



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**BEZDRÁTOVÉ OVLÁDÁNÍ ELEKTRONIKY
MOBILNÍM/EMBEDDED ZAŘÍZENÍM S VYUŽITÍM WIFI**

WIRELESS CONTROL OF ELECTRONIC DEVICES USING MOBILE/EMBEDDED DEVICE USING

WIFI

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

PETR MAREK

VEDOUCÍ PRÁCE

SUPERVISOR

prof. Dr. Ing. PAVEL ZEMČÍK

BRNO 2021

Zadání bakalářské práce



Student: **Marek Petr**
Program: Informační technologie
Název: **Bezdrátové ovládání elektroniky mobilním/embedded zařízením s využitím WiFi**
Wireless Control of Electronic Devices Using Mobile/Embedded Device Using WiFi
Kategorie: Vestavěné systémy

Zadání:

1. Prostudujte dostupná řešení a literaturu na téma ovládání embedded zařízení s využitím WiFi, případně jiných přenosových bezdrátových protokolů, zaměřte se i na možnosti využití Raspberry Pi.
2. Vytipujte vhodnou aplikaci, například pro domácnost, diskutujte dosažitelné vlastnosti řešení takové aplikace, například ovládání světel a domácích spotřebičů s tím, že samotným ovladačem by mohl být i mobilní telefon.
3. Navrhněte způsob implementace vybrané aplikace a popište možnosti a dosažitelné vlastnosti.
4. Implementujte aplikaci a její ovládání a demonstруйте vlastnosti a možnosti na vhodném příkladu.
5. Diskutujte dosažené výsledky a další možnosti pokračování práce.

Literatura:

- Dle pokynů vedoucího

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

Vedoucí práce: **Zemčík Pavel, prof. Dr. Ing.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2020

Datum odevzdání: 12. května 2021

Datum schválení: 30. dubna 2021

Abstrakt

Cílem této bakalářské práce bylo navrhnout a realizovat systém automatizované domácnosti s využitím Wi-Fi. Celý systém byl postaven na platformách Raspberry Pi a ESP8266, jako komunikační protokol byl zvolen CoAP. Výsledné řešení funguje v lokální síti i přes Internet. Zahrnuje využití několika typů snímačů a možnost ovládání výstupů modulů ESP8266. Systém podporuje jednoduchou automatizaci na základě hodnot snímačů a nastavených časovačů.

Abstract

The aim of this bachelor thesis was to design and implement an automated home system using Wi-Fi. The whole system was built on the Raspberry Pi and ESP8266 platforms, CoAP was chosen as the communication protocol. The resulting solution works in the local network and also over the Internet. It includes the use of several types of sensors and the ability to control the outputs of the ESP8266 modules. The system supports simple automation based on values of sensors and set timers.

Klíčová slova

Raspberry Pi, ESP8266, Wi-Fi, CoAP, bezdrátové ovládání, automatizace domácnosti

Keywords

Raspberry Pi, ESP8266, Wi-Fi, CoAP, remote control, home automation

Citace

MAREK, Petr. *Bezdrátové ovládání elektroniky mobilním/embedded zařízením s využitím WiFi*. Brno, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce prof. Dr. Ing. Pavel Zemčík

Bezdrátové ovládání elektroniky mobilním/embedded zařízením s využitím WiFi

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana prof. Dr. Ing. Pavla Zemčíka. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Petr Marek
10. května 2021

Poděkování

Na tomto místě bych chtěl poděkovat panu prof. Dr. Ing. Pavlu Zemčíkovi za odborné vedení mé práce, za jeho čas, cenné poznámky a připomínky, a také za možnost častých osobních konzultací.

Obsah

1	Úvod	2
2	Automatizace domácnosti a související technologie	3
2.1	Internet věcí a automatizace domácnosti	3
2.2	Komunikační protokol CoAP	7
2.3	MQTT protokol	9
2.4	Bezdrátový přenos dat a signálů	10
2.5	Existující řešení chytrých domácností	11
3	Vestavěné systémy a bezdrátové technologie	23
3.1	Vestavěné systémy	23
3.2	Jednodeskový počítač Raspberry Pi	24
3.3	Moduly společnosti Espressif Systems	26
3.4	Technologie bezdrátového přenosu Wi-Fi	28
3.5	Ostatní technologie bezdrátového přenosu	29
4	Zhodnocení možností systémů chytrých domácností a plán práce	31
4.1	Zhodnocení současného stavu v oblasti automatizace domácnosti	31
4.2	Návrh technického řešení systému automatizace domácnosti	33
4.3	Požadované vlastnosti navrhovaného systému	34
5	Realizace systému automatizace domácnosti a testování	36
5.1	Celkový návrh systému automatizace domácnosti	36
5.2	Návrh a implementace webové aplikace	40
5.3	Databáze, autentizace v systému, webhosting a lokální funkce systému	44
5.4	Implementace serveru na Raspberry Pi a aplikace pro koncové moduly	46
5.5	Testování a vyhodnocení systému	49
6	Závěr	53
	Literatura	54
A	Obsah přiloženého paměťového média	60

Kapitola 1

Úvod

Nacházíme se v době rychlého vývoje techniky, který pronikl i do našich domácností. Každým dnem roste počet zařízení denní potřeby, která jsou připojená do celosvětové sítě – Internetu. Spolu s tímto trendem roste také komfort a možnosti bydlení, které z toho plynou. Díky rychlému rozvoji automatizace je možné provádět úkoly, které by dříve ani nebyly možné. Internet věcí lidem umožňuje vykonávat úkony – jako vypínat spotřebiče při opuštění domácnosti, nastavit efektivní režim vytápění před příjezdem domů či například zajistit zabezpečení celé domácnosti. A právě automatizaci domácnosti se věnuje moje práce.

Na trhu se nachází nejrůznější typy systémů domácí automatizace. V závislosti na ceně je možné pořídit jednoduchá zařízení, jako jednotlivé zásuvky či světla, až po různá komplexní řešení určená k zabudování do rozvaděčů domácnosti za účelem ovládní celé domácnosti. Komplexní řešení mohou pokrývat řadu různých snímačů a zařízení, které spolu mohou vytvořit opravdu automatizovaný dům. Přestože jde vývoj rychle dopředu, tak je cena u komplexnějších systémů velmi vysoká a většina domácností se tedy bude muset bez automatizace obejít.

A právě kvůli zajímavosti těchto systémů a jejich doposud vysoké ceně jsem se i já rozhodl v této práci zabývat možnostmi ovládní zařízení na dálku, potažmo možností jejich automatizace. Rozhodl jsem se, že veškerý kód, který v rámci práce vznikne, bude uvolněn jako open source s možností dalšího vývoje.

V následující kapitole je shrnut současný stav v oblasti automatizace domácnosti, technologií, které využívá a některých existujících řešení. Další kapitola se věnuje vestavěným zařízením, které tvoří součást automatizovaných domácností a bezdrátovým technologiím, které tyto vestavěné systémy využívají. Dále je v kapitole 5 uvedeno zhodnocení současného stavu a návrh mého řešení, kterého realizaci je věnována kapitola 6. V poslední kapitole je práce shrnuta a je naznačen směr, kterým by se v budoucnu mohlo realizované řešení vyvíjet.

Kapitola 2

Automatizace domácnosti a související technologie

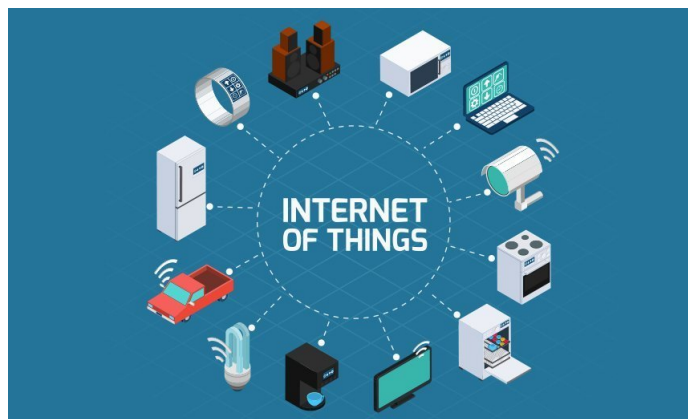
Tato kapitola je shrnutím současného stavu v oblasti chytré či automatizované domácnosti, technologií, které využívá a existujících řešení. Není encyklopedickým výkladem problematiky, ale souhrnem informací, které mají k práci bezprostřední vztah. Protože je téma poněkud rozsáhlé, jsou zde zmíněny pouze informace v přiměřeném rozsahu pro tuto práci.

2.1 Internet věcí a automatizace domácnosti

Tato podkapitola se věnuje internetu věcí a automatizaci domácnosti. Kromě vysvětlení těchto dvou pojmů shrnuje, jaké jsou možnosti využití automatizace v domácnosti, jaké má přínosy, jak ji dělíme, jaké obvykle podporuje funkce a z jakých komponent se skládá.

Internet věcí a chytrá zařízení

K automatizaci domácnosti neodmyslitelně patří tzv. Internet of Things (IoT), tedy internet věcí, který celý koncept automatizace významně rozšiřuje.



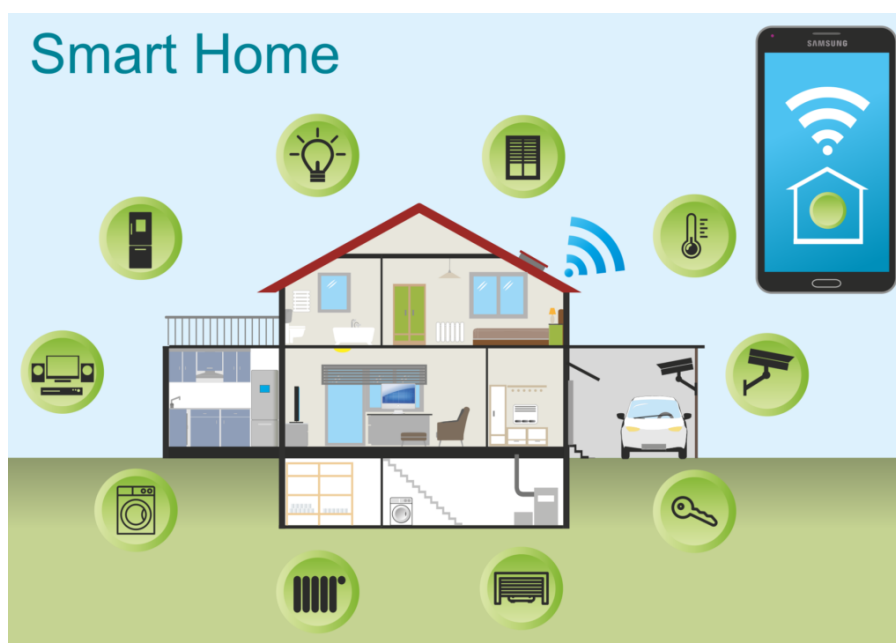
Obrázek 2.1: Koncepte internetu věcí, spojující nejrůznější chytrá zařízení¹.

¹Převzato z <https://engalaxy.com/internet-of-things-iot/>

Zatímco pod pojmem automatizace je možné si představit jakoukoli automatizovanou činnost, internet věcí označuje systém propojení různých zařízení, aplikací, snímačů a akčních členů, které mezi sebou mohou navzájem komunikovat a reagovat na sebe [33]. Mezi aplikace konceptu internetu věcí patří široké spektrum odvětví jako chytrý průmysl, chytrá města, chytré zemědělství, chytré zdravotnictví nebo konečně chytré domácnosti.

Automatizace domácnosti a chytrý dům

Automatizace domácnosti spočívá v automatizování činností, které řídí domácnost, normálně vykonávané člověkem. Můžeme ji definovat jako mechanismus, který nahrazuje lidskou námahu (z ovládní domácnosti) natolik, nakolik je to jen možné [61]. V souvislosti s tím někdy hovoříme o inteligentní, řízené či chytré domácnosti. Jedná se o kolekci zařízení a (pod)systémů, které jsou schopny spolu komunikovat či fungovat nezávisle.



Obrázek 2.2: Znázornění některých aplikací automatizované (či chytré) domácnosti².

Automatizovaný „dům budoucnosti“ slibují výrobci domácích zařízení prakticky již téměř od počátku minulého století [22]. Chytrá domácnost (či smart home) je také definována jako dům, vybavený výpočetní a informační technologií, které předvídá uživatelské potřeby a odpovídá na ně, a přitom dbá na jeho pohodlí, bezpečnost a zábavu [25]. Často se tedy tyto dva pojmy — automatizovaná a chytrá domácnost — zaměňují. Chytrý dům však označuje spíše skupinu zařízení, schopných spolu nějakým způsobem (zejména po síti) komunikovat, zatímco automatizovaný dům je pojem, který popisuje, jakým způsobem jsou tato zařízení využívána [43].

Možnosti využití automatizace v domácnosti

Automatizace v mnohém usnadňuje život a umožňuje provádění akcí, které by jinak byly prakticky nemožné (například zabezpečení domu, efektivní řízení vytápění domácnosti

²Převzato z <https://alejandraslife.com/smart-home-how-to-manage-your-energy/>

a spotřeby energie). V současné době patří automatizace domácnosti mezi rychle se rozvíjející technologie [50]. Mezi její typické aplikace dle literatury [24] patří:

- zabezpečovací systém,
- systém pro inteligentní vytápění a ventilaci (HVAC),
- zábava a multimédia,
- komunikace,
- ovládání spotřebičů a osvětlení.

A také mnoho dalšího. Tento výčet obsahuje nejčastější použití chytrých domácností. Jelikož se však jedná o stále se rozvíjející obor, tak adekvátně přibývají i možnosti využití. Zmíněné aplikace jsou nejtypičtější, ke kterým se automatizace používá, ovšem na trhu vzniká čím dál více inovací, jako chytrá lednice či kávovar. Na obrázku 2.2 můžeme vidět symbolické znázornění některých chytrých prvků v domácnosti, které je dnes již možné automatizovat.

Přínosy automatizace domácnosti

Přidání inteligence do domácnosti přináší do života lidí řadu přínosů [26][56]. Jde zejména o následující:

- Bezpečí – Chytré domy mohou používat různé senzory, které detekují nebezpečí a v souvislosti s nimi provést patřičné akce k jejich zabránění, případně minimalizaci škod. Příkladem mohou být záplavové a kouřové senzory a v neposlední řadě také zabezpečovací systém domácnosti.
- Komfort – Chytré domácnosti svými funkcemi nabízejí různé způsoby, jak jejich uživatelům zpříjemnit různé rutinní akce. Mohou se postarat o automatické nastavování žaluzií dle intenzity venkovního světla, přes dotykový displej na dálku ztlumit světlo či hlasovým pokynem uvést celý byt do jiného světelného režimu.
- Přehled o provozu – Systémy pro automatizaci domácnosti zahrnují i displeje s přehledem o stavu jednotlivých zařízení a čidel. Také je v některých systémech možné tyto informace sledovat i z chytrých telefonů, tabletů či počítačů (a to i vzdáleně). V některých komplexnějších systémech, které například zahrnují komunikaci přes mobilní síť, je možné získávat přehled o provozu dokonce pomocí SMS zprávy (hodí se třeba při absenci internetového připojení).
- Úspora – V chytrých domácnostech je možné použít inteligentní vytápění domu založené na údajích z teplotních čidel, denní doby, případně nastaveném režimu domácnosti. Společnost ELKO EP odhaduje, že díky bezdrátové regulaci topení je možné ušetřit až 30% nákladů na energii.

Úsporu rovněž zajistí automatizovaná světla, o kterých je možné mít v automatizované domácnosti vždy přehled, na dálku je zapínat/vypínat dle potřeby a rovněž je napojit na senzory, které je budou ovládat například na základě přítomnosti osob v místnosti.

Základní klasifikace chytré domácnosti

Chytrou domácnost můžeme rozdělit dle použité technologie [26]. Mezi možnosti patří rozdělení na:

- drátovou,
- bezdrátovou,
- kombinovanou.

Pokud má být domácnost komplexně automatizovaná, je často vhodnější mít celý systém propojený pomocí kabelů, jelikož takový systém bude spolehlivější a v případě potřeby nabízí rychlejší přenos dat (například pokud mají být součástí systému multimédia). Bezdrátové systémy se hodí zejména tam, kde není žádané zasahovat do elektroinstalace, či pokud uživatel potřebuje pouze jednodušší systém s ovládním několika málo zařízení. Připravená kabeláž pro automatizaci domácnosti rovněž přináší výhodu snadnějšího řešení napájení jednotlivých chytrých zařízení, které se tak může rozvádět po bytě spolu s datovými kabely. Přitom pro propojení jednotlivých chytrých zařízení mezi sebou je možné využít různé typy kabelů (např. ethernetový). Systémy s kombinovanou kabeláží pak vycházejí z klasické kabelové instalace s možností použití některých bezdrátových prvků (např. snímačů).

Kromě kabeláže můžeme systémy chytré domácnosti také rozdělit na základě toho, zda je systém komplexní (a vytvářený na míru specializovanou firmou) nebo sestavený uživatelem z různých částí na základě jeho potřeb [27]. Nevýhodou systémů od specializovaných firem je především obvykle mnohem vyšší cena. Systémy sestavené uživatelem mohou využívat nějakého open source projektu a cena je tak daná jen použitými komponentami, které mohou být nepoměrně levnější oproti těm od specializovaných firem. Samozřejmě má toto řešení řadu nevýhod, jelikož si uživatel musí celý systém navrhnout a sestavit sám, a je tedy sám zodpovědný za jeho funkčnost. Také můžeme systémy rozdělit na ty, které je možné ovládat hlasem a které nikoliv [8].

Funkce používané v chytrých domácnostech

V chytré domácnosti se uplatňují různé funkce [26]. Některé systémy mohou mít svůj název pro danou funkci, ale obecně se jedná o následující funkce:

- přímé ovládání spotřebičů,
- nastavení scény,
- automatické akce.

Přímé ovládání spotřebičů se může provádět dálkovým ovládním, kde se často využívá rádiové komunikace na frekvenci 433.92 MHz. Příkladem takového zařízení mohou být bezdrátové zásuvky od společnosti Emos³. Jiný způsob přímého ovládním rovněž zahrnuje použití jiného chytrého zařízení (například chytrého telefonu), pokud ovládané zařízení umí komunikovat pomocí stejné technologie (Wi-Fi či Bluetooth). Ovládním pomocí telefonu či podobného chytrého zařízení je možné i v případě, že ovládané zařízení neumí komunikovat stejnou technologií, ale v domácnosti existuje centrální prvek (tzv. hub), který podporuje

³<https://www.alza.cz/emos-dalkove-ovladana-zasuvka-bila-d5640290.htm>

obě technologie a funguje zde jako prostředník mezi oběma zařízeními. Funkce nastavení scény obvykle jistým způsobem sdružuje několik příkazů přímého ovládní. Může se jednat o scénu odchodu z domu, která vypne všechna světla, odpojí spotřebiče od elektrické sítě a aktivuje zabezpečovací systém. Některé chytré domácnosti mohou rozpoznat i večerní zapnutí televize a nastavit komfortním způsobem osvětlení. Dalším principem uplatňovaným v chytré domácnosti je vlastní automatizace pomocí předem definovaných podmínek, tedy automatické akce. Ty spočívají v tom, že pokud dojde ke splnění předem definované podmínky, tak systém zareaguje předem nastavenou akcí. Například může systém díky nastaveným podmínkám udržovat teplotu v domácnosti na určité úrovni nebo třeba zapnout venkovní osvětlení při detekci osoby za tmy.

Zde stojí za zmínku webová služba IFTTT (If This Then That), která přidává nastavování podmínek poněkud širší rozměr. Podle oficiálních stránek služby⁴ umožňuje propojit i spolu nijak nesouvisející služby či zařízení, které spolu běžně nekomunikují.

Komponenty chytré domácnosti

Na trhu dnes existuje nepřeberné množství různých systémů. Tyto systémy se mezi sebou liší složitostí, cenou, dosahem komunikace apod. Chytrá domácnost může obsahovat některé z následujících komponent [26]:

- vstupní prvky (různá čidla, tlačítka, dotykové displeje...),
- výstupní prvky (světla, spotřebiče a různá zařízení),
- centrální jednotka,
- aplikace pro řízení domácnosti z chytrých zařízení (telefonu, tabletu, počítače...).

Systém nemusí obsahovat všechny zmíněné komponenty. Záleží na komplexnosti a složitosti daného systému. Některé systémy mohou jako centrální jednotku využívat virtuálního (hlasového) asistenta jako je Google Assistant či Amazon Alexa [11].

2.2 Komunikační protokol CoAP

CoAP protokol je protokol, určený pro komunikaci mezi uzly (zařízeními) s omezenými zdroji v omezené síti (tedy např. s vysokou ztrátovostí paketů) [53]. Je definován v RFC 7252, ze kterého pochází některé informace uvedené v této podkapitole. Tento protokol je speciálně navržený zejména pro tzv. machine-to-machine (M2M) komunikaci v rámci internetu věcí. CoAP protokol se svojí strukturou podobá aplikačnímu protokolu HTTP, ačkoli každý z těchto protokolů má jiné zaměření. Mezi velkou výhodou CoAP protokolu patří jeho malá režie a jednoduchost [5].

Používané principy

Jak už bylo zmíněno, tento protokol je velmi podobný protokolu HTTP. Podobně jako při komunikaci přes HTTP i v CoAP existují 2 role, které zařízení může zastávat – klient a server. Klient posílá požadavek na server a ten vrací odpověď. Na rozdíl od HTTP je však při použití CoAP běžné, že zařízení vystupuje v obou rolích (vzhledem k povaze

⁴<https://ifttt.com/>

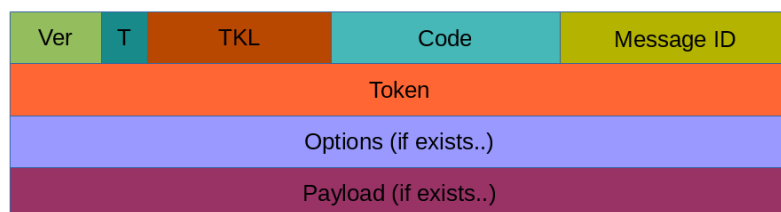
M2M interakce). V jednu chvíli tedy může zařízení jako klient posílat požadavky jinému zařízení a následně samo v roli serveru obsluhovat klienty. Dalším rozdílem oproti HTTP je rovněž to, že se jedná o asynchronní protokol (používá UDP protokol). Z toho důvodu je nutné nějakým způsobem zaručit doručení dat (pokud je to pro danou aplikaci nutné). CoAP protokol proto definuje 4 typy zpráv:

- confirmable (potvrditelná),
- non-confirmable (nepotvrditelná),
- acknowledgment (potvrzení),
- reset.

