

**Univerzita Hradec Králové**  
**Fakulta informatiky a managementu**  
**Katedra informatiky a kvantitativních metod**

**Chytrý dům na platformě Raspberry Pi**

Diplomová práce

Autor: Bc. Jakub Kiele

Studijní obor: Aplikovaná informatika

Vedoucí práce: doc. Mgr. Tomáš Kozel, Ph.D.



Prohlášení:

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 28.8.2017

Jakub Kiele

Poděkování:

Rád bych touto cestou poděkoval vedoucímu práce doc. Mgr. Tomáši Kozlovi, Ph.D. za cenné rady, ochotu a věnovaný čas.

Taktéž bych rád poděkoval své rodině za veškerou podporu.

## **Anotace**

První část diplomové práce se obecně zabývá problematikou chytrých domácností. Popisuje jednotlivé podsystémy, které se podílejí na chodu chytrého domu a také jednotlivé hardwarové prvky těmito subsystemy využívané. Teoretická část také zmiňuje komunikační protokoly a technologie používané v systémech chytrých domů a představuje několik existujících řešení systémů pro chytré domy. Hlavním cílem práce je pak návrh architektury systému pro chytré domy a implementace prototypu na základě tohoto návrhu. Systém je navržen tak, aby bylo možné integrovat komponenty založené na rozdílných komunikačních technologiích a protokolech. Prototyp je vytvořen s pomocí open-source technologií a jako centrální řídicí prvek je využit mikropočítač Raspberry Pi 2. V práci jsou popsány některé části prototypu a problémy, které bylo třeba řešit v průběhu návrhu a implementace.

## **Annotation**

### ***Title: Smart Home using RaspberryPI Platform***

The first part of this diploma thesis is focused on general problematic of smart houses. It describes particular subsystems of smart house as well as particular hardware devices used in these subsystems. Theoretical part also describes communication protocols and technologies used in systems for smart houses and introduce some existing smart home systems. The main goal of this diploma thesis is design of architecture of smart home system and implementation of prototype based on this design. System is designed to integrate components based on different communication technologies and protocols. Prototype is created with open-source technologies with Raspberry Pi 2 used as central element. This diploma thesis also describes some parts of prototype and issue that were needed to solve during designing of architecture and implementation of prototype.



# Obsah

1	Úvod.....	1
2	Inteligentní dům a jeho subsystémy .....	2
2.1	Subsystémy inteligentního domu .....	3
2.1.1	Osvětlení .....	4
2.1.2	Systemy vytápění.....	9
2.1.3	Bezpečnostní systémy .....	11
2.1.4	Multimédia .....	15
3	Hardwarové prvky v inteligentním domě .....	18
3.1	Řídící jednotky.....	18
3.2	Senzory.....	20
3.3	Aktuátory.....	21
3.4	Ovládací prvky.....	22
4	Technologie pro komunikaci v inteligentním domě.....	24
4.1	Drátové technologie .....	24
4.1.1	Power line Communication .....	24
4.1.2	X10.....	25
4.1.3	Universal powerline bus.....	26
4.2	Bezdrátové technologie .....	27
4.2.1	Wi-Fi .....	27
4.2.2	Bluetooth.....	28
4.2.3	ZigBee.....	29
4.2.4	Z-Wave .....	29
4.2.5	EnOcean.....	31
4.3	Komunikační protokoly a standardy .....	31
4.3.1	KNX.....	31

4.3.2	LonWorks .....	32
4.3.3	Modbus .....	33
5	Představení vybraných existujících řešení.....	34
5.1	Loxone .....	34
5.2	Insteon.....	34
5.3	OpenHAB .....	35
5.4	Home Assistant.....	36
5.5	Rex.....	37
6	Využití Raspberry Pi v inteligentním domě .....	40
6.1	Raspberry Pi 2 B a rozšiřující moduly.....	40
6.1.1	Rozšiřující modul UniPi board.....	42
6.1.2	Rozšiřující modul RaZberry (Z-Wave) .....	44
6.1.3	Rozšiřující modul XBee (ZigBee) .....	44
6.1.4	Rozšiřující modul EnOcean Pi 868 (EnOcean) .....	45
6.2	Hardware použitý k vytvoření prototypu.....	46
6.2.1	Senzor teploty a vlhkosti .....	46
6.2.2	Senzor intenzity světla .....	47
6.2.3	Senzor pohybu .....	48
6.2.4	Chytrá žárovka LIFX .....	49
7	Návrh systému řízení inteligentního domu .....	50
7.1	Návrh a architektura systému .....	50
7.1.1	Big Picture návrhu systému.....	50
7.1.2	Případy užití systému .....	52
7.1.3	Model jádra systému .....	54
7.1.4	Architektura systému.....	56
7.2	Použité technologie .....	57



7.2.1	PostgreSQL.....	57
7.2.2	Java.....	58
7.2.3	Spring.....	58
7.2.4	Maven.....	59
7.2.5	Hibernate.....	59
7.2.6	Webové služby.....	59
7.2.7	Aplikační servery.....	61
7.2.8	Frontendové technologie.....	62
8	Implementace.....	63
8.1	Základní konfigurace projektu.....	63
8.2	Datová část.....	63
8.3	Core modul – jádro systému.....	64
8.4	Komunikační modul a komunikace se sensory a aktory.....	64
8.4.1	Python skripty pro ovládání sensorů.....	65
8.4.2	Rest API.....	66
8.4.3	LIFX http API.....	67
8.5	Event Manager modul.....	68
8.5.1	Quartz Jobs.....	68
8.5.2	JEXL.....	70
8.5.3	Java reflexe.....	71
8.5.4	Emailové notifikace.....	72
8.6	Zabezpečení.....	73
9	Výsledky.....	74
10	Závěr.....	77
11	Seznam použité literatury.....	79
12	Přílohy.....	87

## Seznam obrázků

Obrázek 1: Prvky chytrého domu – vyobrazení od firmy Loxone. Zdroj [3] .....	3
Obrázek 2: Philips Hue Color startovací sada (Bridge + 3 žárovky). Zdroj [4] .....	5
Obrázek 3: LIFX Color 1000. Zdroj [5].....	5
Obrázek 4: MiPow Playbulb Color. Zdroj [7].....	5
Obrázek 5: chytrý nástěnný termostat Nest. Zdroj [12] .....	10
Obrázek 6: Honeywell EvoHome – centrální ovládací jednotka a ventil topného tělesa. Zdroj [15] .....	10
Obrázek 7: český bezpečnostní systém Angee. Zdroj [20].....	14
Obrázek 8: bezpečnostní systém ISmartAlarm. Zdroj [21] .....	14
Obrázek 9: bezpečnostní sada Iris a centrální prvek Iris. Zdroj [23] .....	14
Obrázek 10: zapojení multimediálního systému v chytrém domě na platformě X10. Zdroj [25] .....	16
Obrázek 11: Loxone music server. Zdroj [27] .....	17
Obrázek 12: řídicí jednotka Loxone Miniserver. Zdroj [29].....	18
Obrázek 13: bezdrátová řídicí jednotka Loxone Miniserver Go. Zdroj [29] .....	18
Obrázek 14: PLC – programovatelný automat FATEK. Zdroj [30].....	19
Obrázek 15: schéma inteligentního senzoru. Zdroj [34] .....	20
Obrázek 16: Princip technologie Power line Communication. Přenos informací pomocí elektrických rozvodů. Zdroj [39] .....	25
Obrázek 17: Architektura jádra systému Home Assistant. Zdroj [75] .....	37
Obrázek 18: Vizualizace řídicího systému Rex a jeho komponent. Zdroj [77] .....	38
Obrázek 19: Raspberry Pi 2 model B. Zdroj [79].....	41
Obrázek 20: Rozšiřující deska UniPi board, která společně s Raspberry Pi vytváří programovatelnou řídicí jednotku. Zdroj [81] .....	43
Obrázek 21: Modul RaZberry (Z-Wave). Zdroj [83] .....	44

Obrázek 22: Moduly XBee a XBee-Pro (s anténou). Zdroj [84] .....	45
Obrázek 23: USB adaptér pro XBee moduly. Zdroj [85] .....	45
Obrázek 24: XBee modul umístěný na USB adaptéru. Zdroj [85] .....	45
Obrázek 25: Rozšiřující modul EnOcean Pi 868. Zdroj [87] .....	46
Obrázek 26: Senzor teploty a vlhkosti vzduchu, přední a zadní strana.....	47
Obrázek 27: senzor intenzity osvětlení.....	48
Obrázek 28: PIR senzor – pasivní infračervený senzor pohybu. Zdroj [90] .....	48
Obrázek 29: Big Picture navrženého systému pro domácí automatizaci .....	51
Obrázek 30: Zjednodušený Use Case diagram navrženého systému domácí automatizace.....	53
Obrázek 31: Class diagram jádra systému .....	55
Obrázek 32: Vícevrstvá architektura systému .....	56
Obrázek 33: Webové rozhraní v prohlížeči FireFox.....	75

## **Seznam tabulek**

Tabulka 1: Porovnání vybraných systémů chytrého osvětlení.....	8
Tabulka 2: Porovnání technických parametrů technologií UPB a X10. Zdroj [44] ..	27
Tabulka 3: Porovnání vybraných bezdrátových technologií. Vychází z [57][58] .....	30

# 1 Úvod

Žijeme v informační době. S rozvojem internetu se stalo získávání velkého množství informací a jejich přenos velmi snadný. V součinnosti s miniaturizací a zvyšováním výkonu výpočetních zařízení se nám naskytují nové možnosti získávání informací, ale i jejich využívání.

Z těchto možností těží celý obor nazvaný internet věcí, neboli Internet of Things (IoT). Pomocí vzájemně propojených chytrých zařízení můžeme vytvářet celá inteligentní prostředí. Jedním z příkladů je chytrý dům. Cílem této práce je seznámit čtenáře právě s problematikou chytrých domů a návrhu řídicího systému pro chytrý dům.

První část práce se zabývá vysvětlením pojmu chytrý nebo inteligentní dům. Také je zde nastíněna problematika jednotlivých subsystémů, které mohou být součástí inteligentního domu. V dalších částech jsou shrnuty a popsány komunikační sběrnice a hardwarové prvky, které pomocí sběrnic propojujeme do větších systémů.

Nakonec jsou představena některá existující řešení. Ty jsou výběrem z různých možností hardwarové a softwarové implementace chytrých domů.

Na základě předchozích poznatků je zpracována praktická část diplomové práce, která si bere za cíl, vytvořit návrh řídicího systému pro chytrý dům. Společně s tímto návrhem jsou uvedeny i ukázky z implementovaného prototypu.

## 2 Inteligentní dům a jeho subsystemy

Za inteligentní dům neboli smart home považujeme takový dům, který pomocí moderních technologií a domácí automatizace zvyšuje komfort bydlení svým obyvatelům. Domácí automatizace se v různých měřítkách využívá již několik desítek let (komunikační protokol X10 vznikl v roce 1975 [1]) a položila základ pro celé odvětví nazývané Intelligent environments – *inteligentní prostředí*.

Inteligentní prostředí je dle [2] definované jako takové prostředí, které interaguje se svými obyvateli přirozeným způsobem a zároveň se umí přizpůsobit aktuálním aktivitám obyvatel s jejich minimálním vyrušením od prováděné činnosti. Inteligentní prostředí by mělo jeho obyvatelům pomoci automatizovat a usnadňovat každodenní činnosti a mělo by být schopné se na své obyvatele adaptovat za pomoci vlastního učení.

Další příklady inteligentního prostředí mimo chytrého domu mohou být prostředí více specializovaná – zdravotnické zařízení, učebna, interiér vozidla, posilovna, kancelář, nádražní hala atd. Každé z nich má jiný účel, ale prostředky, kterými bude prostředí svých cílů dosahovat, jsou podobné.

Základními stavebními prvky inteligentního prostředí jsou senzory a aktuátory. Senzory slouží k zjišťování informací z prostředí, zatímco aktuátory umí vykonat činnost potřebnou ke změně v prostředí. Například světelný senzor zaznamená dopadající světlo a aktuátory v podobě motorků zatáhnou žaluzie.

Vývoj technologie v oblasti aktuátorů a senzorů je pro stavbu inteligentních prostředí velmi důležitý. Cílem je především zmenšování, energetická soběstačnost a bezdrátová komunikace. Tyto tři aspekty totiž zajistí, že prvky mohou být v prostředí instalovány, aniž by o nich uživatel věděl a musel se o ně starat. Prostředí tak není narušeno cizími prvky a zůstává pro své obyvatele stále přirozené. Miniaturizace nám také umožňuje umisťovat senzory na místa, kde to nebylo dříve možné a tím získáváme nové možnosti. V současné době jsou již běžně využívány chytré telefony a rozrůstá se i množství nositelné elektroniky. Díky těmto vymoženostem může prostředí získávat další užitečné informace o uživatelích a lépe se tak přizpůsobovat jejich potřebám.



sobě dobrým základem pro inteligentní dům. Případně můžeme zkombinovat oba způsoby a funkce inteligentní elektroinstalace rozšířit o další specializované systémy, nebo systémy využívající bezdrátové technologie.

Následující podkapitoly jsou shrnutím nejdůležitějších subsystémů domu. Jejich kombinací je možné pokrýt velkou část života uživatelů a zpřístupnit jim nové možnosti užívání některých prvků. Některé systémy a jejich funkce nejsou žádnou novinkou a jsou již běžně využívány. Nabytí dalších funkcí a inteligentního chování lze dosáhnout jejich vzájemnou integrací, a především díky centrálnímu řídicímu prvku. Ten je mozkiem každého chytrého domu a využití ostatních systémů záleží především na jeho softwaru.

### **2.1.1 Osvětlení**

Základní automatizace osvětlení je jednoduchý způsob, jak zvýšit komfort bydlení v domě. V kombinaci s pohybovými senzory dosáhneme automatického rozsvěcení při příchodu do místnosti, nebo automatické osvětlení schodiště. Mimo automatického řízení mohou uživatelé osvětlení stále ovládat. Buďto klasickými vypínači, nebo vzdáleně, pomocí ovladače (např. chytrého telefonu). Běžnou funkcí pak bývá nastavování intenzity světla. Pokud jsou instalovány vícebarevné LED žárovky, mohou si uživatelé nastavovat různé světelné scény pro jednotlivé místnosti, denní dobu nebo situaci.

Chytrého osvětlení v domě či bytě lze dosáhnout několika způsoby. Zvažovat instalaci chytrého osvětlení je vhodné už při stavbě či rekonstrukci objektu, pak je možné rovnou přizpůsobit elektroinstalaci potřebám osvětlení. Existují ale i systémy, které uživateli umožní dosáhnout stejného výsledku bez jakýchkoliv stavebních zásahů. Díky tomu je možné instalovat je i do hotových bytů nebo domů bez nutnosti rekonstrukce. Na Obrázek 2, Obrázek 3 a Obrázek 4 jsou vidět vybraní zástupci z řad chytrých žárovek, jejichž popis následuje dále v této kapitole.





**Obrázek 2: Philips Hue Color startovací sada (Bridge + 3 žárovky). Zdroj [4]**

**Obrázek 3: LIFX Color 1000. Zdroj [5]**

**Obrázek 4: MiPow Playbulb Color. Zdroj [7]**

## Philips Hue

Bezdrátové řešení chytrého osvětlení od společnosti Philips dostalo název Hue. Nabízí uživateli jednoduchou instalaci chytrého osvětlení v různém rozsahu bez nutnosti jakéhokoliv zásahu do elektroinstalace. V nabídce produktů Philips Hue najdeme několik typů žárovek, LED pásků, stolních svítidel a lamp, přepínačů, pohybový senzor a stropní svítidla. To je, pro většinu případů, dostatečný výběr zařízení na osvětlení celého domova.

Philips nabízí několik základních sad obsahujících dvě nebo tři žárovky a centrální prvek. Uživatel si tak může zvolit mezi čistě bílými žárovkami, nebo barevnými žárovkami (případně kombinaci obojího). Nejdůležitější součástí těchto základních sad je ovšem centrální prvek zvaný Bridge. Jde o centrální řídicí prvek připojený k domácímu routeru, který řídí všechny žárovky a další příslušenství k němu připojené. Připojení k Wi-Fi routeru slouží k ovládání osvětlení přes internet. Samotné žárovky a příslušenství s centrálním prvkem komunikují pomocí technologie Zig-Bee. [4]

Díky systému Philips Hue získá uživatel širokou škálu možností ovládání osvětlení svého domova pomocí řady nabízených funkcí:

- Výběr ze 16 miliónů barev (u barevných žárovek).
- Synchronizace světla s hudbou a filmy.
- Nastavení odstínů bílého světla.
- Konfigurace barevných scén pro různé příležitosti a činnosti.

- Probouzení pomocí zvyšování intenzity světla.
- Ovládání světel odkudkoliv a ovládání pomocí chytrého telefonu/tabletu.
- Automatizace a nastavení světelných rozvrhů.
- Philips Hue Bridge zvládne obsloužit až 50 světel a 10 kusů příslušenství.
- Kompatibilita s technologií Apple HomeKit.

## LIFX

Alternativou k Philips Hue jsou žárovky od společnosti LIFX (čteno Life-X), které mohly vzniknout i díky úspěšné crowdfundingové kampani na Kickerstarteru. V době vzniku této práce existuje již třetí generace žárovek LIFX. Stejně jako u Philips Hue se jedná o bezdrátové řešení osvětlení, a to pomocí úsporných LED žárovek. Není tedy nutné žádných větších zásahů, stačí nainstalovat žárovky a aplikaci do chytrého telefonu. Produktová řada LIFX není tak rozsáhlá jako u systému Hue. Obsahuje pouze několik typů žárovek rozdělených podle barvy světla a barevné LED pásy.

Největším rozdílem oproti řešení Philips Hue je komunikace jednotlivých prvků osvětlení. Žárovky LIFX komunikují prostřednictvím Wi-Fi a obsahují vše potřebné přímo v sobě. Tím je eliminována potřeba dalšího centrálního prvku pro jejich řízení a komunikace se žárovkami probíhá skrze router. K routeru se ovšem nepřipojují všechny žárovky. Komunikace probíhá v režimu master/slave, kdy vždy jedna žárovka je v režimu master připojena k routeru skrze standard 802.11n a ostatní žárovky v režimu slave se připojí k master žárovce pomocí standardu 802.15.4 a vytvoří mesh síť. [5][6]

Škála funkcí, které žárovky LIFX nabízí, se velmi podobá těm u konkurenčního Philips Hue. Uživatel má možnost měnit barvu, odstín a intenzitu osvětlení. U RGB modelů v rozsahu 16 milionů barev. U bílých modelů pak lze vybírat mezi 1000 odstíny bílé, od studených po teplé odstíny. Samozřejmostí je slučování žárovek do logických uskupení a vytváření světelných scén. Dále pak synchronizace s hudbou, vytváření denních rozvrhů a schémat pro svícení. Žárovky LIFX se umí integrovat s chytrými systémy třetích stran, kterými jsou například Nest, Amazon Echo, Samsung

SmartThing, IFTTT (If This Than That – aplikace pro tvorbu různých pravidel automatizace) a další. To umožňuje vytvářet i složitější logiku a nechat tak světlo reagovat na různé události typu příchozích emailů či sms zpráv nebo změny počasí. Ve třetí generaci jsou žárovky LIFX vhodné i pro venkovní použití a model LIFX+ disponuje i infračerveným světlem pro zvýšení účinnosti bezpečnostních kamer. [5]

### **MiPow**

Posledním vybraným zástupcem řešení chytrého osvětlení jsou žárovky a svítidla firmy MiPow. Chytré osvětlení je pro firmu MiPow jedním z největších bodů zájmu, vedle výroby powerbank a audio zařízení. Možná právě proto, že se firma zabývá i výrobou audio zařízení jako jsou sluchátka či reproduktory, jsou některé chytré žárovky od tohoto výrobce vybaveny právě reproduktory, což přidává systému osvětlení další zajímavé funkce.

Vestavěné reproduktory umožňují přidat další rozměr světelným scénám, které si uživatel nadefinuje a ozvučit tak celý obytný prostor. Mimo ozvučení jsou ale funkce žárovek MiPow velmi podobné již dříve představené konkurenci. Také nabízejí možnost nastavení osvětlení v široké barevné škále, slučování světel do skupin, definování světelných scén a používání různých světelných efektů. Společně s dalším softwarem typu IFTTT se možnosti využití dále rozrostou. Nabídka osvětlení firmy MiPow je ale širší než u firmy LIFX, a dokonce i než u systému Philips Hue. Obsahuje žárovky různých tvarů a využití (např. klasické nebo bodové), ale i stolní nebo zahradní svítidla (odolné proti dešti, nabíjené slunečnou energií) a LED pásy.

Instalace osvětlení je opět velmi jednoduchá. Stačí zašroubovat žárovky nebo aktivovat dané svítidlo. Není potřeba žádný speciální centrální prvek, ke kterému by se svítidla připojovala. Svítidla komunikují pomocí technologie bluetooth, takže centrálním prvkem pro jejich řízení, se stane chytrý telefon nebo tablet uživatele. [7][8][9]

Následuje Tabulka 1 porovnávající jednotlivá vybraná řešení. Jde především o srovnání parametrů jednotlivých chytrých žárovek a shrnutí základních vlastností.

	<b>Philips Hue</b>	<b>LIFX</b>	<b>MiPow</b>
<b>Centrální prvek</b>	Ano	Ne	Ne
<b>Komunikační technologie</b>	ZigBee	Wi-Fi (IEEE 802.15.4)	Bluetooth
<b>Příkon</b>	10 W	11 W	5 W
<b>Životnost</b>	25000 h	25000 h	30000 h
<b>Barevná škála</b>	16 milionů barev	16 milionů barev	16 milionů barev
<b>Světelnost</b>	Až 800 lm	Až 1055 lm	280 lm
<b>Chromaticnost</b>	2000–6500 K	2500–9000 K	3000 K
<b>Stmívatelné</b>	Softwarově	Softwarově	Softwarově
<b>Ovládací aplikace</b>	iOS / Android	iOS / Android / Windows	iOS / Android
<b>Cena</b>	65 + 80 \$ / 185 \$ 1700 + 2000 Kč / 4600 Kč	45–60 \$ 1100-1500 Kč	40–80 \$ 1000-2000 Kč

**Tabulka 1: Porovnání vybraných systémů chytrého osvětlení**

V tabulce můžeme vyčíst, že většina parametrů chytrých žárovek jsou velmi podobné. Z hlediska osvětlení jsou žárovky Philips Hue (model White and color ambiance) srovnatelný s produktem Color 1000 firmy LIFX. Oba nabízejí vysokou svítivost a nastavení teploty bílého světla. MiPow Rainbow sice nedosahuje stejných parametrů ale s příkonem 5 W je o něco úspornější. Velký rozdíl pak vidíme v ceně. Ta se u produktů firmy LIFX pohybuje mezi 1100 až 1500 Kč a uživatel má na výběr v podstatě jen z jednotlivých generací žárovek. U MiPow je cenový rozsah 1000-2000 Kč, ale dražší model žárovky s názvem MiPow Color nabízí kromě osvětlení i zabudovaný reproduktor. Nejvýrazněji vyčnívá cena za produkty Philips Hue. To je zapříčiněno především řešením celého systému pomocí centrálního prvku, který ovšem poskytuje možnost připojit i jiná chytrá zařízení firmy Philips. Uživatel si tak

může zakoupit jednu žárovku a centrální prvek Bridge za cca 3700 Kč anebo startovní balíček, který Philips nabízí. Ten obsahuje centrální prvek a tři chytré žárovky za cca 4600 Kč.

Další možností, jak vybavit dům chytrým osvětlením je pořízení inteligentní elektroinstalace. Toto řešení je vhodné zejména pro novostavby, nebo při rekonstrukci. Inteligentní elektroinstalace bude oproti předchozím řešením finančně značně náročnější a pro její zapojení je potřeba odborný pracovník nebo firma, což cenu také zvedá. Výsledek je ale o mnoho univerzálnější a inteligentní elektroinstalace je základ pro celý inteligentní dům. Můžeme tak ovládat nejen osvětlení, ale například i žaluzie, dveře, elektrospotřebiče atd. Takového chování lze se specializovanými systémy dosáhnout s velkou námahou, pokud zvolíme takové systémy, které půjde integrovat dohromady. Srdcem inteligentní elektroinstalace je řídicí jednotka v podobě PLC neboli programovatelného logického automatu. Řídicí jednotky a PLC jsou blíže popsány v kapitole 3.1 3.1 Řídicí jednotky.

