

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra kybernetiky

BAKALÁŘSKÁ PRÁCE

Hlasové ovládání inteligentní domácnosti

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni. Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne 18.5.2018

.....
vlastnoruční podpis

Poděkování

Touto cestou bych rád poděkoval vedoucímu mé bakalářské práce Ing. Adamovi Chýlkovi za jeho cenné rady a čas, který mi během vypracování bakalářské práce věnoval.

Dále bych rád poděkoval všem, kteří se podíleli na testování daného systému a vyplnění dotazníků.

Anotace

Cílem bakalářské práce bylo vytvoření inteligentních domácích spotřebičů a následnou komunikaci mezi nimi. Dalším krokem byla tvorba uživatelského rozhraní s možností hlasového ovládní daných inteligentních prvků. Hlavním prvkem této domácnosti je server, který funguje na platformě Orange PI. Již zmíněný server funguje jako zprostředkovatel pro komunikaci mezi ním a spotřebiči v domácnosti, dále jako webový a databázový server. Domácí spotřebiče fungují na platformě Espressif s integrovaným čipem ESP 8266.

Klíčová slova:

inteligentní domácnost, hlasový dialogový systém, komunikace, server, místnost, zařízení, rozpoznávání řeči, Arduino, Espressif, broadcast, internet věcí

Abstract

The aim of this thesis was invention of smart home appliances and subsequent communication between them. Next step was creation of the user interface with an opportunity voice control of a certain elements. Main element of this household is server, which runs on platform Orange PI. This server works as a communication provider between itself and home appliances, and works as web and database server too. These home appliances run on platform Espressif with an integrated chip ESP 8266.

Key words:

intelligent household, voice dialog system, communication, server, room, device, speech recognition, Arduino, Espressif, broadcast, internet of things

Obsah

1. Úvod	3
1.1. Cíle práce	3
1.2. Vytvořený systém inteligentní domácnosti	4
1.3. Testování systému	4
2. Inteligentní domácnost	5
2.1. Úvod	5
2.2. Využití	5
2.3. Internet věcí	6
3. Hlasový dialogový systém	6
3.1. Úvod	6
3.2. Obecná architektura hlasového dialogového systému	7
3.3. Detekce řeči	7
3.4. Rozpoznávání řeči	8
3.5. Syntéza řeči	8
3.6. Konstrukce dialogového systému	8
3.6.1. Gramatika	8
4. Komunikace	9
4.1. MQTT	9
4.2. WebSocket	9
4.3. SpeechCloud	9
5. Klientská zařízení	10
5.1. Nastavení	10
5.1.1. Zapsání SSID a hesla do paměti	11
5.1.2. Restartování zařízení	11
5.1.3. Vymazání FLASH paměti	12
5.1.4. Napájení	12
5.2. MQTT připojení	13
5.2.1. Zaslání požadavku	13
5.2.2. Obdržení zprávy	14
5.2.3. Připojení	14
5.2.4. Téma home/init	15
5.2.5. Téma home/logger	15

5.2.6.	Téma home/request.....	16
5.2.7.	Téma home/state	16
5.3.	Fyzická zařízení	17
5.3.1.	Inteligentní osvětlení.....	17
5.3.2.	Inteligentní květináč.....	18
5.3.3.	Poštovní schránka.....	19
5.4.	Virtuální zařízení.....	21
5.4.1.	MAC adresa	21
5.4.2.	Žaluzie.....	21
5.4.3.	Zámek	23
6.	Server.....	25
6.1.	Zajištění IP adresy.....	25
6.2.	Odebírání MQTT zpráv	25
6.2.1.	Inicializace zařízení	25
6.2.2.	Zápis stavů	26
6.2.3.	Ovládání místností.....	27
7.	Aplikace	28
7.1.	Přihlašování	28
7.2.	Místnosti.....	29
7.3.	Zařízení	30
7.4.	Grafické ovládání.....	31
7.4.1.	Příkaz na změnu nastavení.....	32
7.4.2.	Detekování změny zařízení.....	32
7.4.3.	Vyčtení aktuálního stavu	33
7.5.	Hlasové ovládání.....	34
7.5.1.	SpeechCloud	34
7.5.2.	Konstrukce gramatiky.....	35
7.5.3.	Vypnutí/zapnutí.....	35
7.5.4.	Detekce rušného pozadí.....	35
7.5.5.	Dialog.....	36
7.5.6.	Dotaz na stav	37
7.5.7.	Příkaz	39
8.	Testování	42
9.	Závěr.....	46

1. Úvod

Bakalářská práce se zaměřuje na inteligentní spotřebiče a prvky v inteligentní domácnosti. Využitím softwaru a těchto inteligentních prvků můžeme docílit mnoha ulehčení uživatelům v dané domácnosti. K poskytnutí informací o daném objektu či prvku slouží již zmíněný software, který může být ve formě webové, desktopové či mobilní aplikace.

V dnešní době existuje mnoho firem, které se využitím automatizace domácností nebo firem, zabývají. Velký důraz na tyto inteligentní systémy je modularita a přístupnost. Tedy možnost, aby uživatel nemusel zakupovat celý set prvků, ale mohl přidat jeden do již fungující domácnosti. Dalším již zmíněným požadavkem na systém je přístupnost. Jedná se o kontrolu a ovládání inteligentních prvků mimo objekt, kde se tyto prvky nachází, tedy možnost připojit se odkudkoliv.

Možnou nadstavbou těchto systémů je hlasové ovládání, které může velice usnadnit chod domácnosti. Navíc pro zrakově postiženého uživatele, který chce systém využívat, je to velice užitečná funkce, jelikož nemusí být schopen ovládat domácnost pomocí grafického prostředí.

Důvody zavedení inteligentních systémů pro rodinné či firemní objekty můžeme shrnout v několika bodech.

- Zvýšení komfortu
- Úspora energie
- Kontrola
- Úspora času

1.1. Cíle práce

Prvním cílem bakalářské práce je prostudování problematiky automatického rozpoznávání řeči a dialogu s počítačem mluvenou řečí. Druhým cílem je analýza přístupů hlasového ovládání inteligentní domácnosti. Dále je zde realizace fyzických a virtuálních inteligentních prvků, jejich možnosti ovládání a komunikace mezi nimi. Posledním cílem je zrealizovaný systém vhodným způsobem otestovat na několika vytvořených modulech inteligentní domácnosti.

1.2. Vytvořený systém inteligentní domácnosti

Jedná se o systém, který klade důraz na modularitu. Díky tomu, že poskytuje některé své služby, je možné přidat například mobilní nebo desktopovou aplikaci, díky které budeme moci též ovládat danou domácnost. Aplikace je podle výsledků testů snadno ovladatelná a uživatelé se v ní dobře orientují. Možnosti hlasového ovládání mohou plně nahradit grafické ovládání a v několika případech je i předstihnou. Pokud uživatel používá ovládání hlasem, aplikace je schopna rozlišit zvuky v pozadí a upozornit na budoucí nesprávné rozpoznávání hlasu. V případě, kdy uživatel nechce ovládat domácnost pomocí hlasových příkazů, může rozpoznávání jednoduše vypnout pomocí tlačítka.

Vytvořená fyzická zařízení je možné připojit k bezdrátové síti, která vysílá v pásmu 2,4GHz. Pokud je zařízení v továrním nastavení, má uživatel možnost nastavit do jaké sítě chce toto zařízení připojit. Další možností inteligentních prvků je možnost převést zařízení zpět do továrního nastavení a nastavit zařízení znovu. Pokud se zařízení úspěšně připojí do sítě, je automaticky nalezen MQTT server, díky broadcast dotazům, na které daný server odpovídá. Veškerou komunikaci mezi databází a aplikací řídí serverová služba, která loguje stavy, které zařízení odešle.

Zařízení jsou rozdělena na vstupní a výstupní, přičemž u výstupních zařízení je důraz na nízkou spotřebu, která je řešena pomocí režimu spánku. Tato zařízení jsou probouzena jednou za 5 minut a odesílají naměřený stav do databáze. U vstupních zařízení je tomu naopak, počítá se s tím, že zařízení bude napájeno pomocí elektrické sítě a odesílají stav pouze při změně nastavení. Požadavek permanentního napájení je z toho důvodu, aby mohlo každé zařízení tohoto typu permanentně komunikovat a reagovat na požadavky od uživatele.

1.3. Testování systému

Systém byl testován osmi uživateli, kde každý z nich testoval již fungující domácnost. Každý uživatel dostal manuál k ovládání domácnosti, který je přiložen v přílohách tohoto dokumentu. Jejich úkol bylo ovládání domácnosti pomocí grafického systému a následně stejně otestovat domácnost pomocí hlasových příkazů. Následně byli požádáni, aby zkusili rozpoznávání hlasu s rušným pozadím. Obvykle se jednalo o zapnutou televizi nebo hudbu.

2. Inteligentní domácnost

2.1. Úvod

V tomto století patří inteligentní domácnosti mezi velké trendy, se kterou se můžete setkat čím dál tím častěji. Jedná se o funkční propojení více inteligentních prvků, které můžeme vzdáleně ovládat pomocí různých ovládačů, či mobilních aplikací. V ČR se za posledních 10 let objevilo několik firem, které se instalací těchto inteligentních domácností zabývá. Některé z nich dokonce staví i novostavby se zabudovanými funkcemi dle přání majitele. Cena takovéto instalace se odvíjí od množství, či funkce, které poskytují dané prvky pro inteligentní domácnost. Při komplexní instalaci může být cena i ve výši miliónu korun.

2.2. Využití

Inteligentní domácnost se dá popsat jako soubor sjednocených prvků, které za vás dělá a hlídá věci, které jste dříve museli dělat vy, a v případě potřeby vás upozorní nebo umožní na jednotlivé akce okamžitě reagovat. Mezi základní systémy inteligentní domácnosti patří ovládání osvětlení, zabezpečení a termoregulace. Dále záleží pouze na daném uživateli, jakou funkcionalitu od zařízení požaduje a kolik je ochoten do systému investovat. Využití inteligentních prvků se můžeme setkat u rodinných domů, jako je například ovládání rolet, světel nebo automatické zavírání vrat od garáže. Na druhou stranu se tyto inteligentní prvky hojně využívají ve firmách, jako jsou například docházkové systémy, otevírání dveří pomocí RFID čteček, či zautomatizované výrobní linky. Velký důraz je na vzdálené ovládání nebo kontrolu stavu těchto zařízení. Uživatel může například vypínat nebo zapínat osvětlení v místnosti při cestě domů, z práce nebo z druhé strany planety. Ukázka uživatelského rozhraní aplikace firmy Jablotron je na vyobrazena na obrázku 2.1. (převzato z [8]).



Obrázek 2. 1. Ukázka uživatelského rozhraní aplikace firmy Jablotron

2.3. Internet věcí

Internet of Things je znám pod označením IoT. Pod tímto označením si můžeme představit ucelený systém inteligentních prvků propojené skrz internet. Tato zařízení používají drátová i bezdrátová řešení k připojení do sítě, přičemž u bezdrátových zařízení je často kladen důraz na dlouhodobou výdrž baterie. Možných sítí, do kterých lze zařízení připojit, rychle přibývá. V České republice jsou zde například sítě LoRa nebo SIGFOX, které poskytují své služby zejména za účelem připojení těchto zařízení. Připojením zařízení však uživatelé vystavují veškerá data, která se odesílají skrz internet, možnému riziku napadení nebo odposlouchávání. [1] Jako příklad těchto zařízení můžeme uvést například měření teploty, rychlosti vzduchu, ale například i možnost sledování pohybu.

3. Hlasový dialogový systém

3.1. Úvod

Nejpříznivějším projevem komunikace mezi lidmi je zaručeně prostřednictvím mluvené řeči. Díky tomu mají hlasové dialogové systémy postupem let čím dál větší uplatnění, a i přes zatím nedokonalé rozpoznávání jsou uplatňovány v aplikacích ovládání inteligentních domácností, hlasových rozhovorech v kontaktních centrech nebo například při převodu mluvené řeči na text. Pro implementaci těchto systémů je zapotřebí experta, který se v dané problematice pohybuje. Počítá se tedy, že se expert v dané oblasti vyzná a má o ní dostatečné znalosti. Bohužel je třeba podotknout, že plnohodnotný dialog mezi počítačem a člověkem zatím není možný. Je to zejména z důvodu problémů s rozpoznáváním spontánní řeči a porozumění ze strany počítače ve smyslu klasifikovaných slov a vět. I přes tyto nedostatky jsou dialogové systémy implementovány čím dál častěji. Ve většině případů se jedná o systémy, kde je komunikace s počítačem omezena na vyhrazenou oblast tematiky, které se systém týká. Příklad systému je například objednání jídla pomocí telefonátu, kde uživatel promlouvá k počítači a ten se ho ptá na doplňující informace. Tento příklad systému dokáže rozpoznat slova, která jsou určena pro vymezenou problematiku, není tedy možné, aby počítač rozuměl problematice z kategorie zdravotnictví.

