



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

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INTELIGENTNÍCH SYSTÉMŮ**  
DEPARTMENT OF INTELLIGENT SYSTEMS

# **NÁVRH A REALIZACE IOT PRO MONITOROVÁNÍ A ŘÍZENÍ CHYTRÉ DOMÁCNOSTI**

**DESIGN AND IMPLEMENTATION OF IOT FOR MONITORING AND CONTROL OF SMART  
HOMES**

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**JAKUB SMEJKAL**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**Doc. Ing. VLADIMÍR JANOUŠEK, Ph.D.**

**BRNO 2021**

## Zadání bakalářské práce



23957

Student: **Smejkal Jakub**

Program: Informační technologie

Název: **Návrh a realizace IoT pro monitorování a řízení chytré domácnosti**  
**Design and Implementation of IoT for Monitoring and Control of Smart Homes**

Kategorie: Softwarové inženýrství

Zadání:

1. Prostudujte problematiku návrhu a realizace řídicích systémů inteligentních budov a chytrých domácností. Seznamte se s existujícími volně dostupnými technologiemi a vhodnými technickými prostředky.
2. Stanovte požadavky na dohledový a řídicí systém chytré domácnosti. Berte v úvahu sledování přítomnosti osob, řízení osvětlení a HVAC (heating-ventilation-air-condition).
3. Navrhněte systém podle zvolených požadavků. Chytré senzory navrhněte s využitím hardwarových komponent typu Arduino s vhodnými senzory a aktuátory. Pro komunikaci s řídicím a monitorovacím systémem použijte protokol MQTT. Řídicí a monitorovací systém může být navržen jako kombinace existujících komponent (NodeRED, Domoticz apod.) a vlastního řešení.
4. Navržený systém prototypově realizujte a otestujte jeho funkčnost. V prototypové realizaci lze částečně použít i simulované komponenty.
5. Ověrte funkčnost výsledného řešení a vyhodnoťte dosažené výsledky.

Literatura:

- Home Assistant. URL: <https://www.home-assistant.io>
- Domoticz. URL: <https://www.domoticz.com>
- Tasmota. URL: <https://tasmota.github.io/docs/>
- Automatizace. URL: <http://www.automatizace.cz/>

Pro udělení zápočtu za první semestr je požadováno:

- První 3 body zadání.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Janoušek Vladimír, doc. Ing., Ph.D.**

Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.

Datum zadání: 1. listopadu 2020

Datum odevzdání: 12. května 2021

Datum schválení: 11. listopadu 2020

## **Abstrakt**

Tato bakalářská práce se zabývá generickým návrhem a realizací systému pro IoT, konkrétně pak chytrou domácnost. Cílem této práce je porovnat existující systémy a hardwarové prostředky. Součástí práce je také několik ukázkových firmwarů a také prototyp softwaru pro dohledový a řídicí systém.

Zvolený problém je vyřešen pomocí návrhu obecného systému pro chytrou domácnost. Pro vzorovou implementaci systému byla využita hardwarová zařízení od české firmy HARDWARIO a senzory od různých výrobců. Ve vzorové implementaci budou použity různé enviromentální senzory, detektory pohybu apod.

Vytvořený prototyp aplikace bude umožňovat přehledný a snadný monitoring celé domácnosti na jednom místě pomocí UI, které bude plně upravitelné pro potřeby uživatele.

## **Abstract**

This thesis deals with a generic design and implementation of an IoT system, specifically for a smart home. The thesis aims to compare existing systems and hardware solutions. In this thesis there are a few examples of hardware firmwares and a prototype of a software for controlling and monitoring.

The selected problem is resolved by designing a general system for a smart home. As a sample implementation, the system will be created by using hardware devices from czech company HARDAWRIO and sensors from various vendors. It will contain some environmental sensors, motion detectors, etc.

The implemented prototype of an application allows easy monitoring of the whole home in one place thanks to an uncluttered UI, which is fully customizable for the user's needs.

## **Klíčová slova**

domácí automatizace, chytrá domácnost, IoT, environmentální monitoring, monitoring, mikroprocesory, mikrokontrolery, MQTT, node-red, Blynk, monitoring domácnosti, obecný návrh systému

## **Keywords**

home automation, smart home, IoT, environmental monitoring, monitoring, microprocessor, microcontrollers, MQTT, node-red, Blynk, home monitoring, generic system design

## **Citace**

SMEJKAL, Jakub. *Návrh a realizace IoT pro monitorování a řízení chytré domácnosti*. Brno, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Doc. Ing. Vladimír Janoušek, Ph.D.

# **Návrh a realizace IoT pro monitorování a řízení chytré domácnosti**

## **Prohlášení**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Doc. Ing. Vladimíra Janouška, Ph.D. Další informace mi poskytli zaměstnanci firmy HARDWARIO s.r.o. se kterými spolupracuji i mimo řešení bakalářské práce. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....  
Jakub Smejkal  
3. května 2021

## **Poděkování**

Děkuji Doc. Ing. Vladimíru Janouškovi, Ph.D. za pomoc při řešení bakalářské práce. Mé poděkování patří též zaměstnancům HARDWARIO s.r.o. za spolupráci při získávání údajů pro výzkumnou část práce a zapůjčení hardwarových sad pro vzorovou implementaci a testování.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Současný stav IoT a Chytré domácnosti</b>	<b>4</b>
2.1	Dohledové a řídicí systémy . . . . .	4
2.2	Hardware používaný pro chytré domácnosti . . . . .	9
2.3	Další software používaný v chytré domácnosti . . . . .	24
<b>3</b>	<b>Požadavky na dohledový a řídicí systém</b>	<b>31</b>
3.1	Hardwarové prvky a jejich využití v navrženém systému . . . . .	32
3.2	Software pro dohledový a řídicí systém . . . . .	34
3.3	Přístup k chytré domácnosti a zabezpečení . . . . .	35
<b>4</b>	<b>Realizace řešení</b>	<b>37</b>
4.1	Serverové řešení . . . . .	37
4.2	Hardwarové řešení . . . . .	38
4.3	Softwarové řešení . . . . .	43
<b>5</b>	<b>Závěr</b>	<b>53</b>
5.1	Srovnání řešení . . . . .	53
5.2	Shrnutí . . . . .	55
<b>Literatura</b>		<b>57</b>
<b>A</b>	<b>Ukázka hardwarových zařízení</b>	<b>60</b>

# Kapitola 1

## Úvod

Chytrá domácnost a IoT je již poměrně známý pojem i mezi širší veřejností. Softwarových řešení se na trhu nachází spousta, taktéž i připravených hardwareových prvků.

Díky těmto technologiím je možné zvýšit produktivitu na pracovištích nebo zlepšit životní prostředí v domácnosti. Správně navržený monitorovací a řídicí systém pro budovu také může zvýšit efektivitu vytápění, osvětlení atp. a zároveň snížit peněžní náklady.

IoT neboli Internet of Things samozřejmě zahrnuje mnohem více odvětví než jen chytré domácnosti. Některé z nich jsou například chytrá města, chytré továrny, výrobní linky atp. Těmi se tato práce nezabývá.

Oblast IoT, chytrých domácností a budov se v posledních několika letech rozrůstá. S přibývajícími systémy pro řízení, hardwareovými zařízeními připravenými pro použití jen vsunutím baterií či celých firem zabývajících se pouze touto problematikou se otevírá několik možností, jak si zařídit profesionální řešení.

Dále, díky několika firmám a systémům jako jsou například Arduino, Domoticz, Node-RED se rozšířilo řešení chytrých domácností i mezi tzv. *bastlíře*, což jsou lidé, kteří si chtějí zařídit chytrou domácnost po svém, pomocí právě zmíněných technologií nebo v kombinaci s nějakým profesionálním řešením.

Podle dosavadního vývoje to vypadá, že se takovéto řešení domácnosti, ať už profesionální či méně profesionální bude rozrůstat čím dál více a tím pádem bude více domácností monitorováno a ovládáno pomocí nějakých inteligentních systémů.

Práce se zaměřuje na obecný návrh chytré domácnosti a systému pro řízení a monitorování. Pro názornost bude v rámci práce vytvořeno vzorové fyzické řešení. Toto ukázkové řešení bude realizováno pomocí zařízení od společnosti HARDARIO s.r.o. se kterou autor práce v době psaní spolupracuje a podílí se na vývoji pro tuto firmu. Jako příklad bude uveden autorův byt, kde bude nasazeno několik vzorových zařízení.

Výsledkem práce bude také prototypová aplikace pro PC se systémem Windows, ve které bude možné vytvořit několik dashboardů s ovládacími prvky, pomocí kterých bude uživatel monitorovat a ovládat celou domácnost. Tato aplikace bude porovnána s řešením vytvořeným pomocí Home Assistant.

V první kapitole 2 budou probrána vybraná řešení monitorovacích systémů a dostupných hardwareových technologií, jejich výhody a nevýhody a porovnání všech těchto systémů a zařízení. V této kapitole budou také probrány některé používané senzory a aktuátory včetně jejich funkčních principů.

Z větší časti budou probírány open-source systémy, a to vzhledem k jejich dostupnosti a rozšířitelnosti. Ze strany hardwareových řešení jsou vybrána známá řešení jako Arduino společně s méně známým a již zmíněným HARDARIO s.r.o. pocházejícím z České republiky.

Firma HARDWARIO poskytla několik zařízení z jejich hardwarové stavebnice TOWER pro vytvoření neinvazivního vzorového řešení.

Ve druhé kapitole **3** budou stanoveny některé základní hardwarové prvky důležité pro řídicí a monitorovací systém chytré domácnosti. Dále zde budou zmíněny některé základní požadavky na prototypovou aplikaci, která bude navržena v další kapitole. Dále zde budou zmíněny požadavky na přístup k dohledovému a řídícímu systému a jeho zabezpečení.

Ve třetí kapitole **4** bude probráno samotné řešení vzorové implementace systému. Hardwarové řešení bude zahrnovat několik firmwarů vytvořených pro stavebnici HARDWARIO TOWER vzorově implementovaných pro konkrétní použití v chytré domácnosti. Softwarová část bude popisovat prototypovou aplikaci a také vzorové řešení pomocí aplikace Home Assistant. Další částí bude popis serverového řešení použitého při implementaci.

V závěru **5** bude provedeno porovnání vlastního řešení s řešením pomocí aplikace Home Assistant. Dále bude v závěru provedeno závěrečné shrnutí práce.

## Kapitola 2

# Současný stav IoT a Chytré domácnosti

V této kapitole se bude práce zaměřovat na současný stav řešení pro chytrou domácnost.

V první části 2.1 budou zpracovány platformy pro dohled a ovládání. Na konci této části se nachází tabulka s porovnáním probíraných plaforem.

Další částí v kapitole je srovnání několika hardwarových mikrokontrolerů, používaných v tomto odvětví. Některé z nich se používají jako hotová řešení a jiná jsou spíše připravena pro vlastní vývoj a další rozšíření. Po srovnání zmíněných mikrokontrolerů budou probrány senzory a aktuátory využívané pro chytrou domácnost.

Na závěr budou shrnutý znalosti získané ze zkoumání dohledových a řídicích systémů z první části a pomocí nich stanoveny požadavky na finální aplikaci implementovanou v pozdějších částech této práce.

### 2.1 Dohledové a řídicí systémy

Dohledový systém je software, který propojuje komponenty chytré domácnosti k němu připojené. Tyto komponenty mohou zařizovat různé funkce jako vytápění, osvětlení, měření kvality prostředí, zabezpečení a další. Uživateli je díky tomuto systému umožněno na tato data dohlížet a domácnost ovládat pomocí uživatelského rozhraní. Častou platformou zvolenou pro zobrazení uživatelského rozhraní je web z důvodu dostupnosti na jakémkoli zařízení s webovým prohlížečem, pro pohodlnější ovládání jsou dostupné mobilní aplikace či jednoduché dotykové obrazovky.

Zařízení, na kterém je tento systém provozován, se nazývá Hub. Toto zařízení slouží jako prostředník mezi zařízeními, která mohou komunikovat pomocí různých technologií či protokolů. Díky tomuto zařízení je možné spojit například osvětlení komunikující pomocí protokolu Zigbee s mobilním telefonem komunikujícím pomocí Wi-Fi.

Uživatelské rozhraní v dohledovém a řídicím systému může obsahovat i další možnosti úprav jako je nastavení serveru či uživatelů. Pro zobrazení samotných údajů o chytré domácnosti se používá takzvaný dashboard, který obsahuje elementy jako jsou spínače, grafy a hodnotová pole.

Existuje několik uzavřených neboli *closed-source* systémů vyvíjených různými firmami jako je Microsoft HomeOS, ale čím dál více se na trhu objevují open-source systémy. Tato práce se bude zabývat primárně technologiemi open-source z důvodu jednoduché integrace s jinými systémy a důvody uvedenými v seznamu níže.

Vysoká preference open-source systémů koncovým uživatelem je způsobena především z důvodů, které jsou uvedeny v následujícím seznamu.

- **Vysoká kontrola nad daty a soukromím** - Díky otevřenosti softwaru a popřípadě i hardwaru je možné kontrolovat, která data a jak se posílají. U uzavřených systémů nemá uživatel nad tímto plnou kontrolu, a i když se většinou jedná o systémy velkých firem, je možné, že dojde k úniku dat ať už úmyslnému, či neúmyslnému.
- **Udržitelnost systému po ukončení podpory** - Otevřený software umožňuje zachování podpory a popřípadě další vývoj ze strany komunity, i když původní vývojáři projekt opustí. Při ukončení podpory uzavřených systémů mohou být uživatelé zanecháni bez podpory.
- **Dohled a přispívání k vývoji** - Při vývoji otevřeného softwaru je kód dostupný všem. Tato skutečnost znamená, že pokud bude vývojářská firma vytvářet málo za-bezpečený, pomalý či všeobecně špatný výrobek, nebude to možné před zákazníky ukrýt.
- **Zkvalitnění produktu díky příspěvkům komunity** - Do otevřeného softwaru může přispívat kdokoli a kdekoli na světě. Znamená to, že firma nemusí spoléhat pouze na znalosti a zkušenosti svých zaměstnanců. Čím větší komunita se kolem produktu vytvoří, tím se vývoj může zrychlit a zkvalitnit.

V této části se práce zaměří na některá z nejpoužívanějších řešení, která jsou oblíbená mezi vývojáři a uživateli chytrých domácností.

## Domoticz

Domoticz je open-source systém určený pro operační systémy Windows, Linux, macOS a vestavěné neboli embedded zařízení jako je například Raspberry Pi. Pro komunikaci se zařízeními v chytré domácnosti je možné použít celou řadu technologií jako je LoRa, Bluetooth, Zigbee a Ethernet. Seznam všech podporovaných zařízení je možné nalézt na wiki tohoto systému.<sup>1</sup>

K zobrazení dat a ovládání domácnosti je možné využít responzivní webové uživatelské rozhraní podporované ve všech prohlížečích. Na mobilních zařízeních je dále možné využít mobilní aplikace.<sup>2</sup>

---

<sup>1</sup>Seznam podporovaných hardwarových zařízení Domoticz <https://www.domoticz.com/wiki/Hardware>

<sup>2</sup>iOS aplikace pro Domoticz <https://www.domoticz.com/wiki/IOS>



Obrázek 2.1: Příklad zobrazení dat a ovládacích prvků s rozšířením Dashticz. Foto dashticz.readthedocs.io [10]

### Hlavní přednosti

- Jednoduchá instalace
- Podrobná dokumentace a návody
- Podpora všech hlavních OS (Windows, Linux, macOS, embedded)
- Přehledné webové rozhraní
- Aktivní komunita a fórum<sup>3</sup>

### OpenHAB

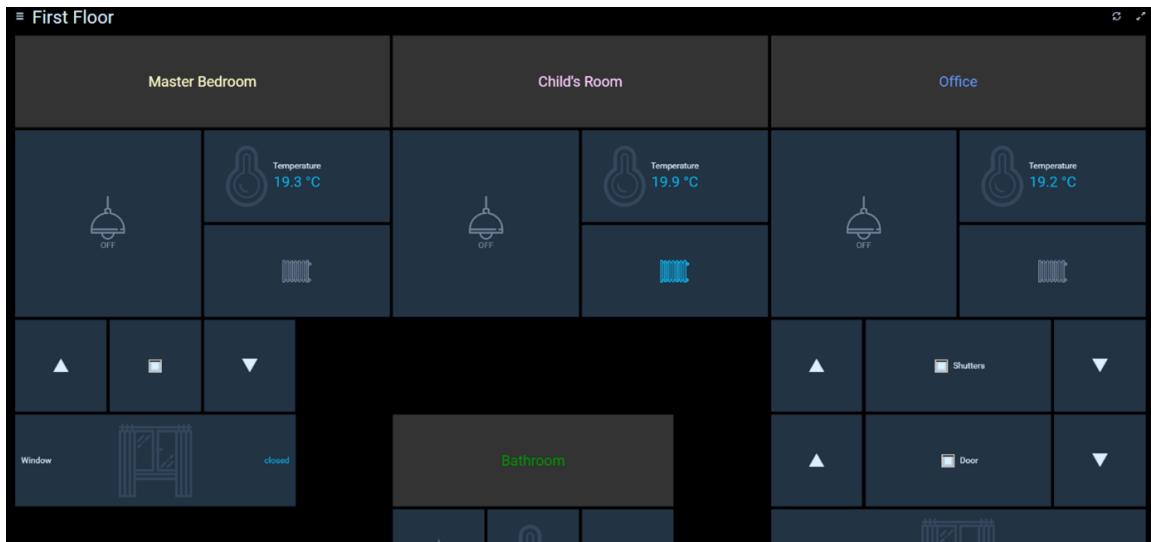
OpenHAB je další open-source software pro dohled a řízení, který lze provozovat na operačních systémech Windows, Linux, macOS a dalších embedded zařízeních. Ke komunikaci se zařízeními v chytré domácnosti je možné pomocí technologií jako je Bluetooth, Z-Wave a na rozdíl od předchozího systému i nativně pomocí protokolu MQTT, který je využíván v této práci.

Seznam všech podporovaných zařízení a technologií včetně informací o každé integraci je možné nalézt v dokumentaci systému<sup>4</sup>.

Pro ovládání a zobrazení dat o domácnosti je možné využít webové rozhraní nebo mobilní aplikace dostupné pro systémy Android i iOS.

<sup>3</sup>Odkaz na komunitní fórum <https://www.domoticz.com/forum/>

<sup>4</sup>Seznam add-onů pro OpenHAB [https://www.openhab.org/addons/](https://www.openhab.org/addons)



Obrázek 2.2: Demo aplikace v openHAB. Foto autor

### Hlavní přednosti

- Průvodce nastavením podle zkušeností uživatele
- Nativní mobilní aplikace pro Android i iOS zařízení
- Přehledné uživatelské rozhraní

### Home Assistant

Home Assistant je poslední probíraný open-source systém. Tento systém je možné nainstalovat na všechny operační systémy jako předchozí dohledové softwary. Některé druhy instalací ovšem neumožňují přístup ke všem funkcím.