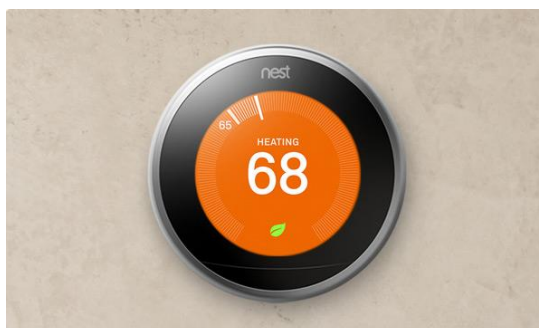
### **2.1.2 Systémy vytápění**

Automatizace či určité úsporné způsoby řízení vytápění byly používány již dávno před vznikem konceptu chytrých budov. Pokud se bavíme o inteligentním vytápění, očekáváme oproti klasickým systémům vytápění nějakou přidanou hodnotu. Chytré vytápění by tak mělo přinést úsporu financí i zdrojů a vyšší komfort pro obyvatele domácnosti. Další funkce a možnosti optimalizace vytápění lze získat i díky integraci s ostatními subsystemy. Chytré vytápění se tak může naučit časy kdy uživatelé nejsou doma a v této době vytápění ztlumit nebo úplně vypnout. Začít opět vytápět by mělo s dostatečným předstihem, aby s příchodem uživatelů měly místnosti požadovanou teplotu. Samozřejmostí je možnost regulace vytápění v každé místnosti zvlášť, jeho kontrola a vzdálené ovládání nebo nastavení časových plánů pro vytápění v průběhu dne. Vytápěcí systém se také může přizpůsobovat aktuálnímu počasí venku, pokud mu tuto informaci dodáme.

V rámci systému vytápění je vhodná také integrace s ventilačními nebo klimatizačními systémy. To nám zaručí, že vytápění nebude zapnuté společně s klimatizací a nebudou tak pracovat proti sobě. Dále je možné naučit vytápění vypínat se v době

kdy jsou otevřená okna kvůli větrání. K tomu jsou samozřejmě zapotřebí senzory kontrolující okna. Těmi disponují například některé základní balíčky bezpečnostních systému představených dále. [10][11]

Základními prvky inteligentního vytápění jsou termostat či teplotní čidla, ventily topení a řídicí jednotka tepelného zdroje. V bytech napojených na centrální systém vytápění, kde nemáme přístup ke zdroji tepla (např. kotli) se musíme spokojit s monitorováním teploty a řízením vytápění pomocí ventilů na topení. Dále následuje popis vybraných řešení chytrého vytápění, kterými jsou termostat Nest zobrazený na Obrázek 5 a Honeywell EvoHome na Obrázek 6.



**Obrázek 5: chytrý nástěnný termostat Nest. Zdroj [12]**



**Obrázek 6: Honeywell EvoHome – centrální ovládací jednotka a ventil topného tělesa. Zdroj [15]**

## **Nest Thermostat**

Termostat Nest je chytrou náhradou klasického termostatu. Dokáže se učit podle rozvrhu jeho uživatelů, požadovaných teplot a počasí, a snížit tak náklady za vytápění. Dle [13] činí úspora průměrné domácnosti až 12 % nákladů při přechodu na termostat Nest. Investice do termostatu se tak vrací zhruba každé dva roky.

Teplotu je samozřejmě i nadále možné upravovat ručně a termostat při každém nastavení radí nejúspornější teplotu. Kromě učení a úspory nabízí i další funkce. Při příchodu do místnosti se rozsvítí a zobrazí aktuální informace. Nabízí přehledné měsíční reporty o vytápění, které je možné si prohlédnout na chytrém telefonu nebo tabletu a nabízí tak další tipy, jak ušetřit. Termostat umí ovládat i další systémy, jako jsou klimatizace, ventilace atd.

Termostat je díky jednotce Heat Link, která je součástí balení a používá technologii OpenTherm, kompatibilní s většinou evropských kotlů. Heat Link se ke kotli připojí

kabelem nebo pomocí rozvodů původního termostatu. S termostatem Nest pak komunikuje bezdrátově, nebo nízko napěťovým 12 V kabelem ve zdi. Tím je pokrytá potřeba ovládání tepelného zdroje. Je pak lhostejné, jaké ventily jsou na radiátorech. Jestli manuální, termostatické, nebo programovatelné. Během instalace systému najde uživatel doporučení, jak ventily nastavit pro optimální výsledek, dle jejich typu. Viz zdroj [14]. Samozřejmostí pak je komunikace s ostatními produkty značky Nest, ale i s produkty jiných výrobců jako jsou například chytré žárovky Philips Hue (viz kapitola Philips Hue). [12][13]

### **Honeywell EvoHome**

Honeywell EvoHome je komplexním řešením vytápění pro chytrý dům. Na rozdíl od termostatu Nest se nespolehá pouze na jednu referenční místnost, ale hned v základu je navržen pro monitoring a řízení teploty v několika zónách objektu. V základním balíčku tak nalezneme řídicí jednotku, ovládací panel a několik digitálních ventilů na topení, které slouží k monitoringu i vzdálenému nastavení teploty. Ventily jsou napájeny dvěma tužkovými bateriemi. Mezi příslušenství systému EvoHome patří kromě různých typů ventilů pro různé topné jednotky také termostaty, kterými lze rozšířit nebo zpřesnit monitoring teploty. Kromě ovládacího panelu je možné celý systém řídit i přes chytrý telefon nebo tablet.

Stejně jako Nest nabízí EvoHome všechny funkce, které můžeme od chytrého systému vytápění očekávat. Sám se naučí denní rozvrh uživatelů a preferovanou teplotu v jednotlivých zónách. Také se naučí, jak dlouho trvá každou zónu vytopit na požadovanou teplotu. Díky těmto funkcím slibuje úspory energie za vytápění až 40 %. Veškerá komunikace mezi jednotlivými zařízeními probíhá skrze Wi-Fi a samozřejmostí je spolupráce systému s aplikací IFTTT. [16]

### **2.1.3 Bezpečnostní systémy**

Bezpečnostní systémy byly do domů a budov instalovány dlouho před vznikem konceptu inteligentních budov. Integrace s ostatními systémy v rámci inteligentního domu pak přináší nové možnosti. Například ve spojení s řízením osvětlení je možné simulovat přítomnost osob v domě, pokud je dům delší dobu opuštěný a odradit tak

případné zloděje. Dle [17] bylo v roce 2013 v České republice hlášeno 11 117 vloupání do bytů a rodinných domů. Celkem pak 62 384 vloupání do všech typů objektů. Je proto zřejmé, že bezpečnostní systém je užitečnou a vítanou součástí domu.

Mezi bezpečnostní systémy řadíme takové systémy, které mají objekt chránit před nechtěným vniknutím nebo vloupáním. Jde o soubor technických a organizačních opatření, které mají za úkol majetkové nebo jiné hodnoty před odcizením, poškozením nebo zničením. Běžně se jedná o systémy využívající pohybové senzory, senzory rozbitých oken, monitorovací a kamerové jednotky, případně alarmy. Mezi bezpečnostní systémy můžeme zařadit i systémy, které mají za úkol chránit obyvatele objektu před nebezpečími v podobě požáru, kouře, úniku plynu, nebo třeba zatopení. [18]

Bezpečnostní systém může být velmi nákladná a složitá záležitost. Z těchto důvodů může být pro většinu domácností nedostupným luxusem, a tak zůstává mnoho domácností nechráněno nejen před vloupáním. Dále je představeno několik vybraných řešení domácí bezpečnosti, které se snaží bezpečnostní systémy zpřístupnit široké veřejnosti a nabídnout řešení na míru, dle velikosti bytu či domu za rozumnou cenu.

Mezi vybranými systémy jsou český systém Angee, iSmartAlarm a Iris. Všechny tyto systémy jsou založené na podobném principu, kdy využívají jeden centrální prvek a několik periférií rozmístěných po objektu, sloužících ke sběru informací z prostředí. Všechno to jsou také systémy, které si může do bytu či domu nainstalovat uživatel sám a poté je ovládat pomocí mobilního telefonu. Není tedy potřeba instalace od odborníků a tím je také možné dosáhnout nižší ceny.

### **Angee**

Angee je produktem českých vývojářů, který mohl vzniknout i díky úspěšné kic-kerstarterové kampani [19], na které vybral přes půl milionu dolarů. Jedná se o autonomní bezpečnostní systém, který je řízen jedním nebo více hlavními prvky v podobě černého válce s kamerou a zabezpečovacími značkami, rozmístěnými na dveřích či oknech. Angee nabízí následující funkce:

- rotace kamery o 360 stupňů – kamera se automaticky otáčí za pohybem díky detekci zvuku a pohybu



- automatické zapínání – systém se automaticky zapíná a vypíná dle přítomnosti obyvatel domácnosti
- rozpoznávání hlasu – díky rozpoznávání hlasu je autentifikace uživatelů na vysoké úrovni
- zabezpečení celého objektu – díky využití bezpečnostních tagů je možné pokrýt systémem celý objekt
- okamžité notifikace – Angee nabízí mobilní aplikace pro iOS i Android, takže uživatel je vždy notifikován o dění v domácnosti
- vestavěná baterie – baterie v systému vydrží nabitá 10 hodin v případě výpadku proudu

Angee nabízí i další funkce jako diskrétní mód, komunikaci nebo streamování videa. Centrální prvek je navíc plně přenosný. Cena základního balíčku, který obsahuje jeden centrální prvek a jeden senzor začíná na částce 430\$, tedy necelých 11 000Kč. [20]

### **ISmartAlarm**

Je systém oceněný několika odbornými magazíny jako jsou Cnet, PCmag a Engadget. V letech 2014 a 2015 získal i ocenění reddot za design [21]. Stejně jako Angee funguje na základě řídicího prvku, který komunikuje s periferiemi. V tomto případě jsou periferie rozmanitější, ať už se jedná o kamery, kontaktní a pohybové senzory, nebo chytré zásuvky. ISmartAlarm také nabízí mobilní aplikaci pro jeho ovládání.

Systém je schopný si uživatel sestavit sám bez větších potíží, a tak opět ušetří za odbornou pomoc [22]. Mezi jeho nevýhody patří neschopnost kamery natáčet video. Ta pořizuje pouze série snímků. Výhodami pak jsou vedle jednoduché instalace i vysoká možnost rozšiřitelnosti. Ovšem s větším množstvím periférií se musí počítat i s vyšší cenou na pořízení systému. V základním balíčku je možné pořídit iSmartAlarm za 200\$ tedy cca 5000Kč. Ten obsahuje jeden centrální prvek, dva kontaktní senzory, dva bezpečnostní tagy, jeden pohybový senzor a okenní nálepku. [21]

## Iris

Z vybraných řešení je to zřejmě nejkomplexnější, jelikož pomocí tohoto systému je možné vybavit již celý chytrý dům. Ovšem v základním balíčku si zachovává možnost využít pouze bezpečnostní funkce a tím se stává silnou konkurencí. Jelikož cena základního bezpečnostního balíčku začíná na 130\$ (cca 3250Kč), a ten můžete nadále rozšiřovat i jinými, než bezpečnostními periferiemi a získat plnohodnotný chytrý domov.

Stejně jako předchozí řešení se jedná o vytvoření sítě periferií řízených centrálním prvkem. Celý systém je pak řízen pomocí mobilní aplikace. V již zmíněném výchozím balíčku pro bezpečnostní systém se nachází jeden centrální prvek, dva senzory pro kontrolu dveří nebo oken, jeden pohybový senzor a bezdrátová klávesnice. V závislosti na zvoleném rozsahu periferií je možné, že bude při instalaci potřeba pomoc odborníka, což může prvotní nízkou cenu o něco navýšit.[23]



**Obrázek 7: český bezpečnostní systém Angee. Zdroj [20]**



**Obrázek 8: bezpečnostní systém ISmartAlarm. Zdroj [21]**



**Obrázek 9: bezpečnostní sada Iris a centrální prvek Iris. Zdroj [23]**

Předchozí obrázky ukazují všechny popsané bezpečnostní systémy. Na Obrázek 7 je český bezpečnostní systém Angee. Konkrétně centrální prvek v podobě válečkovitého válečku a jeden snímač. Na Obrázek 8 ISmartAlarm s centrálním prvkem, pohybovým senzorem, kamerou a kontaktními senzory. Obrázek 9 pak zobrazuje základní balíček systému Iris, obsahující centrální prvek, pohybový senzor, kontaktní senzory pro kontrolu dveří a oken a ovládací klávesnice. Základní funkce popsaných systémů jsou stejné. Všechny umí monitorovat daný prostor pomocí senzorů a kamery, naučit se kdy jsou

obyvatelé domácnosti pryč a v případě potřeby je notifikovat o neoprávněném přístupu. V případě potřeby je pak možné systémy rozšířit pomocí dalších senzorů, jako jsou detektory kouře nebo zaplavení a dalších.

U bezpečnostních systémů je potřeba ovládat a komunikovat s vícero různorodými prvky (senzory pohybu, kamery, okenní senzory, detektory kouře atd.). Proto je vhodné řešení s centrálním řídicím prvkem, jako u představených řešení. Potom už je jen krůček k tomu, získat plně inteligentní domácnost, pokud nám centrální prvek umožní integrovat a ovládat i jiné systémy a periferie než ty související s bezpečností.

Představená řešení jsou levnou variantou bezpečnostních systémů v porovnání s těmi, které můžeme vidat v některých veřejně přístupných budovách nebo soukromých firmách. Většinou nabízejí jen jednu kameru a pár dalších senzorů. To ale poskytuje základní funkce pro monitorování domácnosti, pohybu v ní a případné vniknutí nechtěných osob. Na tyto zvláštní události je pak obyvatel domácnosti upozorněn prostřednictvím svého telefonu. Základní sady jsou dostatečné pro pokrytí běžného bytu a pro větší byty či domy by bylo nutné dokoupit další periferie. Rozšíření základních sestav je naštěstí jednoduché a uživatel si tak může řešení postavit na míru a základní balíčky rozšířit o nové funkce. Získá také solidní základ pro vybudování chytré domácnosti.

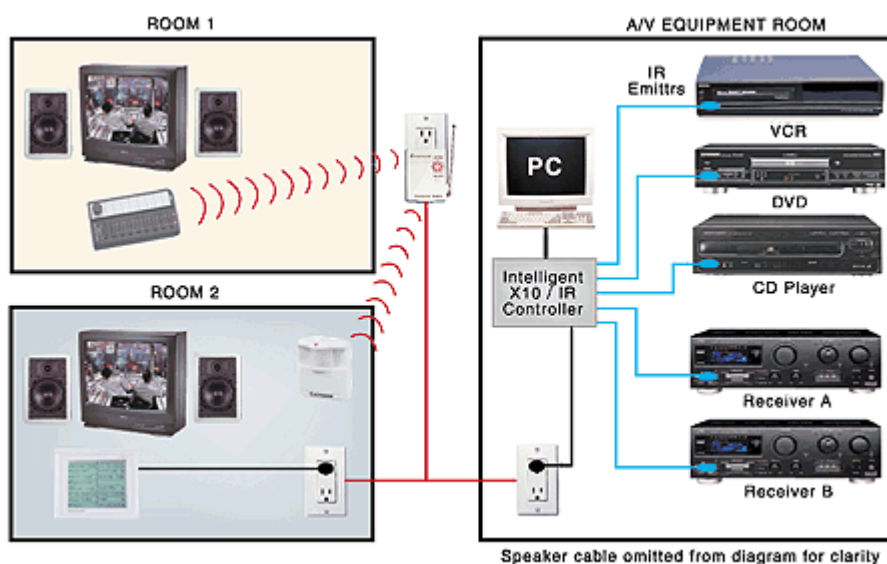
#### **2.1.4 Multimédia**

Multimediální systémy bývají, stejně jako systémy vytápění, celkem běžnou výbavou domácnosti. Ať už se jedná o domácí kino, různé audio systémy, nebo v poslední době stále rozšířenější NAS servery. Multimediální systémy jsou samostatnou kapitolou samy o sobě a také v rozpočtu při vybavování nového domu či bytu mohou být značně vysokou položkou v závislosti na velikosti pokrývaného objektu, požadovaných funkcí a kvalitě výstupu.

Úkolem multimediálního systému je propojit vstupy pro multimédia se správnými výstupy. Vstupem může být hudební přehrávač, televize, počítač, NAS server atd. Mezi výstupy patří například sestavy reproduktorů, obrazovky nebo projektory. Televize může sloužit jako výstup pro obraz a zároveň vstup pro zvuk, do kvalitního

stereo ozvučení. Všechno je pak propojené přes AV přijímač nebo multimediální centrum. Obojí může být součástí centrálního prvku chytrého domu. Propojení mezi jednotlivými prvky může být drátové i bezdrátové. Výhodou je, pokud AV přijímač umožňuje zapojení do domácí datové sítě a podporuje síťové protokoly UPnP (Universal Plug and Play). Pokud chceme ozvučit každou místnost jinak, je potřeba aby AV přijímač disponoval funkcí multizone. Ta umožní vytvořit různé zóny ozvučení a poslouchat tak v každé zóně (místnosti, podlaží) audio z jiného zdroje. [24][25]

Na Obrázek 10 je vyobrazeno zapojení multimediálního systému do chytrého domu. V pravé části je vidět několik zdrojů a AV přijímače, zapojené pro podporu multizone audia. V levé části pak výstupní zařízení v různých místnostech. Vše propojené přes centrální prvek chytrého domu. V tomto konkrétním případě na platformě X10.



**Obrázek 10: zapojení multimediálního systému v chytrém domě na platformě X10. Zdroj [25]**

Chytré funkce do multimediálního systému přináší až propojení s řídicím prvkem chytrého domu a integrací s ostatními systémy. Hudba může uživatele sledovat během jeho pohybu po domě a pouštět se podle určitého plánu (např. místo budíku), situace, nebo nálady uživatele. Audio systém také poslouží pro komunikaci chytrého domu s uživatelem a může ho notifikovat o veškerém v jeho jazyce. Dále poslouží multimediální systém i při propojení se zabezpečovacím systémem nebo zvonkem

u dveří. V prvním případě může posloužit hlasitá hudba místo alarmu. V tom druhém bude zvonek zvonit jen v určité místnosti, případně v noci bude místo zvuku využito blikání světel. [26] Tyto funkce přináší například inteligentní dům od firmy Loxone ve spojení s jejich hudebním serverem (na Obrázek 11).



**Obrázek 11: Loxone music server. Zdroj [27]**

### 3 Hardwarové prvky v inteligentním domě

Zdroj [28] definuje pojem Smart Home jako systém automatizace budovy, konkrétně soukromého domu nebo bytu, ve kterém jsou samostatné komponenty (senzory a aktuátory) propojené do jedné sítě, aby zformovaly inteligentní systém. Sensory a aktuátory tvoří technologický základ pro rozsáhlý sběr dat a vykonávání akcí v daném prostředí. Nashromážděná data jsou pak poslána a vyhodnocena v řídicím prvku celého systému.

Z této definice vycházejí základní hardwarové prvky inteligentního domu, ale i jakéhokoliv jiného inteligentního prostředí. Jedné se právě o řídicí jednotku, senzory a aktuátory, doplněné o ovládací prvky, pomocí kterých může uživatel prostředí ovládat a komunikovat s ním.

#### 3.1 Řídicí jednotky

Řídicí jednotka je mozkem celého chytrého domu. Řídicí jednotka ovládá všechny komponenty v domě, které jsou k ní připojené a na základě dat z různých senzorů, času, aktuálního počasí atd. je schopná činit autonomní rozhodnutí bez nutnosti zásahu uživatele. Takto definuje svou řídicí jednotku zvanou Miniserver firma Loxone, zabývající se výstavbou inteligentních domů [29]. Na Obrázek 12 je klasická verze řídicí jednotky Loxone Miniserver. Na Obrázek 13 pak jeho bezdrátová verze Miniserver Go. 81



Obrázek 12: řídicí jednotka Loxone Miniserver. Zdroj [29]



Obrázek 13: bezdrátová řídicí jednotka Loxone Miniserver Go. Zdroj [29]

Řídící jednotky firem zabývajících se inteligentními budovami a elektroinstalací jsou v podstatě rozvaděči s počítačem velmi podobným jednotkám PLC – Programmable Logic Controller, neboli programovatelným automatům. V dnešní době jde o lehce programovatelné, modulární jednotky s mnoha vstupy a výstupy. Ty umožňují připojení velkého množství periférií a naprogramování jejich chování pomocí uloženého vývojového softwaru. Ten je v dnešní době vybaven grafickým rozhraním a požadované chování je možné pomocí něj definovat i bez programování. PLC tak nahrazují skříňě osazené velkým množstvím relé a umožňují upravit chování pro konkrétní sestavu zařízení. Původ PLC je v průmyslové automatizaci, a proto jsou připravené na zpracování velkého množství informací s minimální dobou odezvy. [30][31][32]



**Obrázek 14: PLC – programovatelný automat FATEK. Zdroj [30]**

Předchozí odstavce naznačují vlastnosti, které by měla řídicí jednotka mít. Z nutnosti připojit velké množství různých zařízení vyplývá, že řídicí jednotka musí disponovat hardwarovými i softwarovými rozhraními umožňující tato zařízení připojit. Už kvůli svému určení je zřejmé, že také musí obsahovat řídicí software vykonávající všechna rozhodnutí a komunikující s ostatními zařízeními v inteligentním domě.

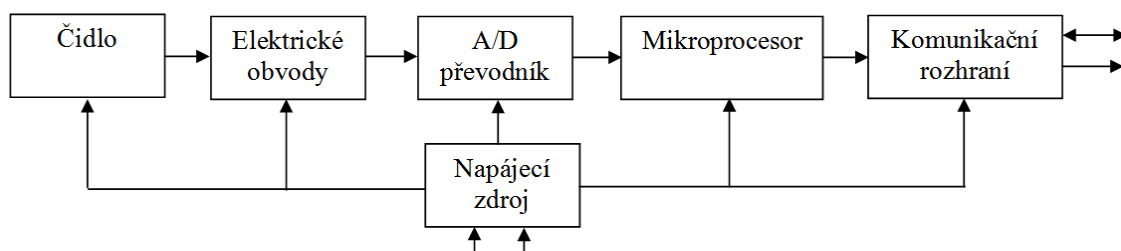
Vzhledem k rozsáhlé škále technologií umožňujících použití v inteligentním domě je vhodné podporovat alespoň nejvíce zaběhlé technologie a protokoly běžně využívané pro stavbu inteligentních domů. Případně umožnit rozšíření řídicí jednotky o moduly pro komunikaci s technologiemi, které samotná jednotka nepodporuje.

### 3.2 Senzory

Senzor, nebo také snímač či čidlo, je součástka schopná měřit určitou fyzikální veličinu a převést jí na signál. Tento signál je zpravidla předáván řídicí jednotce, která ho umí vyhodnotit. [33]

V problematice chytrých domů a automatizace budov jsou senzory vstupní periferií, která umožňuje sbírat potřebné informace o daném prostředí. Proto senzory tvoří jednu ze zásadních částí systému chytrého domu.

Zdroj [34] uvádí pojem inteligentní senzor jehož schéma zobrazuje Obrázek 15. Jedná se v podstatě o kombinaci čidla (citlivá část snímače) a dalších integrovaných obvodů na křemíkovém čipu. Jeho součástí mohou být A/D převodník a mikroprocesor pro předzpracování signálu z čidla. Důležitými součástmi pak jsou komunikační rozhraní a napájecí zdroj.



Obrázek 15: schéma inteligentního senzoru. Zdroj [34]

Čidla připojené do systému pomocí drátu využívají tyto dráty jako komunikační rozhraní i zdroj energie. Nevýhodou tohoto řešení je právě přítomnost drátu, který je potřeba schovat a s instalací těchto zařízení je vhodné počítat už v návrhu před samotnou stavbou nebo rekonstrukcí. Pokud jde ale o bezdrátové senzory, tak právě komunikační rozhraní v podobě rádiových vysílače a přijímače a zdroj napájení zabírají velkou část konstrukce celého senzoru. Zároveň je bezdrátová komunikace náročná na spotřebu energie.

Mezi bezdrátovými periferiemi pro domácí automatizaci vyčnívají produkty firmy EnOcean. Jejich platforma EnOcean Dolphin je zaměřena právě na vývoj autonomních bezdrátových periferií, které jako zdroj energie používají energii, kterou lze získat z prostředí. Jejich senzory a aktuátory obsahují převodníky energie, a to konkrétně pro převod lineárního pohybu, světla nebo tepla na energii postačující



pro fungování zařízení a bezdrátovou komunikaci. Příkladem mohou být termoelektrické hlavice topení, které získávají energii z převodu tepla, nebo snímače intenzity světla využívající solární energii. Tím odpadá potřeba napájecího zdroje v podobě baterie, kterou je potřeba kontrolovat a jednou za čas vyměnit. [35]

Konstrukce senzoru ovlivňuje další důležitou vlastnost senzoru a tou je design. Zatímco v průmyslové automatizaci je vhodné snímače zviditelnit, aby nedošlo k jejich poškození při provozu nebo údržbě, v rámci domácí automatizace se klade důraz spíše na nenápadnost nebo úplné skrytí senzorů, aby nenarušovali vzhled prostředí. Důležité je dbát na umístění senzorů, aby nebylo ovlivněno jejich měření. Například teplotní čidlo v blízkosti zdroje tepla. Některé senzory mohou být do určité míry uživatelsky nastavitelné a sloužit jako ovládání. Příkladem je nástěnný termostat. U takovýchto zařízení je velkým plusem nenápadný design a co nejjednodušší ovládání. [36]

Dle [33] lze senzory dělit pomocí jejich parametrů. Konkrétně podle:

- Měřené veličiny.
- Média sloužícího k přenosu signálu (elektrické, hydraulické, pneumatické atd.).
- Fyzikálního principu (způsob převodu naměřené hodnoty na hodnotu signálu).
- Druhu styku s prostředím (dotykové či bezdotykové)
- Podle stupně integrace.