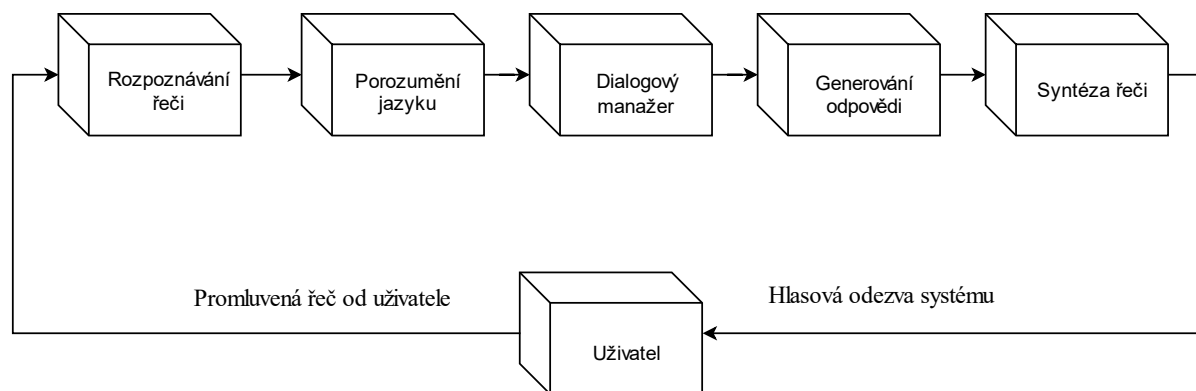
Široce využívány jsou i dílčí komponenty hlasových dialogových systémů, to jsou například moduly syntézy a rozpoznávání řeči. Všeobecné využití těchto technologií je například při automatickém přepisu mluvené řeči na text, které našlo veliké uplatnění ve zdravotnictví, kdy osoba, která tento systém využívá, je obvykle zaneprázdněna jinými

činnostmi. Další výhodou těchto systémů je možnost ovládní pro tělesně handicapované lidi. V těchto případech obvykle není možné, aby uživatel využíval grafická rozhraní nebo reagoval na zvukové projevy. Usnadněním může být například automatické titulkování pořadů nebo předčítání příchozích SMS zpráv.

Pro každý systém ovládaný hlasem, založen na expertním návrhu systému, jsou nutné dílčí procesy, jako je například detekce a rozpoznávání řeči. Dále je tu nutnost zkonstruování dialogu pomocí tzv. gramatiky a vhodné navržení vedení dialogu.

3.2. Obecná architektura hlasového dialogového systému

Na obrázku 3.1. je znázorněno blokové schéma typického hlasového dialogového systému. Toto schéma zahrnuje integraci několika modulů, které obstarávají následující funkce – rozpoznávání řeči, porozumění jazyku, generování odezvy včetně počítačové syntézy řeči. Ústředím a nejdůležitějším modulem je však dialogový manažer. Ten řídí interaktivní strategii a spojuje různé typy informace a znalostí opatřené ostatními moduly dialogového systému tak, aby úspěšně dokončil úlohu „řešenou“ dialogem.



Obrázek 3.1. Architektura hlasového dialogového systému

3.3. Detekce řeči

Rozpoznání lidské řeči je pro člověka všední záležitost a není na ní třeba mnoho úsilí. Dokážeme rozpoznat dva rozhovory, které probíhají současně nebo dokážeme potlačit vnímanou hudbu, která hraje v pozadí. Z hlediska počítačového zpracování to bohužel je složitější. Záleží na okolním ruchu, kde uživatel promlouvá, ale také pocítujeme velký rozdíl při zpracování hlasu nekvalitním mikrofonom.

3.4. Rozpoznávání řeči

I když byl na poli zpracování řečového signálu a jeho klasifikace učiněn od pionýrských dob do současnosti obrovský pokrok, přesto je zatím konstrukce zařízení, které by bylo schopno rozpoznat promluvu jakéhokoliv řečníka užívajícího libovolná slova daného jazyka, ještě poměrně vzdálenou budoucností [2]. Obtížnost rozpoznávání souvisí s prostředím, kde je daná řeč pronášena, ale i se složitostí řešené úlohy.

3.5. Syntéza řeči

Jedná se o proces, při němž se uměle vytváří řeč. V případě počítačové syntézy se řeč uměle vytváří počítačem. Umělé vytváření řeči počítačem si klade za cíl učinit komunikaci člověka s počítačem více přívětivou a stát se tak rovnocenným partnerem tradiční vizuální komunikaci. Zařízení, které proces vytváření řeči provádí, nazýváme syntetizér řeči. Syntetizér je jádrem každého systému konverze textu na řeč. Konečným cílem syntézy řeči je vytvářet řeč v takové formě a kvalitě, aby nebyla rozpoznatelná od řeči člověka. Syntetická řeč by tedy neměla působit jednotvárně, měla by znít přirozeně a její poslech by neměl unavovat ani vyžadovat zvýšenou pozornost. [2]

3.6. Konstrukce dialogového systému

Pro konstrukci systému byl zapojen expert, který v dané oblasti vypsál některé možné promluvy a slova pro ovládání inteligentní domácnosti. Takto zkonstruovaný systém se nazývá expertní systém. V době testování systému byly přidány další slova, aby systém lépe reagoval na promluvy uživatele.

3.6.1. Gramatika

Základním elementem navržení expertního systému, pro tvorbu dialogu je gramatika. Ta zahrnuje veškerá důležitá slova z vymezené oblasti, která definuje množinu promluv. Vzhledem k paměťovým a výpočetním nárokům je právě gramatika jeden z hlavních problémů, proč nelze vytvořit dialogový systém, zpracovávající informace z jiné oblasti než pro něj vymezenou. Proto se pro každý systém musí navrhnout taková gramatika, aby následně mohla být vhodně zpracována při rozhovoru s uživatelem. Nejčastější gramatikou je bezkontextová gramatika, která pro daný systém nenesí žádnou informaci.

4. Komunikace

Pro komunikaci v síti mezi jednotlivými zařízeními je používán protokol MQTT. Dále pro rozpoznávání řeči byla použita javascriptová knihovna SpeechCloud fungující na protokolu WebSocket.

4.1. MQTT

MQTT (dříve: MessageQueuing Telemetry Transport, dnes MQ Telemetry Transport) je jednoduchý a nenáročný protokol navržený firmou IBM pro předávání zpráv mezi klienty prostřednictvím centrálního bodu – brokeru. Díky této nenáročnosti a jednoduchosti je snadno implementovatelný i do zařízení s méně výkonnými procesory a poměrně rychle se rozšířil.

U protokolu MQTT probíhá přenos pomocí TCP a používá návrhový vzor publisher – subscriber. Tedy existuje jeden centrální bod (MQTT broker), který se stará o výměnu zpráv. Zprávy jsou tříděny do tzv. témat (topic) a zařízení buď publikuje v daném tématu (publish), to znamená, že posílá data brokeru, který je ukládá a distribuuje dalším zařízením, nebo je přihlášeno k odběru tématu či témat (subscribe), a broker pak všechny zprávy s daným tématem posílá do zařízení. Jedno zařízení samozřejmě může najednou být v některých tématech publisher, v jiných subscriber. [3]

4.2. WebSocket

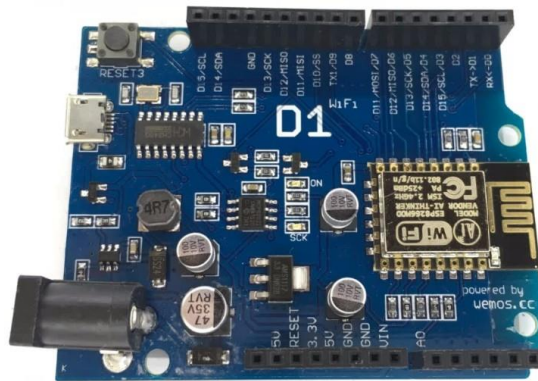
Web Sockets vychází ze starších technologií, jako AJAX (asynchronní jednorázový požadavek od klienta na server přes HTTP protokol) či Comet (dlouhodobé vyhrazené HTTP spojení) a nabízí programátorům to, co tyto technologie nedokázaly. Jednoduché rozhraní pro navázání spojení a vzájemnou výměnu zpráv mezi klientem a serverem. Na straně klienta jsou WebSockets implementovány v JavaScriptu jako třída WebSocket. Programátor může vytvořit instanci této třídy, a tím navázat spojení se serverem. [4]

4.3. SpeechCloud

Jedná se o API propojující hlavní řečové technologie jako je rozpoznávání, syntéza a porozumění řeči. Komunikace mezi klientem a serverem je pomocí protokolu WebSocket, kde uživatel vznesl požadavek pomocí některých metod a následně je mu v řádech setin vteřiny odeslána odpověď odpovídající události. [5]

5. Klientská zařízení

Veškerá zařízení, která byla navržena pro inteligentní domácnost, fungují na vývojové desce WeMos D1, která je zobrazena na obrázku 5.1. (převzato z [7]). Tento model obsahuje integrovaný čip ESP8266, díky kterému můžeme pracovat s Wi-Fi na frekvenci 2,4 GHz. Zároveň lze využít všechny ostatní piny této desky stejně jako u Arduino Uno.



Obrázek 5.1. WeMos D1

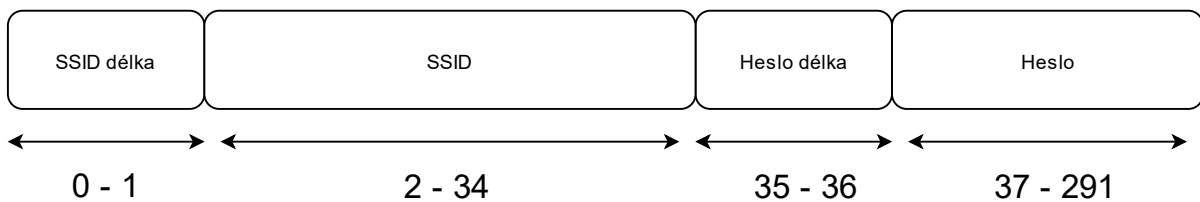
5.1. Nastavení

Pokud je zařízení ve výchozím nastavení, funguje jako webový server, který zároveň vysílá Wi-Fi síť s názvem *Nové zařízení*. Daná síť je bez hesla, tudíž se uživatel může připojit pomocí jakéhokoliv zařízení podporující komunikaci po 2,4 GHz Wi-Fi. Po úspěšném připojení dostane zařízení IP adresu v rozsahu 192.168.1.2–192.168.1.244. Pokud chce uživatel přejít do nastavení, musí do internetového vyhledávače zadat adresu *192.168.1.1*. Na webové stránce se objeví dvě pole, která slouží k nastavení zařízení. Díky tomu, že čip ESP8266 může fungovat jako vysílač i přijímač zároveň, objeví se v prvním poli seznam Wi-Fi sítí v okolí, ve kterém má uživatel možnost vybrat SSID sítě, do které chce zařízení přidat. Druhé pole slouží pro zadání hesla vybrané sítě.

5.1.1. Zapsání SSID a hesla do paměti

Po odeslání formuláře obsahující informace o připojení se informace musí zapsat do flash paměti pro případ, kdyby zařízení bylo vypnuto. Pokud by se tato data nezapisovala do flash paměti, tak by byla vždy vymazána po vypnutí/restartu zařízení.

Do flash paměti se tedy zapíše SSID a heslo sítě, do které chce uživatel připojit zařízení. Je ale nutné, aby bylo zapsáno a následně přečteno v určité struktuře. Pro zápis je tedy uvolněno maximálně 291 B a informace jsou zapsány v následném pořadí. Na pozici 0–1 je zapsána délka SSID, maximální délka této hodnoty je 32 B, tudíž je na pozici 2–34 zapsána SSID sítě. Následně je na pozici 35–36 zapsána délka zadaného hesla a zbylé pozice, tedy 37–291 jsou vyhrazeny pro zápis hesla.

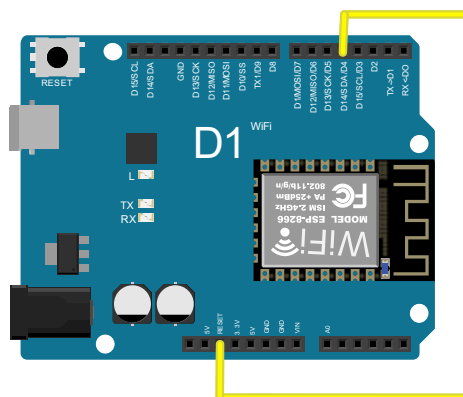


Obrázek 5.2. Struktura flash paměti

Po zapsání těchto dat do paměti se zařízení automaticky restartuje.