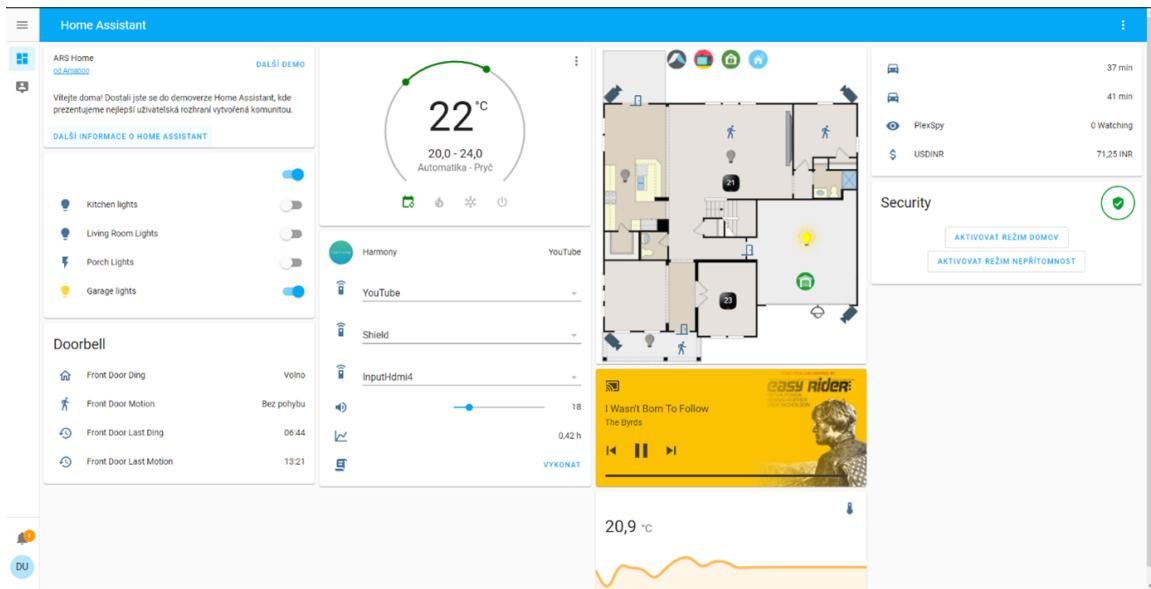
Seznam integrací je možné nalézt na wiki stránkách tohoto systému<sup>5</sup>, kde se nachází více než 1700 integrací.

Pro dohled nad prvky chytré domácnosti je možné použít webové rozhraní či mobilní aplikaci dostupnou na systém Android i iOS. Dashboardy na webu i v aplikaci jsou ekvivalentní.

Home Assistant poskytuje demo dashboardy [2], které byly vytvořené uživateli této platformy. Díky demu je možné si vyzkoušet celý systém i bez jakékoli investice do fyzických zařízení.

---

<sup>5</sup>Seznam integrací Home Assistant: <https://www.home-assistant.io/integrations/>



Obrázek 2.3: Demo aplikace dostupná na oficiálních stránkách Home Assistant. Foto autor

### Hlavní přednosti

- Přehledný Dashboard
- Demo aplikace na vyzkoušení bez investice do hardwaru
- Propracovaná dokumentace
- Velké množství integrací

## 2.2 Hardware používaný pro chytré domácnosti

V této kapitole budou probrány mikrokontrolery, senzory, aktuátory a další hardwarové prvky využívané při implementaci chytré domácnosti.

### Mikrokontrolery

Mikrokontroler je zařízení schopné získávat data z připojených senzorů a ovládat aktuátory jako je relé, písty a další. Mikrokontroler je integrovaný obvod, ve kterém se nachází mikroprocesor spolu s pamětí a dalšími obvody schopnými ovládat a monitorovat další elektronická zařízení.

Pro získávání dat a odesílání dat externím zařízením využívají mikrokontrolery sběrnice  $I^2C$  či 1-wire. Dále je možné využít rozhraní SPI, UART a pro komunikaci například s kamerou rozhraní USB. Tyto druhy komunikace budou více probrány v kapitole 2.1.1.

Mikrokontroler by také měl být schopný komunikovat s dalšími zařízeními pomocí komunikačních technologií jako je Bluetooth, Wi-Fi, rádio nebo Ethernet.

### Arduino

Arduino je označení jednodeskových počítačů založených na mikrokontrolerech ATmega od firmy Atmel původně od italské společnosti Smart Projects. Mikrokontrolery ATmega využívají mikročipy typu RISC s Harwardovskou architekturou.

Desky jsou vybaveny několika vstupními piny. Počet pinů se liší podle typu desky a jedná se o kombinaci digitálních a analogových pinů. Pro komunikaci s externími senzory je možné využít sběrnice  $I^2C$ , 1-wire či rozhraní SPI.

Komunikace a nahrávání firmwaru do zařízení probíhá pomocí sériového rozhraní UART, některé desky ulehčují komunikaci díky USB rozhraní. K programování firmwarů se využívá jazyk C či C++.

Základní konfigurace desek neumožňuje komunikaci pomocí jiné metody než pomocí fyzického propojení. Pro komunikaci s dalšími zařízeními pomocí jiných technologií je možné využít takzvané Shieldy a rozšiřující moduly.

V následující tabulce (Tabulka 1) jsou porovnány některé vybrané typy desek Arduino dostupné na trhu.

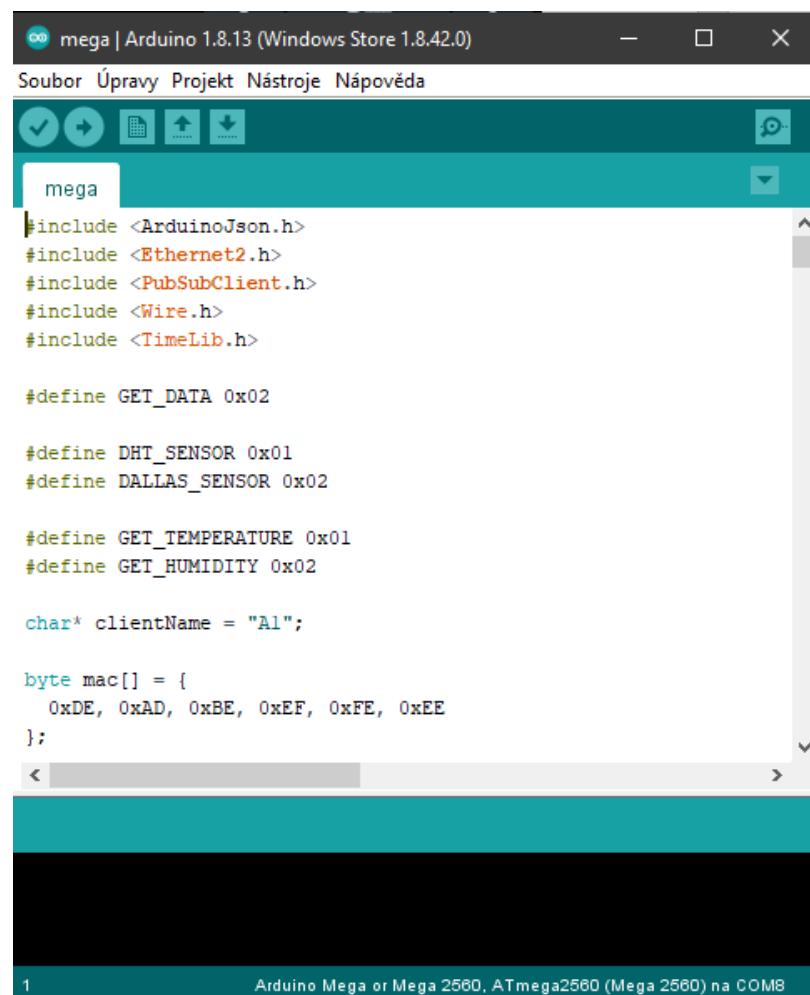
	Procesor	Flash	SRAM	Digitální I/O Piny	PWM Piny	Analogové Vstupy	Rozměry
<b>Micro</b>	16 MHz Atmega 32U4	32 Kb	2.5 Kb	20	7	12	18x49mm
<b>Uno</b>	16 MHz Atmega 328	32 Kb	2 Kb	14	6	6	53x75mm
<b>Mega</b>	16 MHz Atmega 2560	256 Kb	8 Kb	54	14	16	53x102mm

Tabulka 2.1: Porovnání několika Arduino desek různých velikostí. [15]

## Výhody

Jedna z největších výhod pro vývojáře firmwaru pro Arduino desky je IDE (Integrated Development Environment), které umožňuje vkládání nových knihoven pro různé senzory, poskytuje podporu pro všechny desky a umožňuje rychlé a jednoduché přeložení i nahrání firmwaru do zvolených Arduino desek. Tento program je dostupný pro všechny běžné operační systémy.

Další výhodou je velká a aktivní komunita. Členové komunity vytvářejí nové integrace pro další prvky, které nejsou dosud dostupné pro tuto platformu.



The screenshot shows the Arduino IDE interface. The title bar reads "mega | Arduino 1.8.13 (Windows Store 1.8.42.0)". The menu bar includes "Soubor", "Úpravy", "Projekt", "Nástroje", and "Nápověda". Below the menu is a toolbar with icons for file operations. The main area is a code editor with the following content:

```
#include <ArduinoJson.h>
#include <Ethernet2.h>
#include <PubSubClient.h>
#include <Wire.h>
#include <TimeLib.h>

#define GET_DATA 0x02

#define DHT_SENSOR 0x01
#define DALLAS_SENSOR 0x02

#define GET_TEMPERATURE 0x01
#define GET_HUMIDITY 0x02

char* clientName = "A1";

byte mac[] = {
    0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xEE
};
```

The status bar at the bottom indicates "1" and "Arduino Mega or Mega 2560, ATmega2560 (Mega 2560) na COM8".

Obrázek 2.4: Ukázka programu v Arduino IDE. Foto autor

## HARDWARIO TOWER – Industrial IoT Kit

Vývojový kit od české hardwarové společnosti. Hlavní částí je takzvaný Core Module (červený modul na Obrázku 2.5, na kterém se nachází 32bitový ARM mikrokontroler.

Pro komunikaci s dalšími moduly a externími senzory je možné využít sběrnice  $I^2C$  a SPI, 18 vstupně výstupních portů a tři sériová rozhraní UART.

Na rozdíl od desek Arduino, Core Module obsahuje několik integrovaných senzorů. Jedná se o digitální teplotní senzor TMP112, akcelerometr LIS2DH12[23]. Dále je na desce dostupný rádiový modul založený na SPIRIT1 od firmy ST, který pracuje na frekvencích 868/915 MHz (Evropa/USA)[24]. Tento modul umožňuje desce komunikovat bezdrátově s Hubem v chytré domácnosti bez nutnosti dalších rozšiřujících modulů jako tomu bylo u předchozí platformy 2.2. Tato komunikace je podmíněná přítomností USB donglu, který přijímá a odesílá data Core modulům, které jsou s tímto donglem spárovány.

Podobně, jako jsou u platformy Arduino dostupné rozšiřující Shieldy, jsou v HARDWARIO TOWER dostupné rozšiřující moduly, které je možné použít mimo jiné jako napájení, či obsahují další senzory a aktuátory.

Na Obrázku 2.5 je možné vidět příklad těchto modulů. Modrý  $CO_2$  modul sloužící k měření koncentrace  $CO_2$  v ovzduší a žlutý Battery modul dodávající celému zařízení napájení. Zbylé 3 černé moduly neboli takzvané tagy slouží jako dodatečné externí senzory.



Obrázek 2.5: Měřič  $CO_2$  postavený pomocí HARDWARIO IoT Kit. Foto autor

Zařízení založená na tomto kitu je možné napájet i několik let pomocí 1.5V AAA baterií díky spotřebě menší než 5  $\mu$ A při uspaném procesoru. Spotřeba stoupá při měření hodnot či odesílání dat, poté se procesor znova uspí.

Vývoj na toto hardwarové zařízení probíhá v jazyce C, pro jednoduchost je dostupné SDK<sup>6</sup> pro ovládání všech dostupných modulů a senzorů.

Při vývoji firmwaru je možno využít všeobecné rozšíření dostupné pro klasická IDE jako je například Visual Studio Code, s názvem PlatformIO. Pomocí tohoto rozšíření je možné překládat a nahrávat firmware.

## Výhody

Výhodou je komunikace přes sub-ghz rádio, které pro většinu domácností dostačuje a má dosah po celém domě. Díky této komunikaci není nutné jakoukoli jednotku připojovat přímo

<sup>6</sup>Dxygen dokumentace k HARDWARIO SDK: <https://sdk.hardwario.com>

k domácí místní síti. Spojení přes rádiový dongle přidává také další úroveň zabezpečení, jelikož zařízení nekomunikují přímo s použitým serverem nýbrž využívají dongle jako prostředníka.

### Srovnání mikrokontrolerů používaných v chytré domácnosti

	<b>Procesor</b>	<b>Programovací jazyk</b>	<b>IDE</b>	<b>Implicitní komunikace</b>	<b>Typy desek</b>	<b>Cena*</b>
<b>HARDWARIO TOWER</b>	ARM Cortex M0+	C	No	Sub-GHz Radio	One Core module	€36,30
<b>Arduino</b>	ATmega (various types)	C/C++	Yes	Nonewithout additional modules	Multiple sizes and types	€20.00

Tabulka 2.2: Porovnání probraných mikrokontrolerů

\* Cena se vztahuje na HARDWARIO Core Module vs Arduino Uno.

### Senzory a aktuátory chytré domácnosti

Na začátku této podkapitoly budou probrány technologie, pomocí kterých probíhá komunikace mezi externím zařízením, jako je senzor, a připojeným mikrokontrolerem.

V další části budou probrány využívané senzory a aktuátory používané v chytré domácnosti a jejich obecný princip fungování. U některých, pokud existuje více různých typů, budou probrány i tyto jednotlivé typy a principy jejich fungování. U každého senzoru budou dále zmíněny obecné možnosti jejich využití.

### Technologie pro komunikaci mezi mikrokontrolerem a externím zařízením

V předchozí kapitole 2.2 bylo u jednotlivých mikrokontrolerů zmíněno, jaké mají možnosti komunikace s externími zařízeními.

V IoT se pro tuto komunikaci a ovládání běžně používají tyto technologie:[25]

- $I^2C$
- SPI
- 1-wire
- UART
- GPIO

#### $I^2C$ (Inter-Integrated Circuit)

$I^2C$  je synchronní half-duplex multi-master sběrnice používající pouze 2 signálové vodiče pro přenos dat. Pro tento přenos se využívá vodič SDA pro data (half-duplex) a SCL pro synchronizační hodinový signál (synchronní sběrnice). Multi-master sběrnice umožňuje, aby

bylo slave zařízení(senzor) připojeno pomocí  $I^2C$  k více masterům, kteří mohou zahajovat komunikaci.

Komunikaci zahajuje vždy master směrem ke slave zařízení.  $I^2C$  paket umožňuje specifikovaní zápisu či čtení.

Informace byly převzaty a přeloženy z článku *Basics of the  $I^2C$  communication protocol*[6] a  $I^2C$ [20].

### SPI (Serial Peripheral Interface)

SPI je synchronní sériové full-duplex multi-master rozhraní využívající 4 signálové vodiče MISO (Master In Slave Out), MOSI (Master Out Slave In), SCK (Serial Clock), SS (State Select).

Komunikaci opět zahajuje master tím, že pomocí signálu SS zvolí, se kterým zařízením bude komunikovat. Synchronizovaně pomocí hodinového signálu probíhá komunikace oběma směry. Slave odesílá data po vodiči MISO masterovi a ten odesílá data po vodiči MOSI. Při každém vysílání si obě zařízení vyměňují data.

Informace byly převzaty a přeloženy z článku *Basics of the SPI communication protocol*[7].

### 1-wire

1-wire je asynchronní half-duplex sběrnice využívající pouze jeden signálový vodič a zemní vodič. Jako první ze zmíněných sběrnic se nejedná o multi-master sběrniči, tudíž je možné komunikovat pomocí jednoho vodiče pouze s jedním slave zařízením. Výhodou této sběrnice je možnost napájení externího zařízení pomocí signálového vodiče, po kterém jsou také přenášena data.

Informace byly převzaty a přeloženy z článku *Guide to 1-wire communication*[17].

### UART (Universal asynchronous receiver-transmitter)

UART je asynchronní master-slave full-duplex rozhraní používající dva signálové vodiče Rx (Recieve) a Tx (Transmit). Pro odesílání dat je na prvním zařízení využíván výstup Tx, který je připojen na druhém zařízení do vstupu Rx. Díky těmto vodičům můžou obě zařízení odesílat data současně, tudíž se jedná o full-duplex komunikaci.

Toto rozhraní je často používáno k nahrávání firmwarů do mikrokontrolerů. Další využití je pro sériový monitor, který je možné využít k ladění kódu pomocí odesílání dat z mikrokontroleru do připojeného PC.

Informace byly převzaty a přeloženy z článku *Basics of UART communication*[8].

### GPIO (General-Purpose Input/Output)

GPIO piny jsou piny, které jsou plně dostupné a nastavitelné programátorem firmwaru. GPIO pin je možné nastavit jako vstupní či výstupní.

Vstupní pin může přijímat digitální data pomocí nastavené napěťové úrovně. Tato úroveň se liší u různých mikrokontrolerů, ale většinou se jedná o 3V3 či 5V pro digitální 1 a 0V pro digitální 0.

Výstupní pin může vysílat stejně napěťové úrovně do připojených externích zařízení. Pomocí výstupního pinu je možné například rozsvítit LED či ovládat relé.

Informace byly převzaty a přeloženy z článku *What is GPIO?*[14].

## HVAC systém

HVAC neboli Heating, Ventilation and Air Conditioning je soustava několika komponent, které spolupracují, aby zajistily vyhovující podmínky v domácnosti[9]. Tento systém obsahuje různé senzory a také několik ovládacích prvků.

### Senzory

- Teplotní senzor
- Senzor vlhkosti
- $CO_2$  monitor
- VOC senzor

### Ovládací prvky

- Relé

### Teplotní senzor

Teplotní senzor je zařízení, které je schopné pomocí přímého kontaktu či bezkontaktně měřit teplotu a převádět ji na elektrické signály rozpoznatelné mikrokontrolerem.

Informace o principech fungování byly převzaty a přeloženy z článku *Temperature Sensors - Types, Working & Operation*[11].

### Principy

- **Termočlánek** - Tento typ teplotního senzoru je vytvořen spojením dvou různých kovů, které jsou spojeny na dvou koncích. Jeden z konců slouží jako referenční a druhý slouží k samotnému měření. Pokud je rozdíl v teplotách mezi těmito konci, tak dojde ke generování malého napětí. Toto napětí je možné detektovat a převést na teplotu.
- **RTD (Resistance Temperature Detector)** - RTD je zařízení jehož odpor se mění s teplotou. Obvykle je vytvořen z platinového pásku, ale je možné nalézt i zařízení z jiných materiálů či v jiných tvarech. Pro měření teploty je nutné přivést konstantní proud do platinového pásku a měřit napětí. Pomocí těchto dvou hodnot je možné získat odpor pásku. Pro zjištění teploty je nutné znát dostupnou tabulkou s hodnotami odporu při teplotách <sup>7</sup>.
- **Termistor** - Termistor je zařízení, které podobně jako RTD mění odpor vzhledem k teplotě. Rozdíl je v materiálu. Zatímco RTD je vyrobeno pomocí kovového pásku, termistor je vyroben pomocí polovodičových materiálů. Měření probíhá stejně jako u RTD. Pomocí termistorů je možné dosáhnout větší přesnosti měření.

### Využití

Teplotní senzor je možno využít pro regulaci teploty v domácnosti. Při spojení s relé je možno vytvořit termostat. Je možné jej využít i v jiných částech domácnosti, jako například pro monitorování otevřené lednice nebo pro výpočet rosného bodu. Díky monitoringu těchto hodnot je možné ušetřit energii a zlepšit životní podmínky v domácnosti.

<sup>7</sup>Příklad tabulky ukazující hodnoty odporu při různých teplotách pro platinový RTD: [https://www.pyromation.com/Downloads/Data/385\\_c.pdf](https://www.pyromation.com/Downloads/Data/385_c.pdf)

Správné typy, jako venkovní  $I^2C$  teplotní senzor DS18B20, je možné použít i pod vodou, pro měření teploty vody v bazénu nebo ve vodní nádrži.