Jako příklady senzorů používaných v inteligentních domech lze uvést kamery a mikrofony, snímače pohybu, detektory kouře, znečištění vzduchu nebo zaplavení, detektory otevřených oken, světelné senzory, teploměry a vlhkoměry.

### **3.3 Aktuátory**

Aktuátor, nebo aktor je v systému akční člen. Tento pojem se používá v mnoha oborech a v každém oboru může představovat trochu jiná zařízení. Za aktuátory tak lze označit lidské končetiny, stejně jako motorčky v podvozku robota. Zatímco senzory slouží v systému jako vstupy, aktuátory můžeme považovat za výstupy.

Zdroj [37] uvádí následující definici z pohledu robotiky. Aktuátor je zařízení měnící energii (elektrický signál) na pohyb. Obecnější definice z [38] říká, že aktuátor je nástroj k uvedení něčeho do akce.

Tyto definice můžeme upravit pro použití v oblasti inteligentních domů. Aktuátor je akční člen, který vykonává změny v prostředí inteligentního domu. V tomto případě se akce neomezuje pouze na pohyb, ale může jít o jakoukoliv změnu v prostředí inteligentního domu (např.: teplota, osvětlení). Prostředí pak nahrazuje to “něco” uvedené do akce, zmíněné v předchozí definici.

Z periferií použitých v inteligentním domě tak můžeme za aktuátory považovat například následující:

- Světla, žárovky
- Ventily, topení, vzduchotechnika a klimatizace
- Motorky (například na ovládání stínění nebo posuvných dveří), pumpy, čerpadla
- Reproduktory
- Zámky

Stejně jako senzory musí aktuátory disponovat komunikačním rozhraním a napájecím zdrojem. Jelikož aktuátory vykonávají více práce než senzory, je i jejich energetická náročnost vyšší. Proto je u nich častější využití drátů pro zapojení do systému. Ty zajistí dostatečné napájení i komunikační kanál.

V případě aktuátorů je také řešen design jako u senzorů s tím rozdílem, že některé periferie, které lze za aktuátory označit bývají standartní součástí i v obyčejných domech, a proto není důležité, aby byly skryté či nenápadné. Naopak je potřeba umožnit uživatelům některé aktuátory ovládat ručně, stejně jako v klasickém domě. Příkladem může být termostat, nebo zámek na dveřích.

### **3.4 Ovládací prvky**

Pro ovládací prvky v inteligentním domě by mělo platit to, že zůstanou zachovány stejně jako v normálním domě a umožní ovládání klasickým způsobem. Inteligentní dům by měl přinášet další možnosti ovládání, přesnější nastavení a větší komfort.

Inteligentní dům by tak měl nabízet kombinaci hned několika možností jeho ovládní:

- 1. Klasické ovládací prvky.** Sem spadají všechny ovládací prvky z klasického domu, které by měly být zachovány i v inteligentním domě. Vypínače, ovladače stínění, nástěnné ovládací panely klimatizace nebo vytápění. Slouží jako jedna z možností ovládní ale i jako záloha, pokud ostatní způsoby z nějakého důvodu selžou.
- 2. Automatické řízení.** Inteligentní dům by měl přinést uživatelům větší komfort s co nejmenším zapojením obyvatel. Proto by velkou část ovládní všech částí domu měl zajistit řídicí systém automaticky na základě úvodního nastavení uživatelů, postupného učení jejich preferencí a přednastaveného chování na standartní situace.
- 3. Ovládní pomocí PC nebo smartphonu.** Ovládní pomocí osobních zařízení obyvatel domu je jednou z přidaných hodnot inteligentního domu. Uživatel může ovládat celý systém pomocí svého chytrého telefonu, tabletu nebo PC. Díky tomu může dům ovládat v podstatě odkudkoliv. Stačí aby řídicí systém disponoval rozhraním pro komunikaci s mobilní nebo webovou aplikací. Stejně tak je to způsob, jak uživatelům zprostředkovat informace o systému, statistiky, úspory atd.
- 4. Ovládní hlasem.** Jedná se o jednu z nejpokročilejších metod ovládní, která je náročná z hardwarového i softwarového hlediska. Pro fungování je potřebné vybavit objekt dostatečným množstvím mikrofونů. Z pohledu softwaru j potřeba do řídicího systému implementovat rozpoznávání lidské řeči a do určité míry porozumění zadávaným příkazům.

## 4 Technologie pro komunikaci v inteligentním domě

Koncept inteligentních domů vznikl na základech průmyslové automatizace, kdy se od velkých výrobních hal, mrakodrapů a budov jako jsou nemocnice došlo až k menším objektům v podobě rodinných domů či bytů. Během tohoto vývoje vzniklo mnoho technologií a protokolů umožňující vybudovat systém chytré budovy.

Některé z těchto technologií jsou popsány dále v této kapitole. Technologie lze rozdělit podle způsobu propojení jednotlivých komponent výsledného systému, a to na drátové a bezdrátové. Především mezi technologiemi využívající zapojení pomocí drátů lze nalézt protokoly a sběrnice, které vznikaly společně s celým konceptem automatizace budov, kdy bezdrátové technologie ještě nebyly tak běžné.

### 4.1 Drátové technologie

Do této skupiny spadají veškeré technologie, které používají k propojení prvků systému a jako přenosové médium drát. Může se jednat o jakýkoliv typ drátu. Například elektrické rozvody v domě, kroucená dvojlinka nebo optické vlákno

#### 4.1.1 Power line Communication

Power line communication, zkráceně PLC je technologie využívající rozvodů elektrického vedení k přenosu dat. Lze tak vytvořit domácí síť s využitím stávající elektroinstalace bez nutnosti instalace další kabeláže. Stačí pouze do zásuvek připojit PLC adaptéry, zaměnit vypínače, případně další prvky elektroinstalace za kompatibilní s PLC.

Technologie funguje díky rozdílné frekvenci elektrického napětí a napětí datového signálu. Zatímco elektrické napětí je velké a pomalé, datový signál má napětí malé ale s vysokou frekvencí. Proto lze oba signály spojit do jednoho, aniž by docházelo k nežádoucímu rušení. Znárodnění lze vidět na Obrázek 16. Vlevo podoba samotného napětí, uprostřed vlnovka datového signálu a vpravo výsledek jejich spojení. Oddělení obou signálů je pak úkolem již zmíněných PLC adaptérů. [39]



**Obrázek 16: Princip technologie Power line Communication. Přenos informací pomocí elektrických rozvodů. Zdroj [39]**

Z principu PLC pak vycházejí i některé dále popsané technologie.

#### 4.1.2 X10

X10 je průkopnickou technologií v oblasti domácí automatizace využívající řízení přes elektroinstalaci, která byla představena již v roce 1975. Projekt X10 vznikl jako desátý projekt firmy Pico Electronics, která se zabývala výrobou kalkulátorů. Brzy se produkty využívající technologie X10 začaly objevovat v předních amerických elektroprodejnách. Jednalo se především o vypínače, moduly pro ovládání lamp či osvětlení a zásuvky. V nadcházejících letech se produktová řada X10 začala rozšiřovat o řídicí prvek, který umožnil ovládání dálkovým ovladačem, interface pro komunikaci přes počítač, nebo plnohodnotný bezpečnostní systém. Díky své modulárnosti a dostupnosti pro velkou masu lidí se technologie X10 stala standardem v domácí automatizaci. [1]

X10 vysílá signály o frekvenci 120 kHz skrze elektrické rozvody domu a činí tak ve chvílích, kdy sinusoida napětí prochází nulou, aby docházelo k co nejmenšímu rušení. Většina periférií X10 jsou krabičky připojené do zásuvek, do kterých se dále zapojují spotřebiče. Případně se instalují jako součást elektrického obvodu daného systému (např.: garážová vrata). Většina těchto krabiček disponuje dvěma kolečky pro nastavení adresy dané jednotky. Každému produktu tak může být nastavena adresa v rozsahu A-P (kód domu) a 1-16 (kód jednotky), což poskytuje 256 unikátních adres. [40]

Kód domu byl zamýšlen jako ochrana před rušením mezi více domy v sousedství využívající X10. V případě, že domy sdílí připojení do elektrické distribuční sítě,

mohlo docházet k protečení signálu do vedlejšího domu, jelikož signál se v síti může volně šířit. [41]

Technologie X10 je jednosměrná, což znamená, že nedochází k potvrzování přijetí signálu, a proto je každý signál poslán dvakrát. Signálu trvá přibližně půl až tři čtvrtě vteřiny, než dorazí k příjemci. To vede k viditelným prodlevám v ovládání. Zřetelně je to poznat například při stmívání a rozsvěcení světel. X10 je oproti modernějším technologiím poměrně náchylná k útlumu signálu a rušení. Proto jsou běžně využívány různé filtry, spojky a opakovače. [40]

### **4.1.3 Universal powerline bus**

Universal powerline bus, zkráceně UPB je proprietární protokol komunikace prostřednictvím PLC, tedy pomocí elektrických rozvodů. Je jedním z následovníků protokolu X10 a snaží se vylepšit jeho nedostatky. UPB poskytuje vysokou spolehlivost dosahující 99,9 % oproti 70-80% spolehlivosti X10. Spolehlivost je definována jako počet párů vysílačích a přijímacích jednotek, které správně fungují po prvotním zapojení. Zdroje [42], [43] tak uvádí 100-1000krát větší spolehlivost oproti X10 a 10-100 krát větší spolehlivost oproti technologiím CEBUS a LonWorks. [42] [43]

UPB nabízí jednoduchost, cenovou dostupnost i spolehlivost, bez nutnosti instalace další kabeláže. Za pomoci stávající kabeláže a příslušenství, které je cenově srovnatelné s běžným příslušenstvím, lze dosáhnout vyšší kvality bydlení. Pro jednodušší systémy lze využít peer-to-peer komunikace, která nevyžaduje žádný centrální prvek. Komunikace v UPB je navržena jako obousměrná, tudíž je přijetí signálu potvrzováno. Systém adresace je navržen tak, aby zvládl obsloužit až 256 domů na jednom transformátoru a 256 zařízení v každém domě. Adresní prostor tak umožňuje přiřadit 64 000 adres. [43]

UPB využívá takzvané pulzně poziční modulace jako metody pro posílání dat. Signály mají napětí 40 voltů a frekvenci 4-40 kHz. Tyto vlastnosti zajišťují menší náchylnost na šum a útlum signálu. Signál UPB je možné přenášet až jednu míli daleko. Rychlost signálu je 20–40krát vyšší než u X10. Latence je zhruba 0,1 vteřiny a lze tedy dosáhnout až deseti zpracovaných příkazů za vteřinu. Porovnání technických parametrů technologií UPB a X10 ukazuje Tabulka 2. [43] [44]

	<b>UPB</b>	<b>X10</b>
<b>Napětí signálu</b>	40 V	5–7 V
<b>Frekvence signálu</b>	4–40 kHz	120 kHz
<b>Datová propustnost</b>	240 bit/s	120 bit/s
<b>Obousměrná komunikace</b>	Ano	Pouze u dražších zařízení
<b>Adresní prostor</b>	~64 000	256 (v základu, bez rozšíření)

**Tabulka 2: Porovnání technických parametrů technologií UPB a X10. Zdroj [44]**

## **4.2 Bezdrátové technologie**

Jako bezdrátové technologie můžeme označit veškeré technologie, používající k propojení dvou prvků jiného než mechanického způsobu (kabelem). Obecně se jedná o přenos pomocí elektromagnetického záření, případně zvuku. V oblasti inteligentních budov se setkáme především s využitím rádiové frekvence případně infračerveného světla. [45]

V této kapitole jsou popsány především nejrozšířenější technologie založené na rádiových vlnách. Jejich porovnání je shrnuto na konci kapitoly v Tabulka 3.

### **4.2.1 Wi-Fi**

Wi-Fi je nejrozšířenější bezdrátovou technologií pro budování lokálních sítí a přístupu k internetu. Jedná o obchodní značku, za kterou se skrývají standardy pro bezdrátovou komunikaci IEEE 802.11. O její rozvoj se stará Wi-Fi aliance založená v roce 2000. [46]

Wi-Fi operuje v ISM pásmech<sup>1</sup> a to především 2,4 GHz a 5 GHz. Je stavěná pro velkou rychlost a přenosy velkého množství dat. Její dosah je závislý na prostředí a použité frekvenci, ale pohybuje se v desítkách až stovkách metrů. V extrémních případech může být dosah až několik kilometrů. [47]

Díky těmto parametrům je Wi-Fi energeticky velmi náročná. Z hlediska domácí automatizace je tak vhodná spíše pro prvky připojené stále do elektrické sítě, jako jsou multimediální prvky (domácí kina, ozvučení, centrální prvky, spotřebiče), nebo ostatní chytrá zařízení v domácnosti (smartphony, tablety, laptopy). Není vhodná pro prvky napájené baterií, jako jsou zámky nebo senzory.

### **4.2.2 Bluetooth**

Bluetooth je technologie pro bezdrátovou komunikaci na krátké vzdálenosti, vyvinutá v roce 1994 firmou Ericsson. Standardizační organizace IEEE tuto technologii standardizovala a označuje IEEE 802.15.1, ale již tento standard neudrhuje. To má na starosti organizace Bluetooth Special Interest Group (SIG). [49]

K výměně dat na krátkou vzdálenost využívá Bluetooth krátkých rádiových vln o frekvenci 2,4 GHz až 2,485 GHz (ISM pásmo). Bluetooth bylo od začátku určeno pro přenosná zařízení, a proto dbalo na co nejmenší energetickou náročnost. S tím se pojí i jeho dosah, který se pohyboval kolem deseti metrů. V závislosti na prostředí šlo dosáhnout i sta metrů. Tuto hranici ovlivní i verze bluetooth, jelikož novější verze se snaží být rychlejší, úspornější a s větším dosahem.

S verzí 4.0 bylo představeni i takzvané Bluetooth Low Energy (BLE), které je zaměřené právě na senzory a IoT obecně, kde je energetická úspornost vyžadována. Bluetooth se snaží vylepšit vlastnosti předchozích verzí, a ještě se tak přiblížit použití v IoT. V nové verzi 5 slibuje až dvakrát vyšší rychlost (2 Mbit/s) a čtyřikrát větší dosah. [50][51]

---

<sup>1</sup> ISM pásmo – industrial, scientific and medical band. Nelicencovaná pásma radiového spektra vyhrazená pro využití v průmyslu, vědě a medicíně. Některé z těchto pásem se využívají i pro datové přenosy. [48]



### 4.2.3 ZigBee

ZigBee je bezdrátová technologie založená na standardu IEEE 802.15.4, který je určený k vytváření mesh sítí<sup>2</sup>. ZigBee je díky tomu velmi jednoduchá technologie umožňující vyvářet složité sítě s minimálními energetickými nároky. Cenou za to je nižší přenosová rychlost a dosah.

Globálně funguje ZigBee v pásmu ISM tedy 2,4 GHz s rychlostí až 250 kb/s. V Americe a Austrálii na frekvenci 915 MHz, v Evropě 868 MHz a v Číně na 784 MHz. U těch už rychlost dosahuje pouze 20 kb/s. Teoretický dosah v ideálních podmínkách může být až 100 metrů. Prakticky je dosah v nízkých desítkách metrů, tedy kolem 20 metrů maximálně. [53]

Využití ZigBee lze najít v mnoha odvětvích. Může řídit osvětlení, žaluzie, topení a domácí spotřebiče jako myčky a pračky. Proto je vhodný pro domácí automatizaci. Dále pak pro průmyslovou automatizaci, zdravotnictví a průmyslový monitoring. O ZigBee se stará mezinárodní společenství elektrotechnických firem ZigBee Alliance, mezi kterými lze najít světovou špičku (Samsung, Philips, AT&T, Cisco, Huawei, Intel, Whirlpool, LG a mnoho dalších). [54]

### 4.2.4 Z-Wave

Z-Wave je další bezdrátovou technologií. Vznikla především pro domácí automatizaci a smart homes. Správcem je Z-Wave aliance založená v roce 2005, jejímiž členy jsou přední technologické firmy jako LG Uplus, Samsung SmartThing, Honeywell či Huawei.

Z-Wave funguje na principu mesh sítí. Umožňuje vytvoření různě velkých sítí, které je možné jednoduše rozšiřovat a modifikovat. Pracuje na rádiové frekvenci do 1 GHz, čímž je odolná rušení prvky v pásmu 2,4 GHz, jako je Wi-Fi nebo Bluetooth. Pro Evropu jde o frekvenci 868 MHz a pro Ameriku 908–916 MHz. Jedná se o obdobné frekvence jako u technologie ZigBee.

---

<sup>2</sup> Mesh síť – síť s hustým propojením prvků. Pokud je každý prvek sítě propojený s každým dalším, hovoříme o plné mesh síti. Každý prvek v síti si je rovnocenný a může tak sloužit k šíření zpráv v síti. [52]

Jedná se o nízkoenergetickou technologii, čemuž odpovídá i přenosová rychlost. Ta dosahuje maximálně 100 kb/s. Dosah je kolem 30 metrů. U novějších čipů zhruba 40 metrů. Pro mesh síť je tato vzdálenost dostačující, jelikož jednotlivé prvky (ty které závislé na baterii) přeposílají zprávy dále do cílové destinace, a tak není problém pokrýt celou domácnost. Velikost sítě může být až 232 prvků, přičemž při větším počtu prvků je síť robustnější a lépe optimalizuje cestu zasílaných zpráv. [55][56]

	<b>Wi-Fi</b>	<b>Bluetooth / BLE</b>	<b>ZigBee</b>	<b>Z-Wave</b>
<b>Teoretická velikost sítě</b>	256	9 / neuvedeno	256	232
<b>Způsob přenosu</b>	Rádiové vlny	Rádiové vlny	Rádiové vlny	Rádiové vlny
<b>Frekvence</b>	2,4 GHz, 5 GHz	2,4 GHz	0,868 MHz, 0,915 MHz, 2,4 GHz	<1 GHz
<b>Přenosová rychlost</b>	Až 600 Mb/s (dle konkrétního standardu)	Až 2 Mb/s	20 kb/s (<1 GHz), 250 kb/s (2,4 GHz)	100 kb/s
<b>Maximální dosah</b>	Stovky metrů	<100 m / ~50 m	~20 m	~30-40 m
<b>Topologie sítě</b>	Hvězda	Hvězda	Mesh	Hvězda + mesh
<b>Energetická náročnost</b>	Vysoká	Střední / velmi nízká	Velmi nízká	Velmi nízká

**Tabulka 3: Porovnání vybraných bezdrátových technologií. Vychází z [57][58]**

#### **4.2.5 EnOcean**

Jedná se taktéž o bezdrátovou technologii, která je méně známá než výše popsané, ale je třeba jí zmínit. Je výjimečná svým konceptem. Jde v podstatě o platformu pro bezúdržbové a bezdrátové prvky (senzory, aktory...). EnOcean je patentovaná technologie bezdrátové komunikace sbírající vlastní energii. Díky tomu je zajištěno, že prvky nepotřebují baterie k napájení a žádné dráty k připojení. To umožňuje jednoduše rozšiřovat systémy postavené na této technologii. Ty jsou navíc velmi modulární a jednotlivé prvky lze jednoduše přemísťovat dle potřeby.

Komunikace probíhá po rádiových vlnách ve frekvencích ISM pásma a to 868 MHz pro Evropu, 902 MHz pro USA a Kanadu, 928 MHz pro Japonsko a 2,4 GHz celosvětově. Právě využití frekvence 2,4 GHz zajišťuje možnost spolupráce se systémy ZigBee a BLE. EnOcean je také kompatibilní se sběrnici KNX, LON (představenými v následující kapitole), nebo i DALI, BACnet a TCP/IP.

EnOcean je energeticky velmi nenáročný, a navíc si sbírá vlastní energii. To může probíhat třemi způsoby, dle potřeb konkrétního prvky. Energie je sbírána ze světla, tepla nebo pohybu. Energetická nenáročnost je dosažena vysíláním velmi krátkých signálů. Jejich dosah může být až 30 metrů uvnitř budovy. [59]

### **4.3 Komunikační protokoly a standardy**

V této kapitole jsou zahrnuty technologie, které využívají jak drátové, tak bezdrátové komunikace. Jedná se tak spíše o protokoly či standardy využívající k fyzickému přenosu informací některé z technologií popsaných v předchozích dvou kapitolách.

#### **4.3.1 KNX**

KNX je celosvětový standard pro domácí automatizaci. Jedná se o sběrnici k propojování prvků v systému domácí automatizace. Je spravován KNX Asociací a vychází a sjednocuje dřívější standardy automatizace, kterými jsou BatiBus, EHS a především EIB (European Installation Bus), na kterém stojí základ KNX. [60]

Asociace KNX vznikla v roce 1999. KNX je schválena jako standard v Evropě, Kanadě a Číně, což v roce 2006 vedlo ke vzniku mezinárodního standardu ISO/IEC 14543-3. KNX umí k přenosu využívat následující média:

- Kroucená dvojlinka (zdeděno od BatiBus a EIB standardů)
- Elektroinstalace (zdeděno od EIB a EHS)
- Rádiové vlny
- IR
- Ethernet

KNX definuje transportní a aplikační vrstvu, hierarchii adresování, strukturu uzlů a komunikujících zařízení. KNX umožňuje napájení prvků přímo po sběrnici. Maximální velikost sítě činí 1000 metrů a maximální vzdálenost mezi dvěma prvky může být 700 metrů. Mezi prvky funguje peer-to-peer komunikace s možností režimu Multicast a Broadcast. Adresace umožňuje mít v síti až 65 tisíc prvků, tím připadá 256 prvků na každou podsít'.

Sběrnice KNX dokáže řídit všechny systémy důležité pro provoz inteligentního obydli, které již byly zmíněny v předchozích kapitolách (světla, stínění, bezpečnost, klimatizace a topení, multimedia, spotřebiče). [61][62]

### **4.3.2 LonWorks**

Firma Echelon vyvinula technologii zvanou LonWorks počátkem 90. let minulého století, jako levné komunikační spojení pro různá technická použití na nejnižší automatizační úrovni. Echelon se nyní obecně zabývá vývojem IoT platformy pro chytrá města, kampusy a budovy. [63]

Technologie LonWorks se skládá ze sběrnice LON (Local Operating Network), protokolu LonTalk a čipu zvaného Neuron. LonTalk je protokol, založený na referenčním modelu ISO/OSI, jehož vrstvy definuje. Echelon pak vyvinul komunikační čip Neuron, součástí jehož firmwaru je právě protokol LonTalk, a který umožňuje zařízením komunikovat po sběrnici LON. Topologie je odvozena od počítačových sítí. Technologie LonWorks je otevřená a protokol LonTalk už je veden pod vlastním standardem, takže je nezávislý na čipu Neuron.

Jako fyzická vrstva může sloužit v podstatě jakékoliv běžné přenosové médium. Konkrétně jde o kroucenou dvojlinku, koaxiální kabel, elektroinstalaci, rádiové vlny či infračervený přenos. Komunikace pak probíhá formou peer-to-peer a v síti může být připojeno až 32000 zařízení. Technologie LonWorks je vhodná pro širokou škálu využití. Nejen pro domácí automatizaci, ale i pro průmysl, monitoring spotřeby, nebo řízení dopravy. [64][65][66]

### 4.3.3 Modbus

Modbus je otevřený protokol pro komunikaci různých zařízení (PLC, dotykové displeje, I/O rozhraní...), který umožňuje komunikaci typu klient-server, neboli master-slave, pomocí různých typů sběrnic a sítí. Vyvinut byl v roce 1979 firmou Modicon a kromě průmyslového využití najde uplatnění i v odvětvích energetických, dopravě a domácí automatizace.

Protokol Modbus funguje na úrovni aplikační vrstvy modelu ISO/OSI a je protokolem typu žádost / odpověď (request / response) a poskytuje přenos služeb specifikované kódem funkce. Kód funkce říká serveru, jakou operaci má vykonat. Rozsah kódů je 1 až 255, přičemž kódy 128 až 255 jsou vyhrazené pro záporně odpovědi (chyby). Některé kódy funkcí obsahují i kódy podfunkce, které tak operaci ještě upřesňují. Modbus definuje strukturu zprávy nezávisle na typu komunikační vrstvy na úrovni protokolu (PDU – Protocol Data Unit). Na základě použitého typu sítě, na které je protokol použit, je pak zpráva rozšířena o další části a tvoří zprávu na aplikační úrovni (ADU – Application Data Unit). [67][68]

Modbus definuje kromě aplikační vrstvy ISO/OSI i implementace pro další typy sběrnic nebo sítí. Konkrétně se jedná o tyto:

- Modbus na TCP/IP
- Modbus na sériové lince (RS - 232C, RS - 422, RS - 485, vlákno, rádiový přenos)
- Modbus Plus – vysokorychlostní síť

## 5 Představení vybraných existujících řešení

V této kapitole jsou představeny existující systémy pro řízení chytré domácnosti. Jsou zastoupeny profesionální systémy od firem, které poskytují řešení na klíč, nebo open source systémy, které lze použít při stavbě vlastních řešení.