5.1.2. Restartování zařízení

Po zapsání dat do flash paměti je nutné zařízení restartovat. Toho docílíme pomocí zapsání hodnoty LOW na pin D4 při aktuálním zapojení.

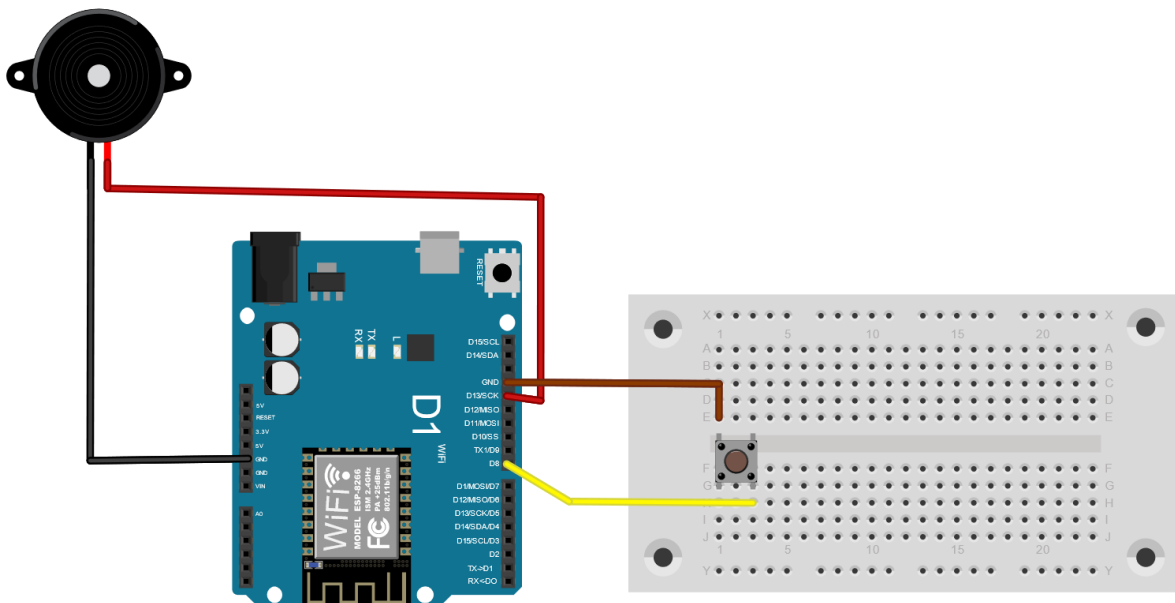


Obrázek 5.3. Zapojení pro programové restartování zařízení

Na pin je ihned po startu zařízení zapsána vždy hodnota HIGH, aby se zařízení nerestartovalo.

5.1.3. Vymazání FLASH paměti

Je možné, že uživatel zadá špatné heslo nebo název Wi-Fi sítě při prvotním nastavení zařízení. Další případ, který může nastat, je ten že se změní hierarchie sítě a bude nutno změnit nastavení zařízení. Pro tyto případy byla zřízena funkce, která vymaže všechna data z FLASH paměti. Tuto funkci může uživatel vyvolat podržením tlačítka do doby, než začne akustický bzučák vydávat zvuk. Některá zařízení jsou ve většině času v režimu spánku, tudíž je nutné, aby uživatel stiskl prvně resetovací tlačítko a poté držel tlačítko pro vymazání paměti



Obrázek 5.4. Zapojení pro vymazání flash paměti a následné upozornění

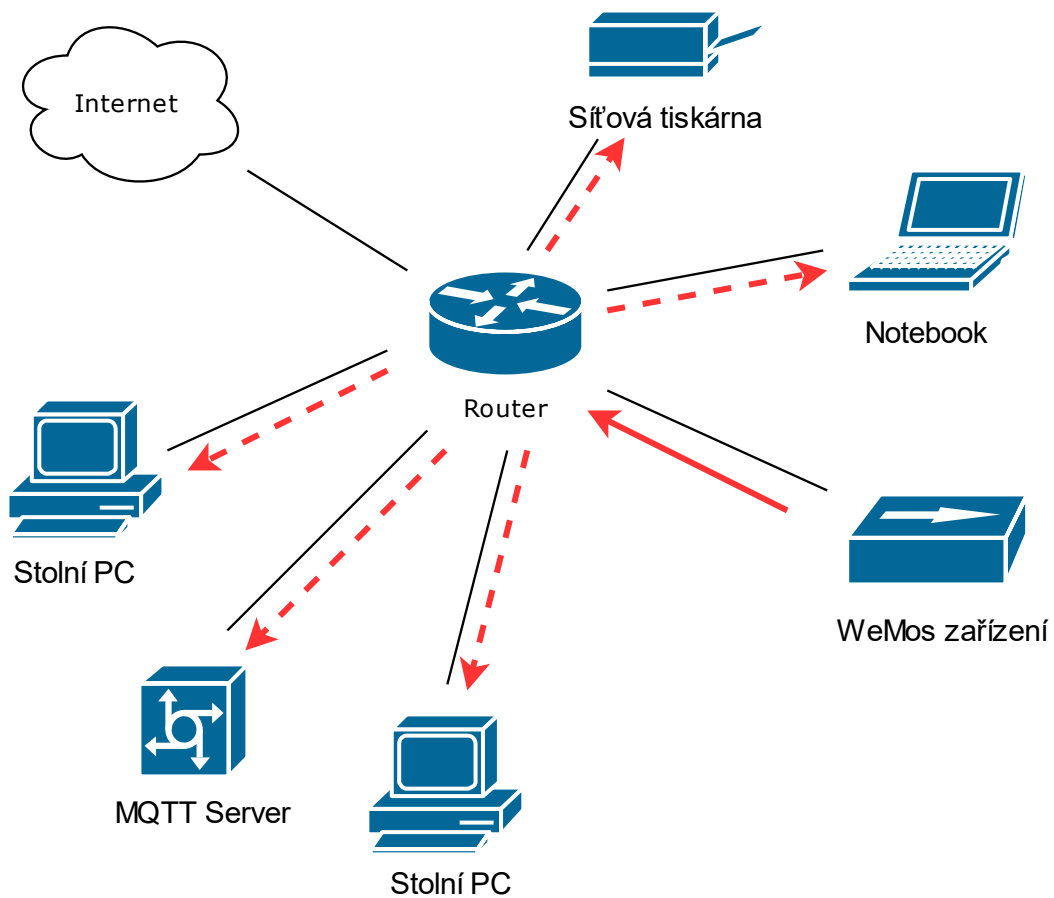
5.1.4. Napájení

Napájení jednotlivých prvků je možné pomocí dvou tužkových baterií, nebo pomocí napájení z elektrické sítě pomocí 9 až 24 VDC napájecího zdroje. Jednotlivá zařízení mají jinou spotřebu podle toho, o jaký typ zařízení se jedná. Obvykle je spotřeba závislá na tom, jestli je zařízení vstupní nebo výstupní. Dalším aspektem, který hraje ve spotřebě roli, je doba spánku zařízení. Ta se odvíjí od toho, jakou odezvu požadujeme od daného zařízení.

5.2. MQTT připojení

5.2.1. Zaslání požadavku

Po úspěšném připojení do sítě je nutné, aby se zařízení připojilo k Brokeru. Toho docílí tak, že zašle UDP broadcast paket všem zařízením v síti se zprávou "Hello! My ipaddress: " + IP adresu daného zařízení“.

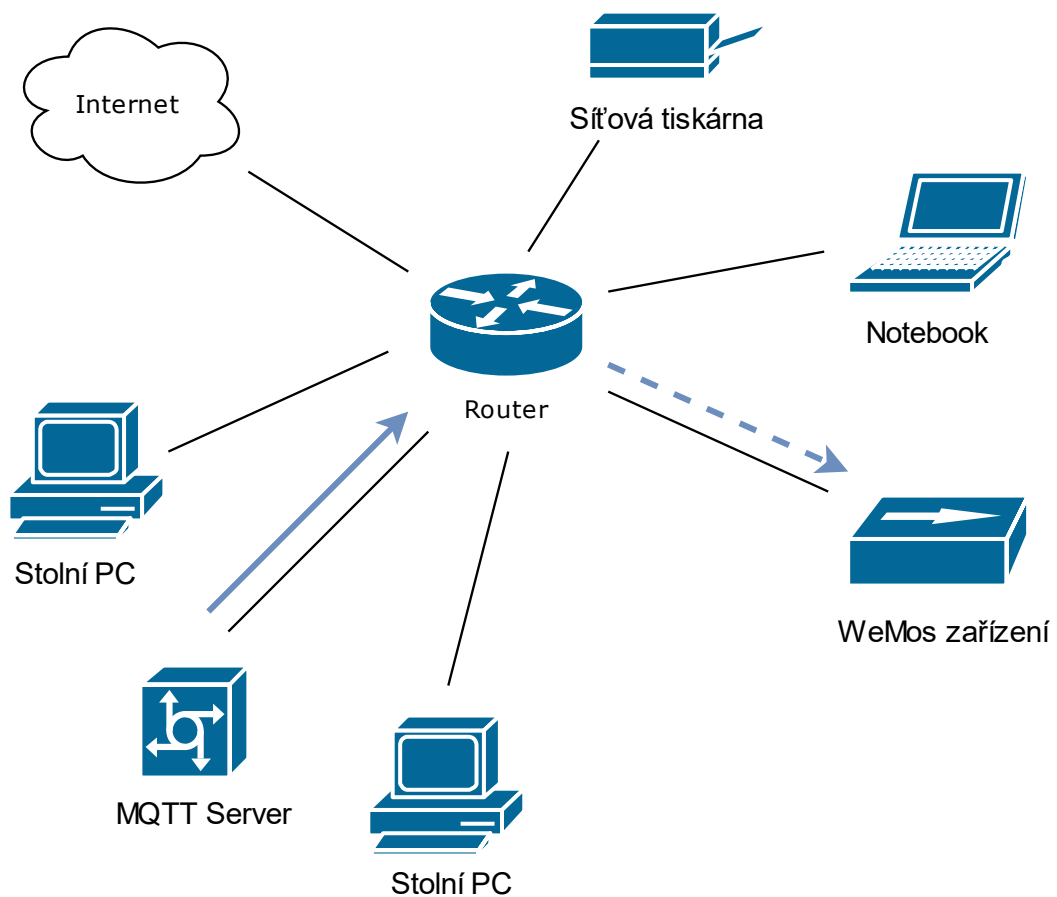


Obrázek 5.5. Broadcast zpráva od zařízení WeMos

Pokud WeMos zařízení nedostane odpověď na tento packet od serveru, tak tuto akci opakuje každých 5 vteřin. Neobdržení odpovědi od serveru může znamenat chybu v síti, nespouštěnou službu nebo jinou chybu na straně serveru.

5.2.2. Obdržení zprávy

Na serveru je spuštěn skript, který odchytává tyto packety a následně na ně odpovídá. Skript načte část zprávy, přesněji IP adresu a na danou adresu odešle UDP zprávu "Welcome! My ipaddress: " + adresu serveru.



Obrázek 5.6. Zpráva od MQTT serveru

Po obdržení zprávy zařízení ukončí pravidelné odesílání dotazovací se zprávy a pokusí se připojit na IP adresu, která mu byla zaslána. Tuto adresu si zařízení neuchovává ve flash paměti, kvůli pravděpodobnému přidělování IP adresy DHCP serverem dané sítě. Zapisování této IP adresy by mělo cenu jen tehdy, kdybychom věděli, že IP adresa MQTT serveru nebude nikdy v budoucnu změněna.

5.2.3. Připojení

Hlavním identifikátorem v komunikaci je MAC adresa zařízení. Jedná se o jednoznačný identifikátor síťového rozhraní, skládající se ze šesti dvojic hexadecimálních čísel. Tato

adresa je přidělena výrobcem daného zařízení, který by měl zaručit její světovou jedinečnost. Všechna WeMos zařízení se tedy připojují k MQTT serveru pod tímto identifikátorem.

Následně je důležité, aby se zařízení přihlásilo k tématům, pomocí kterých chce v síti komunikovat. Všechna možná témata, na kterých mohou zařízení komunikovat, jsou popsána v následující tabulce 5.1. Koncová označení *out/in* znamenají, zda zprávy odeslány ze zařízení/do zařízení.

<i>Topic</i>	
home/init/MAC_ADRESA/out	Pro inicializaci zařízení v síti
home/state/MAC_ADRESA/out	Pro odeslání aktuálního stavu zařízení
home/logger/MAC_ADRESA/out	Pro zasilání dat, zapisující se do databáze
home/request/MAC_ADRESA/in	Pro zaslání požadavku změny nastavení

Tabulka 5.1. Tabulka témat

Komunikace na těchto topicích se odvíjí na tom, o jaké zařízení se jedná. Dále je tu téma *home/room/in*, na kterém odebírá zpráva pouze serverová služba. Toto téma je k ovládání vstupních zařízení v celé místnosti.

