské rozhraní je navrženo pomocí vektorů, lze ho tak přizpůsobit jakémukoliv rozlišení.
- Rozhraní je vykreslováno pomocí grafické karty a je tak při správném hardwaru méně náročné.
- Rozsáhlý rámec obsahující mnoho tříd pro tvorbu libovolného grafického uživatelského rozhraní.
- Existují kvalitní nástroje pro tvorbu WPF rozhraní (Blend, Visual Studio)
- Využívá pro sestavení rozhraní XAML kód (jazyk založený na XML). Rozhraní je editovatelné přímo v tomto kódu a snadno se upravuje.
- Existuje několik kvalitních návrhových vzorů oddělující vzhled od logické části.
- Propracovaný Databinding (Data mohou být provázána přes vrstvy obousměrně s rozhraním).
- Přesné vykreslování objektů (WPF využívá datového typu double k umístování objektů).
- Zjednodušená lokalizace a globalizace.
- Snadná práce s médii

#### Nevýhody WPF:

- Nefunkční na Windows 2000 a nižším
- Při použití špatného hardware mírně náročnější než WinForms

Při tvorbě uživatelského rozhraní jsem kladl důraz na fyzické postižení klientky, která má omezenou motoriku a další již zmíněné fyzické indispozice. Proto musel být návrh grafického rozhraní jednoduchý, ovládací prvky musely být velké a dostatečně tolerantní na akci kliku.

Na číslo 10 je fotografie hlavního menu programu, který je spuštěný na serveru. Grafické uspořádání je jednoduché a přehledné. Barvy jsou vybrány tak, aby byly kontrastní a dobře viditelné.



Obr.10 - hlavní menu

### 5.1.1 Style a DataTemplate

Styly ve WPF fungují podobně jako kaskádové styly u webových stránek. V následujícím příkladu jsme nadefinovali jednoduchý styl, který nastaví parametry typu Button (`TargetType="Button"`). Důležité je definovat atribut `"x:Key"`, který přiřadí našemu stylu jednoznačné jméno, na které se při použití odkazujeme. Obdobně můžeme změnit styl jakékoliv komponentě WPF.

```
<Style x:Key="btnStyler" TargetType="Button">
  <Setter Property="Background" Value="LightGreen" />
  <Setter Property="Margin" Value="5" />
  <Setter Property="FontWeight" Value="Bold" />
</Style>
```

Data template umožňuje programátorovi získat úplnou kontrolu nad způsobem prezentace dat. Při vývoji WPF se počítalo s tím, že se běžně komponenty nenechávají ve svém výchozí grafickém zobrazení, a proto je programátorovi k dispozici několik způsobů jak změnit vzhled komponenty z jednoho centrálního místa. Další typy šablon představují ještě Hierarchical data templates a Control templates. Na následujícím příkladu demonstrujeme vytvoření vzhledu dat pro `listBoxItem` a přiřazení tohoto vzhledu dané komponentě.

```

<DataTemplate x:Key="listBoxTemplateDark">
  <Grid x:Name="LayoutRoot" Width="auto" Height="auto" >

    <Grid.RowDefinitions>
      <RowDefinition Height="0.04*"/>
      <RowDefinition Height="Auto"/>
    </Grid.RowDefinitions>

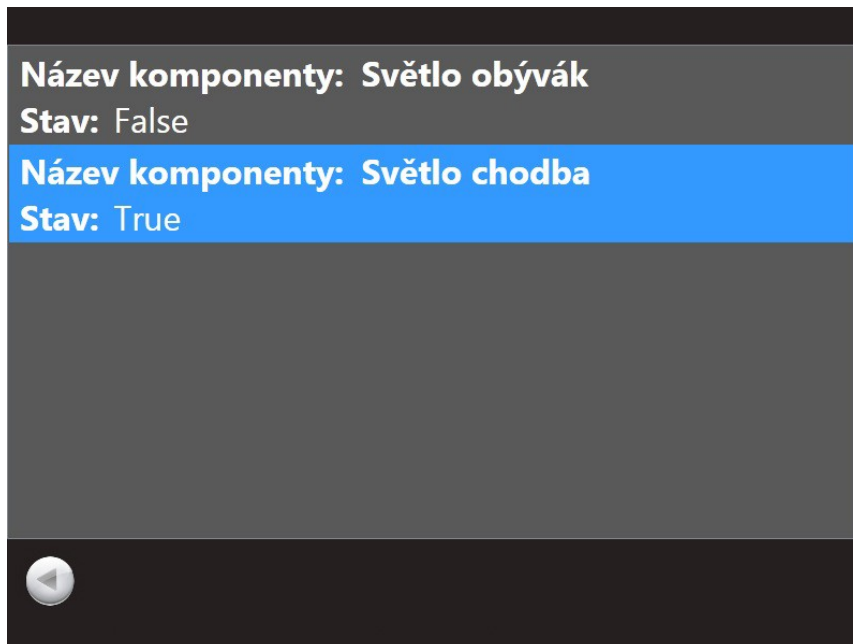
    <TextBlock FontFamily="Calibri" Foreground="White" FontSize="30"
      Grid.Row="1" TextWrapping="Wrap" Text="{Binding NameOfComponent}"/>
    <Image HorizontalAlignment="Right" Width="136" Grid.Row="1"
      Grid.RowSpan="2" Source="{Binding StateUrl}"/>

  </Grid>
</DataTemplate>

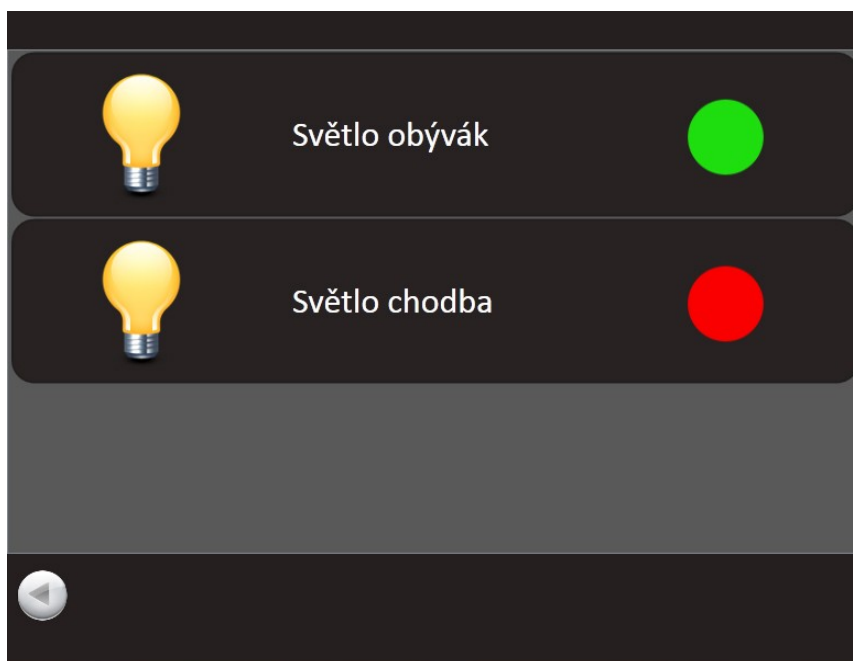
<ListBox ItemTemplate="{StaticResource listBoxTemplateDark}"/>