### Příklady teplotních senzorů

- TMP112 (termistor)
- DS18B20



Obrázek 2.6: Klasický teplotní senzor TMP112. Foto HARDWARIO s.r.o.



Obrázek 2.7: Izolovaný teploměr pro měření například vody v bazénu. Foto HARDWARIO s.r.o.

### Senzor vlhkosti

Sensor vlhkosti je schopný měřit koncentraci vodních par ve vzduchu a tuto hodnotu prezentovat jako digitální data mikrokontroleru. Měření probíhá pomocí měření odporu či kapacity pomocí materiálů, které mění svoje vlastnosti se změnou vlhkosti.

Informace o principech fungování byly převzaty a přeloženy z článku *Humidity Sensor – Types and Working Principle*[1].

### Principy

- **Senzor vlhkosti s kapacitním měřením** - Senzor využívající kapacitní měření sestává z hygroskopického nevodivého materiálu, který je uzavřen mezi dvě elektrody. Díky této konstrukci je vytvořen kondenzátor. Hygroskopický materiál je materiál, který mění svoji nevodivou konstantu při změně vlhkosti. Tento princip znamená že při změně vlhkosti se změní i kapacita kondenzátoru. Tato hodnota značí relativní vlhkost vzduchu.
- **Senzor vlhkosti s odporovým měřením** - Tyto senzory jsou vytvořeny z materiálů s malým odporem, tento odpor se ovšem prudce mění se změnou vlhkosti. Materiál s malým odporem je umístěn přes dvě diody. Tyto diody jsou umístěny inter digitálně pro zvýšení kontaktní plochy. Odpor mezi těmito diodami se mění s vlhkostí díky vlastnostem použitého materiálu. Tato změna odporu je měřitelná a značí relativní vlhkost v ovzduší.

## Využití

Při vysoké relativní vlhkosti nad 70% se zvyšuje riziko plísni, rezivění vybavení či rosení na studených površích. V horších případech může vysoká relativní vlhkost způsobit i hypotermii vzhledem k znemožnění přirozeného odvádění tepla z těla.

Naopak nízká vlhkost může způsobovat pálení očí, podráždění dýchacího ústrojí, prašnost.

Vlhkost je další nutná veličina pro výpočet rosného bodu vzduchu.

Na základě naměřených dat je možné regulovat vlhkost pomocí větrání, klimatizace či zvlhčovačů na požadované hodnoty.

Lidem příjemné hodnoty jsou mezi 40-60% při teplotách 22-27 °C.



Obrázek 2.8: HARDWARIO Humidity Tag s vlhkoměrem SHT20. Foto HARDWARIO s.r.o.

## $CO_2$ monitor

$CO_2$  monitor či senzor je typ senzoru, který měří koncentraci  $CO_2$  v ovzduší většinou v jednotkách ppm, tato data jsou poslána v digitální podobě mikrokontroleru. Pro měření se využívá několika principů, každý z těchto principů má jiné využití v praxi.

Informace o principech fungování byly převzaty a přeloženy z článku *CO<sub>2</sub> Sensors: Which Type Should You Be Looking For?*[21].

### Principy

- **NDIR (Non-Dispersive Infrared) senzor** - Tento typ senzorů využívá vlastnosti pohlcování některých typů a vlnových délek světla. Tuto vlastnost mají všechny obějky, molekuly i atomy. Vzduch vstoupí do senzoru. Z jedné strany je vysláno světlo o vlnové délce pro  $CO_2$ , která je většinou kolem čtyř mikronů. Druhý konec senzoru obsahuje detektory světla, které změří intenzitu proniknutého světla. Čím více  $CO_2$  ve vzduchu, tím více světla bude pohlceno.

Senzor může fungovat velice dlouhou dobu (někdy i 10 let a více). Měření není ovlivněno dalšími látkami přítomnými ve vzduchu. Je vhodný pro běžné koncentrace  $CO_2$  v domácnostech (kolem 1000 ppm).

Nevýhodou je, že měření může být ovlivněno vlhkostí a teplotou.

- **Electromechanical sensor** - Tento senzor využívá měření elektrického proudu či vodivosti pro zjištění koncentrace  $CO_2$ . Při vstupu  $CO_2$  do senzoru dojde uvnitř k chemické reakci. Podle typu senzoru dojde při chemické reakci k měření proudu, změně stávajícího proudu nebo ke změně vodivosti senzoru. Následně je tato změna a změřená hodnota použita k výpočtu koncentrace  $CO_2$ .

Elektromechanický senzor je méně náchylný na ovlivnění teplotou a vlhkostí než předchozí senzor.

Oproti předchozímu senzoru ovšem nemá tak dlouhou výdrž a měření může být ovlivněno jinými látkami ve vzduchu. Další nevýhodou je možná ztráta přesnosti.

- **MOS (Metal Oxide Semiconductor)** - Senzor využívající změnu odporu kovových sloučenin pomocí změny chemického složení těchto sloučenin. MOS senzory mohou měřit i jiné plyny v ovzduší, jelikož různé sloučeniny reagují s různě s odlišnými plyny. V senzoru se nachází odhalená část kovového pásku. Skrz tento pásek prochází konstantní proud. Při vstupu měřeného plynu do senzoru dojde k chemické reakci s odhalenou částí pásku. Tato reakce zvýší odpor či vodivost kovového pásku. Tato změna je detekována a velikost změny určuje koncentraci zkoumaného plynu v ovzduší.

Design senzoru je velice jednoduchý, a tudíž je snadné jej použít.

Stejně jako senzor typu NDIR může být měření ovlivněno teplotou, vlhkostí a dalšími látkami nacházejícími se ve vzduchu. Běžné využití je v prostorách, kde je vyšší koncentrace  $CO_2$  (2000 a více).

## Využití

Vliv  $CO_2$  na produktivitu a zdraví je prokázán několika studiemi. Díky jednoduchému monitoringu  $CO_2$  pomocí senzorů je možné připojit ventilaci, přidat do prostor s větší koncentrací  $CO_2$  rostliny či zvážit omezení počtu lidí v daných prostorách.

### Úroveň $CO_2$ a vliv na zdraví (ppm = parts per milion) [12]

- 350-400 ppm - úroveň venkovního prostředí
- do 1000 ppm - doporučená úroveň  $CO_2$  ve vnitřních prostorách
- 1200-1500 ppm - doporučená maximální úroveň  $CO_2$  ve vnitřních prostorách
- 1000-2000 ppm - nastávají příznaky únavy a snižování koncentrace
- 2000-5000 ppm - nastávají možné bolesti hlavy
- 5000 ppm - maximální bezpečná koncentrace bez zdravotních rizik
- 5000 ppm - nevolnost a zvýšený tep
- 15000 ppm - dýchací potíže
- 40000 ppm - možná ztráta vědomí



Obrázek 2.9: HARDWARIO  $CO_2$  Module se senzorem LP8. Foto HARDWARIO s.r.o.

### VOC senzor

VOC senzor je senzor, který měří koncentraci těkavých organických látek v ovzduší. Tyto látky jsou vyloučovány různými zdroji jako jsou deodoranty, při vaření, tisku či kopírování apod.

Krátkodobé vystavení vyšší koncentraci těchto látek může vést k nepříjemným zdravotním problémům jako podráždění očí a dýchacích cest nebo bolest hlavy. Delší vystavení může způsobit i těžké zdravotní potíže jako poškození jater, ledvin či centrálního nervového systému.<sup>[13]</sup>

### Úroveň VOC (ppb = parts per billion)<sup>[22]</sup>

- 0–65 ppb - bezpečná úroveň
- 65–220 ppb - mírně zvýšená úroveň
- 220–660 ppb - zvýšená úroveň, maximální doba vystavení kolem 12 měsíců
- 660–2200 ppb - mírně nebezpečná úroveň, maximální doba vystavení v rámci jednoho měsíce
- 2200–5500 ppb - nebezpečná úroveň, maximální doba vystavení je jedna hodina

Informace o principech fungování byly převzaty a přeloženy z článku *How to Measure Volatile Organic Compounds In the Air*<sup>[18]</sup>.

### Principy

- **PID (Photoionization detector)** - Foto ionizační detektor využívá ultrafialové světlo pro rozložení těkavých organických látek ve vzduchu na pozitivní či negativní ionty. Po rozložení je možné změřit náboj ionizovaného plynu. Tímto je získána hodnota VOC.
- **FID (Flame ionization detector)** - Tento typ senzoru vystaví vzorek měřeného plynu vodíkovému plamenu. Toto vystavení způsobí, že uhlovodíky ve vzorku začnou produkovat ionty. Tyto ionty jsou poté detekovány detektorem kovů.

- **MOS (Metal Oxide Semiconductor sensors)** - MOS senzor je schopný detektovat velkou škálu plynů. Senzor využívá citlivý film, který je schopný odeslat signál při dosažení nebezpečné toxické úrovně.

### Využití

Hodnota VOC se častěji měří v industriálním prostředí, ale i v domácnosti se mohou objevit různé zdroje, jako například čistící prostředky, barva, izolační pěna, 3D či obyčejný tisk

Doporučené místo pro monitoring VOC by mohla být garáž či dílna. V těchto místech se často vyskytují zdroje těkavých látek. Díky monitoringu je možné snížit vystavení na minimum pomocí ventilace nebo opuštění pracoviště, dokud nebude úroveň opět přijatelná a bezpečná.



Obrázek 2.10: HARDWARIO VOC Tag se senzorem SGP30. Foto HARDWARIO s.r.o.

### Relé

Relé je elektromechanické zařízení, které slouží ke spojování či rozpojování elektrických spojení. Relé v podstatě plní funkci klasického spínače, jehož stav je možné změnit elektrickým signálem.

### Principy

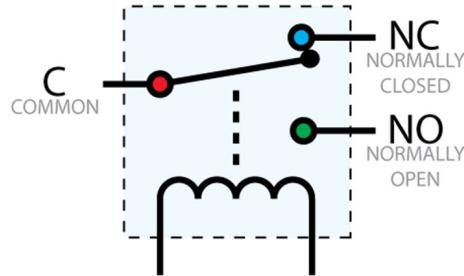
- **Elektro termální relé** - Toto relé využívá spojení dvou odlišných materiálů do bimetalového pásku. Přivedení napětí na tento pásek způsobí jeho ohnutí. Zařízení je vytvořeno způsobem, kdy ohnutí bimetalového pásku způsobí spojení kontaktů relé.
- **Elektromechanické relé** - Relé využívající mechanických součástí. Při přivedení napětí na elektromagnet je provedeno spojení či rozpojení okruhu.
- **Solid State relé** - Tento typ relé, na rozdíl od předchozích dvou, využívá polovodičové zařízení místo mechanických částí. Spínání díky tomu probíhá mnohem rychleji a zařízení má mnohem delší životnost.

### Využití

Velice užitečný prvek chytré domácnosti a jakékoli automatizace, který umožňuje ovládání a

kontrolu většiny zařízení pomocí spínání a rozpínání okruhů. Pomocí relé je možné simulovaltlačítka nebo zapínat a vypínat zařízení díky přivedení napájení.

Všechny výše zmíněné senzory mohou komunikovat s relé, které spouští například vytápění, ventilaci nebo klimatizaci.



Obrázek 2.11: Schéma jednoduchého relé. Foto [www.bambusekd.cz](http://www.bambusekd.cz) [5]

Na obrázku můžete vidět nákres běžného relé se dvěma okruhy. Do vstupu C je přivedeno napájení například z elektrické zásuvky, konkrétně fázový vodič. Z výstupu NO vede opět fázový vodič a dokud není do relé přivedeno spínací napětí, například 5V, tak je okruh rozpojen. Po sepnutí je na zařízení přivedeno napětí a za předpokladu všech ostatních vodičů připojených správně k zařízení, je zařízení napájeno.

### Ostatní hardware prvky chytré domácnosti

Mimo HVAC systém a jeho prvky existuje i mnoho dalších senzorů a modulů.

- Pohybový senzor
- Infra senzor
- Senzor svítivosti
- IP kamera
- Rotační enkodér

#### Pohybový senzor

Nejčastěji se jedná o takzvaná PIR(Pasive InfraRed) čidla neboli pasivní infračervené čidlo, která detekují pohyb pomocí infračerveného záření, které vyzařují všechny objekty, které mají nějakou teplotu. Samotný detektor žádné záření nevysílá, pouze je schopný detektovat různé úrovně infračerveného záření pomocí pyroelektrického senzoru.

#### Využití

Tyto senzory se používají v alarmech, bezpečnostních kamerech nebo automaticky spínáných světlech. S pomocí mikrokontroleru je možno informaci o pohybu odeslat a zpracovat do sepnutí mnoha dalších komponent, jako například vytápění, pokud je někdo v místnosti, či notifikace se zapnutím přenosu z kamery pro detailnější pohled na to, kdo je v místnosti přítomen.



Obrázek 2.12: Ukázka pohybového senzoru. Foto HARDWARIO s.r.o.

### **Infra senzor**

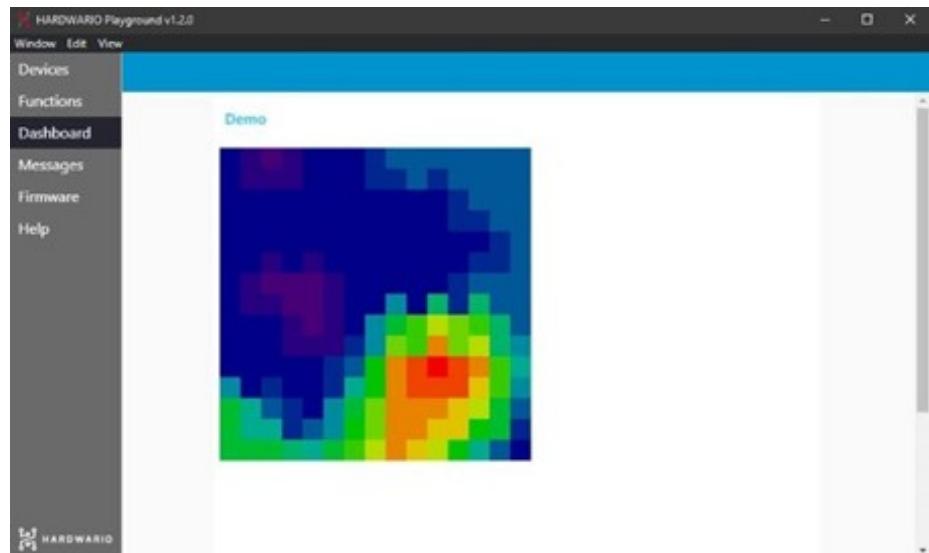
Tyto senzory jsou velice podobné předchozímu [2.2](#), s rozdílem, že je možné přesně zjistit teplotu na jednotlivých segmentech senzoru.

Dostupná jsou různá rozlišení pro tyto senzory, například s 64 senzory v mřížce 8x8. Pro přesnější měření je samozřejmě lepší co největší rozlišení, které však zvyšuje i cenu produktu.

### **Využití**

V době psaní práce a situace s onemocněním SARS-CoV-2 tyto senzory získaly na popularitě z důvodu bezdotykového měření teploty pomocí jednoduchého displeje s instrukcemi.

Samozřejmě je tento senzor možné využít pro detekci osob, ovšem pro tento případ užití je lepší dříve zmíněné PIR čidlo.



Obrázek 2.13: Ukázka výstupu z 8x8 infragrid senzoru(můžete vidět obrys obličeje). Foto autor

### Sensor intenzity světla

Někdy také nazývaný LUX senzor podle jednotky, kterou senzor měří, a to je jednotka intenzity světla.

### Využití

Pomocí tohoto senzoru je možné ovládat například světla v domě či stahování žaluzií.

Zařízení měřící intenzitu světla může být umístěno například na balkón kdy při vyšších hodnotách intenzity světla jsou během dne žaluzie vytaženy a světla zhasnuty kvůli snížení spotřeby elektrické energie s využitím přirozeného světla. Naopak při snížení intenzity slunečního svitu jsou světla postupně rozsvěcována, aby byla udržena stejná komfortní viditelnost.



Obrázek 2.14: HARDWARIO Lux Meter Tag se senzorem OPT3001. Foto HARDWARIO s.r.o.

## IP kamera

Kamery jsou využívány jako bezpečnostní či dozorovací prvek. Díky IP kamerám je možné zobrazovat záznam na jiném zařízení pomocí Ethernetu či Wi-Fi. Některé dohledové systémy podporují zobrazování IP kamer. Většinou je nutný server pro tyto kamery, kde se data shromažďují.

## Rotační enkodér

Pomocí rotačního enkodéru je možné nastavit například bezdrátově intenzitu světla na LED pásku, a to pomocí jednoduché otočné hřídele. Tyto enkodéry jsou využitelné v jakémkoli případě, kde je nutné nastavovat nějakou hodnotu po menších krocích v obou směrech.

Některé enkodéry, jako například ten uvedený na obrázku, umožňují i funkci stisku či podržení tlačítka, tím se jejich využitelnost zvyšuje, jelikož je možné reagovat i na tyto rozšiřující akce.



Obrázek 2.15: Ukázka rotačního enkodéru. Foto HARDWARIO s.r.o.

## Zařízení komunikující pomocí protokolu Zigbee

Komunikační protokol Zigbee byl vytvořen pro spojení a ovládání hardwarových zařízení pro chytrou domácnost. Díky tomuto protokolu je možné spojit více zařízení od různých výrobců, pokud podporují komunikaci právě pomocí protokolu Zigbee.

Pro spojení s dalšími zařízeními, která nekomunikují pomocí Zigbee, může být použito například programovací prostředí Node-RED. Pro toto spojení je ovšem většinou nutné pořídit i Hub, který bude přepošílat informace o zařízeních dále právě do prostředí Node-RED. Tyto Huby jsou často nutné i pro ovládání koncových zařízení pomocí mobilních aplikací výrobců.

Jako příklad lze uvést chytré osvětlení od firmy IKEA, kde je možné osvětlení ovládat pomocí fyzických bezdrátových spínačů, ale pro ovládání pomocí mobilní aplikace a spojení s dalšími zařízeními je nutné pořídit Hub od stejné společnosti, který následně slouží jako prostředník mezi koncovými zařízeními a mobilní aplikací či nějakým dalším ovládáním.

Zařízení podporujících protokol Zigbee je několik a zahrnují různé využití jako vytápění, osvětlení a další.

## Příklad hardwarových zařízení:

- **Chytrá termo hlavice Forseti eValve** - slouží k chytrému vytápění každé místnosti zvlášť pomocí měření teploty přímo na hlavici a automatické regulace
- **Ikea Tradfri** - tento název zahrnuje všechny prvky chytrého osvětlení od společnosti IKEA, jako jsou ovladače a samotné žárovky
- **Chytré zámky Yale** - souprava zabezpečovacích prvků na dveře otevíratelných pomocí kódu nebo čipu
- **Zabezpečovací systémy Bosch** - různé zabezpečovací zařízení jako jsou alarmy apod.
- **A další...**

## 2.3 Další software používaný v chytré domácnosti

### Protokol MQTT

MQTT (Message Queuing Telemetry Transport), je jednoduchý komunikační protokol používaný v IoT, a tedy i chytré domácnosti, komunikující pomocí subscribe/publish systému.

Tento protokol je široce využíván v automatizacích továren a domácností pro přenos dat ze senzorů spolu s možností ovládání aktuátorů jako jsou relé.

Některé další využití je například u platforem Messenger a Instagram od firmy Facebook, kde je využíván pro odesílání a přijímání zpráv.

Zprávy v tomto protokolu jsou složeny ze dvou částí, které se nazývají topic (téma) a payload.

