

# **Ovládání a kontrola domácnosti přes Internet**

Controlling and Supervising of a Household via the Internet

Bc. Jiří Horák

---

Diplomová práce  
2012



Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

akademický rok: 2011/2012

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jiří HORÁK**  
Osobní číslo: **A10378**  
Studijní program: **N 3902 Inženýrská informatika**  
Studijní obor: **Informační technologie**

Téma práce: **Ovládání a kontrola domácnosti přes Internet**

Zásady pro vypracování:

1. Zpracujte přehled současného stavu řešené problematiky.
2. Navrhněte architekturu propojení mikrokontroléru se senzory a rozhraním připojeným do sítě Internet.
3. Naprogramujte mikrokontroler.
4. Naprogramujte webový server v síti Internet na platformě Microsoft .NET Framework v jazyce C Sharp, kde se výsledky z mikrokontroléru budou ukládat do databáze.
5. Webový server bude zobrazovat uživateli data ve formě tabulek, grafů a stavů. Přes webové rozhraní bude možné nastavovat notifikace, měnit nastavení a posílat povely mikrokontroléru.
6. Otestujte výsledné řešení v reálném prostředí domácnosti a proveďte závěrečné zhodnocení.

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **NAGEL, Christian at al. Professional C 4 and .Net 4. Indianapolis, IN: Wiley Publishing, 2010, ISBN 978-0-470-50225-9.**
2. **REYNOLDS-HAERTLE, Robin A. OOP - objektově orientované programování: Visual Basic .NET, Visual C .NET krok za krokem. Vyd. 1. Přeložil RNDr. Jan POKORNÝ. Knihy iDNES, 2002. ISBN 80-86953-25-8.**
3. **BANZI, Massimo. Getting started with Arduino. 1st ed. Make: OReilly, 2009. ISBN 978-0-596-15551-3.**
4. **HENDL, Jan. Přehled statistických metod zpracování dat: analýza a metaanalýza dat. 1. vyd. Praha: Portál, 2004, 583 s. ISBN 80-717-8820-1.**
5. **AGARWAL, Vidya Vrat a James HUDDLESTON. Databáze v C 2008: průvodce programátora. Vyd. 1. Přeložil Lukáš KREJČÍ. Brno: Computer Press, 2009, 424 s. ISBN 978-802-5123-096.**

Vedoucí diplomové práce:

**Ing. Bc. Pavel Vařacha, Ph.D.**

Ústav informatiky a umělé inteligence

Datum zadání diplomové práce:

**24. února 2012**

Termín odevzdání diplomové práce:

**21. května 2012**

Ve Zlíně dne 24. února 2012

prof. Ing. Vladimír Vašek, CSc.  
*děkan*



doc. Mgr. Roman Jašek, Ph.D.  
*ředitel ústavu*

## **ABSTRAKT**

Diplomová práce se zabývá propojením mikrokontroléru se senzory, rozhraním připojeným do sítě Internet a webového serveru v síti Internet. Mikrokontrolér umí reagovat na události a vyvolat odpovídající akce. Vybrané senzory jsou vhodné především pro použití v domácnosti. Webový server v síti Internet získává z mikrokontroléru hodnoty a ukládá je do databáze. Každý uživatel má přístup jen k jeho registrovaným mikrokontrolérům. Na webovém serveru se data zobrazují uživateli ve formě tabulek a grafů. Přes webové rozhraní je možné nastavovat notifikace, měnit nastavení a posílat povely mikrokontroléru. Webový server lze nastavit pro různé další senzory připojené k mikrokontroléru.

Klíčová slova:

Arduino, mikrokontrolér, Atmel, webový server, C#, .NET, MS SQL, databáze, Internet

## **ABSTRACT**

This diploma work deals with the interconnection of sensors and a microcontroller and with an interface to the internet through a web server. Many microcontrollers can be connected to a single web server. Microcontrollers can react to events and invoke appropriate actions. Selected sensors are especially suitable for the home usage.

A web server, which is connected to the internet, receives from microcontrollers values and stores them in a database. Each user has access only to its registered microcontrollers. The web server can then show data for a user in the form of tables or graphs. Through a web interface, a user can set required notifications or he can change the settings to send commands directly to selected microcontroller. The web server can be adjusted for various other sensors connected to the microcontroller.

Keywords:

Arduino, microcontroller, Atmel, Web Server, C #, .NET, MS SQL, database, Internet

Poděkování

Rád bych poděkoval Ing. Bc. Pavlovi Vařachovi, Ph.D., za užitečné rady, konzultace a vstřícnost.

**Prohlašuji, že**

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

**Prohlašuji,**

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....  
podpis diplomanta

**OBSAH**

<b>ÚVOD</b> .....	<b>10</b>
<b>I TEORETICKÁ ČÁST</b> .....	<b>11</b>
<b>1 INTELIGENTNÍ DOMÁCNOST</b> .....	<b>12</b>
1.1 ROZDĚLENÍ INTELIGENTNÍCH DOMÁCNOSTÍ .....	12
1.2 SOUČASNÝ STAV .....	13
1.2.1 Vytápění .....	13
1.2.2 Ventilace .....	13
1.2.3 Klimatizace .....	13
1.2.4 Stínění .....	13
1.2.5 Ohřev teplé vody .....	13
1.2.6 Osvětlení .....	14
1.2.7 Provoz spotřebičů.....	14
1.3 INTERNET V DOMÁCNOSTECH.....	14
1.4 SBĚRNICOVÉ SYSTÉMY PRO INTELIGENTNÍ BUDOVY .....	15
1.4.1 KNX/EIB.....	15
1.4.2 LON.....	15
1.4.3 Luxmate.....	16
1.4.4 ZigBee .....	16
1.5 NÁVRH ARCHITEKTURY.....	16
<b>2 MIKROKONTROLÉR</b> .....	<b>19</b>
2.1 ARDUINO.....	19
2.1.1 Digitální vstup/výstup .....	21
2.1.2 Analogový vstup .....	22
2.1.3 Analogový výstup .....	22
2.2 ETHERNETOVÁ KARTA .....	22
2.3 SENZORY .....	24
2.3.1 Teplota.....	24
2.3.2 Vlhkoměr.....	26
2.3.3 Intenzita osvětlení .....	28
2.3.4 Otevřené/zavřené dveře nebo okno .....	29
2.4 AKČNÍ ČLENY .....	30
2.4.1 Elektromagnetické relé.....	30
2.4.2 Servomotor .....	31
2.5 ZOBRAZOVACÍ PRVKY .....	32
2.5.1 Displej .....	32
2.5.2 LED diody .....	34
<b>3 WEBOVÝ SERVER</b> .....	<b>35</b>
3.1 ZÁKLADNÍ MOŽNOSTI SYSTÉMU .....	35
3.2 PŘÍSTUP K SYSTÉMU .....	35
3.2.1 Nepřihlášený uživatel.....	35
3.2.2 Přihlášený uživatel .....	35
3.2.3 Administrátor .....	35

3.3	USE CASE DIAGRAM .....	36
3.4	ER DIAGRAM.....	37
3.4.1	Databázová tabulka Uzivatele.....	37
3.4.2	Databázová tabulka Arduino.....	38
3.4.3	Databázová tabulka alarm_values.....	38
3.4.4	Databázová tabulka Alert_values_user_definition .....	38
3.4.5	Databázová tabulka alarm_values_user .....	38
3.4.6	Databázová tabulka Arduino_command.....	38
3.4.7	Databázová tabulka measurement_values.....	38
3.5	DIAGRAM TŘÍD.....	39
3.6	WEBOVÉ TECHNOLOGIE .....	43
3.6.1	HTML .....	43
3.6.2	CSS.....	43
3.6.3	JavaScript .....	43
3.6.4	ASP.NET.....	43
3.6.5	AJAX.....	43
3.6.6	AJAX Extensions .....	44
3.6.7	XML.....	44
<b>II</b>	<b>PRAKTICKÁ ČÁST .....</b>	<b>45</b>
<b>4</b>	<b>IMPLEMENTACE .....</b>	<b>46</b>
4.1	PROSTŘEDÍ IMPLEMENTACE.....	46
4.1.1	Visual Studio .NET .....	46
4.1.2	.NET Framework architektura.....	47
4.1.3	Microsoft SQL server.....	47
4.1.4	Microsoft Windows server 2008 R2 .....	47
4.1.5	Vývojové prostředí Arduina.....	47
4.2	ARDUNINO .....	48
4.2.1	Ethernetová karta .....	48
4.2.2	Teplota.....	48
4.2.3	Vlhkoměr.....	49
4.2.4	Intenzita osvětlení .....	49
4.2.5	Otevřené/zavřené dveře nebo okno .....	49
4.2.6	Displej .....	50
4.3	UŽIVATELSKÉ ROZHRANÍ .....	50
4.3.1	Přihlášení.....	50
4.3.2	Registrace .....	51
4.3.3	Home .....	51
4.3.4	Uživatelský profil.....	51
4.3.5	Domov přehled.....	52
4.3.6	Přehled varování.....	54
4.3.7	Domov nastavení.....	55
4.3.8	O aplikaci .....	55
4.3.9	Administrace .....	55
4.4	ROZHRANÍ PRO UKLÁDÁNÍ HODNOT NA WEBOVÉM SERVERU.....	56
4.4.1	Periodicky posílané hodnoty .....	56
4.4.2	Hodnoty typu alarm.....	57

4.5	INSTALACE A KONFIGURACE .....	57
4.5.1	Arduino .....	57
4.5.2	Web server .....	57
<b>5</b>	<b>KOMUNIKACE ARDUINA A WEBOVÉHO SERVERU .....</b>	<b>59</b>
5.1	ARDUINO SMĚREM NA WEBOVÝ SERVER.....	59
5.1.1	Periodické posílání aktuálního stavu.....	59
5.1.2	Posílání stavu alarm .....	60
5.2	WEBOVÝ SERVER SMĚREM K ARDUINU .....	60
<b>6</b>	<b>BEZPEČNOST .....</b>	<b>62</b>
6.1.1	Arduino .....	62
6.1.2	Webové rozhraní .....	62
6.1.3	Databáze .....	62
6.1.4	Ukládání hesel .....	62
<b>7</b>	<b>TESTOVÁNÍ .....</b>	<b>64</b>
7.1	TESTOVÁNÍ MIKROKONTROLÉRU .....	64
7.2	TESTOVÁNÍ WEB SERVERU.....	66
<b>8</b>	<b>MOŽNÉ ROZŠÍŘENÍ .....</b>	<b>68</b>
8.1	VYTVOŘENÍ APLIKACE PRO CHYTRÝ MOBILNÍ TELEFON .....	68
8.2	VYTVOŘIT ANALÝZU A IMPLEMENTOVAT BEZPEČNOU KOMUNIKACI PŘES INTERNET .....	68
8.3	VYTVOŘIT ANALÝZU PRO MOŽNOST KOMUNIKACE ARDUINO A DALŠÍ ARDUINO (PEER-TO-PEER).....	68
8.4	NASTAVOVÁNÍ HODNOT ALARM NA STRANĚ ARDUINA .....	69
8.5	VYTVOŘIT STRÁNKU S NASTAVENÍM NA ARDUINU .....	69
	<b>ZÁVĚR .....</b>	<b>70</b>
	<b>ZÁVĚR V ANGLIČTINĚ.....</b>	<b>72</b>
	<b>SEZNAM POUŽITÉ LITERATURY.....</b>	<b>74</b>
	<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....</b>	<b>76</b>
	<b>SEZNAM OBRÁZKŮ .....</b>	<b>78</b>
	<b>SEZNAM TABULEK.....</b>	<b>79</b>
	<b>SEZNAM PŘÍLOH.....</b>	<b>80</b>

## ÚVOD

S rychlostí, jakou se dnes rozšiřuje Internet, mobilní Internet a zařízení označovaná jako „chytré“ telefony nebo tablety, je možné mít domácnost neustále pod kontrolou, i když právě nejsme doma. Nemáme-li dostupný chytrý telefon či Internet v mobilu, lze použít pro dohled nad domácností počítač v internetové kavárně, počítač v zaměstnání či u známých nebo jakékoliv jiné zařízení s dostupným připojením k Internetu. Pokud ani to není dostupné, lze ovládání a komunikaci s domácností zajistit prostřednictvím služby SMS.

Pro řízení a zjišťování stavu domácnosti se nejčastěji užívají jednočipové počítače, protože jsou dostatečně výkonné, cenově přijatelné a mají minimální spotřebu energie. Nejrozumnější základní senzory vhodné pro domácnost jsou cenově dostupné, protože jsou používány téměř v každém elektronickém zařízení. Toho lze využít k sestavení cenově dostupného systému, který bude kontrolovat a upravovat chod domácnosti. Nastane-li nějaká situace, která je v rozporu s přednastaveným stavem, systém na to neprodleně upozorní a umožní uživateli odpovídající reakci. Systém například může upozornit na situaci, když někdo otevře vchodové dveře nebo okno. Uživatel může dále zjistit, jakou teplotu má v bytě a může ji vzdáleně upravit. Dále je možné zkontrolovat, zda jsou vypnuta všechna světla nebo takto lze zapnout nebo vypnout nějaký spotřebič, který je připojený k mikrokontroléru. Tím máme na mysli například přepnutí topení z režimu temperování na režim vytápění, vrátí-li se rodina z dovolené nebo víkendu dříve, než plánovala apod. Zvýší se tak uživatelský komfort a zároveň se šetří energie. Takovouto domácnost pak můžeme nazvat inteligentní domácností.

Cílem této diplomové práce je popsat a objasnit systém inteligentní domácnosti pro praktické použití. Jelikož je oblast senzorů velice rozsáhlá, nemůžeme prozkoumat všechny existující senzory, a tak jsou v práci popsány vybrané dostupné a užitečné senzory. Dále je cílem práce seznámit se s vhodným využitím možností paměťově a výkonově omezeného jednočipového počítače a serveru, kam budou data ukládána. Přístup k serveru je umožněn prostřednictvím Internetu pomocí tenkého klienta. Uživatel si tak bude moci nastavovat systém ovládání domácnosti a zobrazovat přehled událostí přes libovolný internetový prohlížeč.

Pro využití v domácnosti byly použity digitální a analogové senzory - teploměr, vlhkoměr, senzor otevřených dveří a senzor světla.

## **I. TEORETICKÁ ČÁST**

## 1 INTELIGENTNÍ DOMÁCNOST

Domácnost, kterou lze kontrolovat a ovládat přes Internet a v níž se vyhodnocuje a rozhoduje dění na základě získaných a přednastavených vlastností, můžeme nazvat jako inteligentní domácnost. Valeš definuje inteligentní dům nebo chytrou domácnost jako: „*Inteligentní dům v nejširším možném smyslu slova je budova vybavená počítačovou a komunikační technikou, která předvídá a reaguje na potřeby obyvatel s cílem zvýšit jejich komfort, pohodlí, snížit spotřebu energií a poskytnout jim bezpečí a zábavu pomocí řízení všech technologií v domě a jejich interakcí s vnějším světem.*“ [1, s. 1]

### 1.1 Rozdělení inteligentních domácností

Richard Harper (2003) uvádí rozdělení inteligentních domácností následovně [2]:

#### 1. Domy se samostatnými inteligentními zařízeními

Domy mající inteligentní zařízení fungující nezávisle na ostatních inteligentních zařízeních.

#### 2. Domy obsahující inteligentní a komunikující zařízení

Domy obsahující inteligentní zařízení umožňující vzájemnou komunikaci a výměnu informací.

#### 3. Připojený dům

Dům má interní a externí síťové připojení, které umožňuje dálkové ovládání a přístup k službám a informacím, a to z domova nebo mimo domov.

#### 4. Učící se dům

Dům má systém zaznamenávající chování jeho obyvatel. Ze zaznamenaného chování jsou vytvořeny vzory, které jsou následně použity v nastavení inteligentního zařízení k předvídání potřeb obyvatel.

#### 5. Aktivní a osoby lokalizující dům

Dům má systém sledující činnosti a pozice lidí, které jsou v domě neustále registrovány, a to v reálném čase. Tyto informace se použijí pro řízení v technologiích při předvídání požadavků uživatelů.

## 1.2 Současný stav

V současné době se staví domy, které spadají do kategorie 1-3. Vyšší kategorie typů domů jsou ve stádiu experimentálních projektů. Prozatím jsou zkoumány na univerzitách nebo testovány v laboratorních podmínkách [1].

### 1.2.1 Vytápění

Dnes je běžné, že je v bytě jeden termostat a podle něj se řídí vytápění v ostatních místnostech. Je komfortnější a úspornější regulovat vytápění v každé místnosti zvlášť. Pokud chceme zavést inteligentní vytápění, tak se musí měřit v každé místnosti teplota a na radiátor použít elektricky ovládanou hlavici. Je nutné mít vhodný topný systém, který zvládá vytápět místnosti i jednotlivě. Systém lze doplnit i o detektor přítomnosti osob [1].

### 1.2.2 Ventilace

Optimální výměnu vzduchu lze zajistit rozbořem vzduchu pomocí CO<sub>2</sub> snímačů, měřičů vlhkosti nebo snímačů pachu. Je možné využít rekuperaci, kdy se vzduch přiváděný do budovy předehřívá odpadním vzduchem a nedochází tak k velkým energetickým ztrátám [1].

### 1.2.3 Klimatizace

Strategie v dosažení maximálních úspor je podobná jako v kapitole 1.2.1, kde jsme se věnovali vytápění. Navíc je možné blokovat klimatizaci při otevřeném oknu a pro úsporu energií se doporučují další stavební úpravy [1].

### 1.2.4 Stínění

K ochlazování nemusíme používat pouze energeticky náročnou klimatizaci, ale osluněná okna můžeme chránit před nežádoucími tepelnými zisky pomocí clonícího zařízení. Regulace je důležitá i z hlediska vnitřního osvětlení. Existují žaluzie, u kterých se dá natočení lamel řídit s přesností jednoho stupně. V noci naopak žaluzie snižují tepelné ztráty, protože slouží jako další tepelně-izolační vrstva [1].

### 1.2.5 Ohřev teplé vody

K ohřevu teplé vody je možné využít solárních kolektorů, které v létě mohou pokrýt až 100% potřeby energií. V celoročním průměru lze využít solární kolektory z 50 - 60 %. Jestliže pro ohřev teplé vody nestačí solární kolektor nebo jiný alternativní ohřev, pak

docílíme uživatelského komfortu a optimální spotřeby energie prostřednictvím automatického zapínání ohřevu vody s využitím elektřiny nebo zemního plynu. Dalšího snížení spotřeby teplé i studené vody můžeme docílit automatickým řízením teploty ve vodovodní baterii, kde by při ručním seřizování uteklo zbytečně mnoho vody [1].

### **1.2.6 Osvětlení**

Je pohodlné a zároveň úsporné zapínat osvětlení pouze v době, kdy se v místnosti nacházejí lidé. Jsou místnosti jako například předsíň, kde se lidé příliš nezdržují. Pomocí detektorů pohybu nebo osob můžeme osvětlení zapínat jen tehdy, kdy je to nutné. Je-li dostatek přirozeného osvětlení, lze umělé osvětlení vypnout. Pokud tomu tak není, je možno umělé osvětlení částečně zapnout, nebo v případě nedostatečného přirozeného osvětlení zapnout úplně.

### **1.2.7 Provoz spotřebičů**

Pokud obyvatelé opustí místnost, systém může některé spotřebiče vypnout, a to automaticky podle detektoru lidí, nebo když obyvatel vypne veškeré osvětlení v místnosti – např. televizi. Po vstupu do místnosti musí být zajištěno, že se televize rychle zapne se stejným nastavením [1].

Systém může některé spotřebiče zapínat až tehdy, je-li dostupný levnější proud – např. pračku nebo akumulaci kamna a brát ohled na další parametry (roční období u kamen).