```

Praktické použití DataTemplate je demonstrováno na obrázku 11 , jehož componenta ListBox zobrazuje stejná data jako jako komponenta ListBox, zobrazená na obrázku 12, používá ovšem jiný DataTemplate. V rámci projektu se tento template používal pro otestování čtecího softwaru, který vyžadoval více textový způsob zobrazení komponent.



Obr. 11 - výběr periferie



Obr. 12 - výběr periferie GUI

### 5.1.2 Grafický návrh pro mobilní zařízení a server

Mezi osvědčené praktiky grafického návrhu patří unifikovaný design komponent stejné funkcionality. Ukázkou této praktiky můžeme sledovat na grafickém rozhraní serveru a mobilního zařízení. V tomto případě se nejedná pouze o stejné zobrazení jedné funkce, ale o připodobnění grafického návrhu stejné sekci.

V procesu návrhu "drátěného modelu" (wireframe) jsem vycházel z následujících předpokladů:

- Jakákoliv funkce systému musí být dosažitelná do třetí grafické vrstvy (tedy za použití 3 kliků mohu spustit jakoukoliv funkci systému).
- Grafické rozhraní musí být stejné, nebo téměř stejné na serveru, tak i v mobilním zařízení. Design se bude tedy primárně podřizovat mobilnímu zařízení.
- Ovládání bude intuitivní.

Konečný grafický návrh pak můžeme porovnat na obrázku 12 pro server a obrázku 13 pro mobilní zařízení. Díky použití jazyka XAML pro definici vzhledu jsem byl schopen řešení jednoduše převést na definici grafiky mobilního telefonu, který má své View definované v jazyce Silverlight. Tím jsem zajistil maximální možnou podobnost obou programů.



Obr.13 - výběr periferie MS



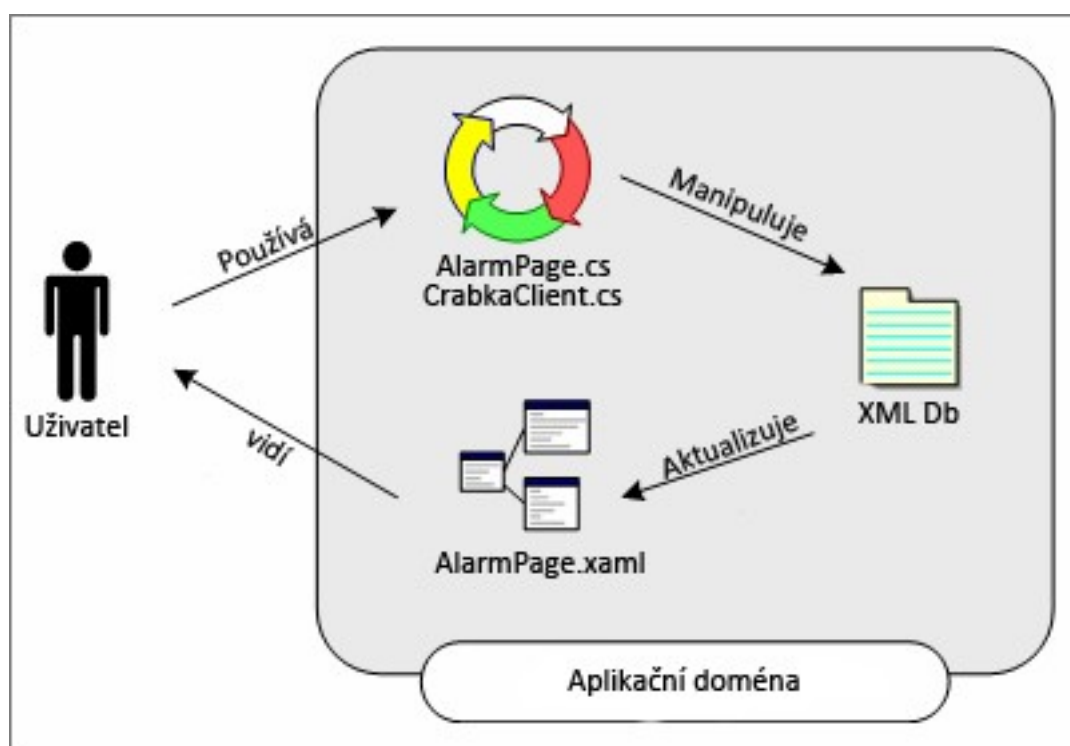
## 5.2 Konkrétní implementace Model-View-Controller návrhového vzoru

Na obrázku 14 je popsána konkrétní implementace návrhového vzoru model-view-controller v našem projektu.

View: je reprezentováno XAML třídami, kde je definována grafika komponent.