### 5.1 Loxone

Firma Loxone vznikla v roce 2008 jako projekt Thomase Mosera a Martina Öllera, kteří hledali řešení domácí automatizace pro vlastní domovy. Od té doby se Loxone snaží vyvíjet své řešení Loxone Smart Home, tak aby dělalo uživatelům radost. Jeho úkolem je dělat to co uživatel sám nechce, nebo nemůže, ale zároveň nechat uživateli vše pod kontrolou.

Loxone vyvíjí svůj vlastní hardware i software. Dokáže ovládat všechny důležité systémy v domě a učít se dle požadavků uživatele. Srdcem Loxone Smart Home je řídicí jednotka zvaná Miniserver, která je představena již v kapitole 3.1 Řídicí jednotky. Je vyhotovena ve dvou verzích. Klasické (PLC) a Air (bezdrátové). Díky tomu je systém vhodný pro novostavby i hotová obydlí. Loxone vyvinul i svou sběrnici Loxone Tree, která umožňuje jednoduché přidávání prvku a snižuje potřebné množství kabeláže. Také zajišťuje snadnou aktualizovatelnost prvků a škálovatelnost. V nabídce má Loxone celou řadu prvků, včetně vlastního multimediálního serveru a zesilovače. Systém Loxone Smart Home, je tak schopen nabídnout řízení osvětlení, žaluzií a stínění, regulaci teploty, zabezpečení, kontrolu energií a řízení multimédií. Zajímavým prvkem je i Loxone vypínač, který umožňuje ovládání hned několika prvků. Hlavně osvětlení a hudby, v závislosti na počtu či délce stisknutí jednoho z pěti tlačítek skrytém ve vypínači, a to vše v jednom prvkem.

Řídicí jednotky fungují na systému Loxone OS, který je řídí. Jsou rozšiřitelné pomocí rozšiřujících modulů a díky nim je zajištěna i kompatibilita s ostatními technologiemi, jako jsou KNX, sériové linky, nebo EnOcean. [69]

### 5.2 Insteon

Insteon je samostatná technologie, která umožňuje postavit systém domácí automatizace. K tomuto účelu nabízí Insteon přes dvě stě zařízení vlastní výroby, jako jsou

centrální prvky, žárovky, vypínače, senzory atd. Insteon se snaží být co nejjednodušší, spolehlivý a cenově dostupný. Instalovat tento systém dokáže i zručný uživatel bez pomoci odborníka.

Zařízení od Insteonu jsou zpětně i dopředně vzájemně kompatibilní. Insteon umí komunikovat skrze elektrické rozvody, ale i bezdrátově pomocí full mesh sítě. Dokonce oba způsoby kombinuje a zprávy mezi nimi mohou plynule přecházet. Díky tomu je zajištěna vysoká spolehlivost a systém je vhodný pro novostavby i do hotových domů. Tento systém umožňuje ovládat všechny důležité systémy v domě, stejně jako výše zmíněný Loxone.

Do tohoto bodu je Insteon podobný ostatním systémům a dle [70] se snaží vyzdvihnout některé své výhody. Jednou z nich je využití simulcastu ve své full mesh síti. Jedná se o zasílání zpráv v síti přes více cest, což zajišťuje vysokou spolehlivost. Každý prvek v síti funguje jako opakovač a rozesílá zprávy dál do sítě. Díky tomu jsou ušetřeny zdroje na routování v síti a síť lze jednoduše modifikovat. Druhou zmíněnou výhodou pak je technologie Insteonu zvaná Statelink. Ta zajišťuje kompatibilitu všech zařízení skrze verze. Jednotlivá zařízení nemusejí znát všechny příkazy, které systém využívá, stačí když znají stav scény, ve které se nacházejí, když jsou do ní přidávány. [70] [71]

### **5.3 OpenHAB**

OpenHAB je open source řídicí systém naprogramovaný v jazyku Java, který si dává za úkol sjednotit a zjednodušit integraci různorodých systémů pro domácí automatizaci. Jeho cílem je tedy sjednotit tyto systémy do jednoho řešení.

Z tohoto cíle vyplývá, že systém se snaží být co nejvíce modulární a nezávislý na jakémkoliv výrobci či značce. Jelikož je vyvinut v Javě, je možné jej provozovat na jakémkoliv počítači s běžícím JVM (Linux, OS X, Windows, Raspberry Pi, Pine64...). Pro všechny tyto systémy poskytuje jednotné grafické rozhraní. Jako open source systém je spravován komunitou. Data lze ukládat do relačních i objektových databází.

OpenHAB disponuje systémem pro tvorbu pravidel a naprogramování požadované automatizace. OpenHAB ovšem není krabicovým řešením, které lze jednoduše použít. Je třeba se se systémem řádně seznámit a naučit se ho nastavovat. Tím pádem není vhodný pro všechny.

Popis ze zdroje [72] uvádí, že OpenHAB lze označit za systém systémů. Je orientovaný především na uživatele. Zatímco ostatní systémy nabízí sadu funkcí, OpenHAB dává uživateli možnost vytvářet funkce a případy užití dle vlastních potřeb a fantazie. To zaručí, že systém nezastarává tak rychle a je možné modifikovat jeho funkce. S ohledem na uživatele se OpenHAB zaměřuje i na data a jejich bezpečnost. Veškerá data v systému (data ze senzorů, příkazy aktuátorů) patří výhradně uživateli. Žádná data se nikam neodesílají a k systému není možné ani vzdáleně přistupovat, pokud to není uživatelem vyžadováno. Systém je tedy uzavřený a funguje v podstatě jako intranet i bez přístupu k internetu.

Zdroj [73] obsahuje seznam kompatibilních technologií. Mezi nimi jsou i nejznámější a nejpoužívanější systémy, z nich některé byly představeny v dřívějších kapitolách. Například KNX, EnOcean, Insteon, ale i HomeMatic a digitalStrom.

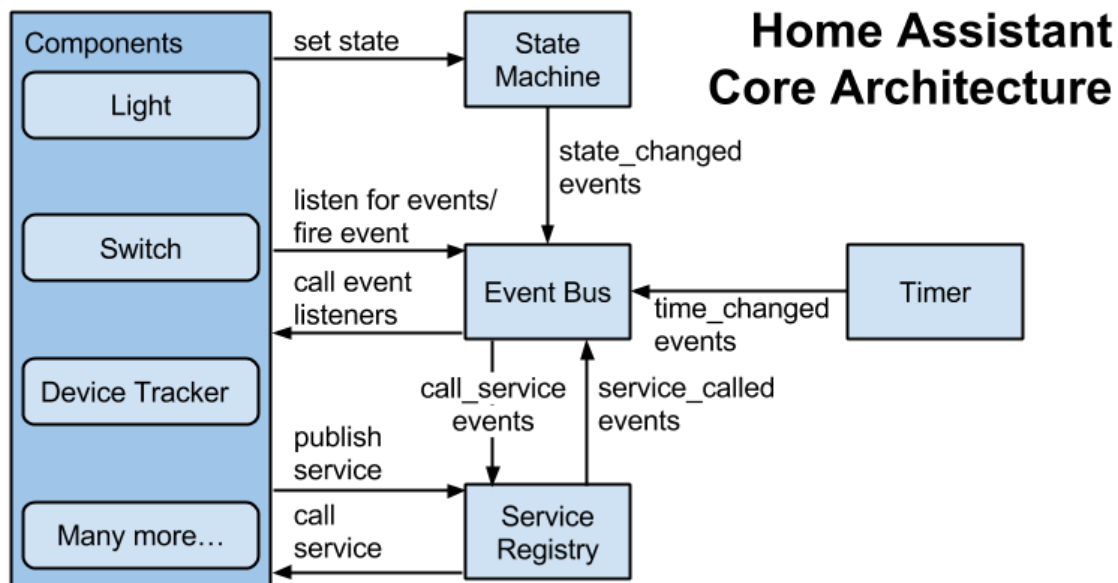
## **5.4 Home Assistant**

Dalším zástupcem open source řídicího systému domácí automatizace je Home Assistant. Tentokrát je systém naprogramován v jazyku Python 3 a stejně jako OpenHAB je vhodný pro provoz na Raspberry Pi.

Home Assistant je méně komplexní než OpenHAB, ale také umožňuje monitoring stavu všech zařízení v domě od různých výrobců, různých systémů a využívající různé technologie. Zdroj [74] konkrétně uvádí jako kompatibilní systémy Nest, Philips Hue, Google Cast, Plest i zařízení od firmy Ikea a mnohé další. Tento systém taky umožňuje sledování stavu všech zařízení pomocí jednotného uživatelského rozhraní pro telefony či web. V neposlední řadě nabízí automatizaci pomocí konfigurace pokročilých sad pravidel. [74]

Na Obrázek 17 je vyobrazena architektura jádra systému Home Assistant a způsob komunikace mezi jeho součástmi.





Obrázek 17: Architektura jádra systému Home Assistant. Zdroj [75]

Součásti jádra systému Home Assistant v pravé části Obrázek 17 v modrých obdélnících mají následující funkce:

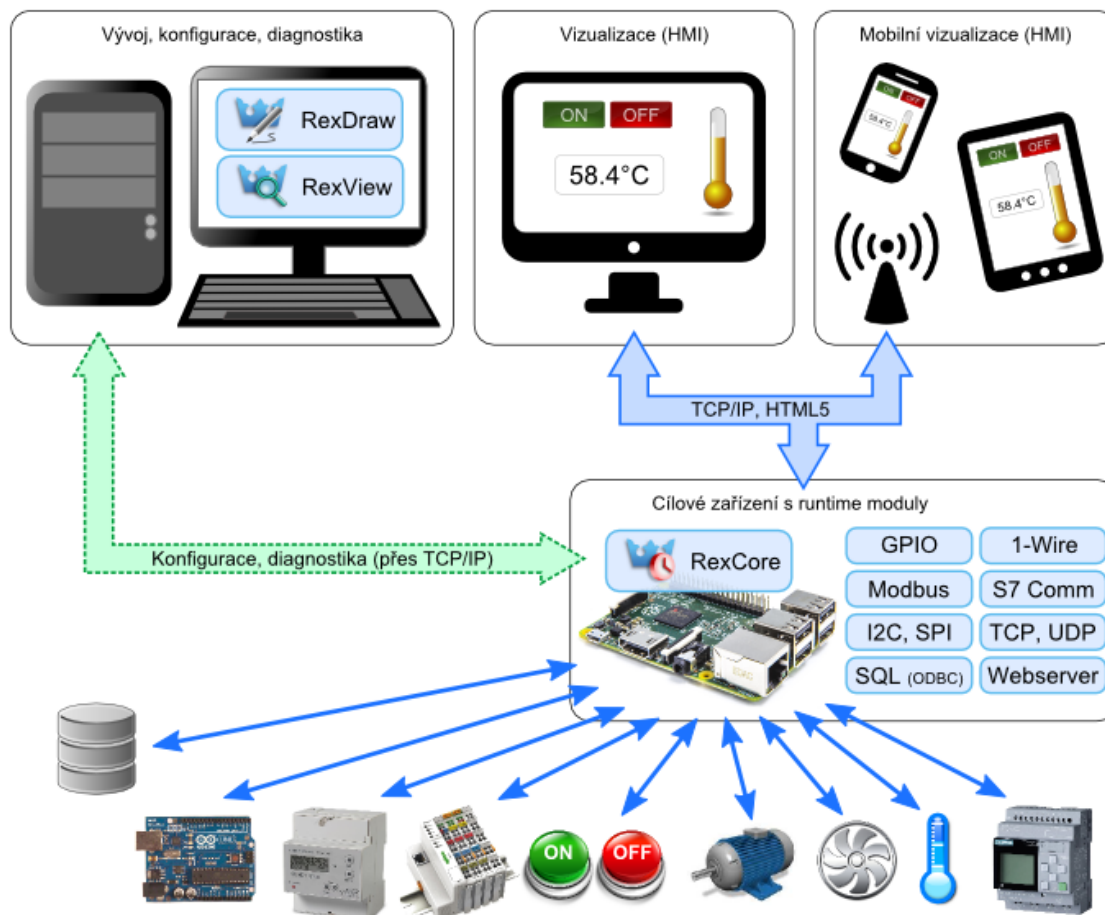
- **Event bus** – srdce systému. Naslouchá událostem v systému a obstarává jejich spouštění.
- **State machine** – monitoruje stav prvků v systému a vyvolává události spojené s touto změnou.
- **Service registry** – naslouchá sběrnici událostí a umožňuje registraci služeb.
- **Timer** – zastupuje v systému hodiny nebo časovač, podle kterého se řídí sběrnice událostí. [75]

## 5.5 Rex

Řídicí systém Rex je soubor softwarových nástrojů pro řízení strojů, technologií a procesů. Lze použít ve všech oblastech automatizace, robotiky, měření a regulace. Systém Rex nabízí možnost grafického programování bez nutnosti psát kód ručně, programování řídicích jednotek na běžném PC nebo notebooku a uživatelské rozhraní pro tablety, desktopy i smartphony. Podporuje širokou škálu zařízení a vstupně-výstupních jednotek.

System Rex je dostupný ve třech verzích pro různé platformy a použití. Konkrétně se jedná o verzi pro průmyslová PC (IPC) na Linuxu, verzi pro Raspberry Pi a verzi pro standardní PC (Windows).

Pro tuto práci je zajímavá především verze pro Raspberry Pi. Řídicí systém Rex ho promění v univerzální řídicí jednotku. Podporovány jsou všechny verze Raspberry Pi a rozšiřujících desek PiFace Digital, Intellisys PIO a UniPi.



**Obrázek 18: Vizualizace řídicího systému Rex a jeho komponent. Zdroj [77]**

Na Obrázek 18 je vidět řídicí systém Rex včetně všech jeho komponent a nástrojů. RexDraw je vývojové prostředí, ve kterém lze vytvářet složité algoritmy pomocí celé řady funkčních bloků. Vzniklé algoritmy je možné ihned zkompileovat a spustit. K převodu algoritmů do binární podoby slouží překladač RexComp. RexCore je jádrem systému, které jako jediná z komponent běží na cílové jednotce (Raspberry Pi) a zajišťuje chod systému dle naprogramovaných algoritmů. Veškerý chod systému je možné sledovat pomocí diagnostického nástroje s názvem RexView.

Uživatelské rozhraní si může uživatel nechat automaticky vygenerovat, nebo vytvořit zcela vlastní. Systém Rex nabízí uživatelské rozhraní v jazyce HTML 5, které je funkční na všech operačních systémech Windows, Linux, Android i iOS.

Kromě ovládání zařízení na GPIO pinech a rozšiřujících deskách umí systém ovládat i zařízení ze sběrnic 1-Wire, Modbus TCP nebo RTU, I2C a sériových linek RS-232, RS-485 nebo UART. Také umí ovládat PLC firmy Siemens a zařízení s komunikací TDP nebo UDP.

Řídící systém Rex má volnou testovací verzi, ve které algoritmy běží dvě hodiny a pak je třeba systém restartovat. Veškeré vývojové nástroje jsou také zdarma. Pro stálý chod algoritmů je potřeba zakoupit licenci. [76] [77]

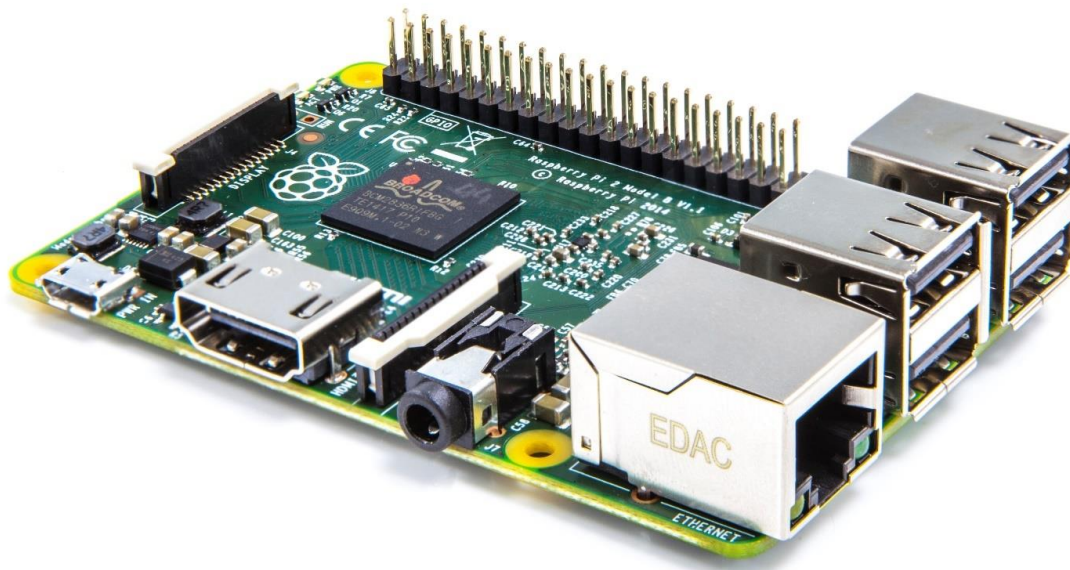
## 6 Využití Raspberry Pi v inteligentním domě

Tato kapitola se zabývá mini počítačem Raspberry Pi a jeho možnostmi k využití jako centrálního prvku v inteligentním domě. V kapitole je představeno samotné Raspberry Pi a moduly které lze pro návrh obdobných systémů využít. Dále jsou popsány hardwarové prvky použité k vytvoření a otestování prototypu.

### 6.1 *Raspberry Pi 2 B a rozšiřující moduly*

Raspberry Pi je miniaturní počítač velikosti kreditní karty, který byl vytvořen k výukovým účelům organizací Raspberry PI foundation. K tomuto účelu přispívá i nízká pořizovací cen, která se pohybuje kolem 1000Kč. Jelikož se jedná o plnohodnotný počítač se všemi potřebnými rozhraními, je možné jej použít i jako klasickou pracovní stanicí. Raspberry Pi je vybaveno i GPIO (General purpose input/output) piny [78], které slouží pro komunikaci s hardwarem na nejnižší úrovni. Tyto piny jsou softwarově programovatelné a lze do nich posílat, nebo z nich číst elektrický signál. Stejně tak je možné přes tyto piny instalovat další rozšiřující desky a tím rozšířit možnosti využití Raspberry Pi.

Díky těmto vlastnostem lze Raspberry Pi využívat nejen pro studijní účely. Běžně je využíváno jako základní řídicí prvek pro projekty z oblasti robotiky, automatizace, domácích serverových řešení a mnohé další.



**Obrázek 19: Raspberry Pi 2 model B. Zdroj [79]**

Pro účely této práce byl použit model Raspberry Pi 2 B, který je už druhou generací mikropočítačů Raspberry. Jeho specifikace jsou shrnuty v následujícím výčtu [79]:

- A 900MHz quad-core ARM Cortex-A7 CPU
- 1 GB RAM
- 4 USB porty
- 40 GPIO pinů
- Full HDMI port
- Ethernet port
- kombinovaný 3.5mm audio jack
- konektor kamerového rozhraní (CSI)
- Display interface (DSI)
- slot pro Micro SD kartu
- VideoCore IV 3D grafická karta

Procesor je architektury ARMv7. To znamená, že na něm mohou běžet operační systémy, které jsou pro architekturu ARM připravené. Jedná se o řadu GNU/Linux distribucí (např.: Debian, Raspbian, Gentoo) nebo i Windows 10.

V době psaní této práce již existuje třetí generace Raspberry Pi, které vedle rychlejšího procesoru nabízí i lepší možnosti konektivity. Konkrétně se jedná o zabudované moduly WiFi (802.11n Wireless LAN) a Bluetooth (Bluetooth 4.1 a Bluetooth Low Energy). [80]

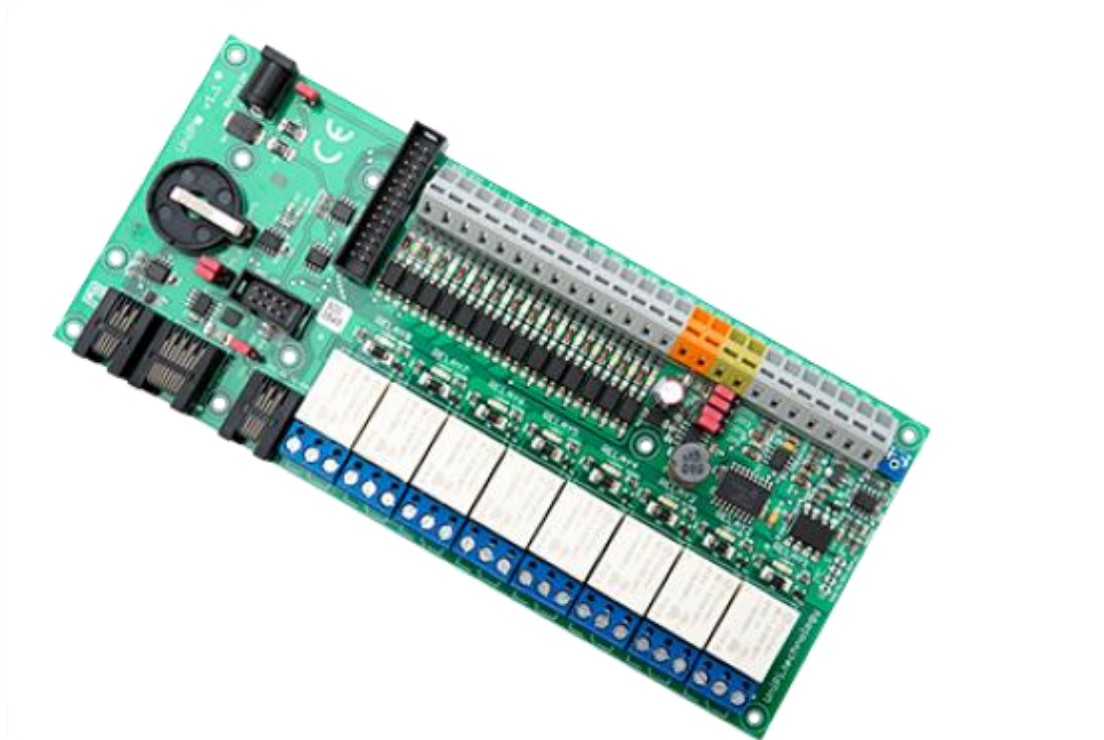
Dále jsou představeny rozšiřující moduly pro Raspberry Pi, které přidávají tomuto minipočítači nové možnosti komunikace. Jelikož aktuální verze Raspberry Pi již obsahuje integrovaný modul pro WiFi a Bluetooth, jsou zde zmíněny především moduly ostatních technologií (Z-Wave, ZigBee...).

### **6.1.1 Rozšiřující modul UniPi board**

Deska UniPi board (Obrázek 20) je rozšíření pro Raspberry Pi z české výroby. Společně vytvoří programovatelnou řídicí jednotku, kterou lze využít k řízení chytré domácnosti, pro měřicí a regulační systémy, systémy managementu budov či pro vývoj inovací v oblasti IoT. Cena tohoto rozšíření se je aktuálně 3800 Kč. Je kompatibilní s Raspberry model B+ Raspberry 2B a 3B. Spojením Raspberry, které poskytne CPU a UniPi board vznikne PLC podobné těm popsaným v kapitole 3.1 Řídicí jednotky.

Klíčové vlastnosti UniPi board jsou:

- 8× Přepínací relé pro 250 VAC/5A nebo 24VDC/5A
- 12× Galvanicky izolované digitalní vstupy spouštěné 5-24 V DC s minimální délkou pulsu 5 ms
- 2× 0-10V Analogový vstup
- 1× 0-10V Analogový výstup
- 1× 1-Wire port pro připojení 1-wire teploměrů a dalších senzorů
- 1× I<sup>2</sup>C port pro rozšíření
- 1× UART port pro externí sériovou komunikaci
- Slot pro baterii na napájení RTC
- Propojení s Raspberry Pi pomocí plochého kabelu
- Jednotku lze rozšířit o rozšiřující moduly přes I2C
- Lze doplnit o uchycení na DIN lištu a plastový kryt
- Napájení externím adaptérem 5V 2.5A s 2.1mm DC Jackem



**Obrázek 20: Rozšiřující deska UniPi board, která společně s Raspberry Pi vytváří programovatelnou řídicí jednotku. Zdroj [81]**

Celý systém ukládá data do databáze a lze jej monitorovat pomocí SCADA systému (dispečerské řízení a sběr dat). Systém lze rozšiřovat přes sběrnici I<sup>2</sup>C a umí komunikovat se zařízeními třetích stran pomocí sériového připojení RS485 na sběrnici Modbus.