#### Topic

Adresa, která označuje danou zprávu. Může se jednat o jednoslovné označení, jako například temperature(teplota) nebo humidity(vlhkost) tento postup ovšem není doporučený, jelikož není ve větších projektech udržitelný.[\[26\]](#)

Běžně jsou používány topicy obsahující více úrovní, tyto úrovně slouží jako jednoznačná identifikace hodnot v celém systému. Jednotlivé úrovně jsou odděleny běžným lomítkem.

Například *home/living-room/temperature* jednoznačně označuje zařízení které se nachází v domě v místnosti obývací pokoj a měří teplotu, pokud by například některé zařízení odesíalo více hodnot je možné přidat další úrovně, které by označovaly konkrétní zařízení. Maximální délka je 65536 znaků.

MQTT umožňuje také vnitřní monitoring stavu zařízení, které slouží jako Broker. K této informacím je možno se dostat pomocí speciálních topiců které začínají znakem \$ například \$SYS/broker/uptime pro zjištění, jak dlouho daný server běží bez výpadku.

#### Payload

Obsah odeslané zprávy, tato část nemusí být vždy přítomna

#### Příklady obsahu zprávy:

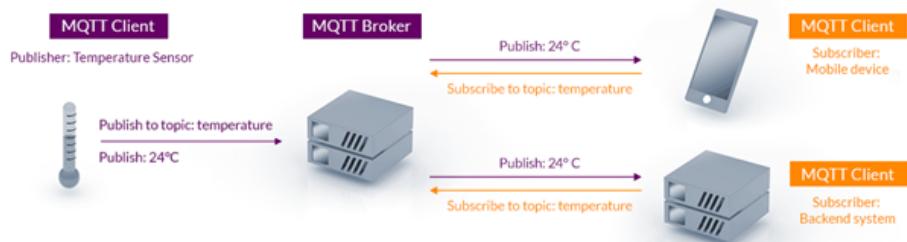
- Krátký text určující hodnotu ze senzoru jako je teplota
- Stav aktuátoru, zda je například relé, co řídí vytápění sepnuto
- Složitější struktury pomocí textu ve formátu JSON

- Bitové pole se zakódovaným obrázkem z kamery

Díky těmto možnostem by mělo být možné odeslat všechny potřebné informace pro ovládání a monitoring chytré domácnosti. Díky maximální délce až 268 435 456 znaků je možné odesílat i velice složité struktury obsahující mnoho informací.

## Princip

Při komunikaci pomocí MQTT protokolu je nutnou součástí sítě takzvaný MQTT Broker, který slouží jako prostředník pro všechny zprávy. Dále může existovat několik MQTT klientů, kteří mohou zprávy s topicem vysílat (Publish) a zároveň mohou očekávat zprávy na nějakém topicu (Subscribe).



Obrázek 2.16: Ukázka MQTT sítě. Foto mqtt.org [16]

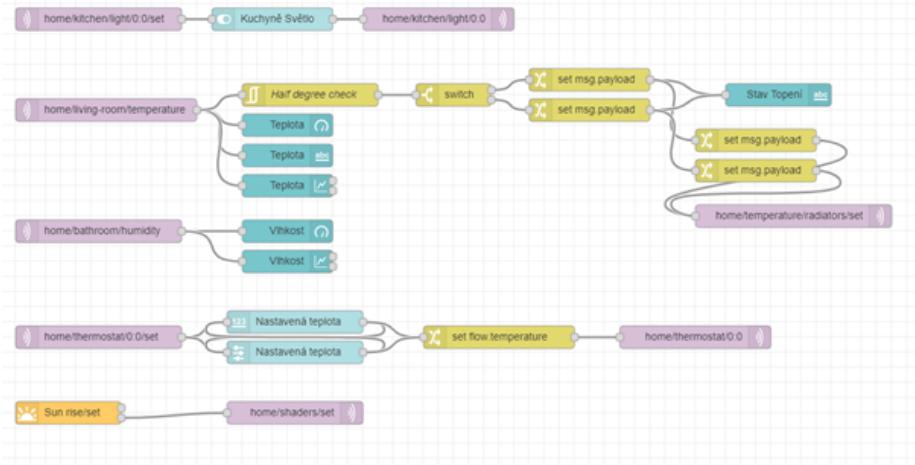
Na Obrázku 2.16 je jednoduchý nákres sítě využívající protokol MQTT. V této síti se nachází jeden Broker, na který jsou připojeni 3 klienti. Jeden z těchto klientů (teploměr vlevo) odesílá svoji teplotu v payloadu zprávy na topicu temperature. Dále na pravé straně jsou 2 klienti, kteří poslouchají zprávy na topicu temperature. Jakmile je odeslána teplota z teploměru, Broker ji přijme a rozešle ji na všechny klienty, kteří mají o tuto zprávu zájem. Pokud by se v síti nacházeli i klienti, kteří neodebírají zprávy na topicu temperature, zpráva jim nebude přeposlána.

## Node-RED

Toto programovací prostředí, které slouží k propojování hardwarových zařízení mezi sebou, stejně jako s dalšími API a online službami, je využíváno v oblasti chytré domácnosti a jiných automatizací. Jeden z hlavních důvodů, proč tomu tak je, je využívaný způsob programování. Celou domácnost je možné propojit pomocí takzvaných nodů, tyto nody jsou poměrně široce upravitelné a díky komunitním rozšířením existuje již mnoho dalších integrací včetně v této práci používaného IKEA Smart Home<sup>8</sup>.

V případě, že nestačí jakýkoli dostupný node může být využit programovatelný function node do kterého je možné vložit vlastní kód napsaný v jazyce JavaScript.

<sup>8</sup>Stránka knihovny použité v této práci: <https://flows.nodered.org/node/@fetchbot/node-red-contrib-ikea-home-smart>

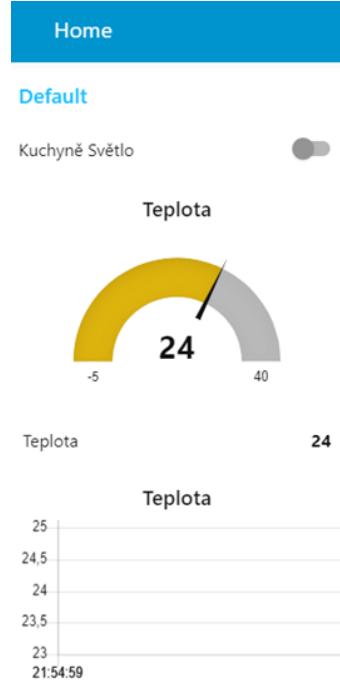


Obrázek 2.17: Ukázka Node-RED flow. Foto autor

Na obrázku 2.20 je ukázka jednoduchého Node-RED flow s několika propojeními a ovládáním. Díky možnosti jednoduchého sdílení je možné si toto a jakékoli jiné flow vložit a upravit dle vlastních potřeb, pokud znáte konfigurační JSON.

Další výhodou systému Node-RED je možnost integrovaného Dashboardu pro zobrazování hodnotových polí, grafů apod. Opět existuje několik rozšíření, která přidávají další prvky pro Dashboard.<sup>9</sup>

V případě potřeby vlastního rozšíření je možné si pomocí HTML a JavaScript takové rozšíření vytvořit.



Obrázek 2.18: Ukázka Node-RED Dashboard. Foto autor

<sup>9</sup>Seznam Node-RED rozšíření: <https://flows.nodered.org/search?term=dashboard>

## Databáze

Databáze ukládá data o chytré domácnosti, která je následně možné zobrazit či upravit v různých aplikacích. Tato data jsou důležitá z důvodu dlouhodobé analýzy dat či kontrolování senzorů a zařízení.

Existuje několik typů databázových systémů. Každý druh má své výhody a nevýhody.

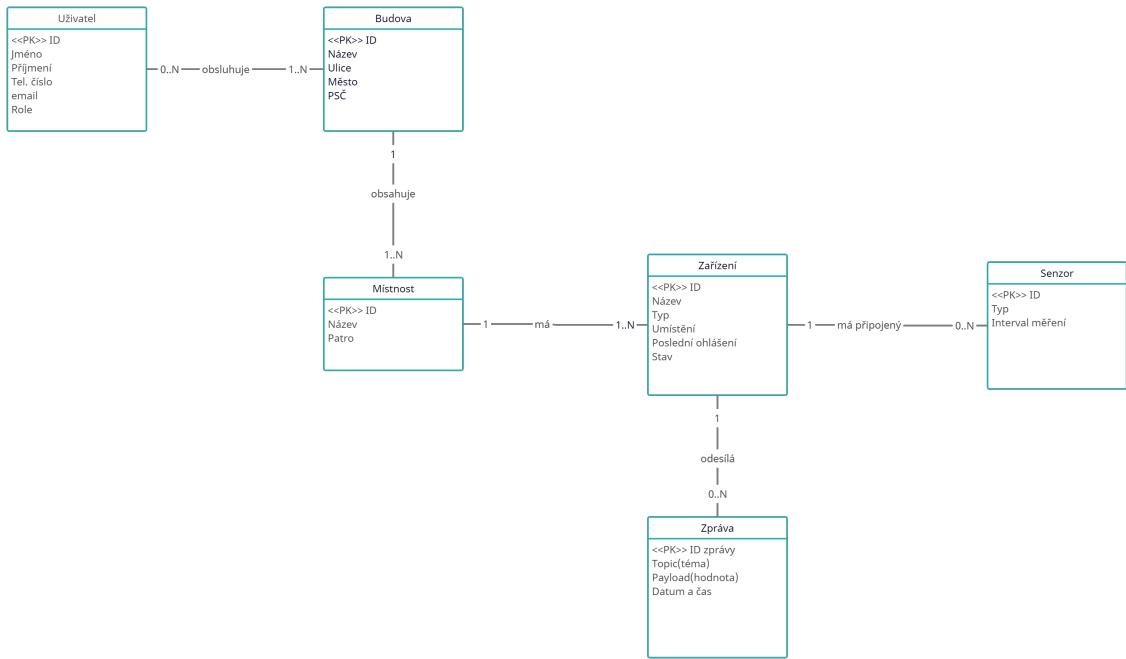
### Seznam běžně používaných databázových systémů[19]

- Hierarchické databáze
- Sítové databáze
- Relační databáze
- Objektově orientované databáze
- Grafové databáze
- ER model databáze
- Dokumentové databáze
- NoSQL databáze

Pro IoT a chytrou domácnost jsou často používány takzvané Time series databáze. Tyto databáze umožňují jednoduchou agregaci dat v čase pro rychlé zobrazení dat například v grafech. Jako příklad tohoto druhu databáze je možné uvést InfluxDB. Tato databáze se řadí do typu NoSQL tudíž pro databázové dotazy používá vlastní jazyk podobný běžnému SQL.

V rámci této práce při vytváření vzorového systému byla použita databáze MySQL z důvodu menšího objemu dat a jednoduchosti.

## Návrh databáze pro chytrou domácnost



Obrázek 2.19: ERD pro chytrou domácnost. Foto autor

Na Obrázku 2.19 je možné vidět vzorový ER diagram pro chytrou domácnost.

Tento diagram zahrnuje několik budov s různými místnostmi. V těchto místnostech se nachází zařízení různých typů jako je kamera, Arduino či HARDWARIO TOWER, která odesílají zprávy s daty z připojených senzorů.

Díky této struktuře databáze je možné provádět i správu jednotlivých zařízení a senzorů.

## Mobilní aplikace

Pro možnost ovládání a monitorování chytré domácnosti i mimo domov či při absenci počítače se využívají mobilní aplikace, které běžně poskytují jednoduché dashboardy s grafy, texty apod. Díky těmto aplikacím může uživatel domácnost sledovat a ovládat odkudkoli a kdykoli.

Mobilní aplikace dále poskytují rychlé notifikace uživateli i bez nutnosti odesílání SMS či emailů.

Možnosti mobilních aplikací je několik a většinou záleží na preferenci uživatele.

Pro tuto práci byla zvolena mobilní aplikace Blynk.

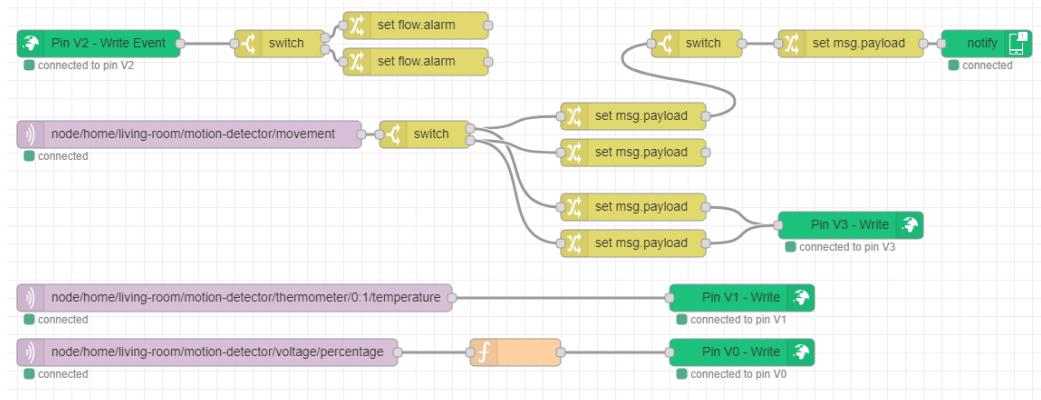
## Blynk

Platforma Blynk umožňuje rychle vytvářet kvalitní a přehledné dashboardy pro dohled a řízení chytré domácnosti a ostatních IoT zařízení pomocí drag-and-drop systému přímo na mobilním zařízení. Takto vytvořený dashboard se pomocí QR kódu dá sdílet s mnoha uživateli, jakmile je sdílen tak se jakákoli úprava promítne do všech zařízení.

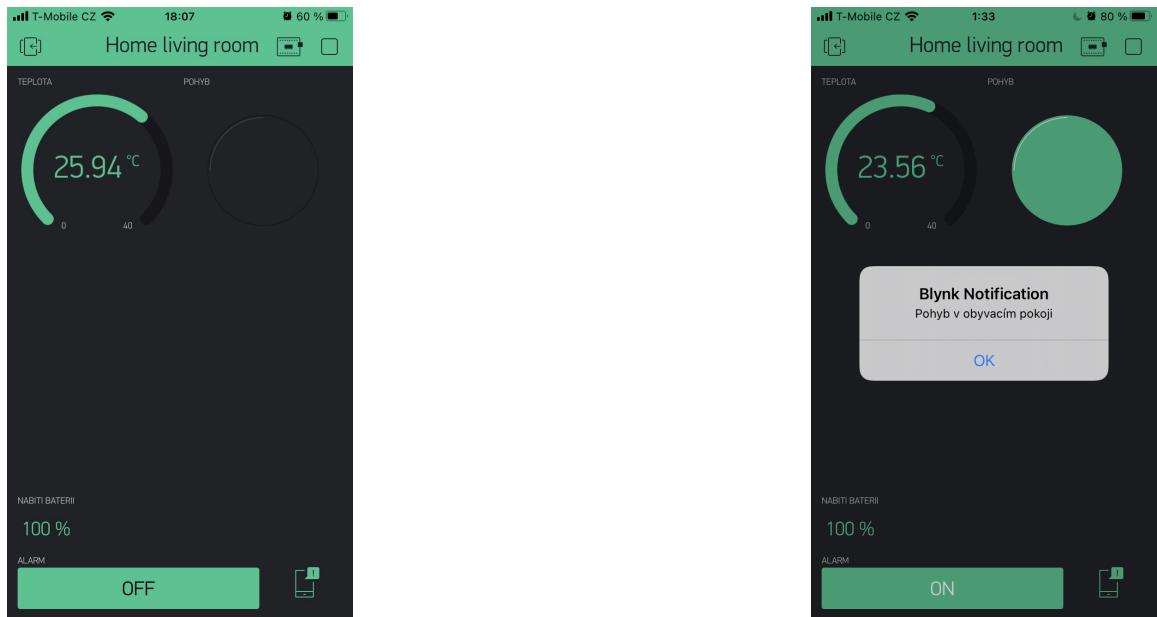
Blynk server je možné spustit na lokálním počítači jako je Raspberry Pi a zpřístupnit například přes veřejnou IP adresu nebo VPN.

Pro jednodušší použití lze využít cloudové servery přímo od společnosti Blynk. V případě použití cloudových serverů je zajištěna bezproblémová dostupnost a rychlé nasazení. Tyto výhody jsou vykoupeny tím, že každý umístěný widget je nutno zaplatit tzv. energií, která se dobíjí pomocí peněz v aplikaci.

Pro propojení s Node-RED je dostupné rozšíření obsahující všechny potřebné nody. Po vytvoření projektu v aplikaci pod svým uživatelským účtem a vygenerováním kódu stačí nastavit v Node-RED k jakémukoli projektu daný node patří pomocí adresy serveru a právě tohoto vygenerovaného kódu.



Obrázek 2.20: Ukázka Node-RED Dashboard s integrací Blynk nodů. Foto autor



Obrázek 2.21: Vzorový dashboard v aplikaci Blynk. Foto autor

Obrázek 2.22: Ukázka notifikace v aplikaci Blynk. Foto autor

Na obrázku 2.20 je vidět jednoduché flow vytvořené v prostředí Node-RED, které je propojeno do dashboardu v aplikaci Blynk, který je vidět na obrázku 2.21. Hardwarová zařízení posílají data přes MQTT do Node-RED a tam se pomocí nodů přenáší do Blynk

dashboardu. Dále je možné z aplikace Blynk odesílat například stisk tlačítka pro zapnutí alarmu či sepnutí relé.

Tato vzorová aplikace sbírá údaje o teplotě a pohybu v místnosti, dále je ve spodní části možno vidět kolik procent baterie zbývá na fyzickém zařízení, aby bylo možno je včas vyměnit.

Díky sledování pohybu je možné tlačítkem zapnout alarm. Pokud je alarm zapnutý a je detekován pohyb tak bude na všechna mobilní zařízení, která používají tuto aplikaci, odeslána notifikace s textem *Pohyb v obývacím pokoji*.