Jelikož je CoAP protokol navržen pro ztrátovou síť, počítá se s jistými ztrátami jednotlivých zpráv. Protokol tedy rozlišuje mezi potvrditelnou a nepotvrditelnou zprávou. Pokud příjemce obdrží potvrditelnou zprávu, musí odpovědět zprávou typu acknowledgment (potvrzení). V případě, že vyprší časový limit a odesílatel neobdrží tuto odpověď, znovu odešle původní zprávu. Toto provádí dokud není dosaženo maximálního počtu opakovaných přenosů [38]. Tento počet je definován dokumentem RFC 7252 v rámci přenosových parametrů a je roven číslu 4, ale je možné jej (spolu s ostatními parametry) upravit pro danou aplikaci. Odesílatel vyhodnotí případnou přijatou zprávu *potvrzení* jako odpověď podle identifikátoru zprávy, který se posílá spolu se zprávou. Ten musí být v odpovědi stejný jako v původním požadavku. Pokud zařízení posílá nepotvrditelnou zprávu, neočekává se, že příjemce pošle zpět potvrzení. Zpráva typu reset se používá, pokud chce příjemce odesílateli sdělit, že potvrditelnou zprávu nedokáže zpracovat, resp. ani vrátit vhodnou chybovou odpověď.

Formát CoAP zprávy

Formát CoAP zprávy je možné vidět na obr. 2.3.



Obrázek 2.3: Formát CoAP zprávy⁵

CoAP zpráva je zakódovaná pomocí binárních dat. Začíná povinnou 4-bitovou hlavičkou, obsahující informaci o:

- verzi CoAP protokolu (Ver) – 2 bity,
- typ zprávy (T) – 2 bity.

Verze protokolu musí být nastavena na 1 (zbylé možnosti jsou určeny pro budoucí verze protokolu). Typ zprávy je myšlen z hlediska potvrditelnosti, jak bylo definováno výše. Jednotlivým hodnotám odpovídají následující typy:

⁵Převzato z <https://www.survivingwithandroid.com/wp-content/uploads/2018/11/coap-message-format.png>

- 0 – confirmable,
- 1 – non-confirmable,
- 2 – acknowledgment,
- 3 – reset.

Po typu zprávy je ve zprávě specifikována délka přítomného tokenu (4 bity). Na základě této délky může mít token 0-8 bajtů (délka 9-15 je rezervována a nesmí být nastavená). Následuje kód zprávy, kterým může být buď metoda požadavku (CoAP je totiž podobně jako HTTP postaven na REST architektuře [10]), nebo stavový kód odpovědi, který má opět podobný účel jako stavové kódy u HTTP protokolu. Tento kód má délku 8 bitů. Poslední součástí hlavičky CoAP zprávy je identifikátor zprávy, který má délku 16 bitů. Používá se k identifikaci duplikátních zpráv a rovněž ke spojení potvrditelné zprávy s potvrzením (resp. reset zprávou). Po hlavičce následuje token, který má délku 0-8 bajtů, jak bylo vysvětleno dříve. Tento token se používá ke spojení požadavku a odpovědi. Při komunikaci totiž může odesílatel poslat požadavek na server, na který sice může server ihned odpovědět potvrzením (zde tedy posílá stejný identifikátor zprávy, jaký měl požadavek), ale daný požadavek vyřeší až později. A tehdy posílá server klientovi stejný token, jako obdržel v původní zprávě od klienta, ačkoli identifikátor zprávy je již jiný. To je tedy rozdíl mezi identifikátorem a tokenem. Po tokenu následují tzv. options, tedy volby, které mají podobnou funkci jako některé hlavičky v HTTP protokolu. Tyto volby mohou mít různou délku a rovněž jejich počet může být různý (ani nemusí být přítomny). Poslední částí CoAP zprávy tvoří nepovinný datový obsah (payload).

2.3 MQTT protokol

MQTT (Message Queuing Telemetry Transport protokol) je aplikační protokol, určený podobně jako CoAP pro přenos dat v nestabilní síti. Na rozdíl od CoAP se však nejedná o protokol typu klient – server, ale používá tzv. publish – subscribe mechanismus [58]. Jednotlivé uzly v síti mohou mít jednu ze tří funkcí:

- publisher,
- subscriber,
- MQTT broker.

Publish – subscribe mechanismus

Mechanismus publish – subscribe je založen na tom, že v systému existují zařízení, která sdílí nějaké údaje (publish) [41]. V síti je přítomen MQTT broker, což je zařízení, které má funkci centrálního prvku, přes který se odehrává komunikace.

Pokud chce nějaké zařízení naslouchat novým informacím od jiného zařízení, musí se přihlásit k odběru jeho zpráv (u MQTT brokeru). Stane se tak z něj subscriber daných zpráv. Zmíněné druhé zařízení, které bude později zprávy posílat zde plní roli publishera. Identifikace zpráv, kterým bude zařízení naslouchat, se pak provádí přes tzv. topic (téma). To specifikuje cestu k danému zdroji a má hierarchickou strukturu. Může mít podobu například „domov/kuchyně/světlo“. Jelikož jsou témata v MQTT zprávě zakódovány v UTF-8, mohou používat dokonce diakritiku.

Pokud následně zařízení typu publisher získá nějaký údaj, který sdílí (např. hodnotu měřené veličiny), odešle zprávu MQTT brokeru. Ten zprávu přešle všem zařízením, které jsou přihlášeny k odběru daného tématu. Jedno zařízení může samozřejmě vystupovat pro určitá témata jako publisher a pro jiná jako subscriber.

Quality of Service

Jelikož je protokol určen pro funkci i v nestabilní síti, definuje 3 stupně tzv. QoS (Quality of Service) [4]:

- 0 – At most once,
- 1 – At least once,
- 2 – Exactly once.

Stupeň 0 říká, že odesílatel zprávu odešle a zahodí, nekontroluje tedy, zda zprávu přijali příjemci [49]. V podstatě se jedná o obdobu nepotvrditelné zprávy u CoAP protokolu. Stupeň 2 rozhoduje, že odesílatel po odeslání zprávy čeká na potvrzení od příjemce, že zprávu dostal. V případě, že vyprší stanovený časový limit a toto potvrzení neobdrží, tak zprávu odesílá znovu. Poslední stupeň se používá, pokud chce mít odesílatel jistotu, že příjemce dostal (resp. zpracoval) zprávu přesně jednou. K tomuto účelu se používá čtveřice zpráv. Nejprve odesílatel posílá sdílená data. Následně čeká na potvrzení. Pokud od příjemce nepřijde potvrzení, posílá zprávu znovu se značkou definující, že se jedná o duplicitní zprávu. Po přijetí potvrzení posílá příjemci další zprávu jako potvrzení, že obdržel potvrzení. Nakonec příjemce posílá zpět zprávu pro potvrzení, že byl přenos dat úspěšně dokončen.

2.4 Bezdrátový přenos dat a signálů

Pro přenos dat či řídicích signálů je vždy potřeba zvolit vhodné médium, přes které se budou tyto informace přenášet. V některých situacích není pro přenos vhodné (a někdy dokonce ani možné) používat kabely (ať už metalické nebo optické). V těchto případech je potřeba přenášet informace bezdrátově, tj. za využití jiných médií, jako je vzduch [37]. Podobně jako je nutné u kabelového spojení využít vhodný způsob komunikace (například zvolit vhodnou sběrnici a nastavit ji správné parametry), je potřeba se způsobem komunikace zabývat rovněž u bezdrátového přenosu. Zde je nutné zejména zvolit vhodnou technologii (jako je Wi-Fi, Bluetooth či ZigBee) a její parametry.

Výhody bezdrátového přenosu

Bezdrátová komunikace má oproti kabelové řadě výhod. Zejména se jedná o následující:

- Jednodušší připojení – zařízení není potřeba připojovat kabelem a nemusí tedy být vybaveno konektorem pro toto připojení. Z toho rovněž plyne, že není potřeba měnit strukturu sítě kvůli změnám v místnosti a rovněž není potřeba myslet na konkrétní strukturu sítě ještě před budováním.
- Větší spolehlivost – častým zdrojem problémů s kabelovým připojením jsou chyby na straně kabelů – jejich poškození. Použitím bezdrátových technologií se lze vyhnout tomuto typu chyb.

- Snadná rozšiřitelnost sítě – u kabelového připojení je potřeba řešit způsob rozšíření sítě a v případě, že stávající struktura sítě rozšíření nepodporuje, tak je potřeba ji celou pozměnit. Bezdrátové sítě tento problém eliminují.
- Nižší cena – použitím bezdrátových technologií se může značně snížit pořizovací cena sítě – není potřeba kupovat drahou kabeláž. Rovněž instalace kabelů do starých budov může být velmi nákladná a problémová.

Kabelové spojení se může časem opotřebovávat a způsobit problémy. Větší spolehlivostí bezdrátové komunikace se zde naráží na tento jev, který pochopitelně bezdrátovou komunikaci neprovází. Samozřejmě celkově vzato mohou být kabelová spojení spolehlivější (zejm. pokud se přenáší velké množství dat) [26].

Nevýhody bezdrátového přenosu

Kromě množství výhod, které bezdrátová komunikace představuje jsou zde rovněž některé nevýhody tohoto typu komunikace [54]. Jde například o:

- Rušení signálu – zařízení, využívající bezdrátové technologie, může způsobovat rušení ostatních zařízení a rovněž opačně – dané zařízení může být rušeno od ostatních zařízení, pracujících na podobném principu.
- Bezpečnost – bezdrátová komunikace často vysílá (a přijímá) signály do relativně rozsáhlého otevřeného prostoru, tudíž jsou takto vysílaná data často daleko méně chráněná než u kabelového přenosu (kde je k získávání dat potřeba mít fyzické připojení k síti, ve které se data přenáší). Je tedy nutné zabezpečit přenos dat.

Toto bylo tedy shrnutí nevýhod bezdrátové komunikace a zbývající část této kapitoly se věnuje již konkrétním existujícím řešením chytrých domácností.

2.5 Existující řešení chytrých domácností

Dnes je na trhu velké množství různých systémů automatizace domácnosti [26]. Ty se mohou lišit v ceně, komplexnosti, způsobu komunikace a dalšími parametry. Není možné na jednotlivé systémy pohlížet stejně, protože každý systém má své vlastnosti a některé systémy je jen stěží možné srovnávat jako alternativy. Při výběru systému vždy záleží na potřebách konkrétního uživatele. Některé systémy mezi sebou dokáží dokonce komunikovat a spolupracovat. Například Apple HomeKit, někteří hlasoví asistenti či dříve zmíněná služba IFTTT, pomocí které je možné propojit některé systémy.

Jednou z nejčastějších aplikací automatizace, kterou různé systémy nabízejí, je ovládání světel a zásuvek. Mezi další aplikace patří ovládání hlavic radiátorů, chytré termostaty, ovládání ventilátorů, stínící techniky, alarm a jiné. Níže uvádím seznam některých existujících řešení:

- Loxone,
- Jablotron,
- Kangtai,
- Apple HomeKit,

- Home Connect.

Samozřejmě se jedná jen o příklady a podobných systémů existuje velké množství. Kromě komerčně prodávaných systémů jsou k dispozici rovněž open source řešení, mezi známější patří například:

- Home Assistant⁶,
- ESPHome⁷.

I open source řešení samozřejmě existuje mnohem více. Jednotlivé systémy se mezi sebou různým způsobem liší. Každý systém má svůj způsob přidávání nových zařízení, podporu, rozšiřitelnost, architekturu apod. V jistém smyslu speciálním typem systémů chytré domácnosti je systém, využívající hlasového asistenta. Takový systém obvykle bývá snadno rozšiřitelný.

Loxone

Loxone je společnost, zaměřující se na automatizaci budov v rozsahu od malých bytů, přes hotely, až po rozsáhlé budovy a výrobní haly. Informace o této společnosti a jejich systému byly získány z oficiálních stránek společnosti⁸.



Obrázek 2.4: Ukázka vzhledu aplikace Loxone App na různých zařízeních⁹

Loxone se zaměřuje na širokou škálu aplikací [36]. V oblasti automatizace domácnosti jde zejména o následující:

- bezpečnost (pohybové senzory, dveřní a okenní senzory),

⁶<https://www.home-assistant.io/>

⁷<https://esphome.io/>

⁸<https://www.loxone.com>

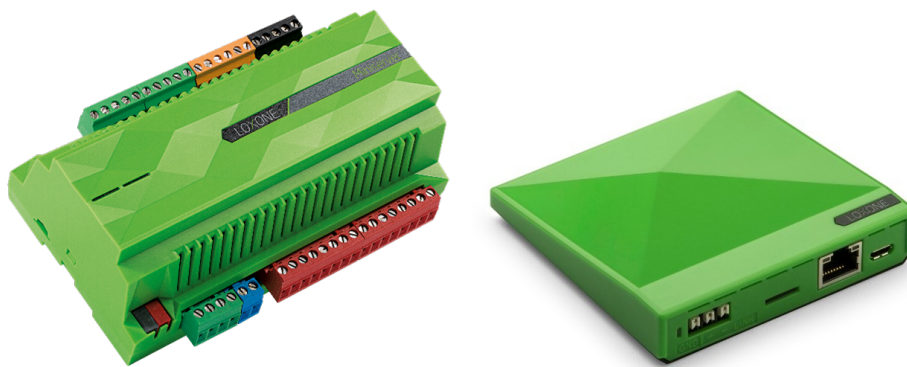
⁹Převzato z <https://www.loxone.com/cscz/o-nas/press/>

- přístup do budovy (přístup kódem zadávaným na klávesnici, NFC přívěškem či iButtonem, kamera),
- řízení filtrace bazénu,
- větrání (automatické řízení ventilace, například na základě přítomnosti osoby, vlhkosti, teplotě apod.),
- regulace teploty (Loxone je dle stránek výrobce možné připojit k jakémukoli zdroji teploty i chlazení),
- úspora energie (ovládání budovy k úsporám energie, např. automatické stínění jako ochrana přetopení ze slunečního tepla),
- osvětlení (ovládání bodového světla, LED pásků či Loxone závěsných světel),
- multimédia (ovládání audia, TV),
- stínění (ovládání stínící techniky pomáhá při vytápění a chlazení v domě).

System od Loxone pro automatizaci domácnosti se skládá z několika různých prvků:

- Miniserver,
- rozšíření,
- příslušenství (Loxone Tree zařízení),
- Loxone Tree a Loxone Link kabeláž,
- aplikace Loxone App a Loxone Config.

Loxone prvky ke své činnosti potřebují tzv. Miniserver. Ten v systému funguje jako centrální řídicí jednotka, která se stará o automatizaci domácnosti.



Obrázek 2.5: Centrální řídicí jednotky systému Loxone Miniserver gen. 1 (vlevo) a Miniserver Go (vpravo)¹¹.

Loxone nabízí celkem 2 různé typy Miniserveru, každou aktuálně v jedné ze dvou generací – Miniserver a Miniserver Go. Obě verze je možné vidět na obrázku 2.5. Miniserver

¹¹Převzato z <https://www.loxone.com/cscz/o-nas/press/>

1. i 2. generace mají oba 8 digitálních a 4 analogové vstupy a 8 digitálních výstupů (relé spínající maximálně 250 VAC/30 VDC). Miniserver 1. generace má ještě navíc 4 analogové výstupy [46]. Obě generace miniserveru slouží pro kabelovou komunikaci a jsou určeny k instalaci na DIN lištu. Také jsou obě generace vybaveny rozhraním Loxone Link (pro kabelové připojení až 30 tzv. rozšíření) a LAN port (Fast ethernet). Pouze první generace obsahuje KNX rozhraní, naopak pouze druhá generace a verze Go obsahují již integrované rozhraní Loxone Tree (k první generaci je pro komunikaci přes Loxone Tree sběrnici nutné přidat rozšiřující modul) [45].

Pokud si uživatel přeje využívat bezdrátové komunikace mezi prvky systému Loxone (zejména pro bezdrátové ovládání), pak Loxone nabízí 2. typ miniserveru – Miniserver Go [47]. Ten komunikuje s bezdrátovými periferiemi (rozšířeními a příslušenstvím) rádiovou komunikací na frekvenci 868 MHz pro SRD pásmo pro Evropu (na 4 kanálech), případně 915 MHz pro ITU region 2 (10 kanálů) s maximálním výkonem 3.16 mW. Obsahuje také LAN port (Fast Ethernet) a rozhraní Loxone Link. K této verzi miniserveru je možné bezdrátově připojit až 128 periferií [65]. Druhá generace tohoto miniserveru pak přináší zejména výrazně vyšší výkon a bezpečnost [28].

Všechny verze Miniserveru v sobě obsahují Loxone OS¹² s integrovaným webovým serverem, jsou konfigurovatelné z programu Loxone Config a ovladatelné přes mobilní aplikaci Loxone App¹³. Tu je možné spatřit na obrázku 2.4. Všechny verze Miniserveru rovněž obsahují slot pro SD kartu (s firmwarem).

Loxone Extensions (rozšíření) slouží pro rozšíření funkcí Miniserveru. K Miniserveru se připojují pomocí sběrnice Loxone Link (kterou obsahují všechny verze Miniserveru). Tato sběrnice může být až 500 m dlouhá. Díky rozšířením může uživatel zakoupit systém pouze s těmi technologiemi, které chce opravdu využívat a nemusí tak platit za zbytečné vlastnosti systému. Příkladem rozšíření může být Tree Extension. Ten slouží pro připojení až 100 Tree zařízení. Je určen zejména pro doplnění Miniserveru 1. generace, který neobsahuje rozhraní pro komunikaci přes tree sběrnici [57]. Dalším příkladem rozšíření je Air Base Extension. Doplnuje Miniserver 1. a 2. gen. o bezdrátovou komunikaci [1]. Mezi rozšířeními je možné najít i Dimmer Extension, který slouží pro stmívání světel [12]. Rozšíření však Loxone nabízí mnohem více¹⁴. Většina z nich se cenově pohybuje v řádu jednotek tisíc Kč.

Loxone nabízí pro automatizaci domácnosti více než 400 produktů [36]. Pro propojení prvků v systému společnost vyvinula tzv. Loxone Tree technologii. Jedná se o sběrnici, na kterou je možné připojit až 50 prvků a Loxone uvádí, že díky tomu je možné ušetřit až 80 % kabeláže. Podobně jako Loxone Link i Loxone Tree může sahát až 500 m daleko.

V oblasti inteligentního vytápění nabízí Loxone souhrn technologií pro vytápění, chlazení, rekuperaci, a automatizovanou stínící techniku, což přináší do regulace vytápění vysokou efektivitu¹⁵.

Z hlediska regulace teploty nabízí tzv. „zónové“ vytápění. Jedná se o inteligentní topení, které na rozdíl od klasického vytápění (zahrnující nějaký bezdrátový termostat, Wi-Fi termostatické hlavice apod.) umožňuje inteligentněji řídit teplotu tím, že uživatel zvolí, ve které místnosti (případně i ve který čas) má být jaká teplota. Uživatel chytrého domu s tímto systémem si tak může zvolit například větší teplo v koupelně oproti místnosti, kde spí. Tento systém tak umožňuje mít větší kontrolu nad vytápěnými místnostmi.

¹²OS – Operating system (operační systém)

¹³<https://www.loxone.com/cscz/kb/miniserver-go/>

¹⁴<https://shop.loxone.com/cscz/extensions.html/>

¹⁵<https://www.loxone.com/cscz/produkty/topeni-klimatizace/>

Inteligentní vytápění Loxone podporuje režim učení. Systém tedy na základě předchozích zkušeností spustí vytápění tak, aby byla v dané místnosti požadovaná teplota ve správný čas. Uživatel si může nastavit například to, aby měl v 7:00 vyhřátou koupelnu na 23°C. Loxone vytápění má dle oficiálních stránek¹⁶ společnosti následující výhodné vlastnosti:

- Inteligentní řízení teploty – využití již zmíněného režimu učení k dosažení požadované teploty v žádaný čas. Loxone rovněž při regulaci zohledňuje venkovní teplotu.
- Úspora nákladů – Loxone dokáže inteligentně rozhodovat o nejefektivnějším řešení. Například energeticky náročnou klimatizaci může nahradit energeticky výhodnějším stíněním.
- Režim nepřítomnosti – systém od Loxone podporuje úsporný režim pro chvíle, kdy uživatel není doma.
- Ochrana budovy – Loxone dokáže reagovat na různá nebezpečí, například v případě vzniku požáru vypnout ventilaci i rekuperaci.
- Loxone aplikace a statistiky – Loxone nabízí zdarma aplikaci na zařízení s androidem, přes které uživatel může sledovat i nastavovat teplotu v domě vzdáleně.
- Notifikace – v případě problému s některou technologií Loxone upozorní uživatele.
- Státní svátky – na základě znalosti státních svátků může Loxone adekvátně upravovat svoji činnost.
- Údržba – Loxone uživatele upozorňuje na termín pravidelné údržby.

Z hlediska automatizace domácnosti je rovněž důležité, že Loxone nabízí i chytré zásuvky. Ty s Miniserverem komunikují technologií Loxone Air. Mají v sobě teplotní čidlo a také elektroměr s vyhodnocením výkonu a spotřeby.

Jablotron

Jablotron je česká firma, která se od svého založení zaměřuje především na zabezpečovací systémy. Kromě nich se také zabývá zabezpečením a monitoringem vozidel, topením a ventilací, monitoringem dechu a rovněž ovládáním a automatizací domácnosti. Informace o společnosti a jejich systémech byly získány hlavně z oficiálních stránek společnosti¹⁷.

Jablotron nabízí několik různých systémů. Dva nejnovější jsou Jablotron 100 a Jablotron 100+. Primárním úkolem obou systémů je zabezpečení budov, ovšem je možné je využít i v oblasti automatizace (zejména díky programovatelným výstupům) [32]. V rámci automatizace je možné využít také samotné zabezpečení (automatické zapnutí světel při odkódování alarmu). Na své systémy poskytuje Jablotron při splnění podmínek až 7letou záruku¹⁸.

Pro odjištění/zajištění systému se vždy musí provést autorizace uživatele. Systém totiž uchovává informaci o oprávnění jednotlivých uživatelů. Každému z uživatelů je možné pro účely autorizace přiřadit jeden kód (4, 6 nebo 8místný) a až dva RFID čipy.

¹⁶<https://www.loxone.com/cscz/produkty/topeni-klimatizace/>

¹⁷<https://www.jablotron.com/cz/>

¹⁸<https://www.jablotron.com/cz/produkty/alarmy/alarm-do-domu/>

Funkce, které je možné v těchto systémech použít jsou:

- zapínání a vypínání programovatelných výstupů,
- akce v kalendáři (časovače),
- automatické akce (reakce na snímače).

Mezi akce, které lze automatizovat v systému Jablotron, patří zejména ovládání světel, ovládání žaluzií, chytrá termoregulace (řízení vytápění a klimatizace) či ovládání jiných zařízení pomocí programovatelných výstupů [48].