## **1.3 Internet v domácnostech**

Jednou z dalších možností, jak využít již zavedené připojení k Internetu do domácnosti, je kontrola a ovládání inteligentního domu nebo domácnosti přes síť Internet.

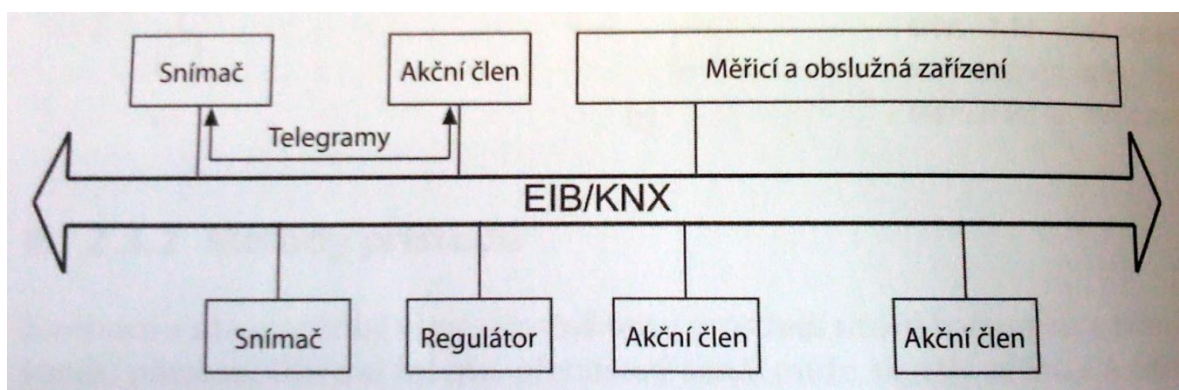
Podle předběžných dostupných výsledků Sčítání lidu, domů a bytů v roce 2011 je v České republice celkem 3 894 210 obydlených bytů. Z toho je vybaveno osobním počítačem s Internetem 2 204 487 bytů, což je téměř 57 % bytů [3]. Lze předpokládat, že připojená domácnost k Internetu bez počítače bude tvořit jen promile z průzkumu (služby typu IP TV, VOIP apod.). U inteligentních domů, kde má systém na starosti i zabezpečení, lze předpokládat také záložní přístup k Internetu, a to například pomocí mobilního připojení.

## 1.4 Sběrníkové systémy pro inteligentní budovy

### 1.4.1 KNX/EIB

Základ systému EIB (European Installation Bus) začal vznikat v osmdesátých letech minulého století v Německu u firmy Siemens. Začalo se projevovat, že klasická elektroinstalace nemůže splnit požadavky na komfort, ovládání a řízení moderního domu s hospodárným využitím energií. Při vývoji vývojový tým společnosti Siemens přišel s myšlenkou standardizované datové komunikace jednotlivých prvků tak, aby spolu mohly komunikovat produkty různých výrobců. Dále se ukázalo, že centralizované systémy nejsou schopny odpovídajícím způsobem pokrýt celou budovu, proto jsou další systémy decentralizované. V roce 2003 byl definován standard KNX (konnex), který zastřešil EIB, francouzský Batibus a Evropskou normovanou komunikaci pro domácí spotřebiče EHS [4].

KNX/EIB je sběrníkový systém s decentralizovanými inteligentními prvky, které umožňují ovládání mnoha komponent domu. Na sběrnici jsou připojena jednotlivá zařízení vyměňující si mezi sebou zprávy. Každá zpráva obsahuje informaci o úkonu a adresu zařízení, které má daný úkon vykonat.



Obr. 1. Informatické zasíťování sběrníky KNX/EIB [5]

### 1.4.2 LON

S vývojem různých systémů začalo vznikat množství sběrníkových systémů využívaných v průmyslové automatizaci, v palubní technice automobilů, lodí a letadel nebo v budovách. Aby jednotlivé nové vývojové týmy nemusely řešit vývoj nového sběrníkového systému a problémy s tím spojené, byla založena v Kalifornii v roce 1986 firma ACM. Později byla přejmenována na Echelon a roku 1991 Local Operating Network. Tato sběrníková

technologie se měla stát univerzálním řešením pro decentralizované sběrníkové systémy. V začátcích vedla volnost systému k vzájemně nekompatibilním řešením, a tak v roce 1994 vzniklo standardizační uskupení [4].

### 1.4.3 Luxmate

Jedná se především o systém pro řízení osvětlení od rakouské firmy Zumtobel. Firma má tři úrovně systému [6]:

LUXMATE Basic umožňuje základní využití modulů, které jsou použity samostatně a slouží pro řízení světelných prvků. Není zde možnost propojovat moduly se sběrníci.

LUXMATE Emotion obsahuje řídicí jednotku obsaženou v dotekovém audiovizuálním panelu. Ta umožňuje dva samostatné DALI (digital addressable lighting interface) systémy. K dispozici je 128 adres, 32 skupin a 16 scén definovaných pro všechny prvky v systému.

LUXMATE PROFESSIONAL je systém pro komplexní ovládání technologií umělého osvětlení včetně integrace zastíňovacích elementů. Možná automatizace vychází z centrálního řízení, i když je budova pokryta decentralizovaným systémem. Systém lze propojit s LON a EIB.

### 1.4.4 ZigBee

Bezdrátový komunikační standard ZigBee umožňující vzájemnou komunikaci mnoha zařízení na vzdálenost desítek až stovek metrů vznikl, aby vyplnil mezeru mezi rozšířenými technologiemi jako je WIFI a Bluetooth. Standard byl schválen v roce 2004 a je označen jako IEEE 802.15.4. V roce 2005 se začala připravovat verze 1.1. ZigBee má nízké nároky na hardware a nízkou spotřebu. Má nízkou přenosovou rychlost a umožňuje až tisíce účastníků. Standard lze uplatnit v řízení budov, spotřební elektronice, zdravotní péči, sportu a průmyslu [7].

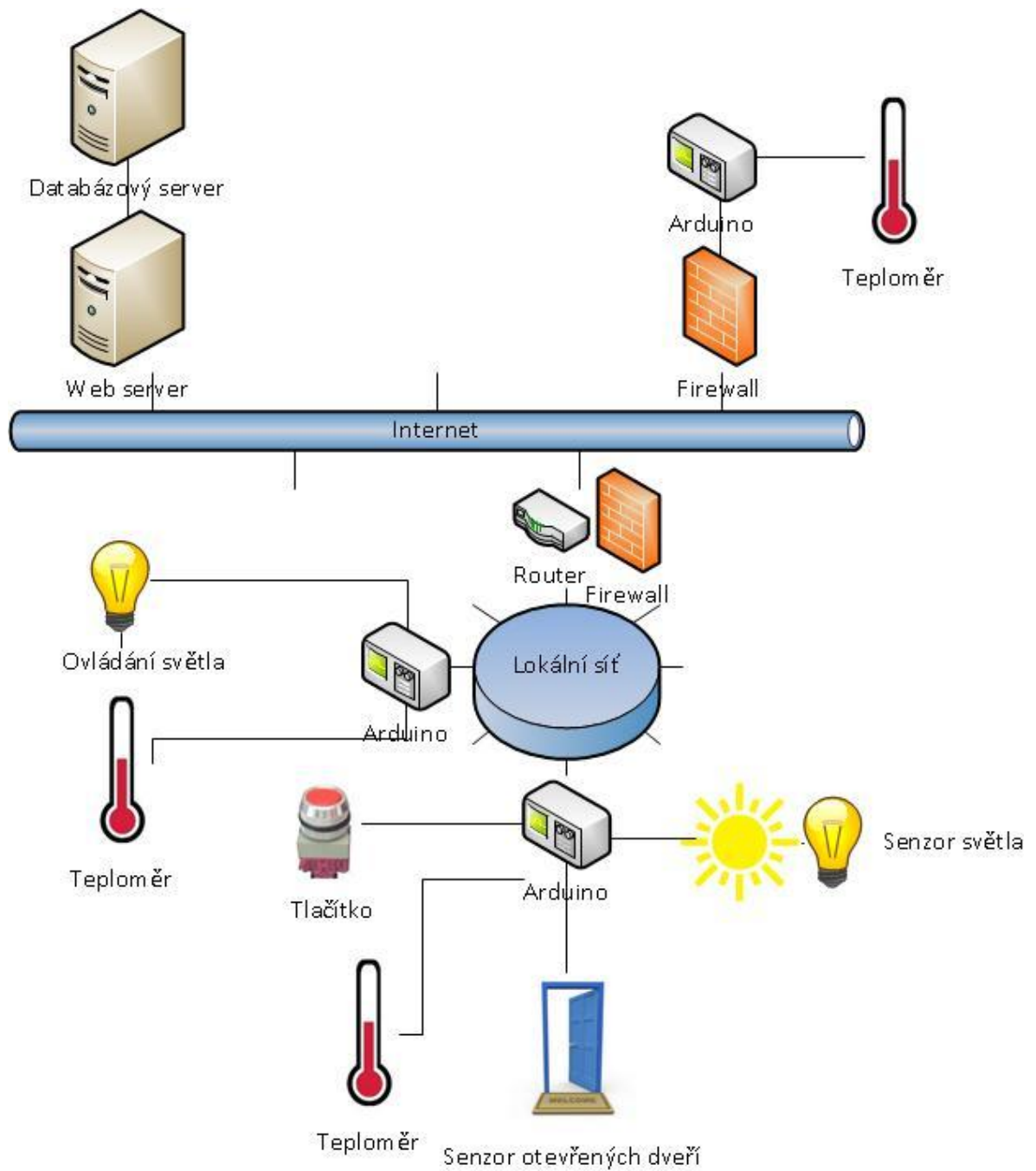
Je možné zakoupit modul, který umí spolupracovat s platformou Arduino. Tento modul nese název Xbee Shield.

## 1.5 Návrh architektury

Inteligentní prvky Arduino a webový server budou připojeny do sítě Internet. Některé prvky například databázový server je vhodné kvůli bezpečnosti připojit pouze do lokální sítě, jelikož není nutné, aby byl databázový server vidět ze sítě Internet. K počítači, kde se

nachází služba databázového serveru, přistupuje pouze počítač s webovým serverem. K inteligentním prvkům jsou připojeny senzory, akční členy a zobrazovací prvky. Inteligentní prvky mohou být připojeny do lokální sítě nebo k Internetu, ale musí být viditelné z Internetu, jestliže vyžadujeme vzdálené ovládání. Arduino připojené přímo k Internetu je vhodné umístit za FireWall, protože síťový provoz na Internetu může tento jednoduchý hardware zatěžovat požadavky a následně může dojít k omezení funkčnosti.

Pokud se nachází inteligentní prvky Arduino i webový server v jedné lokální síti, pak nemusí být žádný prvek viditelný ze sítě Internet. Požadujeme-li připojení k webovému serveru ze sítě Internet, pouze webový server umístíme a nakonfigurujeme tak, aby byl viditelný z Internetu. Možných řešení zapojení je více, protože propojení je řešeno přes standardní počítačovou síť.



Obr. 2. Návrh architektury

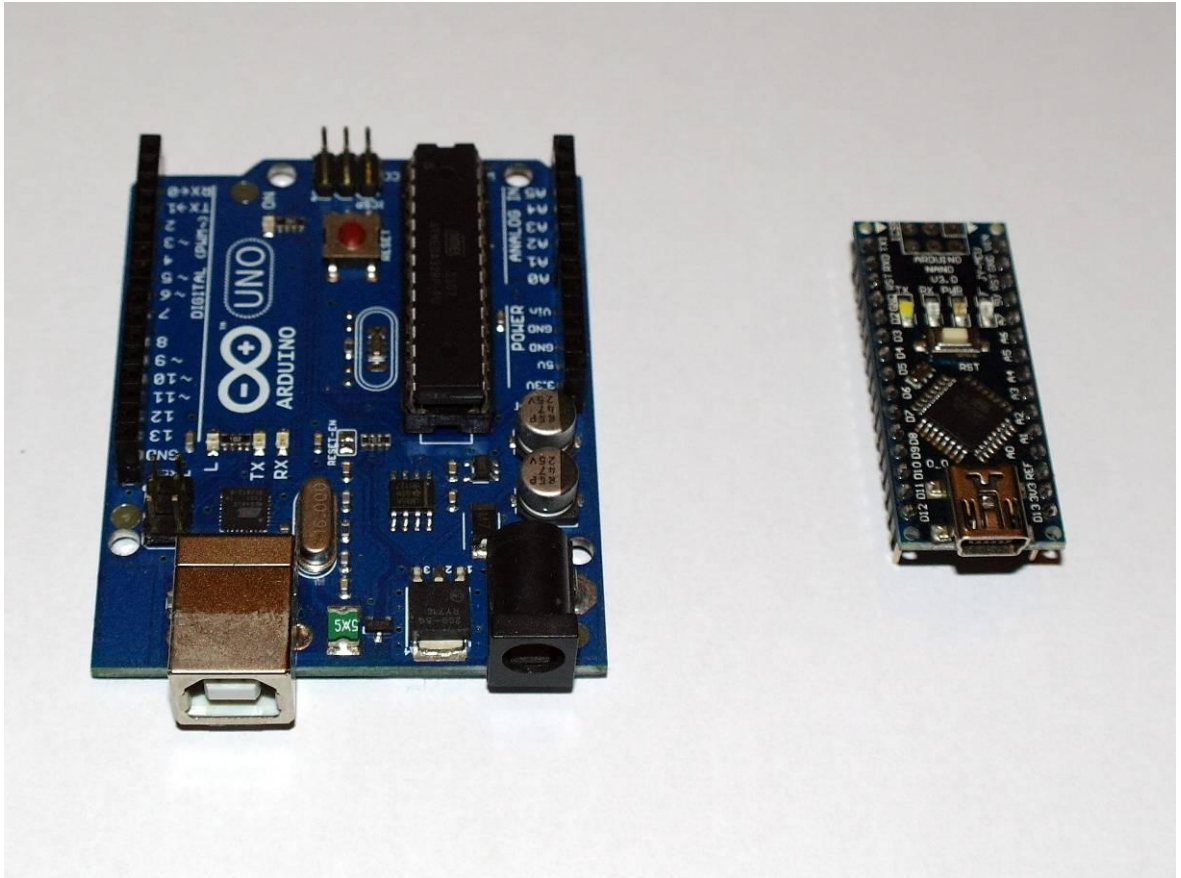
## 2 MIKROKONTROLÉR

Základním prostředkem k řízení inteligentní domácnosti je mikrokontrolér - malý kompletní počítač na jednom integrovaném obvodu obsahující jádro procesoru, paměť a programovatelné vstupní a výstupní periferie. Za první mikroprocesor je považován mikroprocesor Intel 4004 uvedený v roce 1971. Tento mikroprocesor potřeboval pro svoji činnost další externí čipy pro fungování celého systému. Za první mikrokontrolér je považován mikrokontrolér TMS 1000 od firmy TEXAS INSTRUMENTS vytvořený v roce 1971 a do prodeje uveden v roce 1974. Firma Intel vyvinula jako odpověď na TMS 1000 svůj mikrokontrolér 8048, který je prvním mikrokontrolérem firmy Intel [8].

### 2.1 Arduino

Arduino je open source hardware. Licence umožňuje komukoliv vylepšovat, stavět nebo rozšiřovat Arduino (licence Creative Commons Attribution Share-Alike 2.5). Tato licence vedla k velké oblibě platformy a jejímu rozšíření. Původní Arduino i jeho vývojové prostředí vzniklo v roce 2005 v Itálii ve firmě SmartProjects. Projekt získal o rok později ocenění na Prix Ars Electronica 2006. K tomuto projektu vzniklo několik klonů, které většinou obsahují nějaké vylepšení a jsou kompatibilní s Arduinem a jeho vývojovým prostředím. Nesmí však nést jméno Arduino, ale například FreeDuino (především název, který může kdokoliv použít), Boarduino, Seeduino, Roboduino (určené pro návrh robotů) a další.

Mikrokontrolér je z rodiny ATmega a výrobcem těchto čipů je norská firma Atmel [9].



Obr. 3. Arduinino Uno a Arduino Nano

Tab. 1. Srovnání parametrů vybraných aktuálních typů desek platformy Arduino [10]

	Arduino Uno	Arduino Nano	Arduino Mega 2560
Mikrokontrolér	ATmega328	ATmega168 nebo ATmega328	ATmega2560
Pracovní napětí	5 V		
Vstupní napětí (doporučené)	7-12 V		
Vstupní napětí	6-20 V		
Digitální vstup/výstup počet pinů	14 (6 PWM)		54 (14 PWM)
Analogový vstup počet pinů	6	8	16
Paměť Flash / SRAM / EEPROM	32 kB / 2 kB / 1 kB	16 kB / 1 kB / 512 B nebo 32 kB / 2 kB / 1 kB	256 kB / 8 kB / 4 kB
Flash – použito zavaděčem	0,5 kB	2 kB	8 kB
Frekvence	16 MHz		
Velikost (mm)	68*53	43*19	101*53

Poznámky:

EEPROM – zápis trvá 3,3 ms a paměť má životnost 100 000 zápisů

### 2.1.1 Digitální vstup/výstup

Digitální vstup/výstup pracuje s binárními hodnotami 0 a 1, v některých literaturách označovanými jako Low a High hodnota. Logická 0 odpovídá 0 V a logická 1 odpovídá 5 V. Některé starší verze Arduina (např. jedna z verzí Pro nebo Pro Mini) mají logickou 1 na hodnotě 3,3 V [10].

### 2.1.2 Analogový vstup

Analogový vstup u Arduina má 10 bit analogovo - digitální převodník. To nám poskytuje číselnou hodnotu od 0 do 1023 [11]. Hodnota 1023 odpovídá napětí 5 V a hodnota 0 odpovídá 0 V na analogovém vstupu. Rozdíl jedné hodnoty odpovídá 0,0048 V.

Některé verze Arduina umí využít analogové vstupní piny i jako digitální vstup a výstup (Atmega8, Atmega168, Atmega328 nebo Atmega1280). Při současném analogovém čtení a přepínání digitálního binárního stavu může dojít k šumu na analogovém pinu a doporučuje se užít krátká časová prodleva mezi popsányými operacemi [10].

### 2.1.3 Analogový výstup

Arduino má několik digitálních pinů označených jako PWM – Pulse Width Modulation, které lze naprogramovat na analogový výstup [11]. Toho se docílí pomocí modulace šířkou pulsu. Frekvence je okolo 490 Hz. Hodnota se zadává v rozmezí 0 až 255.

## 2.2 Ethernetová karta

Ethernetová karta pro Arduino patří mezi oficiální Arduino Shields. Novější verze obsahuje i slot na paměťovou kartu typu Security Digital (dále SD). Verze bez SD karty připojená na rozhraní Arduina používá 4 digitální piny (pin číslo 10 - 13 na Arduinu), verze s SD kartou 5 pinů (pin číslo 9 - 13 na Arduinu) [10]. Tyto verze karet používají různé verze ethernetového řadiče. Novější verze karty používá řadič Wiznet W5100. Dle zvolených řadičů se musí použít odpovídající knihovny.



Obr. 4. Ethernetová karta pro Arduino - Ethernet Shield V1.1

Vlastností karty je, že neobsahuje přednastavenou MAC adresu. Programátor musí zvolit unikátní MAC adresu v počítačové síti, kam je mikrokontrolér připojen.