5.2.4. Téma home/init

Každé zařízení komunikuje na tomto tématu vždy po připojení k MQTT brokeru. Zpráva pochází ze zařízení a je odesílána za účelem inicializace zařízení v síti pomocí serverové služby. Obsah této zprávy je typ a MAC adresa zařízení. Příklad zprávy je zobrazen v kódu 5.1. Popis inicializace je popsán níže v kapitole 6.2.1.

```
{“deviceType“:“Light“, “init“:“60-01-94-70-20-92“}
```

Kód 5.1. Struktura init zprávy

5.2.5. Téma home/logger

Obsah zprávy je totožný se zprávou, odesílající se na tématu *home/logger/MAC_ADRESA/out*. Zařízení odesílá vždy zprávu v okamžik, kdy se změní její poslední stavová hodnota na opačnou. Například pokud světlo bylo prvně zhasnuto a následně rozsvíceno, odešle se zpráva na tomto tématu. Další případ pro odeslání této zprávy je vždy po probuzení výstupního zařízení z režimu spánku. Děje se tomu tak

například u zařízení květináč. Logování těchto zpráv je za účelem zapisování těchto dat do serverové databáze pro vyčítání historie zařízení. Konkrétnější popis logování stavů je popsán v kapitole 6.2.2.

5.2.6. Téma `home/request`

Zprávy na tomto tématu jsou odesílány za účelem změny nastavení WeMos zařízení. Tyto zprávy jsou odesílány pomocí webové aplikace a obsahují pouze hodnotu, na kterou se má změnit nastavení cílového zařízení. Struktura této zprávy je v kódu 5.2. Pokud je příchozí hodnota odlišná od aktuální, tak jí zařízení změni na příchozí a zároveň odešle zprávu na tématu `home/logger/MAC_ADRESA/out`. Může však nastat případ, kdy je požadovaný stav stejný, jako aktuální. V tomto případě zařízení odesílá stav na tématu `home/state/MAC_ADRESA/out`. Samozřejmě na tématu `home/MAC_ADRESA/request` komunikují jen taková zařízení, u kterých je možné změnit aktuální nastavení. Jedná se například o inteligentní osvětlení nebo rolety. Zařízení, která jsou pouze výstupní, na tomto tématu nekomunikují.

```
{“state“: 1}
```

Kód 5.2. Struktura request zprávy

5.2.7. Téma `home/state`

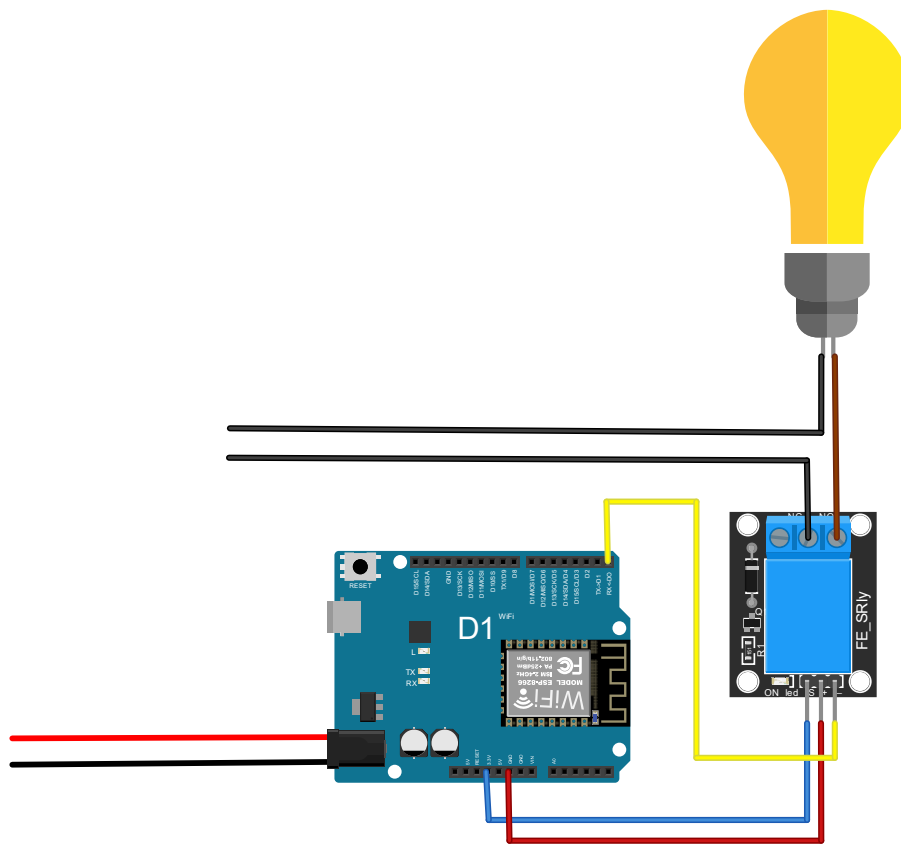
Je možné, že se poslední logovaný stav neuloží do databáze. Může se tak stát při špatné komunikaci nebo při výpadku služby na serveru. V tomto případě se webové aplikaci předloží špatná hodnota a zobrazuje se zařízení například jako zapnuto, ale je vypnuto. Tudíž pokud uživatel bude chtít zaslat požadavek, tak bude posílat hodnotu, která je již v cílovém zařízení zaevidována. Jelikož se následně nic nezmění, může to vypadat, že zařízení nereaguje. Proto se odešle hodnota s aktuálním stavem na tomto tématu a následně je zpracována webovou aplikací.

5.3. Fyzická zařízení

Pro inteligentní domácnost byly vytvořeny celkem 3 fyzická zařízení fungující na vývojové desce WeMos D1, z nichž každé plní jinou funkci v domácnosti.

5.3.1. Inteligentní osvětlení

Jedná se o vstupní zařízení, pomocí kterého lze rozsvěcet nebo zhasínat žárovku, připojenou do elektrické sítě. Součástí schématu zapojení zařízení je spínací relé. Toto schéma je vyobrazeno na obrázku 5.7.



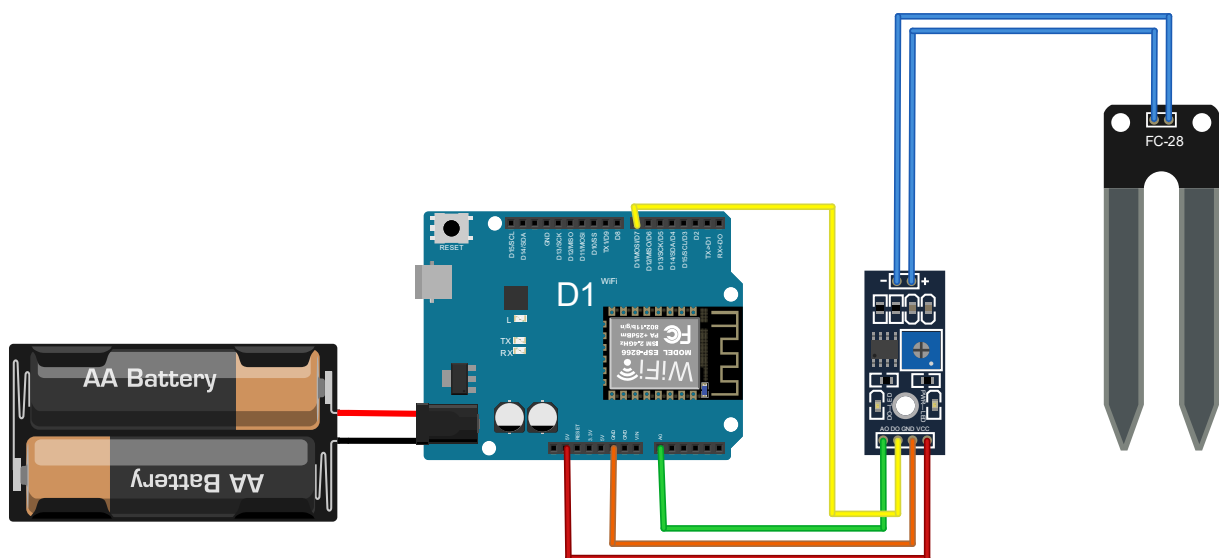
Obrázek 5.7. Schéma zapojení inteligentního osvětlení

Jelikož od tohoto zařízení požadujeme rychlou odezvu, zařízení nepřechází do režimu spánku. Je tedy vhodné, aby toto zařízení bylo napájeno stejně jako žárovka z elektrické sítě z důvodu většího odběru energie. Pokud není umožněno připojit napájení ze sítě, je možné zařízení taktéž napájet pomocí baterií. Je však nutné, aby vždy bylo dostatečné napájení pro spínání relé modulu. Jako spínací relé byl použit model JQC-3FF, na jehož vstupní port COM bylo přivedeno napájení pro žárovku a spínací pin NO byl propojen s žárovkou přímo.

Defaultní hodnota tohoto zařízení je vypnuto, tedy v programové logice hodnota *enabled* nabývá hodnoty *false*. Je tomu tak proto, aby se například žárovka nerozsvítila v nevhodnou chvíli, jako je například nečekaný restart po výpadku elektrického proudu. Pokud však chce uživatel rozsvítit, musí pomocí webové aplikace poslat požadavek na změnu. Po přijetí požadavku zapíše hodnotu z příchozí zprávy do hodnoty *enabled* a v dalším kroku se porovná, zda je stejná nebo rozdílná od předchozí. Pokud jsou tyto hodnoty rozdílné, zapíše se na pin D0 hodnota, která byla obsažena ve zprávě. To způsobí sepnutí nebo rozepnutí relé a tím i rozsvícení nebo zhasnutí žárovky, která je součástí schématu. Pokud žárovka není připojena k napětí, není samozřejmě schopna zářit. Zařízení tím ale není nijak ovlivněno, ke změně stavu relé dojde i bez napájení žárovky.

5.3.2. Inteligentní květináč

Ve své podstatě se jedná o zařízení, snímající vlhkost půdy pomocí půdního vlhkoměru a komparátoru typu LM393. Schéma tohoto zařízení je zobrazeno na obrázku 5.8. S tímto komparátorem se často setkáváme při použití s Arduino deskami, kde jeho hlavní účel je zpracování vstupního signálu. Jedná se tedy o elektrický obvod, jehož logická hodnota odpovídá porovnáním napětí na jeho vstupech. Je-li hodnota napětí na kladném vstupu větší než na záporném, na výstupu je generována logická 1. Pokud je tomu naopak, je generována opačná hodnota, tedy logická 0.



Obrázek 5.8. Schéma zapojení inteligentního květináče

Tuto logickou hodnotu lze číst z pinu D0, kde práh sepnutí mezi logickou 1 a 0 lze nastavit pomocí potenciometru, který je integrován na komparátoru. Pokud však chceme

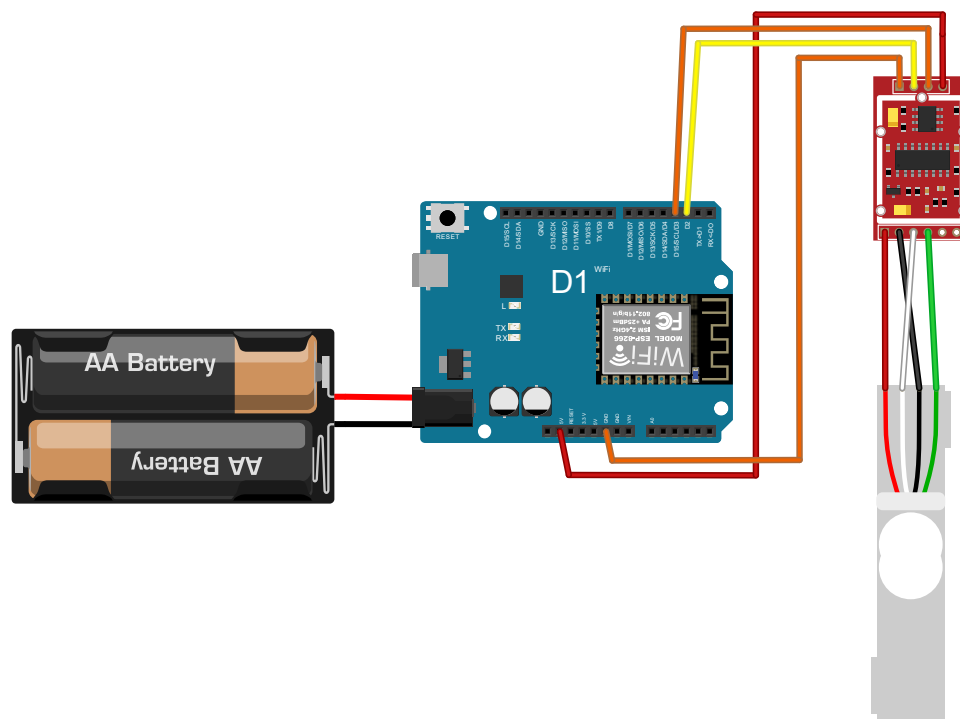
vyčítat analogovou hodnotu, můžeme jí vyčíst z pinu A0. V případě takto vyčítané hodnoty by bylo na programátorovi, aby určil daný práh, definující sucho nebo vlhko. Pro vyčítání hodnoty byl využit pin D0 z důvodu možného nastavení prahu pomocí potenciometru uživatelem.