Systém od Jablotronu se skládá z následujících součástí:

- ústředna,
- různé vstupní či výstupní prvky,
- aplikace MyJablotron,
- program J-Link.

Ústředna v systémech Jablotron slouží jako centrální prvek, který shromažďuje informace ze snímačů a patřičně na ně reaguje. Komunikace mezi prvky systému a ústřednou může probíhat podobně jako u systému Loxone buď pomocí kabelů, nebo bezdrátově. Zařízení, která komunikují pomocí kabelu se zde nazývají sběrníkové [32]. Mezi produkty firmy Jablotron pro automatizaci domácnosti je možné najít:

- záplavový detektor,
- snímač teploty,
- magnetický detektor (detekce otevření dveří/okna),
- termoelektrická hlavice,
- relé na DIN lištu ,
- a další¹⁹.

Systém Jablotron 100+ je možné ovládat celkem čtyřmi způsoby, a to:

- přístupovým modulem,
- mobilní aplikací pro chytré telefony (MyJablotron),
- webovou aplikací (rovněž MyJablotron),
- či klíčenkou²⁰.

Přístupový modul slouží pro rychlé odjištění/zajištění objektu, případně k dalším funkcím automatizace. Jablotron nabízí celkem 3 typy těchto přístupových modulů:

- čtečka RFID karet,

¹⁹<https://www.jablotron.com/cz/katalog-produktu/>

²⁰<https://www.jablotron.com/cz/produkty/chytre-ovladani/ja-100-pg/>

- klávesnice se čtečkou RFID karet,
- klávesnice s displejem a čtečkou RFID karet.

Ke každému z modulů je možné připojit až 20 segmentů. Ty obsahují popisek a dvě prosvětlená tlačítka. Jejich funkcí může být buďto zajištění/odjištění, signalizace stavu (například signalizace otevření garážových vrat) nebo ovládání zařízení v rámci automatizace (například žaluzií) [32]. Barvy prosvětlení odpovídají semaforu, kde červená odpovídá stavům jako zajištěno/zapnuto, žlutá zajištěno částečně a zelená znamená odjištěno/vypnuto. Přístupový modul je možné vidět na obrázku 2.6.



Obrázek 2.6: Přístupový modul Jablotron²¹.

Jak již bylo zmíněno, systém od Jablotronu lze ovládat rovněž mobilní aplikací My-Jablotron. Je k dispozici jak na Google Play (pro zařízení s androidem), tak i na App Store (pro iOS zařízení). Kromě toho existuje i její webová verze. Jablotron tak nabízí rychlý přehled o tom, co se děje v domácnosti. Ovládání domácnosti přes aplikaci funguje podobným způsobem jako přístupový modul – pomocí tlačítek s barvami semaforu.

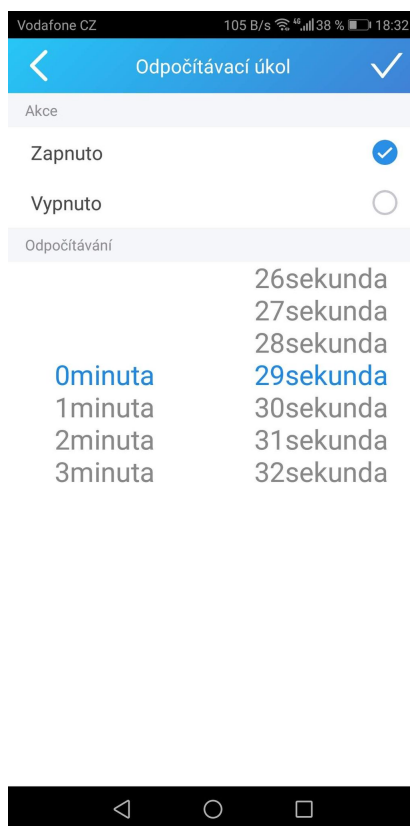
Klíčenka k ovládání systému je dostupná ve dvou verzích – jednosměrný a obousměrný ovladač. Použití toho druhého má za výhodu to, že provedení akce je potvrzeno kontrolkou na ovladači. V případě chyby tak uživatel ví, že je mimo dosah ústředny a akce se neprovedla [32].

K nastavení uživatelských parametrů v systému (jako je oprávnění) slouží program J-Link [31]. V něm je možné definovat uživatele i s jejich přístupovými oprávněními, provádět diagnostiku systému, kontrolu programovatelných výstupů a vytvářet či upravovat kalendář akcí (pro ovládání automatizovaných funkcí).

²¹Převzato z <https://www.tzb-info.cz/poplachove-a-zabezpecovaci-systemy/14937-vyjimecny-bezpecnostni-alarm-jablotron-100>

Kangtai

Kangtai je společnost, která se mj. zaměřuje na výrobu chytrých zařízení, jak uvádí oficiální web společnosti²².



Obrázek 2.7: Ukázka nastavení časovače pro Wi-Fi zásuvku Kangtai.

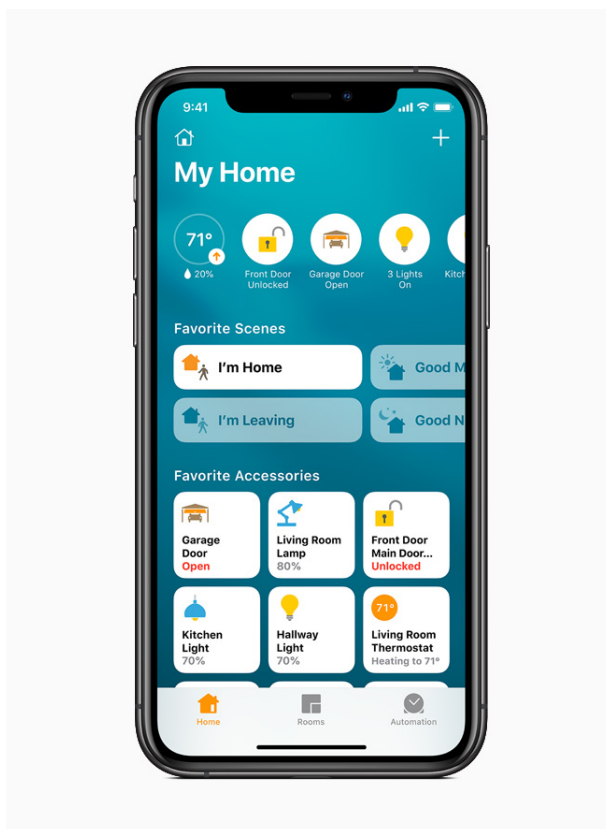
Vyrábějí zejména chytré zásuvky, zvonky a zkoušečky napětí. Jejich produkty využívají technologie Wi-Fi, Zigbee a rádiovou komunikaci na frekvencích 433.92 MHz a 868.3 MHz.

Apple HomeKit

Apple HomeKit je systém, který umožňuje uživateli ovládat nejrůznější chytrá zařízení v domácnosti. Na rozdíl od systémů jako je Loxone, je HomeKit orientován spíše na bezdrátovou komunikaci (ačkoliv je možné používat i komunikaci pomocí kabelů). V základu je systém založen na komunikaci pomocí Wi-Fi a Bluetooth [62]. Pro podporu dalších přenosových technologií je potřeba do systému přidat tzv. bridge, který potom slouží jako prostředník mezi zařízeními. Zařízení v systému mezi sebou komunikují pomocí aplikačního protokolu HAP. Pokud dané koncové zařízení tento protokol nezná, tak s ním komunikuje bridge zařízení pomocí protokolu, kterému rozumí, takže i v tomto případě bridge funguje jako prostředník.

Systém Apple HomeKit je možné ovládat pomocí chytrého telefonu iPhone. V tomto případě je však možné ovládání pouze v lokální síti. Druhou variantou je ovládání přes Internet, ale to již vyžaduje použití nějakého centrálního prvku (např. Apple TV či iPad).

²²<http://www.kangtai.com.cn/lang/culture.asp>



Obrázek 2.8: Aplikace Domácnost, přes kterou je možné zařízení v systému Apple HomeKit ovládat²⁴.

Pro ovládání systému pomocí telefonu s operačním systémem Android je potřeba mít v systému nějakého hlasového asistenta, který je se systémem Apple HomeKit kompatibilní (např. Amazon Alexa či Google Assistant), jak uvádí společnost Allocacoc²⁵.

Mezi zařízení, která je možné v systému Apple HomeKit používat, patří např. nejrůznější tlačítka, světla, kamery, zásuvky a mnoho dalšího (viz seznam na stránkách společnosti Apple²⁶). Systém Apple HomeKit představuje otevřený systém, do kterého postupně přibývají nové prvky, přičemž se nejedná jen o zařízení od společnosti Apple [55]. Obchodníci s těmito zařízeními pak uvádějí u konkrétních zařízení, že jsou s Apple HomeKit kompatibilní, pokud tak tomu je.

Po přidání zařízení do systému (v aplikaci Domácnost²⁷), je možné zařízení ovládat jednak okamžitým nastavováním hodnoty [16], ale také je možné využít automatizace [17], která umožňuje zakomponovat některou z následujících situací:

- někdo přijde,
- někdo odejde,
- nastane určitá denní doba,

²⁴Převzato z <https://www.apple.com/nz/newsroom/2020/06/apple-reimagines-the-iphone-experience-with-ios-14/>

²⁵<https://www.vocolinc.cz/blog/jak-zacit-s-chytrou-domacnosti>

²⁶<https://www.apple.com/ios/home/accessories/>

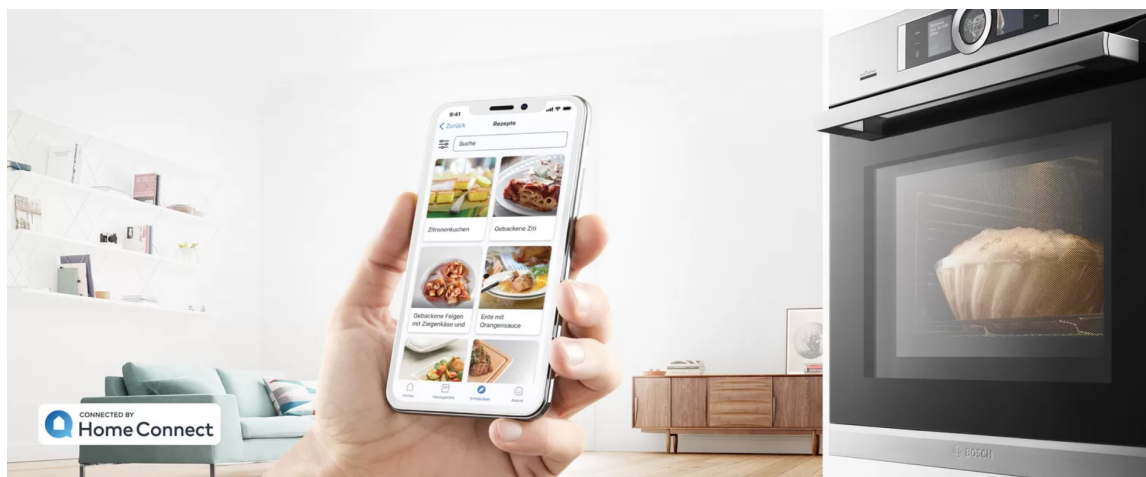
²⁷<https://support.apple.com/cs-cz/guide/iphone/iph22d98bbca/ios>

- změna stavu příslušenství,
- čidlo něco detekovalo.

Na zařízení s operačními systémy iOS, iPadOS, watchOS či macOS je možné přidaná zařízení ovládat pomocí aplikace Domácnost. Případně je také možné použít aplikaci Zkratky či ovládání pomocí hlasového asistenta Siri [18]. Nakonec je samozřejmě také možné jednotlivá chytrá zařízení ovládat pomocí aplikace od jejich dodavatele [16]. Protože jak bylo zmíněno, systém HomeKit podporuje i jiné výrobce.

Home connect

Poněkud jiný přístup k chytré domácnosti přináší technologie Home connect. Tou jsou vybavena některá zařízení, například od společnosti Bosh²⁸. Předchozí zmíněné systémy slouží k ovládání a automatizaci spíše jednoduchých prvků domácnosti, jako jsou světla, zásuvky, relé moduly apod. Naproti tomu technologie Home connect je přítomna v některých „složitějších“ spotřebičích, jako je například pračka, lednice či kávovar [7].



Obrázek 2.9: Aplikace Home Connect a chytrá trouba²⁹.

Aplikace³⁰, která s technologií Home connect přichází, je dostupná na operační systém Android, ale také pro hlasové asistenty Amazon Alexa či Google Assistant. Díky aplikaci je možné získávat některé informace o spotřebičích (např. zda nejsou dveře lednice pootevřené) a také zařízení ovládat (např. na dálku zapnout troubu a předeheat ji tak).

Virtuální hlasoví asistenti

Virtuální osobní asistent (VPA) je osobní asistent, který zajišťuje interakci mezi uživatelem chytré domácnosti a zařízeními v ní. Jako jiné označení se rovněž používá inteligentní, digitální osobní či mobilní asistent. Je-li ovládaný hlasem, pak se někdy označuje jako hlasový asistent [42]. V následujícím textu této kapitoly je vždy asistentem míněn právě hlasový asistent, není-li specifikováno jinak.

²⁸<https://www.bosch-home.com/cz/novinky/home-connect>

²⁹Převzato z <https://www.bosch-home.com/cz/novinky/home-connect/produkty>

³⁰<https://play.google.com/store/apps/details?id=com.bshg.homeconnect.android.release>

³²Převzato z <https://www.pcmag.com/reviews/amazon-echo-dot-4th-generation>



Obrázek 2.10: Amazon Echo Dot 4. generace, který obsahuje hlasového asistenta Amazon Alexu³².

Hlasový asistent je software, jehož úlohou je asistovat uživateli při nejrůznějších příležitostech, mezi jinými i při ovládání domácnosti. Hlasoví asistenti běží na některém zařízení s reproduktorem a mikrofony nebo mobilním zařízením. Dnes jich existuje na trhu velké množství (zejména pro chytré telefony). Mezi nejznámější představitele hlasových asistentů v současné době patří [29]:

- Apple Siri,
- Amazon Alexa,
- Microsoft Cortana,
- Google Assistant.

Hlasový asistent (resp. zařízení, které ho obsahuje) má obvykle funkci centrální jednotky, spojující a řídící jednotlivá zařízení. Pokud uživatel zamýšlí v systému hlasového asistenta používat, pak je vhodné se při výběru chytrých zařízení ujistit, zda daná zařízení podporují zvoleného hlasového asistenta³³.

V současné době žádný z výše uvedených hlasových systémů nepodporuje plně češtinu, nicméně Assistant od společnosti Google by ji v dohledné době mohl podporovat [60]. Zřejmě i Siri od Apple bude brzy podporovat češtinu [19]. Amazon Alexa pak sice nerozumí česky, ale již dokáže číst některé knihy v češtině. Pro českého uživatele jsou však zatím 2 možnosti – buďto používat asistenta v angličtině a spokojit se s případnou absencí některých funkcí, které nejsou v Česku podporované, nebo použít některého českého virtuálního asistenta, ovšem obvykle s omezenou funkcionalitou oproti jejich vyvinutějším protějškům. Mezi české hlasové asistenty patří například Emma. Vytvořil jej David Beck pomocí aplikace Zkratky (na systému iOS). Nativním virtuálním asistentem pro iPhone je Siry, ta však zatím neumí česky, jak již bylo zmíněno. A to se rozhodl změnit David Beck, jak uvádí na stránkách věnujících se asistentovi Emma³⁴. Nejedná se o samostatnou aplikaci, ale o zkratku v aplikaci Zkratky na systému iOS. Tato aplikace Zkratky umožňuje uživatelům sloučit různé akce do jedné zkratky. Celkově pro zkratku Emma nastavil více než 7 tisíc akcí a další se chystá přidávat. Systém v současné době podporuje kromě češtiny 6 dalších jazyků [6].

³³<https://www.alza.cz/jak-postavit-chytrou-domacnost>

³⁴<https://helloemma.cz/>

Každý z virtuálních asistentů má své vlastní specifikace. Jsou však úlohy, které je možné považovat za typické [29]. Následuje seznam funkcí, které mohou virtuální asistenti podporovat:

- číst a psát SMS a emailové zprávy, uskutečňovat hovory,
- nastavovat časové a kalendářové akce (časovače, upomínky atd.),
- odpovídat na některé základní informativní otázky (počasí, čas, převody jednotek apod.),
- ovládat média jako televizi či připojené reproduktory (pouštět filmy, hudbu),
- vyprávět vtipy a příběhy,
- konečně ovládat prvky chytré domácnosti.

Používání virtuálních osobních asistentů nejen, že umožňuje přistupovat k různým úkolům inovativním a interaktivním způsobem, ale v mnoha případech i zjednodušuje jinak relativně zdlouhavou činnost. Dobrým příkladem je manuální nastavení budíku (bez použití VPA). Na mobilním telefonu Nexus 5 je dle literatury [42] potřeba vykonat následující akce:

1. kliknout na tlačítko pro návrat na domovskou obrazovku (pokud se tam uživatel nenachází),
2. kliknout na ikonu hodin,
3. v otevřené aplikaci najít ikonu budíku a kliknout na ni,
4. kliknout na tlačítko „+“ pro přidání budíku,
5. nastavit hodinu, překliknout na volbu minuty a nastavit minuty,
6. potvrdit kliknutím na tlačítko „OK“.

Při použití hlasového asistenta je celá úloha značně zredukována pouze na aktivování asistenta a vyslovení požadovaného úkolu.

Bez použití VPA je dokonce řada úkolů nerealizovatelná. Například připomenout či udělat něco v okamžiku, kdy se uživatel vrátí domů – což je funkce, kterou někteří virtuální asistenti podporují [11]. Interaktivitu zajišťují virtuální asistenti i při automatizaci domácnosti. Pro její řízení není potřeba otevírat k tomu určené (a mnohdy jednoúčelové) aplikace, ale stačí vyslovit žádost, třeba i s jistou vzdáleností od zařízení s hlasovým asistentem a ten se již o vše postará. Kromě toho v sobě virtuální asistenti mohou mít i funkce sloužící přímo pro automatizaci domácnosti. Například nastavení podmínek, scénářů atd. Virtuální asistenti díky svým funkcím a vlastnostem k chytré domácnosti patří, ovšem je nutné si uvědomit, že ne v každém systému jsou potřeba (a některé systémy je dokonce ani nepodporují). Často bývají zabudováni do chytrého zařízení, plnicí funkci centrálního prvku (hubu), jako tomu je v případě zařízení Echo od společnosti Amazon, který je možné vidět na obrázku 2.10.

Kapitola 3

Vestavěné systémy a bezdrátové technologie

Tato kapitola je shrnutím současného stavu v oblasti vestavěných systémů a některých bezdrátových technologií. Ani tato kapitola není encyklopedickým výkladem problematiky, ale pouze souhrnem informací, které mají k práci bezprostřední vztah. První část se věnuje vestavěným systémům, jelikož se využívají při automatizaci domácnosti. Následně jsou uvedeny informace k Raspberry Pi a modulům společnosti Espressif systems. Následuje krátké seznámení s technologiemi Wi-Fi, Bluetooth a ZigBee, jelikož jsou to technologie, které se při automatizaci domácnosti a u vestavěných systémů používají. Protože je téma poněkud rozsáhlé, jsou zde zmíněny pouze informace v přiměřeném rozsahu pro tuto práci.

3.1 Vestavěné systémy

Vestavěný systém je možné definovat jako software spolu s počítačem, zabudovaným do nějakého zařízení takovým způsobem, že jej uživatel nevidí jako počítač [64]. Tento počítač je většinou jednoúčelový, určený pro předem navržené použití [34]. Tím se liší od univerzálních počítačů, které mohou poskytovat různé funkce a jejichž uplatnění se může měnit (jako je tomu u osobního počítače). Mezi hlavní charakteristiky vestavěných systémů patří [15]:

- Jsou jednoúčelové – vestavěné systémy jsou navrženy pro konkrétní účel a ten se v průběhu jejich funkce nemění.
- Jsou na ně kladena omezení – při návrhu vestavěných systémů se klade důraz zvláště na parametry jako jsou cena, velikost, spotřeba energie a výkon tak, aby měl výsledný systém ideální poměr těchto parametrů.
- Reakce v reálném čase – spousta vestavěných systémů musí neustále kontrolovat změny prostředí, pro které byly navrženy a musí na ně v reálném čase odpovídat.
- Jsou založené na mikroprocesorech či mikrokontrolérech.
- Mívají připojené vstupní a výstupní zařízení.

Vestavěný systém může buďto fungovat sám o sobě nebo být součástí jiného systému.

Architektura vestavěných systémů

Složitost jednotlivých vestavěných systémů se mezi sebou liší, nicméně většina z těchto systémů je tvořena 3 částmi [39]:

- hardware – jedná se o součásti systému jako je mikroprocesor, paměti a podobně,
- software – program, který vestavěný systém vykonává,
- real-time operační systém definuje, jakým způsobem systém funguje. U jednodušších systémů nemusí být přítomen.

Z hlediska hardwaru jsou pak systémy využívající vestavěný systém složeny z několika komponent. Snímače zde převádějí snímaná fyzická data na elektrický signál. Také jsou zde analogově-digitální převodníky, které převádějí analogový elektrický signál na digitální. Procesor v systému zpracovává snímaná data a ukládá je do paměti. Naopak digitálně-analogové převodníky převádějí digitální signál z procesoru na analogový a akční členy nakonec převádějí výstupní data na výstupní akci.

3.2 Jednodeskový počítač Raspberry Pi

Raspberry Pi je levný univerzální počítač malých rozměrů. Poskytuje široké možnosti v oblasti multimédií a 3D grafiky. Předpokládá se, že bude časem využíván i jako herní platforma [59].

Historie

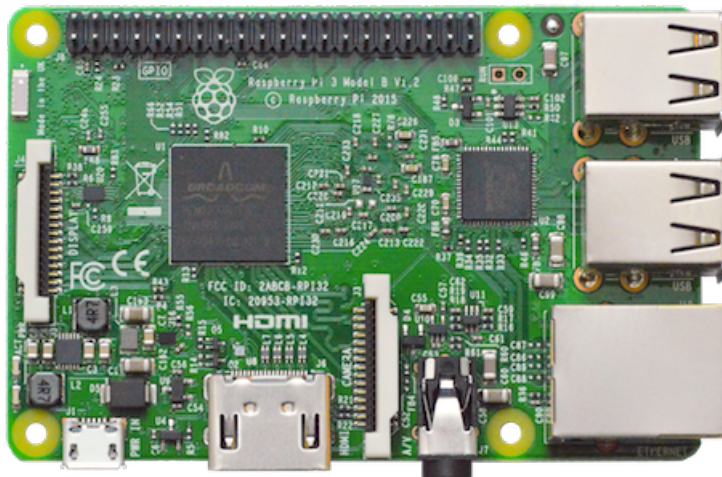
Raspberry Pi vzniklo v r. 2006 za přispění studijního ředitele pro informatiku na Cambridge-ské univerzitě za účelem lokálních potřeb. Měl to být nástroj, který by poskytl prvotní impuls studentů k některému z univerzitních kurzů [59]. Vytvořil jej univerzitní profesor Eben Upton. V roce 2009 založil nadaci Raspberry Pi Foundation pro budoucí vývoj Raspberry Pi. Jeho cílem bylo vyvinout levný počítač, na kterém by se mladí studenti mohli učit programovat [20]. Výsledkem byl počítač s cenou okolo 30ti dolarů, který měl školám pomoci ve výuce výpočetní techniky. Raspberry Pi se pak pro širokou veřejnost začalo prodávat oficiálně v únoru roku 2012.