### Výhody

- Rychlé nasazení
- Propojení s Node-RED
- Spousta widgetů, které je možno použít
- Možné rozdělení na několik obrazovek pro zpřehlednění

### Nevýhody

- Při použití cloudových serverů cena za widgety v aplikaci

## Kapitola 3

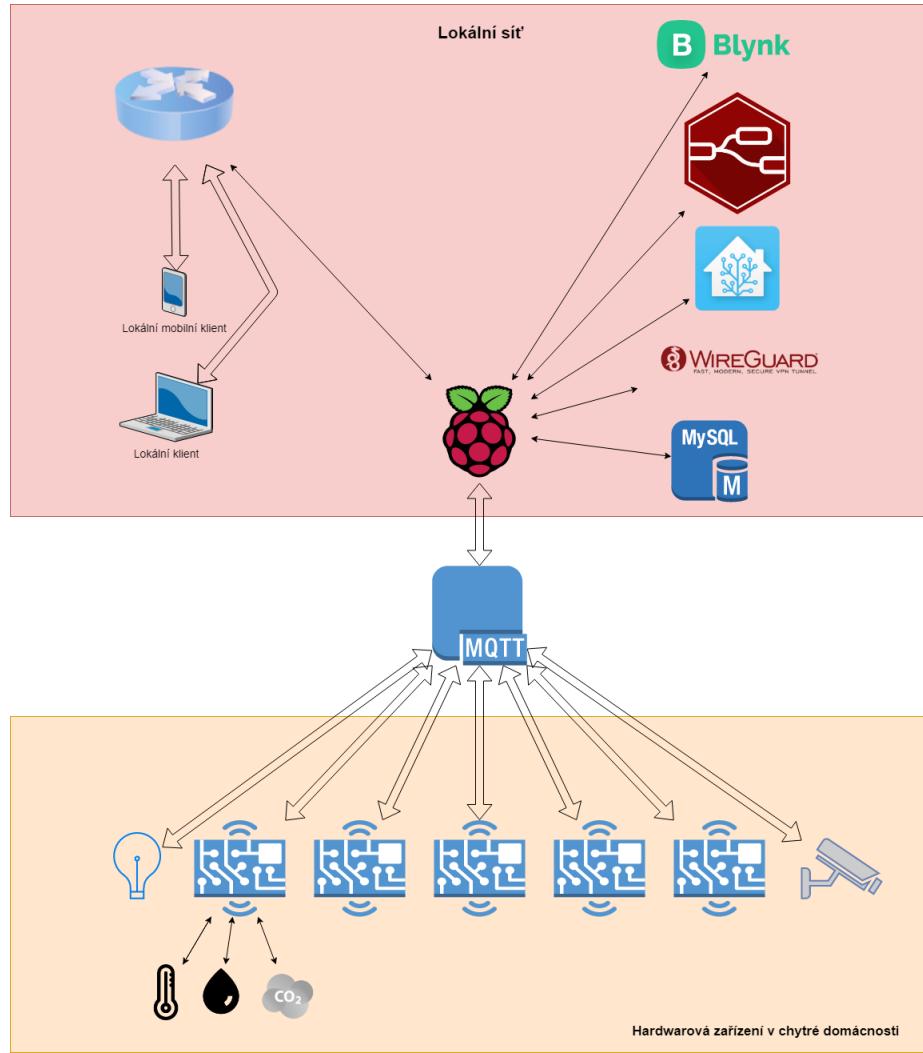
# Požadavky na dohledový a řídicí systém

Tato kapitola bude opět rozdělena do dvou částí, hardwarová 3.1 a softwarová 3.2.

V první části bude probráno, jaká zařízení, detektory a senzory jsou vhodné pro domácí automatizaci, jak tyto senzory použít, spojit mezi sebou a proč jsou pro chytrou domácnost důležité.

V druhé části budou určeny požadavky na dohledový software, který bude implementován v pozdějších částech této práce. Požadavky jsou stanoveny na základě dříve probraných systémů, jako například Home assistant, tak aby bylo spojeno co nejvíce výhod z těchto různých systémů.

Na závěr této kapitoly budou probrány možnosti přístupu do systému pro chytrou domácnost. Těchto možností existuje několik, v této kapitole budou zmíněny tři běžně používané.



Obrázek 3.1: Diagram systému chytré domácnosti. Foto autor

Na obrázku je možné vidět návrh celkového řešení, kde fyzická zařízení komunikují pomocí MQTT se serverem běžícím na Raspberry Pi, který spouští všechny potřebné služby.

Uživatelé v lokální síti dále komunikují se serverem pro zobrazování informací na klientském zařízení ve vytvořené desktopové aplikaci, mobilní aplikaci Blynk či ve webovém prohlížeči pomocí aplikace Home Assistant.

Pro komunikaci s chytrou domácností pomocí klientského zařízení mimo lokální síť bude možné využít VPN. Toto spojení bude popsáno v kapitole 3.3.

### 3.1 Hardwarové prvky a jejich využití v navrženém systému

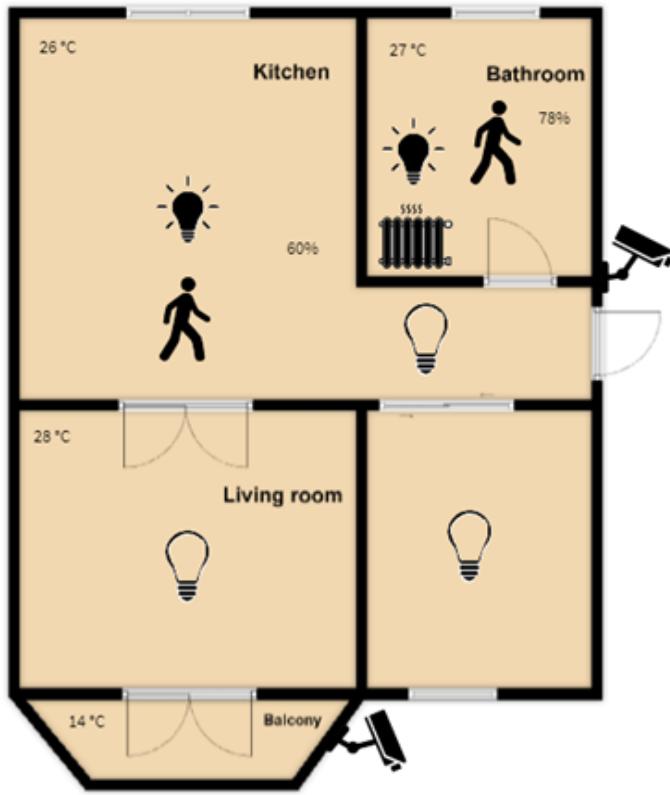
V této části budou probírány senzory, aktuátory a jejich využití v chytré domácnosti. Nebudou opět zmiňovány principy fungování, které jsou probírány v předchozí kapitole 2.2.

Vzorový systém vytvořený v rámci této práce bude obsahovat několik prvků zmíněných výše. Díky firmě HARDWARIO s.r.o. bude většina z nich fyzicky přítomna, aby bylo možno data sbírat co nejpřesněji. Některé další prvky budou simulované pomocí vytvořeného simulátoru 4.3.

## Použité prvky

- Jednoduchý termostat pro řízení vytápění
- Relé pro zapínání a vypínání kotle
- Teplovody pro orientační sběr teplot v domě
- Vlhkoměr pro zabránění vzniku plísni a případné pravidelné větrání
- Detektor  $CO_2$  pro příjemné a zdravé prostředí
- Detektor VOC do dílny pro kontrolu těkavých látek
- Soustava relé pro spínání různých komponent jako vytápění a ventilace
- Pohybové senzory pro detekci osob v jednotlivých místnostech a jako alarm proti vniknutí
- Sensor svítivosti pro změnu teploty světla žárovky a vytahování/zatahování žaluzií
- IP kamery u vstupů do domu
- Rotační enkodér na zvýšení či snížení intenzity světla u některých LED pásků či žárovek
- Chytrý zvonek s notifikacemi při příchodu
- Dálkově ovládaný bzučák na vstupních dveřích

Je samozřejmě možné použít mnohem více zařízení pro detekci několika dalších hodnot, některé z nich jsou zmíněny dříve u jednotlivých prvků. Toto je pouze návrh nejběžnějších použití u chytré domácnosti.



Obrázek 3.2: Ukázka několika senzorů a spínačů na nákresu bytu. Foto autor

Na obrázku 3.2 je přibližný nákres bytu s několika senzory, které jsou reprezentované hodnotami a několik ovládacích prvků reprezentovaných spínači, tyto spínače mají různé ikony jako radiátor a žárovka. Poslední ikonou je kamera, která dohlíží na vchod a verandu.

Tento návrh bude v následující kapitole 4.2 implementován jako vzorové řešení pro chytrou domácnost pomocí fyzických či simulovaných zařízení.

## 3.2 Software pro dohledový a řídicí systém

Aplikace bude implementována pomocí .NET Core v jazyce C#. Aplikace má působit jako vzorová implementace pro chytrou domácnost. Bude využit framework WPF, tento framework je pouze pro systémy s operačním systémem Windows. Pro multiplatformní aplikaci by bylo možné využít MAUI nebo Multi-platform App UI, který je ale v době psaní práce stále ve vývoji.

Při návrhu systému bylo vzato v potaz několik zkoumaných již existujících systémů.

Další část systému bude server, na kterém poběží programovací prostředí Node-RED, databáze MySQL, server Blynk a bude také sloužit jako MQTT Broker.

V Node-RED budou naprogramována spojení mezi jednotlivými prvky v domácnosti.

Do databáze budou ukládána data, která bude možno zobrazit v desktopové aplikaci.

Aplikace Blynk bude sloužit k jednoduchému monitorování a ovládání chytré domácnosti odkudkoli pomocí mobilního zařízení.

## Požadavky na aplikaci

V dohledové aplikaci je důležitá přehlednost a jednoduchost.

Jednoduchostí se rozumí jednoduchost celkového používání aplikace od editace prostředí, přes nastavení spojení s jednotlivými monitorovanými zařízeními až po interpretaci dat v jednoduché a srozumitelné formě.

Přehlednost je důležitá, a to z důvodu rychlého zorientování v aplikaci. Díky tomuto je možné před funkční a nastavené prostředí posadit v podstatě jakéhokoli uživatele, který bez větších technických znalostí bude schopný aplikaci ovládat a tím pádem ovládat domácnost. Uživatel s pokročilejší technickou znalostí by měl být bez větších obtíží schopný aplikaci i nastavit a provozovat.

S těmito požadavky na mysli byl navržen prototyp aplikace pro dohled a řízení. Její předběžný návrh je možno vidět v další kapitole v rámci softwarového řešení [4.3](#).

## 3.3 Přístup k chytré domácnosti a zabezpečení

V rámci chytré domácnosti jsou v síti dostupná velice citlivá data, která mohou být zneužita. Toto zneužití dat může vést až k vykrazení domu či bytu na základě vědomosti, že se v budově nikdo nenachází. Zjistit tuto skutečnost je možné podle teploty, dat z pohybových čidel či koncentrace  $CO_2$  ve vzduchu. Mezi další nebezpeční patří odemčení vstupních dveří, vysoké zvýšení teploty, ovládání aktuátorů po celém domě a další.

Základem kvalitního zabezpečení jsou silná a unikátní hesla pro přístup do každé služby. Pokud jsou služby v síti takto zabezpečené, je možné umožnit přístup i z jiné než lokální sítě a tím umožnit monitorování a ovládání i mimo domov.

Při monitoringu chytré domácnosti pouze v rámci lokální sítě, může být zabezpečení prvků poměrně slabé, za předpokladu, že je samotná síť kvalitně zabezpečena.

Takto zařízená chytrá domácnost má ovšem spoustu omezení a nebylo by ji možné ovládat či monitorovat ve chvíli kdy by byl uživatel mimo lokální síť. Existuje několik způsobů, jak umožnit přístup i z ostatních sítí.

Kvalitní zabezpečení znamená, že je možné provozovat všechny dříve zmíněné služby jako je Blynk server či Home Assistant bez nutnosti používání clouдовých řešení.

Většina služeb poskytuje právě zmíněné cloudové řešení, které garantuje zabezpečení a dostupnost.

Tato řešení jsou většinou dostupná pomocí buď jednorázových či opakovaných plateb, je ovšem možné je rychle nasadit a jednoduše provozovat. Jediné, co je nutné udělat je zařídit funkční hardware a připojení ke službě. Servery jako takové jsou ve vlastnictví a správě provozovatele služby.

## Přístup pomocí VPN

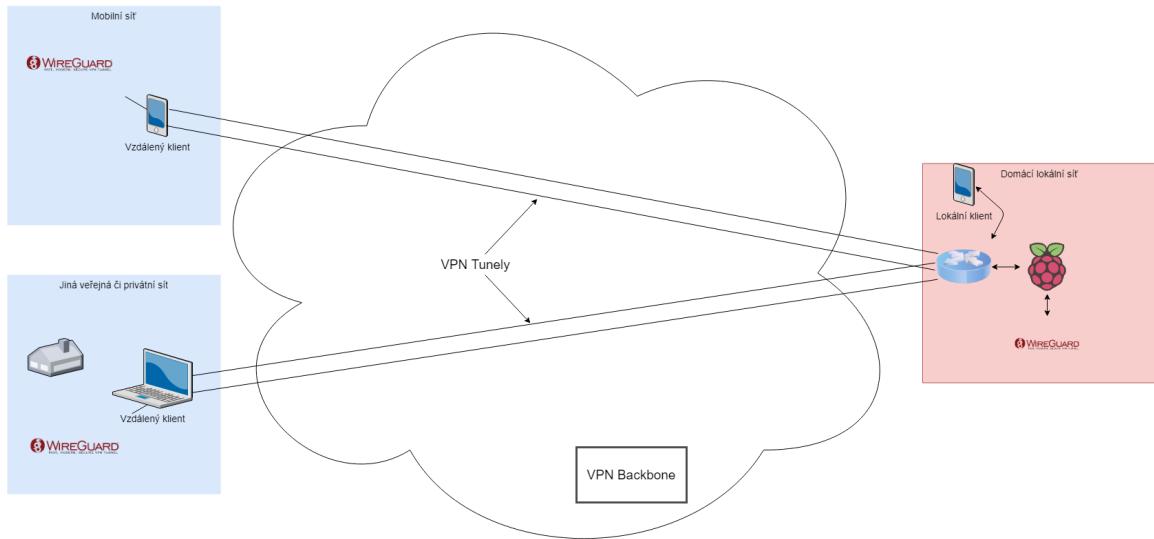
V této práci byl zvolen přístup pomocí VPN za pomocí protokolu WireGuard, který realizuje virtuální privátní síť na síťové vrstvě.

WireGuard server bude provozován opět na dříve zmíněném Raspberry Pi 3b. Aplikace WireGuard umožňující přístup do VPN je dostupná na všechna běžná mobilní zařízení i operační systémy.

Pro každé zařízení, které má mít přístup do lokální sítě, bude vygenerován profil s dvojicí klíčů veřejný a soukromý klíč. Tato data jsou v konfiguračním souboru nacházejícím se na Raspberry Pi, který provozuje VPN server. Konfigurační soubor stačí importovat do

aplikace spuštěné na konkrétním klientském zařízení a to je tímto připraveno na připojení do sítě pomocí VPN.

Díky tomuto spojení je následně možné se připojovat do všech služeb a aplikací, které běží v lokální síti i bez nutnosti otevřání portů na veřejné IP adresy domácího routeru.



Obrázek 3.3: VPN diagram. Foto autor

### Přístup přes veřejnou IP adresu

Přístup přes veřejnou IP umožňuje připojení k aplikacím a službám v domácí síti pomocí veřejné IP adresy domácí brány neboli routeru. Pro funkčnost tohoto přístupu je nutné přesměrovat porty na této veřejné adrese na IP adresu serveru a port, kde je spuštěná daná služba či aplikace.

Předpokladem pro toto spojení je nutnost velice kvalitního zabezpečení uživatelskými jmény a hesly na každé přístupné aplikaci. V případě tohoto řešení je možné, na rozdíl od předchozího řešení pomocí VPN, aby se do domácí sítě připojil kdokoli, kdo zná veřejnou IP a port, na který se připojit. V případě nekvalitního zabezpečení je tedy velice jednoduché se dostat k datům a ovládání chytré domácnosti.

# Kapitola 4

## Realizace řešení

V této kapitole bude popsána vlastní realizace hardwarového, softwarového a serverového řešení.

V serverovém řešení budou zmíněny všechny použité služby, k čemu v chytré domácnosti slouží a na jakém zařízení je možné je provozovat.

Hardwarová část řešení bude z větší části řešena pomocí stavebnice HARDWARIO TOWER, a to z důvodu možnosti neinvazivní realizace. Samozřejmě by bylo možné tyto firmwary vytvořit i na jiných zařízeních jako je například Arduino. Testování vytvořených firmwarů proběhne na fyzických zařízeních. Pro simulaci dalších zařízení, která nejsou fyzicky dostupná jako například kamery, bude využit vytvořený program v jazyce Python v kombinaci s PyQt5.

V softwarové části bude popsána implementace prototypové dohledové a řídicí desktopové aplikace, která bude později porovnána s již existujícím řešením Home Assistant. V aplikaci Home Assistant bylo vytvořeno několik dashboardů pro celý dům. Dále bude probrána implementace flow v programovacím prostředí Node-RED a několik vzorových projektů v aplikaci Blynk.

Dalším softwarovým prvkem vyvinutým v rámci školního projektu a této bakalářské práce je simulátor MQTT provozu, který je schopný odesílat a přijímat data různých typů (hodnoty, obrázky, JSON) přes MQTT.

Jedna část kapitoly softwarového řešení bude věnována možnostem přístupu k chytré domácnosti z domácí sítě a také z externích sítí včetně možností zabezpečení tohoto přístupu. Na konci této kapitoly bude zhodnoceno celé vytvořené řešení a srovnání využitých systémů. Dále budou zmíněna nějaká možná rozšíření celého dohledového a řídicího systému.

### 4.1 Serverové řešení

Velmi důležitá část chytré domácnosti je server, někdy také nazývaný Hub, na kterém jsou spuštěny všechny potřebné služby jako databáze, Node-RED, a který také slouží jako MQTT Broker.

Jako server je možné použít přechodně i počítač s operačním systémem Windows například ve fázi prototypování. Pro plné nasazení je ovšem dobré zvolit nějaký jednodeskový počítač s operačním systémem na bázi Linuxu.

Některé běžně používané servery jsou například Raspberry Pi či Synology NAS. Tyto servery běžně postačují k většině aplikací, mají nižší spotřebu a mohou být spuštěné skoro

nepřetržitě. Některé výše zmíněné systémy jako OpenHAB poskytují již připravený image systému.

Image systému je obraz vytvořený exportováním již funkčního systému. Tento systém je dále možné distribuovat vždy ve stejné podobě. Díky tomu mohou vývojáři aplikací jako je Home Assistant vytvořit tento image pro zařízení jako je Raspberry Pi, kde jsou nainstalované a otestované všechny nutné balíčky a nástroje. Uživatel si následně image pouze stáhne a použije jej jako operační systém svého serveru. Image je samozřejmě možné dále upravovat.

Pro tuto práci byl jako server zvolen jednodeskový počítač Raspberry Pi 3b, na který byl nahrán image hio-raspbian. Tento image byl použit pro zjednodušení práce s jednotkami složenými z kitu HARDWARIO TOWER. Samozřejmě by bylo možné všechny potřebné části nainstalovat jako balíčky pomocí instalačního scriptu dostupného na GitHubu<sup>1</sup>. Jako server by bylo možné použít i několik dalších zařízení. Doporučený je Linuxový operační systém.

#### Služby na serveru použité pro tuto práci:

- HARDWARIO Hub
- Mosquitto MQTT Broker
- Node-RED
- Blynk local server
- Home Assistant
- WireGuard

Většina těchto služeb byla již probrána podrobněji v předchozích kapitolách. Konkrétní využití každé z těchto služeb bude probráno v kapitole softwarového řešení. Služba WireGuard bude popsána v kapitole o zabezpečení.

Některé z těchto služeb mohou být provozovány pomocí cloudového řešení či veřejně dostupných služeb. Nevýhodou takového řešení je, že jsou data odesílána mimo domácí síť a velice často jsou takovéto služby také zpoplatněny.

## 4.2 Hardwarové řešení

V rámci této práce bylo vyvinuto několik firmwarů pro zařízení vytvořená ze stavebnice HARDWARIO TOWER, jako ukázky možnosti využití. Tyto firmwary jsou zaměřeny na běžný monitoring a ovládání chytré domácnosti. Postupně bude popsán každý z nich. Ukázky fyzických zařízení je možné vidět v příloze A.

Všechny firmwary byly vytvořeny ze šablony neboli takzvaného skeletonu dostupného od firmy HARDWARIO. Programovacím jazykem je jazyk C ovšem pro jednoduchost programování je možné použít API, ve kterém jsou dostupné funkce pro ovládání všech ve stavebnici dostupných senzorů a aktuátorů. Dále je možné pomocí tohoto API ovládat i další funkce mikrokontroleru jako jsou komunikace po sběrnicích apod.

Z důvodu open-source politiky této firmy a spoustě veřejně dostupných projektů je pravděpodobné, že si budou v této práci vytvořené firmwary podobné s některými již vytvořenými. Všechny funkční kódy v rámci souborů src/application.c byly vytvořeny od začátku pouze se základem z již zmíněné šablony neboli skeletonu.