U tohoto zařízení není nutné, aby byla rychlá odezva na změnu vlhkosti půdy. Zařízení tedy může být probuzeno například jednou za pět minut, aby zkontrolovalo daný stav a následně stav odeslalo.

Instalace zařízení je jednoduchá, stačí pouze vložit vlhkoměr do substrátu v květináči a následně zřídit napájení WeMos desce.

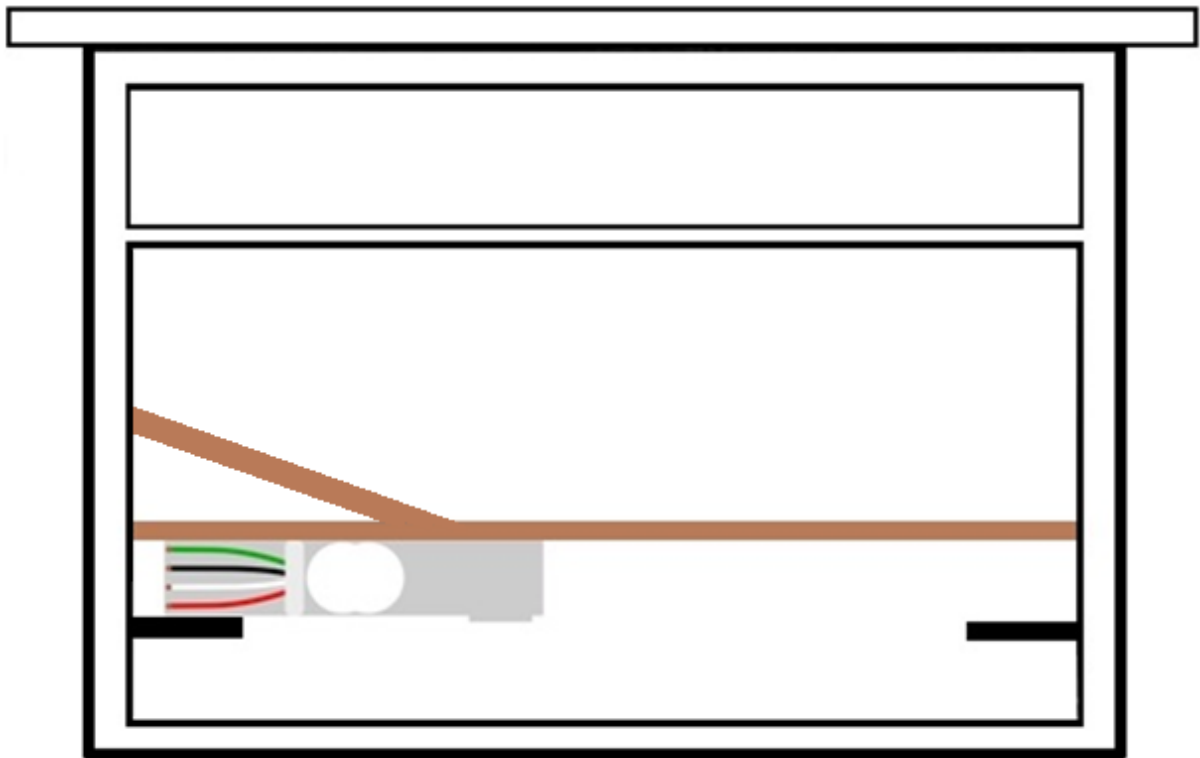
5.3.3. Poštovní schránka

Princip tohoto zařízení spočívá v detekci váhy, dopadající na hmotnostní senzor. Celý modul se tedy skládá z váhového senzoru a z A/D převodníku HX711. Jedná se o převodník, který je ve velké míře používán v oblasti měření signálu. Obsahuje dva analogové vstupy a vyznačuje se vysokou přesností měření.



Obrázek 5.9. Schéma zapojení poštovní schránky

Oproti předchozím zařízením se jedná o komplikovanější instalaci. Rozměry váhového senzoru jsou přibližně 12,7 mm x 80 mm, tudíž je zapotřebí, aby předmět, který se vhodí do poštovní schránky, dopadl na váhové čidlo. Toho docílíme například tím, že přimontujeme váhové čidlo na plechový výstupek, který je součástí některých poštovních schránek. Na dané čidlo dále přišroubujeme rovnou dřevěnou desku nebo plechový plát, to nám zajistí, aby byla pokryta celá plocha vnitřku schránky, na které můžeme detekovat váhu. Zároveň musíme zajistit, aby předmět nespadol mimo oblast, kterou je čidlo schopno zachytit. Tato instalace je popsána na obrázku 5.10.



Obrázek 5.10. Návrh konstrukce poštovní schránky

Pod touto deskou nebo plátem následně vznikne místo pro WeMos destičku s napájením. Toto zařízení je pouze výstupní a přechází do režimu spánku. Bylo tedy nutné, aby si pamatovalo váhu destičky, která je k čidlu připevněna. V opačném případě by vždy destičku zařízení bralo například jako pohled nebo dopis. Při nastavení zařízení si do paměti zároveň zapíše i váhu této destičky, kterou vždy při zpracování váhy odečte.

5.4. Virtuální zařízení

Pro vytvoření rozsáhlejší možnosti ovládní domácnosti byly vytvořeny virtuální zařízení v programovém jazyce Python. Jedná se o dvě zařízení, která evidují stav zapnuto/vypnuto. Konkrétně se jedná o zařízení žaluzie a zámek. Skripty jsou určeny pro spuštění z příkazové řádky na serveru nebo na jiných zařízeních připojených do sítě.

5.4.1. MAC adresa

Pro identifikaci zařízení bylo nutné do programu integrovat funkci, která vytvoří textovou podobu MAC adresy. Ta funguje na principu vytvoření náhodného čísla v intervalu 0 až 255, které se do výsledného textového řetězce zapíše v hexadecimálním tvaru.

Zároveň je možné, aby zařízení pracovalo na předem definované MAC adrese. Stačí pouze vložit danou adresu v hexadecimálním tvaru jako argument při spuštění z příkazové řádky systému. Program následně zjistí tvar textového řetězce a jednotlivá čísla v hexadecimálním tvaru se pokusí převést do dekadického čísla. Pokud jsou tyto kroky úspěšné, aplikaci je přiřazena daná MAC adresa. V opačném případě program pokračuje vygenerováním MAC adresy, který byl popsán v předchozím kroku.

5.4.2. Žaluzie

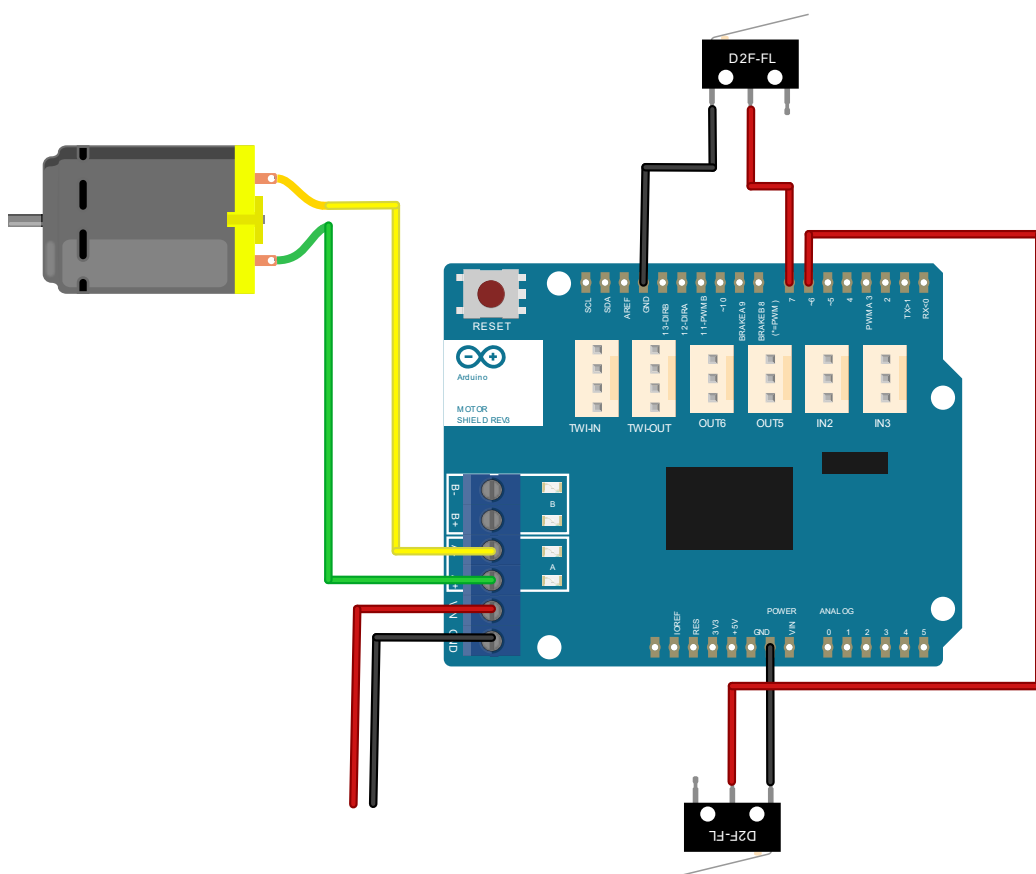
Chování tohoto zařízení reprezentuje fyzické zařízení na ovládní žaluzií. Konkrétně by se jednalo například o malý motůrek, ovládný řídicí deskou. Stav zařízení by byl generován na základě sepnutí jednoho ze spínačů.



Obrázek 5.11. Ovládní žaluzií pomocí kliky

Na obrázku 5.11. (převzato z [6]) je schéma ovládní žaluzií, pro které by bylo možné navrhnout jiné řešení, jak ho ovládat. Klika, která je určena k ovládní, by byla nahrazena

například stejnosměrným motorem. Díky tomu, že by byl použit typ desky WeMos D1_u, bylo možné použít některý z Arduino motor shieldů. Jedná se o periférii, díky které je možno ovládat stejnosměrné nebo krokové motory. V tomto případě konkrétně motor pro rolování s žaluzií. Dále by součástí musely být dojezdové spínače, které by definovaly stav zařízení. Tedy pokud by svojí váhou sepnuly žaluzie dolní spínač, motor by neměl povoleno rolovat žaluzie ve směru dolů a zařízení by vyslalo na tématu *home/logger/MAC_ADRESA/out* hodnotu 0, tedy *zahrnuto*. V opačném případě, kdy by se žaluzie vyhrnovala, byla by při sepnutí horního spínače odeslána hodnota 1. Možné zapojení tohoto zařízení je na obrázku 5.12.



Obrázek 5.12. Schéma zapojení ovládání žaluzií

5.4.3. Zámek

Realizace tohoto zařízení by byla finančně náročná, tudíž je toto zařízení v systému reprezentováno pouze pomocí skriptu. Pro zkonstruování by byl potřeba elektronický zámek, jehož cena se odvíjí od kvality. V ceně hraje roli například odolnost vůči vylomení západky apod.