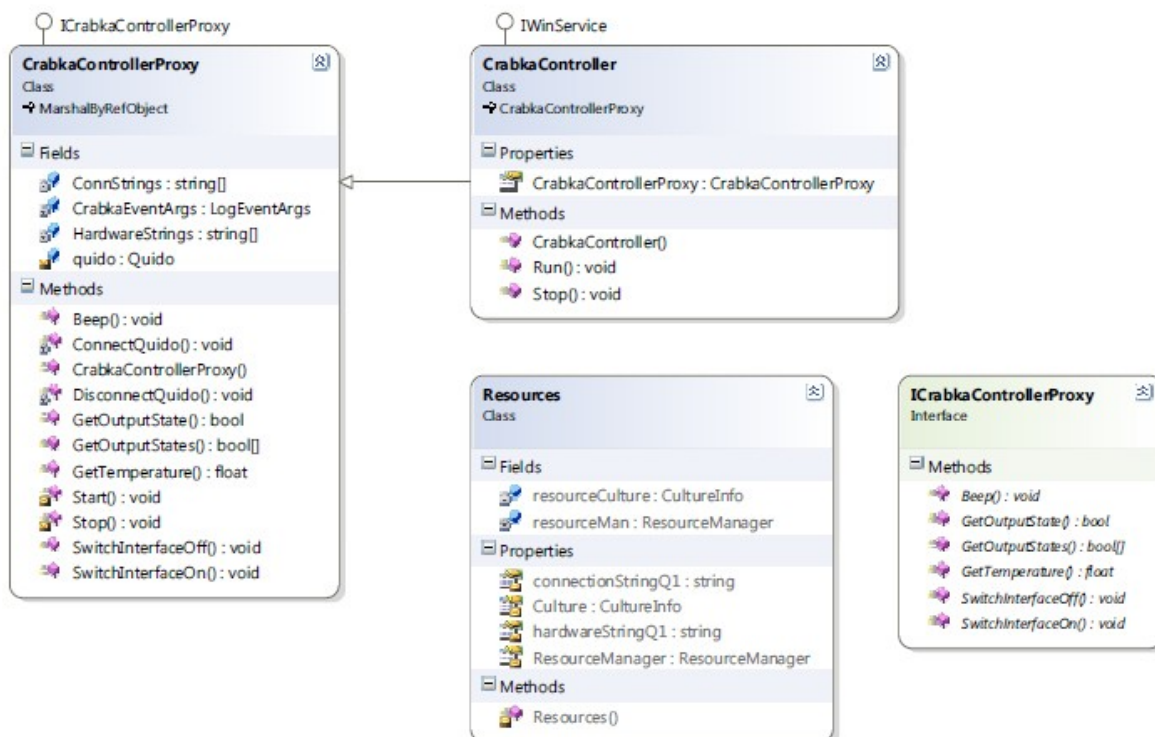
Model: tvoří jej třídy, které načítají data z XML databáze a následně je reprezentují formou objektů.

Controller: tvoří jej třídy, které se starají o spojení se serverem.



Obr.14 - implementace MVC

### 5.3 Zjednodušení přístupu k funkcím - návrhový vzor fasáda



Obr.15 - návrhový vzor fasáda

Díky použití návrhového vzoru fasáda reprezentujeme funkcionalitu na úrovni interfacu a díky tomu "zapouzdříme" samotnou implementaci. Výsledný interface pak pro konkrétní diagram vypadá následovně:

```

namespace WinServ.CrabkaLibrary.Controllers
{
    public interface ICrabkaControllerProxy
    {
        bool GetOutputState(int interfaceNumber, int quidoIndex);
        bool[] GetOutputStates(int quidoIndex);

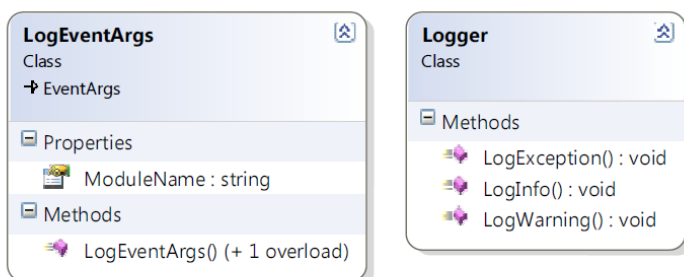
        void SwitchInterfaceOn(int interfaceNumber, int quidoIndex);
        void SwitchInterfaceOff(int interfaceNumber, int quidoIndex);

        void Beep(int interfaceNumber, int quidoIndex, int durationInSecs);

        float GetTemperature(int quidoIndex);
    }
}
    
```

## 5.4 Vyjímky, varování, informativní zprávy

Jelikož jsme dopředu plánovali nasazení projektu, vyvstal nám problém uchovávání chyb a varování, které mohou v průběhu testování vyvstat. Tento problém jsme vyřešili logování chyb do systémového EventLogu.



```
public static void LogWarning(string warningMsg, LogEventArgs logEventArgs)
{
    EventLog.WriteEntry("Zpráva protokolu události",
        String.Format("Modul:{0}\n{1}",
            logEventArgs.ModuleName, warningMsg),
        EventLogEntryType.Warning);
}
```

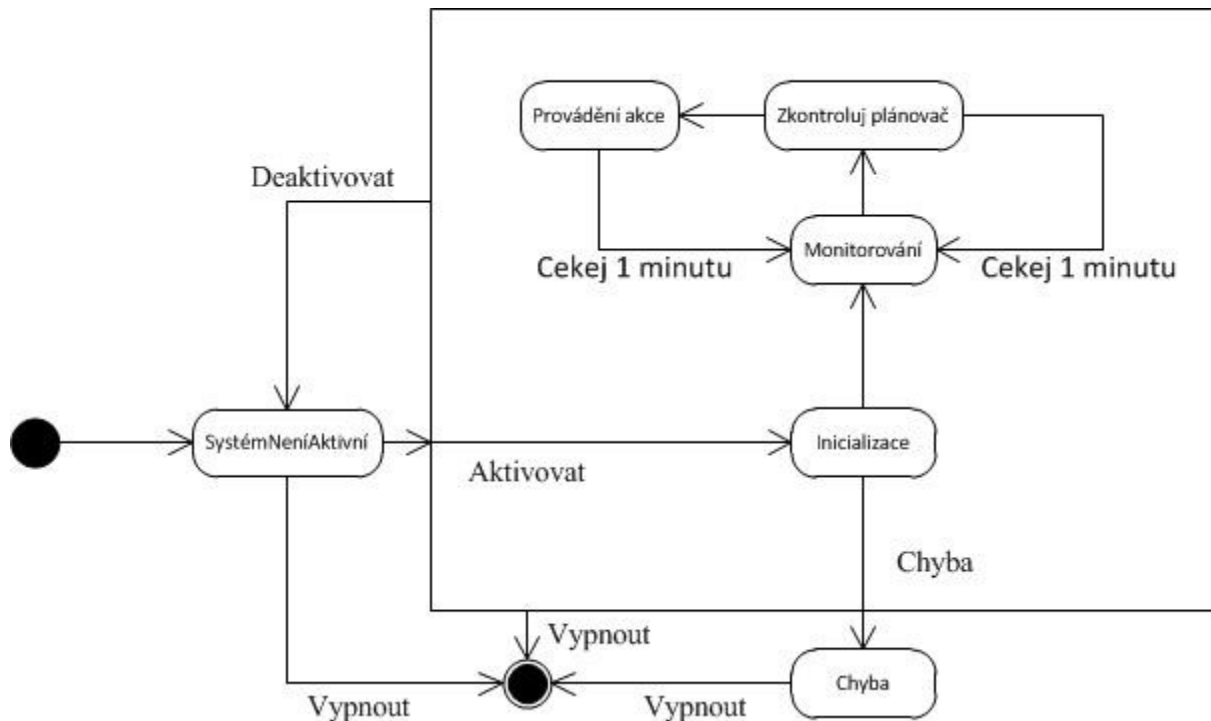
```
private void ConnectQuido(int quidoIndex)
{
    var message = "ConnectQuido({0}) \nresponse = {1}";
    try
    {
        quido = new Quido(ConnStrings[quidoIndex],
            HardwareStrings[quidoIndex]);
        quido.Opened = quido.Open();

        Logger.LogWarning(String.Format(message,
            quido.Opened), CrabkaEventArgs);
    }
    catch (Exception exception)
    {
        Logger.LogException(exception, CrabkaEventArgs);
    }
}
```