---

<sup>1</sup>Odkaz na instalační skript: <https://github.com/hardwario/hio-raspbian/blob/master/install.sh>

Kromě HARDWARIO TOWER je využito i chytré osvětlení od společnosti IKEA, tato zařízení fungují pomocí technologie Zigbee. Pro spojení těchto zařízení s ostatními bude využito prostředí Node-RED, kde již existuje několik rozšíření pro IKEA Smart home, která obsahují všechny potřebné nody pro monitoring a ovládání.

## Postup vývoje firmwaru pro HARDWARIO TOWER

Pro základ každého firmwaru je vytvořen klon volně dostupné šablony twr-skeleton z GitHub. Díky této šabloně je firmware připraven na kompliaci i nahrání do zařízení Core Module.

Funkční kód se upravuje v souboru src/application.c. V tomto souboru je možné volně používat všechny funkce dostupné v API i bez dalšího upravování struktury. Dále je možné vytvořit kód pro komunikaci se senzory, které nejsou ve stavebnici dostupné. Hlavíčkové soubory jsou vkládány do složky include a zdrojové kódy do složky src.

Pro porozumění API je možné využít příkladů v dokumentaci či na GitHubu.<sup>2</sup>

## Detektor pohybu s měřením teploty

Jednoduché zařízení schopné detektovat pohyb pomocí digitálního PIR senzoru PYQ-1648-7053 a měřit teplotu pomocí teploměru TMP112, který je dostupný na každém Core Modulu v kitu HARDWARIO TOWER.

Pokud je registrován pohyb, tak je odeslána zpráva přes MQTT, při další detekci pohybu nejsou odesílány další zprávy pro snížení spotřeby baterie. Až pokud nedojde během jedné minuty k pohybu před senzorem, tak je odeslána zpráva ohlašující, že nebyl pohyb detekován.

Odesílání teploty probíhá v pravidelných intervalech 15 minut, pokud nedojde k jakémoli výrazné změně. Pokud dojde během těchto 15 minut ke změně o 0.2 °C v obou směrech, tak je odeslána změna okamžitě. Toto chování je stejné i u dalších firmwarů, které používají teploměr TMP112.

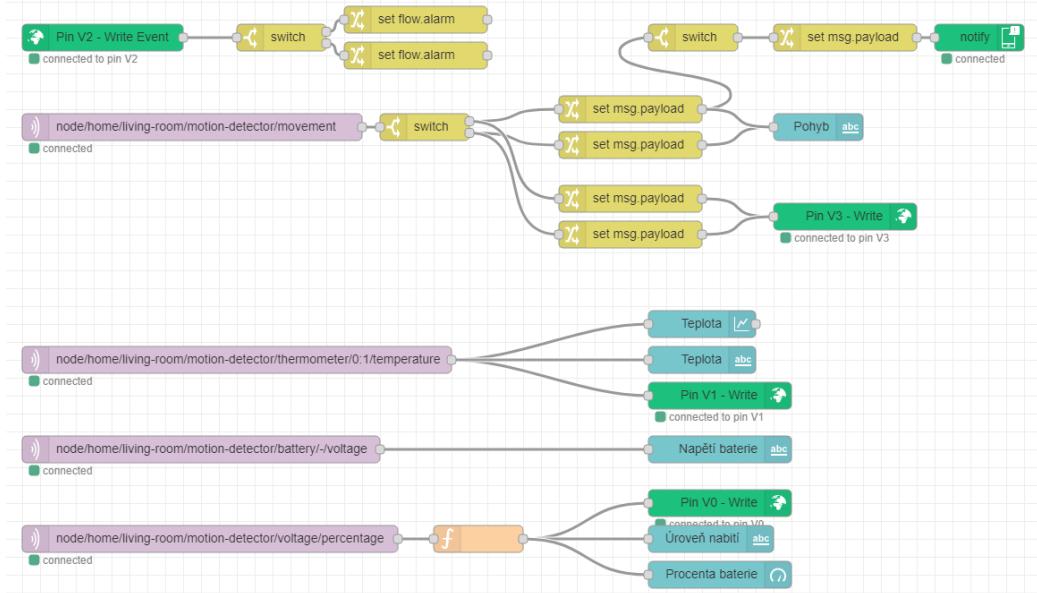
Další společnou vlastností zařízení s firmwary, které jsou schopné fungovat na bateriích, je odesílání stavu nabité baterií každých 60 minut. Díky tomuto je možné sledovat vzdáleně nabité baterií a například nastavit oznámení při klesnutí pod určitou hranici, aby mohly být baterie včas vyměněny.

Pomocí zařízení s tímto firmwarem je možno monitorovat teplotu na několika místech v domě a s jednoduchou úpravou v Node-RED a propojením do aplikace Blynk je možné pomocí stisknutí tlačítka zapnout funkci alarmu, kdy při detekovaném pohybu bude poslána notifikace na mobilní telefon.

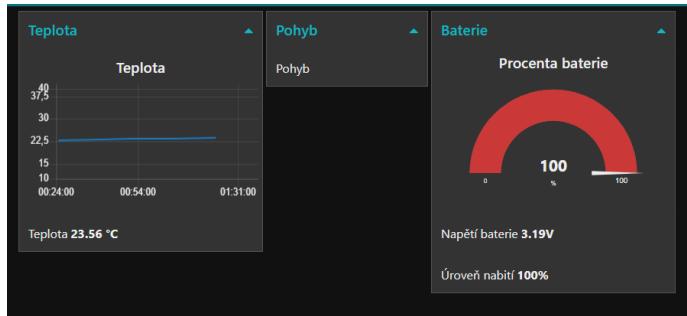
Takové zařízení je možné nasadit v podstatě do všech místností a tím získat jak zabezpečení, tak například průměrnou teplotu v celém domě a pomocí této hodnoty řídit centrální vytápění v domě, popřípadě vytápění v každé místnosti zvlášť, pokud je to možné.

---

<sup>2</sup>Příklady firmarů HARDWARIO: <https://github.com/search?p=2&q=org%3Ahardwario+twr-&type=Repositories>



Obrázek 4.1: Ukázka Node-RED Dashboard pro tento firmware. Foto autor



Obrázek 4.2: Ukázka Blynk Dashboard pro tento firmware. Foto autor

### Detektor pohybu s měřením klimatických podmínek

Toto zařízení je schopné, kromě vlastností zmíněných u předchozího firmwaru 4.2, také měřit podrobnější data o kvalitě životního prostředí v místnosti.

Pomocí měření vlhkosti, teploty a koncentrace  $CO_2$  je možné udržet velmi kvalitní podmínky v místnosti. Zařízení měřící tyto hodnoty je vhodné umístit například do ložnice, pracovny či jakékoli jiné místnosti, kde tráví lidé více času.

Pro měření  $CO_2$  je využit senzor LP8 od firmy SenseAir typu NDIR, jehož výhody jsou zmíněny v první části práce.

Vlhkost je měřena pomocí kapacitního senzoru SHT20, který komunikuje pomocí sběrnice  $I^2C$ .

Díky těmto údajům je možné například zapínat a vypínat topení, klimatizaci či odsávače vlhkosti k dosažení optimálních hodnot všech měřených veličin.

Díky Node-RED v kombinaci s Google kalendářem je možné vytvářet profily pro různé dny a hodiny v těchto dnech.

## Detektor kvality ovzduší s rotačním enkodérem a LED páskem

Toto o něco složitější zařízení umožňuje monitorování kvality prostředí. Jako základní hodnoty je schopné měřit teplotu, vlhkost a barometrický tlak.

Další měřenou veličinou je VOC (Volatile Organic Compound) měřená v ppm (parts per milion). Díky tomuto měření je zařízení vhodné do garáží a dílen, kde je možný únik nějakých nebezpečných těkavých látek do ovzduší, například při 3D tisku nebo při lakování a barvení.

Další možnou funkcí je připojení LED pásku, který lze využít k osvětlení pracovní plochy. V případě programovatelného RGBW LED pásku je možné pomocí rotačního enkodéru i ovládat úroveň svítivosti otáčením enkodéru či měnit efekty na LED pásku pomocí stisknutí. Při stisknutí a podržení enkodéru se LED pásek zapne či vypne.

Tato zařízení jsou napájená ze zásuvky, díky tomu je možno hodnoty ze senzorů odesílat častěji a popřípadě použít i relé ke spínání například větráku při vysoké koncentraci VOC.

LED pásek i relé je možné ovládat pomocí MQTT zpráv, pro zjednodušení ovládání byly vyvinuty dva jednoduché nody do prostředí Node-RED, tyto nody jsou popsány v softwarové části řešení.

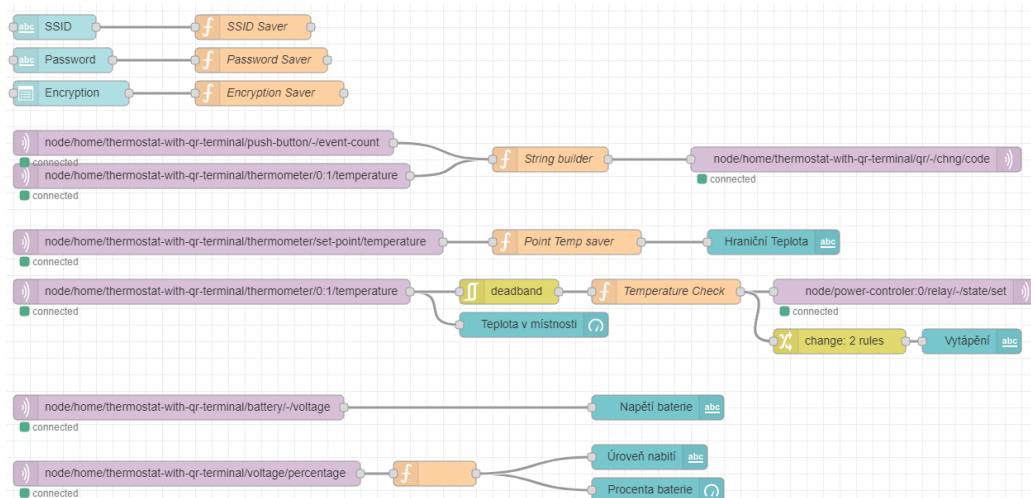
## Jednoduchý termostat s LCD displejem

Pro možnost nastavení požadované teploty v bytě či domě může sloužit toto zařízení. Jednoduchým stisknutím tlačítka je možno přidat či ubrat půl stupně z požadované teploty. Toto zařízení zároveň také měří teplotu pomocí TMP112, tudíž je možno jej používat samostatně i bez dalších teploměrů po domě.

Pro lepší výsledky je možné použít více teploměrů a počítat průměrnou teplotu.

Pomocí jednoduchého spojení v Node-RED je možné ovládat relé, které spíná v případě, že je teplota nižší než požadovaná teplota nastavená na termostatu.

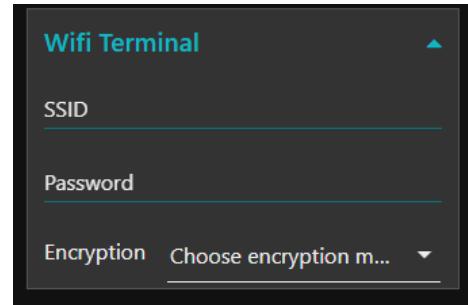
Jako bonus k termostatu byla implementována možnost přepnout pomocí podržení pravého tlačítka na QR kód s informacemi o domácí Wi-Fi síti. Díky tomuhle se může návštěva či ostatní obyvatelé domu připojit jednoduše k síti i bez opisování dlouhých hesel, pouze naskenováním QR kódu. Data v QR kódu je možné nastavit opět pomocí MQTT zprávy.



Obrázek 4.3: Ukázka Node-RED flow pro tento firmware. Foto autor



Obrázek 4.4: Vzorový dashboard pro tento firmware. Foto autor



Obrázek 4.5: Ukázka nastavení Wi-Fi QR kódu. Foto autor

## Záplavový detektor

Toto zařízení využívá záplavový senzor LD-81 od firmy Jablotron. Tento senzor plní funkci obyčejného tlačítka, a to tak že pokud jsou oba kontakty spojeny v tomto případě vodou tak je tlačítko považované za stisknuté. Následně je odeslána informace o spojení pomocí MQTT do prostředí Node-RED.

V prostředí Node-RED je poté možné poslat email či notifikaci do mobilního zařízení uživatele, který může konat na základě této informace.

Tato opatření proti poškození vodou je možno ještě vylepšit pomocí chytrého uzavíratelného vodního ventilu, který uzavře přívod vody v případě úniku vody. Tímto vylepšením je možné snížit poškození na minimum. Na trhu se nachází několik těchto chytrých ventilů ovládaných například pomocí technologie Zigbee, podobně jako třeba chytré žárovky IKEA. Pro spojení Zigbee zařízení a zařízení komunikujících přes MQTT je opět možné využít prostředí Node-RED.

## Obecné tlačítko

Toto poměrně obecné zařízení může mít v chytré domácnosti několik využití. Jedná se o firmware, který umožňuje detektovat spojení jakýchkoli dvou kontaktů.

Díky této vlastnosti je možné vytvořit bezdrátové vypínače v kombinaci s chytrými žárovkami nebo chytrými LED pásky, které jsou schopné komunikovat pomocí MQTT či Wi-Fi.

Dále může být detekován stisk tlačítka jako třeba zvonku u vstupních dveří. Po stisknutí bude odeslána notifikace na mobilní zařízení opět spojením do aplikace Blynk přes prostředí Node-RED.

Mezi další využití patří třeba ovládání žaluzií, rychlé zapnutí/vypnutí vytápění, a další.

## Obecný aktuátor

Zařízení s tímto firmwarem je schopné spínat okruhy díky relé. Díky této funkci je možné zapínat či vypínat topení, ovládat automatické žaluzie a ovládat mnoho dalších funkcí v chytré domácnosti.

Jak již bylo zmíněno dříve, relé je schopné simulovat jakékoli fyzické tlačítko bez nutnosti fyzického kontaktu uživatele. Pomocí tohoto je možné vytvořit například chytré osvětlení i bez chytrých žárovek pouze nahrazením fyzického vypínače za vypínač pomocí relé.

Pro demonstraci již dříve zmíněných open-source projektů dostupných od firmy HARD-WARIO je využit firmware s názvem twr-radio-power-controller, který mimo jiné umožňuje ovládání relé pomocí rádia.

### Zařízení pro obecné měření klimatických podmínek

S tímto zařízením je možné zjišťovat veškeré klimatické podmínky, teplotu, vlhkost, barometrický tlak, popřípadě i nadmořskou výšku. Dále oproti předchozím přibývá měření intenzity světla v jednotkách LUX.

Opět se jedná o bateriemi napájené zařízení a je možné jej využít uvnitř, i venku pod střechou například na balkóně. Při použití mimo střechu je nutno použít instalační krabici s vhodnou izolací, v tu chvíli odpadá možnost měřit intenzitu osvětlení, jelikož bude senzor zakrytý.

Využití těchto dat je různé. Teplota, vlhkost a barometrický tlak lze využít k základní předpovědi počasí nebo prostému sledování venkovních a vnitřních dat o klimatu.

Hodnota intenzity světla může ovládat automatické žaluzie, kdy ráno budou žaluzie vytaženy, následně dokud je přirozené světlo tak budou žaluzie vytaženy. Při snížení intenzity venkovního světla mohou být žaluzie zataženy a následně rozsvíceno vnitřní osvětlení.

### Chytré osvětlení IKEA TRÅDFRI

Jak bylo zmíněno již dříve, chytré osvětlení od firmy IKEA používá komunikační protokol Zigbee. Pro běžné ovládání v rámci domu stačí většinou spínače, které jsou v podstatě nutné k funkčnosti osvětlení jako takového. Pomocí spínačů je možné žárovku rozsvítit, zhasnout, snížit či zvýšit intenzitu, teplotu a u některých i barvu světla.

Pro zapojení do chytré domácnosti a možnosti monitoringu a případného ovládání osvětlení je nutné mít žárovky a spínače spárované s IKEA TRÅDFRI bránou.

Když je osvětlení spárováno, pak je možné sbírat data (stav, barvu a intenzitu světla apod.) a zároveň tyto data měnit v programovacím prostředí Node-RED. Díky tomuto je možné mimo jiné například zobrazovat stav žárovek v dashboardu, měnit barvu či intenzitu světla vzhledem k intenzitě přirozeného venkovního osvětlení měřenou jiným zařízením, vytvářet scény s dalšími zařízeními (ztlumení světel, zatažení žaluzií a puštění hudby po příchodu z práce).

## 4.3 Softwarové řešení

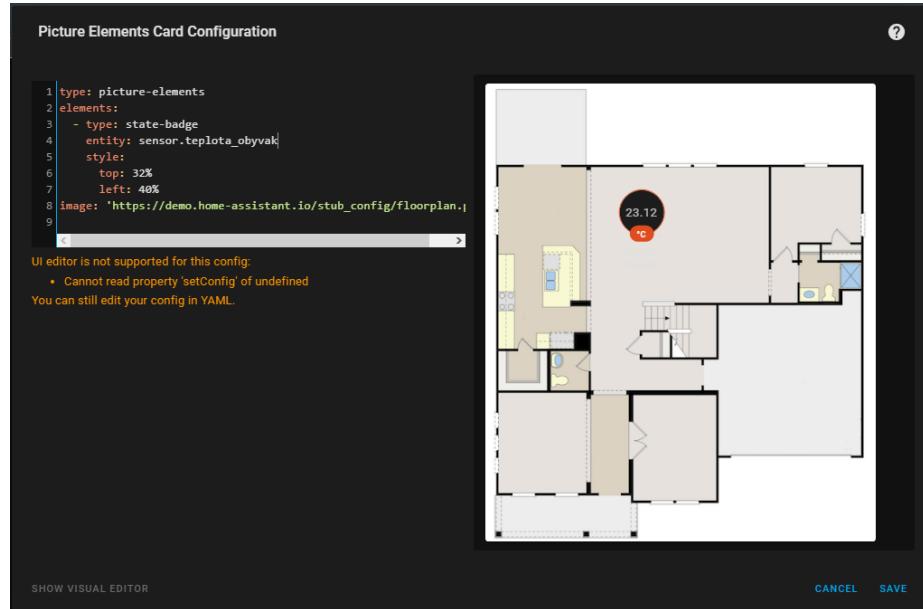
V softwarové části řešení bylo vyvinuto několik aplikací, vzorové projekty v aplikaci Blynk a také flow v programovacím prostředí Node-RED pro spojení jednotlivých prvků chytré domácnosti.

### Řešení pomocí vlastní implementace

#### Vzorová aplikace pro dohled a řízení chytré domácnosti

Tato aplikace, zmíněná v předešlých kapitolách, slouží jako vzorová implementace srozumitelné a jednoduché aplikace pro chytrou domácnost. Hlavní zaměření je na přehlednost grafického uživatelského rozhraní. Toto vytvořené uživatelské rozhraní je navrženo tak, aby bylo srozumitelné a bez problému použitelné i pro běžného uživatele s malou či žádnou technickou znalostí.

Inspirace pro aplikaci vychází z elementu ve službě Home Assistant, který umožňuje zobrazit data ze senzorů a dalších elementů a umístit je do zvoleného obrázku. Nepříjemná vlastnost toho elementu je, že je nutné jej konfigurovat pomocí souboru typu YAML. Elementy jsou do obrázku umisťovány pomocí absolutního pozicování.



Obrázek 4.6: Home Assistant pictures element. Foto autor

Vzhledem k tomuto vzoru bylo v aplikaci pohlíženo hlavně na jednoduché vytváření, umisťování a konfiguraci jednotlivých elementů.