Tab. 2 Vlastnosti ENC28J60 [12]

IEEE 802.3. kompatibilní Ethernet řadič
Plně kompatibilní s 10/100/1000 Base-T sítí
Integrovaná MAC a 10 Base-T fyzická vrstva
Podpora jednoho 10 Base-T Portu s automatickou detekcí polarity a korekcí
Podpora módů plný a poloviční duplex
Nastavitelné automatické opakované posílání signálu při kolizi
Programovatelné generování CRC kódu
Nastavitelné automatické zamítnutí chybných paketů

Arduino bude připojeno prostřednictvím ethernetového kabelu a koncovky RJ-45 do počítačové sítě. Komunikace mezi Arduinem a serverem bude probíhat pomocí Internetu nebo lokální sítě Ethernetu. Z toho plyne, že topologie sítě je hvězdicová. Výhoda tohoto způsobu komunikace spočívá v použití ethernetové kabeláže, která již bývá v dnešní době

obvyklým standardem. Kabeláž dále můžeme použít pro připojení jiných síťových prvků jako osobní počítač k Internetu, VOIP telefon, IP kamera apod. Nemusíme pro každou službu zavádět extra kabeláž, která by výrazně prodražila veškerou další použitou službu. Ethernetový kabel může zajišťovat napájení po datovém síťovém kabelu PoE (Power over Ethernet), kdy nemusíme vést k Arduino další kabel, a to pro napájení. Pokud bychom zvolili možnost napájení PoE pro Arduino, musíme zvolit příslušnou kartu.

## 2.3 Senzory

*„Senzor je funkční prvek tvořící vstupní blok měřicího řetězce, který je v přímém styku s měřeným prostředím. Pojem senzor je ekvivalentní k pojmu snímač, převodník nebo detektor. Citlivá část senzoru se občas označuje jako čidlo. Senzor jako primární zdroj informace snímá sledovanou fyzikální, chemickou nebo biologickou veličinu a dle určitého definovaného principu ji transformuje na měřící veličinu – nejčastěji na veličinu elektrickou. Dále existují senzory, u nichž je neelektrická veličina přímo transformována na číslicový signál.“ [13, s. 13]*

Rozdělení senzorů [13]:

- Dle měřené veličiny.
- Dle fyzikálního principu.
- Dle styku senzoru s měřeným prostředím.
- Dle transformace signálu.
- Dle výrobní technologie.

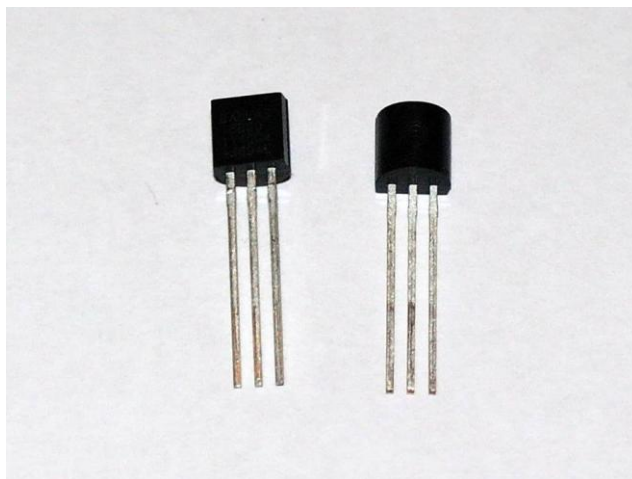
Inteligentní senzor je takový senzor, který obsahuje čidlo, obvody pro zpracování, analýzu a unifikaci signálu v jediném kompaktním provedení [13].

Pro účely práce byly zvoleny senzory, kde jsou dostupné příklady, schémata a zdrojové kódy, jak je používat s Arduinem nebo jiným mikrokontrolérem.

### 2.3.1 Teplota

Pro měření teploty byl zvolen senzor Dallas DS18B20. K tomuto zvolenému senzoru teploměru je na Internetu dostatek informací a příkladů, jak jej k Arduino zapojit. Také jsou dostupné zdrojové kódy, knihovny a příklady. Senzor je číslicový teploměr, který se připojí pouze k jednomu datovému pinu, který tvoří sběrnici výrobcem označenou jako 1-Wire. Připojení ke sběrnici umožňuje propojit více senzorů nebo jiných zařízení (1-Wire)

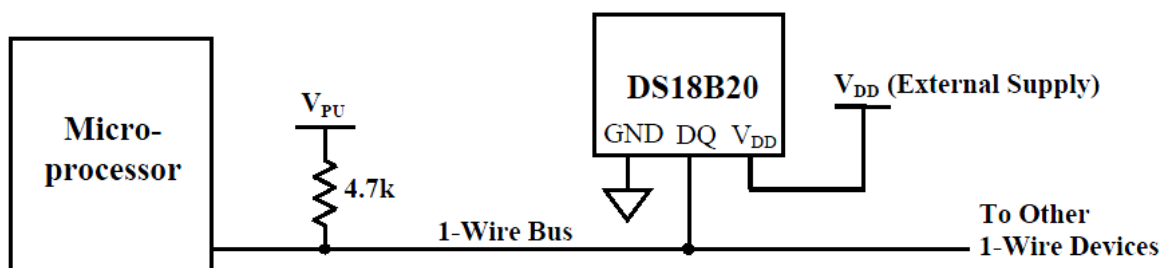
na jeden datový pin [14]. Každý člen má svoji unikátní adresu přednastavenou z výroby. Pokud potřebujeme připojit více teploměrů k Arduino, můžeme využít sběrnice připojení a jiné digitální piny na Arduino připojit k jiným sensorům nebo kartám. Pro domácí použití je přesnost senzoru vynikající. Teploměr je možné zakoupit v České republice. Popis knihovny a stažení knihovny je dostupné na WWW: [http://milesburton.com/index.php?title=Dallas\\_Temperature\\_Control\\_Library](http://milesburton.com/index.php?title=Dallas_Temperature_Control_Library).



Obr. 5. Číslicový teploměr DS18B20

Tab. 3. Teploměr Dallas DS18B20 [14]

Přesnost	+/-0,5 °C (při teplotách -10 °C až +85 °C)
Rozsah měření	-55 °C až +125 °C
Přesnost výstupu	Volitelná 9 až 12 bitů 0,5 °C, 0,25°C, 0,125 °C
Adresa teploměru	Unikátní adresa - 64 bitů
Napájení	3.0 až 5.5 V
Rychlost	Max. 750 ms při 12 bit přesnosti výstupu
Cena	52 Kč



Obr. 6. Schéma možného zapojení jednoho nebo více číslicových teploměrů DS18B20 [14]

Možnosti využití teploměru:

- Ovládání topného systému domácnosti.
- Kontrola správné činnosti spotřebičů (klimatizační jednotka, kotel pro vytápění, lednička, mrazicí zařízení).
- Informování o teplotě v domácnosti nebo venku.

### 2.3.2 Vlhkoměr

Pro zjištění vlhkosti byl použit senzor DHT 11. Tento senzor není možné zakoupit v obchodech v České republice. Lze jej však objednat například v celosvětovém v internetovém obchodě eBay. Zároveň umí senzor měřit teplotu. Z těchto údajů můžeme dopočítat další údaje, např. rosný bod. Popis knihovny a možnosti stažení jsou dostupné na <http://arduino.cc/playground/Main/DHT11Lib>.



Obr. 7. Senzor vlhkoměr DHT11

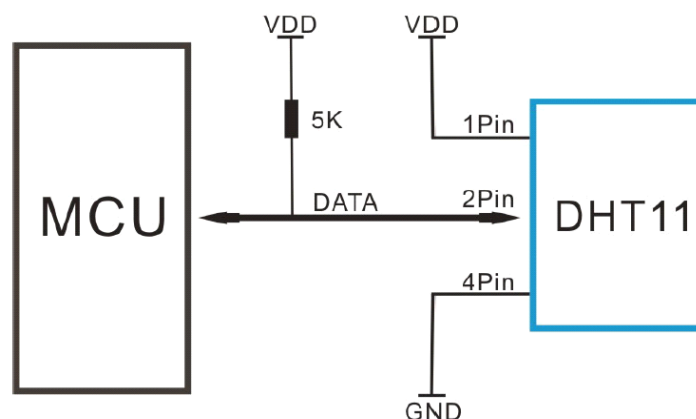
Tab. 4. DHT 11 vlhkoměr a teploměr [15]

Rozsah vlhkoměru	20 % ~ 90 %
Rozsah měření teploty	0 až +50 °C
Přesnost vlhkoměru	± 5.0 %
Přesnost teploměru	± 2.0 °C
Přesnost výstupu	8 bitů 1 °C
Napájení	3.0 - 5.5 V
Proud čtení hodnot / vyčkávání	0,5-2,5 mA / 100 -150 uA
Rychlost	< 5 s
Cena	3 GBP - 90 Kč včetně poštovného

Možnosti využití vlhkoměru:

- Informování o vlhkosti v domácnosti nebo venku.
- Předcházení vysušenému vzduchu nebo plísním v domácnosti.

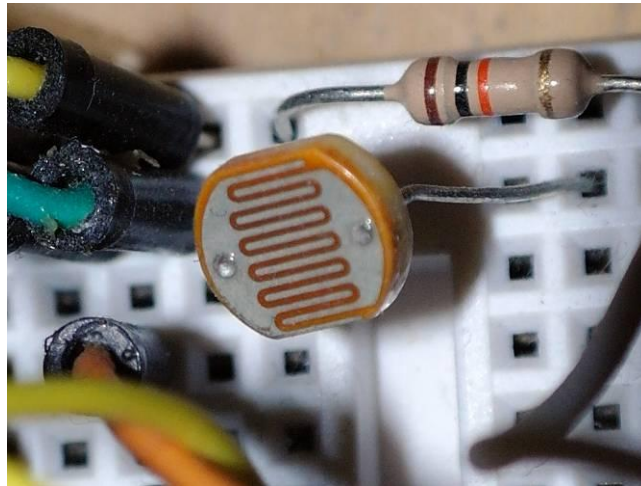
Ideální vlhkost v domácnosti by měla být v rozmezí 45-60 %. Jsou-li hodnoty vyšší, hrozí v domácnosti plísně. U nižších hodnot je riziko vysoušení sliznic dýchacích cest a jiné zdravotní problémy [16].



Obr. 8. Zapojení senzoru DHT11 [15]

### 2.3.3 Intenzita osvětlení

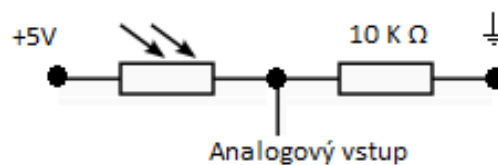
Senzor pro zjištění intenzity osvětlení v místnosti můžeme vytvořit z fotorezistoru. Světlo je forma elektromagnetického záření s vlnovou délkou 380-780 nm [13]. Fotorezistor se připojuje k Arduino k analogovému vstupu. Arduino má 10 bit analogový digitální převodník. To nám poskytuje 1024 různých hodnot.



Obr. 9. Fotorezistor pro zjištění intenzity osvětlení

Tab. 5. Fotorezistor VT83N4

Odpor při světle 10 lux - 2850 K	50 – 150 k $\Omega$
Odpor při tmě	2 M $\Omega$
Maximální ztrátový výkon	175 mW
Cena	18 Kč



Obr. 10. Zapojení fotorezistoru k Arduino

Možnosti využití senzoru světla:

- Kontrola vypnutých světel po odchodu z domácnosti.
- Automatická regulace světel v místnosti na základě přirozeného světla.
- Kontrola fungujícího osvětlení.

### 2.3.4 Otevřené/zavřené dveře nebo okno

Pro detekování otevřených či zavřených dveří nebo oken, můžeme použít magnetický dveřní spínač. Tento spínač lze připojit na digitální nebo analogový pin. Pokud spínač připojíme na analogový pin, pak musíme určit hranici otevřeno/zavřeno. Tato hodnota nemůže být blízko hraničním hodnotám, protože dochází k rušení.



Obr. 11. Magnetický dveřní spínač

Tab. 6. Magnetický dveřní spínač

Velikost	27.4 x 7.8 mm
Maximální napětí/proud	100 V / 500 mA
Aktivní mezera mezi protikusy	15 mm
Cena	43 Kč

Možnosti využití magnetického okenního/dveřního spínače:

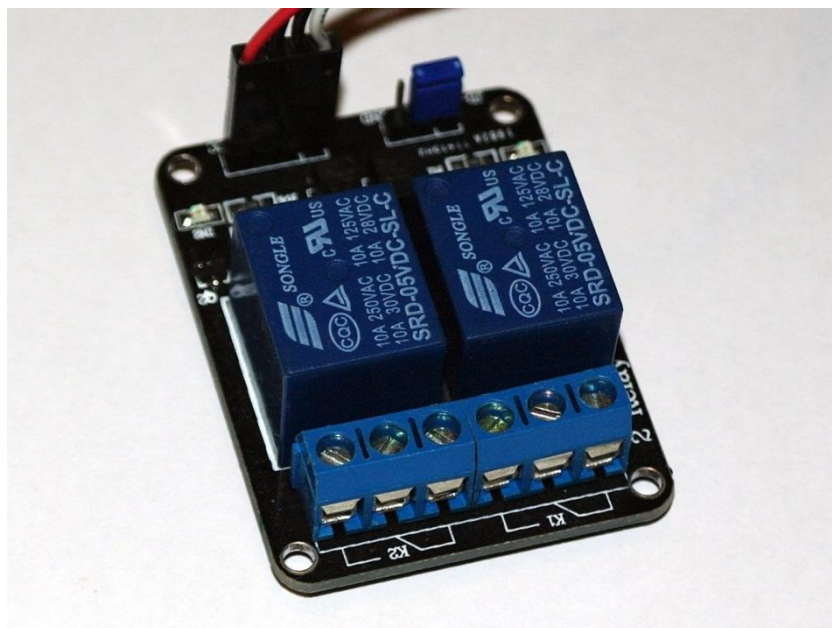
- Kontrola vstupu do místnosti/bytu.
- Kontrola otevřených/zavřených oken.
- Kontrola spotřebičů, které mají být otevřeny/zavřeny.

## 2.4 Akční členy

Akčními členy rozumíme všechny prvky určené k využití zpracované informace. Jsou to prvky na konci řetězce zpracování informace. Nastavují velikost akční veličiny, tj. realizují vstup do regulované soustavy. Jejich nejčastějšími představiteli jsou pohony a na ně navazující regulační orgány. Akční člen je technické zařízení přenášející výstupní signál z regulátoru do regulované soustavy.

### 2.4.1 Elektromagnetické relé

Elektromagnetické relé je elektronická součástka skládající se ze spínače a cívky, která působením elektrického proudu v jednom obvodu zapne/přeruší elektrický proud v druhém elektrickém obvodu. Bylo zvoleno sestavené relé, připravené pro různé typy mikrokontrolérů. Prodává se sestavené s počtem 1-16 relé. Každé relé se připojí na jeden datový pin. Není složité si vytvořit obvod s relé pro mikrokontrolér. K tomu zapotřebí relé, dioda, kondenzátor a odpor.



Obr. 12. Elektromagnetické relé

Tab. 7. Elektromagnetické relé

Počet relé / binárních řídicích vstupů	2
Proud pro sepnutí	15-20 mA
Maximální napětí/proud ovládaného zařízení	AC 250 V 10 A DC 30 V 10 A
Cena	4 GBP asi 120 Kč včetně poštovního

Možnosti využití relé:

- Ovládání topného systému domácnosti.
- Řízení a spínání elektrických spotřebičů – topení, světlo.

#### 2.4.2 Servomotor

Servomotor je motor pro pohony, u kterých lze na rozdíl od běžného motoru nastavit přesnou polohu natočení osy. Signál snímače polohy je přiveden pomocí zpětné vazby na regulátor porovnávající skutečnou polohu motoru s žádanou polohou. Na základě rozdílu žádané a skutečné polohy regulátor řídí měnič, a tak nastavuje motor na žádanou polohu.



Obr. 13. Servomotor SG90

Tab. 8. Servomotor SG90

Napětí	3.5-8.4 V
Rychlost	0.12 sec/ 60° (4.8V)
Tah	2 kg/cm
Rozměry	21x12x22 mm
Váha	9 g
Cena	100 Kč

Možnosti využití servomotoru:

- Ovládání armatur.
- Otevírání/zavírání.

## 2.5 Zobrazovací prvky

Zobrazovací prvky jsou elektronickými signály ovládaná zařízení zprostředkovávající pozorovateli určitou vizuální informaci o sledovaném procesu [17].

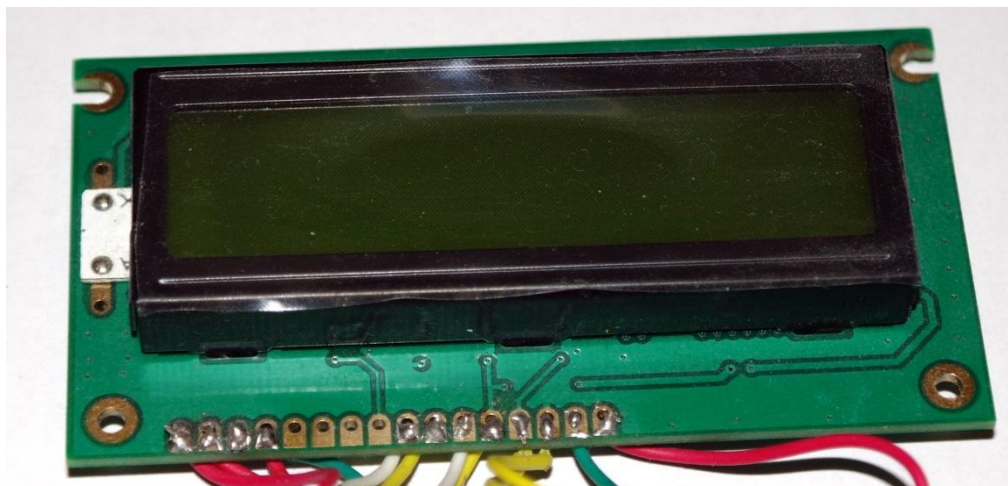
### 2.5.1 Displej

Displej (display) je výstupní zařízení pro zobrazování informací např. textu, obrazu v různé podobě. Existuje několik druhů displejů.

Důležité vlastnosti LCD displejů [17]:

- Nízká cena.
- Nízká hmotnost.
- Malé rozměry.
- Velmi nízká spotřeba elektrické energie.

LCD displej můžeme použít pouze při vývoji nebo jej můžeme ponechat v zařízení pro zobrazování informací, které by jinak byly v naší práci přístupné pouze přes webové rozhraní. Displej MC1602 obsahuje řadič, který se připojuje k Arduinu v režimu 4 nebo 8 bitů (použije 4 nebo 8 pinů). Lze použít i jiné typy displejů. Pro ovládání displeje byla použita knihovna LiquidCrystal, ta je součástí balíčku s IDE prostředí [10].



Obr. 14. Znakový LCD displej MC1602E-SYL/H obsahující 2 řádky 16 znaků

Tab. 9. Znakový LCD displej MC1602E-SYL/H

Velikost modulu / displeje	84x44x12,5 mm/64,5x16,4mm
Počet znaků / řádků	16 / 2
LCD řadič / ovladač	S6A0069
Počet bodů na znak	5x8
Podsvícení	LED - žluto-zelené
Napětí / proud pro logiku	4,5-5,5 V/1-2 mA
Napětí pro LCD	4,0-5,0 V
Proud pro podsvícení	maximálně 80 mA
Cena srpen 2011 / únor 2012	125 Kč / 210 Kč

Možnosti využití LCD displeje:

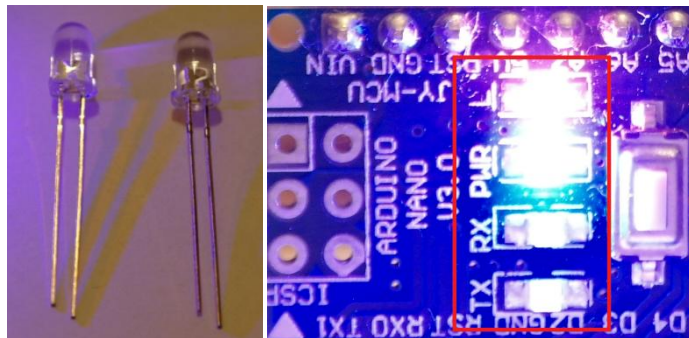
- Zobrazení stavů senzorů - aktuální hodnoty, historické hodnoty aj.
- Zobrazení chybových hlášení.
- Nastavení zařízení.