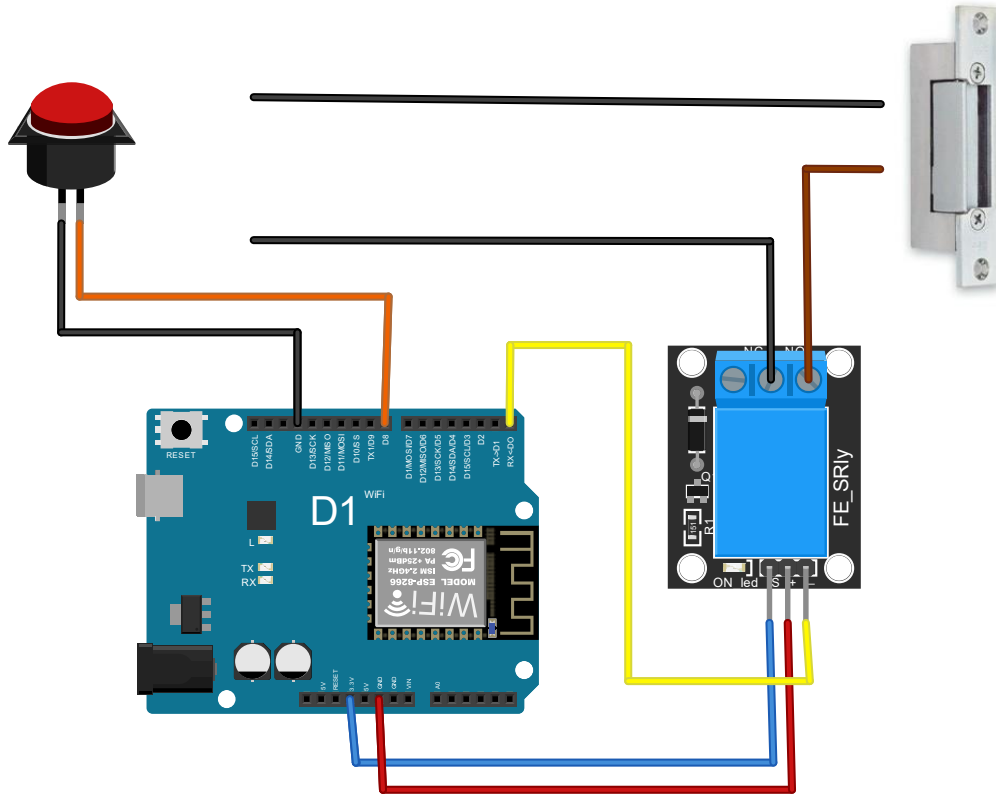
Obrázek 5.13. Elektronický zámek

Pokud by se zařízení realizovalo, mohlo by být použit elektronický zámek od firmy *ASSA ABLOY Czech & Slovakia s. r. o.* Konkrétně model *FAB KLASIK 511MB*. Tento model je v poloze otevřeno pouze po dobu trvání napěťového impulzu, tedy pokud není připojen k elektrické síti, jsou dveře, kde je tento model nainstalován, uzavřeny. Bylo by tedy možné realizovat toto zařízení pomocí kombinace tohoto zámku, obyčejného spínacího relé a *Espressif* desky *WeMos D1*.

Zařízení by dále obsahovalo tlačítko, které by sloužilo k okamžitému otevření dveří v případě, pokud by bylo zavřeno. Toto tlačítko by bylo umístěno z vnitřní strany dveří. Rozšířením tohoto zařízení by byla *RFID* čtečka, která by odemykala dveře na základě přístupových karet. Schéma tohoto zařízení je na obrázku 5.14.

Jednou z výhod tohoto modelu zámku je bezpečnost i při výpadku elektrického proudu. Tedy jinak řečeno by byl objekt zabezpečen i při výpadku elektřiny. Bohužel tato vlastnost se v některých případech může stát ohromnou nevýhodou. Například při požáru objektu a zároveň výpadku elektrického proudu jsou dveře uzamčeny a lidem uvnitř objektu není umožněn únik z budovy. V tomto případě by pro všechna tato zařízení musela zřídit záložní

elektrická síť, která by vyslala trvalý impuls všem zařízením tohoto typu v době požáru. Samozřejmě je možnost zaměnit tento model zámku za jiný, který se bude chovat opačně. Zámek by v tomto případě byl odemčen vždy, když na něj není přivedeno napětí.



Obrázek 5.14. Schéma zapojení zámku

6. Server

Server funguje jako zprostředkovatel komunikace mezi ním a jednotlivými zařízeními v síti. Tedy pokud se chce jakékoliv zařízení připojit do sítě, s úmyslem komunikovat, tak mu není bez nastaveného serveru komunikace umožněna. Je nutné, aby na daném serveru fungovaly tzv. služby, které budou umožňovat komunikaci mezi jednotlivými prvky. Veškeré služby jsou napsány v programovacím jazyce Python.

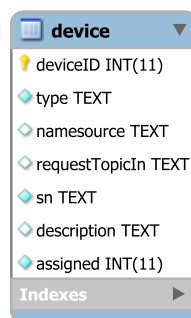
6.1. Zajištění IP adresy

Jak již bylo v předchozí kapitole 5.2. řečeno, zařízení se po připojení do sítě musí dozvědět, zda je v síti přítomný server, jehož funkce jsou určeny pro dané zařízení. Klientské zařízení pošle do sítě požadavek, který server zpracuje a následně odpoví zprávou, ve které je obsažena potřebná IP adresa.

6.2. Odebírání MQTT zpráv

6.2.1. Inicializace zařízení

Pokud zařízení pošle zprávu na tématu pro inicializaci zařízení, musí obsahovat položky definující typ zařízení a jeho MAC adresu. Následně je zjištěno, jestli je již toto zařízení zaevidováno v databázi, či ne. Pokud v databázi toto zařízení není evidováno, služba zapíše MAC adresu, typ zařízení, a pokud se jedná o vstupní zařízení, zapíše se téma, na kterém zařízení poslouchá. Tyto údaje se zapisují do tabulky *device*, jejíž kompletní struktura je na obrázku 6.1.



device
deviceID INT(11)
type TEXT
namesource TEXT
requestTopicIn TEXT
sn TEXT
description TEXT
assigned INT(11)
Indexes

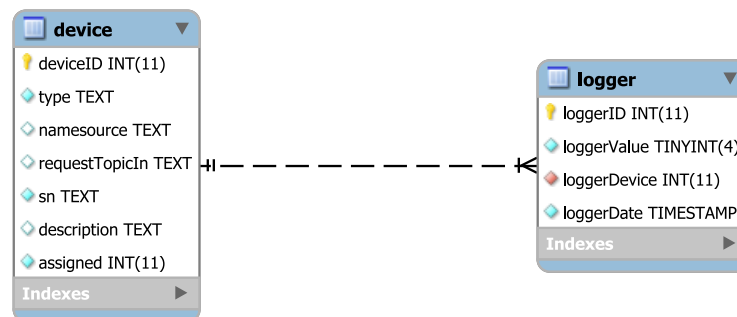
Obrázek 6.1. Tabulka pro zápis nových zařízení

Dále se v tabulce nachází hodnota *assigned*, která značí, do jaké virtuální místnosti je zařízení přiřazeno. Jedná se o číselnou položku a zapisují se zde primární identifikátory

místností, které jsou vytvořeny. Hodnoty těchto identifikátorů jsou v případě přiřazení zařízení k místnosti nezáporné. Výchozí hodnota této položky je -1, tedy nepřirazené. Zbylé hodnoty se vyplňují při registraci zařízení.

6.2.2. Zápis stavů

Každé zařízení po probuzení z režimu spánku odešle svůj stav na tématu `home/logger/MAC_ADRESA/out`. Ten musí být zachycen a následná hodnota se musí přiřadit k zaevidovanému zařízení. Pokud není v síti zaregistrované, není možné tuto zprávu zpracovat. Dále je nutné podotknout, že po vymazání zařízení z databáze zanikají všechny jeho uložené stavy.

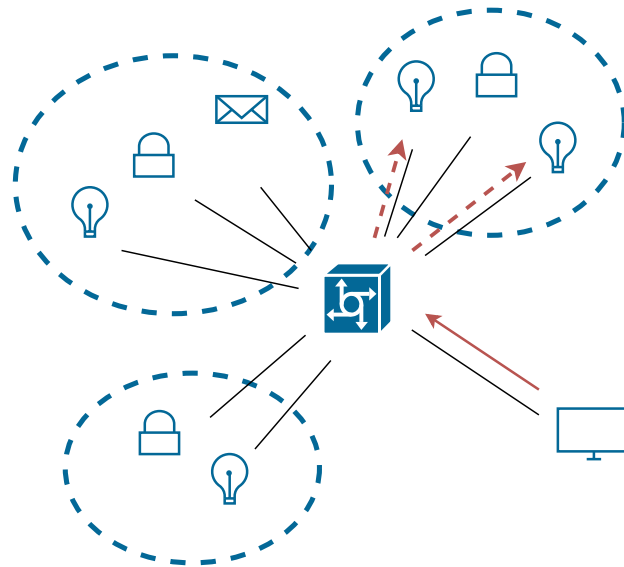


Obrázek 6.2. Struktura tabulky pro zápis stavů

Zpráva obsahuje stav a MAC adresu zařízení. Je tedy prvně nalezen identifikátor tohoto zařízení podle MAC adresy a následně je zapsána hodnota spolu s tímto identifikátorem a časem příchozí zprávy do logovací tabulky.

6.2.3. Ovládání místností

Díky této službě lze ovládat veškerá zařízení jednotlivých typů uvnitř virtuální místnosti. Pokud přijde na tématu *home/room/in* zpráva obsahující název místnosti, typ zařízení a hodnotu stavu, daná služba odešle požadavek na změnu stavu všem zařízením daného typu obsaženým ve zprávě. Na obrázku 6.4. můžeme vidět případ, když byl zaslán požadavek pro změnu stavu světel v jedné z místností.



Obrázek. 6.4. Změna stavu světel v celé místnosti

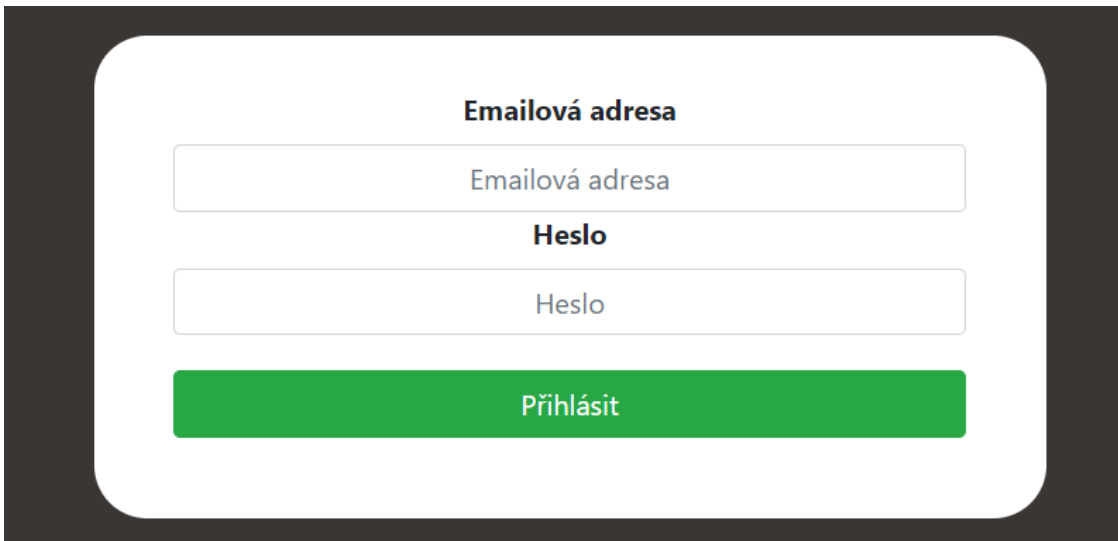
7. Aplikace

Uživatelské rozhraní, díky kterému lze ovládat jednotlivé prvky v domácnosti, funguje na principu webové aplikace. Je tedy možné, pokud je server, na kterém je webová aplikace nainstalována, správně nastavený, spustit danou aplikaci odkudkoliv na světě. Výhodou u webových aplikací je přenositelnost. Jedná se o vlastnost aplikace, která není závislá na operačním systému klientského zařízení, pomocí kterého se uživatel připojuje.

Aplikace nabízí mnohé funkce po přihlašování a vytváření jednotlivých uživatelů, ovládání prvků v domácnosti nebo vytváření virtuálních místností, do kterých lze jednotlivá zařízení později přidat. Dále je tu grafické rozhraní a dialogový systém, díky nimž lze jednotlivé prvky inteligentní domácnosti ovládat.

7.1. Přihlašování

Veškeré funkce, jako je ovládání domácnosti pomocí grafického rozhraní nebo pomocí dialogu, nejsou zpřístupněny, pokud uživatel není do systému přihlášen. Přihlášení do aplikace je možné pomocí přihlašovacího formuláře, který je zobrazen na obrázku 7.1.



The image shows a login form with a white background and rounded corners, set against a dark grey background. At the top, the text "Emailová adresa" is centered. Below it is a white input field with a light grey border containing the placeholder text "Emailová adresa". Underneath the first field, the text "Heslo" is centered. Below that is another white input field with a light grey border containing the placeholder text "Heslo". At the bottom of the form is a prominent green button with the white text "Přihlásit" centered on it.

Obrázek 7.1. Přihlašovací okno

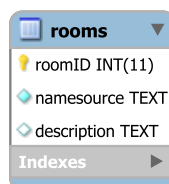
Při prvotním nastavení webové aplikace je možné přihlášení pod administrátorským účtem s názvem *root*, který je dostupný pod heslem *1234*. Ten je vždy součástí webové aplikace při výchozím nastavení z důvodu, že vytváření dalších účtů je možné pouze po přihlášení do systému. Pokud by se chtěl tedy někdo přihlásit do systému, musí požádat správce nebo kteréhokoliv uživatele, který má umožněn přístup. Při vytváření účtů se

zadává email a heslo, pod kterými se uživatel bude přihlašovat. Následně je uživateli umožněno mazat účty. Je zde ale ošetřeno, aby alespoň jeden v systému zůstal.