Verze a modely Raspberry Pi

Jak již bylo zmíněno, při návrhu Raspberry Pi bylo přihlíženo zejména k ceně. Z tohoto důvodu hned u první oficiální verze vznikly 2 modely – A a B. Model B měl být výkonnější (a tedy dražší) variantou modelu A [20]. Verze 1, model B, měl následující výbavu:

- CPU – 700Mhz ARM procesor,
- RAM – 512MB,
- 2 USB porty,
- HDMI, RCA Audio a SD slot,
- 10/100 Ethernet,
- 8 GPIO pins.

Model A měl poloviční RAM paměť a jen jeden USB port. Od prvního vydání Raspberry Pi již vyšlo mnoho dalších verzí. Aktuálně nejvýkonnější verzí je 4, která je osazena čtyřjádrovým procesorem Cortex-A72 (1,5 GHz) a v závislosti na modelu je dostupná s RAM pamětí až 8GB³⁵.



Obrázek 3.1: Raspberry Pi, verze 3B³⁶.

Níže je uveden seznam některých zajímavých parametrů Raspberry Pi verze 3B+ na základě informací ze stránek Raspebrry Pi Foundation³⁷, protože tato verze Raspberry Pi je později použita v realizační části práce:

- čtyřjádrový procesor Cortex-A53 s frekvencí 1.4 GHz,
- 1 GB LPDDR2 SDRAM,
- HDMI,
- 4 USB porty (verze 2.0),
- Wi-Fi a Bluetooth.

Kromě jednodeskových počítačů vydala nadace Raspebrry Pi Foundation Raspberry Pico. Jedná se o mikrokontrolér (neobsahuje tedy OS) [40].

Operační systém

Na každou z verzí Raspberry Pi (kromě Pico) je možné nainstalovat některý z operačních systémů. Oficiálně podporovaným OS ze strany Raspebrry Pi Foundation je Raspberry Pi OS (dříve známý jako Raspbian)³⁸. Ten obsahuje některé předinstalované nástroje jako internetový prohlížeč (Chromium). Existuje několik verzí tohoto OS [44].

³⁵<https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>

³⁶Převzato z <https://developer.android.com/things/hardware/raspberrypi>

³⁷<https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>

³⁸<https://www.raspberrypi.org/software/>

Alternativy k Raspberry Pi

Přestože má Raspberry Pi za sebou úspěšný vývoj a jedná se o jednu z nepopulárnějších platform, na trhu samozřejmě existují alternativní počítače [9]. Někteří uživatelé mohou vyžadovat vyšší výkon, nižší cenu nebo některou kombinaci parametrů, kterou Raspberry Pi nenabízí. Mezi některé ze známějších alternativních jednodeskových počítačů patří:

- ASUS Tinker Board S,
- Banana Pi M64,
- Odroid-XU4.

Jednotlivé jednodeskové počítače se mezi sebou liší různými parametry jako je velikost a typ paměti RAM, použitý procesor, vstupní a výstupní porty, podporované OS apod.

3.3 Moduly společnosti Espressif Systems

Jak bylo uvedeno v úvodu této kapitoly, vestavěné systémy mohou být mj. založeny na mikrokontrolérech. Mikrokontrolérů je dnes na trhu celá řada a vzájemně se mezi sebou liší množstvím různých parametrů. Pro snazší vývoj se mohou jednotlivé mikrokontroléry zabudovat do tzv. vývojových desek (někdy označované také pod pojmem vývojový kit), které pak umožňují snadný přístup ke GPIO³⁹, sběrnícím a usnadňují nahrávání nového kódu do daného mikrokontroléru [30].

Moduly ESP od společnosti Espressif Systems patří mezi jedny z nejznámějších mikrokontrolérů. Vyznačují se nízkou cenou, Wi-Fi stackem a schopností provozu RTOS⁴⁰. Existují již celkem 4 řady (generace) těchto čipů:

- ESP8266,
- ESP32,
- ESP32-S2,
- ESP32-C3.

Všechny moduly obsahují 32bitový mikroprocesor. V tabulce 3.1 je možné vidět porovnání vlastností prvních dvou řad modulů (ESP8266 a ESP32).

Jednotlivé řady se mezi sebou liší některými vylepšeními, zejména v oblasti bezpečnosti a dále výkonem. Řady ESP32 a ESP32-C3 těchto mikrokontrolérů obsahují rovněž Bluetooth, jak uvádí společnost Espressif Systems na svých webových stránkách⁴².

ESP moduly se prodávají jednak jako samostatné čipy, ale je možné je zakoupit i jako přídatný komunikační modul k jinému mikrokontroléru⁴³. Také je možné je pořídit jako součást vývojové desky. Mezi nejznámější vývojové desky s ESP8266 moduly patří ESP-01, Lua NodeMcu a WeMos D1 Mini [52].

³⁹GPIO – General-purpose input/output (univerzální vstupní/výstupní pin)

⁴⁰RTOS – Realtime-operating system (real-time operační systém)

⁴¹Převzato z <https://www.cnx-software.com/2016/03/25/esp8266-and-esp32-differences-in-one-single-table/>

⁴²<https://www.espressif.com/en/products/modules>

⁴³<https://dratek.cz/arduino/910-tcp-ip-wifi-modul.html>

Specifications	ESP8266	ESP32
MCU	Xtensa® Single-Core 32-bit L106	Xtensa® Dual-Core 32-bit LX6 600 DMIPS
802.11 b/g/n Wi-Fi	Yes, HT20	Yes, HT40
Bluetooth	None	Bluetooth 4.2 and below
Typical Frequency	80 MHz	160 MHz
SRAM	160 kBytes	512 kBytes
Flash	SPI Flash , up to 16 MBytes	SPI Flash , up to 16 MBytes
GPIO	17	36
Hardware / Software PWM	None / 8 Channels	1 / 16 Channels
SPI / I2C / I2S / UART	2/1/2/2	4/2/2/2
ADC	10-bit	12-bit
CAN	None	1
Ethernet MAC Interface	None	1
Touch Sensor	None	Yes
Temperature Sensor	None	Yes
Working Temperature	- 40°C ~ 125°C	- 40°C ~ 125°C

Tabulka 3.1: Srovnání specifikace modulů ESP8266 a ESP32⁴¹

Ostatní mikrokontrolery a vývojové desky

Mezi nejnámější vývojové desky patří Arduino. Většina informací byla získána z oficiálních stránek Arduina⁴⁴. Jedná se o platformu využívající převážně 8mi bitové mikrokontrolery. Její součástí je C/C++ framework (založený na programovacím jazyce Wiring) a IDE⁴⁵ pro snadný vývoj [51]. Nejnámější varianta Arduino Uno obsahuje mikrokontrolér ATmega328, má 14 digitálních vstupně/výstupních pinů (z toho 6 s možností PWM modulace) a 6 analogových vstupních pinů. K vývojovým deskám Arduino vzniklo mnoho klonů. Pro možnost komunikace přes Wi-Fi je nutné buď využít verzi Arduino s Wi-Fi čipem nebo dokoupit tzv. WiFi shield⁴⁶. Nutno dodat, že cena Arduina se zmíněným Wi-Fi čipem/shieldem mnohokrát převyšuje cenu dříve zmíněných modulů ESP. Arduino s Wi-Fi čipem je možné koupit na českém trhu za cenu okolo 1300 Kč⁴⁷, zatímco vývojová deska Lua NodeMcu s ESP8266 čipem stojí okolo 150ti Kč⁴⁸.

Jako alternativy Arduino vývojových desek je obecně možné považovat různé klony Arduina, kterých je opravdu velké množství. Často mají tyto vývojové desky i odvozený název, jako je Seeeduno⁴⁹, Freeduino⁵⁰ apod. Vybrané platformy ESP a Arduino mají velkou podporu mezi uživateli a jsou relativně levné. Vývojových desek (resp. mikrokontrolerů) však samozřejmě existuje velké množství, ale zmíněné patří mezi nejnámější.

⁴⁴<https://www.arduino.cc/>

⁴⁵IDE – Integrated Development Environment (Vývojové prostředí)

⁴⁶<https://store.arduino.cc/arduino-wifi-shield>

⁴⁷<https://www.alza.cz/arduino-uno-wifi-rev2-d5655517.htm>

⁴⁸<https://www.laskarduino.cz/iot-esp8266-lua-nodemcu-v2-wifi-modul--tcp-ip/>

⁴⁹<https://www.seeedstudio.com/Seeeduno-V4-2-p-2517.html>

⁵⁰https://freeduino.org/freeduino_open_designs.html

3.4 Technologie bezdrátového přenosu Wi-Fi

Wi-Fi je technologie, využívající standardů z rodiny IEEE 802.11. První verze tohoto standardu byla organizací IEEE schválena v roce 1977 [37]. Od té doby vyšlo mnoho dalších verzí standardů.

IEEE 802.11 PHY Standards							
Release date	Standard	Frequency Band	Bandwidth	Transmission Scheme	Max Modulation	MIMO	Max Data Rate
1997	802.11	2.4 GHz	20 MHz	DSSS, FHSS	QPSK	N/A	2 Mb/s
1999	802.11b	2.4 GHz	20 MHz	DSSS	QPSK	N/A	11 Mb/s
1999	802.11a	5 GHz	20 MHz	OFDM	64QAM	N/A	54 Mb/s
2003	802.11g	2.4 GHz	20 MHz	DSSS, OFDM	64QAM	N/A	54 Mb/s
2009	802.11n	2.4 GHz 5 GHz	20 MHz 40 MHz	OFDM	64QAM	4x4	600 Mb/s
2013	802.11ac	5 GHz	20 MHz 40 MHz 80 MHz 160 MHz	OFDM	256QAM	8x8	6.93 Gb/s
2018	802.11ad	60 GHz	2160 MHz	SC, OFDM	256QAM	Beamforming	6.93 Gb/s



Tabulka 3.2: Některé důležité verze standardu IEEE 802.11 a jejich parametry⁵¹

Jednotlivé verze se od sebe mohou odlišovat různými parametry, například frekvenčním pásmem, šířkou pásma jednotlivých kanálů, maximální rychlostí přenosu apod. Organizace Wi-Fi Alliance rozlišuje některé standardy IEEE 802.11 číslem generace Wi-Fi, nejnovější je zatím 6. a oznámena byla již i 7. generace [63]. Přehled vlastností některých verzí Wi-Fi je možné spatřit v tabulce 3.2.

Wi-Fi funguje na principu vysílání a přijímání rádiových vln. Organizace IEEE rozhodla využít pro technologii Wi-Fi frekvence z ISM⁵² pásma [66]. Wi-Fi standardně využívá frekvencí 2,4 GHz a 5 GHz. Nejprve byla zařízení Wi-Fi schopná pracovat pouze v jednom z těchto dvou frekvenčních pásem, ale 4. generace (IEEE 802.11n) přidává možnost práce v obou zmíněných pásmech. Moderní zařízení s Wi-Fi si tak mohou vybrat (a dokonce během své činnosti měnit) frekvenci, na které budou spolu komunikovat. Obě pásma mají svá pro i proti. Mezi výhody pásma 2,4 GHz patří zejména větší pokrytí signálu a rovněž větší kompatibilita (platí spíše pro starší zařízení). Na druhou stranu pásmo 5 GHz nabízí podstatně vyšší přenosové rychlosti a dále větší množství komunikačních kanálů [3]. Wi-Fi generace 6E a připravovaná generace 7 umožňuje komunikaci i na frekvenci 6 GHz [63].

Režim sítě

Wi-Fi nachází uplatnění v bezdrátových lokálních sítích. V nich pak rozlišujeme 3 režimy na základě toho, jak se Wi-Fi zařízení v síti mezi sebou navzájem spojují (jakou plní roli):

- režim infrastruktury,
- ad hoc režim,

⁵¹Převzato z <https://www.grandmetric.com/2018/05/29/wi-fi-standards-evolution/>

⁵²ISM – industrial, scientific and medical (nelicencované frekvenční pásmo pro průmysl, vědu a lékařství)

- smíšený režim.

V režimu infrastruktury je v síti přítomen minimálně jeden centrální prvek (tzv. přístupový bod), který zprostředkovává komunikaci mezi jednotlivými prvky (klienty) sítě, případně poskytuje připojení do jiné sítě přes distribuční systém (*DS*). V tomto režimu sítě je výhoda, že je snadné připojit do stávající infrastruktury nový prvek.

Ad hoc je režim bezdrátové sítě, ve které není přítomen žádný centrální prvek (přístupový bod), se kterým by prvky sítě komunikovali, ani zde není žádné spojení s pevnou sítí přes distribuční systém. Jedná se tedy o decentralizovanou síť. Jednotlivé prvky mezi sebou navzájem komunikují přímo (toto spojení se někdy označuje jako tzv. *peer-to-peer*). V tomto režimu má síť rovněž SSID identifikátor, kterým je možné síť identifikovat [66]

3.5 Ostatní technologie bezdrátového přenosu

V této podkapitole se nachází informace k dalším dvěma technologiím bezdrátového přenosu – Bluetooth a ZigBee.

Technologie Bluetooth

Bluetooth je standard, definovaný v IEEE 802.15.1. Vytvořila jej firma Ericsson v roce 1994 a od té doby vyšlo několik nových verzí [37]. Podobně jako Wi-Fi pracuje v ISM pásmu 2,4 GHz. Na rozdíl od Wi-Fi však není definován pouze na prvních dvou vrstvách ISO/OSI, ale definuje protokoly na všech sedmi vrstvách tohoto modelu. Na nejnižší úrovni, kde definuje způsob přenosu jednotlivých bitů, využívá metodu FHSS, která zajišťuje, že při přenosu bitů vysílač přeskakuje mezi několika frekvencemi [21].

Zařízením, které jej využívají, umožňuje vytvořit tzv. PAN (osobní síť). V těchto sítích má každé zařízení přiřazeno unikátní 48bitovou adresu `BD_ADDR` (BlueTooth Device Address) – jedná se o obdobu MAC adresy u ethernetu. Tu používá pro komunikaci s ostatními zařízeními. Jedno zařízení může být v roli master (řídící), slave (podřízená) nebo obojího [35]. K jedné řídící stanici se připojuje jedno a více podřízených zařízení (používá se pouze adhoc komunikace mezi master a slave stanicí). Zde hovoříme o tzv. piconetu (pikosíť). Maximální počet zařízení v jedné pikosíťi je 8 (jedna řídící stanice a až 7 podřízených). Stanice náležící do jedné pikosíťe může zároveň patřit do jiné pikosíťe. Jedná se tedy o rozšíření sítě mezi zařízeními. Takto vytvořenou síť nazýváme tzv. *scatternet* (rozprostřená síť). V každé rozprostřené síťi má každá pikosíť unikátní identifikátor – je jím `BD_ADDR` její řídící stanice. Díky rozlišení jednotlivých pikosíťi pak může každá tato síť využívat jiné skokové sekvence (frekvenčních kanálů, na kterých se vysílají/přijímají data) [14]. Výhodou Bluetooth technologie je její nízká spotřeba energie (zejména od verze 4) [2].

Technologie ZigBee

Zigbee je bezdrátová technologie, založená na standardu IEEE 802.15.4. Je určena pro vytváření sítě PAN (osobní síť) a pracuje v pásmu ISM 868 MHz, 902-928 MHz a 2,4 GHz [37].

ZigBee standard specifikuje 2 typy zařízení – FFL (Full Function Device) a RFD (Reduced Function Device). FFL zařízení je obvykle schopné mnoha funkcí a je stále aktivní, zatímco RFD se nachází většinu času v režimu spánku, ze kterého se občas probudí, například aby odeslalo hodnoty naměřené na nějakém senzoru.

V síti pak každé ze zařízení plní některou ze 3 funkcí:

- koordinátor,
- koncové zařízení,
- směrovač.

Na základě definovaných zařízení pak existují 3 možné topologie ZigBee sítě [13]:

- hvězda,
- strom,
- mesh síť.

ZigBee podobně jako Bluetooth definuje komunikaci na všech úrovních modelu ISO/OSI, nekopíruje však přesně jednotlivé vrstvy [23]. První 3 vrstvy modelů ISO/OSI a ZigBee si odpovídají, ale vrstvy L4-L7 jsou spojené do vrstev APS (Application Support) a ZDO (ZigBee Device Object).

Kapitola 4

Zhodnocení možností systémů chytrých domácností a plán práce

V této kapitole se věnuji zhodnocení již existujících řešení automatizace domácnosti. Následně je uveden můj návrh řešení na základě nastudovaných řešení ve vhodném rozsahu práce. Nakonec jsou v bodech stanoveny cíle vyplývající z návrhu řešení, které se v práci snažím splnit.

4.1 Zhodnocení současného stavu v oblasti automatizace domácnosti

Na trhu se v současné době nachází velké množství systémů automatizace domácnosti. V následující tabulce se nachází porovnání jednotlivých systémů mezi sebou a pod tabulkou je pak rozbor těchto systémů. Data pro tabulku a rozbor byla získána jednak z kapitoly 2.5, dále ze stránek výrobců daných systémů, z vlastní zkušenosti z používání některých z nich a některá z dat bylo potřeba do jisté míry hodnotit subjektivně (jako třeba cenu nebo komplexnost).

Vlastnost	Loxone	Jablotron	Kangtai	HomeKit	Home Connect
Cena	Vyšší	Vyšší	Nižší	Nižší	Vyšší
Systém je otevřený	Ne	Ne	Ne	Ano	Ne
Funguje s hlasovými asistenty	Ano	Omezeně	Ano	Ano	Ano
Vyžaduje centrální jednotku	Ano	Ano	Ne	Ne	Ne
Funguje přes internet	Ano	Ano	Ano	Jen s centrální jednotkou	Ano
Možnost vyzkoušet aplikaci bez zakoupených zařízení	Ano	Ne	Omezeně	Omezeně	Ano
Vyžaduje odbornou montáž	Ano	Ano	Ne	Ne	Ne
Komplexní	Ano	Ne	Ne	Ano	Ne

Tabulka 4.1: Porovnání jednotlivých systémů mezi sebou.

Ze systémů popsaných v části 2.5 je nejrozvinutějším ten od společnosti Loxone. Zabírá opravdu širokou škálu možností a jen stěží by se hledala aplikace, pro kterou by nebyl vhodný (z hlediska automatizace). Kromě komplexnosti u něj oceňuji rovněž českou ja-

zykovou lokalizaci. V češtině je k dispozici jak aplikace na ovládání (Loxone App), tak rovněž program pro konfiguraci systému (Loxone Config). Čím mě Loxone mile překvapilo je, že jsem si jejich aplikaci Loxone App mohl vyzkoušet v demoverzi i bez zakoupených komponent.

Jako nevýhodu Loxone vidím příliš vysokou cenu. Například při pořízení 3 chytrých zásuvek a miniserveru, který je k ovládání zásuvek potřebný, zaplatí uživatel přibližně 15 000 Kč⁵³. Přitom alternativní řešení od jiných firem, jako Sonoff (při propojení s Apple Homekit) bude stát necelé 3 000 Kč, což je velký rozdíl – a při rozšiřování domácnosti o další prvky tento rozdíl znatelně roste. Na druhou stranu pokud uživatel staví nový dům, může řešení od Loxone stát srovnatelnou cenu jako konkurenční „neinteligentní“ instalace⁵⁴. Jako další nevýhodu vidím to, že celkově je instalace systému orientovaná spíše pro profesionální montáž pro pracovníky s příslušnou klasifikací (většina produktů je určena k zabudování do rozvaděče, příp. ke komunikaci s moduly v něm). Celkově je však řešení od Loxone na hodně vysoké úrovni.

Řešení od firmy Jablotron je zajímavé jejich dvoutlačítkovými (rozšiřitelnými) segmenty. Zdá se mi však nepraktické spojovat přístupovou klávesnici do domu s prvky automatizace domácnosti. Působí to poněkud omezeně. Navíc rozhraní pro ovládání domácnosti a alarmu v aplikaci MyJablotron se snaží napodobovat onu klávesnici, což příliš k přehlednosti nepřispívá. Na druhou stranu je systém od firmy Jablotron ideální volbou pro uživatele, jehož hlavním požadavkem je zabezpečení objektu a pouze doplňková automatizace domácnosti (jako rozsvícení světel při odjištění domu). Cenově pak toto řešení vychází podobně⁵⁵ jako Loxone, ale vždy samozřejmě záleží na volených komponentách, které uživatel vyžaduje. Pokud však není vyžadováno zabezpečení domácnosti (ale pouze automatizace světel), pak je toto řešení celkem drahé. Zajímavou možností však je spojení obou zmíněných systémů (Jablotron a Loxone), které je možné díky převodníku *JA-121T*⁵⁶.

Systém HomeKit je zajímavý v tom, že zde není potřeba žádný „speciální“ centrální prvek – pokud již uživatel vlastní například iPad či jiný produkt, který zastoupí funkci centrálního prvku. Samozřejmě pokud takový prvek v domácnosti schází, tak se jeho pořízení stává další investicí. Co je ale na systému HomeKit pozitivní, je jeho nízká cena ve srovnání se systémy od společnosti Loxone či Jablotron. Systém HomeKit se stále rozrůstá a má velkou podporu v rozmanitosti produktů. A na rozdíl od předchozích zmíněných systémů je více orientovaný na běžné uživatele v tom smyslu, že nevyžaduje montáž od specialistu. Jednou z nevýhod je zde to, že je systém orientovaný zejména na bezdrátovou komunikaci, která samozřejmě někdy může být méně spolehlivá. Tím spíše, že mnoho produktů komunikuje pouze pomocí Bluetooth, takže si uživatelé musejí hlídat dosah zařízení. Obecně se však jedná o cenově dostupnou variantu chytré domácnosti.