## Návrh aplikace



Obrázek 4.7: Use case diagram pro aplikaci. Foto autor

Na Obrázku 4.7 je možno vidět Use Case diagram navržené aplikace, který zahrnuje všechny akce, které může uživatel v rámci aplikace vykonat. V diagramu se také nachází role Admin, který může upravovat rozložení aplikace a všechna nastavení a také dědí všechny akce, které může provádět běžný Uživatel.

### Zobrazení domu

V této části aplikace bude možno vložit plán domu a následně do něj vkládat hodnotová pole, spínače (aktualizovány v reálném čase), či kamery s možností přístupu k obrazu. Tento plán bude možné uložit do souboru k aplikaci. Tento soubor bude obnoven při dalším spuštění aplikace. Soubory s konfigurací bude možné volně přesouvat na další zařízení, na kterých bude aplikace spuštěna. Díky konfiguračním souborům bude možné aplikaci spustit kdekoli ve stejném stavu.

Plán domu bude nutno vytvořit v plánovacím softwaru a vložit jako soubor typu JPEG nebo PNG.

Následné ovládání tohoto zobrazení bylo vytvořeno tak aby bylo velice jednoduché pro každého uživatele. Pomocí pravého tlačítka myši je možné vložit jeden ze 3 elementů (hodnotové pole, spínač, kamera). Tyto elementy je poté možné nakonfigurovat pomocí dvojkliku a jednoduchého menu. Mezi možnostmi konfigurace jsou u jednotlivých elementů drobné změny. Jedno společné nastavení je MQTT topic, který je nutný pro přijímaní dat a přiřazení těchto dat ke správnému elementu.



Obrázek 4.8: Wireframe pro zobrazení domu. Foto autor

#### Elementy dostupné v aplikaci:

- **Hodnotové pole** - U hodnotového pole lze nastavit prefix a sufix. Hodnota bude následně zobrazena jako PREFIX + přijatá\_hodnota + SUFIX.

V případě že bude uživatel chtít například hodnotové pole zobrazující data z teploměru může zadat prefix "Teplota" a sufiks "°C". Následně při přijetí hodnoty "27", která byla odeslána pomocí MQTT na zadaný topic u hodnotového pole, bude v hodnotovém poli zobrazeno: "Teplota 27 °C".

- **Spínač** - Spínač může sloužit k zobrazování dat z jakéhokoli binárního senzoru či aktuátoru. Limitace tohoto elementu je nutnost odesílání pouze dat, která lze převést na hodnotu bool.

Pro správné nastavení je nutné zvolit ikonu pro stav *Zapnuto* a *Vypnuto*. Spínače jsou aktualizovány při změně stavu v reálném čase, dále je možné tyto spínače přepínat přímo v UI aplikace. Pokud by nebylo žádoucí, aby bylo možné elementy přepínat v aplikaci, jako například při využití na detekci pohybu, je možné v nastavení zvolit, že se jedná o neinteraktivní element a poté nebude kliknutí registrováno.

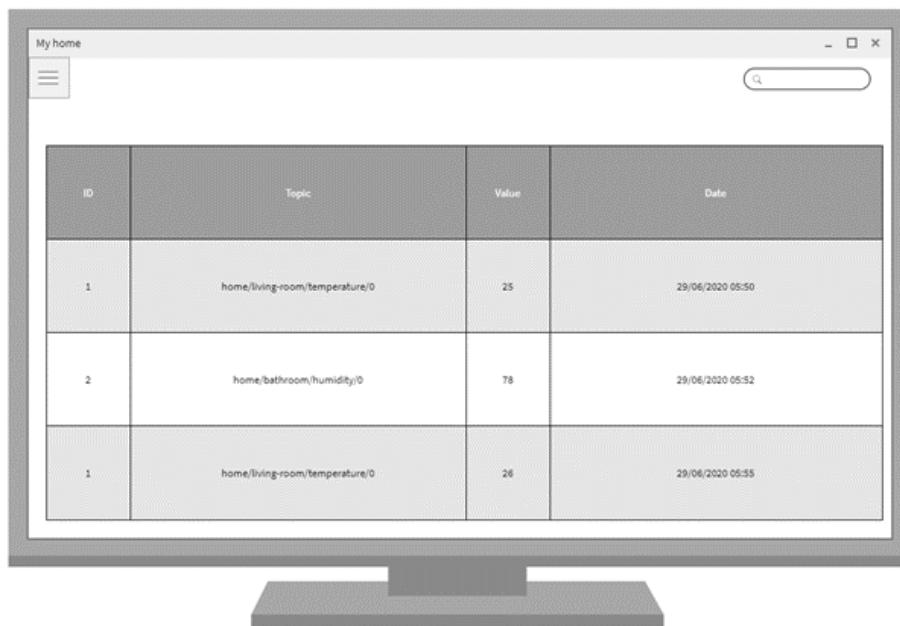
- **Kamera** - Kamery v aplikaci slouží k zobrazování pouze jednotlivých obrazů, ne video streamu. Pro správnou funkčnost tohoto elementu stačí pouze nastavit topic. Následně je nutné odesílat na tento topic obrazy v bitových polích. Takto je možno kamery simulovat pomocí vyvinutého MQTT simulátoru, který je popsán v další části této kapitoly 4.3.

Při najetí myši na správně nakonfigurovaný element, který již přijal nějaká data bude zobrazen poslední přijatý obraz přímo u konkrétního elementu kamery.

#### Zobrazení dat v tabulkách

V tomto zobrazení bude možno zobrazovat data ze senzorů v domě a filtrovat tato data

podle určitých parametrů jako například datum či hodnota. Díky tomuto zobrazení bude možné jednoduše data zpětně sledovat a analyzovat podle potřeby.



The wireframe shows a computer monitor displaying a table titled "My home". The table has four columns: "ID", "Topic", "Value", and "Date". There are three rows of data:

ID	Topic	Value	Date
1	home/living-room/temperature/0	25	29/06/2020 05:50
2	home/bathroom/humidity/0	78	29/06/2020 05:52
1	home/living-room/temperature/0	26	29/06/2020 05:55

Obrázek 4.9: Wireframe pro zobrazení dat z databáze. Foto autor

### Zobrazení dat v grafech

Toto zobrazení bude velice podobné předchozímu 4.3 s rozdílem, že budou data zobrazována graficky pro větší přehlednost a rychlejší analýzu.

Pro zobrazení grafu byl využit NuGet balíček LiveCharts.Wpf, který umožňuje zobrazovat jednoduché grafy se zvýrazněnými body. Graf je možný přiblížovat či oddalovat a následně graf analyzovat i pomocí posouvání po ose X.



Obrázek 4.10: Wireframe pro zobrazení dat pomocí grafů. Foto autor

### Implementační jazyk

Aplikace byla vytvořena v jazyce C# za pomocí frameworku WPF, aplikace s tímto frameworkem jsou spustitelné pouze na počítačích s operačním systémem Windows. Pokud by měla aplikace být multiplatformní bylo by možné využít například webových technologií či frameworku MAUI, který je ale v době psaní práce stále ve vývoji.

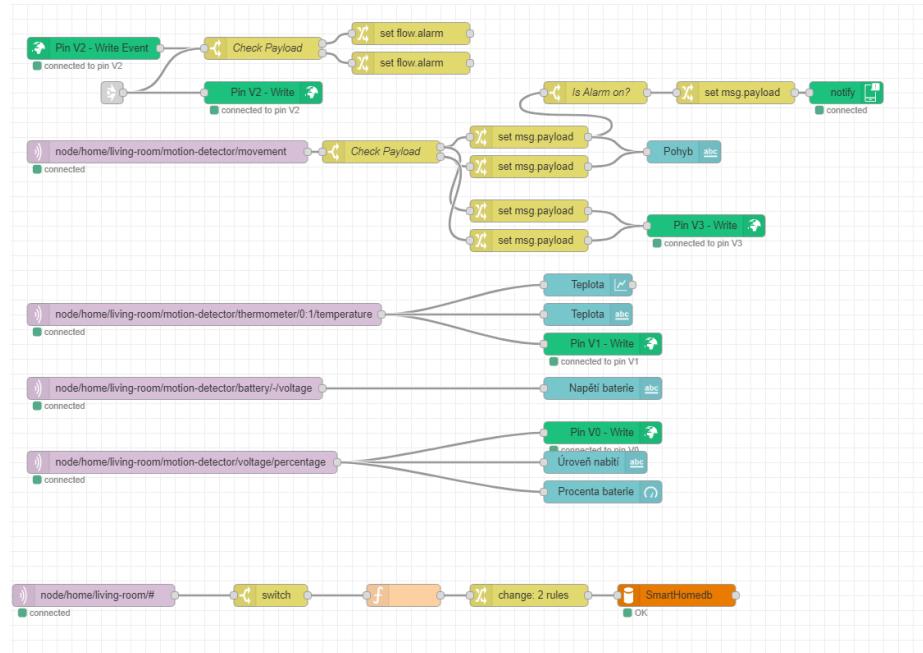
Při vývoji byl použit návrhový vzor MVVC (Model View View-Model) a Toolkit MVVM Light, který ulehčuje práci právě s tímto návrhovým vzorem v jazyce C#.

### Node-RED flows

Pro spojení hardwarových zařízení s aplikací Blynk, chytrými žárovkami a dalšími prvky chytré domácnosti bylo využito programovací prostředí Node-RED.

Pro každou místnost v domě bylo vytvořeno flow, které tato data přijímá, upravuje, a přeposílá ostatním službám jako je databáze.

Rozdělení flow do jednotlivých místností umožňuje rozšiřitelnost a udržitelnost. Spojení jednotlivých flow lze docílit pomocí nodů *link-in* a *link-out*, tyto nody mezi sebou vytváří virtuální spojení, které je možné využít v rámci jednoho či více flow.



Obrázek 4.11: Příklad flow pro obyvací pokoj. Foto autor

Tento příklad ukazuje provázání dat ze senzorů do grafů v Dashboardu Node-Red a do aplikace Blynk.

Horní část ukazuje použití link-out nodu, který má vstup v jiném flow.

Ve spodní části flow je možné vidět zapisování dat ze všech senzorů do databáze.

### Simulátory zařízení

Pro testování celkového řešení v jakémkoli prostředí i bez nutnosti instalace fyzických zařízení bylo vytvořeno několik simulátorů v jazyce Python.

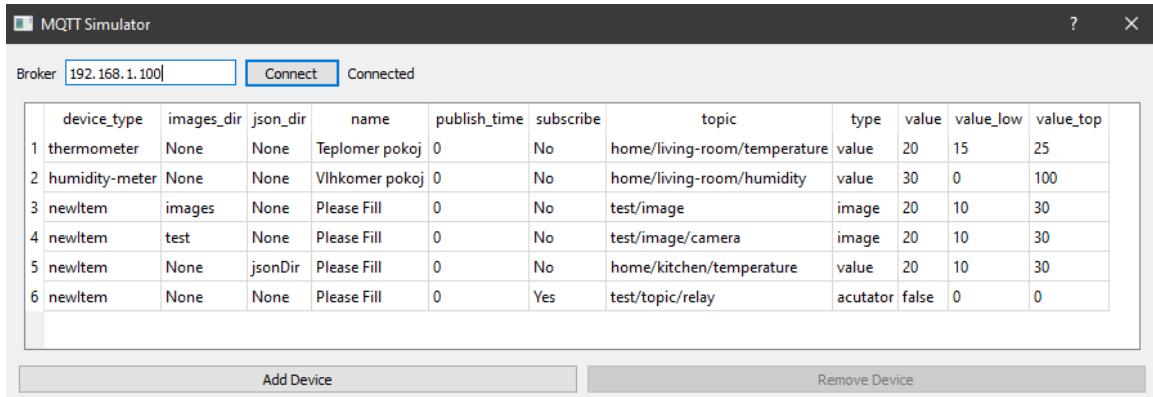
### Simulátor MQTT provozovu

Aplikace vznikala v rámci školního projektu paralelně s psaním této bakalářské práce a je využíván k simulování několika hodnotových senzorů, aktuátorů, spínačů a kamer.

Pro realističtější simulaci kamer byl vytvořen skript schopný simulovat kameru pomocí kamery dostupné na laptopu či Raspberry Pi 4.3.

### Implementační jazyk

Implementace proběhla v jazyce Python s využitím PyQt5, díky této kombinaci je možné aplikaci konfigurovat nejen souborem ve formátu JSON, ale také pomocí grafického uživatelského rozhraní, kde je možné v reálném čase upravovat jakékoli údaje o zvoleném zařízení.



Obrázek 4.12: Ukázka MQTT simulátoru. Foto autor

Jazyk Python byl také zvolen díky jeho možnosti spuštění na většině hlavních operačních systémů s minimálními nároky na instalaci. Pomocí nástrojů PyInstaller je možné vytvořit jednotný spustitelný soubor obsahující vše potřebné pro běh aplikace.

### Simulátor kamery

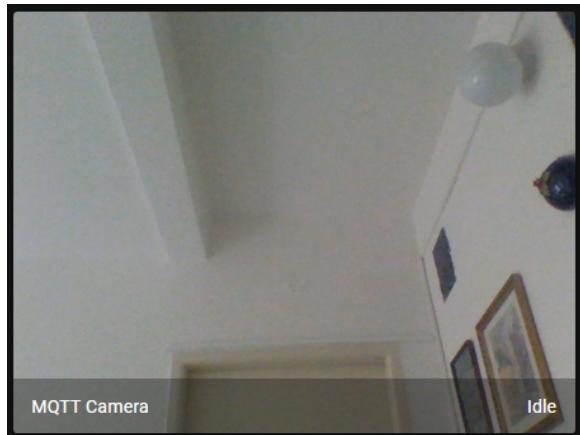
Pro simulování kamery byl vytvořen jednoduchý skript využívající opencv pro snímání obrazu z připojené kamery. Pro simulaci je možné použít laptop nebo jakýkoli jiný počítač s připojenou kamerou, který je schopný spustit Python skript.

Obraz z kamery je odesílan ve formáty bytového pole pomocí MQTT na zvolený server.

Skript je možné modifikovat pomocí argumentů při spuštění.

### Argumenty:

- -a, --address - IPv4 adresa zvoleného MQTT Brokeru. Pokud chybí je zvolena adresa 127.0.0.1
- -p, --port - Port, na kterém zvolený Broker běží. Pokud chybí je zvolen port 1883
- -u, --update - Hodnota v sekundách určující prodlevu mezi odeslanými snímky. Pokud chybí je zvolena hodnota 1 sekunda.
- -t, --topic - Topic, se kterým budou obrazová data z kamery odesílána. Pokud chybí je zvolena hodnota home/living-room/camera.



Obrázek 4.13: Ukázka výstupu z kamery v aplikaci Home Assistant. Foto autor

## Řešení pomocí Home Assistant

Pro porovnání vlastního softwarového řešení byla také vytvořena vzorová implementace pomocí této již hotové a používané aplikace. Home Assistant je spuštěn zároveň s vlastním řešením, které je popsáno v předchozí kapitole 4.3 pro porovnání.

### Instalace

Pro spuštění této služby na Raspberry Pi je možné využít několik možností.

**Možnosti instalace**[4]:

- **Image operačního systému** - rychlý a jednoduchý způsob, pokud je již Raspberry Pi využíváno k jiným účelům, tak bude nutné znova nainstalovat a spustit všechny využívané služby na starém systému
- **Kontejner v aplikaci Docker** - tato možnost umožňuje instalaci na již běžící server. Jednou z nevýhod je nutné explicitní zpřístupňování fyzicky připojených zařízení právě Dockera.
- **Home Assistant Core** - klasická instalace běžná pro linuxové systémy. Je potřeba nainstalovat několik python balíčků a následně samotný balíček homeassistant pomocí balíčkovacího systému pip.

Po instalaci jakoukoli z výše uvedených cest by mělo být prostředí aplikace dostupné na adrese `http://<host>:8123`, kde <host> je IP adresa či doménové jméno použitého Raspberry Pi

Při vývoji pro tuto práci byla zvolena možnost Home Assistant Core z důvodu možnosti instalace přímo na Raspberry Pi i bez potřeby přeinstalování celého systému.

### Spuštění

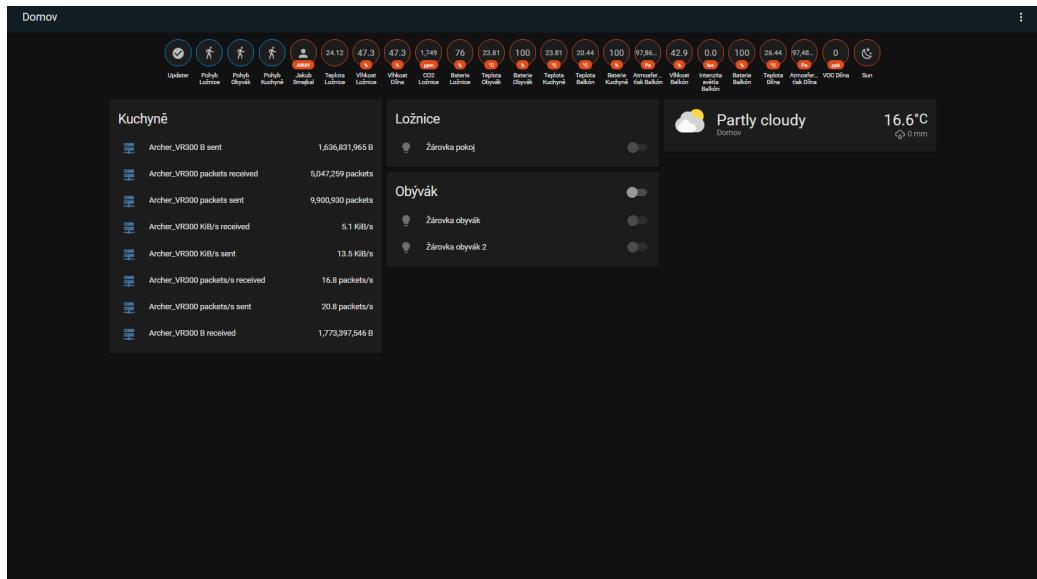
Po instalaci proběhne první nastavení aplikace. Velká výhoda systému je automatická detekce některých jiných prvků a Hubů v místní síti a zobrazení dostupných integrací pro tato detekovaná zařízení.

Pokud jsou zvoleny integrace, které přidávají další prvky do finálního dashboardu, tak jsou tyto prvky po dokončení instalace automaticky přidány na hlavní dashboard, který je přítomný a nakonfigurovaný ihned po spuštění.

## Konfigurace

Pro konfiguraci jednotlivých zařízení, které mají přímou integraci do systému Home Assistant je možné z velké části využít uživatelské rozhraní.

Hlavní přednastavený dashboard je aktualizován s jakýmkoli přidaným senzorem, aktuátorem či jiným prvkem. Tento způsob přidávání prvků se může velice rychle stát nepřehledným.



Obrázek 4.14: Automaticky generovaný dashboard v aplikaci Home Assistant. Foto autor

Tento dashboard je možné využívat v menších projektech či jako všeobecný přehled. Pro organizovanější a více upravitelné prostředí je možné kdykoli vytvořit nový dashboard.

Zařízení komunikující pomocí protokolu MQTT je ovšem nutné nastavit pomocí konfiguračního souboru typu YAML nacházejícího se v adresáři “/home/homeassistant/.homeassistant“. Výhodou této konfigurace je poměrně široká možnost nastavení daného senzoru včetně například ikony podle parametru `device_class`.

```
sensor:  
  - platform: mqtt  
    state_topic: "node/home/bedroom/co2-monitor/thermometer/0:0/temperature"  
    name: "Teplota Ložnice"  
    device_class: "temperature"  
    unit_of_measurement: "C"
```

# Kapitola 5

## Závěr

### 5.1 Srovnání řešení

V této kapitole bude srovnáno řešení pomocí Home Assistant s řešením pomocí aplikace vytvořené v rámci této práce. Porovnání proběhne pomocí několika stanovených parametrů, které jsou v obou aplikacích odlišitelné.