Z pohledu softwaru má uživatel možnost volby. UniPi board může být řízen různým softwarem. Výrobce doporučuje například svůj vlastní řídicí software EVOK, celosvětovou řídicí platformu Codesys, nebo systémy Mervis či Rex.

Evolucí UniPi board je UniPi Neuron od stejných vývojářů. Rozdíl, mezi těmito dvěma technologiemi je především v potřebě Raspberry Pi. Zatímco UniPi board je rozšiřující modul a bez Raspberry Pi je nefunkční, UniPi Neuron už je plnohodnotné PLC s vlastním čipem, které je navrženo modulárně pro jednoduchou rozšiřitelnost a přidává nové funkce. [81]

### 6.1.2 Rozšiřující modul RaZberry (Z-Wave)

RaZberry je malá deska pro Raspberry Pi rozšiřující jeho možnosti o komunikaci skrze technologii Z-Wave. RaZberry (na Obrázek 21) je připojováno přímo k GPIO pinům Raspberry a neblokuje tak USB porty. Modul je napájen 3,3 volty a komunikuje sériově pomocí TTL (tranzistor-tranzistor logic) signálů. Firmware obsahuje všechny ovladače Z-Wave stack.

Modul disponuje transceiverem Sigma Designs ZM5202 Z-Wave, 32 Kb flash paměti a PCBA vestavěnou anténou. Kromě té lze připojit externí anténa a IPEX konektor. Dvě LED diody na modulu slouží k indikaci stavu Z-Wave čipu. Spotřeba modulu se pohybuje kolem 18 mA při 3,3 V, ale může se vyhoupnout až na 40 mA když modul vysílá. Modul je plně kompatibilní s Raspberry ve verzích B+, model 2 a model 3. [82] [83]



Obrázek 21: Modul RaZberry (Z-Wave). Zdroj [83]

### 6.1.3 Rozšiřující modul XBee (ZigBee)

K tomu, aby se Raspberry Pi naučilo komunikovat pomocí technologie ZigBee lze využít modul zvaný XBee od firmy Digi. Ten lze pořídit hned v několika variantách dle aktuální potřeby. V základu se dělí na dva typy: XBee a XBee-Pro. Rozdíl mezi nimi je v přidané anténě k modulu XBee-Pro, které má díky tomu i větší dosah. Obě verze jsou vidět na Obrázek 22.

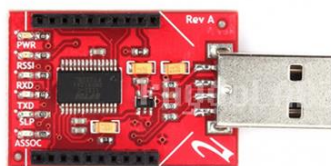


Firma Digi nabízí celou řadu produktů XBee, které se mohou lišit použitou frekvencí (868 MHz, 900 MHz a 2,4 GHz), nebo implementovanou technologií obecně. Nejjednodušší rozdělení modulů na čipy pro ZigBee/mesh sítě a na čipy typu point-to-multipoint. Do první skupiny patří moduly implementující technologie ZigBee, Protokol IEEE 802.15.4 pro mesh sítě, ale i implementace DigiMesh a ZNet 2.5 Mesh. Do skupiny point-to-multipoint lze zařadit moduly XBee-Pro 868 OEM RF Modul a XBee-Pro XSC. Jsou vhodné pro použití venku, jelikož jejich venkovní dosah je až několik desítek kilometrů (až 24 km u modelu XSC a až 40 km u modelu 868). [84]

Pro co nejjednodušší připojení XBee modulů k Raspberry Pi lze využít USB adaptér (Obrázek 23 a Obrázek 24), na který se modul umístí a následně je možné ho připojit pomocí USB portu. [85]



**Obrázek 22: Moduly XBee a XBee-Pro (s anténou). Zdroj [84]**



**Obrázek 23: USB adaptér pro XBee moduly. Zdroj [85]**



**Obrázek 24: XBee modul umístěný na USB adaptéru. Zdroj [85]**

#### **6.1.4 Rozšiřující modul EnOcean Pi 868 (EnOcean)**

Díky modulu EnOcean Pi je možné z Raspberry Pi vytvořit bránu do světa EnOcean a jeho energeticky nezávislých senzorů. Jelikož EnOcean nabízí širokou škálu prvků, které bude možné díky tomuto rozšíření skrze Raspberry Pi ovládat, jedná se o velmi užitečný modul. Lze s ním vytvořit základ pro systém inteligentního obydlí.

Modul pracuje na frekvenci 868 MHz a je napájen napětím 2,85 V až 3,5 V. Umí přenášet data rychlostí až 125 kb/s. EnOcean Pi disponuje radiovým a sériovým rozhraním pro příjem i odesílání zpráv. Radio zprávy jsou přenášeny transparentně

skrz sériové rozhraní v obou směrech do a z externě připojeného hostitelského procesoru či hostitelského PC. K tomuto modulu je možné stáhnout middleware zvaný EnOcean Link, který umožní rychlejší vývoj a integraci ostatních systémů. Tento modul může pracovat jako postmaster pro až 15 obousměrných senzorů pomocí Smart Ack technologie. Modul (Obrázek 25) lze pořídit za cenu kolem 600–700 Kč. [86] [87]



Obrázek 25: Rozšiřující modul EnOcean Pi 868. Zdroj [87]

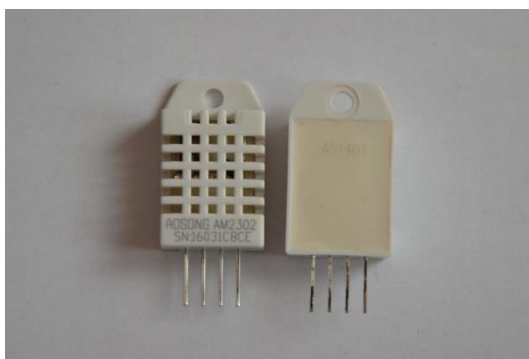
## 6.2 Hardware použitý k vytvoření prototypu

Součástí této práce je i návrh architektury systému pro řízení inteligentního domu. Na základě tohoto návrhu vznikl prototyp, k jehož otestování bylo potřeba několika hardwarových prvků, které jsou představeny v této podkapitole. Jedná se o skupinu senzorů a chytrou žárovku. Všechny komunikují skrze Raspberry Pi, které je centrálním prvkem systému.

### 6.2.1 Senzor teploty a vlhkosti

Obrázek 26 zobrazuje senzor teploty a vlhkosti vzduchu. Jedná se o čidlo DHT22, které pracuje pouze na jednom vodiči (protokol 1-wire). Senzor disponuje čtyřmi piny. Konkrétně 6V napájením, datovým pinem a zemí. Poslední pin je nevyužitý. Pracovní napětí je 3,3V-6V. Senzor je celkem přesný a dokáže snímat vlhkost vzduchu v rozsahu 0%-100% s přesností 2 až 5 %. Teplotu umí snímat v rozsahu -40 až

80 stupňů celsia s přesností půl stupně celsia. Frekvence snímání je 0,5 Hz, což znamená nové data každé dvě sekundy. Rozměry senzoru jsou 25 mm x 15 mm x 8 mm. Senzor slouží především k monitoringu tepla a vlhkosti a v případě potřeby by bylo možné na datech získaných ze senzoru ovládat topení v místnosti, případně klimatizaci a větrání okny. [88]

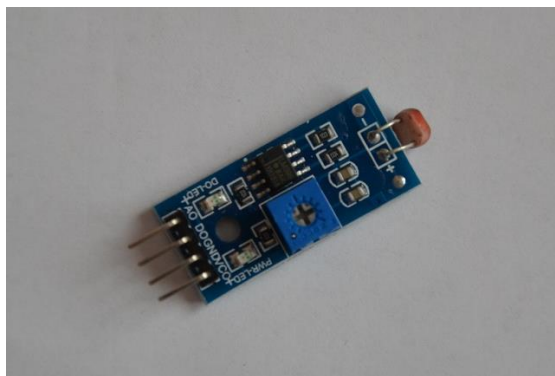


Obrázek 26: Senzor teploty a vlhkosti vzduchu, přední a zadní strana

### 6.2.2 Senzor intenzity světla

Na Obrázek 27 je vidět světelný senzor použitý v prototypu. Je velmi spolehlivý a lze jednoduše zapojit. Má nastavitelnou citlivost a disponuje analogovým i digitálním výstupem. Digitální výstup udává pouze dvě hodnoty – jedna reprezentující světlo, druhá tmu. Analogový výstup udává intenzitu světla (tmy) a to plynulou změnou napětí v rozsahu 0-5 V. Oba výstupy fungují současně. Senzor má 4 piny, a to 5 V napájení, zem, analogový a digitální výstup. Velikost celé desky je 38 x 16 mm. [89]

Za pomoci tohoto snímače lze ovládat osvětlení v místnosti dle světla venku. S nadcházejícím stmíváním je možné rozsvítit, a díky analogovému výstupu lze upravovat i intenzitu osvětlení uvnitř. Například plné světlo na začátku stmívání a tlumené v noci. Senzor intenzity světla poslouží i pro ovládání stínění a žaluzií.



**Obrázek 27: senzor intenzity osvětlení**

### **6.2.3 Senzor pohybu**

K detekci pohybu je využít PIR senzor (na Obrázek 28), neboli pasivní infračervený senzor. Senzor funguje na základě pyroelektrického jevu, tedy na změně teploty v okolí, které je na snímač promítáno jako infračervený obraz. Díky snímání infračerveného obrazu dokáže senzor fungovat i ve špatných světelných podmínkách nebo tmě.

Senzor má nastavitelnou citlivost a frekvenci snímání. Dokáže snímat na vzdálenost až 10 metrů v úhlu 120 stupňů, což stačí na pokrytí celé místnosti při dobře umístěném senzoru. Výstupem ze senzoru je buď logická 0 nebo 1, pokud je detekován pohyb. [90]

Data ze senzoru mohou být využita nejen k detekci nedovoleného vniknutí do objektu, ale také ke sledování obyvatel domu. Díky tomu je možné v noci rozsvítit tlumené světlo na cestu, nebo třeba změnit výstup hudby a nechat ji přehrávat pouze v místnosti, kde se uživatel pohybuje.



**Obrázek 28: PIR senzor - pasivní infračervený senzor pohybu. Zdroj [90]**

#### **6.2.4 Chytrá žárovka LIFX**

Chytrá žárovka LIFX byla představena již v kapitole 2.1.1 Osvětlení. Jde o žárovku komunikující skrze WiFi s možností ovládní přes telefon, případně zasláním vlastní naprogramované zprávy. Lze ovládat intenzitu i barvu osvětlení, nastavovat barevné scény, reagovat na určité události či nastavit časovač. V prototypu funguje jako aktor a je ovládána především na základě dat z ostatních senzorů.

## 7 Návrh systému řízení inteligentního domu

V rámci této práce byl navržen vlastní systém pro řízení chodu inteligentní domácnosti. V této kapitole je shrnuto vzniklé řešení z pohledu analýzy, návrhu architektury a použitých technologií.

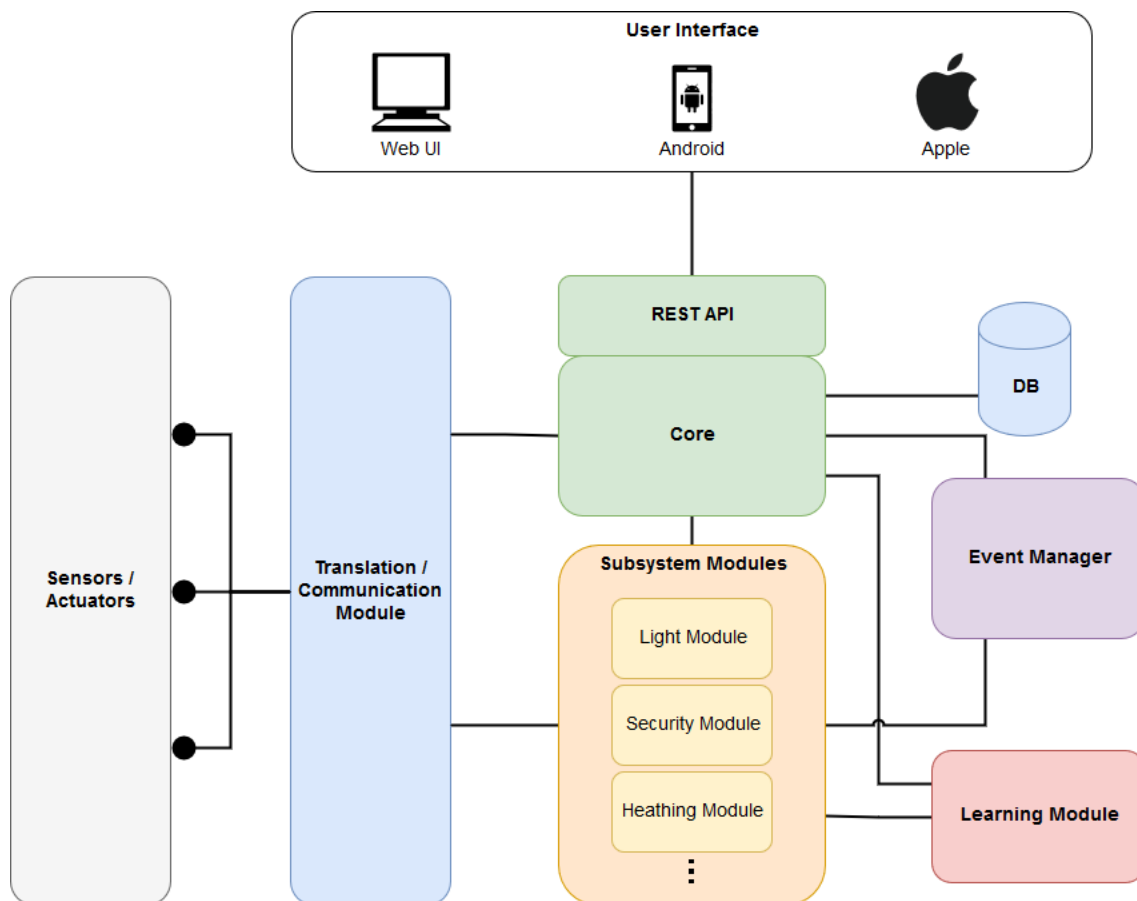
### 7.1 Návrh a architektura systému

Následující podkapitoly popisují návrh a architekturu výsledného systému od celkového pohledu na systém až k dílčím rozhodnutím a případům užití.

#### 7.1.1 Big Picture návrhu systému

Na trhu je k dostání velké množství zařízení, které lze využít k tvorbě systému pro domácí automatizaci. K dosažení požadovaného výsledku může být potřeba kombinovat zařízení od různých výrobců, využívající různé komunikační technologie. Proto byl výsledný systém navržen modulárně, aby bylo možné lehce zaměnit část systému komunikující s hardwarem. Modulární přístup také umožňuje nainstalovat systém pouze s částí funkcionality. Uživatel tak může systém využít například pouze k ovládání světel a vynechat další podsystémy (vytápění, bezpečnost...).

Big Picture, neboli pohled na systém z největší perspektivy je vidět na Obrázek 29. Vyobrazeny jsou všechny důležité moduly systému a jejich vzájemná propojení. Stejnými moduly jsou jádro systému (Core), komunikační modul (Translator / Communicator) a modul pro automatické řízení systému (Event Manager). Ty doplňují moduly subsystému, které lze použít podle požadavků uživatele na systém a učící modul. V tom je možné definovat chování pro strojové učení systému.



Obrázek 29: Big Picture navrženého systému pro domácí automatizaci

**Core** je modul obsahující celé jádro systému. Je v něm definován datový model v podobě POJO tříd a také persistentní vrstva a konfigurace pro komunikaci s databází. Jádro systému je využíváno ve všech ostatních modulech a jsou zde definována i některá rozhraní, která musí ostatní moduly implementovat. Součástí jádra je i REST API, které se využívá pro komunikaci s grafickým rozhraním. Díky němu je možné poskytovat data jednotně do uživatelského rozhraní postaveného na libovolné platformě.

Modul **Event Manager** slouží především pro obsluhu událostí v systému. Jsou v něm definovány služby, které periodicky kontrolují stav prostředí na základě dat získaných ze senzorů. Tyto data kontrolují s uživatelem předdefinovanými pravidly. Pokud jsou pravidla splněna, vytváří se zde a obsluhují události, díky kterým dochází k požadovaným změnám v prostředí.

**Learning modul** je samostatný modul vyhrazený pro strojové učení systému, ať už se bude jednat o neuronové sítě, nebo jiné způsoby úpravy nashromážděných dat za účelem učení a optimalizace.

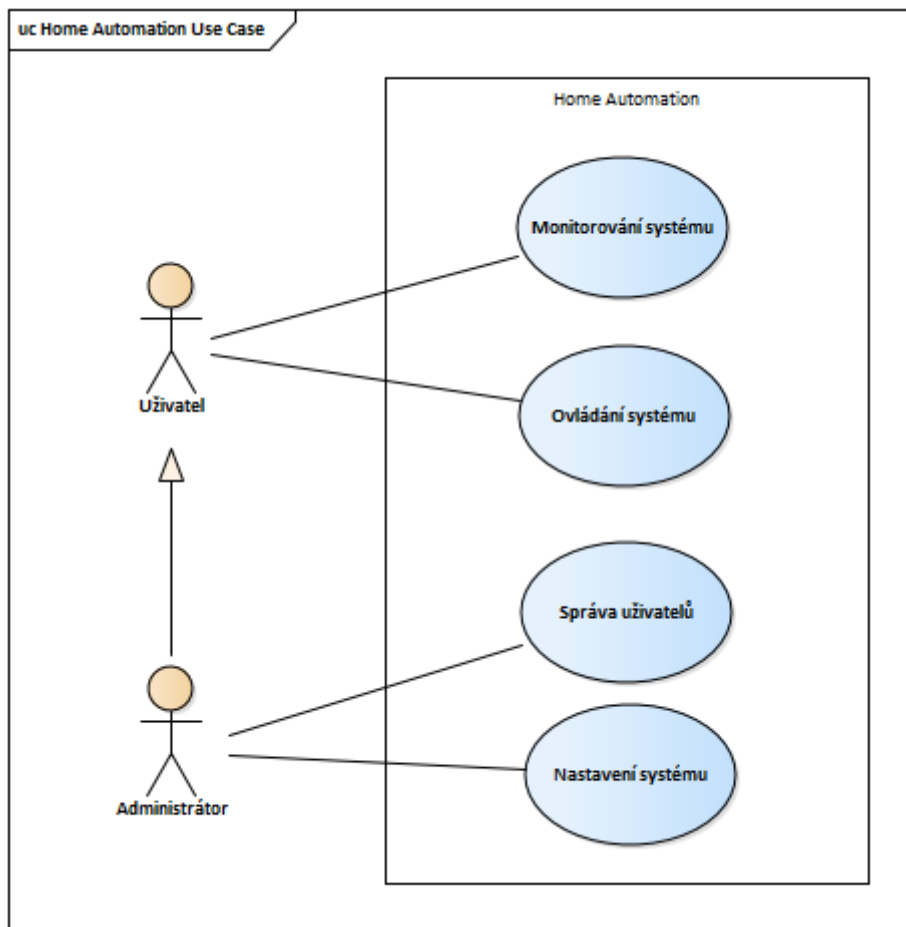
**Subsystem modules** aneb moduly sloužící pro ovládání jednotlivých částí systémů. V těchto modulech je definované veškeré možné chování a ovládání jednotlivých součástí systému a tím tedy i hlavní funkce systému pro ovládání aktuátorů. Kromě těch zmíněných v Obrázek 29 může jít o moduly ovládání multimédií, klimatizace atd.

**Translator / Communicator** modul je jako jediný přímo závislý na použitém hardwaru, a proto je třeba naprogramovat jej přímo pro konkrétní zvolené řešení. Slouží především pro přijímání dat ze senzorů a komunikaci s aktuátory. Shromážděná data přetransformuje do podoby modelu se kterým pracuje jádro systému.

### 7.1.2 Případy užití systému

System pro řízení inteligentního domu by měl ideálně umět ovládat cílové prostředí z větší části autonomně. I přesto ale vznikají na systém určité nároky a požadavky ze strany uživatele. Z těch vychází zjednodušený diagram užití (use case diagram) na Obrázek 30 popisující hlavní funkcionalitu systému.





**Obrázek 30: Zjednodušený Use Case diagram navrženého systému domácí automatizace**

Případy užití závisejí na celkové podobě systému jak z hardwarového hlediska, tak i z množství využitých modulů subsystémů. Jinak by vypadaly, pokud se uživatel rozhodne ovládat například pouze topení a jinak pokud využije možnosti ovládat veškerá zařízení v domě. Proto je uveden zjednodušený diagram případu užití, pokrývající obecné požadavky na systém.

V základu rozlišuje systém mezi dvěma typy uživatelů. Jedním je běžný uživatel a druhým administrátor. Administrátor má rozšířené možnosti o konfiguraci celého systému a správu uživatelů. Tímto oprávněním by měl disponovat alespoň jeden člen domácnosti, aby mohl spravovat ostatní uživatele, případně přidávat a upravovat pravidla chování systému.

Pod případem „*Monitorování systému*“ je zastoupeno veškeré získávání informací o prostředí. Uživatel musí mít možnost zobrazit si aktuální teplotu, vlhkost, stavy

osvětlení a pohybu v jednotlivých místnostech. Stejně tak by měl vidět předdefinované hodnoty a nastavení.

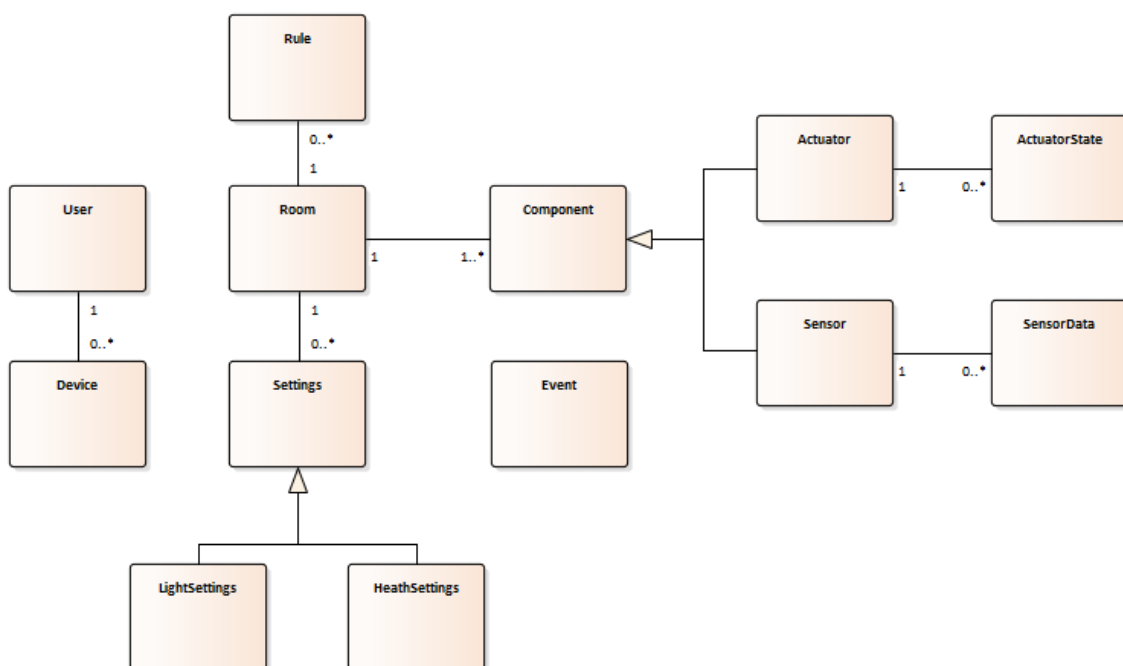
„*Ovládání systému*“ zastává veškeré manuální nastavování aktuátorů v systému. Může se jednat o rozsvěcení a zhasínání světel, nastavování topení, ovládání stínění v místnosti a v podstatě vše, co by jinak mohl udělat hardwarovým ovladačem.

Administrátor má právo také na „*Správu uživatelů*“. Tou je zamýšlená veškerá administrace uživatelů a jejich chytrých zařízení, která se mohou v systému pohybovat. Administrátor tak může přidávat nové členy domácnosti nebo její dočasné hosty a zaregistrovat jejich chytré telefony nebo počítače.

„*Nastavení systému*“ v sobě zahrnuje úvodní konfiguraci systému. Do té spadá vytvoření místností chytré domácnosti a zaregistrování senzorů a aktuátorů. Dále pak konfigurace chování a pravidel pro řízení systému. Mimo úvodní konfigurace sem patří i veškeré pozdější úpravy v nastavení systému.

### **7.1.3 Model jádra systému**

Modul jádra systému obsahuje základní model tříd, se kterým pak zbytek systému nakládá. Model je vyobrazen v class diagramu na Obrázek 31 a reprezentuje základní představu chytré domácnosti. Pro přehlednost jsou uvedeny pouze třídy a vazby mezi nimi bez atributů tříd. V diagramu jsou vynechány i výčtové typy. Ty jsou ale zmíněny v následujících odstavcích u popisů jednotlivých tříd. Detailnější diagram s atributy tříd a relevantními metodami je přiložen na konec práce jako Příloha 1.