### 2.5.2 LED diody

LED (Light-Emitting Diode) je elektronická polovodičová součástka obsahující přechod P-N. LED dioda na rozdíl od klasických diod vyzařuje viditelné světlo, infračervené záření nebo UV záření v úzkém spektru barev [17].

LED diody můžeme použít pro vývoj nebo jako prvek k signalizaci stavu zařízení. Výhoda použití LED diod spočívá ve velké životnosti, v ceně, v jednoduchém zapojení, ve velkém množství různých typů (barva, velikost, svítivost) a v energetické úspornosti. Jedna LED dioda je využitelná pro signalizaci více stavů, které lze signalizovat svícením, blikáním, a to rychle nebo pomalu, zhasnutím a blikáním nepravidelným. Využití více stavů signalizace jedné nebo několika LED diod má příznivý dopad na cenu celého zařízení, avšak může nastat problém s uživateli, kdy uživatel bude mít potíže s rozpoznáním stavu zařízení a nebude vědět, co který stav znamená. Pokud je potřeba sdělit uživateli více stavů, je vhodnější použít LCD displej.

Při zapojení LED diody k Arduino, musíme LED diodu připojit k vhodnému rezistoru, jinak diodu zničíme.



Obr. 15. LED diody

Možnosti využití LED diody:

- Signalizace stavu.
- Energeticky úsporné osvětlení.
- V maticovém sestavení jako zobrazovač informací.

## **3 WEBOVÝ SERVER**

### **3.1 Základní možnosti systému**

Uživatelé budou přistupovat k webovému serveru, který je na síti Internet. Existuje možnost přistupovat přímo k Arduinu, ale z hlediska nevykonného hardware, na kterém nelze vytvořit uživatelsky přívětivý a víceuživatelský přístup, tuto možnost vynecháme. Webový server přistupuje přes protokol HTTP k rozhraní Arduina a posílá mu příkazy, které uživatel zvolí ve webové aplikaci.

Uživatelé se prostřednictvím sítě Internet mohou připojovat přes libovolného tenkého klienta k webovému serveru a kontrolovat tak stav domácnosti nebo nastavovat Arduino, pokud je vybaveno pro povely. Mohou si zobrazit historii stavu domácnosti.

### **3.2 Přístup k systému**

Informační systém je tvořen webovou aplikací pro přístup z Internetu bez ohledu na použitý prohlížeč a operační systém.

Přístup k systému:

- Nepřihlášený uživatel.
- Přihlášený uživatel.
- Administrátor.

#### **3.2.1 Nepřihlášený uživatel**

Nepřihlášený uživatel se může registrovat nebo přihlásit, má-li zřízený účet. Dále se může dostat do nezabezpečené části, kde nalezne informace o projektu.

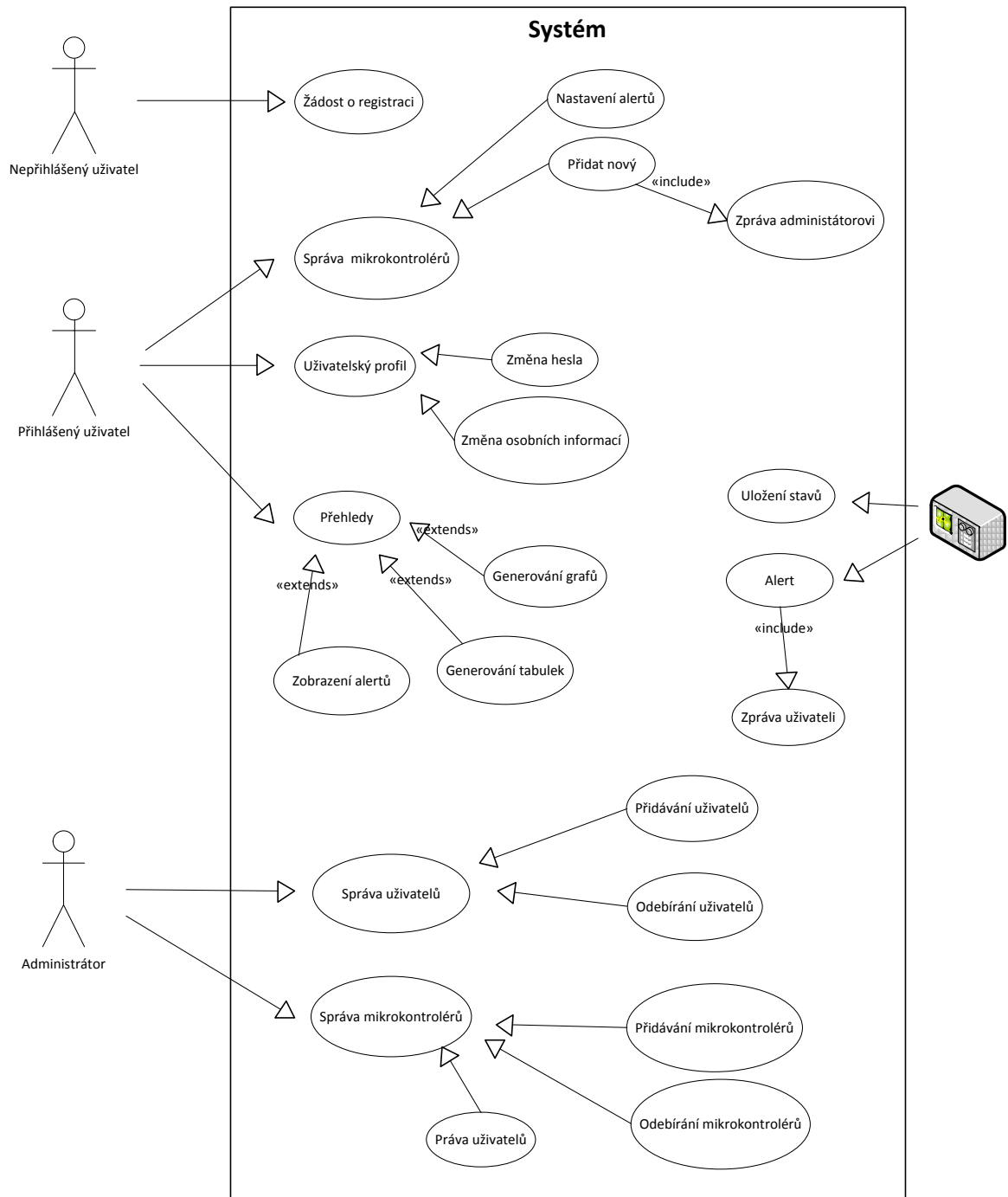
#### **3.2.2 Přihlášený uživatel**

Přihlášený uživatel může prohlížet své registrované jednotky, přidávat další jednotky, nastavovat alerty (varování), zobrazovat hlášení, prohlížet stav domácnosti a posílat příkazy Arduinu.

#### **3.2.3 Administrátor**

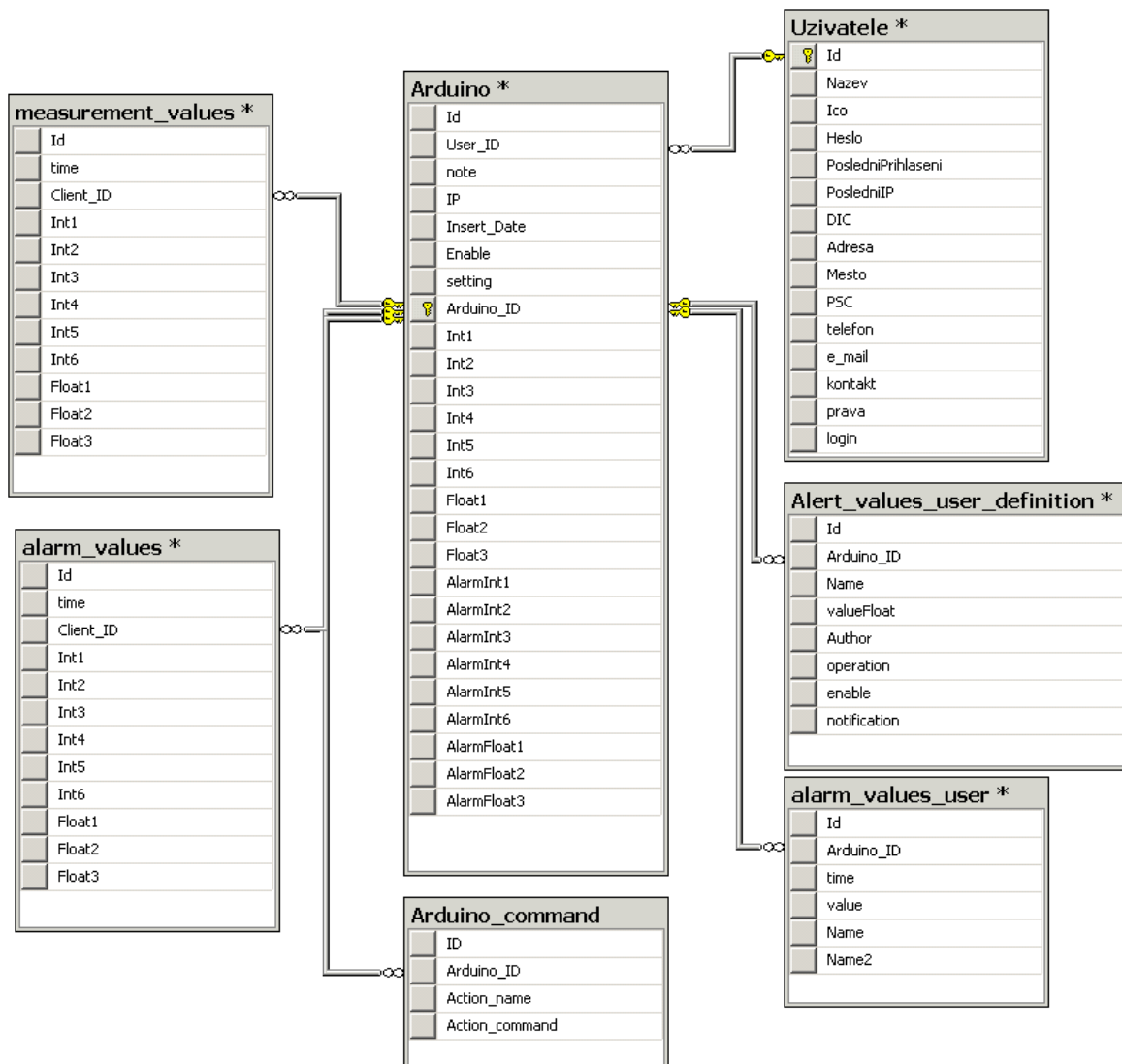
Administrátor může přidávat a odebírat uživatele a schvalovat zařízení domácnosti registrované uživatelem.

### 3.3 Use case diagram



Obr. 16. Use Case diagram webového serveru

### 3.4 ER diagram



Obr. 17. ER diagram

#### 3.4.1 Databázová tabulka Uzivatele

Tabulka obsahuje veškeré údaje o uživateli. Sloupec ID obsahuje jednoznačný unikátní identifikátor uživatele, který je následně použit v tabulce Arduino. Z hlediska aplikace jsou další důležité sloupce login a heslo, kde je uživatelské jméno a heslo pro přihlášení do aplikace. Heslo je uloženo v podobě otisku (hash) hesla. Sloupec e\_mail, slouží pro posílání notifikací uživateli.

### 3.4.2 Databázová tabulka Arduino

Tabulka obsahuje veškeré údaje o mikrokontroléru Arduino. Sloupec Arduino\_ID obsahuje jednoznačný unikátní identifikátor Arduina. Sloupec User\_ID obsahuje ID uživatele, který má právo přistupovat k nastavení a hodnotám Arduina. Sloupce Int1- Int6, Float1- Float 3, AlarmInt1- AlarmInt6 a AlarmFloat1- AlarmFloat3 slouží k pojmenování sledovaných hodnot.

### 3.4.3 Databázová tabulka alarm\_values

Tabulka obsahuje hodnoty, které Arduino zaznamenalo při stavu alarm. Je uloženo datum vzniku události a stavy jednotlivých hodnot. Tyto stavy jsou posílány okamžitě při překročení naprogramovaných hodnot a uživatel je nemůže sám měnit, protože je nutné jednotku přeprogramovat.

### 3.4.4 Databázová tabulka Alert\_values\_user\_definition

Tabulka obsahuje nastavení pro hodnoty, které se mají vyhodnotit jako hodnoty typu alarm. Tyto hodnoty bude vyhodnocovat webový server a uživatel je může snadno měnit. Dále je uložen stav, zda je alarm povolen a přeje-li si uživatel posílat notifikace při překročení nastavených hodnot.

### 3.4.5 Databázová tabulka alarm\_values\_user

Tabulka obsahuje hodnoty, které systém vyhodnotí dle uživatelského nastavení jako hodnoty alarm. Jsou to hodnoty, o kterých chce být uživatel informován nebo chce, aby tyto hodnoty byly zvlášť uchovány. Tyto hodnoty jsou vyhodnoceny na straně webového serveru při periodickém posílání stavu Arduina.

### 3.4.6 Databázová tabulka Arduino\_command

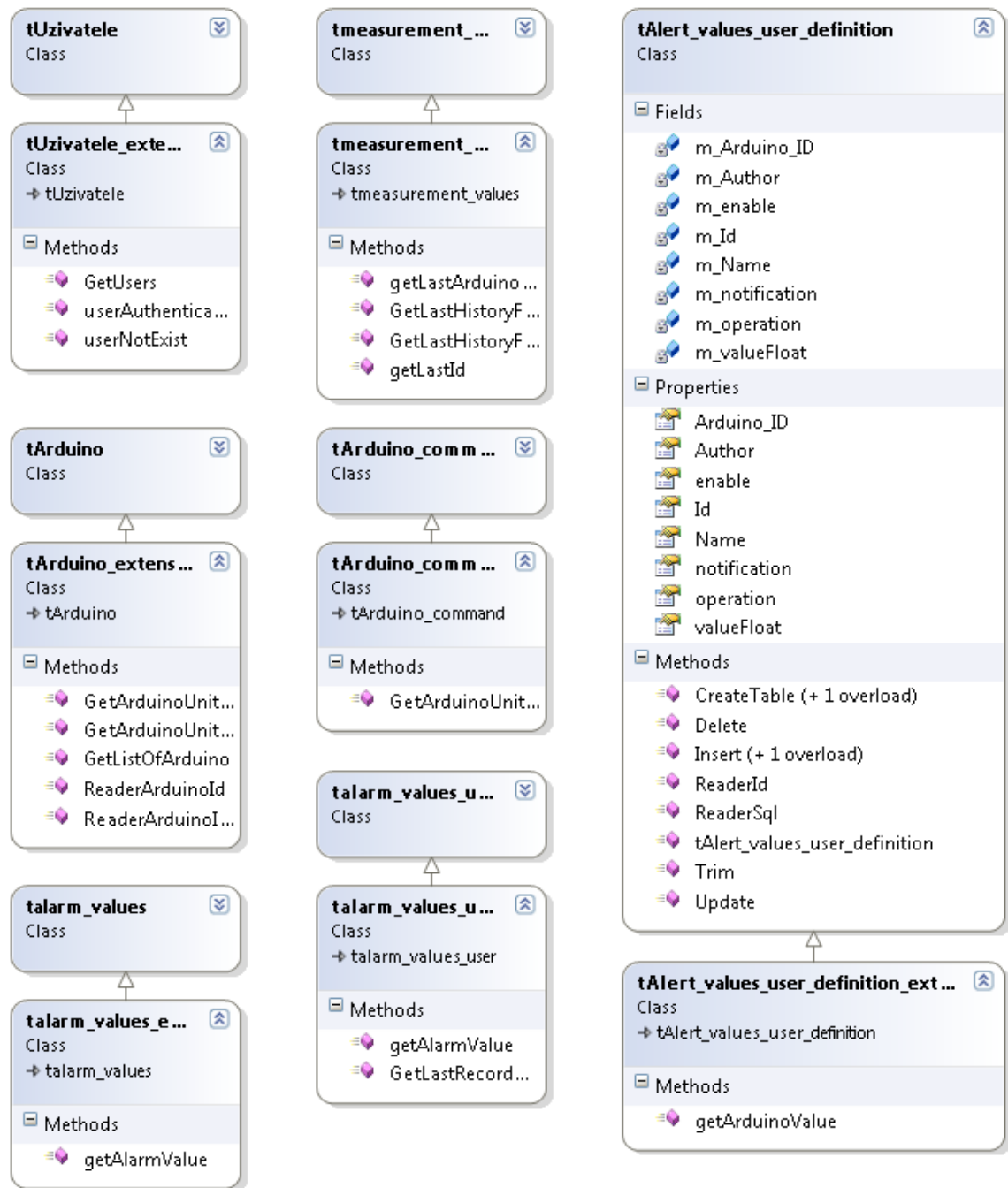
Tabulka obsahuje hodnoty příkazů pro Arduino a názvy těchto příkazů. Uživatel vybere příkaz, který pošle webový server Arduinu a to vykoná přeprogramovanou činnost. Arduino musí být naprogramováno na příslušný příkaz.

### 3.4.7 Databázová tabulka measurement\_values

Tabulka obsahuje veškeré hodnoty senzorů, jež posílá periodicky Arduino.

### 3.5 Diagram tříd

Diagram tříd je rozdělen dle vrstev webového serveru do několika samostatných obrázků a popisů, kde jsou jednotlivé třídy popsány.



Obr. 18. Diagram tříd databázové vrstvy

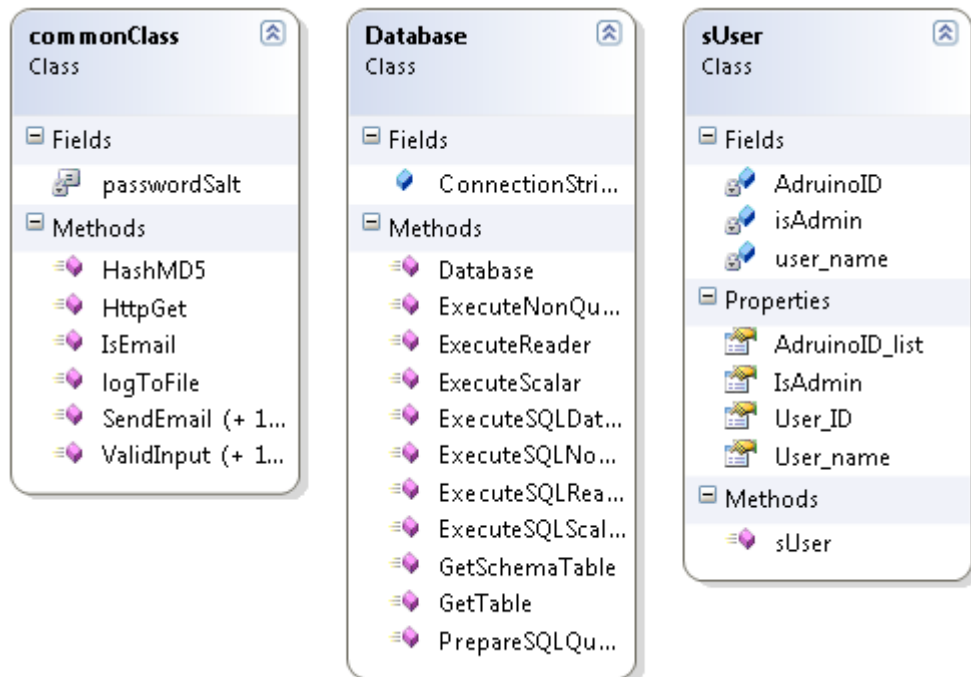
Diagram tříd databázové vrstvy reprezentuje vrstvu databáze. Jednotlivé třídy jsou shodné s tabulkami v databázi a nabízejí základní operace nad daty. Třídy, které dědí základní metody, jsou pojmenovány shodně, ale na konci názvu mají přidáno `_extension`. Důvodem

je to, že základní třídy jsou generované C# kódy na základě schémat z databáze, které se mohou změnit při změně definice databázové tabulky. Generátor těchto zdrojových kódů byl napsán v rámci jiného staršího projektu a generuje kód na základě vytvořené tabulky v databázi. Třídy rozšiřující základní funkčnost jsou dopsané programátorem dle požadavků na systém. Na Obr. 18 je rozbaleno jedno zobrazení základní třídy pro demonstraci a porovnání s ER Diagramem. Další třídy mají shodné objekty s databázovou tabulkou a metody shodné s ostatními základními třídami. Pro lepší názornost jsou skryté.