## 5.5 Služba realizující časové události

V rámci projektu jsme také řešili časové a podmíněné ovládání událostí. Na obrázku 16 je zobrazen princip práce vlákna, které provádělo následující úkony:

- Vlákno běží na pozadí, což nezatěžuje server.
- Každou minutu zkontroluje DB, kde se nacházejí záznamy určující datum a čas začátku události. Pokud se v dané minutě aktivita vyskytuje, vlákno příslušně reaguje.
- Díky reakci na čas jsme do systému zakomponovali i reakci na událost. Každou minutu získáváme teplotu z čidla a pokud je příliš nízká nebo vysoká, tak se zapne, popřípadě vypne relé ovládající termostat.
- Obdobný princip využívá i vlákno kontrolující konfigurační soubor na disku, ve kterém jsou zapsány stavy periférií. V případě pádu systému by totiž mohla vzniknout nekonzistence stavů.



Obr. 16 - stavový diagram vlákna

## 6 Nasazení v reálném provozu

Tento projekt byl od počátku zamýšlen jako pomoc konkrétním lidem v mém okolí, a proto se v rámci diplomové práce počítalo i s pilotním provozem zařízení přímo v reálném prostředí. Testování trvalo šest týdnů a přineslo mnoho zajímavých poznámek k systému jako takovému. Některé z těchto poznámek (chyb) byly opraveny.

### **Testování bylo prováděno v následujícím prostředí:**

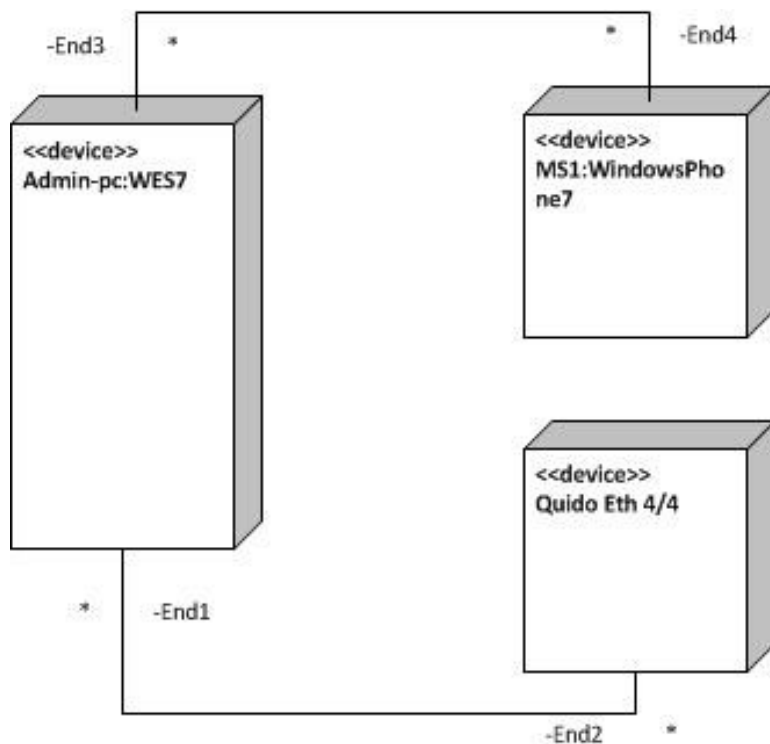
Klient, který náš systém testoval je fyzicky postižený a v současné době pobírá plný invalidní důchod. Klientovi byla amputována pravá noha nad kolenem, levá ruka v lokti a na pravé ruce mu zbyly pouze dva prsty. I přes stávající zdravotní postižení je klient schopen pohybu s protézou, avšak tento pohyb je pro něj fyzicky náročný.

### 6.1 Nasazení

V testovaném domě jsme použili následující konfiguraci hardwaru:

- Napojení světla v obývacím pokoji a ložnici.
- Napojení sirény v hlavní chodbě
- Napojení teploměru v ložnici
- Napojení kamery u vchodových dveří

### 6.1.1 Diagram nasazení



### 6.1.2 Postup při nasazení

#### Instalace hardwaru

1. Uložení serveru do ochranného pouta a umístění do vhodné vzdálenosti od zdroje elektrického proudu
2. Upevnění LCD panelu na zeď ve vzdálenosti max. 1m od serveru
3. Upevnění boxu se zařízením Quido ETH 4/4
4. Upevnění sirény do vzdálenosti max. 15m
5. Upevnění kamery na místo s vhodnou viditelností
6. Odborná instalace digitálního relé do místního elektrického okruhu provedená elektrikářem

#### Konfigurace sítě LAN

1. Připojení serveru k routeru (přímý síťový kabel UTP 5e)
2. Připojení Quida ETH 4/4 k routeru
3. Nastavení DNS jména serveru
4. Konfigurace IP adresy Quida

## Instalace softwaru

1. Služba již je přeinstalována
2. Spuštění windows service služby a nastavení zotavení služby v případě pádu
3. Konfigurace služby pro automatické zavedení
4. Spuštění programu pro LCD panel
5. Instalace programu dálkového ovládání do telefonu

## Instruktaž použití

1. Postupné vyzkoušení základních funkcí ovládání
2. Zkouška vypnutí a zapnutí zařízení serveru

### **6.1.3 Postup v případě pádu systému**

Pád systému může nastat z několika důvodů. V případě pádu na základě kritické chyby systém automaticky chybu zalogue a my pak můžeme chybu odladit. V případě pádu z důvodu přerušení dodávky elektrického proudu je nutné systém znovu zapnout. Tento postup spočívá v zapnutí serveru, který po inicializaci operačního systému automaticky spustí windows služby, které ovládají periferie a také aplikaci.