Obrázek 31: Class diagram jádra systému

Ústředním bodem diagramu je třída *Room*, na kterou navazují ostatní třídy zastupující hardware, a to *Sensor* a *Actuator*. Třída *Room* slouží pouze pro reprezentaci místnosti v chytré domácnosti pomocí id a jména. Pro každou místnost může být registrováno několik pravidel jejího chování. Tato pravidla reprezentuje třída *Rule*. Obsahuje logický výraz, který umí vyhodnotit. Do toho výrazu jsou doplňovány aktuální data z požadovaných sensorů. Pravidlo také definuje aktuátor a akci kterou má v případě splněné podmínky provést.

Každá místnost může mít také různé nastavení zastoupené třídou *Settings*. Jedná se o uživatelské nastavení například pro intenzitu světla nebo požadovanou teplotu. Proto třída *Settings* slouží jako předek Třídám *LightSettings*, *HeathSettings* atd, které jsou definovány až v modulech konkrétních subsystémů.

Do každé místnosti je možné registrovat sensory a aktuátory. Ty jsou reprezentovány odpovídajícími třídami *Sensor* a *Actuator*. Společným předkem obou je třída *Component*, která uchovává kromě id i referenci na místnost ve které se nachází a popis umístění a jméno zařízení. Třída *Sensor* v sobě definuje i typ sensoru, který je zastoupený výčtovým typem *SensorType*. Obdobně je v třídě *Actuator* zastoupen jeho typ výčtovým type *ActuatorType*. Data ze sensorů jsou reprezentovány třídou

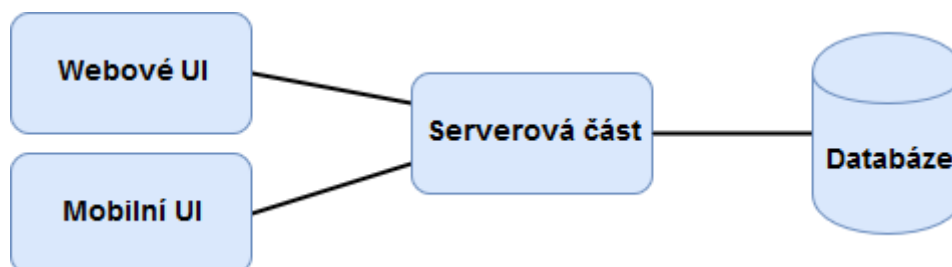
*SensorData*, která uchovává vedle hodnoty také její jednotku z výčtového typu *Unit* a časový otisk, kdy data do systému přišla. Aktuální stav aktuátoru je zaznamenán třídou *ActuatorState*.

Trochu stranou doposud zmíněných tříd stojí *User*, *Device* a *Event*. *User* je třída reprezentující uživatele pomocí jeho jména, příjmení, přezdívky, emailu a hesla. Uživatel má svou roli, která je definována ve výčtovém typu *Role*. *Device* představuje jakékoliv zařízení daného uživatele. Hlavními atributy zařízení jsou jeho jméno a mac adresa.

Třída *Event* slouží k reprezentaci události, která by měla v systému nastat. Obsahuje příkaz k manipulaci s aktuátorem, který získává při vytvoření z pravidla nebo od vyvolání akce uživatelem a zná čas kdy byla vytvořena. O konkrétní události a jejich vyvolání se stará modul Event Manager.

#### 7.1.4 Architektura systému

Z pohledu architektury se u navrženého systému jedná o vícevrstvou architekturu (viz Obrázek 32). Ta vychází především z požadavku na ovládání systému z různých typů zařízení.



Obrázek 32: Vícevrstvá architektura systému

Webové uživatelské rozhraní a mobilní webové rozhraní jsou samostatné projekty, které lze označit za klientskou část systému. Slouží především pro komunikaci uživatele se systémem. Díky nim uživatel může sledovat data získaná ze senzorů v domácnosti a ovládat potřebné aktuátory. Pod mobilním uživatelským rozhraním je možné si představit aplikaci na zařízeních Android i iOS.

Serverová část je hlavní částí celého systému. Jedná se o výslednou aplikaci vzniklou sestavením modulů představených na Big Picture Obrázek 29. Především pak o komunikační modul a event manager, kteří jsou nasazeny na aplikačním serveru a vykonávají svou činnost v závislosti na jádře a ve spolupráci se sensory a aktuátory.

Serverová část komunikuje s databází, do které ukládá svá data. Datovou částí by mohla být jakákoliv databáze, ale v tomto konkrétním případě byla použita databáze PostgreSQL. Oddělení datové části je vhodné především kvůli použití Raspberry Pi jako centrálního prvku. To využívá místo disku SD kartu, na které je uložen operační systém a veškerá data. Ta není vhodným úložištěm pro data nejen kvůli malé velikosti, ale především kvůli její životnosti a množství možných provedení zápisů, kterých bude z dlouhodobého hlediska mnoho. Proto je vhodné datovou část umístit na jiný server, případně na sdílený či multimediální server v domácnosti, pokud jím výsledný systém disponuje. Minimálně je vhodné připojit k Raspberry Pi externí disk a přesměrovat ukládání dat z databáze na něj.

## **7.2 Použité technologie**

V této kapitole jsou představeny technologie použité při implementaci prototypu systému pro řízení inteligentního domu. Popsány jsou technologie pokrývající celý technologický stack od spodní vrstvy, kterou je databáze, přes aplikační část, až k frontendovým technologiím. Pro prototyp byly vybrány open source technologie, které mohou plnohodnotně zastoupit jejich komerční alternativy.

### **7.2.1 PostgreSQL**

PostgreSQL je open source relačně objektový databázový systém. Má za sebou více než 15 let aktivního vývoje a získal si dobrou pověst svou spolehlivostí, správností a integritou dat. PostgreSQL podporuje širokou škálu běžně používaných operačních systémů – Windows, Linux i UNIXové systémy. Je plně ACID kompatibilní, má plnou podporu pro cizí klíče, joinování tabulek, pohledy, spouště a uložené procedury. Podporuje také většinu standardních SQL datových typů, včetně ukládání velkých binárních souborů, jako jsou fotky, videa nebo zvuky.

PostgreSQL také nabízí pokročilé funkce pro práci s databází. Patří mezi ně například řízení přístupu pomocí systému Multi-Version Concurrency Control (MVCC), obnovení ze zálohy do konkrétního bodu v čase (point in time recovery), asynchronní replikace, online zálohování, plánování a optimalizování dotazů, analytické funkce a mnohé další. S touto škálou funkcí se PostgreSQL hodí jako open source náhrada konkurenčních řešení Oracle nebo MSSQL Server. [91]

### 7.2.2 Java

Java je objektově orientovaný programovací jazyk vyvíjený firmou Sun Microsystems. Je jedním z nejpoužívanějších programovacích jazyků na světě. Je navržen, aby byl dostatečně jednoduchý a pro programátory bylo jednoduché se ho naučit. Býraz Java můžeme chápat taky ve významu platforma neboli prostředí, na kterém mohou Java aplikace běžet.

Java je interpretovaný jazyk, což mu zajišťuje vysokou přenositelnost a platformní nezávislost. Pro běh aplikací založených na Javě stačí mít na cílovém zařízení dostupný interpret Javy – tzv. Java Virtual Machine. Java je dostupná v několika edicích, a proto je vhodná pro širokou škálu užití. Od čipových karet, přes mobilní a desktopové aplikace, až po rozsáhlé distribuované systémy. Mezi příklady reálného použití patří třeba sociální síť Twitter, internetová televize Netflix, nebo Rakouský zdravotní systém. [92][93]

### 7.2.3 Spring

Spring Framework je aplikační rámec nabízející komplexní soubor funkcí pro vývoj moderní enterprise aplikací založených na platformě Java. Spring umožňuje vytvářet pokročilé Java aplikace na základě POJO objektů (klasických Java tříd). Klíčovým prvkem Springu je poskytnout vývojáři podporu na úrovni infrastruktury, aby se vývojář mohl plně věnovat vývoji business logiky vytvářené aplikace.

Spring je modulární framework, a tak si vývojář může zvolit které z nabízených funkcí využije. Základní funkcí, kterou Spring poskytuje, je vkládání závislostí (*Dependency Injection*). Mezi další funkce poskytované Springem nebo jeho moduly jsou například Spring MVC a Spring Web pro tvorbu webových aplikací s využitím návrhového vzoru model-view controller, modul pro podporu aspektově orientovaného

programování, přímá podpora JDBC, ORM, JPA, JMS a dalších nástrojů pro práci s daty, podpora testování nebo třeba Spring Security pro řízení přístupu a zabezpečení aplikace. [94]

#### **7.2.4 Maven**

Maven slouží k sestavování aplikací založených na platformě Java a poskytuje jednotný systém sestavování aplikací. Dále slouží ke správě závislostí potřebných pro chod dané aplikace, ať už se jedná o externí knihovny nebo jiné moduly jedné aplikace. Činí tak pomocí *Project Object Modelu (POM)*. To je v podstatě jeden popisný soubor, shrnující všechny potřebné informace o dané aplikaci. [95]

Tento nástroj je velmi vhodný k tvorbě modulárních aplikací založených na javě a velmi zjednodušuje jejich provázání a sestavení. To jsou důvody, proč byl zvolen k implementaci prototypu v praktické části této práce.

#### **7.2.5 Hibernate**

Hibernate je nástroj sloužící k relačně-objektovému mapování (ORM) v aplikacích. ORM zajišťuje převod objektů napsaných v programovacím jazyce do struktury tabulek relační databáze.

Hibernate vedle svého nativního API nabízí i implementaci JPA (Java Persistence API) a proto je možné ho jednoduše použít v prostředích podporujících Java SE, Java EE, OSGi kontejnery atd. Dalšími výhodami Hibernatu jsou škálovatelnost pro použití v clusteru, nebo optimalizace pro vysoký výkon aplikací (např. různé strategie zamykání tabulek, lazy inicializace, vytváření SQL dotazů již při spuštění, a ne za běhu). [96]

#### **7.2.6 Webové služby**

W3C definuje webovou službu jako systém navržený pro vzájemnou komunikaci dvou počítačů přes síť. Webová služba má rozhraní předepsané ve strojově zpracovatelné podobě (konkrétně WSDL – Web Service Description Language). Ostatní systémy pak s webovou službou komunikují tímto předepsaným způsobem pomocí zpráv přes http protokol. Momentálně máme dva základní způsoby, jak přistupovat

k vytváření webových služeb. Starším z nich je Simple Object Access Protocol neboli SOAP a novějším pak Representational State Transfer (REST). [97]

### **Simple Object Access Protocol (SOAP)**

SOAP je standardem pro webové služby původně vyvinutý firmou Microsoft jako náhrada za starší, hůře fungující řešení. Ty byly založeny na binárním zasílání zpráv, zatímco SOAP k tomu využívá výhradně zpráv ve formátu XML. SOAP také podporuje různá rozšíření (například pro bezpečnost nebo transakce) a uživatel si tak může vybrat, která rozšíření použije.

Výsledné XML pro komunikaci může být ovšem velmi složité a v některých programovacích jazycích je potřeba jej sestavovat ručně. To může být problém, protože SOAP netoleruje chyby. Naopak v jiných jazycích se s tímto problémem nikdy nesečkáte. Využívají totiž takzvaného WSDL (Web Services Description Language), což je další dokument propojený se SOAP. Obsahuje popis fungování konkrétní webové služby. Pomocí WSDL lze v některých vývojových prostředích vygenerovat vše potřebné pro používání webových služeb automaticky.

Jednou z užitečných vlastností SOAPu je vestavěná práce s chybami. Pokud je v požadavku cokoliv špatně, odpověď bude obsahovat popis chyb, který je možné použít k jejímu vyřešení. Takto popsaná chyba obsahuje i standardizovaný chybový kód, takže je možné zautomatizovat práci s chybovými odpověďmi v cílové aplikaci. [97]

### **Representational State Transfer (REST)**

REST je naproti tomu lehčí řešení webových služeb. Jedná se o architekturu definující jednotné rozhraní pro přístup ke zdrojům (data, stavy aplikace) pomocí URI (Uniform Resource Identifier – jednotný identifikátor zdroje) a základních HTTP volání. Těmi jsou metody PUT, GET, POST a DELETE. Prakticky jde o odkazy v internetové síti.

Architektura REST je navržena tak, aby výsledné řešení dosahovalo vysokého výkonu (bylo rychlé), bylo dobře škálovatelné a modifikovatelné. REST také není omezený pouze na posílání zpráv ve formátu XML, ale je možné použít i formáty CSV,



JSON nebo RSS. Práce s těmito formáty může být v některých programovacích jazycích jednodušší než s XML. Také se jedná o formáty méně upovídané než XML, tudíž jsou výsledné zprávy jsou menší a šetří se zdroje i přenosové pásmo. [97]

Pro implementaci prototypu jsou použity webové služby typu REST, především pro svou jednoduchost a rychlost a možnost pracovat s jiným formátem než XML. Uživatel chytrého domu očekává co nejrychlejší reakci na své povely a nastalé akce, takže je vhodné lehčí a rychlé řešení.

### **7.2.7 Aplikační servery**

Aplikační server tvoří vrstvu mezi operačním systémem a aplikacemi a vytváří tak prostředí, ve kterém aplikace mohou běžet. Toto prostředí obsahuje knihovny implementující veškerá API obsažená v Java EE a poskytují tyto funkce aplikacím v něm běžícím. Důležitou součástí aplikačního serveru je HTTP listener, který zajišťuje možnost komunikace. Příkladem funkcí zajišťovaných Java EE kontejnerem pak může být persistence objektů do databáze, transakční zpracování požadavků nebo zasílání zpráv mezi aplikacemi. Díky tomu je ro vývojáře snazší soustředit se na vývoj konkrétní aplikace. [98]

Další užitečné funkce poskytované aplikačním serverem pak mohou být administrátorská konzole, logování událostí nebo funkce na podporu kvality služeb (QoS – Quality of Service) a spolehlivosti (HA – High Availability). Mezi funkce zajišťující kvalitu služeb patří například ochrana proti DDoS útokům nebo management přenosového pásma. High Availability je pak zajišťováno funkcemi jako je loadbalancing a podpora clusteringu.

Pro aplikace na platformě Java existuje celá řada aplikačních serverů. Některé jsou plnohodnotnými aplikačními servery, jiné jsou označovány spíše jen jako webové kontejnery. Poskytují totiž pouze funkce pro webovou komunikaci (implementují práci s Java servlety), ale již některými funkcemi definovanými v Java EE. Mezi nejznámější patří Glassfish od Oraclu, Apache Geronimo, Jetty a Virgo od Eclipse Foundation, WildFly (původně JBoss od RedHatu) a Apache Tomcat. Poslední dva zmiňované jsou použity i při implementaci prototypu navrženého v této práci.

## **Tomcat**

Tomcat je open source webový kontejner od vyvíjený organizací Apache pod licenci Apache license version 2. Jedná se o lightweight webový server, který je běžně používán v produkčním prostředí robustních průmyslových aplikací a je dobře zdokumentován. Implementuje Java Servlet, Java Server Pages, Java Expression Language a Java Web Socket a poskytuje aplikacím i funkce HTTP serveru. [99]

## **WildFly**

WildFly je flexibilní, lightweight aplikační server, který vznikl jako nástupce aplikačního serveru JBoss. WildFly, stejně jako předním JBoss, vyvíjí firma RedHat. Ta stojí za stejnojmennou distribucí operačního systému linux a nabízí i další produkty a služby v oblasti IT. Například nástroje pro cloud computing, virtualizaci, management nebo uložení dat. [100]

WildFly implementuje technologický stack Java EE7 a je postaven na nejlepších standalone open source projektech, které vývojáři běžně používají samostatně. Příklady jsou Hibernate, Infinispan, HornetQ a další. Také podporuje standardy pro vývoj moderních webových aplikací (jako je REST) a zajištění vysoké dostupnosti.

Jelikož se WildFly snaží být co nejvíce lightweight řešením, je jeho architektura postavena na odnímatelných subsystémech. Vývojář si tak může běhové prostředí serveru přizpůsobit svým potřebám pouhým odstraněním/přidáním subsystému z/do konfigurace serveru. Tím se může výrazně snížit místo na disku vyžadované serverem. Dále má WildFly implementovanou vlastní agresivní správu paměti k co největšímu snížení alokované paměti. Díky tomu je možné provozovat WildFly server s defaultním nastavením parametrů JVM a může běžet i na méně výkonných zařízeních. [101]

### **7.2.8 Frontendové technologie**

Pro vývoj frontendu byla zvolena technologie JSP (Java Server Pages). Ta umožňuje jednoduše vytvářet dynamické webové stránky pomocí šablon s vloženým java kódem nebo speciálními značkami zvanými tagliby. Tuto technologii doplňuje HTML a CSS. Pro práci s CSS byl použit framework Bootstrap. Ten je vhodný k rychlému stylování pomocí předpřipravených CSS stylů.

## 8 Implementace

V této kapitole jsou popsány zajímavé části z implementace některých částí prototypu navrženého systému domácí automatizace. Jsou zde blíže popsány některé technologie, frameworky a jejich využití při implementaci systému.

### 8.1 Základní konfigurace projektu

Aplikace je navržena pro běh na aplikačním serveru. Pro běh hlavní části je navrženo použití aplikačního serveru WildFly, který je nástupcem serveru RedHat Jboss. Pro běh webového rozhraní je pak dostačující webový server Apache Tomcat.

System je navržen modulárně a ke konfiguraci modulů a závislostí mezi nimi je využit nástroj Maven. Ten umí spravovat veškeré závislosti projektů, což kromě závislostí mezi moduly, znamená i správu knihoven třetích stran a frameworků potřebných k fungování výsledného systému.

V některých modulech je využito logování realizované pomocí nástrojů Log4j a SLF4J. Nastavení logování je možné pomocí samostatného properties souboru.

### 8.2 Datová část

Pro ukládání dat byl zvolen databázový systém PostgreSQL, který je opensource a umožňuje jednoduché použití. Byl zvolen i kvůli předchozím zkušenostem autora. V práci je použit objektově relační framework Hibernate, díky kterému je možné jednoduše zaměnit databázový systém pouze s minimální úpravou konfigurace. Díky tomu by mělo být možné využívat se systémem libovolnou databázi.

Hlavní část systému běží v aplikačním serveru Wildfly 9. Ten potřebuje mít pro komunikaci s databází PostgreSQL nastavený vlastní datový zdroj (data source). Toto nastavení se provádí v konfiguračním souboru serveru standalone-full.xml. Datový zdroj definuje url databáze a přihlašovací údaje uživatele. V konfiguraci lze také nastavit využívání transakcí. Důležitou součástí je i definování jména pro datový zdroj, podle kterého si tento zdroj najde hibernate a lze na něj odkazovat v konfiguračním souboru hibernatu persistence.xml. Konfigurace vypadá následovně:

```
<datasource jta="true" jndi-name="java:/PostGreDS" pool-name="Postgre-Pool" enabled="true">
```

```

        <connection-url>jdbc:postgresql://localhost/home-
automation</connection-url>
        <driver>postgresql</driver>
        <security>
            <user-name>postgres</user-name>
            <password>password</password>
        </security>
    </datasource>

```

### 8.3 Core modul – jádro systému

Jádro systému je postaveno na frameworku Spring, který zajišťuje provázání jednotlivých komponent a základní konfiguraci projektu. Automatické provázání komponent funguje díky označení POJO tříd anotací @Entity (případně @Component atd). Automatické vyhledávání takových tříd je nastaveno v souboru applicationContext-core.xml, který je hlavním konfiguračním souborem pro Spring v tomto modulu.

Core modul obsahuje i persistenční vrstvu využívající ORM framework Hibernate a JPA Repositories. Hibernate umožňuje automaticky vytvářet tabulky a jejich sloupce na základě správně oannotovaných POJO tříd. JPA repozitáře umožňují definovat rozhraní s metodami pro komunikaci s databází. Na základě jmenné konvence metod definovaných v JPA rozhraních jsou automaticky vytvořeny dotazy do databáze. Příkladem takového rozhraní a metody je:

```

@Repository
public interface SensorDataDao extends JpaRepository<SensorData,
Long>, JpaRepositoryExecutor<SensorData> {

    SensorData findFirstBySensorOrderByTimestampDesc(Sensor sensor);
}

```

Tento interface umožňuje komunikaci s tabulkou obsahující data ze sensorů. Definovaná metoda z databáze získá poslední uložená data pro konkrétní sensor, předaný jako parametr.

### 8.4 Komunikační modul a komunikace se sensory a aktory

Komunikační modul zajišťuje veškerou komunikaci s hardwarem v inteligentní domácnosti. Pro ovládání sensorů v prototypu jsou použity python skripty, které čtou

data ze sensorů skrze GPIO piny Raspberry Pi. Data pak posílají pomocí http requestu na REST API komunikačního modulu. REST je použit i pro komunikaci s chytrou žárovkou Lix, které zasílá potřebné příkazy.

### 8.4.1 Python skripty pro ovládání sensorů

Pro získávání dat ze sensorů připojených k GPIO pinům Raspberry Pi jsou využity skripty napsané v jazyce Python. Následuje příklad pro čtení dat z pohybového sensoru:

```
import RPi.GPIO as GPIO
import time
import requests
import json
import sys

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO_PIR = 21
GPIO.setup(GPIO_PIR, GPIO.IN)

sensorId = 'movementPIRSensor_1'

controllerMapping = 'movementSensorDataCollect'
serverUrl = sys.argv[1]
url = serverUrl + controllerMapping
headers = {'content-type': 'application/json'}
values = []

def sendRequest( values ):
    payload = {'sensorId' : sensorId, 'movementData' : values}
    requests.post(url, data=json.dumps(payload), headers=headers)

print 'Movement data collecting started'
while True:
    values.append(GPIO.input(GPIO_PIR))
    time.sleep(2)

    if len(values) == 30:
        sendRequest( values )
        values = []
```

Na začátku příkladu je možné vidět importování důležitých knihoven pro chod skriptu. Pod ním následuje nastavení GPIO pinů Raspberry Pi pro čtení. Dále už jsou atributy pro sestavení a odeslání requestu do podoby jsonu. Skript běží v cyklu a sbírá data z pohybového sensoru každé dvě sekundy. Po jedné minutě odesílá data pomocí metody sendRequest, která sestaví request a naplní ho daty ve formátu json. Tento request odešle metodou post. Adresa, na kterou se má request odeslat je zís-

kána z atributů `serverUrl` a `controllerMapping`. `ServerUrl` je hodnota získána ze startovacího skriptu, který ho předá jako atribut do python skriptu. `ControllerMapping` je adresa kontroleru který umí data z tohoto sensoru zpracovat.

Obdobně fungují všechny python skripty sbírající data ze senzorů. Konkrétně ze sensoru intenzity světla a teplotního čidla s vlhkoměrem. Pro spuštění všech potřebných python skriptů byl vytvořen jeden spouštěcí bash skript.

## 8.4.2 Rest API

Pro příjem zpráv z python skriptů obsluhujících senzory je využito REST API. Pro každý druh sensoru je vystaven kontrolér mapující přijatá data do podoby modelu jádra systému. Pro implementaci REST API je využit modul Springu – Spring Web. Kontroléry vypadají podobně jako ty vytvořené ve Spring MVC, akorát je použita anotace `@RestController`, kterou je třída uvozena.

```
@RequestMapping(value = "/movementSensorDataCollect", method = RequestMethod.POST, headers = {"Content-type=application/json"}, consumes = MediaType.APPLICATION_JSON_VALUE)
public @ResponseBody
void detectMovement (@RequestBody MovementSensorDataCollector movementData) {
    .
    .
    .
}
```

Uvedená ukázka znázorňuje použití anotací pro konfiguraci metod v implementovaném REST API. Tělo metody je vynecháno pro větší přehlednost ukázky. Anotace `@RequestMapping` slouží k propojení URL adresy http požadavku na požadovanou metodu kontroléru. V této anotaci je kromě URL adresy definována i http metoda kterou jsou data poslána a jejich formát. Pro jednu hodnotu namapované URL adresy mohou být použity různé http metody a Spring je zvládne automaticky namapovat. Data do kontroléru přijdou ve formátu JSON a pokud mají správnou strukturu, jsou namapována na java třídu. Ta je uvedena jako parametr metody a uvozena anotací `@RequestBody`. Stejným způsobem jsou definovány i kontroléry v REST API jádra systému.

### 8.4.3 LIFX http API

Žárovky Lifx umožňují vývojářům dva základní způsoby ovládání. Jedním je skládání vlastních paketů a druhým komunikace přes vlastní http API. Každý uživatel si po koupi žárovky Lifx zaregistruje svůj účet, pod kterým může své žárovky spravovat a ovládat je přes internet z oficiální aplikace. Obdobně je možné žárovky Lifx ovládat z vlastní aplikace.