Vygenerované metody:

- Konstruktor – nastaví proměnné na počáteční hodnoty.
- CreateTable – vytvoří novou tabulku v databázi.
- Delete – smaže záznam.
- Insert – vloží nový záznam do databáze s hodnotami proměnných v objektu.
- ReaderID – naplní proměnné objektu dle parametru ID z databáze.
- ReaderSQL - naplní proměnné objektu dle prvního záznamu z SQL příkazu.
- Trim - odstraní bílé znaky na začátku a na konci řetězce.
- Update – aktualizuje záznam v databázi z proměnných v objektu.

Třídy vytváří SQL dotazy pro databázi a používají třídu Database obsahující metody pro komunikaci s databází na úrovni SQL dotazů. Pokud je výsledkem dotazu získání dat z databáze, tak se těmito daty naplní objekt do příslušných proměnných.

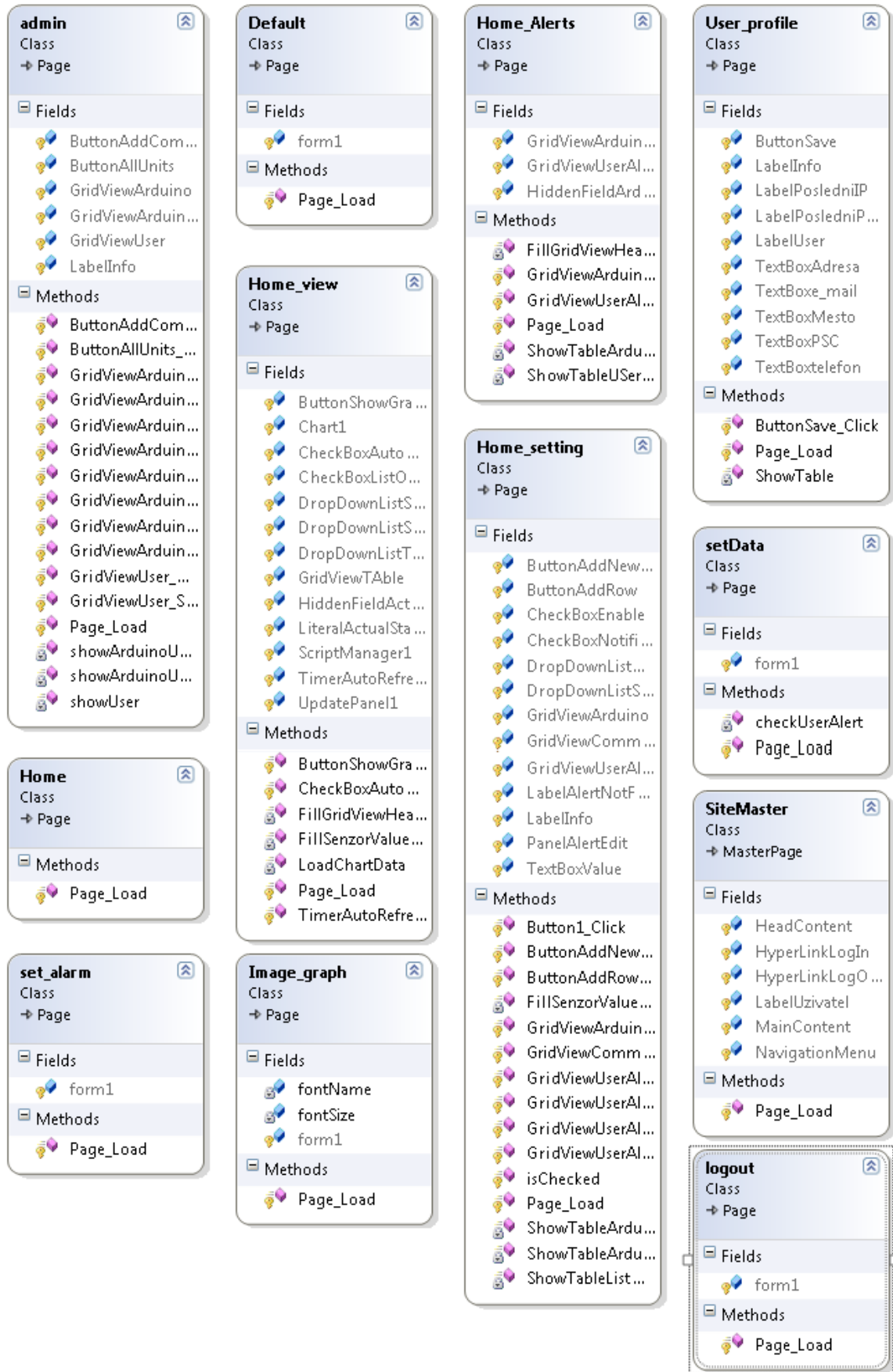


Obr. 19. Obecné třídy

Třída CommonClass obsahuje metody vhodné pro použití v celé práci. Generování hash (MD5) se používá pro uložení hesla a pro generování při přihlášení a porovnání s otiskem uloženým v databázi. HttpGet je metoda pro zavolání požadavku pro Arduino. IsEmail kontroluje pomocí regulárního výrazu, jestli uživatel zadal platný e-mail. LogToFile slouží k uchování provozních informací uložených do souboru na disku serveru. SendEmail zajišťuje posílání e-mailu. ValidInput zabezpečuje kontrolu SQL dotazů proti SQL injection.

Třída Database zajišťuje komunikaci s SQL. Jsou zde metody, které provedou příkaz v databázi, dále metody vracející data z databáze a metoda, která vrací na základě SQL dotazu C# objekt typu DataTable, jež může programátor velmi komfortně dále zpracovávat nebo přímo zobrazit na webové stránce přes objekt GridView.

Instance třídy sUser zajišťuje uchování informací o přihlášeném uživateli. Je to objekt vytvořený po přihlášení uživatele a je uložený do session. Pokud uživatel delší dobu nekomunikuje nebo se odhlásil, je objekt z paměti smazán.



Obr. 20. Use case jednotlivých tříd webových stránek

Třídy zobrazené na obrázku (Obr. 20) reprezentují jednotlivé internetové webové stránky a jejich metody. Jednotlivé webové stránky jsou popsány v kapitole 4.3 Uživatelské rozhraní. Třídy obsahují převážně metody pro obsluhu objektů na webové stránce.

## **3.6 Webové technologie**

### **3.6.1 HTML**

HTML (HyperText Markup Language) je značkovací jazyk pro hypertext. Je jedním z jazyků pro vytváření stránek v systému World Wide Web umožňující publikaci dokumentů na Internetu.

### **3.6.2 CSS**

Kaskádové styly (CSS) je jazyk pro podrobnější popis vzhledu stránek.

### **3.6.3 JavaScript**

JavaScript je programovací jazyk běžící na straně klienta v prohlížeči. Umožňuje měnit obsah, vzhled a webové stránky. JavaScript kód může být rozdílný pro různé prohlížeče, protože každý prohlížeč interpretuje JavaScript jiným způsobem.

### **3.6.4 ASP.NET**

ASP.NET je technologie, která je součástí .NET Framework pro tvorbu webových aplikací a služeb. Je nástupcem technologie ASP (Active Server Pages) a je založena na CLR (Common Language Runtime) sdíleném všemi aplikacemi postavenými na .NET Frameworku [18].

### **3.6.5 AJAX**

(Asynchronous JavaScript and XML) je označení pro skupinu technologií vývoje interaktivních webových aplikací, které mohou komunikovat se serverem bez nutnosti znovunačtení stránky – např. lze měnit obsah stránek bez nutnosti jejich celého znovunačtení. Technologie poskytuje uživatelsky příjemnější prostředí webových stránek na rozdíl od klasických webových aplikací.

### 3.6.6 AJAX Extensions

Komponenty pro ASP.NET využívající AJAX knihovnu. Z hlediska ASP.NET se jedná o standardní server controls. Tyto komponenty jsou podporovány standardními komponentami ve Visual Studiu a obsahují funkce kompatibilní s různými prohlížeči.

### 3.6.7 XML

(eXtensible Markup Language) je značkovací jazyk definující sadu pravidel pro vytváření dokumentu. Byl vyvinut a standardizován konsorciem W3C. Jazyk je určen především pro výměnu dat mezi aplikacemi.

## **II. PRAKTICKÁ ČÁST**

## 4 IMPLEMENTACE

Pro zaznamenávání chodu domácnosti byla zvolena platforma Arduino. Platforma Arduino je velmi populární a je možné ji propojovat s různými senzory, akčními členy a zobrazovacími členy. Arduino lze naprogramovat jazykem podobnému jazyku C.

Část projektu přístupná z Internetu je vytvořena pomocí technologie Microsoft Visual Studio .NET. Výhoda tohoto zvoleného řešení pro uživatele spočívá v možném přístupu pomocí tenkého klienta - přístup prostřednictvím webového prohlížeče pod libovolným operačním systémem. Výhody použití technologie .NET [18]:

- Objektově orientované programování.
- Dobrý návrh.
- Jazyková nezávislost.
- Lepší podpora pro dynamické webové stránky.
- Efektivní přístup k datům.
- Sdílení kódu.
- Větší zabezpečení.
- Nulové dopady instalace.
- Podpora pro Web service.
- Prostředí Visual Studio 2010.

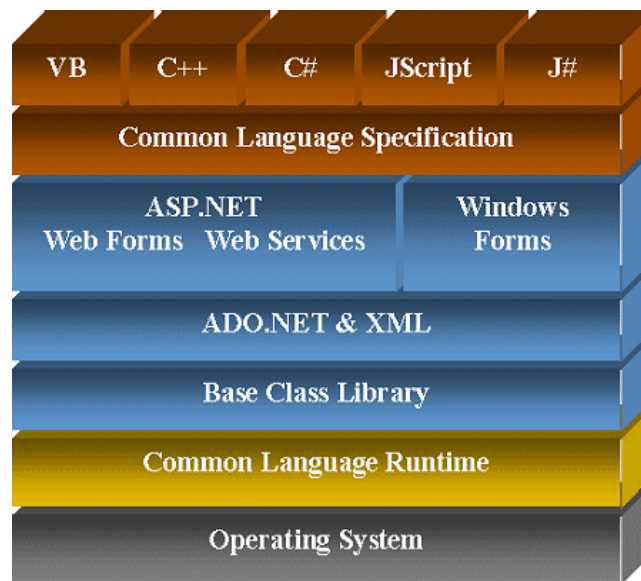
### 4.1 Prostředí implementace

- Microsoft Visual Studio .NET 2010.
- .NET Framework.
- Microsoft SQL server.
- OS Microsoft Windows.
- webový prohlížeč Internet Explorer 9.0, Mozilla Firefox.
- IIS (Internetová informační služba).
- Vývojové prostředí Arduina.

#### 4.1.1 Visual Studio .NET

Microsoft Visual Studio je vývojové prostředí (IDE) od společnosti Microsoft. Toto vývojové prostředí není pouze editorem kódu, ale obsahuje spoustu nástrojů, které zjednodušují návrh, vývoj, testování software a další činnosti spojené s vývojem software.

#### 4.1.2 .NET Framework architektura



Obr. 21. Net Framework [19]

#### 4.1.3 Microsoft SQL server

Při vytváření a používání webových aplikací je nutné použít oddělené místo od webové aplikace pro ukládání dat. Pro podobné úlohy je vhodné použít relační databázový server, kde jsou data uložena. Pro databázi byl zvolen databázový MS SQL server 2008 společnosti Microsoft, protože je přímo podporován ze strany Visual Studio. Použití jiného databázového serveru než MS SQL je možné, ale je nutné předělat v aplikaci databázovou vrstvu, protože SQL příkazy nejsou plně kompatibilní s různými SQL servery.

#### 4.1.4 Microsoft Windows server 2008 R2

Pro přístup na webový server byl použit operační systém Windows server 2008 R2. Tento operační systém je optimalizován pro server v počítačové síti. Mimo jiné obsahuje webovou službu zvanou Internetová informační služba a lze do něj doinstalovat plnohodnotný databázový server MS SQL.

#### 4.1.5 Vývojové prostředí Arduina

Programy pro Arduino se vytvářejí ve vývojovém prostředí založeném na projektu Processing. Processing je otevřený nástroj používaný pro vizualizaci, animaci a řízení hardware přímo z prostředí osobního počítače [9].

Pro vývoj byla použita verze IDE Arduino 0022 (alpha). Po této verzi vydali autoři verzi 0023 a dále aktuální verzi Arduino 1.0 (stable). Některé knihovny nemusí být plně kompatibilní se stabilní verzí, protože autoři měnili knihovny a jejich umístění.

## 4.2 Arduino

### 4.2.1 Ethernetová karta

Byla použita starší verze ethernetové karty obsahující řadič ENC28J60, protože není potřeba použít paměťovou kartu, pro kterou obsahuje podporu novější verze karty. Použité knihovny pro použití karty s řadičem ENC28J60 jsou dostupné na <http://blog.thiseldo.co.uk/?p=329>.

Karta je použita pro posílání hodnot na server, zobrazení testovací stránky Arduina a zachytávání příkazů, které posílá web server v síti Internet, a to pomocí protokolu HTTP a metody GET.

Zjednodušený zdrojový kód pro periodické posílání stavu senzorů z Arduina:

```
1 sprintf(statusstr,
  "?f1=%s&p1=%i&p2=%s&p3=%s&p4=%s&p5=%i&ID=%i&pk=konec", inTempStr,
  inTest, inTeplo2, inSvetlo, inVlkost, RadomValue, arduinoID);
2 es.ES_client_set_wwwip(webip);
3 es.ES_client_browse_url(PSTR(WEBURL), statusstr, PSTR(WEB_VHOST),
  &browserresult_callback);
```

### 4.2.2 Teplota

Senzorů teploměru může být připojených k jednomu datovému pinu na Arduinu více. Komunikaci zajišťuje knihovna OneWire a knihovna DallasTemperature. Zdrojový kód ukazuje získání teploty z jediného připojeného čidla teploměru:

```
1 float getTemperature()
2 { sensors.requestTemperatures();
3   float TemperatureC=-1;
4   // Search the wire for address
5   if(sensors.getAddress(tempDeviceAddress, 0)) //prvni tepl.
6   {
7     TemperatureC = sensors.getTempC(tempDeviceAddress);
8     Serial.print("Temp C: ");Serial.println(TemperatureC);
9   }
10  return TemperatureC;
11 }
```

### 4.2.3 Vlhkoměr

Knihovna DHT11 umožňuje čtení dat ze senzoru. Komunikace se senzorem je poměrně jednoduchá. Senzor posílá jako odpověď na požadavek 40 bit dlouhou zprávu. Převedení na pole byte - na 1. pozici je vlhkost, na 3. pozici se nachází teplota, 2. a 4. pozice je prázdná. Na 5. pozici je kontrolní součet teploty a vlhkosti.

Zjednodušený zdrojový kód čtení hodnot z vlhkoměru:

```
1 int chk = DHT11.read(DHT11PIN);
2 switch (chk)
3 {
4   case 0: Serial.println("OK"); break;
5   case -1: Serial.println("Checksum error"); return; break;
6   case -2: Serial.println("Time out error"); return; break;
7   default: Serial.println("Unknown error"); return; break;
8 }
9 value_humidity= DHT11.humidity;
10 value_temperature2 = DHT11.temperature;
```

### 4.2.4 Intenzita osvětlení

Údaj o intenzitě osvětlení získáme z fotorezistoru, který je připojen k analogovému vstupu na Arduino. Arduino převede analogové hodnoty přes A/D převodník do digitálních hodnot 0 - 1023 hodnot. Je na programátorovi, jak hodnotu dále zpracuje. Experimentálně bylo zjištěno, že hodnota 0 - 10 odpovídá tmě (přibližně do 1 lx), hodnota 150 (přibližně 20 lx) umělému osvětlení v místnosti ve vzdálenosti 3 metry od žárovky 3 \* 60 W, hodnota 250 (přibližně 50 lx) odpovídá přirozenému světlu v místnosti 3 metry od okna, kdy slunce je za mrakem, hodnota 450 (přibližně 230 lx) odpovídá místnosti ozářené sluncem a hodnota 930 (přibližně 30 000 lx) přímému svícení ze vzdálenosti 5 cm cyklistickou svítlnou, kde se nachází 5 LED diod.

Zjednodušený zdrojový kód:

```
1 int sensorPinLight = A4;
2 int LightValue= analogRead(sensorPinLight);
```

### 4.2.5 Otevřené/zavřené dveře nebo okno

Pro připojení magnetického dveřního spínače Arduino si můžeme vybrat, zda zvolíme připojení k analogovému nebo digitálnímu pinu na Arduino. Pokud zvolíme digitální pin, tak za nás mikrokontrolér rozhodne otevření a zavření. U připojení k analogovému pinu

musíme zvolit hranici pro rozhodování v kódu. Protože byly použity všechny digitální piny, bylo zvoleno připojení k analogovému pinu. Dva metry dlouhý kabel připojený k senzoru vykazoval rušení, které bylo zaznamenáno na Arduinu do hodnoty 18 odpovídající přibližně 0,09 V. Pro rozhodnutí otevřeno/zavřeno byla zvolena hodnota 150 odpovídající 0,73 V.

Zjednodušený zdrojový kód:

```
1 const int sensorPinDvere = A3; //označení pinu
2 int i_sensorPinDoor = analogRead(sensorPinDvere); //čtení hodnoty
3 bool b_sensorPinDoor = (!(i_sensorPinDoor > 150));
```

## 4.2.6 Displej

Bylo použito zapojení 4 bitů z důvodu šetření digitálních pinů na straně Arduina.

Zjednodušený zdrojový kód:

```
1 lcd.begin(16, 2);           //inicializace počet znaků a počet řádků
2 lcd.setCursor(2, 0);       //nastavení kurzoru
3 float temper = getTemperature(); //zjištění aktuální teploty
4 lcd.print(temper);         //zobrazení na displeji
```

## 4.3 Uživatelské rozhraní

### 4.3.1 Přihlášení

Přihlášení uživatele zajišťuje stránka s přihlášením do uživatelské části systému. Pokud uživatel nemá účet, může se registrovat po kliknutí na link registrace a následně je přesměrován na stránku s registrací. Uživatel zadá přihlašovací jméno a heslo. Uživatelské jméno a heslo se solí se převede na hash (viz kapitola 6.1.4 Ukládání hesel) a porovná v databázi. V případě nenalezení záznamu v databázi je vypsán text, že přihlášení nebylo úspěšné. Pokud byl uživatel autorizován, je přesměrován na stránku home.aspx a aplikace si v session drží informace o uživateli.

### 4.3.2 Registrace

Nový uživatel se musí registrovat. Po registraci se může přihlásit do systému. Při registraci je uložen záznam do databáze a vytvořen otisk hesla.