Chytrá zařízení od společnosti Kangtai pak nabízí podobné možnosti jako HomeKit, a i cena je srovnatelná. Je tedy možné na ně pohlížet jako na alternativy. Homekit je ale rozšířenější a je možné k němu koupit mnohem více chytrých zařízení. Samotné řešení od společnosti Kangtai jsem sám vyzkoušel v podobě Wi-Fi zásuvky. Všeobecně se jedná o spolehlivé zařízení, výtky mám pouze k jejich aplikaci. Ve starší aplikaci⁵⁷, určené k ovládání zásuvky, která byla uvedena v příloženém manuálu jako výchozí⁵⁸, se mi podle manuálu

⁵³Ceny byly stanoveny na základě e-shopu na stránkách výrobce, viz <https://shop.loxone.com/cscz/miniservers.html>

⁵⁴<https://www.loxone.com/cscz/blog/kolik-stoji-chytry-dum-cena/>

⁵⁵<https://www.lupa.cz/clanky/zabezpecovaci-zarizeni-jablotron-100-co-nabizi-a-co-umi/>

⁵⁶<https://www.loxone.com/cscz/blog/bezpecny-chytry-dum-od-jednicek-na-trhu/>

⁵⁷<https://play.google.com/store/apps/details?id=com.bugull.kangtai>

⁵⁸<https://www.tipa.eu/en/smart-wifi-socket-kangtai-51062/d-175329/>

nepodařilo nové zařízení do systému přidat. Nová aplikace⁵⁹ je na tom lépe (přidání zařízení proběhlo bez problému), ale má některé menší nedostatky. Zejména není možné spustit více časovačů pro jedno zařízení současně. Nový časovač je možné nastavit vždy až po vypršení předchozího. Kromě toho aplikace nezobrazuje přívětivým způsobem zbývající čas, kdy k nastavené akci dojde, ani k jaké akci dojde. Jediná možnost jak tyto informace zobrazit, je znovu otevřít nastavení časovače (jakoby uživatel chtěl nastavit nový). Zde uživatel uvidí zbývající čas ve vstupu, kde případný nový čas nastavuje a nastavovanou hodnotu, viz obrázek 2.7. Tento způsob ale považuji za neintuitivní (navíc pro aktualizaci zbývajícího času je potřeba se vrátit zpět a opět vstoupit do časovače). Problémovým je i maximální doba, kterou jde na časovači nastavit, která činí méně než hodinu. Také některým uživatelům může vadit absence snímačů (alespoň jejich zobrazování v oficiální aplikaci), ale jelikož je možné systém propojit s některými virtuálními asistenty, tak podpora snímačů ze strany výrobce není nezbytná. Další věc, která se jeví jako nepříjemná, je, že systém sám nedokáže rozpoznat, zda se uživatel nachází v lokální síti (a v případě chybějícího připojení k Internetu přepnout systém na lokální funkci). Toto přepnutí je nutné provést manuálně a vyžaduje odhlášení z uživatelského účtu.

Systém (resp. technologie) Home Connect vnáší do automatizace domácnosti zajímavý koncept. Zatímco některé systémy umožňují automatizovat domácnosti například chytrými zásuvkami či spínači, Home Connect ve spolupráci s jinými společnostmi vyvíjí přímo spotřebiče s prvky chytré domácnosti, čímž tyto spotřebiče obsahují mnohem více „inteligence“, na rozdíl od prostého zapínání/vypínání. Nicméně nevýhodou je zde příliš malý sortiment produktů a tudíž jednoúčelová aplikace navíc, kterou stejně musejí uživatelé doplnit o další aplikace, chtějí-li například ovládat také zásuvky, světla či rolety. Kromě toho je rovněž cena produktů dost vysoká.

4.2 Návrh technického řešení systému automatizace domácnosti

Na základě výzkumu dostupných řešení a jejich zhodnocení jsem se rozhodl pro vyvinutí systému s některými prvky automatizace. Svými funkcemi by se měl podobat řešení jako je to od společnosti Kangtai, ale s vyřešením některých nedostatků, které se u jejich Wi-Fi zásuvek vyskytují. Především musí být možné nastavit více souběžných časovačů. Zobrazování zbývajícího času těchto časovačů je potřeba vyřešit přehlednější/intuitivnější formou. Také nesmí chybět možnost nastavit přímo v aplikaci automatizace na základě snímačů. Rovněž je potřeba se zabývat jednodušším rozlišováním, zda uživatel ovládá systém pouze v rámci lokální sítě nebo přes Internet. V systému se tedy nebude muset nejprve odhlásit ze svého účtu k ovládání v rámci lokální sítě, jako je tomu u systému od Kangtai.

Mým řešením chci doplnit existující řešení o takové, které je svými vlastnostmi určeno spíše pro nadšence v oblasti IT a elektroniky. Systém má poskytovat jednoduchý způsob ovládání mnoha zařízení pomocí jednoho vestavěného zařízení s připojeným displejem a obsahovat českou jazykovou lokalizaci. Svou prací jsem si chtěl rovněž vyzkoušet celý návrh a realizaci systému automatizace domácnosti, který bych následně mohl sám využívat. Také jsem chtěl klást důraz zejména na nízkou cenu, která by měla být v řádu maximálně několika tisíc korun. Rozhodl jsem se, že veškerý kód, který v rámci práce vznikne, bude uvolněn jako open source.

⁵⁹<https://play.google.com/store/apps/details?id=com.ktapp.wifi>

Základem práce má být především dálkové ovládání jednoho zařízení druhým, jelikož je to zadání bakalářské práce. V systému tedy figuruje ovladač a dále ovládané prvky. Ovladačem jsou různá zařízení s displejem, přes která je možné systém ovládat (např. mobilní telefon). V práci má být použito Raspberry Pi jako centrální jednotka a je potřeba zvolit zařízení, přijímající povely od této jednotky. Tato zařízení budou těmi dříve zmíněnými ovládanými prvky. Ovládaný prvek tedy nemá být žádné konkrétní zařízení jako zásuvka či lampička, ale obecný modul se vstupně-výstupními porty, přes které bude možné ovládat jiná zařízení. Pro plnohodnotnou funkci systému je tento modul nutné opatřit dalšími přídatnými součástkami. Jde zejména o relé pro možnost ovládání zapnutí a vypnutí zařízení, připojeného k tomuto modulu (tímto způsobem je možné například ovládat LED pásek či vytvořit bezdrátovou zásuvku). K modulu mohou být připojeny také MOSFET tranzistory pro možnost použití PWM⁶⁰ na výstupech modulů – takový výstup pak bude sloužit třeba pro stmívání světel, zejména LED pásku či bodových LED světel na 12 V. Takže v tomto směru má být výsledný systém podobný řešení jako nabízí Jablotron, které umožňuje ovládání relé modulů a ne řešení od Kangtai, které obecně nabízí zejména ovládání zásuvek (tedy již „hotových“ produktů). Jednotlivé moduly je možné zabudovat i jako součást jiných zařízení či systémů a ty tak doplnit o funkce dálkového ovládání a automatizace. Aby bylo ovládání systému co nejjednodušší, musí zde být možnost ovládání na dotykovém displeji, připojeném k centrální jednotce (tedy k Raspberry Pi). Navržený systém musí podporovat několik typů snímačů. Jejich úlohou je jednak informovat uživatele (o teplotě, vlhkosti vzduchu apod.), ale je možné je využít i při automatizaci (nastavit na určitou úroveň jas LED pásku, pokud intenzita venkovního světla klesne pod určitou úroveň či zapnout světlo při detekci osoby snímačem pohybu). Tyto automatizace musí být možné libovolně zapínat a vypínat. Mezi systémem podporované snímače patří:

- BMP280 (snímač tlaku a teploty),
- SHT21 (snímač teploty a vlhkosti),
- BH1750 (snímač intenzity světla).

Pro účely práce jsou použity jen tyto, nicméně řešení je pojaté jako open source a další snímače bude možné přidat v budoucnu. Kromě zmíněných tří snímačů je v systému nutné podporovat digitální vstup. Díky tomu je možné používat spoustu dalších snímačů jako je HC – SR501 (PIR pohybové čidlo), nejrůznější analogové snímače s komparátorem, plovákové spínače, detektory otevřených dveří (založené na kontaktu), atd. Nakonec musí být také možné výstupy ovládat pomocí nastavených časovačů, jak již bylo naznačeno. Jednotlivé časovače je také možné zapínat a vypínat podle potřeby.

Při návrhu jsem se rozhodl, že budou použity již existující moduly a mikropočítač Raspberry Pi, které však naprogramuji a vhodným způsobem doplním o některé elektronické součástky (jako zmíněné relé či tranzistor). Práce se však nezabývá detailním konstrukčním návrhem prvků systému, ani návrhem DPS pro jednotlivé části hotového systému. Pro případné propojení součástek a zařízení systému jsou použita nepájivá kontaktní pole.

4.3 Požadované vlastnosti navrhovaného systému

Na základě předchozích úvah jsem se na začátku realizace rozhodl pro vytvoření systému, který splňuje následující vlastnosti:

⁶⁰PWM – Pulse Width Modulation (pulzně šířková modulace)

- jako centrální jednotka bude zvolen mikropočítač Raspberry Pi verze 3B+,
- k tomuto mikropočítači bude pro pohodlné ovládání připojen dotykový displej o velikosti nejméně 7",
- pro komunikaci v systému bude využita technologie Wi-Fi,
- k implementaci bude zvolen vhodný programovací jazyk,
- systém bude fungovat jak v lokální síti, tak i přes Internet,
- bude podporována funkce přímého ovládání výstupů ovládaných modulů,
- v systému se budou zobrazovat data ze snímačů,
- snímače bude možné využít k automatizaci výstupů modulů,
- systém bude obsahovat českou jazykovou lokalizaci,
- v systému bude možné nastavovat časovače ovládací výstupy modulů,
- v systému budou figurovat uživatelské účty, přičemž z jednoho účtu bude možné ovládat jen jednu domácnost,
- systém bude navrhován s ohledem na nízkou cenu.

Také jsem se rozhodl, že bude nakonec celý systém uvolněn jako open source, čímž bude cena systému jako takového pro potenciální uživatele minimální (daná pouze cenou použitých součástek a zařízení). Díky tomu si také uživatelé budou moci případně systém upravit na míru.

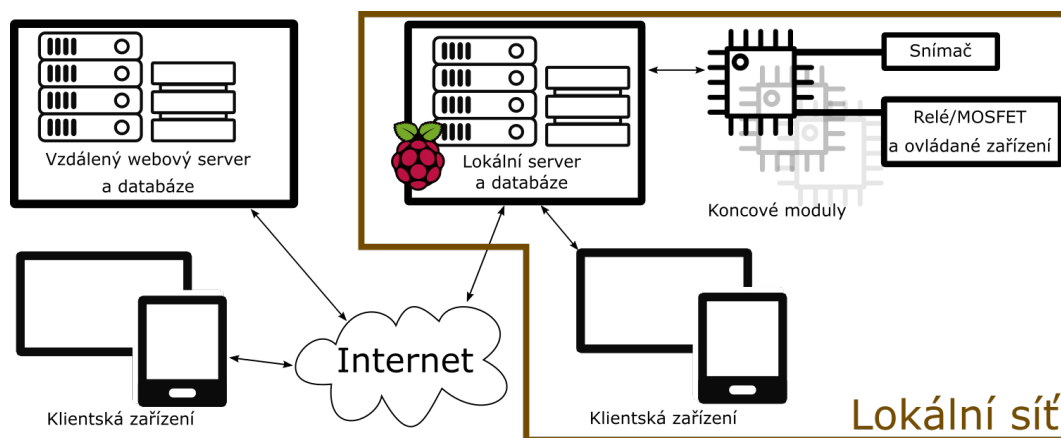
Kapitola 5

Realizace systému automatizace domácnosti a testování

V této kapitole se věnuji vlastní realizaci řešení a následnému testování a vyhodnocení. V první podkapitole je uveden celkový návrh systému. Následně je uveden návrh aplikace, přes kterou bude uživatel systém ovládat. Dále je vysvětleno použití databáze, zvolená implementace autentizace a webhosting. V této části se nachází společné rysy všech tří aplikací, protože je není možné úplně oddělit a popisovat samostatně. V další podkapitole jsou pak detaily k implementaci aplikace pro Raspberry Pi a aplikace běžící na použitých modulech. Nakonec uvádím, jak jsem systém testoval, a k jakým výsledkům jsem došel.

5.1 Celkový návrh systému automatizace domácnosti

Systémy chytré domácnosti bývají různě komplexní a obvykle se skládají z různých zařízení a také aplikací (i když uživatel většinou přistupuje pouze k jedné hlavní s grafickým uživatelským rozhraním).



Obrázek 5.1: Koncepce výsledného systému pro automatizaci domácnosti.

Stejně je tomu v systému, který jsem navrhl já. Celkem jsou zde 3 (v různé míře) spolupracující aplikace a několik typů zařízení. Každé z těchto zařízení a aplikací zde má svou nezastupitelnou úlohu.

Na obrázku 5.1 je možné vidět celkovou koncepci systému. Jak je z obrázku patrné, celý systém se skládá z několika částí:

- různá klientská zařízení s displejem jako je notebook, mobilní telefon apod., přes která může uživatel systém ovládat,
- lokální server s databází, který funguje jako webový server a centrální řídicí jednotka systému,
- vzdálený webový server s databází, který slouží k přístupu k systému přes Internet,
- koncové moduly, které komunikují s lokálním serverem, nastavují své výstupy a posílají lokálnímu serveru informace o naměřených hodnotách na snímačích,
- snímače a výstupní zařízení, připojená k vstupně/výstupním portům koncových modulů.

Pro uživatele jsou zde klíčovými zejména klientská zařízení, přes která může systém konfigurovat, ovládat a přes která je informován o naměřených hodnotách na snímačích. Prostřednictvím grafické aplikace na těchto zařízeních komunikuje s databází. Jedná se o webovou aplikaci, kterou těmto zařízením zprostředkovává webový server.

Lokální server má v systému celkem dva hlavní úkoly. Jeho první funkcí je, že figuruje jako webový server v lokální síti. Pokud se tedy uživatel vyskytuje ve stejné síti jako tento server, může k webové aplikaci přistupovat přes jeho IP adresu. Kromě funkce webového serveru slouží také jako centrální řídicí jednotka celého systému. Je zodpovědný za zpracovávání požadavků od klientských zařízení a na jejich základě následně komunikuje pomocí Wi-Fi s koncovými moduly, kterým předává instrukce k nastavování výstupů. Také získává informace o hodnotách naměřených veličin na snímačích připojených k těmto koncovým zařízením. Je vhodné zmínit, že tyto dvě funkce lokálního serveru jsou implementované v rámci jedné aplikace. Pokud je dále v textu řeč o serveru bez bližšího určení, je tím myšleno na základě kontextu buď právě toto zařízení nebo aplikace, která na něm běží.

Aby bylo možné systém ovládat přes Internet, bylo potřeba zvolit vhodný mechanismus, kterým by bylo možné přistupovat k webové aplikaci i mimo lokální síť. Já jsem se rozhodl pro využití webhostingu a v systému tak hraje roli i vzdálený webový server a databáze. Bylo možné využít i jiný mechanismus. Proč jsem zvolil tento a některé detaily, které s ním souvisí, je možné najít v podkapitole 5.3. Tento vzdálený server pochopitelně funguje jen jako webový server, roli centrální řídicí jednotky plní pouze lokální server. Oba servery mají svoji vlastní databázi. Ta by v ideálním případě měla být synchronizovaná. Nemusí tomu tak být v případě, že lokální server ztratí připojení k Internetu. Pokud chce uživatel přistupovat k webové aplikaci v rámci lokální sítě, připojuje se pomocí IP adresy lokálního serveru. Pokud chce aplikaci ovládat přes Internet, tak musí přejít na adresu <https://automatizace.web.app/>. Tam, kde je v následujícím textu řeč o „verzi“ webové aplikace, tak je tím myšleno právě rozlišení této lokální a internetové webové aplikace.

Dalším zařízením jsou koncové moduly (či jen moduly). Ty mohou mít ke svým vstupům připojeny snímače (na těch v pravidelných intervalech snímají hodnoty a posílají je na server). Na své výstupy pak mají připojeny pomocné součástky (jako relé či tranzistory). K nim jsou připojena ovládaná zařízení, ve kterých spínají kontakt. Může se jednat o prosté zapnutí/vypnutí zařízení, jednoduché stmívání (PWM modulací na výstupu), ale také o pokročilé ovládání při zabudování tohoto modulu do nějakých dalších zařízení.

Bližší specifikace zařízení a aplikací v systému

Jak tedy vyplývá z předchozího popisu koncepce systému, v domácnosti uživatele figuruje několik zařízení. Jako server bylo zvoleno Raspberry Pi verze 3B+, jelikož to bylo dříve definováno v návrhu řešení v části 4.3. Tento mikropočítač má pro danou aplikaci výhodné vlastnosti. Jako server totiž musí být použito zařízení, které je schopné v reálném čase obsluhovat klienty, zpracovávat jejich požadavky a komunikovat s databází – což by vzhledem k parametrům této verze Raspberry Pi neměl být problém. Kromě toho server musí obsahovat port pro připojení displeje a Raspberry Pi 3B+ obsahuje HDMI port.



Obrázek 5.2: Raspberry Pi s připojeným displejem.

Nabízí se zde otázka, proč jako centrální jednotku nepoužít klasický stolní počítač. Důvodem může být především mnohem vyšší spotřeba elektrické energie, hluknost a možné výkyvy výkonu na základě používání počítače uživatelem. Tím spíše, že je tato jednotka určena k nepřetržitému provozu.

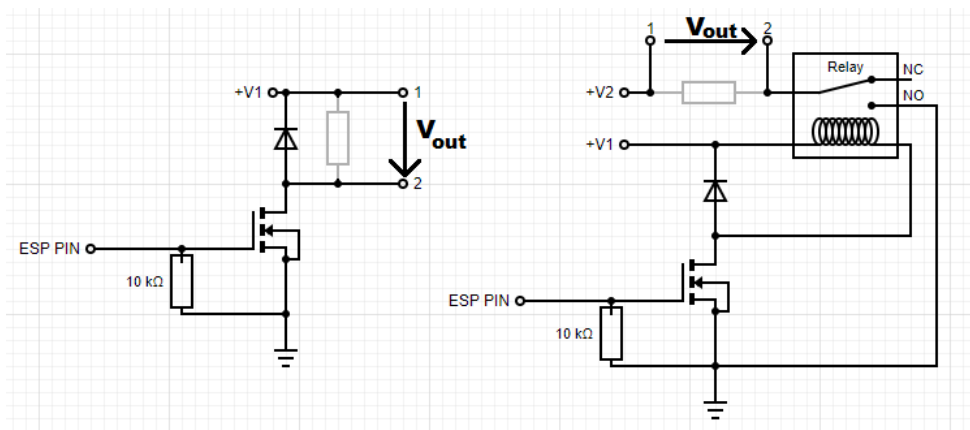
Vybraný dotykový displej⁶¹ má rozlišení 1024x600 px a uhlopříčku 7". To by mělo být dostatečně pro přehledné zobrazení webové aplikace a pohodlné ovládání. Raspberry Pi se díky němu chová nejen jako server, ale i klientské zařízení (pokud na tomto displeji zobrazuje internetový prohlížeč s otevřenou webovou aplikací). Raspberry Pi s připojeným displejem a výslednou webovou aplikací určenou k ovládání systému, je možné vidět na obrázku 5.2.

V části 3.1 byl vestavěný systém definován jako počítač, který se pro uživatele jako (univerzální) počítač nejví. V této práci však záměrně neřeším toto uschování schopností Raspberry Pi před uživatelem. Nejedná se totiž o komerční systém, aby byl k něčemu takovému důvod. Naopak uživatel může vytěžit z toho, že Raspberry Pi (příp. jiný počítač, na který systém nainstaluje) může nadále využívat i k jiným účelům. Kdyby byl systém

⁶¹<https://rpishop.cz/displeje-monitory/1476-waveshare-7-dotykovy-lcd-h-displej-1024x600-ips-kapacitni-hdmi-bez-krabicky.html>

komerčně prodávány, tak by bylo samozřejmě vhodné uživateli vzít možnost nakládat s počítačem dle vlastní libovůle.

Jako modul bylo potřeba vybrat zařízení s nízkou cenou, jelikož těchto zařízení má být v systému obecně více. Kromě toho má modul přijímat pouze jednoduché instrukce od serveru, může se tedy klidně jednat o nějaký mikrokontrolér s omezeným výpočetním výkonem. Je však potřeba, aby byl schopný komunikovat po síti (tento požadavek mj. vyplývá ze zadání mé bakalářské práce). Buďto bylo možné vybrat nějaký mikrokontrolér a k němu připojit modul, který jej o funkci komunikace po síti doplní (např. vývojovou desku Arduino s WiFi shieldem), nebo zvolit takový, který již v sobě obsahuje prostředky nutné k bezdrátové komunikaci. Já jsem se rozhodl pro druhou možnost. Na trhu existuje spousta takových mikrokontrolerů, mezi nejznámější a zároveň nejlevnější patří ESP8266. Jelikož má pro danou aplikaci dostatečný výpočetní výkon, tak jsem použil právě tento modul, resp. vývojové desky *Lua NodeMcu V2*, které jej obsahují⁶². Napájení je zajištěno přes USB port, který moduly obsahují.



Obrázek 5.3: Zapojení součástek k ovládnání výstupů. Vstup $+V1$ slouží pro připojení napájení. Vstup $V2$ umožňuje galvanické oddělení napájení pro zátěž. Ovládané zařízení je možné připojit mezi svorky 1 a 2 , kde je naznačeno jako zátěž pomocí rezistoru.

Jako snímače byly použity ty, které byly vybrány při návrhu řešení (v části 4.2). Ovládanými zařízeními jsou ta, která budou připojena na výstup relé či tranzistoru, připojeného na některý z výstupů modulu. Konkrétněji tedy bude záležet čistě na uživateli, co chce systémem ovládat. Může se jednat o nějaká jednoduchá zařízení, které systém sepnutím relé připojí ke zdroji napájení. Také je možné využít jednoduchého stmívání pomocí PWM modulace (při využití MOSFET tranzistoru). Kromě toho je samozřejmě možné modul použít jako součást jiného zařízení, které vyžaduje vstup od uživatele. Příkladem by mohl být ovladač žaluzií⁶³, který na základě hodnoty na svém vstupu patřičným způsobem nastaví žaluzie. Tento vstup by poskytoval právě koncový modul (na svém výstupu). Jedná se však o příklad a v rámci bakalářské práce ovládnání žaluzií realizováno nebylo. Zapojení součástek, které bylo při realizaci práce připojeno na výstup modulů ESP8266 je možné vidět na obrázku 5.3. Je potřeba si uvědomit, že když je v dalším textu řeč o ovládaných zařízeních či výstupech, je myšleno to samé, protože ovládnáním výstupů modulů uživatel ovládá zároveň relé/tranzistor na tomto výstupu a v důsledku toho i ovládané zařízení.