V prototypu byla zvolena možnost ovládat žárovku pomocí http API, jelikož se jedná o rychlejší řešení a způsob komunikace je stejný jako u programování vlastního REST API pro komunikaci se senzory.

Pro zprovoznění komunikace přes http API je potřeba udělat několik kroků. Prvním z nich je získání tokenu pro autorizaci a autentikaci uživatele na webu výrobce. S ním už je možné sestavit data ve formátu JSON a odeslat data pomocí http requestu. Data pro nastavení žárovky mohou vypadat následovně:

```
{
  "power": "on",
  "color": "blue saturation:0.5",
  "brightness": 0.5,
  "duration": 5,
}
```

Předchozí ukázka slouží k zapnutí žárovky se světle modrou barvou a 50% intenzitou po dobu pěti vteřin. Request musí kromě dat a tokenu znát URL adresu na kterou bude odeslán a použitou http metodu. Pro obecné nastavení žárovek slouží PUT metoda `setState`, která umožňuje nastavit jakýkoliv parametr žárovky (nebo více žárovek dle specifikovaného selektoru).

Pro vytvoření a odeslání requestu je v aplikaci použita knihovna Jersey. Následuje ukázka jejího použití:

```
public Response sendPutJsonRequest(String url, String jsonPayload) {
    Client client = ClientBuilder.newClient();

    WebTarget resource = client.target(url);

    Invocation.Builder request = resource.request();
    request.accept(MediaType.APPLICATION_JSON);
    request.header("Authorization", "Bearer " + AUTH_TOKEN);
}
```

```
    return request.put(Entity.json(jsonPayload));
}
```

Vstupem do metody jsou pouze cílová URL a data nastavení v JSON formátu. Prvním krokem je vytvoření instance třídy Client. Tu je možné předem nakonfigurovat o požadované filtry. V ukázce je z této instance vytvořena instance třídy WebTarget, která už v sobě uchovává cílové URL. Nakonec je z WebTargetu sestaven výsledný request, kterému je nastaven formát na JSON a do hlavičky uložen autorizační token. Poté už je možné request odeslat požadovanou metodou s přiloženými daty. Po odeslání requestu je navržena odpověď v podobě třídy Response, obsahující status zaslání requestu.

## 8.5 Event Manager modul

Event Manager slouží v systému k periodickému vyhodnocování přednastavených podmínek a vykonávání úkonů které z nich vyplynou. K práci s logickými výrazy byla použita knihovna JEXL. Pro periodické vyvolávání metod slouží knihovna QuartzJobs. Obojí je popsáno v následujících kapitolách.

### 8.5.1 Quartz Jobs

Nástroj Quartz Jobs slouží k definování takzvaných Jobů (prací, služeb) v systému, které lze vyvolávat na základě předdefinované spouště. Nastavení těchto jobů je odděleno v samostatném konfiguračním xml souboru eventmanager-quartz-context.xml. Ten vypadá následovně:

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">

    <bean name="ruleCheckerJobDetail" class="org.springframework.scheduling.quartz.JobDetailFactoryBean">
        <property name="jobClass" value="cz.kiele.homeautomation.eventmanager.quartz.RuleCheckerJob" />
        <property name="durability" value="true" />
    </bean>

    <bean name="eventExecutionerJobDetail" class="org.springframework.scheduling.quartz.JobDetailFactoryBean">
        <property name="jobClass" value="cz.kiele.homeautomation.eventmanager.quartz.EventExecutionerJob" />
        <property name="durability" value="true" />
    </bean>
```



```

    <bean id="ruleManagerCronTrigger" class="org.springframework.scheduling.quartz.CronTriggerFactoryBean">
        <property name="jobDetail" ref="ruleCheckerJobDetail" />
        <property name="cronExpression" value="0 */1 * ? * *" />
    </bean>

    <bean id="eventExecutionerCronTrigger" class="org.springframework.scheduling.quartz.CronTriggerFactoryBean">
        <property name="jobDetail" ref="eventExecutionerJobDetail" />
        <property name="cronExpression" value="*/1 * * ? * *" />
    </bean>

    <!-- Scheduler factory bean to glue together jobDetails and triggers to Configure Quartz Scheduler -->
    <bean class="org.springframework.scheduling.quartz.SchedulerFactoryBean">
        <property name="jobFactory">
            <bean class="cz.kiele.homeautomation.eventmanager.quartz.AutowiringSpringBeanJobFactory" />
        </property>
        <property name="jobDetails">
            <list>
                <ref bean="ruleCheckerJobDetail" />
                <ref bean="eventExecutionerJobDetail" />
            </list>
        </property>

        <property name="triggers">
            <list>
                <ref bean="ruleManagerCronTrigger" />
                <ref bean="eventExecutionerCronTrigger" />
            </list>
        </property>
    </bean>
</beans>

```

Všechny potřebné komponenty jsou definovány pomocí klasických bean, stejně jako ve standartním spring contextu. V systému jsou definovány dva joby. *RuleCheckerJob* pro kontrolu pravidel definovaných v pro místnosti a *EventExecutionerJob* pro vyvolání eventu a změny v systému. Obě třídy musí být potomkem třídy *QuartzJobBean* a implementovat třídu *executeInternal*. Toto nastavení zajistí, že Quartz může vytvořit instanci jobu a automaticky na něm spouštět metodu *executeInternal*.

Pro konfiguraci jobů jsou zapotřebí 3 kroky:

1. **Definice detailu.** Vytvoření beany s identifikátorem pro pozdější volání reference. Quartz poskytuje třídu *JobDetailFactoryBean*, která vytvoří požadovanou instanci detailu. Hlavním parametrem předaným do instance *jobClass*, neboli třída, kterou bude job využívat.

2. **Definice spouště.** Spoušť neboli trigger je přes parametr svázána s detailem jobu a definuje pravidlo pro automatické spuštění. V tomto případě se jedná o cron spoušť. Ta umožňuje definovat cron výraz stejně jako unixových operačních systémech. Oba joby se spouštějí každou vteřinu.
3. **Vytvoření scheduleru.** Scheduler je plánovačem, který prováže detaily s jejich spouštěči a naplňuje jejich spuštění dle definice.

Scheduler používá k přípravě jobů *SpringBeanJobFactory*. Pokud ale má job závislost na jiných komponentách, tato továrna neumí využít injektování závislostí springu a automaticky vložit závislost pomocí anotace. Závislost jde vložit v konfiguračním xml souboru, ale pro více závislostí, nebo složitější hierarchii závislostí je to velmi nepraktické. Proto byla scheduleru poskytnuta vlastní implementace JobFactory nazvaná *AutowiringSpringBeanJobFactory*, která tento nedostatek řeší a je možné v jobech využívat anotaci `@Autowired`. Scheduler pak zvládne s pomocí springu joby automaticky sestavit.

```
public class AutowiringSpringBeanJobFactory extends SpringBean-
JobFactory implements ApplicationContextAware {

    private transient AutowireCapableBeanFactory beanFactory;

    @Override
    public void setApplicationContext(final ApplicationContext con-
text) {
        beanFactory = context.getAutowireCapableBeanFactory();
    }

    @Override
    protected Object createJobInstance(final TriggerFiredBundle bun-
dle) throws Exception {
        final Object job = super.createJobInstance(bundle);
        beanFactory.autowireBean(job);
        return job;
    }
}
```

## 8.5.2 JEXL

JEXL (Java Expression Language) je nástroj pro vyhodnocování logických a matematických výrazů. Výraz je do nástroje předán v textové podobě a následně jsou namapovány parametry výrazu s reálnou hodnotou. JEXL je v prototypu použit k vyhodnocování logických výrazů v pravidlech (třída *Rule*) definovaných pro místnosti chytrého domu.

Výrazy v pravidlech jsou v textové podobě. Konkrétně je v nich vždy obsaženo id sensoru, operátor a hodnota proti které se kontroluje aktuální hodnota ze sensoru. JEXL potřebuje k vyhodnocení namapovat parametry ve výrazu a k tomu slouží vlastní třída *RuleExpressionMapper*, která umí namapovat aktuální hodnotu na id sensoru.

```
private static final JexlEngine jexl = new JexlBuilder().cache(512).strict(true).silent(false).create();

public void evaluateRule(Rule rule) {
    JexlExpression expression = jexl.createExpression( rule.getExpression() );
    SensorData data = sensorDataDao.findFirstBySensorOrderByTimestampDesc(rule.getSensor());

    JexlContext context = new MapContext();

    context.set(rule.getSensor().getId(), data.getValue());

    Boolean result = (Boolean) expression.evaluate(context);
}
```

Pro vyhodnocení výrazu je zapotřebí nejdříve vytvořit *JexlEngine*, který umí vytvořit instanci třídy *JexlExpression* z předaného string výrazu. Další důležitou součástí je *JexlContext*, který v sobě uchovává mapu propojující parametr s reálnou hodnotou. Jakmile je kontext připravený, je možné ho předat rovnou instancí třídy *JexlExpression* a pomocí metody *evaluate* nechat výraz vyhodnotit.

### 8.5.3 Java reflexe

Pokud je některé z pravidel vyhodnoceno kladně, vzniká v systému událost, která má za úkol ovládání aktuátorů. Tato událost nemusí vzniknout pouze po kontrole pravidel, ale i pokud uživatel zadá příkaz k ovládání ručně. Příkazy jsou ve třídě *Event* a jsou uloženy jako název metody která se má vyvolat. Aby bylo toto možné je využita reflexe.

Reflexe v Javě umožňuje přistupovat k atributům a metodám tříd přímo za běhu aplikace.

```
Class cls = controller.getClass();
Method method = cls.getDeclaredMethod(command, params);
method.invoke(controller, actuator);
```

V předchozím příkladu je na prvním řádku získána instance objektu Class z třídy, na které chceme metodu vyvolat. Zde je použit controller, což je instance třídy pro ovládní daného typu aktuátoru, vytvořená dříve v kódu. Je sem předána instance jednoho z kontrolérů, které implementují metody pro veškeré ovládní aktuátorů. Na druhém řádku je v této třídě nalezena metoda dle jména. Jméno metody je získáno z třídy Event. Dále jsou předané typy všech parametrů, se kterými chceme metodu zavolat. Nakonec se zavolá požadovaná metoda voláním metody invoke, které jsou předány instance, na které se má metoda zavolat a hodnotami parametrů (v tomto případě aktuátor).

#### 8.5.4 Emailové notifikace

Bezpečnostní modul umí odesílat emailové notifikace a neočekávaném pohybu v objektu. K práci s emailem jsou použity možnosti nabízené Springem i Javou. Konkrétně se jedná o Java Mail Api a knihovnu Spring Context Support. K odeslání emailu je nejprve potřeba nastavení třídy MailSender. Ta obsahuje veškeré nastavení potřebné pro komunikaci s vybraným smtp serverem. Hodnoty se do nastavení načítají ze souboru mailServer.properties, takže je možné je snadno změnit.

```
<bean id="mailSender" class="org.springframework.mail.javamail.JavaMailSenderImpl">
  <property name="host" value="${spring.mail.host}" />
  <property name="port" value="${spring.mail.port}" />
  <property name="username" value="${spring.mail.username}" />
  <property name="password" value="${spring.mail.password}" />

  <property name="javaMailProperties">
    <props>
      <prop key="mail.smtp.auth">true</prop>
      <prop key="mail.smtp.starttls.enable">true</prop>
    </props>
  </property>
</bean>

<bean id="emailNotifier" class="cz.kiele.homeautomation.securitymodule.notifier.EmailNotifier">
  <property name="mailSender" ref="mailSender" />
</bean>
```

Po nastavení MailSenderu je možné tento objekt vložit do vlastní třídy, která obsahuje metodu pro sestavení a odeslání jednoduché emailové zprávy:

```
public void sendMail(String from, String to, String subject, String msg) {
```

```
SimpleMailMessage message = new SimpleMailMessage();