**OVLÁDÁNÍ A KONTROLA DOMÁCNOSTI PŘES INTERNET** Uživatel nepřihlášen [Log In](#)

Home   Uživatelský profil   Domov přehled   Přehled varování   Domov nastavení   O aplikaci

**Přihlašovací informace**

Uživatelské jméno:

E-mail:  \*

Heslo:

Heslo znovu:

Obr. 22. Registrace uživatele

### 4.3.3 Home

Stránka obsahuje uvítání do systému. V budoucnu by na stránce mohly být informace, které by se četly z databáze. Správce systému by informace mohl pohodlně zadávat přes webové rozhraní, nejlépe přes editor podobný MS Word, například Wysiwyg editor nebo FCKeditor aj., a to bez znalosti HTML.

### 4.3.4 Uživatelský profil

Uživatel si může měnit nastavení adresy, e-mailu a telefonu. Dále je zobrazena IP adresa a datum posledního přihlášení. Informace jsou uloženy v databázi.

**OVLÁDÁNÍ A KONTROLA DOMÁCNOSTI PŘES INTERNET** Uživatel: jirkaho [logout](#)

Home   Uživatelský profil   Domov přehled   Přehled varování   Domov nastavení   O aplikaci

**UŽIVATELSKÝ PROFIL**

Uživatel:

Poslední přihlášení: 30.4.2012 23:07:42

Poslední IP: 94.113.222.166

Adresa:

E\_mail:

Město:

PSČ:

Telefon:

Obr. 23. Editace uživatelského profilu

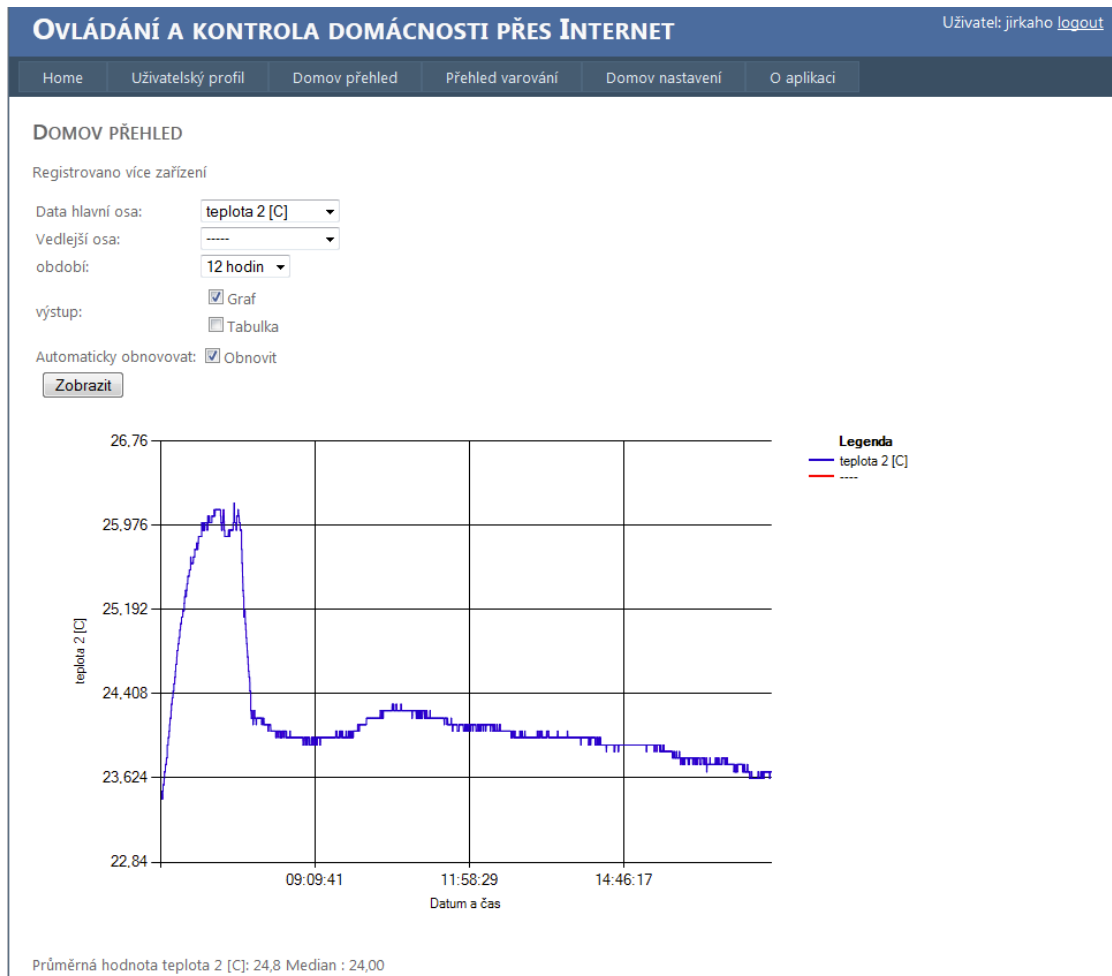
#### 4.3.5 Domov přehled

Stránka obsahuje přehled domácnosti za zvolené období. Uživatel si může zvolit, z jakých hodnot chce zobrazit výstupní data. Výstup je možné zvolit ve formátu grafu nebo tabulky.

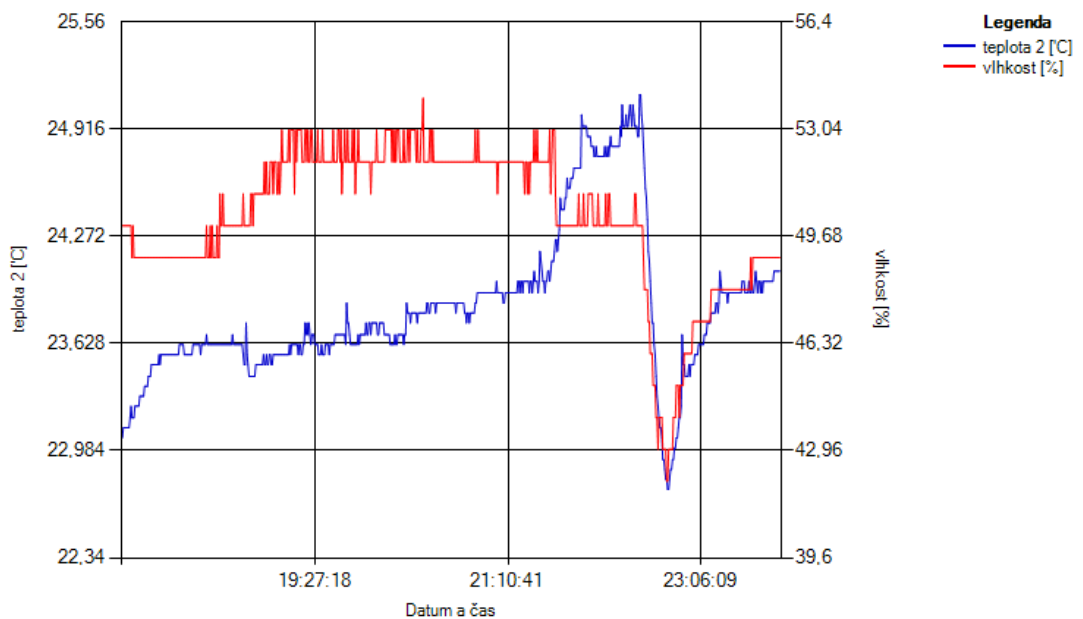
Graf je vytvořen pomocí technologie Chart Controls obsahující Framework 4.0. V grafu si uživatel může vykreslit až dva parametry současně a zvolit si časový úsek, za který chce data zobrazit. Každý parametr bude zobrazen na jiné ose. Názvy datových os jsou brány z databáze ke konkrétnímu Arduinu. Uživatel si může nechat graf pomocí technologie AJAX automaticky obnovovat.

Pod grafem je uvedena průměrná hodnota a hodnota mediánu. Medián je hodnota, která dělí řadu podle velikosti seřazených výsledků na dvě stejně početné poloviny. Jestliže  $n$  je sudé číslo, pak medián je jakékoliv číslo z intervalu  $(x_{n/2}, x_{n/2+1})$  [20].

Tabulka obsahuje pouze platné hodnoty, které jsou definovány k Arduinu. Tabulka vždy obsahuje čas záznamu hodnot, a pak sloupce definované v nastavení Arduina. Uživatel si může rovněž nechat tabulku pomocí technologie AJAX automaticky obnovovat.



Obr. 24. Webová stránka Domov přehled s grafem



Obr. 25. Stav domácnosti vyjádřený grafem s hodnotami na obou osách

Čas	test	teplota 1 [°C]	světlo	vlhkost [%]	test -náhodné č.	teplota 2 [°C]
16.4.2012 17:51:37	12345	23	79	50	875	23,18
16.4.2012 17:52:07	12345	23	83	50	875	23,25
16.4.2012 17:52:37	12345	23	82	49	875	23,18
16.4.2012 17:53:07	12345	23	82	50	875	23,18
16.4.2012 17:53:37	12345	23	82	49	875	23,18
16.4.2012 17:54:07	12345	23	81	49	875	23,25

Obr. 26. Tabulka hodnot senzorů

### 4.3.6 Přehled varování

Stránka obsahuje výpis všech varování. Varování jsou rozdělena na varování z jednotky (viz 5.1.2 Posílání stavu alarm) a varování uživatelsky definované (viz. 5.1.1 Periodické posílání aktuálního stavu). Administrátor zvolí názvy sloupců a v tabulce se zobrazí jen pojmenované sloupce a hodnoty senzoru v daný okamžik, kdy byly přesáhnuty mezní hodnoty. Má-li uživatel povolené notifikace při překročení hodnot, je v uvedené časy upozorněn notifikací.

OVLÁDÁNÍ A KONTROLA DOMÁCNOSTI PŘES INTERNET						Uživatel: jirkaho <a href="#">logout</a>
Home	Uživatelský profil	Domov přehled	Přehled varování	Domov nastavení	O aplikaci	
<b>PŘEHLED VAROVÁNÍ</b>						
<b>VAROVÁNÍ Z JEDNOTKY</b>			<b>UŽIVATELSKÉ ALERTY:</b>			
Čas	Dveře	Tlačítko	Čas	Hodnota	Název	
30.4.2012 23:21:02	1023	1	30.4.2012 14:47:08	28,06	teplota 2 [C]	
30.4.2012 19:58:49	0	1023	30.4.2012 14:45:08	28,25	teplota 2 [C]	
30.4.2012 14:56:05	0	1023	30.4.2012 14:43:07	28,18	teplota 2 [C]	
29.4.2012 13:50:55	1	1023	30.4.2012 14:41:07	28,31	teplota 2 [C]	
29.4.2012 13:48:58	0	1023	30.4.2012 14:39:07	28,06	teplota 2 [C]	
29.4.2012 13:47:49	0	1023	30.4.2012 14:20:35	28,06	teplota 2 [C]	
29.4.2012 12:05:27	0	1023	30.4.2012 13:36:00	28,06	teplota 2 [C]	
29.4.2012 12:04:57	0	1023	30.4.2012 13:27:59	28,18	teplota 2 [C]	
29.4.2012 8:45:46	8	1023	30.4.2012 11:44:47	28,06	teplota 2 [C]	
28.4.2012 19:58:56	0	1023	29.4.2012 10:21:43	26,31	teplota 2 [C]	
28.4.2012 19:18:08	0	1023	29.4.2012 10:19:43	26,31	teplota 2 [C]	
28.4.2012 19:15:40	6	1023	29.4.2012 10:17:43	26,12	teplota 2 [C]	
28.4.2012 19:08:58	0	1023	29.4.2012 10:15:42	26,06	teplota 2 [C]	
28.4.2012 18:48:19	0	1023	29.4.2012 10:10:42	26,06	teplota 2 [C]	
28.4.2012 14:32:28	0	1023	29.4.2012 10:05:41	26,06	teplota 2 [C]	
1 2 3 4 5 6 7 8 9 10 ...			1 2			

Obr. 27. Přehled varování

### 4.3.7 Domov nastavení

Stránka obsahuje seznam Arduin a ke každému Arduinu si uživatel může nastavit alerty. Alert definuje z názvu hodnot sloupce v nastavení Arduina, které nejsou prázdné a dále uživatel definuje operátor, který může nabýt hodnot větší, menší a rovná se a hodnotu k porovnání. Uživatel může alert zakázat nebo nastavit odesílání zprávy při překročení hodnot, a to formou e-mailu. Pokud jsou splněny podmínky pro uživatelský alert, systém jej zaznamená a uživatel si může zobrazit seznam alertů na stránce Přehled varování (viz 4.3.6 Přehled varování).

Poslední tabulka zobrazuje uživateli příkazy pro jednotku Arduino. Tyto příkazy definuje administrátor systému a Arduino musí být pro tuto činnost předem naprogramované a zapojené (viz 5.2 Webový server směrem k Arduinu). Pokud neexistují příkazy, tak tabulka se uživateli nezobrazí.

**OVLÁDÁNÍ A KONTROLA DOMÁCNOSTI PŘES INTERNET** Uživatel: jirkaho [logout](#)

Home    Uživatelský profil    Domov přehled    Přehled varování    Domov nastavení    O aplikaci

**DOMÁCÍ JEDNOTKA NASTAVENÍ**

Id	Poznámka	Datum vložení	Arduino ID	IP
<a href="#">Vybrat</a> 1		24.2012 0:00:00	100	94.113.222.166
<a href="#">Vybrat</a> 7		1.5.2012 14:03:57	-1	

Uživatelské Alerty:

Id	Arduino ID	Název	Operace	Hodnota	Změnil	Povoleno	Zpráva uživateli
<a href="#">Edit</a> 5	100	teplota 2 [C]	>	28		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">Smazat</a>
<a href="#">Edit</a> 2	100	světlo	>	500		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <a href="#">Smazat</a>

Příkazy pro Arduino:

ID	Název
<a href="#">Provést</a> 1	Svetlo

Obr. 28. Nastavení alertů a posílání příkazů Arduinu

### 4.3.8 O aplikaci

Stránka obsahuje stručnou informaci o vytvořené internetové aplikaci.

### 4.3.9 Administrace

Tato stránka je zobrazena pouze přihlášenému administrátorovi. Administrátor má na stránce zobrazený seznam uživatelů, seznam jednotek Arduino a příkazy pro jednotky.

Administrátor může uživatele smazat ze systému nebo vyfiltrovat jeho registrované jednotky, dále může administrátor vybrané jednotce Arduina editovat příkazy, které může uživatel použít. Tento záznam se skládá z volání, na které je Arduino naprogramováno a názvu, který vidí uživatel.

Uživatel: admin [logout](#)

## OVLÁDÁNÍ A KONTROLA DOMÁCOSTI PŘES INTERNET

Home
Uživatelský profil
Domov přehled
Přehled varování
Domov nastavení
O aplikaci
Administrace

### ADMINISTRACE

Uživatelé:

	Id	Nazev	PosledniPrihlaseni	PosledniIP	Adresa	Mesto	PSC	telefon	e_mail	kontakt	login	
<a href="#">Vybrat</a>	1		1.5.2012 14:13:34	::1	Ulice 123	Brno	60200	123123123	horak@intelimedix.eu		jirkaho	<a href="#">Delete</a>
<a href="#">Vybrat</a>	2		8.1.2012 22:46:19	::1					horakjiri@atlas.cz		jirkaho1	<a href="#">Delete</a>
<a href="#">Vybrat</a>	26		1.5.2012 14:19:46	::1							admin	<a href="#">Delete</a>
<a href="#">Vybrat</a>	27		30.4.2012 22:28:50	::1					horakjiri@atlas.cz		jirkaho3	<a href="#">Delete</a>

Jednotky Arduino:

	Id	Poznamka	User_ID	login	IP	Insert_Date	Enable	Arduino_ID	
<a href="#">Select</a>	1		1		94.113.222.166	2.4.2012 0:00:00	True	100	<a href="#">Delete</a>
<a href="#">Select</a>	7		1			1.5.2012 14:03:57	False	-1	<a href="#">Delete</a>

Příkazy jednotky Arduino:

	ID	Arduino_ID	Akce název	Povel jednotce	
<a href="#">Edit</a>	1	100	Svetlo	svetlo	<a href="#">Delete</a>

Obr. 29. Nastavení pro administrátora

## 4.4 Rozhraní pro ukládání hodnot na webovém serveru

Arduino posílá stavy senzorů ve formě hodnot na dvě webové stránky, které se nacházejí na webovém serveru. Periodicky posílané hodnoty se posílají na setData.aspx a pro hodnoty v alarmu se hodnoty posílají na stránku set\_alarm.aspx. Tyto stránky nemají uživatelské rozhraní.

### 4.4.1 Periodicky posílané hodnoty

Stránka po zavolání uloží hodnoty z parametru URL do tabulky measurement\_values a zkontroluje, jestli některé hodnoty nejsou ve stavu uživatelského alarmu. Pokud systém vyhodnotí uživatelský alarm, hodnota nebo hodnoty jsou navíc uloženy do tabulky uživatelských alarmů - alarm\_values\_user. Systém zjistí, zda má uživatel nastaveno posílání notifikací a případně systém pošle e-mail na uživatelem definovanou adresu.

#### 4.4.2 Hodnoty typu alarm

Stránka po zavolání uloží hodnoty z parametru URL do tabulky alarm\_values a pošle notifikaci ve formě e-mailu na uživatelem definovanou adresu.

### 4.5 Instalace a konfigurace

#### 4.5.1 Arduino

Arduino se senzory a jinými komponenty se musí sestavit a zapojit s ohledem na očekávanou funkčnost. Mikrokontroléru se musí nahrát zkompileovaný program pro konkrétní verzi Arduina, kde jsou správně vyplněné konstanty nastavení a zdrojový kód zohledňující použité senzory a jiné komponenty.

Nástavení Arduina:

- IP adresa Arduina (IPv4).
- IP adresa webového serveru (IPv4).
- IP adresa brány (IPv4).
- MAC adresa (musí být v síti zvolena unikátní adresa).
- Port, na kterém Arduino naslouchá (zvolen port 10000).

#### 4.5.2 Web server

Publikovaná verze webového serveru se zkopíruje na server, kde se v Internetové informační službě nastaví nová webová aplikace a její příslušné vlastnosti. V standardním konfiguračním souboru web.config se nastavují hodnoty pro běh aplikace.

Tab. 10. Nastavení webového serveru

Název	Příklad hodnoty	Poznámka
LogDir	d:\TEMP\	Adresář, kde jsou uloženy informace o běhu aplikace
mailFrom	horakjiri@atlas.cz	E-mail adresa odchozí pošty
ChartImageHandler	storage=file;timeout=20;dir=c:\TempImageFiles\;	Nastavení pro grafy
UserAlarmSendNext MailMin	2	Nastavuje čas, kdy nebude uživateli zaslán opětovně e-mail, pokud v tu dobu jsou hodnoty ve stavu alarm
ArduinoPort	10000	Port, na kterém naslouchá Arduino
connectionString	server=192.168.1.2;uid=xx;pwd=xx;database=Arduino	Připojení do relační databáze MS SQL
mailServer	host="mail" password="xx" userName="xx"	Nastavení zvoleného poštovního serveru

## 5 KOMUNIKACE ARDUINA A WEBOVÉHO SERVERU

Komunikace mezi mikrokontrolérem a webovým serverem probíhá přes síť Internet nebo Intranet. Je použit protokol HTTP a metoda GET.

### 5.1 Arduino směrem na webový server

Arduino posílá stav senzorů protokolem HTTP přes metodu GET. V URL jsou jednoznačně identifikovatelné jednotlivé objekty. Jednotlivými objekty rozumíme hodnoty stavů jednotlivých senzorů. Tyto stavy jsou po přijetí serverem následně uloženy do databáze. Periodicky posílaný stav se posílá na stránku webového serveru setData.aspx a stav alarm na stránku set\_alarm.aspx.