7.2. Místnosti

Jedna z funkcí, která je integrovaná v systému, je vytváření virtuálních místností. Ty by měly reprezentovat skupiny, do kterých lze jednotlivé prvky inteligentní domácnosti přiřadit a zároveň by měly reprezentovat jednotlivé místnosti v objektu, kde je systém nainstalován.

Název musí být jednoslovný a odlišný od názvu ostatních místností a zařízení. Odlišný název každé místnosti musí být z důvodu reprezentace těchto názvů v hlasovém rozpoznávání. Kdyby tomu tak nebylo, systém by nevěděl, na jakou místnost nebo zařízení se uživatel dotazuje, jelikož by mělo více prvků stejné názvy. Podmínka jednoslovného názvu je kvůli rozložení příchozí věty od hlasového rozpoznávače na jednotlivá slova, tudíž by systém nerozpoznal, o jaký název zařízení nebo místnosti se jedná. Každá místnost se zapisuje do tabulky *rooms*, jejíž struktura je znázorněna na obrázku 7.2.



rooms
roomID INT(11)
namesource TEXT
description TEXT

Indexes

Obrázek 7.2. Struktura tabulky *rooms*

Uživatel má možnost vytvářet místnosti pomocí formuláře, který je zobrazen na obrázku 7.3. Tento formulář se stará o kontrolu názvů jako je duplicita nebo několika slovný název.



Formulář pro vytvoření místnosti

Název

Název místnosti musí být jedinečný a jednoslovný.



Popis

Vytvořit

Obrázek 7.3. Formulář pro vytvoření místnosti

Následně po výběru již vytvořené místnosti se uživateli ukáže seznam všech zařízení, které jsou k ní přiřazeny. Vzhled této stránky je zobrazen na obrázku 7.4. Součástí výpisu je poslední stav jednotlivých zařízení v podobě přiřazené ikonky a tlačítko Informace pro možnost přejít na stránku, týkající se konkrétního zařízení. Další informací je počet jednotlivých prvků v místnosti v dolní části obrazovky.

Výpis zařízení v místnosti

Zařízení	Typ	
	Trezor	Informace
	Roletka	Informace

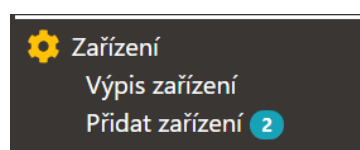
Podrobné informace o místnosti

Světlo	0x
Květináč	0x
Poštovní schránka	0x
Zámek	1x
Rolety	1x

Obrázek 7.4. Výpis informací o místnosti

7.3. Zařízení

Pokud byla úspěšná registrace zařízení do systému pomocí serverové služby, je pro zpřístupnění ostatních funkcí nutné, aby uživatel dané zařízení pojmenoval a přiřadil do místnosti. V případě, kdy bylo přidáno zařízení a je nutné, aby mu byl přiřazen název a popis, je zobrazen počet těchto zařízení v postranním panelu. Tento případ je zobrazen na obrázku 7.5.



Obrázek 7.5. Událost evidující možnost přidat 2 zařízení

Název tohoto zařízení musí být též odlišný od ostatních názvů zařízení a místností ze stejných důvodů, jako je u místností. Následně je možné vymazání a opětovné přidání zařízení po registraci serverovou službou. Spolu s výpisem zařízení podle místností je možné vypsát kompletní seznam zařízení.

7.4. Grafické ovládání

Veškerá zařízení lze ovládat pomocí jednoduchých tlačítek, která jasně definují, v jakém stavu se aktuálně zařízení nachází. Pokud zařízení komunikuje na tématu `home/request`, tak je tlačítku přiřazena funkce pro posílání požadavku na změnu stavu. Odesílá se vždy opačná hodnota, která je evidována v systému na základě barvy řídicího tlačítka. Tedy pokud systém zachytí zprávu o změně nastavení, změní barvu ikonky příslušného zařízení. Pokud uživatel zašle dotaz, na základě barvy pozadí se odešle zpráva.



Obrázek 7.6. Změna stavu tlačítka

Jednotlivá tlačítka mají v hodnotě identifikátoru MAC adresu zařízení, kterému je tlačítko přiřazeno. Dále pokud se jedná o vstupní zařízení je přidělena hodnota `name`, obsahující téma, na kterém zařízení přijímá požadavky na změnu nastavení. Důležitá část HTML struktury je vyznačena v kódu 7.1.

```
<div id="7d-44-37-2e-12" name="home/request/7d-44-37-2e-12/in">
  <buttonclass="btn-yellow">
    <svgclass="icon"></svg>
  </button>
</div>
```

Kód 7.1. Struktura HTML kódu tlačítka

7.4.1. Příkaz na změnu nastavení

Díky tomu, v jaké struktuře jsou generovány tlačítka, je možné díky knihovně jQuery a EclipsePaho Javascript Client odesílat požadavky na zařízení skrz MQTT broker. Po stisknutí tlačítka se vezme barva pozadí, reprezentující uchovaný stav zařízení v aplikaci a pošle se požadavek, který obsahuje opačnou hodnotu. Celá metoda po stisknutí tlačítka je uvedena v kódu 7.2.

```
$(document).on('click', '.device-info-button', function ()
{
  if($mqttConnected&& $topic !== "")
  {
    $topic = $(this).attr('name');
    $color = $(this).find( "button" ).css('backgroundColor');
    $state = 0;
    $color == "rgb(255, 255, 255)" || $color == "rgb(220, 53, 69)"? $state = 1: $state = 0;
    var message = newPaho.MQTT.Message('{"state":' + $state + '}');
    message.destinationName = $topic;
    client.send(message);
  }
}
```

Kód 7.2. Metoda po stisknutí tlačítka

V kódu 7.2. je prvně ošetřeno, zda je klient připojen k MQTT serveru. V případě, že je, tak je načteno téma, na kterém zařízení poslouchá z atributu *name* a následně je dle barvy tlačítka vygenerován požadovaný stav, který se poté odešle na vyčteném tématu.

7.4.2. Detekování změny zařízení

Zařízení po změně nastavení vždy odešle zprávu o aktuálním stavu na tématu *home/logger/MAC_ADRESA/out* a tento stav je pomocí serverové služby zapsán do databáze. Byla tu možnost vyčítat tyto hodnoty z databáze pomocí časovače, ale kvůli okamžité interakci byla zřízena funkce, která též jako serverová služba reaguje na zprávy v síti. Pokud je nějaká zpráva zachycena a téma této zprávy obsahuje část *out*, je zpracována a výsledný stav se promítne do tlačítka, které je zařízení přiřazeno. Toto tlačítko je vyhledáno podle MAC adresy, která je součástí zprávy. Struktura metody pro přijímání zpráv je znázorněna v kódu 7.3.

```

var obj = jQuery.parseJSON(message.payloadString);
$lastPart = message.destinationName.split('/').pop();
if( $lastPart == "out" || message.destinationName.includes("logger"))
{
    $destinationID = "#" + message.destinationName.replace("/out", "").split('/').pop();
    if($destinationID )
    {
        if ( obj.state == 1)
        {
            $($destinationID).children(".btn").css("background-color", "#ffc107");
        }
        else
        {
            $($destinationID).children(".btn").css("background-color", "#dc3545");
        }
    }
}
}

```

Kód 7.3. Metoda příchozí zprávy na straně aplikace

První krok v kódu 7.3. je načtení zprávy a zjištění, zda poslední část tématu obsahuje položku *out* a zároveň obsahuje část *logger*. Pokud jsou tyto podmínky splněny, je podle příchozího stavu zařízení vygenerováno pozadí tlačítka, jehož nadřazený prvek má identifikátor rovný MAC adrese zařízení, které zprávu odeslalo.

7.4.3. Vyčtení aktuálního stavu

Při prvotním vykreslení tlačítka je pozadí bílé, jelikož je stav neznámý. Aplikace na vykreslení prvku zareaguje tím, že položí dotaz serverové službě, poskytující informace o aktuálních stavech zařízení. Následně je zpráva zachycena a tlačítko změněno. Celý kód této události je vypsán v kód 7.4.

```

if($mqttConnected)
{
  $( ".light-button" ).each(function( index )
  {
    $deviceID= $(this).attr('id');
    if($deviceID != null&&$deviceID != "")
    {
      var message = newPaho.MQTT.Message({"serialNumber":"" + $deviceID+""});
      message.destinationName = "home/state/in";
      client.send(message);
    }
  });
}

```

Kód 7.4. Vyčtení stavu zařízení

7.5. Hlasové ovládání

System obsahuje možnost ovládání domácnosti hlasovými příkazy podobným jako u grafického ovládání. Hlasové ovládání je možné díky javascriptové knihovně SpeechCloud, která byla poskytnuta katedrou kybernetiky na Západočeské univerzitě v Plzni. Díky této knihovně se lze připojit na server, kde funguje SpeechCloud a následně s ním komunikovat.

7.5.1. SpeechCloud

Jedná se aplikační rozhraní, které poskytuje služby jako je rozpoznávání, syntéza a porozumění řeči. Tato aplikace vznikla na katedře kybernetiky ve spolupráci s firmou SpeechTech. Komunikace mezi klientem a serverem probíhá skrz protokol WebSocket. Předávání audia je pomocí SIP protokolu a uskutečňuje se pomocí RTP UDP paketů. Tento protokol se hojně využívá v oblasti internetové telefonie.

Gramatika je předávána serveru pomocí textových řetězců. Následně se server podle této gramatiky řídí při rozpoznávání řeči. V průběhu vytváření systému byla zjištěna větší časová prodleva po předložení většího množství textových řetězců definující gramatiku. Je tedy nutné, aby při načtení aplikace byl uživatel upozorněn na situaci, že není rozpoznávání připraveno, ať z důvodu špatného připojení nebo aktuálního nacvičování jazykového modelu. Je tedy upozorňován zablokováním a odblokováním tlačítka v horní liště. Po

příchodu potvrzení od serveru se tlačítko odblokuje a uživatel má možnost začít hlasový dialog se systémem.

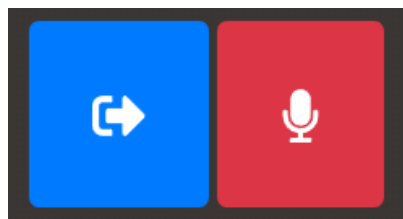
7.5.2. Konstrukce gramatiky

Při tvorbě systému byla navržena taková gramatika, aby co nejvíce pokryla vymezenou oblast pro ovládání domácnosti. K návrhu gramatiky byla využita nejpoužívanější slova a slovní spojení s ohledem na typy zařízení, které umí systém zpracovat a ovládat. Pro jednotlivá zařízení jsou tedy vytvořena pole definující příkazy pro ovládání a následně další pole definující stavy zařízení.

Při načtení stránky jsou pomocí Ajaxových dotazů na databázi vytvořeny pole názvů místností spolu s názvy zařízení. Dále jsou ve skriptu pole definující stavy a příkazy pro jednotlivá zařízení. Všechna tato pole následně spolu s bezkontextovou gramatikou vytvoří jedno velké pole, které se předloží hlasovému rozpoznávači jako gramatika systému.

7.5.3. Vypnutí/zapnutí

Jedná se o funkci, která vypne nebo zapne reagování na hlasové projevy uživatele. V testování se tato funkce velice uplatnila v rušném prostředí, kdy dochází k automatickému vypínání rozpoznávání řeči. Pokud však uživatel chce vypnout nebo zapnout rozpoznávání, je zde tlačítko v horní liště, které je zobrazené na obrázku 7.7.



Obrázek 7.7. Horní lišta webové aplikace

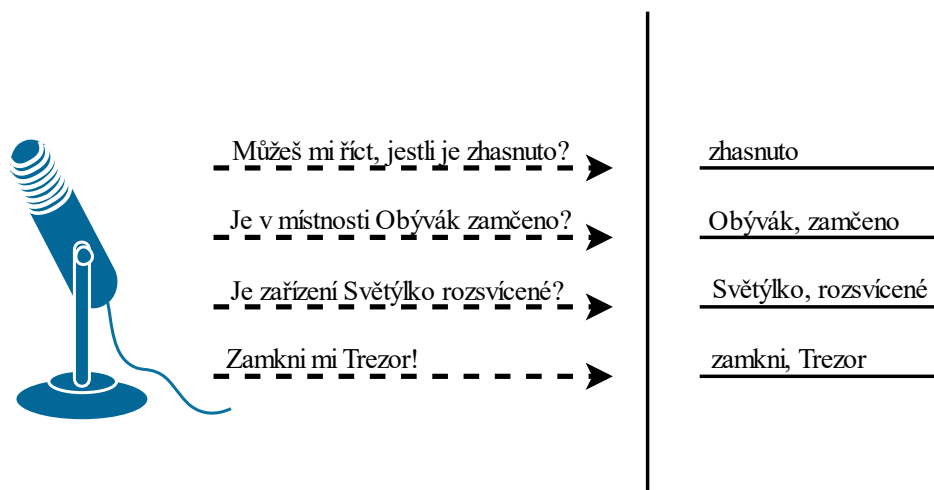
7.5.4. Detekce rušného pozadí

Detekce funguje pouze při zapnutém hlasovém ovládání. Rozpoznávání rušného pozadí funguje tak, že se porovnává hodnota *level* při události *asr_signal* od hlasového rozpoznávače. Pokud se několikrát daná hodnota pohybuje v rozmezí mimo vymezený interval, aplikace uživatele upozorní, že rozpoznávání nemusí být kvalitní. Následně je rozpoznávání vypnuté do té doby, než si je uživatel opět sám zapne.