### **6.1.4 Technické aspekty nasazení**

Jak již bylo v popisu našeho řešení zmíněno, k fyzickému ovládání periferií používáme zařízení "Quido 8/8 ETH", které má limitované spínací napětí a to sice 120V střídavého napětí nebo 60V stejnosměrného. Proto jsme nemohli připojit světla přímo k výstupům Quida. Tento problém jsme vyřešili konstrukcí jednoduché desky se dvěma relé, viz. diplomová práce Bc. Patrik Dubec.

Samotnou instalaci tohoto zařízení a napojení světel prováděl elektrikář s mnohaletou praxí v oboru. Zařízení bylo umístěno v blízkosti vstupních dveří (z důvodu snadného přístupu k elektrickým jističům), kde jsme také umístili server s dotykovým ovládacím panelem.

Po připojení zařízení do wifi sítě, byl klient schopen ovládat server prostřednictvím mobilního telefonu z jakéhokoliv místa v domě.

## 6.1.5 Připomínky klienta

V průběhu testování aplikace bylo zjištěno několik chyb, které jsme postupně opravovali. Jednalo se zejména o pády windows služby, rychlost odezvy serveru a "chyby" v designu aplikace.

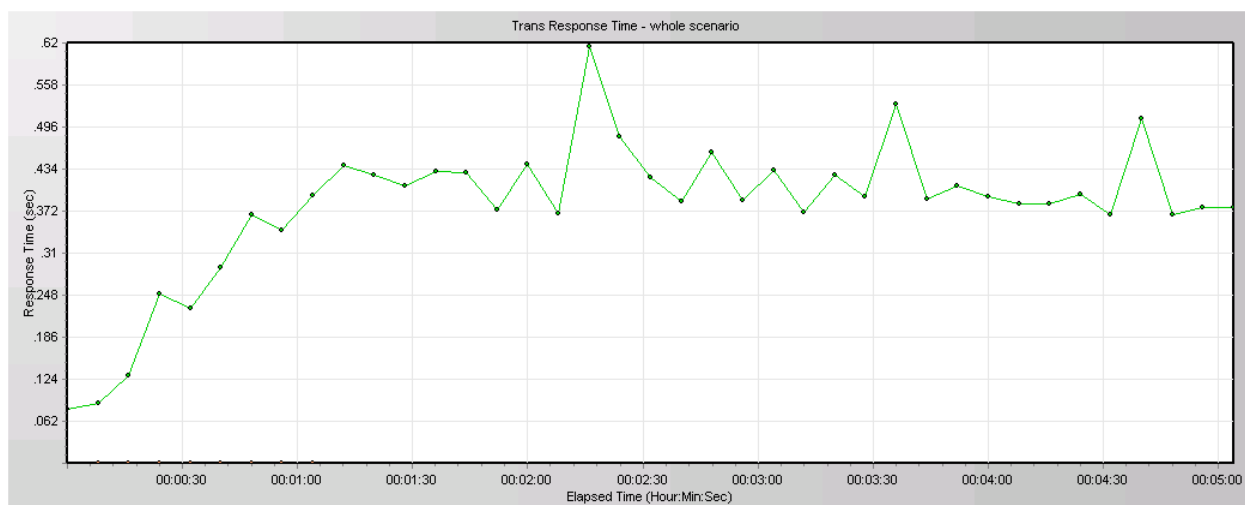
### Nejzávažnější chyba:

- Pomalá odezva serveru:

Původní verze systému využívala ke komunikaci se serverem webové služby, které komunikují se serverem pomocí robustního komunikačního protokolu. Ten výrazně zpomaluje komunikaci se serverem. Během testů jsme zjistili, že průměrná doba odezvy se pohybuje na hranici 3,5 sekund.

### Řešení:

Pro zrychlení komunikace jsme zaměnili technologii webových služeb za technologii WCF, která je o 25%-50% rychlejší (test výkonu: <http://msdn.microsoft.com/en-us/library/bb310550.aspx>). Provedli jsme znovu testování výkonu a vyhodnotili výsledky. Z obrázku 17 je patrné, že po dobu testování (5 hodin) byla maximální odměřená hodnota odezvy 0,62 sekundy.



Obr. 17 - test WCF služeb

WCF několikanásobně zvýšilo rychlost odezvy, nastal však jiný problém. Do této doby jsme využívali operační systém "Windows XP embedded", na který je možné nainstalovat IIS verze 5.1. Tato verze však dokáže hostovat pouze jednu WCF službu, což limitovalo návrh architektury naší aplikace.



Jako řešení jsme zvolili použití operačního systému "Windows embedded standard 7", který obsahuje následující pro nás klíčové technologie:

- .NET Framework 3.5. SP1
- Internet Information Services 7.5

Nároky na hardware operačního systému WES7:

- 1 GHz 32-bit nebo 64-bit
- 1 GB systémové paměti (32-bit)
- 7 GB místa na disku pro plnou instalaci
- DVD-ROM mechanika, USB 2.0.