Arduino musí být předem naprogramováno a v kódu nastaveny tyto hodnoty důležité pro komunikaci:

#### 5.1.1 Periodické posílání aktuálního stavu

Periodické posílání stavu senzorů slouží pro přehled nad domácností přes webové rozhraní. Stav senzorů jsou posílány Arduinem každých 30 sekund a následně ukládány do databáze na straně webového serveru. Těchto stavů může být až 6 celočíselných a 3 s pohyblivou desetinnou čárkou. Na serveru lze definovat uživatelský alert, kdy bude uživatel upozorněn, až hodnota překročí předem nastavenou hodnotu. Pokud by hodnota nastala krátkodobě (do 30 sekund, kdy byla zvolena hodnota pro periodické posílání požadavku z Arduina), uživatel nemusí být upozorněn, protože tento stav se nemusí poslat na webový server. Výhoda tohoto řešení spočívá v rychlé modifikaci uživatelem pomocí webového prohlížeče na webovém serveru.

Příklad posílaného periodického stavu:

```
/Arduino/setData.aspx?f1=24.37&p1=12345&p2=24&p3=94&p4=59&p5=1308&ID=100  
&pk=konec
```

- f1 až f3 – hodnoty s desetinnou čárkou typu Float
- p1 až p6 – hodnoty celočíselné typu Integer
- ID – identifikace mikrokontroléru
- Pk – nepovinný parametr

Z uvedených parametrů vyplývá, že sestavit Arduino se stejným ID není problém. Pokud by v budoucnu nastal problém s podvodným Arduinem nebo falešnými zprávami na Internetu, první ochrana, která částečně eliminuje tento problém, je zavedení komunikace jen na přiřazené, povolené IP adresy. Toto řešení vyžaduje pevnou IP adresu na straně Arduina nebo lokální síť, kde se Arduino nachází. V České republice ale není mnoho poskytovatelů, kteří poskytují bezplatně veřejnou statickou IP adresu. Pro zlepšení bezpečnosti je nutná samostatná analýza.

### 5.1.2 Posílání stavu alarm

Pokud je potřeba nadefinovat přímo na Arduinu hodnoty označující nějakou událost, kterou má Arduino okamžitě vyhodnotit a poslat na webový server, slouží k tomu alerty. Tyto stavy jsou vyhodnocovány v každém cyklu programu na Arduinu a v případě překročení naprogramovaných hodnot, se aktuální hodnoty posílají okamžitě na webový server. Systém je připraven až na 3 celočíselné hodnoty. Výhoda tohoto řešení spočívá v okamžité reakci na událost, ovšem pokud uživatel potřebuje změnit nastavení, musí se Arduino přeprogramovat.

Příklad posílaného stavu alarm:

```
/Arduino/set_alarm.aspx?a1=24&a2=94&ID=100
```

## 5.2 Webový server směrem k Arduinu

Webový server komunikuje s mikrokontrolérem na IP adrese mikrokontroléru. Webový server si při periodickém požadavku od Arduina uloží IP adresu. Každý mikrokontrolér musí být předem naprogramován a sestaven na tyto akce a také příslušně zapojen do systému, který bude ovládat.

Příkaz pro zapnutí nebo vypnutí světla:

```
/svetlo např. http://192.168.12.100:10000/svetlo
```

Tento příkaz neguje proměnnou v mikrokontroléru a dle výsledku posílá na digitální pin logickou nulu nebo jedničku. V tomto případě musí být na pin připojeno relé, které sepne příslušný obvod, zde světlo. Pro ukázkou funkce může být použita pouze LED dioda.

Nevýhoda tohoto řešení spočívá v tom, že Arduino musí mít veřejnou IP adresu nebo povolený port, pokud je v síti za síťovým prvkem NAT. Ten musí mít veřejnou IP a na síťovém prvku FireWall povolený příslušný port.

Zjednodušený zdrojový kód Arduino – hlavní smyčka:

```

1     else if (strncmp("/svetlo", (char *)&(buf[dat_p+4]), 7) == 0) {
2         dat_p = print_webpageSvetlo(buf);
3         es.ES_www_server_reply(buf, dat_p); }

```

Zdrojový kód procedura print\_webpageSvetlo:

```

1 uint16_t print_webpageSvetlo(uint8_t *buf)
2 {     uint16_t plen;
3     plen = http200ok();
4     plen = es.ES_fill_tcp_data_p(buf, plen, PSTR(
    "<html><head><title>svetlo</title></head><body>"));
5     plen = es.ES_fill_tcp_data_p(buf, plen, PSTR("<h1> Svetlo </h1>"));
6     zapnutoS = !zapnutoS;
7     if (zapnutoS) { digitalWrite(PinRele, HIGH);
8         plen = es.ES_fill_tcp_data_p(buf, plen, PSTR("Zapnuto"));
9     }
10    else { digitalWrite(PinRele, LOW);
11        plen = es.ES_fill_tcp_data_p(buf, plen, PSTR("Vypnuto"));
12    }
13    plen = es.ES_fill_tcp_data_p(buf, plen, PSTR("</body></html>"));
14    return(plen);
15 }

```

Zjednodušený C# kód

```

1 nNastroj.tArduino_extension tArduino = new
  nNastroj.tArduino_extension();
2 tArduino.ReaderArduinoId( ID_Arduino);
3 string response = commonClass.HttpGet(tArduino.IP + ":" +
  System.Configuration.ConfigurationManager.AppSettings["ArduinoPort"]
  , tArduino_command.Action_command);

```

Zjednodušený C# kód z metody commonClass.HttpGet

```

1 WebRequest req = WebRequest.Create("http://" + m_URI + "/" +
  m_Parameters);
2 req.ContentType = "text/html"; req.Method = "GET";
3 WebResponse resp = (WebResponse) req.GetResponse();
4 try { WebResponse resp = (WebResponse) req.GetResponse();
5     if (resp == null) { return null; }
6     StreamReader sr = new StreamReader(resp.GetResponseStream());
7     return sr.ReadToEnd().Trim(); }
8 catch (WebException we)
9     { ...

```

## 6 BEZPEČNOST

Každý webový projekt by měl být zabezpečen proti operacím, na něž nemá uživatel přístupující k systému oprávnění. Práce popisuje systém, který může obsahovat citlivé soukromé informace uživatelů, např. kdy uživatel není doma, kdy se mu rozsvítí světla, která jsou jen dálkově zapnuta nebo naprogramována, nebo jaké jsou jeho zvyky. Webový server lze dobře zabezpečit běžnými metodami. Je možné použít protokol SSL společně s certifikátem. Komunikace se serverem bude šifrovaná a zabezpečená. Na straně Arduina je problém s výkonem a pamětí, tudíž nelze využít komunikaci se serverem přes protokol SSL.

### 6.1.1 Arduino

Arduino má příliš malou paměť a malý výkon pro komunikaci přes SSL. Pro bezpečnou komunikaci přes Internet lze použít síťový prvek s podporou zabezpečeného síťového tunelování, který komunikaci Arduina a webového serveru zašifruje.

### 6.1.2 Webové rozhraní

Na webovém serveru bude uživatel přistupovat k aktuálnímu stavu, k nastavení a historii domácnosti. Je tedy nutné mít server dobře zabezpečený a odolný proti útokům typu:

- SQL Injection
- Cross Site Scripting (XSS)
- Krádež session, cookies
- Jiné

### 6.1.3 Databáze

Databázový server by neměl být přístupný z Internetu, ale pouze z lokální sítě. Musí být vypnuty účty, které jsou použity pro instalaci a u každé instalace jsou shodné.

### 6.1.4 Ukládání hesel

Kvůli bezpečnosti je vhodné uživatelská hesla měnit na otisk hesla (např. hash MD5), aby nebyla v otevřené podobě uložena v databázi. Pokud už dojde k průniku do databáze nebo si zobrazí hesla interní zaměstnanec, je velká pravděpodobnost, že heslo nepůjde zpět zrekonstruovat. Nevýhodou metody je to, že pokud uživatel zapomene heslo, musí si sám zadat nové heslo nebo musí systém uživateli nové heslo vygenerovat.

K heslu přidáme dlouhý tajný řetězec a z tohoto řetězce vytváříme otisk hesla. I když uživatel zadá krátké heslo, získáme poměrně dlouhý řetězec (zvaný sůl nebo třešnička). Pokud někdo bude rekonstruovat heslo, doba výpočtu k porovnání hesel a hash nebude přijatelná a snižuje se tak možnost použití databází hash a jejich řetězců na Internetu.

Například na <http://www.techfreakstuff.com/2010/02/tools-decrypt-md5-hash-online.html> je uvedeno několik dekryptorů hash. Zadáme-li hash s 5 znaky, s velkou pravděpodobností dostaneme původní heslo.

Tab. 11. Heslo do MD5 a zpětná rekonstrukce

Řetězec Jirka	Hash MD5: 01b61ad8993fced8614cfa80289d4483
Např. <a href="http://md5.my-addr.com/md5_decrypt-md5_cracker_online/md5_decoder_tool.php">http://md5.my-addr.com/md5_decrypt-md5_cracker_online/md5_decoder_tool.php</a>	Zpět řetězec Jirka

## 7 TESTOVÁNÍ

Zelinka uvádí: „*Testováním budeme rozumět veškeré aktivity, jejichž cílem je odhalení chyb. Může mít podobu posouzení (např. techniky procházení či inspekce), v případě implementace kvality zdrojového kódu, nebo může být založeno na spuštění proveditelného kódu. Ve druhém případě můžeme rozlišit dvě varianty – tzv. Testování ze specifikace (black - box testing) (také označované jako funkční) a testování z kódu (white - box) (také označované jako strukturní). V prvním případě se při návrhu testu vychází ze specifikace testovaného programu (co má umět), zatímco ve druhém se vychází ze znalosti zdrojového kódu.*“ [21, s. 18]

### 7.1 Testování mikrokontroléru

Testování mikrokontroléru byl návrat o 20 let zpět v moderním, běžném programování pro platformu PC. Chyběly nástroje, které jsou v objektově orientovaném programování ve Visual Studiu běžné a programátoři s nimi pracují. Nástroje jako krokování, náhled na vlastnosti objektů nebo proměnných, nebyly dostupné. Když bylo něco takového potřeba, musel být kód upraven pro pomocný výpis. Tento výpis lze zobrazit na displeji mikrokontroléru, poslat jej přes sériovou linku nebo jej uložit do ROM paměti zařízení. Nejjednodušší bylo posílat informace přes sériovou linku a vytvořit jednoduchý program na zobrazení dat přijatých sériovou linkou. Vývojové prostředí pro mikrokontrolér Arduina obsahuje monitor pro sériovou linku, který vždy pošle příkaz k restartu zařízení. Zařízení se inicializuje a běží znovu. Bylo nutné se připojit na již několik hodin běžící mikrokontrolér.

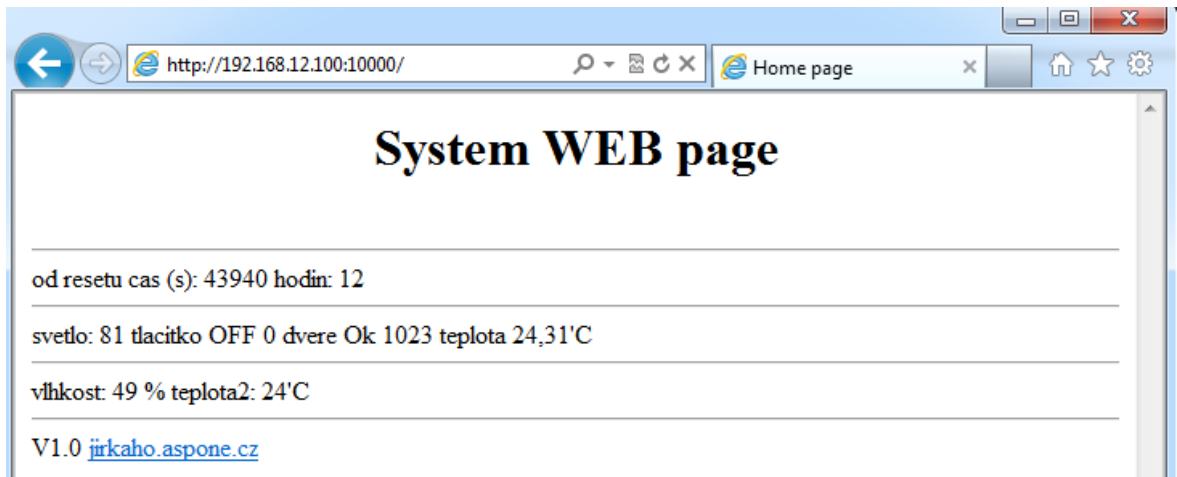
Jednou z chyb, která byla odhalena v průběhu testování, byl problém ve slabém zdroji napájení. Byl použit adaptér do zásuvky, kdy z 230 V střídavého napětí, adaptér konvertuje napětí na stejnosměrných 5 V – napájení pro zařízení přes USB konektor. Tento adaptér slouží pro nabíjení handsfree a má pouze 150 mA. Zpočátku vývoje, kdy nebylo použito mnoho senzorů, vše fungovalo a měřením odběru celého zařízení bylo usouzeno, že zmíněný adaptér stačí dodávat proud. Postupně byly přidány další senzory a posléze i síťová karta. Chyba se projevovala tak, že mikrokontrolér pracoval, ale po čase přestala fungovat síťová část. Toto nastalo občas po minutách, jindy po hodinách. Připojený displej stále ukazoval aktualizované hodnoty. Vznikla domněnka, že „přetekla“ nějaká proměnná. Procházením zdrojového kódu Arduina nebyla nalezena příčina. Po několika týdnech pozorování byl kód optimalizován. Zařízení přestalo číst hodnoty senzorů na vyžádání, což

nastalo při požadavku webového prohlížeče nebo při posílání na server. Zařízení bylo přeprogramováno na čtení hodnot ze senzorů periodicky po několika sekundách a ukládání do sdílených proměnných. Z těchto proměnných se pak údaje zapsaly do uživatelem vyžádané HTML stránky nebo poslaly na server. Optimalizace výrazně urychlila sestavení HTML stránky vyžádané prohlížečem uživatele. Po připojení k notebooku vše fungovalo (přímé připojení z počítače přes USB 2.0 může dodávat proud okolo 500 mA). Po použití zmíněného adaptéru po několika dotazech na HTML stránku zařízení, přestala síťová část opět fungovat. Toto chování vedlo k podezření, že problém může být v napájení a po výměně síťového adaptéru se chyba dále neprojevovala. Později bylo změřeno, že Arduino odebírá proud o velikosti okolo 200 mA.

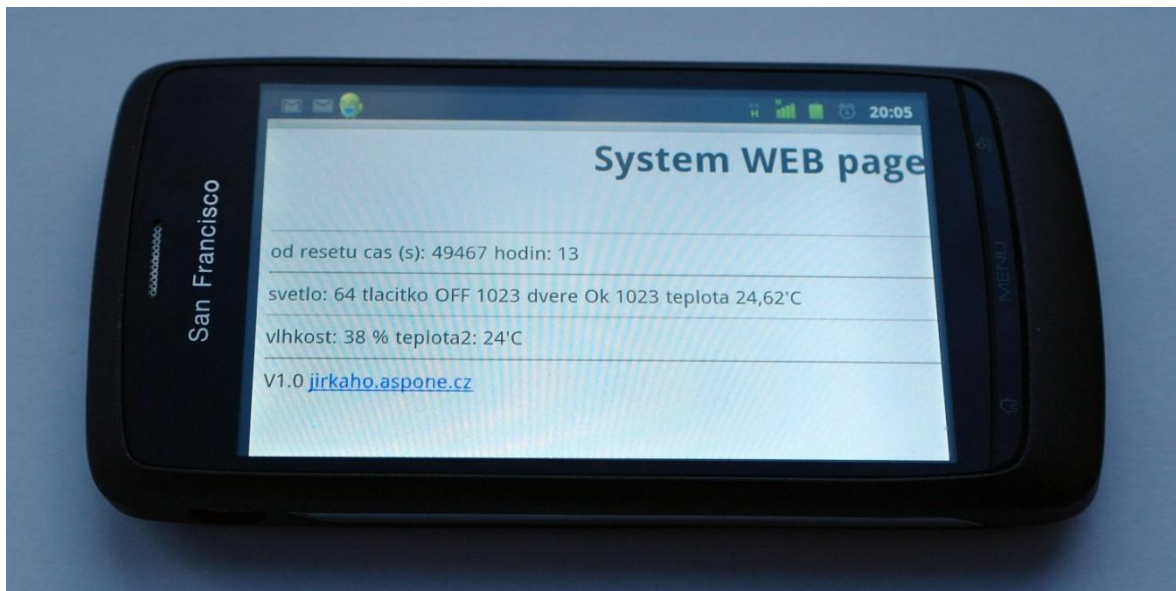
Jedním z dalších problémů byla velikost paměti RAM. 2 kB paměti není mnoho a například pro zobrazení HTML stránky pro uživatele si Arduino alokuje okolo 1 kB v paměti RAM, a to jen pro sestavení HTML.

Pro testování mikrokontroléru byly zvoleny dva způsoby - výstup ve formě HTML stránky a zobrazení hodnot na LCD panelu. Mikrokontrolér má svoji lokální IP adresu a port, na kterém naslouchá. Je přístupný z lokální sítě. Na příslušném domácím routeru lze nastavit v možnostech NAT přeposílání požadavků z Internetu. Tímto způsobem bylo možné mikrokontrolér testovat i z prostředí Internetu, například ze zaměstnání nebo mobilního chytrého telefonu s přístupem na Internet.

Dále se na displeji připojenému k Arduino zobrazoval údaj, zda si někdo vyžádal HTML stránku se stavy senzorů nebo Arduino posílá svůj stav na webový server. Takto byl neustále přehled o funkčnosti zařízení a o tom, jestli funguje připojení k webovému serveru a samotný webový server. Z testování vyplynulo, že tento údaj je velmi důležitý a v případné verzi Arduina bez displeje je vhodné pro upozornění uživatele použít LED diodu.



Obr. 30. Testovací HTML stránka Arduina



Obr. 31. Testovací HTML stránka Arduina na mobilním telefonu

## 7.2 Testování web serveru

Testování webového serveru bylo rozděleno na několik částí. První část se zaměřila na ukládání a optimalizaci získávání a ukládání dat z Arduina, a to hodnoty periodicky ukládané a hodnoty typu alarm. Postupně bylo do databáze uloženo přes 250 tisíc záznamů periodicky zapsaných a přes 1000 záznamů typu alarm. V průběhu testování byl lehce upraven design databázových tabulek a upraven kód na straně serveru a Arduina. Úpravy zlepšily návrh a do budoucna zajistily, že nebude nutné měnit strukturu databáze a zdrojový kód databázové vrstvy.

Následné testování bylo zaměřeno na uživatelské rozhraní. Zjištěné nedostatky byly následně opraveny a znovu otestovány.

Samostatně byly testovány uživatelské alerty. Bylo otestováno, že alerty jsou správně implementované a dle kritérií jsou správně vyhodnoceny stavy z Arduina a na získané hodnoty server správně reaguje. Při testování se projevila chyba, kdy byl uživatel neustále obtěžován (každé 2 minuty) notifikací formou e-mailu, pokud trval dále stav alarm. Tato chyba byla opravena a v aktuální verzi uživatel dostane pouze notifikaci, kdy nastal stav alarm. Pokud hodnota splňuje podmínku alarm, tak již uživatel nedostává nadále zprávu. Uživatel může stav dále sledovat přes webové rozhraní.

Při testování bylo objeveno několik chyb, ale všechny se podařilo úspěšně opravit.

## 8 MOŽNÉ ROZŠÍŘENÍ

Uvedeme možné rozšíření diplomové práce, které by mohlo být zajímavé a užitečné.