#### Platforma

##### Home Assistant

Home Assistant je dostupný v internetovém prohlížeči a zároveň také jako mobilní aplikace jedná se tedy o aplikaci multiplatformní.

##### Vytvořená aplikace

Jak již bylo zmíněno dříve, tak je vytvořená aplikace pouze pro platformu Windows. Tento problém by mohl být eliminován s dokončením vývoje frameworku MAUI, který by umožnil vyvinutí ekvivalentní aplikace i na mobilní zařízení a pro další operační systémy.

Jako řešení problému s dostupností na dalších zařízeních byla využita aplikace Blynk, která umožňuje monitoring a ovládání domácnosti i pomocí mobilního zařízení.

#### Dashboard

##### Home Assistant

V Home Assistant je v základu dostupné několik různých elementů na hlavním dashboardu. Tyto elementy mohou zobrazovat data v různých grafech, hodnotových polích a spínačích.

Toto porovnání se bude ovšem zaměřovat pouze na takzvaný Picture Elements, kterým se vytvářená aplikace inspirovala.

##### Elementy dostupné v Picture Elements:[3]

- **State Badge** - symbol označující stav entity
- **State Icon** - ikona označující stav entity
- **State Label** - zobrazení stavu entity pomocí textu
- **Service Call Button** - tlačítko schopné volat službu
- **Icon** - obyčejná ikona nespojená s entitou

- **Image** - obrázek překrývající obrázek na pozadí
- **Conditional** - element zobrazující sub-elementy na základě stavu entity
- **Custom** - uživatelem vytvořený element

Všechny tyto elementy je možné umístit do zvoleného obrázku pomocí kaskádových stylů.

### **Vytvořená aplikace**

Vytvořená aplikace se zaměřuje na přehlednost a jednoduchost dashboardu. Elementy dostupné v aplikaci byly dopodrobna probrány dříve. Pomocí těchto elementů je možné vytvořit dashboard, který bude schopný ovládat chytrou domácnost a také zobrazovat všechny potřebné hodnoty. Umistování elementů do dashboardu probíhá pomocí jednoduchého menu a následně je možné je umisťovat pomocí myši.

I když je paleta elementů menší, jsou schopny pokrýt většinu funkcí jako elementy zmíněné u Home Assistant.

Jako rozšířenou funkci dashboardu je možné data zobrazovat také v tabulce či grafu.

## **Integrace**

### **Home Assistant**

Home Assistant jako dlouhou dobu dostupná aplikace podporovaná komunitou zahrnuje velké množství integrací. Díky tomu je možné provozovat většinu zařízení přímo pomocí Home Assistant.

Nevýhodou tohoto přístupu je, že je nutné odesílat ze senzorů a aktuátorů data ve správném stavu. Manipulace a úprava dat je kvůli přímému spojení složitá.

### **Vytvořená aplikace**

Všechny senzory a aktuátory, které mají komunikovat s aplikací přímo, musí komunikovat pomocí protokolu MQTT. Vzhledem k integraci s aplikací Blynk je použito programovací prostředí Node-RED, ve kterém je možné propojit do MQTT i další zařízení komunikující i jinými protokoly než MQTT.

Pozn.: Prostředí Node-RED je samozřejmě možné použít i v kombinaci s Home Assistant ale není to nutné pro větší množství integrací.

## **Konfigurace**

### **Home Assistant**

Jak již bylo zmíněno dříve Home Assistant je konfigurovaný pomocí YAML souborů. Opět je tu výhoda několika existujících integrací. U některých těchto integrací jsou zařízení detekována a přidána automaticky.

V rámci této práce bylo vytvořeno řešení, kde zařízení komunikují pomocí protokolu MQTT. I když MQTT je jedna z integrací pro Home Assistant tak nejsou zařízení přidávána automaticky.

Každé MQTT zařízení je nutné nakonfigurovat pomocí YAML souboru nacházejícího se přímo na serveru. Toto nastavení je zdokumentované ale je nutné se připojit na server, lokalizovat soubor a tento soubor upravit.

## Vytvořená aplikace

Ve vytvořené aplikaci byl na konfiguraci dashboardu kladen důraz. Proto je možné konfigurovat aplikaci pomocí několika základních informací.

Co se nastavení propojení s databází a MQTT Brokerem týče, stačí vyplnit údaje jako je IP adresa, uživatelské jméno, heslo, atp. Všechna tato připojení je možné vyzkoušet přímo v aplikaci pro ověření spojení.

Další nastavení dashboardu probíhá čistě a jen v UI aplikace. Pomocí myši je možné umisťovat elementy do plánu domu a ty následně konfigurovat. Konfigurace elementu probíhá v jednoduchém menu, které umožňuje provádět úpravy na zvoleném elementu.

## Celkové srovnání

I když je samozřejmě, že prototyp vytvořené aplikace nemůže celkově soupeřit s již vytvořeným řešením, které je mimo jiné podporované velkou aktivní komunitou, tak původní návrh a požadavky na tuto aplikaci byly splněny.

Aplikaci je možné jednoduše ovládat a nastavit. Během psaní práce a testování všech řešení (Home Assistant, Blynk, Node-RED Dashboard) jsem aplikaci využíval k dohledu a ovládání domácnosti. V případě, kdy je možné aby senzory odesílali a přijímal data ve správném formátu, je jediný potřebný prvek MQTT Broker a dohledový PC se systémem Windows.

Aplikaci může nastavit a provozovat i poměrně nezkušený uživatel.

Ukázka pohybového Velkou nevýhodou je samozřejmě to, že je aplikace pouze pro systém Windows. Pro vyřešení tohoto problému je již nutné provést složitější nastavení a spojení do aplikace Blynk pomocí prostředí Node-RED.

## 5.2 Shrnutí

V rámci práce byly obě řešení spuštěny souběžně.

Díky tomu je umožněn přístup z různých zařízení a pro různé uživatele. Všechny služby byly provozovány na jednom zařízení a to konkrétně na Raspberry Pi 3b a i při souběžném spuštění několika služeb se nevyskytl žádný problém s výkonem. Řešení by bez problémů fungovalo i na jiných verzích tohoto zařízení či na jiném druhu embedded systému.

Pro dohled a řízení chytré domácnosti v domácím prostředí, popřípadě při přístupu ke klasickému PC, je možné využít aplikaci vytvořenou v rámci tohoto projektu. Aplikace poskytuje přehledné plány domu, které je možné rychle nastavit a upravit.

Při dohledu mimo domácnost či z mobilního zařízení je možné využít Home Assistant či aplikaci Blynk, která je spojená pomocí aplikace Node-RED s ostatními zařízeními v chytré domácnosti.

Prototyp vytvořené aplikace by mohl být rozšířen o další elementy v rámci dashboardu a další možná nastavení těchto elementů. Další možné rozšíření by bylo zahrnutí základních automatizací a kalendářů, které by umožnily nastavování různých profilů.

Vzorové hardwarové řešení vytvořené v rámci této práce slouží jako ukázka možností chytré domácnosti. Pomocí dalších prvků zmíněných v první kapitole práce je možné vytvořit další zařízení pro konkrétní využití.

Použití stavebnice HARDWARIO Tower bylo z důvodu ukázky možností jednoduché a rychlé implementace. Samozřejmě by bylo možné použít i jiné druhy mikrokontrolerů jako například Arduino či esp.

Použití systému pro dohled a řízení chytré domácnosti může být v mnoha směrech přínosné ať už se jedná o finanční úspory vzhledem k automatizovanému ovládání několika prvků jako je vytápění či klimatizace. Mezi další výhody patří komfortní monitorování různých hodnot, které umožňují zajišťovat větší kvalitu životního prostředí a popřípadě i například zabezpečení domácnosti či dozor nad celým objektem.

Při návrhu systému je nutné myslit na to jaké hodnoty je důležité v domácnosti monitorovat, popřípadě jak tyto hodnoty můžou být použity k automatickému ovládání dalších prvků v systému. Dalším důležitým rozhodnutím je volba technologií. Zda budou použity open či closed source řešení, ať už hardwarové či softwarové, jakým způsobem bude uživatel přistupovat k datům a odkud mají být data přístupná.

S přístupností dat se přímo pojí zabezpečení celého systému. Ať už je systém dostupný jakýmkoli způsobem z externích sítí je dobré zvolit bezpečná hesla pro všechny služby tak aby bylo co nej obtížnější získat ze systému jakákoli citlivá data.

# Literatura

- [1] ANUSHA. *Humidity Sensor – Types and Working Principle* [online]. <https://www.electronicshub.org/humidity-sensor-types-working-principle/>.
- [2] (ARSABOO), A. S. *Home Assistant Demo* [online]. 2021 [cit. 2021-04-27]. Dostupné z: <https://demo.home-assistant.io/#/lovelace/0>.
- [3] ASSISTANT, H. *Picture Elements Card* [online]. Home Assistant, 2019 [cit. 2021-04-27]. Dostupné z: <https://www.home-assistant.io/lovelace/picture-elements>.
- [4] ASSISTANT, H. *Install Home Assistant Operating System* [online]. Home Assistant, 2020 [cit. 2021-04-27]. Dostupné z: <https://www.home-assistant.io/installation/raspberrypi/>.
- [5] BAMBUŠEK, D. *Relay* [online]. <https://www.bambusekd.cz/dev/raspberry-control-5V-relay>.
- [6] CAMPBELL, S. *Basics of the I2C communication protocol* [online]. Circuit Basics, 2016 [cit. 2021-04-27]. Dostupné z: <https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/>.
- [7] CAMPBELL, S. *Basics of the SPI communication protocol* [online]. Circuit Basics, 2016 [cit. 2021-04-27]. Dostupné z: <https://www.circuitbasics.com/basics-of-the-spi-communication-protocol/>.
- [8] CAMPBELL, S. *Basics of UART communication* [online]. Circuit Basics, 2016 [cit. 2021-04-27]. Dostupné z: <https://www.circuitbasics.com/basics-uart-communication/>.
- [9] CONDITIONING, B. H. . A. *What is an HVAC System and How Does it Work?* [online]. <https://brennanheating.com/how-does-hvac-system-work/>.
- [10] DASHTICZ. *Default Dashboard* [online]. <https://dashticz.readthedocs.io/en/master/gettingstarted/basicdashboard.html>.
- [11] ELPROCUS. *Temperature Sensors – Types, Working and Operation* [online]. <https://www.elprocus.com/temperature-sensors-types-working-operation/>.

- [12] ENECTIVA, T. *Vliv koncentrace CO<sub>2</sub> na zdraví člověka* [online]. <https://www.enectiva.cz>, 2017 [cit. 2021-04-27]. Dostupné z: <https://www.enectiva.cz/cs/blog/2017/04/co2-vnitri-prostory/>.
- [13] HEALTH, M. D. of. *Volatile Organic Compounds in Your Home* [online]. <https://www.health.state.mn.us> [cit. 2021-04-27]. Dostupné z: <https://www.health.state.mn.us/communities/environment/air/toxins/voc.htm>.
- [14] HTTPS://COMMUNITY.ESTIMOTE.COM. *Basics of UART communication* [online]. <https://community.estimote.com> [cit. 2021-04-27]. Dostupné z: <https://community.estimote.com/hc/en-us/articles/217429867-What-is-GPIO->.
- [15] HTTPS://LEARN.ADAFRUIT.COM. *Arduino Comparison Chart* [online]. <https://learn.adafruit.com>, 2021 [cit. 2021-04-27]. Dostupné z: <https://learn.adafruit.com/adafruit-arduino-selection-guide/arduino-comparison-chart>.
- [16] HTTPS://MQTT.ORG. *MQTT Publish / Subscribe Architecture* [online]. <https://mqtt.org>, 2020 [cit. 2021-04-27]. Dostupné z: <https://mqtt.org/assets/img/mqtt-publish-subscribe.png>.
- [17] INTEGRATED, M. *Guide to 1-wire communication* [online]. <https://www.maximintegrated.com>, 2008 [cit. 2021-04-27]. Dostupné z: <https://www.maximintegrated.com/en/design/technical-documents/tutorials/1/1796.html>.
- [18] LAFOND, A. *How to Measure Volatile Organic Compounds In the Air* [online]. <https://foobot.io>, 2021 [cit. 2021-04-27]. Dostupné z: <https://foobot.io/guides/how-to-measure-volatile-organic-compounds-in-air.php>.
- [19] PANWAR, A. *Types of Database Management Systems* [online]. C# Corner, 2020 [cit. 2021-04-27]. Dostupné z: <https://www.c-sharpcorner.com/UploadFile/65fc13/types-of-database-management-systems/>.
- [20] SFUPTOWNMAKER. *I2C* [online]. <https://learn.sparkfun.com>, 2017 [cit. 2021-04-27]. Dostupné z: <https://learn.sparkfun.com/tutorials/i2c/all>.
- [21] SMITH, D. *CO<sub>2</sub> Sensors: Which Type Should You Be Looking For?* [online]. <https://learn.kaiterra.com>, 2019 [cit. 2021-04-27]. Dostupné z: <https://learn.kaiterra.com/en/air-academy/carbon-dioxide-sensors>.
- [22] S.R.O., H. *About VOC Tag* [online]. <https://www.hardwario.com>, 2020 [cit. 2021-04-27]. Dostupné z: <https://tower.hardwario.com/en/latest/hardware/about-voc-tag/>.
- [23] S.R.O., H. *Subghz Radio* [online]. <https://www.hardwario.com>, 2020 [cit. 2021-04-27]. Dostupné z: <https://tower.hardwario.com/en/latest/interfaces/sub-ghz-radio/>.
- [24] S.R.O., H. *Subghz Radio* [online]. <https://www.hardwario.com>, 2020 [cit. 2021-04-27]. Dostupné z: <https://tower.hardwario.com/en/latest/interfaces/sub-ghz-radio/>.
- [25] STAFF, E. *Serial Protocols Compared* [online]. <https://www.embedded.com>, 2002 [cit. 2021-04-27]. Dostupné z: <https://www.embedded.com/serial-protocols-compared/>.

- [26] TEAM, T. H. *MQTT Topics and Best Practices - MQTT Essentials: Part 5* [online].  
https://www.hivemq.com, 2019 [cit. 2021-04-27]. Dostupné z: <https://www.hivemq.com/blog/mqtt-essentials-part-5-mqtt-topics-best-practices/>.

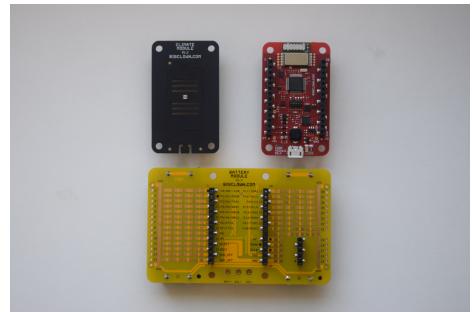
## Příloha A

# Ukázka hardwarových zařízení

V této kapitole jsou ukázána některá hardwarová zařízení používána v rámci této práce. Popis firmwarů pro tato zařízení je možné nalézt v kapitole 4.2.



Obrázek A.1: Monitor klimatických podmínek pro indoor či outdoor. Foto autor

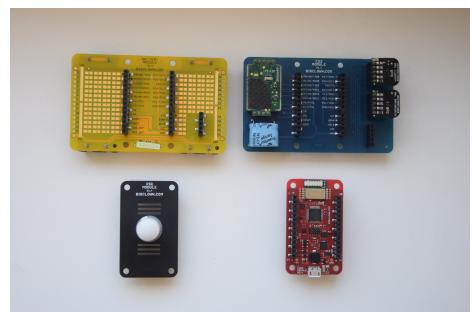


Obrázek A.2: Stejně zařízení rozložené na jednotlivé moduly. Foto autor

Na obrázku A.2 je vidět rozložené zařízení. Žlutý modul je napájecí bateriový modul, červený obsahuje samotný mikroprocesor a ovládá celé zařízení a černý modul se stará o měření klimatických podmínek.



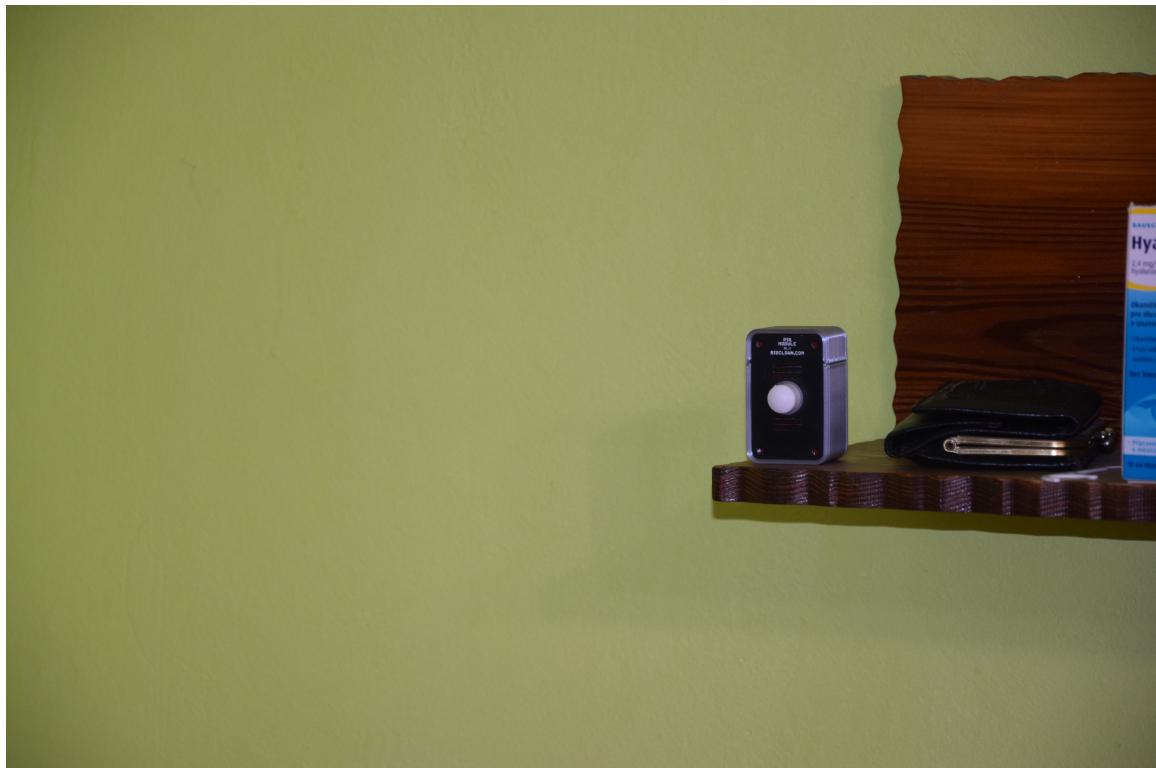
Obrázek A.3: Monitor klimatických podmínek pro indoor s detektorem pohybu. Foto autor



Obrázek A.4: Stejně zařízení rozložené na jednotlivé moduly. Foto autor

Na obrázku A.4 je vidět rozložené zařízení. Žlutý modul je napájecí bateriový modul, červený obsahuje samotný mikroprocesor a ovládá celé zařízení a černý modul se stará o

detekci pohybu pomocí PIR čidla. Modrý modul obsahuje měřič  $CO_2$  a několik takzvaných tagů pro měření dalších klimatických podmínek.



Obrázek A.5: Ukázka umístění hardwarového zařízení v domě. Foto autor