⁶²<https://www.laskarduino.cz/iot-esp8266-lua-nodemcu-v2-wifi-modul--tcp-ip/>

⁶³<https://aququ.cz/product-cze-20579-Ovladac-zaluzii-do-krabicky-dratovy-SRP-01-extend-Zamel-univ.html>

Jako klientská zařízení považuji taková zařízení s displejem, přes která může uživatel ovládat systém. Jde tedy třeba o osobní počítač či chytrý telefon. Požadavky na tato zařízení jsou vzhledem k implementaci schopnost komunikace pomocí Wi-Fi a displej s dostatečnou uhlopříčkou a rozlišením k zobrazení aplikace. Také je potřeba aby byl na daném zařízení nainstalován některý z podporovaných internetových prohlížečů, viz část 5.5.

Samozřejmě zařízení samotná nestačí, a aby systém nějakým způsobem fungoval, bylo potřeba pro některé jeho části vyvinout aplikace. Systém jsem navrhl tak, že obsahuje 3 aplikace:

- Aplikace s grafickým rozhraním – ta je určena pro internetový prohlížeč na klientských zařízeních. Právě přes tuto aplikaci je možné celý systém ovládat a konfigurovat z pohledu uživatele.
- Server běžící na Raspberry Pi, který od aplikace s grafickým rozhraním získává instrukce a vykonává je. Na základě těchto instrukcí server dále komunikuje s jednotlivými moduly. Tento server plní také funkci webového serveru.
- Aplikace, která běží na modulech. Jejím úkolem je ovládat své výstupy na základě pokynů od serveru. Rovněž aplikace v pravidelných intervalech serveru posílá naměřené hodnoty na připojených snímačích.

Kromě toho bude součástí systému (i když nepřímou) vzdálený webový server a vzdálená databáze. Nebudou sice nějakým způsobem umístěny fyzicky u uživatele (že by si je musel nějak pořizovat), nicméně jedná se rovněž o nepostradatelnou součást, pokud chce uživatel využívat možnost ovládání systému přes Internet. Pro webhosting a databázi jsem se rozhodl použít služby z platformy Firebase od společnosti Google. Více se této volbě věnuji v části 5.3.

Jako alternativní řešení prvních dvou aplikací bylo možné vytvořit jen jednu desktopovou aplikaci (s grafickým rozhraním), běžící na Raspberry Pi. Přes tu by se systém ovládal pouze z připojeného displeje. Uživatelé by tak ale přišli o možnost ovládat systém i vzdáleně, resp. i z dalších zařízení, což by bylo neefektivní a nepohodlné. Zvolil jsem tedy řešení dvou aplikací.

5.2 Návrh a implementace webové aplikace

Jak již bylo dříve zmíněno, aplikace určená pro uživatele k ovládání systému byla realizována jako webová aplikace, běží tedy v internetovém prohlížeči. K tomu mě vedlo několik důležitých důvodů:

- webové aplikace jsou obecně vzato multiplatformní, čehož využiji v rámci možnosti ovládat domácnost z různých zařízení a navíc na dálku,
- k přístupu k aplikaci tak bude stačit mít zařízení schopné komunikovat po síti a vybavené obrazovkou pochopitelně,
- webové technologie patří mezi rychle se rozvíjející, což přispívá k budoucímu rozvoji systému.

Jako programovací jazyk implementace jsem vybral Typescript, protože se jedná o populární a hojně používaný jazyk velmi podobný Javascriptu (do kterého se ve skutečnosti překládá). A Javascript je jedním z nejpoužívanějších programovacích jazyků vůbec⁶⁴. Po-

⁶⁴<https://pypl.github.io/PYPL.html>

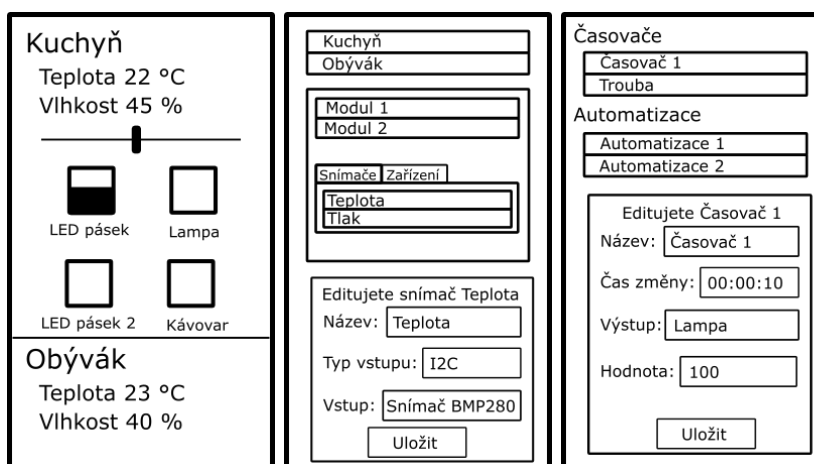
kud se tedy v budoucnu najde programátor, který by chtěl systém rozšířit, je velká pravděpodobnost, že se nebude muset učit nic nového. Použitý jazyk při vývoji (Typescript) má tu výhodu, že umožňuje psát staticky typovaný kód, což napomáhá k eliminaci chyb během vývoje a rovněž některá vývojová prostředí pro Typescript poskytují funkci našeptávání.

Jednostránková aplikace a směrování

Při návrhu jsem se musel rozhodnout, zda půjde o jednostránkovou či vícestránkovou aplikaci. Nakonec jsem zvolil první možnost. K tomu mě vedl zejména použitý programovací jazyk (vícestránkovou aplikaci bych vyvíjel spíše s použitím serverových technologií, zatímco Typescript je orientovaný na klientskou část) a také fakt, že jednostránková aplikace klade na webový server menší zátěž. Server totiž pouze zprostředkuje statické soubory a další změny na stránce jsou již řízeny pomocí Javascriptu. Server pak už jen zpracovává občasné a jednoduché požadavky.

Grafické rozložení stránek a prvků

Aplikaci jsem rozdělil na několik samostatných částí (oken), mezi kterými může uživatel přecházet.

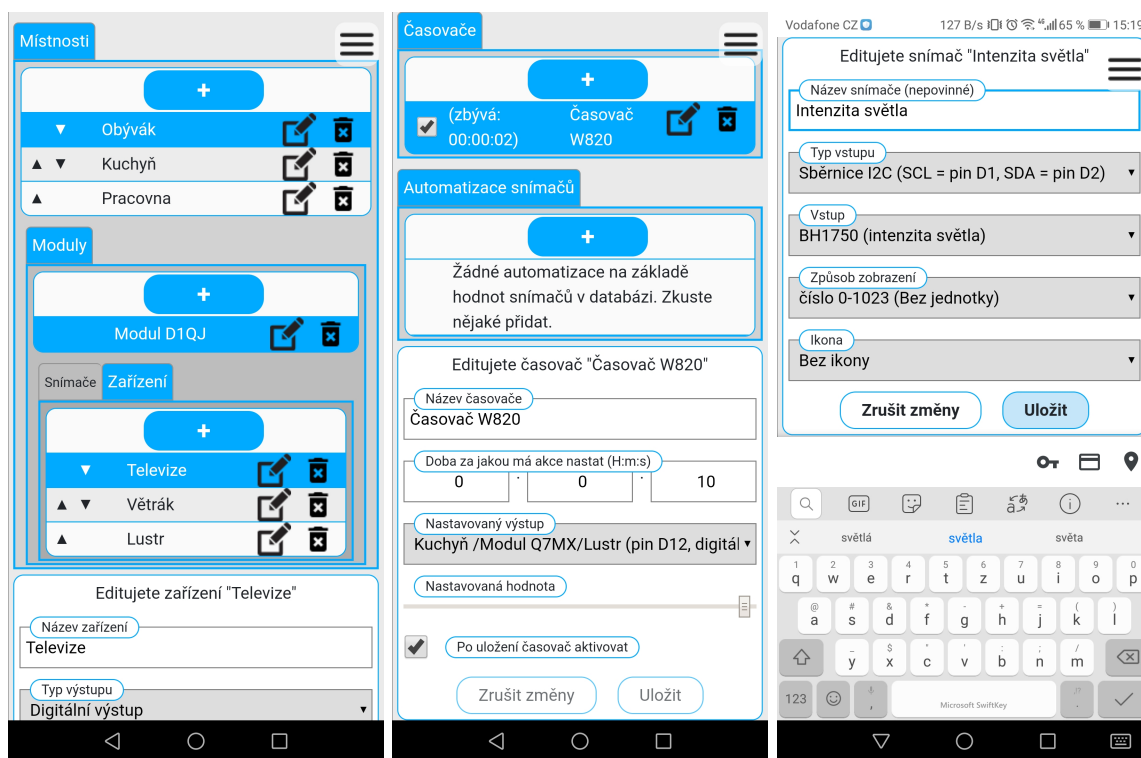


Obrázek 5.4: Grafický návrh rozložení jednotlivých stránek. Vlevo je možné vidět domovskou stránku se snímači a zařízeními. Uprostřed se nachází stránka s konfigurací systému. Napravo pak je nastavení časovačů a automatizací na základě hodnot snímačů.

Jelikož se jedná o jednostránkovou aplikaci, nejde v případě jednotlivých částí o klasické stránky (v tom smyslu, že by každá měla vlastní HTML dokument, který ji generuje). V následujícím textu se však pro jednoduchost budu odkazovat ke každému takovému samostatnému oknu jako k jedné stránce.

Pokud uživatel přistupuje k aplikaci přes Internet, tak se při spuštění aplikace v prohlížeči zobrazí stránka s přihlášením jako úvodní. Z této stránky je možné přejít na stránku s registrací. V případě přístupu k lokální webové aplikaci se jako úvodní zobrazuje domovská stránka aplikace. Samozřejmě i zde je možné přejít k přihlášení, resp. registraci, které zde plní funkci spárování serveru s uživatelským účtem na Internetu. Jelikož však v lokální síti není přihlášení vyžadováno, tak se tyto stránky nezobrazují jako výchozí. Na přihlášení je možné přejít z menu aplikace.

Další částí aplikace je domovská stránka. Ta se zobrazuje po přihlášení, resp. jako úvodní při lokálním přístupu. Tato stránka slouží pro ovládání jednotlivých zařízení. Je zde také přehled o hodnotách naměřených na snímačích. Tuto stránku bude mít uživatel zobrazenou většinu času na displeji Raspberry Pi, pokud jej bude chtít používat jako takový rychlý přehled o stavech ovládaných zařízení a snímačích. Jak je možné spatřit na obrázku 5.4, rozvrhl jsem tuto stránku na jednotlivé místnosti. Každá z nich zobrazuje název místnosti, seznam hodnot na senzorech a ovládaná zařízení, ze kterých je možné vyčíst jejich stav i ovládat je. Pokud chce uživatel ovládat nějaký výstup (zařízení), klikne na něj. V případě, že je výstup digitálního typu (tedy s hodnotami vypnuto/zapnuto), okamžitě se změní stav zařízení. V případě analogových výstupů (tedy poskytujících PWM) se zobrazí nad zařízeními posuvník (nastavený na aktuální hodnotu), kterým je možné měnit hodnotu na výstupu modulu.



Obrázek 5.5: Stránky pro konfiguraci systému (vlevo) a nastavení automatizací (uprostřed). Vpravo je komponenta detail, přes kterou se systém nastavuje. Aktuálně je detail nastaven na snímač.

Také se v aplikaci nachází stránka s nastavením systému (z menu přístupná jako položka *Nastavení*). Zde je možné přidávat nová zařízení a konfigurovat ta stávající. Stránka je navržena jako hierarchie do sebe vnořených seznamů, ve kterých uživatel může zvolit místnost, modul a snímač/zařízení. Pod seznamy se nachází detail, ve kterém může uživatel u aktuálně zvoleného prvku (místnosti/modulu/snímače/zařízení) konfigurovat některé vlastnosti, jako je název, typ vstupu/výstupu apod. Jednotlivá pole této komponenty detailu jsou vždy sestaveny na základě aktuálně zvoleného prvku (jinak bude vypadat pro místnost a jinak pro snímač). Ukázkou detailu výsledné aplikace pro konfiguraci snímače je možné vidět na obrázku 5.5. Pokud je v následujícím textu řeč o konfiguraci nebo nastavení systému v souvislosti s webovým klientem, je tím myšlena tato stránka.

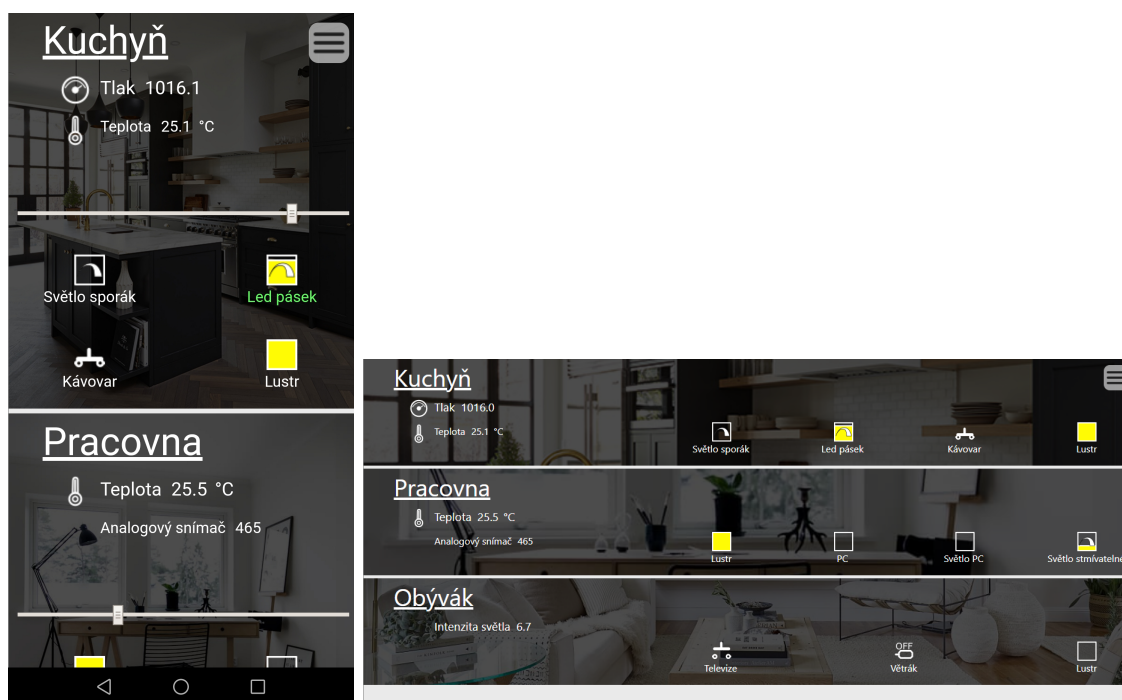
Poslední stránka v aplikaci slouží k nastavení časovačů a automatizací na základě hodnot na snímačích. Opět jsou zde podobné seznamy jako v nastavení, ale ty již do sebe nejsou vnořené. Pod seznamy je opět detail, ve kterém je možné upravovat naposledy zvolený časovač či automatizaci. I zde samozřejmě platí, že detail se dynamicky mění na základě toho, co uživatel upravuje.

V aplikaci se také samozřejmě vyskytuje menu, které se zobrazuje na všech stránkách kromě registrace a přihlášení. Toto menu je vysouvací, aby zbytečně „nepřekáželo“. Pomocí menu je možné přecházet mezi jednotlivými stránkami, odhlásit se (v internetové verzi aplikace), spárovat server s účtem (v lokální verzi aplikace) a také přepnout aplikaci na celoobrazovkový režim.

Je vhodné zmínit, že až na minimální rozdíly, které souvisí s autentizací uživatele, se chovají obě verze aplikace (lokální a internetová) stejně. Uživatel by z nich tedy měl mít stejný pocit.

Responzivní design

Aplikaci jsem implementoval s ohledem na responzivitu, takže v tomto směru je skutečně multiplatformní. Stojí však za zmínku, že jsou zde jisté nedostatky při dynamické změně velikosti obrazovky. Například při změně orientace obrazovky je tedy potřeba aktualizovat aplikaci, ale pak se již zobrazuje korektně. Na obrázku 5.6 je možné vidět porovnání rozložení prvků domovské stránky na mobilním telefonu a na monitoru počítače.



Obrázek 5.6: Domovská stránka výsledné aplikace na mobilním telefonu a monitoru počítače. Na obrázku je možné vidět rozdílné rozložení prvků na základě rozměrů displeje.

Vlastní komponenty

Aplikace při implementaci využívá možnosti definování vlastních HTML komponent (tzv. Custom Components⁶⁵). Za tímto účelem jsem vytvořil třídu *AbstractComponent*. Ta není určená k inicializaci objektů, ale k vytváření nových tříd komponent zděděním.

Každá třída, která z ní dědí, musí přepsat vlastnost *tagName*, která definuje název HTML značky, pod jakým se komponenta bude zobrazovat v HTML dokumentu. Zmíněná třída obsahuje různé pomocné funkce, proto by každá nová komponenta v aplikaci měla skutečně dědit z této třídy. Navíc v konstruktoru přijímá objekt, na základě kterého se může daná komponenta vytvořit. Například je možné již při vytváření komponenty specifikovat, které další komponenty se mají do komponenty vložit, jaké třídy (kvůli css stylům) se mají komponentě přiřadit apod.

Celá aplikace se skládá z více než 60ti různých tříd. Většina tříd slouží právě jako komponenty, které využívají jednotlivé stránky. Tam kde to mělo smysl, jsem se snažil komponenty implementovat co nejobecněji, takže je u nich jistá znovupoužitelnost. Jde to vidět v aplikaci například na opakovaném použití komponenty seznamu (třída *List*) v konfiguraci systému, případně v nastavení automatizace viz obrázek 5.5.

5.3 Databáze, autentizace v systému, webhosting a lokální funkce systému

Jelikož je součástí řešení webová aplikace přístupná z Internetu a manipulace s daty, bylo nutné k tomu zvolit vhodné prostředky.

Databáze

V systému je potřeba nějakým způsobem ukládat data do databáze. Existují k tomu dva hlavní důvody. Prvním je, že některé části systému (především server) nemusejí být vždy v provozu a po opětovném uvedení do provozu je potřeba systém uvést do stavu před vypnutím. Bez uložení tohoto stavu by bylo opětovné nastavení komplikované. Druhým důvodem je, že jsem se rozhodl systém navrhnout tak, aby jej bylo možné ovládat i mimo lokální síť. Toho jsem docílil použitím vzdálené databáze (přístupné z Internetu), ke které má přístup jak webová aplikace, tak i lokální server a prostřednictvím ní spolu mohou snadno komunikovat. Samozřejmě bylo možné dosáhnout této komunikace přes Internet i alternativními řešeními, jako je:

- navázání spojení (z internetové webové aplikace) přímo se serverem na Raspberry Pi (pomocí Web Sockets),
- připojení k lokální webové aplikaci přes veřejnou IP adresu Raspberry Pi (internetová verze webové aplikace by tedy vůbec neexistovala).

Obě z těchto řešení však vyžadují nějakým způsobem Raspberry Pi „zviditelnit“ v síti Internet a to se sebou může nést rizika. Navíc by se tímto musel zabývat uživatel a systém jsem chtěl vytvořit tak, aby nebyl zbytečně složitý na konfiguraci. Proto jsem zvolil jako řešení vzdálenou (společnou) databázi, pomocí které spolu webová aplikace a server komunikují.

⁶⁵<https://html.spec.whatwg.org/multipage/custom-elements.html#custom-elements>

Dalším problémem, kterým jsem se v práci musel zabývat, byl způsob ukládání dat. Mezi možnosti, které jsem zvažoval, patří klasická relační a NoSQL databáze. Nakonec jsem zvolil NoSQL databázi a to z několika důvodů:

- Podobně jako struktura NoSQL databáze je i struktura systému hierarchická, data v databázi tedy v jistém směru přímo reprezentují danou domácnost. Kdyby měla být data uchovávaná v systému ve formě tabulek, byla by práce s databází komplikovanější a samotný návrh databáze náročnější.
- U NoSQL databází obvykle není potřeba investovat zpočátku tolik času návrhu databáze jako je tomu u relačních. Je tak možné pracovat inkrementálně a strukturu databáze průběžně měnit.
- V systému není potřeba dělat žádné složité manipulace s daty jako je třídění a filtrování. Z tohoto hlediska tedy není důvod využívat relační databázi (která by v opačném případě měla opodstatněné použití).

Při volbě konkrétního poskytovatele databáze jsem se rozhodl pro použití Realtime databáze od společnosti Google v rámci platformy Firebase⁶⁶. Zde jsem s výhodou využil funkci, kterou má tato databáze implementovanou a to je možnost naslouchání změnám v databázi. Tam, kde tedy zmiňuji naslouchání změnám v databázi, se nejedná o opakované dotazy na databázi, které by mohlo představovat zbytečnou zátěž pro databázi, ale skutečné naslouchání změnám.

Databáze přístupná z Internetu samozřejmě může představovat bezpečnostní riziko. To je však minimalizováno nastavením přístupových pravidel. Každý uživatel se k datům dostane až po autentizaci a to pouze k těm, která mu patří. Kromě toho je veškerá komunikace se vzdálenou databází zabezpečena pomocí protokolu SSL⁶⁷.

Databáze jsou v systému celkem 2. Jedna přístupná z Internetu (ta na platformě Firebase) a druhá je uložena lokálně na Raspberry Pi. Ve skutečnosti se jedná o prostý textový soubor ve formátu JSON⁶⁸. Ten je zpracováván aplikací serveru. Není potřeba jej ani vytvářet, v případě jeho absence si jej server vytvoří sám. Pokud je server spárovaný s uživatelským účtem, tak je tato lokální verze udržována v synchronizaci s tou na Internetu.

Autentizace

Protože je možné systém ovládat i přes Internet, musel jsem se zabývat autentizací uživatelů. Rozhodl jsem se, že nebudu vytvářet vlastní autentizační server, ale opět využiji služeb, které nabízí platforma Firebase. Ta poskytuje jednoduché API jak pro registraci, tak i přihlášení (autentizaci) uživatelů. Toho pak využívám v aplikaci. V internetové verzi aplikace je pochopitelně nutné se pro přístup do systému přihlásit (resp. zaregistrovat). Při použití lokální verze webové aplikace se pak uživatel přihlásit nemusí. Jen pokud chce systém ovládat přes Internet, tak se může přihlásit – nebo spíše spárovat server se vzdáleným uživatelským účtem. V lokální databázi se neukládají žádné údaje, které by nějak souvisely s uživatelským účtem (jako ID uživatele). Jelikož má systém fungovat jak lokálně, tak i přes Internet, tak bylo nutné vymyslet vhodný mechanismus, jak server s uživatelským účtem spárovat. Zvolil jsem řešení takové, že v případě přihlášení/registrace v lokální verzi

⁶⁶<https://firebase.google.com/>

⁶⁷Secure Sockets Layer

⁶⁸JSON – JavaScript Object Notation. Jedná se o jednoduchý zápis dat v textovém souboru, který je jednoduché převést na objekty v mnoha programovacích jazycích.

aplikace se vyplněný formulář pošle na lokální server a ten si údaje uloží. Následně se sám přihlásí (resp. autentizuje) a dále má již přístup k databázi daného uživatele na Internetu, se kterou již dále svoji databázi synchronizuje.