### 8.1 Vytvoření aplikace pro chytrý mobilní telefon

Jedním z možných rozšíření této práce by mohlo být vyvinutí aplikace pro chytrý mobilní telefon týkající se ovládání a přehledu nad domácností. Tato aplikace by běžela na pozadí příslušného mobilního operačního systému. Pokud by se vyskytla nějaká událost, uživatel by byl upozorněn notifikací a mohl by okamžitě reagovat. Navíc by odpadlo neustálé přihlašování do internetové aplikace, protože aplikace by jednou prošla autorizací a uživatel by dále obtěžován (za předpokladu, kdy by si toto chování uživatel nezrušil). Aplikace by byla vytvořena dle trendů mobilních aplikací a byla by optimalizovaná pro dotykové ovládání. Funkčnost mobilní verze by měla mít podobné možnosti nastavení a přehledů jako internetová verze na webovém serveru. Lze očekávat, že by uživatelé využívali více mobilní verzi než přihlašování přes Internet. Mobilní aplikace by bezpečně komunikovala s webovým serverem, který by byl zdrojem dat pro mobilní aplikaci.

### 8.2 Vytvořit analýzu a implementovat bezpečnou komunikaci přes Internet

System neumožňuje bez přidavných síťových prvků bezpečnou komunikaci mezi webovým serverem a Arduinem po síti Internet. V současném stavu je možné komunikaci odposlechnout a případně vyslat falešné zprávy na oba směry - jak k webovému serveru, tak i k Arduino. Částečně je možné problém vyřešit naprogramováním tak, aby systém komunikoval pouze na přidělené IP adresy. V současné době většina poskytovatelů Internetu mění veřejné IP adresy nebo nemá veřejné IP adresy pro uživatele a uživatelé jsou za NAT síťovým prvkem, proto není systém takto implementován. Komunikaci uživatele s tenkým klientem a webovým serverem je možné běžnými prostředky dobře zabezpečit.

### 8.3 Vytvořit analýzu pro možnost komunikace Arduino a další Arduino (peer-to-peer)

Pokud by nebyla možnost komunikovat se serverem, bylo by možné upravit kód mikrokontroléru tak, aby spolu mohly jednotlivé jednotky Arduina komunikovat. Tato možnost by byla použitelná i v případě, kdy bychom chtěli zajistit komunikaci mezi

jednotlivými Arduiny a nechtěli bychom pokaždé ukládat hodnoty na server nebo by byly hodnoty jen pomocné pro další zpracování a posléze připravené pro odeslání na server.

#### **8.4 Nastavování hodnot alarm na straně Arduina**

Bylo by možné rozšířit zdrojový kód Arduina tak, aby se mohly hodnoty a nastavení zadávat i přes webové rozhraní a posílat do Arduina, kde se uloží do paměti. Ideální stav by byl, aby se tyto hodnoty mohly měnit z webového serveru. Pak je ale nutné zvýšit zabezpečení Arduina (viz. 8.2 Vytvořit analýzu a implementovat bezpečnou komunikaci přes Internet).

#### **8.5 Vytvořit stránku s nastavením na Arduinu**

Arduino by mohlo mít jednoduchou stránku s nastavením jeho hodnot. Tak by si mohl uživatel sám nastavit hodnoty přes webové rozhraní i mimo zdrojový kód. Tato problematika by byla vhodná pro další výzkum, protože se musí vyřešit otázka paměti, zabezpečení, autorizace, umístění a zapojení hardwarového tlačítka resetu pro nastavení zařízení do továrního nastavení a další, s tím spojené otázky implementace. Bylo by možné Arduino naprogramovat pro přijetí některých hodnot nastavení z DHCP serveru.

## ZÁVĚR

Diplomová práce pojednává o možnostech, jak ovládat a kontrolovat domácnost. V teoretické části jsme se seznámili s mikrokontrolérem a platformou Arduino, senzory a návrhem webového serveru. V praktické části jsme ověřili, že je možné vytvořit ovládání a kontrolu domácnosti pomocí běžně dostupných komponent. Celý systém se skládá ze dvou podsystémů. Jedním z nich je mikrokontrolér Arduino umístěný v domácnosti a dalším je webový server. Síťová architektura je typu klient-server, to znamená, že Arduino připojených k webovému serveru může být více.

První podsystém je mikrokontrolér Arduino s připojenou síťovou kartou a připojenými senzory. K Arduino byly připojeny tyto senzory: teploměr, vlhkoměr, senzor intenzity světla, tlačítko a magnetický dveřní spínač. Dále byl připojen displej zobrazující některé hodnoty senzorů a dioda pro demonstraci zapnutí světla (v domácnosti by místo této diody bylo zapojeno elektromagnetické relé, které by sepnulo například světlo). Takto sestavený mikrokontrolér můžeme pořídit přibližně za 1300 Kč. Spotřeba elektrické energie tohoto mikrokontroléru bez zapnutého podsvícení displeje byla změřena přibližně na hodnotě 2,6 W. Uvažujeme-li aktuální cenu elektřiny 4,6 Kč za 1 kWh, stojí roční provoz přibližně 104 Kč a s podsvícením displeje 126 Kč při zanedbání ztrát napájecího zdroje. Zkompilovaný program pro Arduino má velikost 22,5 kB (32 kB je maximum). K Arduino lze připojit i jiné senzory, displeje nebo akční členy. Arduino může posílat předem naprogramované alerty (varování), které ale uživatel nemůže měnit. Například někdo otevře dveře nebo zmáčkne tlačítko. Arduino je připojené do počítačové sítě Ethernet, a tak není nutné zavádět další kabeláž. Zvolené sestavení naprogramovaného Arduina se senzory a síťovou kartou prošlo sedmiměsíčním testováním v reálném prostředí domácnosti, kdy došlo pouze k minimálním změnám. Arduino bylo rozšířeno o další senzory do současné podoby.

Druhým podsystém je webový server. Webový server je naprogramován na technologii .NET. Uživatel má přehled o aktuálním stavu nebo historii domácnosti. Může si zobrazit grafy nebo tabulky aktuálních nebo historických hodnot jednotlivých senzorů. Uživatel má možnost nastavovat a zobrazovat alerty (varování), které si sám nadefinuje a v případě splnění podmínky je systémem upozorněn. Přes webové rozhraní může uživatel poslat příkaz Arduino, které musí být naprogramováno a předem vybaveno elektrickými obvody. Uživatel tak například může vzdáleně zapnout nebo vypnout osvětlení pokoje a přes senzor

intenzity osvětlení ověřit, zda se tak stalo. Webový server je naprogramován tak, aby byl univerzální pro různé senzory nebo akční prvky připojené k Arduino, které se musí pro každý typ senzoru na Arduino naprogramovat.

V závěru práce jsou uvedena další možná rozšíření systému. Jediným problémem, který by bylo nutné vyřešit v případě rozsáhlejšího použití systému v prostředí sítě Internet, je to, že komunikace mezi Arduinem a webovým serverem bez použití síťových prvků, které umí vytvořit bezpečné síťové propojení, je posílána v otevřené podobě a není nijak chráněná.

Výsledky této diplomové práce byly částečně publikovány také v rámci konference Internet, bezpečnost a konkurenceschopnost organizací [22].

## ZÁVĚR V ANGLIČTINĚ

This diploma work discusses ways how to control and monitor a household. In the theoretical discussion, the microcontroller and Arduino platform is introduced as well as sensor functions and Web site design. In the practical part, we verified that it is possible to create a home management and monitoring system using commonly available components. The whole system consists of two subsystems. One of them is the Arduino microcontroller located in the home and the other one is a web server. Network architecture is based on the client-server model. This means that many Arduinos can be connected to a single or multiple web servers.

The first subsystem is connected to the Arduino microcontroller network card and its connected sensors. The Arduino is linked to the following sensors: temperature, humidity, light intensity, push buttons and magnetic door switches.

The Arduino is also connected to the monitor displaying some of the values of sensors and a diode to demonstrate the turning on or off of a light. The sensor would also be able to detect a change taking place directly at the magnetic relay (not just the diode). This makes it possible for the system to detect changes when someone turns on a light using the switch on the wall. A configured microcontroller can be purchased for around 1300 CZK. Power consumption of such a microcontroller without a fast display is measured on average to consume 2.6 W of electricity. If we consider the current price of electricity at 4.6 CZK per 1 kWh, the annual operation cost is approximately 104 CZK (or 126 CZK with a backlight). This doesn't take into account the loss incurred by the power supply. The compiled program for the Arduino is 22.5 kilobytes in size (a 32 kB maximum). Arduinos can be connected to other sensors, actuators and displays. Arduinos can send predefined alerts (warnings) to the microcontroller. Arduinos are connected to the internet via an Ethernet card. Any change made to the circuit through the diode or the electromagnetic relay, are recorded in the software. For example, when someone opens door or presses a button, the Arduino is aware of the change. Since these Arduinos are connected via Ethernet, no additional wiring is necessary. A sample Arduina programmed with sensors and a network card went through 7 months of testing in real households. Only minimal changes were required and the Arduino was extended to other sensors.

The second subsystem is a web server. The web site is coded using the popular .Net architecture. The user can access an overview of both the current status and historical

changes. You can display graphs or tables of current and historical values of the individual sensors. The user can define and display alerts (warnings) and, if met, then the system sends a notification. Using a simple web interface, a user can send commands to the Arduino, which must be programmed in advance and connected to electrical circuits. The user can remotely turn on or off light and via a room lighting sensor. The action can be verified using a luminance sensor. The web server is programmed to be universal for various sensors or actuators connected to the Arduino, but each type of sensor must be programmed.

In conclusion further possible extensions of the system are noted. The only problem that should be considered for a larger system is the public Internet usage for the communication between the Arduino and the web server. Currently the system uses unencrypted protocols so communications between components of the system are in plaintext which is not secure. However, it would be easy enough to tunnel these protocols over encrypted SSL connections.

The results of this thesis was also published in part in the conference Internet, competitiveness and organizational security [22].

**SEZNAM POUŽITÉ LITERATURY**

- [1] VALEŠ, Miroslav. *Inteligentní dům*. 2. vyd. Brno: ERA, 2008, 123 s. ISBN: 978-80-7366-137-3.
- [2] HARPER, Richard. *Inside the smart home*. London: Springer, 2003. ISBN 1-85233-688-9.
- [3] Odbor statistiky obyvatelstva. Předběžné výsledky Sčítání lidu, domů a bytů 2011 Česká republika. *Český statistický úřad*. [Online] 02 2012. [Citace: 27. 02 2012.] [http://www.czso.cz/csu/2012edicniplan.nsf/t/950049F8F7/\\$File/pvcr.pdf](http://www.czso.cz/csu/2012edicniplan.nsf/t/950049F8F7/$File/pvcr.pdf). ISBN 978-80-250-2145-3.
- [4] TOMAN, Karel. Decentralizované sběrníkové systémy. *TZB-info*. [Online] 02. 07 2007. [Citace: 20. 04 2012.] <http://www.tzb-info.cz/4213-decentralizovane-sbernicove-systemy>.
- [5] MERZ, Hermann, Thomas HANSEMANN a Christof HÜBNER. *Automatizované systémy budov: sdělovací systémy KNX/EIB, LON a BACnet*. Praha: Grada, 2008, 261 s. ISBN 978-80-247-2367-9.
- [6] Řídící systémy luxmate a příslušenství. *LUXPLAN*. [Online] [Citace: 26. 04 2012.] <http://www.luxplan.cz/cs/reseni/ridici-systemy-luxmate/>.
- [7] Redakce HW serveru. ZigBee - novinka na poli bezdrátové komunikace. *HW server s.r.o.* [Online] 08. 06 2005. [Citace: 26. 04 2012.] <http://www.hw.cz/Rozhrani/ART1299-ZigBee---novinka-na-poli-bezdratove-komunikace.html>.
- [8] AUGARTEN, Stan. *State of the art: a photographic history of the integrated circuit*. New Haven: Ticknor, 1983. ISBN 0-89919-195-9.
- [9] MALÝ, Martin. Arduino: vývojový kit pro hrátky s hardware. *Root*. [Online] Internet Info, s.r.o, 30. 09 2009. [Citace: 2012. 01 24.] <http://www.root.cz/clanky/arduino-vyvojovy-kit-pro-hratky-s-hardware/>.
- [10] Arduino Team. *arduino.cc*. [Online] [Citace: 1. 2 2012.] <http://arduino.cc>.
- [11] BANZI, Massimo. *Getting started with Arduino*. 1st ed. Sebastopol: O'Reilly Media, 2009. ISBN 978-0-596-15551-3.

- [12] Microchip Technology Inc. ENC28J60 Data Sheet. [Online] 2008. [Citace: 2. 04 2012.] <http://ww1.microchip.com/downloads/en/DeviceDoc/39662c.pdf>.
- [13] ĎAĎO Stanislav. *Senzory a měřicí obvody. 1. vyd.* Praha: České vysoké učení technické, 1996. 315 s. ISBN: 80-01-01500-9.
- [14] *GM electronic.* [Online] 4 revize. [Citace: 13. 02 2012.] [http://www.gme.cz/\\_dokumentace/dokumenty/530/530-067/dsh.530-067.1.pdf](http://www.gme.cz/_dokumentace/dokumenty/530/530-067/dsh.530-067.1.pdf).
- [15] D-Robotics. *D-Robotics UK.* [Online] 30. 07 2010. [Citace: 13. 02 2012.] <http://www.robotshop.com/PDF/dht11.pdf>.
- [16] KUPČEKOVÁ, Alice. Jak se rychle zbavit vlhkosti a plísně v bytě či domě. *Bydlení iDNES.* [Online] MF DNES, 17. 02 2008. [Citace: 08. 02 2012.] [http://bydleni.idnes.cz/jak-se-rychle-zbavit-vlhkosti-a-plisne-v-byte-ci-dome-fyt-stavba.aspx?c=A080215\\_153043\\_rodinne\\_domy\\_web](http://bydleni.idnes.cz/jak-se-rychle-zbavit-vlhkosti-a-plisne-v-byte-ci-dome-fyt-stavba.aspx?c=A080215_153043_rodinne_domy_web).
- [17] DOLEČEK, Jaroslav. *Moderní učebnice elektroniky.* Praha: BEN - technická literatura, 2005. ISBN 80-730-0184-5.
- [18] NAGEL, Christian at al. *Professional C# 4 and .Net 4.* Indianapolis: Wiley Publishing, 2010. ISBN 978-0-470-50225-9.
- [19] Microsoft Corporation. Moving Java Applications to .NET. *Microsoft Developer Network.* [Online] 05 2005. [Citace: 01. 04 2012.] <http://msdn.microsoft.com/en-us/library/ms973842.aspx>.
- [20] HENDL, Jan. *Přehled statistických metod zpracování dat: analýza a metaanalýza dat.* Praha: Portál, 2004. ISBN 80-717-8820-1.
- [21] ZENDULKA, Jaroslav et al. *Analýza a návrh informačních systémů.* Brno : FIT VUT v Brně, 2006.
- [22] VAŘACHA, Pavel a Jiří HORÁK. Technické prostředky ovládání domácnosti přes internet. In: *Internet, competitiveness and organizational security: Process Management and the Use of Modern Technologies.* Zlín: Tomas Bata University, 2012, 182 – 189 s. ISBN 978-80-7454-142-1.

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

AC	Alternating Current - střídavý elektrický proud
DC	Direct Current - stejnosměrný elektrický proud
DHCP	Dynamic Host Configuration Protocol - používá se pro automatickou konfiguraci počítačů připojených do počítačové sítě
EEPROM	Electrically Erasable Programmable Read-Only Memory - elektricky mazatelná paměť
GUI	Graphical User Interface – grafické uživatelské rozhraní
HTML	HyperText Markup Language - značkovací jazyk pro hypertext
HTTP	Hypertext Transfer Protocol - internetový protokol
ID	Identifikace
IDE	Integrated Development Environment - vývojové prostředí
IP adresa	číslo, které jednoznačně identifikuje síťové rozhraní v počítačové síti
LCD	Liquid Crystal Display - displej z tekutých krystalů
LED	Light-Emitting Diode - dioda emitující světlo
MAC adresa	Media Access Control - jedinečný identifikátor síťového zařízení
MD5	Message-Digest algorithm – algoritmus hašovací funkce
NAT	Network Address Translation - překlad síťových adres
PoE	Power over Ethernet - napájení po datovém síťovém kabelu
PWM	Pulse Width Modulation - modulace šířkou pulsu, pin lze použít i pro analogový výstup
RAM	Random-access Memory - paměť s přímým přístupem
SD	Security Digital – typ paměťové karty
SQL	Structured Query Language - standardizovaný dotazovací jazyk používaný pro práci s daty v relačních databázích
SRAM	Static Random Access Memory - je označení pro polovodičovou paměť, která k uchování svých dat nepotřebuje jejich periodickou obnovu

SSL	Secure Sockets Layer – protokol pro bezpečnou komunikaci s internetovými servery
USB	Universal Serial Bus - univerzální sériová sběrnice
VOIP	Voice over Internet Protocol - technologie pro telefonování přes Internet
VPN	Virtual private network - Virtuální privátní síť
WWW	World Wide Web
XML	eXtensible Markup Language - značkovací jazyk

**SEZNAM OBRÁZKŮ**

Obr. 1. Informatické zasíťování sběrnici KNX/EIB [5] .....	15
Obr. 2. Návrh architektury .....	18
Obr. 3. Arduino Uno a Arduino Nano .....	20
Obr. 4. Ethernetová karta pro Arduino - Ethernet Shield V1.1 .....	23
Obr. 5. Číslicový teploměr DS18B20 .....	25
Obr. 6. Schéma možného zapojení jednoho nebo více číslicových teploměrů DS18B20 [14].....	26
Obr. 7. Senzor vlhkoměr DHT11.....	26
Obr. 8. Zapojení senzoru DHT11 [15].....	27
Obr. 9. Fotorezistor pro zjištění intenzity osvětlení.....	28
Obr. 10. Zapojení fotorezistoru k Arduino .....	28
Obr. 11. Magnetický dveřní spínač.....	29
Obr. 12. Elektromagnetické relé .....	30
Obr. 13. Servomotor SG90 .....	31
Obr. 14. Znakový LCD displej MC1602E-SYL/H obsahující 2 řádky 16 znaků.....	33
Obr. 15. LED diody .....	34
Obr. 16. Use Case diagram webového serveru .....	36
Obr. 17. ER diagram .....	37
Obr. 18. Diagram tříd databázové vrstvy.....	39
Obr. 19. Obecné třídy .....	41
Obr. 20. Use case jednotlivých tříd webových stránek.....	42
Obr. 21. Net Framework [19] .....	47
Obr. 22. Registrace uživatele .....	51
Obr. 23. Editace uživatelského profilu .....	51
Obr. 24. Webová stránka Domov přehled s grafem .....	53
Obr. 25. Stav domácnosti vyjádřený grafem s hodnotami na obou osách.....	53
Obr. 26. Tabulka hodnot senzorů.....	54
Obr. 27. Přehled varování .....	54
Obr. 28. Nastavení alertů a posílání příkazů Arduino .....	55
Obr. 29. Nastavení pro administrátora.....	56
Obr. 30. Testovací HTML stránka Arduina.....	66
Obr. 31. Testovací HTML stránka Arduina na mobilním telefonu .....	66

**SEZNAM TABULEK**

Tab. 1. Srovnání parametrů vybraných aktuálních typů desek platformy Arduino [10] .....	21
Tab. 2 Vlastnosti ENC28J60 [12] .....	23
Tab. 3. Teploměr Dallas DS18B20 [14] .....	25
Tab. 4. DHT 11 vlhkoměr a teploměr [15] .....	27
Tab. 5. Fotorezistor VT83N4 .....	28
Tab. 6. Magnetický dveřní spínač .....	29
Tab. 7. Elektromagnetické relé .....	31
Tab. 8. Servomotor SG90 .....	32
Tab. 9. Znakový LCD displej MC1602E-SYL/H .....	33
Tab. 10. Nastavení webového serveru .....	58
Tab. 11. Heslo do MD5 a zpětná rekonstrukce .....	63

## SEZNAM PŘÍLOH

- P I            CD s textem diplomové práce, zdrojovými kódy a katalogovým listem senzorů.