## 7 Shrnutí výsledků nasazení a přínosů pro cílovou skupinu

Naše zařízení prošlo téměř dvou měsíčním testováním v domě klienta, a proto uvedu jeho hodnocení jako první:

Ráda bych se touto cestou podělila o své praktické zkušenosti s využíváním "ovládače domu". Musím nejprve podotknout, že moje životní situace není jednoduchá – sama jsem upoutána na invalidní vozík se značným pohybovým omezením a navíc se musím postarat o svoji starou maminku, která je po mozkové mrtvici. Žijeme v sociálním bytě ve dvoupatrovém rodinném domě, kde je pro nás invalidy značně ztížené prostředí. Bohužel prostředí sociálních bytů „zdravých invalidů“ je narušeno jednak technickými překážkami a jednak samotným asociálním chováním ostatních nájemníků, kteří pocházejí z problémových skupin obyvatelstva. Je třeba ještě zdůraznit, že můj svět znamená vlastně pohyb na vozíku po chodbě s chabým osvětlením a dále vlastní prostor bytu. Vzhledem ke skutečnosti, že jsme dvě ženy odkázané jen na sebe, je logické, že se také obáváme o naši bezpečnost, neboť máme dennodenní zkušenosti s drogovými závislými a různými živly. Pro mne jsou důležité i takové maličkosti, které se zdravému člověku zdají samozřejmé nebo malicherné a to je například možnost vědět, kdo skutečně zazvonil u dveří a zda může mít dobré či špatné úmysly. Přivítala jsem proto možnost využít „ovládač“ s tím, že mi umožní nahlédnout za dveře na chodbu a prozkoumat blízké okolí a připravit mne na cestu k výtahu. Jako neocenitelnou službu považuji možnost dálkového rozsvícení a zhasnutí světel v bytě, neboť se to netýká jen mne, jakožto nechodícího, ale také různých životních situací, kdy v průběhu noci nemusím vynakládat tolik úsilí při vstávání k nejbližšímu vypínači, abych zhasla za svoji matkou, která trpí nespavostí a sklerózou. Ačkoliv patřím mezi lidi již středního věku dnešní doba mne naučila přizpůsobit se, a tak vlastní obsluha mobilu s tímto dálkovým ovládáním mi nečinila po krátkém zaškolení větší problém. Určitě bych uvítala kdyby se dálkově daly otvírat i balkónové dveře. Určitou nevýhodou je cena. Samozřejmě jsem se ptala, zda bude toto zařízení prodejné, ale cena blížící se 60 000 Kč je pro mne bohužel příliš vysoká. Otázkou zůstává, zda by tento produkt nemohl být dotován, naprosto stejně, jako byl i můj elektrický vozík.

Marie Gřundělová

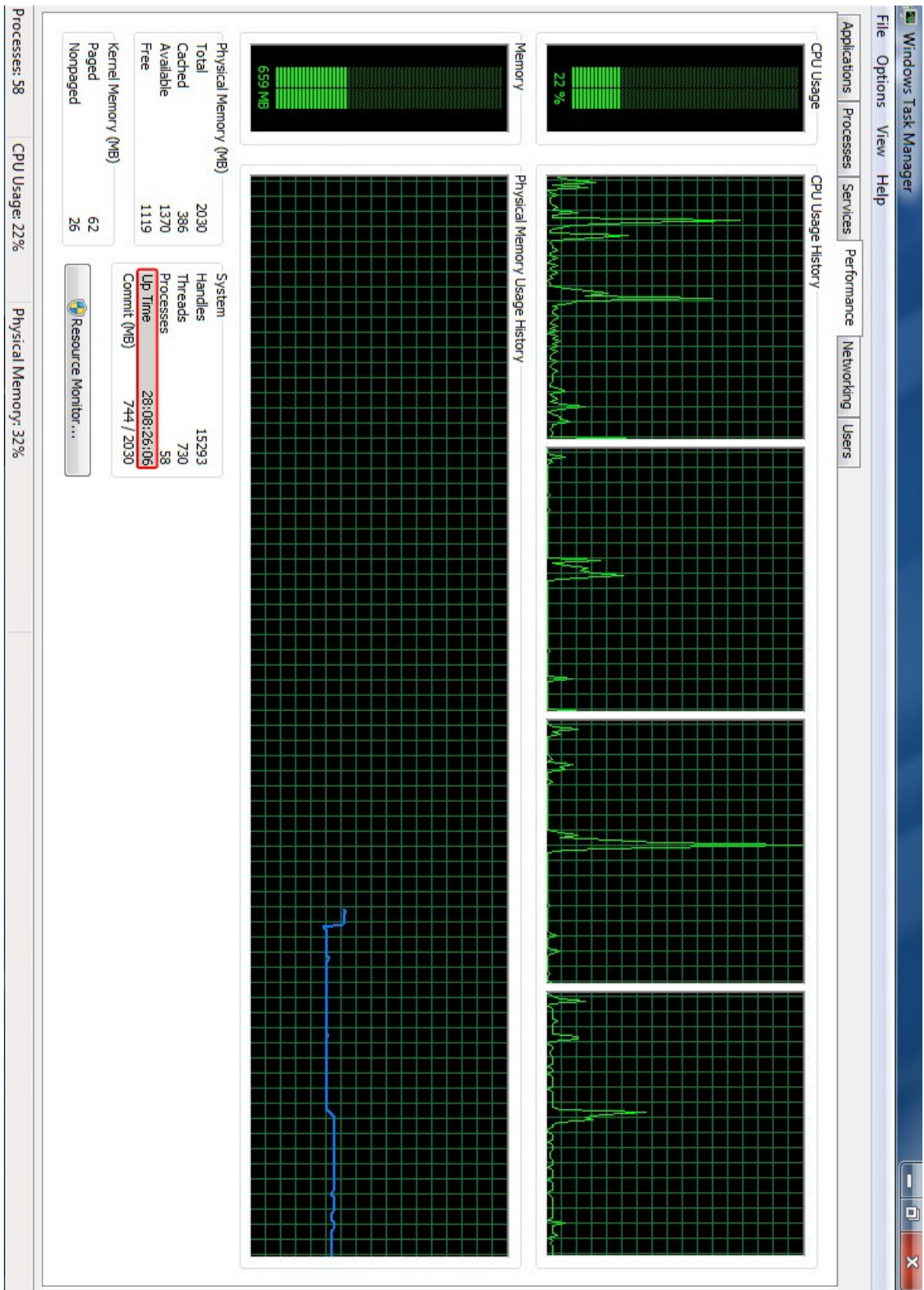
### **Naše reakce:**

Těší nás, že ohlas klienta a několika dalších jedinců byl pozitivní, zároveň však cítíme prostor k vylepšení projektu. Chápeme požadavek klienta na integraci elektrického otevírání dveří a plně s ním souhlasíme. Do našeho systému je možné připojit elektronický zámek již dnes, ovšem v testovacím řešení jsme jej nepoužili ze dvou důvodů:

- Cena: cena elektronického zámku se pohybuje v rozmezí 1200Kč-1500Kč a cena motorku, který by dveře otevíral v rozmezí 3000Kč-5000Kč.
- Nebezpečí selhání: samozřejmě jsme se obávali, že se v systému objeví chyba, která by vážně narušila chod celého systému. Díky poměrně velké dojezdové vzdálenosti by pak mohla nastat situace, kdy by klientka nebyla schopna zavřít dveře třeba i po několik hodin.