Webhosting

Pro webového klienta, přístupného z Internetu, bylo potřeba vyřešit, jak jej zpřístupnit. Jelikož jsem zavrhl možnost, že by se nějakým způsobem uživatelé přímo připojovali k Raspberry Pi (např. přes veřejnou IP adresu), rozhodl jsem se, že v systému bude druhá verze aplikace umístěna v síti Internet za využití webhostingu. Neviděl jsem důvod, proč nevyužít opět platformy Firebase, která i webhosting nabízí. Tato platforma nabízí zdarma registraci domény 3. řádu a SSL certifikát. Samozřejmě je možné k webhostingu připojit i vlastní doménu (2. řádu), ale jelikož je jedním z cílů práce brát ohled na nízkou cenu výsledného řešení, byla využita možnost domény 3. řádu zdarma. Kromě toho zbývající část domény (*.web.app*), kterou Firebase zdarma nabízí, působí i poněkud „atraktivně“ a nepůsobí až tak neprofesionálně. Výsledná internetová verze aplikace se nachází na adrese <https://automatizace.web.app/>.

Co se týče implementace, tak není žádný rozdíl mezi verzí aplikace na Internetu a lokální verzí. Některá nezbytná rozlišení (zda je aplikace spuštěna lokálně, nebo na webové doméně) se nacházejí již v aplikaci a není tak potřeba mít více vývojových verzí aplikace. Také z hlediska uživatele je práce s aplikací téměř stejná bez ohledu na to, zda používá lokální verzi, nebo verzi na Internetu. V podstatě jediným rozdílem je nutnost přihlášení v internetové verzi.

5.4 Implementace serveru na Raspberry Pi a aplikace pro koncové moduly

Podobně jako webový klient je i server napsaný v programovacím jazyce Typescript. Důvod, který mě k tomu vedl, je takový, že je výhodné mít více částí aplikace implementováno v jednom programovacím jazyce. Rovněž se zde opět uplatní i ta výhoda, že se v případě Javascriptu (výstupu Typescriptu) jedná o jeden z nejpopulárnějších a rychle se rozvíjejících programovacích jazyků. Tím bude do jisté míry usnadněn budoucí vývoj. Aby bylo možné na Raspberry Pi spustit Javascript (který je výstupem Typescriptu) mimo internetový prohlížeč, nainstaloval jsem na něj Javascriptové běhové prostředí NodeJS. Protože Raspberry Pi slouží jako centrální jednotka ze které má být možné kdykoli systém ovládat z připojeného displeje, bylo potřeba vypnout režim spánku, aby bylo zařízení neustále v pohotovostním režimu. Toho jsem dosáhl pomocí programu XScreenSaver⁶⁹.

Aplikaci pro koncové moduly jsem implementoval v jazyce Wiring a C++. Jedná se o hojně používané programovací jazyky pro podobné platformy, existuje tedy velká podpora. Takže i zde jsem myslel především na budoucí vývoj.

Komunikační protokol

Pro komunikaci mezi zařízeními bylo potřeba vybrat vhodný aplikační protokol. Mezi možnostmi, které jsem zvažoval, patří MQTT a CoAP. Oba protokoly vyhovují malou strukturou přenášených zpráv. U MQTT protokolu se předpokládá, že centrální prvek bude fungovat

⁶⁹<https://www.jwz.org/xscreensaver/>

jako jistý směrovač zpráv a celková zodpovědnost za interpretaci dat bude na jednotlivých koncových zařízeních. V mé implementaci však centrální prvek funguje spíše jako řídicí jednotka, která rozhoduje, která data je potřeba kam posílat a v jaký okamžik (zejména při automatizaci). Z tohoto důvodu jsem nakonec zvolil CoAP. Pro server na Raspberry Pi jsem využil dostupnou knihovnu *coap*⁷⁰ a pro moduly knihovnu *CoAP simple*⁷¹.

Konfigurace vývojových desek s ESP8266

Před kompilací a uložením kódu pro ESP8266 do paměti modulu je potřeba upravit konfigurační část v hlavičkovém souboru *ESP.h*. V té je potřeba zapsat správný identifikátor Wi-Fi sítě (SSID) a heslo k této síti, ve které má systém fungovat. Nastavování údajů o síti se sice může zdát jako zbytečný krok (systém by se o výměnu těchto informací mohl postarat sám), nicméně v rámci bakalářské práce nebylo řešeno šifrování přenášených dat, které by zde bylo potřeba.

```
15  /**
16   * ZAČÁTEK KONFIGURAČNÍ ČÁSTI (doporučeno změnit)
17   */
18
19  String boardType = "NodeMCU"; // Cokoli z [wemosD1R1, NodeMCU, esp01]
20  const char *ssid = "DOPLNIT";
21  const char *password = "DOPLNIT";
22  IPAddress moduleStaticIP(192, 168, 1, 103); // statická IP modulu, pokud ji uživatel chce používat
23  // Pokud uživatel nechce používat statickou IP, odkomentovat řádek níže a zakomentovat ten výše...
24  //IPAddress moduleStaticIP;
25  IPAddress gateway(192, 168, 1, 1); //IP adresa výchozí brány (pro použití statické IP)
26  IPAddress subnet(255, 255, 255, 0); // Maska podsítě (pro použití statické IP)
27  /**
28   * KONEC KONFIGURAČNÍ ČÁSTI
29   */
```

Obrázek 5.7: Ukázka kódu, ve kterém se konfiguruje modul.

Následující komunikace (po úspěšném připojení modulu do sítě) probíhá výhradně v rámci lokální sítě. Předpokládá se tedy jisté zabezpečení na této úrovni. Navíc zabezpečení při komunikaci v lokální síti (ve smyslu šifrování pomocí protokolu SSL/TLS) nevyužívají i některé komerčně prodávané systémy, např. dálkově ovládané zásuvky od společnosti Kangtai. To jsem si ověřil pomocí aplikace na android Packet Capture⁷². Při nasazení na „produkci“ by bylo samozřejmě vhodné zajistit bezpečnou (šifrovanou) komunikaci i v lokální síti, ale v rámci bakalářské práce nebyl tento problém řešen. V případě potřeby je v konfigurační části možné změnit také typ vývojové desky, který je nastaven na „NodeMCU“ a statickou IP adresu modulu.

Komunikace mezi Raspberry Pi a ESP8266

Na počátku komunikace mezi Raspberry Pi a jednotlivými moduly stojí přidání nového modulu do databáze (typicky v nastavení systému klientské aplikace). Lokální server neustále naslouchá změnám v databázi a pokud zaregistruje přidání modulu, tak se pokusí kontaktovat nový modul CoAP zprávou. V tuto chvíli ještě server nezná IP adresu potenciálního modulu, bylo tedy potřeba zvolit vhodný mechanismus, jakým způsobem modul

⁷⁰<https://www.npmjs.com/package/coap>

⁷¹<https://www.arduino.cc/reference/en/libraries/coap-simple-library/>

⁷²<https://play.google.com/store/apps/details?id=app.greyshirts.sslcapture>

kontaktovat. Server posílá zmíněnou zprávu tzv. *All CoAP Nodes* multicastové skupině (IP adresa *224.0.1.187*). Nové, dosud nepřidané, moduly v této multicastové skupině naslouchají a v případě příchozí zprávy odpoví opět CoAP zprávou, v rámci které posílají informaci o tom, o jaký typ vývojové desky se jedná. Toho pak využívá webový klient, když uživateli nabízí možné V/V piny v nastavení systému pro daný modul. Pokud server získá odpověď, aktualizuje záznam o modulu v databázi (vloží zde IP adresu modulu a typ vývojové desky). Následně znovu kontaktuje již přidáný modul. Ten si až v tuto chvíli ukládá IP adresu serveru (zpráva je již totiž posílána jako unicast na konkrétní modul). Ten od tohoto okamžiku přestane naslouchat multicastovým CoAP zprávám.

K již přidáním modulům lze přes klientskou aplikaci přidávat snímače a výstupní zařízení. Do databáze se vloží příslušný záznam a v reakci na tento nový záznam se daný snímač/výstupní zařízení objeví na domovské stránce ve webové aplikaci. Server zaznamená změnu v databázi a jako reakci pošle zprávu příslušnému modulu. V případě nového snímače odešle modulu žádost o sledování hodnoty na daném vstupu. Od této chvíle modul kontroluje v pravidelných intervalech daný vstup a v případě dostatečné změny pošle informaci na server, který změny uloží do databáze. Pro jednotlivé snímače je v kódu definovaný minimální rozdíl hodnot, která se odešle oproti poslední odeslané a to z toho důvodu, aby nebyla síť zbytečně zahlcovaná a pak také aby server nebyl tak zaneprázdněn přijímáním zbytečných zpráv. Některé snímače totiž mění naměřené hodnoty i v řádu několika desetinných míst, což například u teploty v místnosti nemá smysl sledovat. Tyto informace o naměřených hodnotách se posílají jako nepotvrditelné zprávy, jelikož jich může z modulů přicházet relativně velké množství a ve většině případů není nijak kritické, pokud server nějakou zprávu občas nezachytí.

Dále pokud se přidává nové výstupní zařízení, odešle se na modul zpráva o nastavení daného pinu jako výstup a hodnotu, kterou má modul na výstup nastavit. Při přidání nového výstupního zařízení je tato hodnota 0 (tedy vypnuto). Pokud následně uživatel ovládá hodnotu na daném výstupu z domovské stránky klientské aplikace, odešle se stejná zpráva, jako při přidávání zařízení s tím rozdílem, že tentokrát se přenáší jiná hodnota. Ovládání výstupů je v systému poněkud citlivější záležitost (uživatel chce mít jistotu, že například světla, která vypnul, jsou skutečně vypnutá), a proto se tyto zprávy posílají jako potvrditelná CoAP zpráva. Server tedy po odeslání zprávy čeká na potvrzení a pokud jej v daném časovém limitu neobdrží (pokud nepřekročil maximální počet pokusů), tak se pokusí požadavek poslat znovu.

Při spuštění modul kontroluje, zda má (ve flash paměti) uloženou IP adresu serveru. Pokud ano, tak na server pošle dotaz, jaké vstupy má sledovat a také jak má nastavit své výstupy. Pokud byl modul mezitím odebrán ze systému, tak mu server odpoví příslušnou zprávou a nastavení modulu se resetuje (opět naslouchá v multicastové skupině). Tím je tedy zaručeno, že systém bude fungovat korektně například i při dočasném výpadku napájení.

Statické a dynamické IP adresy

IP adresy jednotlivých modulů jsou uchovávány v databázi. V ideálním případě uživatel jednotlivým modulům v konfigurační části nastaví jejich statickou IP adresu. Stejně tak je vhodné nastavit statickou IP adresu na zařízení, na kterém je spuštěn server (Raspberry Pi). V tomto případě systém bude fungovat bez problému, co se týče IP adres. To od uživatele samozřejmě vyžaduje při prvotním nastavování systému práci navíc – nastavení/kontrolu, aby DHCP server dané IP adresy nepřisuzoval žádným zařízením. Tento krok nastavování statických IP adres a DHCP serveru je v systému možné vynechat. Potom moduly a server

používají dynamické IP adresy, přiřazené DHCP serverem. Za normálních okolností bude systém v tomto nastavení fungovat korektně. Mohl by ale nastat problém v případě nějakého restartu zařízení, na kterém je spuštěn DHCP server (typicky Wi-Fi router) nebo i v případě, že by některý modul byl delší dobu vypnutý. Pak by po spuštění mohl od DHCP serveru obdržet jinou IP adresu, zatímco server by se s ním snažil komunikovat přes tu bývalou. Tomuto problému je možné se tedy vyhnout nastavením statických IP adres pro moduly a to ve vyhrazené konfigurační části. Pak je také vhodné nastavit statickou IP adresu serveru (již manuálně).

5.5 Testování a vyhodnocení systému

Výsledný systém bylo potřeba otestovat, aby bylo zaručeno, že odpovídá specifikaci, uvedené v kapitole 4. Také tyto testy mohou vést k odhalení některých nedostatků a nedokonalostí. Výsledky testování je možné využít k dalšímu vývoji. Testy, které je vhodné provést, jsou následující:

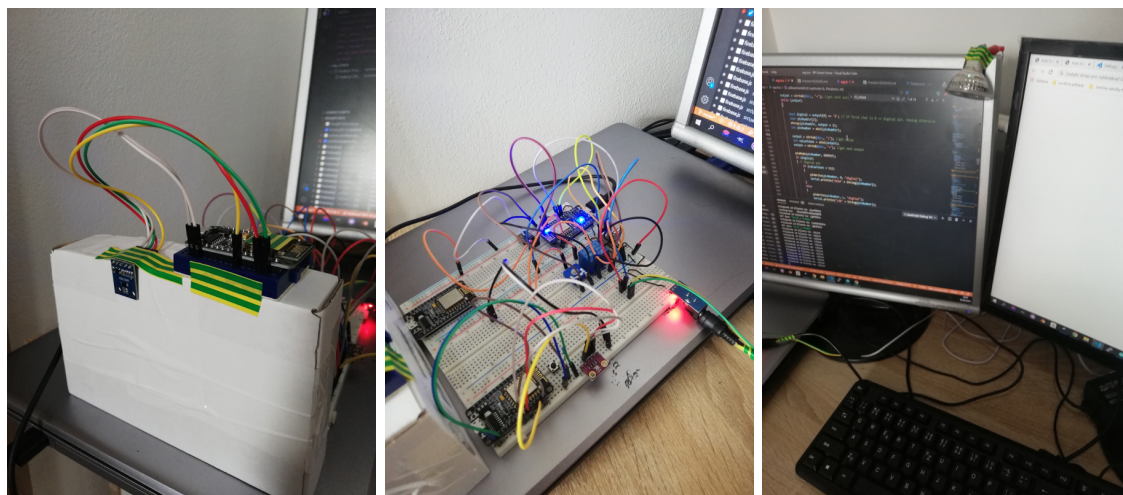
- test přímého ovládání zařízení v systému a časovačů,
- test použití snímačů k zobrazení hodnot a automatizaci,
- test spárování serveru s uživatelským účtem z Internetu,
- test funkce systému v lokální síti a přes Internet,
- test intuitivnosti webového klienta,
- test podporovaných prohlížečů pro běh webového klienta,
- test multiplatformnosti serveru,
- test spolehlivosti systému při velkém množství uživatelů.

V rámci práce však nebylo možné provést všechny tyto testy a navíc v potřebném rozsahu. Obecně aby bylo možné spolehlivě otestovat všechny vlastnosti systému, tak by bylo potřeba mít k dispozici velké množství uživatelů a věnovat tomu mnohem delší čas, než jaký jsem měl k tvorbě této práce. Následuje popis jednotlivých testů, kde je vždy uvedeno, zda (a u některých případně i jak) byl daný test prováděn.

Test přímého ovládání zařízení v systému a časovačů zahrnuje přidání několika modulů do systému. K těm je potřeba připojit relé a další součástky a k nim dále ovládaná zařízení, jako LED světla. Následně je potřeba testovat, zda při ovládání přidaných zařízení z domovské stránky webové aplikace systém adekvátním způsobem mění svůj stav (tzn. zda se světlo rozsvěcuje a zhasíná, případně stmívá na základě požadavků). Otestování časovačů pak spočívá v nastavení několika časovačů ve webové aplikaci a pozorování, zda po uplynutí nastavené doby dochází k žádané akci. Tento test jsem provedl. Při testování jsem používal celkem 4 moduly NodeMcu a testováním jsem nezjistil žádné problémy.

Test použití snímačů k zobrazení hodnot a automatizaci zahrnuje přidání snímačů do systému přes nastavení webové aplikace. Je potřeba porovnat naměřené údaje na snímačích, zda odpovídají skutečnosti (s jistou tolerancí samozřejmě). Test automatizace spočívá v nastavení automatizací výstupů dříve přidaných modulů na základě hodnot snímačů. Tento test jsem rovněž prováděl. Porovnáním hodnot snímačů s meteorologickou stanicí jsem si mohl ověřit funkčnost snímačů BMP280 a SHT21. Funkci snímače BH1750 jsem neměl jak

ověřit a bylo by vhodné ji ověřit v budoucnu. V rámci testu automatizace jsem nastavil, aby se rozsvítilo světlo připojené k modulu, pokud klesne intenzita osvětlení pod určitou úroveň (kterou jsem definoval na základě experimentů). Fotku z testování je možné vidět na obrázku 5.8. Při testování jsem se opět nesetkal s žádnou chybou, použití snímačů v systému a automatizace na základě jejich hodnoty tedy odpovídá specifikaci.



Obrázek 5.8: Testování automatizace na základě hodnoty snímače. Vlevo je možné vidět snímač intenzity osvětlení (*BH1750*) připojený k modulu NodeMcu. Uprostřed je nepáživé pole s dalšími moduly. K modulu s modře svítícími diodami je připojena LED žárovka. Modul spíná relé pro připojení této žárovky k napětí 12 V. Na obrázku vpravo je možné vidět tuto LED žárovku, která se při poklesu úrovně osvětlení na snímači *BH1750* pod jistou úroveň rozsvítí.

Test spárování serveru s uživatelským účtem z Internetu ověří, že funguje správně spárování serveru s účtem na Internetu. Provede se tak, že se uživatel zaregistruje v internetové verzi aplikace (tedy na adrese <https://automatizace.web.app/>). Následně je vhodné vyzkoušet, že internetová a lokální verze nejsou nijak propojené. Například je možné v obou verzích aplikace vytvořit novou místnost, která by neměla být nijak v druhé verzi zobrazena. Následně je potřeba v lokální verzi aplikace přejít na spárování účtu (z menu). Zde je nutné vyplnit správné přihlašovací údaje a kliknout na „Přihlásit“. Dále je potřeba znovu vyzkoušet přidat nějakou místnost v obou verzích aplikace. Nyní by již server měl být s účtem spárován a jakékoli změny provedené v jedné verzi aplikace by se měly ihned reflektovat do druhé verze. Tento test byl proveden podle popsaného postupu a nevyskytly se žádné problémy. Po spárování bylo možné libovolně používat lokální i internetovou verzi aplikace a systém reagoval adekvátním způsobem.

Test funkce systému v lokální síti a přes Internet má za úkol ověřit především 2 věci:

- zda je možné systém provozovat jen v rámci lokální sítě,
- zda je systém schopný při ztrátě připojení k Internetu fungovat správně po opětovném získání připojení.

Test je možné provést tak, že uživatel vypojí a opět zapojí internetový kabel ze svého routeru. Pokud během chvíle, kdy byl systém odpojen, uživatel neprováděl žádnou akci, tak by po připojení k Internetu měl být systém ve stavu, v jakém se nacházel při odpojení.

V případě, že mezitím ovládal systém v rámci lokální sítě, tak by se změny, které udělal, měly promítnout do internetové verze aplikace (resp. její databáze). A naopak v případě, že uživatel dělal v systému změny přes internetovou verzi aplikace během režimu, kdy byl systém bez Internetu (např. přes data na mobilním telefonu či z jiné sítě, která připojení k Internetu má), by se měly změny po připojení projevit. Pokud tedy například během odpojení vypnul nějaké zařízení, musí se zařízení po připojení systému k Internetu vypnout. Je vhodné zde zmínit, že za aktuální se vždy považuje ta verze databáze (internetová nebo lokální), do které se jako poslední zapisovaly změny. To znamená, že pokud uživatel ovládal systém během doby, kdy byl bez Internetu v obou verzích aplikace, tak se projeví jen změny verze, kterou ovládal jako poslední. I tento test jsem provedl podle popsaného postupu a nenalezl jsem žádné problémy.

Test intuitivnosti webového klienta zahrnuje testování systému uživateli, kteří jej neznají. Ti dostanou instrukce co mají provést a je potřeba pozorovat, jak rychle přijdou na řešení. Tímto testem je možné odhalit některé chyby grafického rozvržení a případně aplikaci na základě postřehů uživatelů upravit. Tento test byl proveden. Uživateli, kteří systém testovali, zde bylo několik členů domácnosti. Dostali za úkol provést některé akce, jako například přidat do systému nový snímač, či nastavit automatizaci na základě tohoto snímače. Obecně si vedli dobře a testováním nebyly nalezeny žádné vážné problémy v grafickém rozhraní.

Test podporovaných prohlížečů pro běh webového klienta spočívá v používání webové aplikace v různých prohlížečích a pozorování, zda se chová dle návrhu. V případě zjištění, že aplikace nefunguje korektně pro některý z moderních populárních webových prohlížečů, by bylo vhodné v rámci budoucího vývoje pracovat na podpoře daným prohlížečem. V rámci testování jsem ověřil funkčnost systému ve webových prohlížečích Google Chrome a Microsoft Edge, oba prohlížeče ve verzi 90 pro PC. Dále jsem také testoval Google Chrome pro OS Android (opět verze 90). Tímto testováním jsem nenašel žádné chyby, které by souvisely s podporou prohlížečů. Použití těchto prohlížečů v daných verzích tedy systém podporuje. Dále by bylo vhodné provést testování pro více moderních prohlížečů, případně i v různých verzích. Tyto další testy již prováděny nebyly.

Server byl vytvořen s ohledem na běh na Raspberry Pi. Běží však v prostředí Node.js, které je možné spustit na mnoha jiných zařízeních (resp. operačních systémech). Z toho tedy vyplývá, že by server mohl být multiplatformní. Je tedy vhodné otestovat jeho chování i na jiných zařízeních, protože ne každý vlastní Raspberry Pi. Někteří uživatelé by mohli chtít spustit server i na alternativních mikropočítačích (nebo obecně jakémkoli počítači). Je tedy vhodné se zabývat i multiplatformností serveru. I tyto výsledky by mohly vést k dalšímu vývoji. Server jsem otestoval kromě na Raspberry Pi ve verzi 3B+ také na notebooku s OS Windows 10, procesorem Intel Core i5-7200U a RAM pamětí typu DDR4 o velikosti 4 GB. Server fungoval správně a je tedy možné jej provozovat i na podobných zařízeních. Zajímavé by zde bylo otestovat, zda bude systém plynule fungovat i na levných verzích Raspberry Pi, například na verzi Zero⁷³, kterou je možné pořídit za méně než 500 Kč. V případě bezproblémového chodu by byl projekt skutečně velmi zajímavou alternativou k některým drahým systémům, jako je Loxone, kde se jen cena Miniserveru (který v jejich systému plní podobnou funkci jako Raspberry Pi v mém) pohybuje okolo 10 000 Kč. Testování na jiných verzích Raspberry Pi jsem však neprováděl.