message.setFrom(from);
message.setTo(to);
message.setSubject(subject);
message.setText(msg);
mailSender.send(message);
}
```

## 8.6 Zabezpečení

Pro systém je důležité mít určitou úroveň zabezpečení. Aplikaci může využívat pouze registrovaný uživatel. K reprezentaci uživatele v systému slouží modelové třídy User s definovanou rolí z výčtového typu Role. Uživatel se přihlašuje pomocí své emailové adresy a hesla. Heslo je v databázi uloženo v hashované podobě pomocí algoritmu SHA256.

K zabezpečení je v systému využit framework Spring Security, díky kterému je možné jednoduše autentizovat a autorizovat uživatele. Konfigurace pro tento framework je oddělena ve vlastním xml souboru.

Zabezpečení je v systému řešeno na dvou úrovních, a to přímo v jádru systému a vlastní způsob zabezpečení musí řešit i webová případně mobilní aplikace. Díky Spring Security je možné zabezpečit REST API, které odesílá data do uživatelského rozhraní. Poslat požadavek na REST kontrolér pak může pouze autentizovaný, případně autorizovaný uživatel.

Stejným způsobem jsou zabezpečeny stránky webového rozhraní už na úrovni URL adresy:

- Úvodní stránka mapovaná na adresu /index je dostupná všem a umožňuje přihlášení do aplikace
- Veškeré stránky uvozené adresou /admin/\* jsou dostupné pouze pro uživatele s rolí a oprávněním administrátora
- Ostatní stránky jsou dostupné všem přihlášeným uživatelům

## 9 Výsledky

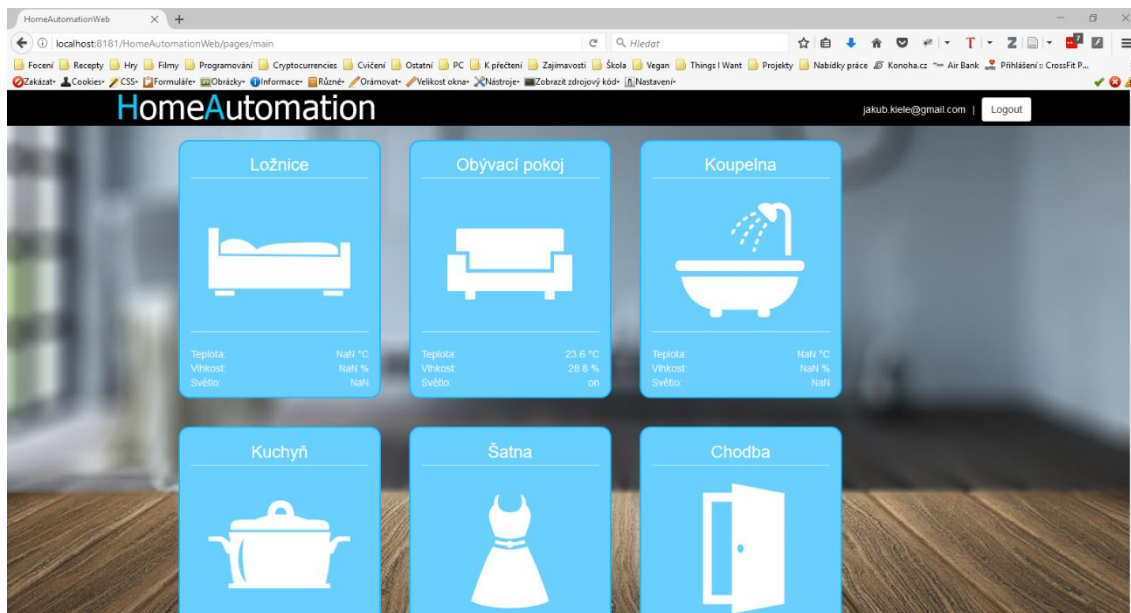
V práci byl navržen software pro řízení systému inteligentního domu. Pro vytvoření systému řízení je možné vybírat na trhu z velkého množství různého hardwaru komunikujícího přes různé technologie. Výsledný software je navržen tak, aby bylo možné zvolit libovolnou z těchto technologií. Díky modulárnímu návrhu je na hardwaru a konkrétní technologii závislá jen menší část systému, kterou lze nahradit bez zásahu do zbytku systému.

Modulární návrh obecně umožňuje vyměnit jakoukoliv část systému. Je tedy možné v hotovém systému domácí automatizace zaměnit například modul osvětlení (například za novější verzi) a získat tak novou funkcionalitu bez nutnosti velkých zásahů do dalších částí systému.

Mikropočítač Raspberry Pi zastává funkci centrálního prvku systému. Je k němu připojena sada sensorů a běží na něm aplikační server s výslednou aplikací. Sensory jsou k Raspberry připojeny pomocí GPIO pinů, což je postačující pro účely testování ale ne pro reálné použití. Vzniklo by několik problémů. Prvním je nutnost použití drátového připojení sensorů a druhým nedostatek GPIO pinů pro pokrytí celé domácnosti. Pro reálné použití by bylo vhodné zvolit si rozšiřující moduly Raspberry pro komunikaci s vybranými senzory a aktuátory. Vhodnou volbou je i rozšiřující deska UniPi board, která zjednoduší především drátovou komunikaci například pro ovládání světel nebo stínění.

S použitím Raspberry se pojí ještě jeden problém, a to umístění databáze a ukládání dat. Raspberry využívá místo disku SD kartu na které je uložen operační systém i všechna data. SD karty mají omezenou kapacitu (i když stále roste) a hlavně omezenou životnost z hlediska počtu zápisů. Jelikož se v systému periodicky sbírají a ukládají data, je vhodné k Raspberry připojit externí disk a nasměrovat databázi na něj, nebo nainstalovat databázi na úplně jiný server. Mimo to je Raspberry schopné fungovat jako centrální prvek automatizačního systému.

Navržený systém umožňuje definovat jednoduchá pravidla pro chování systému v každé místnosti domu zvlášť a na základě těchto pravidel automaticky řídit aktuátory v domácnosti. Kromě automatického ovládání je samozřejmě možné i ruční ovládání uživatelem. V prototypu je implementováno ovládání chytré žárovky Lifx.



**Obrázek 33: Webové rozhraní v prohlížeči FireFox.**

Pro systém bylo navrženo i webové uživatelské rozhraní, jehož náhled je možné spatřit na Obrázek 33. Ten ukazuje základní obrazovku s přihlášeným uživatelem a přehledem místností zaregistrovaných v systému i s aktuální teplotou, vlhkostí a stavem světel.

Výsledný prototyp umožňuje základní automatizaci, ale jeho funkcionality je omezená hardwarově i softwarově. Pro další vylepšení systému by bylo vhodné rozšíření systému o rozumně zvolenou sadu senzorů a aktuátorů a doprogramování funkcí spojených s možnostmi nového hardwaru.

Další možností zlepšení prototypu je doprogramování mobilní aplikace (pro android nebo iOS) a poskytnout tak uživateli další způsob ovládání systému.

Z hlediska dlouhodobého používání systému by momentálně mohl nastat problém s množstvím dat v databázi, a tedy i výkonem. Sensory generují velké množství dat a do budoucna by bylo vhodné navrhnout způsob zálohy či odmazávání dat. Vhodné by bylo uchovávat v aktivních tabulkách pouze omezené množství dat (například

maximálně týden stará) a zbytek odsouvat do archivačních tabulek. Tím by se zamezilo narůstání aktivních tabulek a prohledávání velkého počtu starých záznamů. Dle potřeby by bylo možné určit období po které je nutné data archivovat a vše starší mazat i z archivačních tabulek, aby databáze zbytečně nenabírala na objemu.

Výstupem práce je návrh systému pro řízení chytré domácnosti s využitím mikro-počítače Raspberry Pi jako centrálního prvku. Z tohoto návrhu vznikl i prototyp, který se skládá ze dvou částí:

- Systém řízení automatizace – řídí veškeré procesy odehrávající se v systému.
  - Umožňuje zpracování a persistenci dat.
  - Vykonává periodické úkoly jako je kontrole předdefinovaných pravidel řízení a vytváření události pro obsluhu hardwarových komponent.
  - Bezpečnostní modul monitoruje přítomnost uživatelů a v případě neočekávaného pohybu v objektu zašle emailovou zprávu s oznámením.
- Webové rozhraní – umožňuje uživateli přístup k samotnému systému a jeho ruční ovládání.
  - Umožňuje konfiguraci systému.
  - Nabízí pohled na shromážděná data a aktuální stav prostředí.
  - Poskytuje možnost ovládat osvětlení v místnostech.



## 10 Závěr

Cílem práce bylo seznámit s problematikou systémů pro řízení chytré domácnosti z pohledu hardwaru, technologií i návrhu. Součástí práce je návrh vlastního systému řízení domácnosti a implementace prototypu. K provozu prototypu je jako centrální prvek systému využit mikropočítač Raspberry Pi, senzory pohybu, intenzity světla, teploty a vlhkosti a chytrá žárovka LIFX.

První část práce je zaměřena především na seznámení s problematikou inteligentních domácností. Jsou zde definovány subsystémy, které lze v takové domácnosti využívat. Tyto subsystémy mohou být použity samostatně nebo je možné je integrovat do většího celku. Pro každý systém jsou zde představeny komerční řešení, kterým lze pokrýt potřeby daného subsystému.

Dále jsou definovány hardwarové prvky, které lze v chytrém domově nalézt a opět popsány některé konkrétní případy. Jedná se o řídicí jednotku, senzory, aktuátory a ovládací prvky, ze kterých se výsledné chytré prostředí skládá. Jsou zde představeny a popsány technologie, které je možné použít pro komunikaci mezi těmito hardwarovými prvky a na kterých lze chytré systémy vystavět.

Ke konci teoretické části jsou představeny některá kompletní řešení pro inteligentní domácnost. Je zde představeno profesionální řešení firmy Loxone zahrnující hardware i software, stejně jako open source projekty řídicích systému vystavěných na jazyce Java. Ty je možné taktéž provozovat na počítači Raspberry Pi.

V druhé části práce je představen samotný mikropočítač Raspberry Pi a hardwarové prvky použité k sestavení prototypu řídicího systému chytré domácnosti. Následuje popis návrhu vlastního řídicího systému. Pro jeho implementaci byla zvolena modulární struktura, která umožňuje snadnou záměnu částí systému v závislosti na požadavcích uživatele a použitém hardwaru. Architektura systému je vícevrstvá. Tato volba vychází z potřeb systému, kdy je odděleno uživatelské rozhraní, serverová část a databázová část.

Nakonec jsou představeny použité technologie a popsány některé detaily z implementace prototypu. Stručně popsány jsou problémy, které bylo třeba při implementaci řešit. Výsledný prototyp systému umožňuje sbírání dat ze sensorů připojených

k Raspberry Pi a nabízí uživateli možnost sledovat stav prostředí své domácnosti. Umožňuje také základní ovládání, a to především ovládání chytré žárovky Lix. Systém umožňuje i automatické ovládání na základě předdefinovaných pravidel pro jednotlivé místnosti.

## 11 Seznam použité literatury

- [1] The History of X10 By Edward B.Driscoll, Jr. *Home.planet.nl* [online]. 2002 [cit. 2017-04-22]. Dostupné z: [http://home.planet.nl/~lhendrix/x10\\_history\\_english.htm](http://home.planet.nl/~lhendrix/x10_history_english.htm)
- [2] MONEKOSSO, Dorothy, Yoshinori KUNO a Paolo REMAGNINO, ed. *Intelligent Environments: Methods, Algorithms and Applications*. London: Springer, 2009. 211 s. ISBN 1849967911.
- [3] Smart Homes. *Scotham Alliance Limited* [online]. 2017 [cit. 2017-04-23]. Dostupné z: <http://www.scothamalliance.com/services/smart-homes/>
- [4] Philips: Osobní osvětlení Philips Hue. *Philips: Osobní osvětlení Philips Hue* [online]. 2017 [cit. 2017-04-23]. Dostupné z: [http://www.philips.cz/c-m-li/osobni-bezdratove-osvetleni-hue#poznejte\\_hue](http://www.philips.cz/c-m-li/osobni-bezdratove-osvetleni-hue#poznejte_hue)
- [5] *LIFX* [online]. 2017 [cit. 2017-04-23]. Dostupné z: <https://www.lifx.com/>
- [6] LIFX: The Light Bulb Reinvented. *Kickstarter* [online]. 2017 [cit. 2017-04-24]. Dostupné z: <https://www.kickstarter.com/projects/limemouse/lifx-the-light-bulb-reinvented>
- [7] MiPow Playbulb Color. *MiPow* [online]. 2017 [cit. 2017-04-24]. Dostupné z: <https://www.mipow.com/pages/playbulb-color>
- [8] MiPow Playbulb Rainbow. *MiPow* [online]. [cit. 2017-04-24]. Dostupné z: <https://www.mipow.com/collections/playbulb/products/playbulb-rainbow>
- [9] MiPow Support. *MiPow* [online]. 2017 [cit. 2017-04-24]. Dostupné z: <https://support.mipow.com/hc/en-us/sections/204625308-FAQ>
- [10] Inteligentní topení. TotalISB - Inteligentní domy [online]. [cit. 2017-04-25]. Dostupné z: <http://www.intelligentni-dum.eu/intelligentni-topeni/>
- [11] Loxone - inteligentní regulace teploty. Loxone [online]. 2017 [cit. 2017-04-25]. Dostupné z: <https://www.loxone.com/cscz/chytry-dum/topeni-klimatizace/>
- [12] Meet Nest thermostat. Nest [online]. 2017 [cit. 2017-04-25]. Dostupné z: <https://nest.com/thermostat/meet-nest-thermostat/>
- [13] NKStore - termostat Nest. NKStore [online]. [cit. 2017-04-25]. Dostupné z: <https://nkstore.cz/thermostat/>

- [14] Can I use the Nest Thermostat if I have radiator valves? Nest Support [online]. 2017 [cit. 2017-04-25]. Dostupné z: <https://nest.com/uk/support/article/Can-I-use-the-Nest-Thermostat-if-I-have-radiator-valves>
- [15] Honeywell Evohome. Honeywell [online]. [cit. 2017-04-25]. Dostupné z: <http://connectedproducts.honeywelluk.com/evohome/>
- [16] Honeywell Evohome product specification. Get Connected Honeywell [online]. 2016 [cit. 2017-04-25]. Dostupné z: [https://getconnected.honeywell.com/cz/evohome\\_wi-fi.html#evohome-productspec](https://getconnected.honeywell.com/cz/evohome_wi-fi.html#evohome-productspec)
- [17] Statistické přehledy kriminality za rok 2013: Celková kriminalita za období od 1.1.2013 do 23.12.2013. In: Policie ČR [online]. 2013 [cit. 2017-04-25]. Dostupné z: <http://www.policie.cz/soubor/12-celkova-kriminalita-za-obdobi-od-01-01-2013-do-31-12-2013.aspx>
- [18] Elektronické zabezpečovací systémy. Falco Computer [online]. 2017 [cit. 2017-04-25]. Dostupné z: <http://www.falcocomputer.cz/elektroinstalace/ezs-elektronicke-zabezpecovaci-systemy>
- [19] Angee. The First Truly Autonomous Home Security System. Kickstarter [online]. 2016 [cit. 2016-06-28]. Dostupné z: <https://www.kickstarter.com/projects/tomtu/angee-the-first-truly-autonomous-home-security-sys/description>
- [20] Home Security. Simplified. MeetAngee [online]. 2016 [cit. 2016-06-28]. Dostupné z: <http://meetangee.com/#get-to-know-more>
- [21] ISmart Alarm, Inc. Ismartalarm [online]. 2016 [cit. 2016-06-28]. Dostupné z: <https://www.ismartalarm.com/us/>
- [22] ISmartAlarm Premium Package. Pcmag [online]. 2016 [cit. 2016-06-28]. Dostupné z: <http://www.pcmag.com/article2/0,2817,2455967,00.asp>
- [23] About smart home network. Irisbylowes [online]. 2016 [cit. 2016-06-28]. Dostupné z: <https://www.irisbylowes.com/about/>
- [24] What is Whole House Audio/Video?: Part 1. Smarthome.com: Home Automation Superstore [online]. 2017 [cit. 2017-04-25]. Dostupné z: <http://www.smarthome.com/sh-solutions-what-is-whole-house-audio-video.html>
- [25] What is Whole House Audio/Video?: Part 2. Smarthome.com: Home Automation Superstore [online]. 2017 [cit. 2017-04-25]. Dostupné z:

- <http://www.smarthome.com/sh-solutions-what-is-whole-house-audio-video-part-2.html>
- [26] Loxone multimédia. Loxone [online]. 2017 [cit. 2017-04-25]. Dostupné z: <https://www.loxone.com/cscz/chytry-dum/multimedia/>
- [27] Loxone music server pro 20 zón. Loxone eshop [online]. 2017 [cit. 2017-04-25]. Dostupné z: <http://shop.loxone.com/cscz/music-server-20-zon.html>
- [28] The Connected Home: Smart Home with Batteryless Wireless Technology. In: EnOcean Alliance: No Wires. No Batteries. No Limits. [online]. [cit. 2017-05-29]. Dostupné z: [https://www.enocean-alliance.org/wp-content/uploads/2016/11/Whitepaper\\_EnOcean\\_Alliance\\_White\\_Paper\\_Smart\\_Home\\_EN.pdf](https://www.enocean-alliance.org/wp-content/uploads/2016/11/Whitepaper_EnOcean_Alliance_White_Paper_Smart_Home_EN.pdf)
- [29] Loxone: Miniserver and extensions. Loxone [online]. [cit. 2017-05-16]. Dostupné z: <https://www.loxone.com/cscz/produkty/miniserver-extensions/>
- [30] VOJÁČEK, Antonín. Co se skrývá pod označení PLC. In: Automatizace HW [online]. [cit. 2017-05-18]. Dostupné z: <http://automatizace.hw.cz/co-se-skryva-pod-oznacnim-plc>
- [31] PLC - Programovatelný Logický Automat: Programmable Logic Controller. PLC Automatizace [online]. [cit. 2017-05-18]. Dostupné z: <http://www.plc-automatizace.cz/knihovna/plc.htm>
- [32] ZIMMERMANN, Luděk. Co to je PLC a k čemu? In: Sborník přednášek č. 40: 1. pololetí 2010. 2010.
- [33] Senzor. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-05-30]. Dostupné z: <https://cs.wikipedia.org/wiki/Senzor>
- [34] KOCHANÍČEK, Ludvík. Nejpoužívanější snímače v automatizaci. In: COPTEL - Elektrotechnika [online]. Česká republika [cit. 2017-05-30]. Dostupné z: <https://coptkm.cz/portal/?action=2&doc=7943&instance=1>
- [35] EnOcean Dolphin. In: EnOcean [online]. 2017 [cit. 2017-05-30]. Dostupné z: [https://www.enocean.com/fileadmin/redaktion/pdf/press/enocean\\_dolphin\\_EN.pdf](https://www.enocean.com/fileadmin/redaktion/pdf/press/enocean_dolphin_EN.pdf)
- [36] VIDIM, Jan. Snímače parametrů prostředí v budovách. In: TZB info: stavebnictví, úspory energií, technické zařízení budov [online]. 2017 [cit. 2017-05-

- 30]. Dostupné z: <http://www.tzb-info.cz/provoz-technologie/10125-snimace-pa-rametru-prostredi-v-budovach>
- [37] COLEMAN, Benson. How to Make a Robot – Lesson 3: Making Sense of Actuators. In: RoboShop: Putting robotics at your service [online]. 2017 [cit. 2017-05-30]. Dostupné z: <http://www.robotshop.com/blog/en/how-to-make-a-robot-lesson-3-actuators-2-3703>
- [38] What are Actuators? Learn about Actuators. In: Thomasnet [online]. 2017 [cit. 2017-05-30]. Dostupné z: <http://www.thomasnet.com/about/actuators-301168.html>
- [39] What's PLC?: Technology. In: HD PLC: Power Line Communication [online]. 2015 [cit. 2017-06-14]. Dostupné z: <http://www.hd-plc.org/modules/about/plc.html>
- [40] What is X10?: Home Automation over Powerline. In: SmartHome: Home Automation Superstore [online]. [cit. 2017-06-14]. Dostupné z: <http://www.smarthome.com/sc-what-is-x10-home-automation>
- [41] X10 vs. Z-Wave. In: Vesternet: Smart Home Simplified [online]. 2012 [cit. 2017-06-14]. Dostupné z: <http://www.vesternet.com/resources/technology-comparison/x10-vs-z-wave>
- [42] Universal Powerline Bus. In: Smart Home USA [online]. [cit. 2017-06-15]. Dostupné z: <http://www.smarthomeusa.com/universal-powerline-bus/>
- [43] About UPB Technology. In: Simply Automated [online]. [cit. 2017-06-15]. Dostupné z: [http://www.simply-automated.com/UPB\\_Technology.php](http://www.simply-automated.com/UPB_Technology.php)
- [44] UPB Powerline Carrier Home Control Technology. In: Home Toys: Online Trade Magazine Residential and Commercial Automation, Security & A/V Systems [online]. 2017 [cit. 2017-06-15]. Dostupné z: <http://www.hometoys.com/content.php?url=/htinews/jun04/articles/upb/upb.htm>
- [45] AGARWAL, Tarun. Different Types of Wireless Communication with Applications. In: ELPROCUS: Electronics, Projects, Focus - The budding electronics engineer's knowledge space [online]. [cit. 2017-07-26]. Dostupné z: <https://www.elprocus.com/types-of-wireless-communication-applications/>
- [46] Who we are: History. Wi-Fi Alliance [online]. [cit. 2017-07-26]. Dostupné z: <http://www.wi-fi.org/who-we-are/history>

- [47] Wi-Fi: Wireless Networking. Webopedia [online]. [cit. 2017-07-27]. Dostupné z: [http://www.webopedia.com/TERM/W/Wi\\_Fi.html](http://www.webopedia.com/TERM/W/Wi_Fi.html)
- [48] ISM band. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-07-27]. Dostupné z: [https://en.wikipedia.org/wiki/ISM\\_band](https://en.wikipedia.org/wiki/ISM_band)
- [49] Bluetooth. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-07-28]. Dostupné z: <https://en.wikipedia.org/wiki/Bluetooth>
- [50] Bluetooth [online]. [cit. 2017-07-27]. Dostupné z: <https://www.bluetooth.com/>
- [51] Bluetooth Core Specification 5.0 FAQ. In: Mouser [online]. [cit. 2017-07-28]. Dostupné z: <http://www.mouser.com/pdfdocs/bluetooth-5-faq.pdf>
- [52] Mesh networking. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-07-28]. Dostupné z: [https://en.wikipedia.org/wiki/Mesh\\_networking](https://en.wikipedia.org/wiki/Mesh_networking)
- [53] PAVLIS, Jakub. ZigBee: Když je pomalejší síť výhodnější. In: Notebook.cz [online]. [cit. 2017-07-28]. Dostupné z: <http://notebook.cz/clanky/technologie/2011/ZigBee>
- [54] ZigBee [online]. [cit. 2017-07-28]. Dostupné z: <http://www.zigbee.org/>
- [55] Z-Wave [online]. [cit. 2017-07-28]. Dostupné z: <http://www.z-wave.com/>
- [56] About Z-Wave Technology. Z-Wave Alliance [online]. [cit. 2017-07-28]. Dostupné z: [http://z-wavealliance.org/about\\_z-wave\\_technology/](http://z-wavealliance.org/about_z-wave_technology/)
- [57] What technology? In: Smarthome: Home Automation Superstore [online]. [cit. 2017-07-28]. Dostupné z: <http://www.smarthome.com/sc-what-technology>
- [58] Wireless comparison. In: Sparkfun: Start something [online]. [cit. 2017-07-28]. Dostupné z: <https://learn.sparkfun.com/tutorials/bluetooth-basics/wireless-comparison>
- [59] EnOcean [online]. [cit. 2017-07-28]. Dostupné z: <https://www.enocean.com/en/>
- [60] KNX [online]. [cit. 2017-07-28]. Dostupné z: <https://www.knx.org/cz/index.php>

- [61] MICHALEC, Libor. Úvod do KNX. In: Automatizace.hw: Rady a poslední novinky z oboru [online]. [cit. 2017-07-28]. Dostupné z: <http://automatizace.hw.cz/teorie-a-praxe/knx.html>
- [62] VOJÁČEK, Antonín. Sběrnice KNX pro řízení budov - 1.část. In: Automatizace.hw: Rady a poslední novinky z oboru [online]. [cit. 2017-07-28]. Dostupné z: <http://automatizace.hw.cz/clanek/2006061001>
- [63] Echelon [online]. [cit. 2017-07-28]. Dostupné z: <http://www.echelon.com/>
- [64] MATZ, Václav. Systémy používané v "inteligentních" budovách - přehled komunikačních protokolů. In: TZBinfo: stavebnictví, úspory energií, technická zařízení budov [online]. [cit. 2017-07-28]. Dostupné z: <http://vytapani.tzb-info.cz/mereni-a-regulace/6879-systemy-pouzivane-v-inteligentnich-budovach-prehled-komunikacnich-protokolu>
- [65] VOJÁČEK, Antonín. Sběrnice LonWorks - 1.část - Úvod. In: Automatizace.hw: Rady a poslední novinky z oboru [online]. [cit. 2017-07-28]. Dostupné z: <http://automatizace.hw.cz/clanek/2005040501>
- [66] VOJÁČEK, Antonín. Sběrnice LonWorks - 2.část - LonTalk protokol. In: Automatizace.hw: Rady a poslední novinky z oboru [online]. [cit. 2017-07-28]. Dostupné z: <http://automatizace.hw.cz/clanek/2005041101>
- [67] VOJÁČEK, Antonín. MODBUS. In: Automatizace.hw: Rady a poslední novinky z oboru [online]. [cit. 2017-07-28]. Dostupné z: <http://automatizace.hw.cz/clanek/2004070701>
- [68] RONEŠOVÁ, Andrea. Přehled protokolu MODBUS. Plzeň, 2005, 19 s. Dostupné také z: <http://home.zcu.cz/~ronesova/bastl/files/modbus.pdf>
- [69] Loxone [online]. [cit. 2017-07-31]. Dostupné z: <https://www.loxone.com/cscz>
- [70] What is Insteon home automation. Smart Home: Home Automation Superstore [online]. [cit. 2017-07-31]. Dostupné z: <http://www.smarthome.com/sc-what-is-insteon-home-automation>
- [71] Insteon [online]. [cit. 2017-07-31]. Dostupné z: <http://www.insteon.com/>
- [72] Introduction. OpenHAB: Empowering the smart home [online]. [cit. 2017-07-31]. Dostupné z: <http://www.openhab.org/introduction.html>
- [73] OpenHab: Empowering the smart home [online]. [cit. 2017-07-31]. Dostupné z: <http://www.openhab.org/>



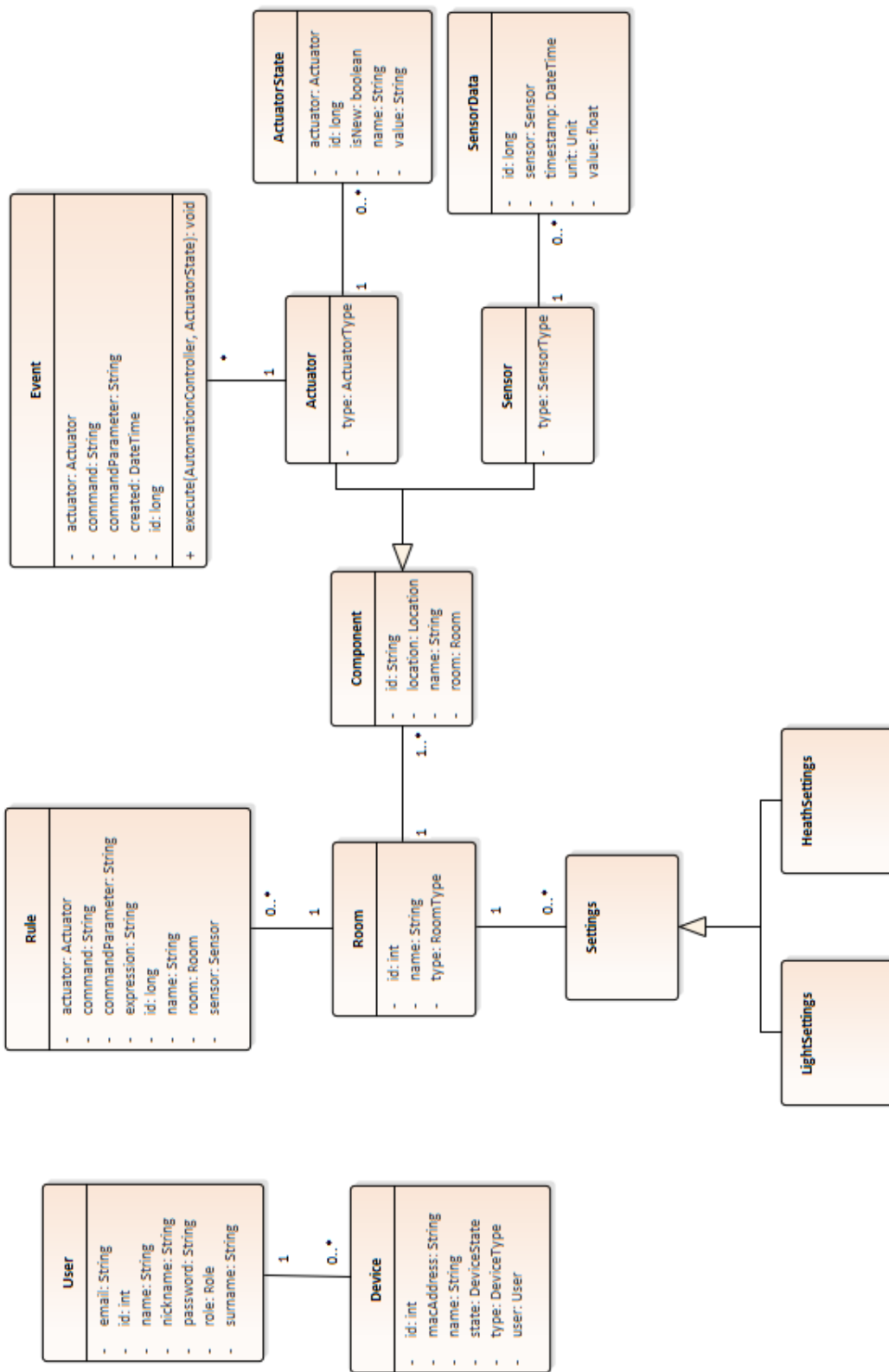
- [74] Home Assistant [online]. [cit. 2017-07-31]. Dostupné z: <https://home-assistant.io/>
- [75] Home Assistant - Architecture. Home Assistant [online]. [cit. 2017-07-31]. Dostupné z: <https://home-assistant.io/developers/architecture/>
- [76] Co je to řídicí systém Rex? Rex Controls [online]. [cit. 2017-07-31]. Dostupné z: <https://www.rexcontrols.cz/rex>
- [77] Řídicí systém REX pro Raspberry Pi. Rex Controls [online]. [cit. 2017-07-31]. Dostupné z: <https://www.rexcontrols.cz/ridici-system-rex-raspberry-pi>
- [78] HORÁČEK, Petr. Raspberry π VIII. - Úvod do GPIO. In: Linuxsoft [online]. [cit. 2017-08-01]. Dostupné z: [http://www.linuxsoft.cz/article.php?id\\_article=1953](http://www.linuxsoft.cz/article.php?id_article=1953)
- [79] Raspberry Pi 2 Model B. Raspberry Pi [online]. [cit. 2017-08-01]. Dostupné z: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>
- [80] Raspberry Pi 3 Model B. Raspberry Pi [online]. [cit. 2017-08-01]. Dostupné z: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- [81] UniPi Technology [online]. [cit. 2017-08-01]. Dostupné z: <https://www.unipi.technology/cs/>
- [82] RaZberry Z-Wave [online]. [cit. 2017-08-01]. Dostupné z: <https://razberry.z-wave.me/>
- [83] RaZberry 2. Smarter Home: Home Automation and Control [online]. [cit. 2017-08-01]. Dostupné z: <https://smarterhome.sk/sk/zwave-pc-adaptery/razberry-2-300.html>
- [84] XBee/XBee-PRO moduly Digi. In: Vývoj.hw: Profesionální elektronika [online]. [cit. 2017-08-01]. Dostupné z: <http://vyvoj.hw.cz/embedded/xbeexbee-pro-moduly-digi.html>
- [85] XBee Explorer Dongle -rhydoLABZ. In: RhydoLabz: Embedding intelligence [online]. [cit. 2017-08-01]. Dostupné z: [http://www.rhydolabz.com/wireless-zigbee-xbee-c-130\\_133/xbee-explorer-dongle-rhydolabz-p-1093.html](http://www.rhydolabz.com/wireless-zigbee-xbee-c-130_133/xbee-explorer-dongle-rhydolabz-p-1093.html)
- [86] TOMAR, Ankur. EnOcean Pi: Transforms Raspberry Pi into a Wireless Gateway. In: *Element 14* [online]. [cit. 2017-08-01]. Dostupné z: <https://www.element14.com/community/docs/DOC-55169/1/enOcean-pi-transforms-raspberry-pi-into-a-wireless-gateway>

- [87] EnOcean Pi 868. *RPiShop: Váš dodavatel Raspberry Pi* [online]. [cit. 2017-08-01]. Dostupné z: <http://rpishop.cz/rozsirujici-moduly/628-enocean-pi-868.html>
- [88] Senzor teploty a vlhkosti vzduchu DHT-22. *Santy: Automatizace a robotizace* [online]. [cit. 2017-08-01]. Dostupné z: <http://www.santy.cz/senzory-c24/dht22-alone-i103/>
- [89] Světelný senzor - detekce světelných podmínek. Digital i analog. *Santy: Automatizace a robotizace* [online]. [cit. 2017-08-01]. Dostupné z: <http://www.santy.cz/moduly-c22/light-sensor-i85/>
- [90] PIR modul. *GM Electronic* [online]. [cit. 2017-08-01]. Dostupné z: <https://www.gme.cz/pir-modul-sb00612a>
- [91] PostgreSQL [online]. [cit. 2017-08-01]. Dostupné z: <https://www.postgresql.org/>
- [92] Java [online]. [cit. 2017-08-01]. Dostupné z: <https://www.java.com/en/>
- [93] Go Java [online]. [cit. 2017-08-01]. Dostupné z: <https://go.java/index.html>
- [94] Spring framework [online]. [cit. 2017-08-01]. Dostupné z: <https://projects.spring.io/spring-framework/>
- [95] What is Maven. Maven [online]. [cit. 2017-08-01]. Dostupné z: <https://maven.apache.org/what-is-maven.html>
- [96] Hibernate [online]. [cit. 2017-08-01]. Dostupné z: <http://hibernate.org>
- [97] MUELLER, John. Understanding SOAP and REST Basics And Differences. In: Smartbear [online]. [cit. 2017-08-01]. Dostupné z: <https://blog.smartbear.com/apis/understanding-soap-and-rest-basics/>
- [98] JBoss aplikační server. In: Root.cz [online]. [cit. 2017-08-01]. Dostupné z: <https://www.root.cz/clanky/jboss-aplikacni-server/>
- [99] Tomcat [online]. [cit. 2017-08-01]. Dostupné z: <https://tomcat.apache.org/>
- [100] RedHat about company. RedHat [online]. [cit. 2017-08-01]. Dostupné z: <https://www.redhat.com/en/about/company>
- [101] Wildfly - about. Wildfly [online]. [cit. 2017-08-01]. Dostupné z: <http://wildfly.org/about/>

## 12 Přílohy

- 1) Příloha 1 - Class diagram jádra systému s atributy
- 2) Příloha 2 - Struktura přiloženého CD

Příloha 1 - Class diagram jádra systému s atributy



- **Aplikace**
  - **Server**
    - **standalone-full.xml** – konfigurační soubor pro aplikační server Wildfly 9 s předdefinovaným data sourcem
    - **database\_structure.sql** – skript pro vytvoření struktury databáze
    - **HomeAutomation\_transcom.war** – komunikační modul (WAR)
    - **HomeAutomation\_eventmanager.war** – eventmanager modul (WAR)
    - **HomeAutomation\_rest\_api.war** – REST API modul (WAR)
    - **HomeAutomation.ear** – předchozí moduly zabalené do jednotného archivu (EAR)
    - **HomeAutomation** – složka se zdrojovými kódy serverové části
  - **Skripty sensorů** – složka obsahující python skripty obsluhy sensorů a bash skript pro jejich start
  - **WebClient**
    - **HomeAutomation\_WebGUI.war** – webové rozhraní aplikace (WAR)
    - **HomeAutomation\_WebGUI** – složka se zdrojovými kódy webového rozhraní
- **Text práce**
  - **Kiele\_Jakub\_diplomova\_prace.png** – text práce ve formátu PDF

**Podklad pro zadání DIPLOMOVÉ práce studenta**

<b>PŘEDKLÁDÁ:</b>	<b>ADRESA</b>	<b>OSOBNÍ ČÍSLO</b>
Kiele Jakub	Na Šancích 1176, Chrudim - Chrudim IV	I1462

**TÉMA ČESKY:**

Chytrý dům na platformě RaspberryPI

**TÉMA ANGLICKY:**

Smart Home using RaspberryPI Platform

**VEDOUCÍ PRÁCE:**

doc. Mgr. Tomáš Kozel, Ph.D. - KIKM

**ZÁSADY PRO VYPRACOVÁNÍ:**

Cíl: Seznámit se s problematikou inteligentních domů a navrhnout vzorové řešení na platformě RaspberryPi.

Osnova:

- 1.Úvod
- 2.Inteligentní dům a jeho subsystemy
- 3.Komunikační sběrnice
- 4.Hardwarové prvky v inteligentním domě
- 5.Představení vybraných existujících řešení
- 6.Návrh systému řízení inteligentního domu s využitím Raspberry Pi
- 7.Výsledky
- 8.Závěr

**SEZNAM DOPORUČENÉ LITERATURY:**

--

**Podpis studenta:** .....

**Datum:** .....

**Podpis vedoucího práce:** .....

**Datum:** .....