Poslední připomínku, na kterou budu reagovat, je připomínka o ceně, která je pro klienta příliš vysoká. Obáváme se, že pokud by nebyla zavedena výroba v mnohatisícových objemech, tak by se cena příliš nesnížila. Zároveň je však nutné podotknout, že v porovnání s jinými řešeními na trhu je naše cena velmi nízká. V kapitole 7.1 Vám pro srovnání představím dvě zajímavá řešení inteligentních domů i s odhadem ceny zavedení do domu.

Na obrázku 18 je vyfocen task manager s ukázkou hardwarových nároků po 28 dnech stálého provozu.



Obr. 18 - chod serveru

## 7.1 Jiná řešení na trhu v porovnání s naším řešením

Na českém trhu je k dispozici několik řešení inteligentních domácností, ale jen některé z nich jsou dostatečně veřejně popsány. Téměř vždy je jak technologie, tak cena řešení před možným klientem skryta až do doby, kdy zástupce dané společnosti vytvoří klientovi finanční analýzu.

### 7.1.1 Insight home

Insight home neboli InHome je řešení pro moderní domy a byty, které vyvinula společnost Insight Home a.s. InHome nabízí kompletní řešení domu pro pohodlný a bezpečný životní styl. Celé řešení bylo měsíce testováno v areálu CITIB, který zahrnuje dva rodinné domy, zahradní domek a pozemek o rozloze 4.000 m<sup>2</sup>. Tento areál mimo jiné plní i funkci vzorových řešení pro možné klienty. Na obrázku 19 je zobrazeno rozhraní pro snímání videa a fotografií z venkovních kamer a na dalším snímku je pak zobrazena hlavní nabídka programu.



Obr. 19 - kamerový systém



Obr.20 - hlavní menu

#### Klíčové vlastnosti řešení:

- **Modulární řešení:** Insight home potřebuje k svému provozu základní infrastrukturu v podobě kabelové sítě. Jakmile je tato síť do domu integrována, je možné celý systém stavět modulárně.
- **Centrální systém:** Celý systém je řízený z jednoho centrálního místa. V základní verzi je k tomuto řízení použita televize a dálkové ovládání a v rozšířené verzi je k dispozici
- **Infrastruktura:** klíčem je strukturovaná kabeláž, rozvody videosignálu, audio signálu, telefonů, počítačová síť, WiFi, stínění, přepěťová ochrana, napojení na satelit a internet, zálohování.
- **Zabezpečení:** systém do svého řešení integruje i sledování alarmu, pohybových čidel a teplotních čidel.
- **Osvětlení:** systém integruje do svého řešení i ovládání světel v celém domě.

#### Zhodnocení řešení:

Velkou výhodou je doba instalace, které se pohybuje v rozmezí dvou až tří dnů. Tato doba samozřejmě záleží na době integrace, kdy za nejlepší případ se pokládá integrace kabelových rozvodů již v průběhu výstavby domu. Mezi nevýhody pak patří samotné ovládání domu. Abychom mohli jednotlivá zařízení ovládat, musíme zapnout televizor s centrálním ovládacím rozhraním, nebo přijít ke konzoli vestavěné do zdi.

Nespornou nevýhodou je také cena, která začíná u základního modelu na 400 000Kč.

## 7.1.2 Haidy

Toto řešení se zaměřuje hlavně na integraci systému do novostaveb. Díky tomu, že se jedná o stavby na klíč je systém vytvořen přesně podle potřeb každého domu.



Obr. 21 - půdorys domu

**Komunikace mezi periferiemi:** pro komunikaci se využívá GSM bran.

- **Modulární řešení:** ač systém nabízí modularitu řešení, je nutné počítat s omezeným počtem zařízení, kterých je možné do systému integrovat. Omezení je dáno hlavně počtem digitálních relé.
- **Zabezpečení:** systém klade důraz hlavně na pohybová čidla a kamerový systém.

**Zhodnocení řešení:**

Výhodu tohoto řešení představuje snadné ovládání domu pomocí dálkového ovládání, které se přímo připojuje k serveru a dává tak uživateli možnost ovládat periferie z jakékoliv části domu. Asi největší výhodou je snadné doplnění řešení o nová zařízení, která se připojují k serveru bezdrátově. Zařízení však musí být k této komunikaci uzpůsobena. Mezi nevýhody systému patří opět cena, která je zobrazena v tabulce Haidy Basic. K cenám v tabulkách je nutné dodat, že se dělí na dvě kategorie:

- "Cena celkem Na Klíč": tato cena je platná pro ty, kteří si objednávají projekt Haidy jako bonusový balíček k jednomu z domů na klíč od stavební firmy GSERVIS.
- "Cena celkem": cena pro klasického koncového uživatele.

Haidy Basic	
Světla	9
Elektronický zámek	1
Topení	7
Detektory pohybu	9
Cena celkem Na Klíč	94 141 Kč
Cena celkem	229 141 Kč

Haidy Optimal	
Světla	25
Elektronický zámek	1
Elektronická zařízení a topení	42
Detektory pohybu	24
Cena celkem Na Klíč	196 669 Kč
Cena celkem	648 669 Kč

Jako dodatek k rozpočtu "Haidy Optimal" dodávám podrobný rozpis komponent i s vyúčtováním. Tato cenová nabídka byla převzata z ceníku vydaného v lednu 2011.