Dále by bylo také vhodné otestovat, jak by se systém choval v případě velkého množství uživatelů v systému. Prováděl by se tak, že by se pozorovalo, zda se výrazně neprodlužuje

⁷³<https://rpishop.cz/raspberry-pi/685-raspberry-pi-zero-wh-4250236816296.html>

odezva systému při častém využívání mnoha domácnostmi, ty totiž v případě spárování s uživatelským účtem sdílejí společnou vzdálenou databázi. Tento test nebyl z časových důvodů realizován a je vhodné jej v budoucnu doplnit.

Nakonec je také potřeba zhodnotit, jak se podařil naplnit cíl nízké ceny výsledného systému. Nejdražší součástí systému jsou Raspberry Pi a připojený dotykový displej, které dohromady (i s nezbytnými součástmi jako je napájecí zdroj, MicroSD karta apod.) přišlo na přibližně 3 500 Kč. Použití dotykového displeje však není v systému pro funkci nezbytnou podmínkou a bez něj je tato cena téměř poloviční. Další součásti systému tvoří moduly. Jejich cena se však pohybuje okolo pouhých 150ti Kč. K těm se připojují senzory s cenou okolo 50ti Kč. Nakonec je samozřejmě potřeba k modulům připojit napájecí zdroj. Jeho cena je přibližně 150 Kč. Cena součástek k ovládání jiných zařízení (jako relé či tranzistorů) je pak vzhledem k celkové ceně zanedbatelná. Jak je možné snadno sečíst, cena výsledného systému zahrnujícího několik modulů, snímačů a pomocných součástek k ovládání výstupů, by se měla pohybovat do 5 000 Kč. Samozřejmě zde záleží na potřebách daného uživatele, ale celkově je možné tento systém ve srovnání s řešením od Loxone či Jablotron považovat za mnohem levnější variantu.

Všechny funkce, které jsem otestoval, byly testovány v rozsahu, který byl v rámci bakalářské práce možný a nebyly přitom odhaleny žádné chyby. Systém tedy splňuje specifikaci, jaká byla definována v kapitole 4. V testování je však vhodné nadále pokračovat, zejména každodenním používáním systému. Také je vhodné doplnit chybějící testy.

Kapitola 6

Závěr

Cílem práce bylo vyvinout systém automatizované domácnosti s využitím Wi-Fi, který by mohl ovládat jiná zařízení a zároveň je automatizovat. Tento cíl byl splněn.

V práci byl shrnut současný stav v oblasti vestavěných systémů a technologií dálkového přenosu. Byl navržen systém pro ovládání nejrůznějších zařízení. Výsledné řešení se skládá ze tří aplikací – serveru, webového klienta a aplikace pro koncové moduly. Server běží na platformě Raspberry Pi a jako koncové moduly byly zvoleny ESP8266. Řešení obsahuje funkci přímého ovládání, ale také časovače a automatizaci na základě hodnot snímačů. K webovému klientovi je možné přistupovat jednak v lokální síti pomocí IP adresy serveru, ale také přes Internet.

Práce mě naučila pracovat s některými službami platformy Firebase. Zejména jsem se mohl zdokonalit v oblasti návrhu a správy NoSQL databáze. Díky rozmanitosti zařízení, který podobné systémy obsahují, jsem si mohl vyzkoušet návrh různých aplikací, které spolu komunikují. V rámci práce jsem tedy získal nové zkušenosti i v oblasti počítačových sítí a návrhu software.

Systém chci uvolnit jako open source, takže by případný zájemce také mohl dále systém rozšiřovat. Aktuální zdrojové soubory je možné nalézt na GitHubu⁷⁴. V práci by bylo vhodné se v rámci pokračování zabývat možnostmi napájení koncových modulů, protože ne všude je možné používat napájení ze sítě. Řešení by mohlo zahrnovat použití akumulátoru a nějakého úsporného mikrokontroléru, který by použité koncové moduly využíval pouze ke komunikaci po síti. Ten by také mohl používat režim spánku, aby se tak šetřil případný akumulátor. Dále by systém mohl podporovat hlasové ovládání či spolupracovat s jinými službami a systémy pomocí IFTTT služby. Já osobně bych se chtěl v budoucnu zaměřit především na přidání podpory více typů snímačů.

⁷⁴<https://github.com/dinoq/RPi-Smart-Home>

Literatura

- [1] *Air Base Extension Datasheet* [online]. Rakousko: Loxone, 2021. Dostupné z: https://www.loxone.com/wp-content/uploads/datasheets/Datasheet_AirBaseExtension.pdf.
- [2] AL SARAWI, S., ANBAR, M., ALIEYAN, K. a ALZUBAIDI, M. Internet of Things (IoT) communication protocols: Review. In: *2017 8th International Conference on Information Technology (ICIT)*. 2017, s. 685–690. DOI: 10.1109/ICITECH.2017.8079928.
- [3] ALBANO, S. WiFi frequency bands: 2.4 GHz and 5 GHz. *Minim: the residential managed WiFi and IoT security platform* [online]. Spojené království: Minim®, 2019 [cit. 17. ledna 2020]. Dostupné z: <https://www.minim.co/blog/wifi-frequency-bands-2.4-ghz-and-5-ghz>.
- [4] ATMOKO, R., RIANTINI, R. a HASIN, M. IoT real time data acquisition using MQTT protocol. *Journal of Physics: Conference Series*. 2017, sv. 853, s. 012003. DOI: 10.1088/1742-6596/853/1/012003.
- [5] AZZOLA, F. CoAP Protocol: Step-by-Step Guide. *DZone* [online]. Spojené státy: DZone®, 2018 [cit. 12. prosince 2020]. Dostupné z: <https://dzone.com/articles/coap-protocol-step-by-step-guide>.
- [6] BECK, D. *Hello Emma Watch - Návod na ovládání CZ* [online]. Česká republika: David Beck, 2020. Dostupné z: <https://helloemma.cz/static/helloemma-watch-navod-5e923eba1890f6c8666ea55131d50793.pdf>.
- [7] CHAN, J. Bosch home connect takes smart appliances mainstream. *Reviewed* [online]. Spojené státy: Gannett Satellite Information Network LLC, 2020 [cit. 10. listopadu 2020]. Dostupné z: <https://www.reviewed.com/refrigerators/features/bosch-home-connect-smart-appliance-review>.
- [8] CHARLTON, A. Yes, you can build and run a smart home without a voice assistant. *GearBrain* [online]. Spojené státy: GearBrain, Inc, 2019 [cit. 2. července 2020]. Dostupné z: <https://www.gearbrain.com/smart-home-without-voice-control-2641525934.html>.
- [9] DAS, A. 15 Single Board Computers: Alternatives to Raspberry Pi. *It's FOSS* [online]. Indie: It's FOSS, 2020 [cit. 1. dubna 2021]. Dostupné z: <https://itsfoss.com/raspberry-pi-alternatives/>.
- [10] DASHKOVA, E. a GURTOV, A. Survey on Congestion Control Mechanisms for Wireless Sensor Networks. In: GALININA, O., ANDREEV, S., BALANDIN, S. a KOUCHERYAVY,

- Y., ed. *Internet of Things, Smart Spaces, and Next Generation Networking*. Německo: Springer Berlin Heidelberg, 2012, s. 75–85. ISBN 978-3-642-32686-8.
- [11] DELANEY, J. R. a MOSCARITOLO, A. What Is a Smart Home Hub (And Do You Need One)? *PCMag* [online]. Spojené státy: Ziff Davis, 2021 [cit. 13. dubna 2021]. Dostupné z: <https://www.pcmag.com/news/what-is-a-smart-home-hub-and-do-you-need-one>.
- [12] *Dimmer Extension Datasheet* [online]. Rakousko: Loxone, 2020. Dostupné z: https://www.loxone.com/wp-content/uploads/datasheets/Datasheet_DimmerExtension.pdf.
- [13] EADY, F. *Hands-On ZigBee: Implementing 802.15.4 with Microcontrollers*. 1. vyd. UK: Elsevier, Newnes, 2007. Embedded technology. ISBN 9780123708878.
- [14] EL BENDARY, M. A. M. *Wireless Personal Communications: Simulation and Complexity*. 1. vyd. Singapur: Springer, 2018. Signals and Communication Technology. ISBN 9789811071300.
- [15] Embedded Systems - Overview. *Tutorials Point* [online]. Indie: Tutorials Point [cit. 3. března 2021]. Dostupné z: https://www.tutorialspoint.com/embedded_systems/es_overview.htm.
- [16] FILIP, J. Začínáme stavět chytrou domácnost na HomeKit – část 2. *Letem světem Applem* [online]. Česká republika: Text Factory s. r. o., 2020 [cit. 9. listopadu 2020]. Dostupné z: <https://www.letemsvetemapplem.eu/2020/01/16/zaciname-stavet-chytrou-domacnost-na-homekit-cast-2/>.
- [17] FILIP, J. Začínáme stavět chytrou domácnost na HomeKit – část 3. *Letem světem Applem* [online]. Česká republika: Text Factory s. r. o., 2020 [cit. 9. listopadu 2020]. Dostupné z: <https://www.letemsvetemapplem.eu/2020/02/03/zaciname-stavet-chytrou-domacnost-na-homekit-cast-3/>.
- [18] FILIP, J. Začínáme stavět chytrou domácnost na HomeKit – část 1. *Letem světem Applem* [online]. Česká republika: Text Factory s. r. o., 2020 [cit. 9. listopadu 2020]. Dostupné z: <https://www.letemsvetemapplem.eu/2020/01/09/zaciname-stavet-chytrou-domacnost-na-homekit-cast-1/>.
- [19] FILIP, J. Apple potvrdil vývoj Siri v češtině. *Letem světem Applem* [online]. Česká republika: Text Factory s. r. o., 2020 [cit. 28. dubna 2021]. Dostupné z: <https://www.letemsvetemapplem.eu/2020/09/29/siri-v-cestine-3/>.
- [20] FROMAGET, P. The Epic Story of the Raspberry Pi. *RaspberryTips* [online]. Francie: Patrick Fromaget [cit. 5. března 2021]. Dostupné z: <https://raspberrytips.com/raspberry-pi-history/>.
- [21] GANGULI, M. *Getting started with Bluetooth*. 1. vyd. USA: Premier Press, 2002. Networking. ISBN 9781931841832.
- [22] GERHART, J. *Home automation and wiring*. UK: McGraw-Hill Publishing, 1999. ISBN 9780070246744.
- [23] GISLASON, D. *Zigbee Wireless Networking*. 1. vyd. UK: Elsevier, Newnes, 2008. ISBN 9780750685979.

- [24] HAMERNIK, P., MUDRONCIK, D. a PAVOL, T. Classification of Functions in Smart Home. *International Journal of Information and Education Technology* [online]. Slovensko: International Journal of Information and Education Technology. Duben 2012, sv. 2, s. 149–155. DOI: 10.7763/ijiet.2012.v2.98. Dostupné z: https://www.researchgate.net/publication/275156028_Classification_of_Functions_in_Smart_Home.
- [25] HARPER, R. *Inside the Smart Home*. 1. vyd. UK: Springer, 2003. ISBN 978-1-85233-688-2.
- [26] HERWIG, B. Co to je a jak funguje chytrý dům, chytrý byt a chytrá domácnost? *Lupa* [online]. Česká republika: Internet Info, s.r.o., 2013 [cit. 20. dubna 2021]. Dostupné z: <https://www.lupa.cz/clanky/co-to-je-a-jak-funguje-chytry-dum-chytry-byt-a-chytra-domacnost/>.
- [27] Home Automation - DIY or Professionally Installed? *Toyama* [online]. Indie: Toyama Controls & Systems Pvt., březen 2021 [cit. 2. dubna 2021]. Dostupné z: <https://www.toyamaworld.com/post/home-automation-diy-or-professionally-installed>.
- [28] HORT, J. Nová generace Miniserveru Go. *Loxone.com* [online]. Česká republika: Loxone, 2020 [cit. 10. listopadu 2020]. Dostupné z: <https://www.loxone.com/cscz/blog/nova-generace-miniserveru-go/>.
- [29] HOY, M. B. Alexa, Siri, Cortana, and More: An Introduction to Voice Assistants. *Medical Reference Services Quarterly*. 2018, sv. 37, s. 81 – 88.
- [30] IBRAHIM, D. Chapter 3 - ARM Cortex microcontroller development boards. In: IBRAHIM, D., ed. *Arm-Based Microcontroller Multitasking Projects*. Newnes, 2021, s. 33–45. DOI: <https://doi.org/10.1016/B978-0-12-821227-1.00003-7>. ISBN 978-0-12-821227-1. Dostupné z: <https://www.sciencedirect.com/science/article/pii/B9780128212271000037>.
- [31] *J-Link – Návod pro instalaci a použití* [online]. Jablonec nad Nisou: Jablotron. Dostupné z: https://www.jablotron.com/cz/o-jablotronu/ke-stazeni/?filename=j-link_cs_mlj26201.pdf&do=downloadFile.
- [32] *Jablotron 100 uživatelský manuál* [online]. Jablonec nad Nisou: Jablotron. Dostupné z: https://www.jablotron.com/cz/o-jablotronu/ke-stazeni/?filename=ja-10xk_user_cs_mlj52010_web.pdf&do=downloadFile.
- [33] JOSE, J. *Internet of Things*. 1. vyd. Indie: Khanna Book Publishing Co. (P) Ltd., leden 2010. ISBN 9789386173591.
- [34] KAMAL, R. *Embedded Systems: Architecture, Programming and Design*. 2. vyd. Indie: Tata McGraw-Hill Publishing Company Limited, 2008. ISBN 9780073404561.
- [35] KAMMER, D., MCNUTT, G. a SENESE, B. *Bluetooth Application Developer's Guide: The Short Range Interconnect Solution*. USA: Syngress Publishing, Inc, 2001. ISBN 9781928994428.
- [36] Komplexní řešení inteligentního ovládání. *Create Automation* [online]. Rakousko: Loxone. 2020, č. 2, s. 30–31. Dostupné z: https://www.loxone.com/cscz/wp-content/uploads/sites/7/2020/05/Loxone_Magazine_2020-CZ-web.pdf.

- [37] LABIOD, H., AFIFI, H. a DESANTIS, C. *Wi-Fi™, Bluetooth™, Zigbee™ and WiMax™*. USA: Springer, 2007. ISBN 9781402053962.
- [38] LEE, J. J., KIM, K. T. a YOUN, H. Y. Enhancement of congestion control of Constrained Application Protocol/Congestion Control/Advanced for Internet of Things environment. *International Journal of Distributed Sensor Networks*. 2016, sv. 12. DOI: 10.1177/1550147716676274.
- [39] LUTKEVICH, B. Embedded system. *IoTAgenda* [online]. Spojené státy: TechTarget, 2020 [cit. 1. března 2021]. Dostupné z: <https://internetofthingsagenda.techtarget.com/definition/embedded-system>.
- [40] MACKENZIE, I. Raspberry Pi Pico release! *PiShop* [online]. Spojené státy: PiShop.us, 2021 [cit. 8. března 2021]. Dostupné z: <https://www.pishop.us/blog/raspberry-pi-pico-release/>.
- [41] MALÝ, M. Protokol MQTT: komunikační standard pro IoT. *Root.cz - informace nejen ze světa Linuxu* [online]. Česká republika: Internet Info, s.r.o., 2016 [cit. 14. prosince 2021]. Dostupné z: <https://www.root.cz/clanky/protokol-mqtt-komunikacni-standard-pro-iot/>.
- [42] MCTEAR, M., CALLEJAS, Z. a GRIOL, D. *The conversational interface: Talking to smart devices*. 1. vyd. Švýcarsko: Springer International Publishing, 2016. ISBN 978-3319329659.
- [43] MIDWINTER, C. J. Smart Home vs. Home Automation – What’s the difference? *149 Walnut* [online]. Kanada: Craig J. Midwinter, srpen 2018 [cit. 1. dubna 2021]. Dostupné z: <https://149walnut.com/2018-07-smart-home-vs-home-automation-whats-the-difference/>.
- [44] MILJKOVAC, I. Raspbian – Raspberry Pi OS Guide. *PCWDL D* [online]. Spojené státy: PCWDL D, 2021 [cit. 18. dubna 2021]. Dostupné z: <https://www.pcwdd.com/raspbian-raspberry-pi-os-guide>.
- [45] *Miniserver Datasheet* [online]. Rakousko: Loxone, 2021. Dostupné z: https://www.loxone.com/wp-content/uploads/datasheets/Datasheet_Miniserver.pdf.
- [46] *Miniserver Gen. 1 Datasheet* [online]. Rakousko: Loxone, 2021. Dostupné z: https://www.loxone.com/wp-content/uploads/datasheets/Datasheet_Miniserver_Gen1.pdf.
- [47] *Miniserver Go Datasheet* [online]. Rakousko: Loxone, 2020. Dostupné z: https://www.loxone.com/wp-content/uploads/datasheets/Datasheet_MiniserverGo.pdf.
- [48] Nejen chytrou domácnost. Můžete mít i chytrou firmu! *Jablotron* [online]. Jablonec nad Nisou: Jablotron, 2019 [cit. 10. srpna 2020]. Dostupné z: <https://www.jablotron.com/cz/o-jablotronu/blog/zabezpecena-chytra-firma/nejen-chytrou-domacnost-muzete-mit-i-chytrou-firmu/>.
- [49] TEAM, T. H. Quality of Service 0,1 & 2 – MQTT Essentials: Part 6. *HiveMQ* [online]. Německo: HiveMQ GmbH, 2015 [cit. 14. prosince 2021]. Dostupné z: <https://www.hivemq.com/blog/mqtt-essentials-part-6-mqtt-quality-of-service-levels/>.

- [50] RICHTER, F. Smart Home Technology Poised for Blockbuster Growth. *Statista* [online]. Německo: Statista GmbH, říjen 2018 [cit. 2. dubna 2021]. Dostupné z: <https://www.statista.com/chart/15736/smart-home-market-forecast/>.
- [51] ROBINSON, S. What is Arduino? *Stack Abuse* [online]. Spojené státy: Stack Abuse [cit. 1. května 2021]. Dostupné z: <https://stackabuse.com/what-is-arduino/>.
- [52] SANTOS, R. Best ESP8266 Wi-Fi Development Board – Buying Guide 2020. *Maker Advisor* [online]. Portugalsko: Random Nerd Tutorials, Unipessoal LDA, 2020 [cit. 2. dubna 2021]. Dostupné z: <https://makeradvisor.com/best-esp8266-wi-fi-development-board/>.
- [53] SHELBY, Z., HARTKE, K. a BORMANN, C. *The Constrained Application Protocol (CoAP)* [Internet Requests for Comments]. RFC 7252. Německo: RFC Editor, June 2014. Dostupné z: <http://www.rfc-editor.org/rfc/rfc7252.txt>.
- [54] SINGAL, T. L. *Wireless communication*. Indie: Tata McGraw Hill Education Private Limited, 2010. ISBN 9780070681781.
- [55] SVOBODA, T. Chytrá domácnost s HomeKitem není dokonalá. Co jí chybí? *Applíště* [online]. Česká republika: Tomáš Svoboda, 2017 [cit. 9. listopadu 2020]. Dostupné z: <https://www.appliste.cz/homekit-dokonalost/>.
- [56] Topte s rozumem aneb jak ještě více ušetřit za topení. *A-Z ELEKTRO* [online]. Praha: GIVERSDON s.r.o. 09/10 2015, září/říjen 2015, s. 46–47. Chytré elektroinstalace. ISSN 1805-1073. Dostupné z: <http://www.azcasopis.cz/casopis/2015/9-10/>.
- [57] *Tree Extension Datasheet* [online]. Rakousko: Loxone, 2020. Dostupné z: https://www.loxone.com/wp-content/uploads/datasheets/Datasheet_TreeExtension.pdf.
- [58] TwinCAT podporuje protokoly pro výměnu dat v IoT. *Automa – časopis pro automatizační techniku*. 1. vyd. 2018, sv. 10, s. 20–21. ISSN 1210-9592.
- [59] UPTON, E., HALFACREE, G. a GONER, J. *Raspberry Pi: uživatelská příručka*. 2., aktualizované vydání. Přeložil Jakub GONER. Brno: Computer Press, 2016. ISBN 9788025148198.
- [60] VACULÍK, P. Google Asistent a čeština: přípravy jsou zřejmě v plném proudu. *SMARTmania* [online]. Česká republika: SMARTmania s.r.o., 2020 [cit. 15. února 2021]. ISSN 1801-3066. Dostupné z: <https://smartmania.cz/google-asistent-a-cestina-pripravy-jsou-zrejme-v-plnem-proudu/>.
- [61] VUJOVIĆ, V. a MAKSIMOVIĆ, M. Raspberry Pi as a Sensor Web node for home automation. *Computers & electrical engineering* [online]. Bosna a Hercegovina: Elsevier Ltd. 2015, sv. 44, s. 153 – 171. DOI: 10.1016/j.compeleceng.2015.01.019. ISSN 0045-7906. Dostupné z: <http://www.sciencedirect.com/science/article/pii/S0045790615000257>.
- [62] VÁCLAVOVIČ, J. Chytrá domácnost a HomeKit. *Medium* [online]. Spojené státy: A Medium Corporation, 2019 [cit. 5. července 2020]. Dostupné z: <https://medium.com/@jvaclavovic/chytr%C3%A1-dom%C3%A1cnost-a-homekit-e16803428604>.

- [63] VÁCLAVÍK, L. Co přinese Wi-Fi 7? Nová generace bude teoreticky skoro pětkrát rychlejší. *Živě* [online]. Česká republika: CZECH NEWS CENTER a.s., 2020 [cit. 1. května 2021]. Dostupné z: <https://www.zive.cz/clanky/co-prinese-wi-fi-7-nova-generace-bude-teoreticky-skoro-petkrat-rychlejsi/sc-3-a-206831/default.aspx>.
- [64] WILMSHURST, T. *Designing Embedded Systems with PIC Microcontrollers: Principles and Applications*. 2. vyd. Velká Británie: Elsevier Books, 2009. ISBN 9781856177504.
- [65] WOOD, L. We Proudly Present: The Miniserver Go! The Loxone Smart Home For Retrofitters! *Loxone.com* [online]. Spojené království: Loxone, 2014 [cit. 12. října 2020]. Dostupné z: <https://www.loxone.com/enen/miniserver-go/>.
- [66] ZANDL, P. *Bezdrátové sítě WiFi*. 1. vyd. Brno: Computer Press, 2003. ISBN 8072266322.

Příloha A

Obsah přiloženého paměťového média

- doc – složka obsahující text technické zprávy a jeho zdrojový kód,
- src – složka zdrojových kódů, obsahuje podsložky:
 - esp - složka se zdrojovými kódy (a knihovnami) pro koncové moduly,
 - server – složka se zdrojovými kódy pro Raspberry Pi,
 - web – složka se zdrojovými kódy pro webovou aplikaci,
- readme.md – Soubor popisující postup instalace systému.