HW Haidy				210142,-
Jenoduchý vypínač	SS2B-C-S-2	3	1256,-	3769,-
Dvojitý vypínač	SS4B-C-S-2	42	1326,-	55713,-
Převodník Ethernet/ CAN	ECC-E-D	4	2145,-	8582,-
Zdroj 12V 2A DIN		1	1000,-	1000,-
Zdroj 24V 6A		2	800,-	1600,-
Modul vstupů a výstupů 2042	IOM2042-C-D-2	10	1064,-	10640,-
Stmívač 2x10W 12V	DIM2X10W12-C-D-2	2	1526,-	3052,-
Stmívač 6x100W	DIM6X100W-C-D6-2	2	7745,-	15491,-
Reléový spínač 250V 6x6A	P6R6A-C-D2-2	5	2201,-	11006,-
Reléový spínač 250V 6x16A	P6R16A-C-D6-2	4	1949,-	7798,-
Ovladač motoru žaluzii 4x6A	M4R6A-C-D2	7	2968,-	20776,-
Senzorový modul	SMF-C-S	1	5796,-	5796,-
Univerzální modul	UR01-C-S-2	1	1354,-	1354,-
Koncentrátor zabezpečovačky	SDC18-C-D2	3	2688,-	8064,-
Zapojení, konfigurace, oživení a testy Haidy		222	250,-	55500,-

Obr. 22 - podrobná kalkulace



<b>HW</b>					<b>54900,-</b>
GSM brána	GSM01	1	3500,-	3500,-	
Server Haidy + Modul základních funkcí + Haidy klient (1licence)	SHPMK1	1	30000,-	30000,-	
Rack 19" 22 100x60x60	RU19/22	1	8400,-	8400,-	
Switch 24/100 port rack	SW24/100R2	1	1800,-	1800,-	
Záložní zdroj pro Haidy	TR1500VA2U	1	11200,-	11200,-	
<b>SW</b>					<b>39950,-</b>
Modul vytápění a klimatizace	MVK1	1	9990,-	9990,-	
Modul zabezpečení	MZ1	1	9990,-	9990,-	
Modul GSM	MGSM1	1	4990,-	4990,-	
Modul Interkom	MI1	1	4990,-	4990,-	
Haidy klient SW licence	HKL1	1	9990,-	9990,-	
<b>Koncové prvky podporované Haidy</b>					<b>28800,-</b>
IP Videovratny 2N		1	16800,-	16800,-	
Dotykový panel 10"		1	12000,-	12000,-	
<b>Ostatní</b>					<b>314876,-</b>
Projektové práce - Elektro+Haidy		1	40621,-	40621,-	
Elektroinstalace (material + práce) - kvalifikovaný odhad		1	274255,-	274255,-	
<b>Cena celkem</b>					<b>648669,-</b>

Obr. 22 - podrobná kalkulace 1

## 8 Závěr

Projekt byl velmi zajímavý díky své návaznosti na oblast fyzických postižení. Již ve fázi analýzy jsem se setkal s mnoha novými fakty. Přínosné bylo zejména studium speciálních pouček týkajících se vývoje softwaru pro fyzicky postižené osoby a to jak po stránce grafické, tak i po stránce programátorské.

Hlavní přínos práce tkví v implementaci a otestování řešení dálkově ovládaného domu, který splňuje konkrétní požadavky reálných osob. Z projektu vyplývá, že na trhu sice existuje konkurence, jejíž řešení jsou komplexnější, ale na druhou stranu nejsou vyhovující pro danou skupinu. Naše řešení nabízí pokrytí všech zásadních potřeb za částky pohybující se v hladině 20% minimální ceny jiných řešení. To je samozřejmě způsobeno tím, že v našem projektu jsou upřednostněny funkce klíčových zařízení.

Dále bych rád vyzdvihнул rozšiřitelnost řešení. Díky použití dobře zavedených praktik návrhu systémů je naše řešení snadno rozšiřitelné a pro nezasvěceného programátora dobře čitelné (programátora se znalostí návrhu systému, návrhových vzorů, znalosti čtení diagramů).

Líbí se mi také hardwarové řešení, které je jednoduché a umožňuje připojení velkého množství zařízení. Samotná komunikace probíhá v síti Wifi, která je dnes běžně používaná v domácnostech.

Mezi nevýhody bych uvedl úzkoprofilový pohled na testování. Testování probíhalo u klientky s fyzickým postižením pohybového ústrojí, a proto byly všechny zařízení uzpůsobeny tomuto faktu. Je pravda, že pro matku klientky byl na stroji nainstalován čtecí software a nový styl výpisu prvků. To však nijak nezbudilo otázky ohledně dalších možností integrace hardwaru.

1. Davies , K.H., Biddulph, R. and Balashek, S. (1952) Automatic Speech Recognition of Spoken Digits, J. Acoust. Soc. Am. 24(6) pp.637 – 642
2. Review of Neural Networks for Speech Recognition, Richard P. Lippmann, MIT Lincoln Laboratory, Lexington, MA 02173, USA
3. Jenny Preece, editor. Human-computer interaction, chapter 11.5 and 11.6, pages 224 - 229. Addison Wesley, 1994. isbn: 0-201-62769-8.
4. [http://www.zmms.tu-berlin.de/de/veranstaltungen/eyes\\_tea/Clarke.pdf](http://www.zmms.tu-berlin.de/de/veranstaltungen/eyes_tea/Clarke.pdf)
5. <http://www.namahn.com/resources/documents/note-eyetracking.pdf>
6. <http://www.freedomsscientific.com/products/fs/jaws-product-page.asp>
7. CUED SPEECH HAND GESTURES RECOGNITION TOOL, Thomas Burger, Alice Caplier and Stéphane Mancini
8. <http://tim.timginn.com/wp-content/uploads/2007/12/497-long-paper.pdf>
9. Kalloniatis, Michael and Luu, Charles. "Psychophysics of Vision: The Perception of Color". Retrieved 2007-04-02.
10. <http://www.insighthome.eu/CITIB.html>
11. Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Software Development Process. s.l. : Addison-Wesley Profesional, 1999.
12. Jim Arlow, Ila Neustadt. UML a unifikovaný proces vývoje aplikací. s.l. : Computer Press, 2003.
13. Interval. [Online] <http://interval.cz/clanky/navrh-aplikaci-v-jazyce-uml-unified-modeling-language/>.
14. Oficiální fórum. [Online] [www.silverlight.net](http://www.silverlight.net).
15. Scott Guthrie's blog. [Online] <http://weblogs.asp.net/scottgu/>.
16. Wynapse. [Online] [http://www.wynapse.com/Silverlight\\_Tutorials.aspx](http://www.wynapse.com/Silverlight_Tutorials.aspx).
17. Microsoft library. [Online] <http://msdn.microsoft.com/en-us/library/ms752059.aspx>.
18. Oficiální stránka ASP.NET. [Online] <http://www.asp.net>.
19. Geek blog. [Online] <http://www.misfitgeek.com/> .