7.5.5. Dialog

Dotazy na stavy, ale i ovládání zařízení je možné pomocí dialogu s aplikací. Na základě hlasových příkazů jsou pokládány mluvené dotazy, na které má uživatel možnost odpovědět do 10 vteřin od konce promluvy systému. Pokud následuje negativní odpověď nebo uživatel neodpoví do již zmíněných 10 vteřin, je poslední dotaz a vyčkání na odpověď zapomenuty. Systém v době, kdy promlouvá, nereaguje na mluvenou řeč od uživatele. Je tedy nutné provést odpověď až po dokončení řeči ze strany aplikace.

Při každé promluvě je řeč rozdělena na jednotlivá slova a z ní jsou dále určena ta slova, která jsou důležitá pro systém zpracování. Důležitá slova jsou například názvy zařízení a místností, která jsou níže vyznačena velkými písmeny. Dále jsou implementovány příkazy pro jednotlivá zařízení jako je například slovo *zamkni* popřípadě dotaz na stav *zhasnuto*.



Obrázek 7.8. Výběr důležitých slov z promluvené řeči

Jelikož je možné, že uživatel bude používat nespisovnou češtinu, je dalším krokem výběr správného tvaru slova zadaného příkazu. Toto slovo je poté dosazeno do věty *Přejete si vědět, jestli je zařízení* + dosazované slovo je dále zpracováván v PHP skriptu při dokončování příkazu. Příklad možného výběru slov je v tabulce 7.1.

Příkaz nebo informace	Výsledný tvar
rozsvít, rozsvítit,	rozsvítit
zhaslé, zhasnuté, zhasnutý	zhasnuto

zamklé, zamknuté	zamknuto
------------------	----------

Tabulka 7.1. Převedení dotazu na spisovný tvar

Definované příkazy a dotazy na stavy jsou v jednotlivých textových polích. Tato pole jsou vypsána v příloze 1. Při porovnávání slova se hledá, jestli je výraz v některém poli. Pokud ano, vybere se první položka z tohoto pole, kde se nachází právě tento potřebný tvar a nahradí vyřčené slovo od uživatele.

Na základě první promluvy systém pokládá otázky, které ho mají směřovat k výslednému úkonu. Zároveň je systém schopný reagovat pouze na příkaz a následně se dotázat na cíl, ale je schopen reagovat i na příkaz s definovaným cílem zařízení nebo místnosti v jedné promluvě.

1. Příklad dialogu

Uživatel: Rád bych se dozvěděl, jestli je rozsvíceno.

Systém: Přejete si vědět, jestli je někde rozsvíceno?

Uživatel: Ano.

Systém: O jaké místnosti nebo zařízení se chcete tento stav dozvědět?

Uživatel: Lampička

Systém: Zařízení Lampička není rozsvíceno.

2. Příklad dialogu 2

Uživatel: Rád bych se dozvěděl, jestli je Lampička rozsvícená.

Systém: Přejete si vědět, jestli je zařízení Lampička rozsvíceno?

Uživatel: Ano

Systém: Zařízení Lampička není rozsvíceno.

7.5.6. Dotaz na stav

Po dokončení dialogu, který směřoval ke zjištění informace o místnosti nebo zařízení, máme textovou hodnotu zařízení (nebo místnosti) a textovou hodnotu definující dotaz na stav. Pokud se ptá uživatel na informace ohledně zařízení, je skriptu pro zpracování předložen název zařízení a dotazovaný stav. Skript ho prvně zpracuje tak, že vezme již zmíněný stav a vygeneruje z něj typ zařízení spolu s číselnou hodnotou, které definuje dotaz. Tento postup je zobrazen v kódu 7.5.

Například pokud se ptá uživatel, jestli je nějaké zařízení rozsvíceno, je hodnota rovna logické jedničce. Pokud se ptá na stav zhasnuto, je hodnota rovna logické 0.

```
if ($request == "rozsvíceno" || $request == "zhasnuto")
{
    $deviceType = "Light";
    $request == "rozsvíceno"? $actualState = 1: $actualState = 0;
}
elseif($request == "zamknuto" || $request == "odemčené")
{
    $deviceType = "Lock";
    $request == "odemčené"? $actualState = 1: $actualState = 0;
}
```

Kód 7.5. Vygenerování typu zařízení na základě dotazu

V tomto kroku, který ukazuje kód 7.5., se zároveň zajistí smysluplnost dialogu. Pokud se uživatel zeptá například jestli má *zamčeno* zařízení s typem *světlo*. Je vygenerovaný typ *světlo* a není tedy vyhledán žádný údaj, který by odpovídal vyhledávání. Vyhledávání zařízení je ukázáno na části kódu 7.6.

```
$sql = "SELECT * FROM device WHERE namesource = '$deviceName' AND type = '$deviceType';"
```

Kód 7.6. Vyhledávání zařízení na základě typu a názvu

Dalším krokem je vyhledat ID zařízení podle typu a názvu. Pokud je toto zařízení nalezeno, pokračuje dál a v opačném případě je uživatel informován o tom, že dané zařízení nebylo nalezeno. K této situaci může dojít tehdy, když se uživatel například zeptá na stav zhasnuto a předloží název zařízení, které je typu jiného než světlo.

Následuje vyhledání posledního zaevidovaného stavu v databázi a porovnání s hodnotou, která byla vygenerována spolu s typem zařízení. Pokud byl předložen dotaz na rozsvícené zařízení s názvem *Lampička* a poslední zaevidovaný stav je roven hodnotě 0, je odpověď „Zařízení Lampička není rozsvíceno“. Pokud je roven hodnotě 1, odpověď bude „Zařízení Lampička je rozsvíceno“. Tento postup je zobrazen v kódu 7.7.

```

if($rowLogger["loggerValue"] == $actualState)
{
    $state = "Zařzení $deviceName je $request";
}
else
{
    $state = "Zařzení $deviceName není $request";
}

```

Kód 7.7. Vygenerování odpovědi

Pokud je požadavek zaměřen na místnost, je postup téměř stejný. Jsou předloženy stejné hodnoty krom hodnoty obsahující název zařízení. Tu nahradí hodnota s názvem místnosti. Prvním krokem je tedy nalezení ID dané místnosti a vygenerování typu zařízení podle předloženého dotazu. Dále jsou nalezeny všechna zařízení dle vygenerovaného typu, která jsou přiřazena k místnosti. Odpověď je závislá na počtu zařízení a na jejich posledním evidovaném stavu v databázi. Pokud je tedy nalezeno více než jedno zařízení, jehož stav je roven hledanému, je vygenerovaná odpověď ve tvaru *V místnosti je + dotazovaný stav*. V případě, kdy bylo nalezeno pouze jedno zařízení, je odpověď vygenerovaná tak, aby informovala o stavu místnosti a zároveň bylo zmíněno dané zařízení. Pokud nastane, že žádné zařízení v místnosti nenabývá hledaného stavu, je vygenerována odpověď rovna *V místnosti není + dotazovaný stav*.

7.5.7. Příkaz

Aby byla promluva od uživatele vyhodnocena jako příkaz pro domácnost, je nutné, aby byly hodnoty pro zařízení nebo místnost spolu s příkazovou hodnotou vyplněny. Zařízení, která reagují na požadavky ke změně stavu, mají definované pole příkazů. Pokud se v promluvě objeví některé slovo z tohoto pole, je klasifikováno jako příkaz.

```

var light_enable = ["rozsvítit", "rozsviť", "zapni", "zapnout"];
var light_disable = ["zhasnout", "zhasni", "vypni", "vypnout"];

```

Kód. 7.3. Příkazy pro zařízení světlo

Dále je rozlišováno, zda je příkaz mířený na konkrétní zařízení nebo celou místnost. Pokud je cílem zařízení, je vyhledána MAC adresa zařízení podle názvu, který byl

uživatelé zadán. Po předložení názvu zařízení se tedy vrátí MAC adresa, typ zařízení a téma, na kterém by měla být zpráva následně odeslána. Dalším krokem je zjištění, zda příkaz je v poli příkazů daného typu zařízení. Pokud ne, je uživatel upozorněn a příkaz není proveden.

1. Příklad dialogu

Uživatel: Zhasni mi prosím zařízení Lampička.

Systém: Přejete si zhasnout zařízení Lampička?

Uživatel: Ano.

Systém: [Odeslání požadavku na zhasnutí]

2. Příklad dialogu 2

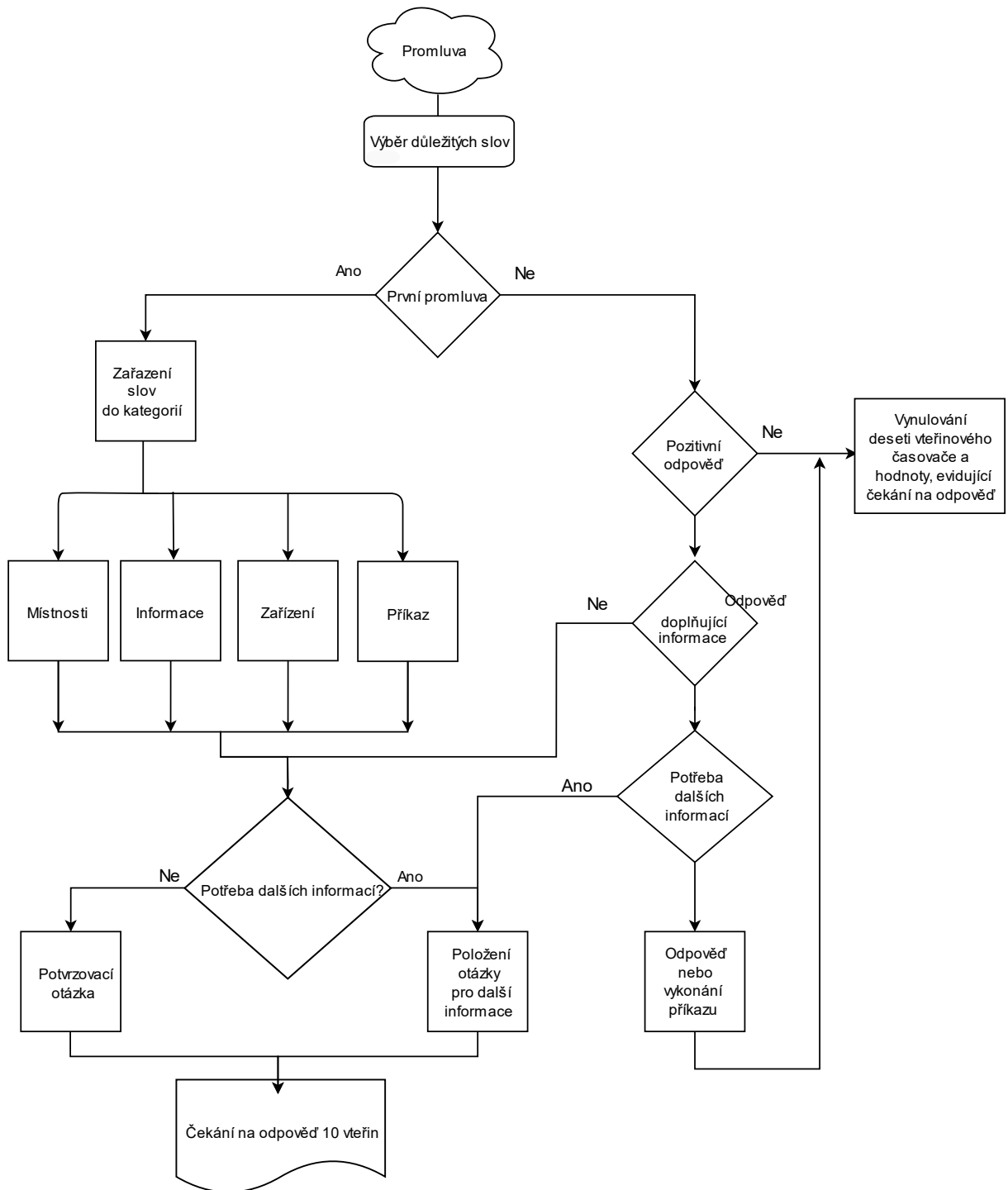
Uživatel: Zhasni mi prosím zařízení s názvem Květináček.

Systém: Přejete si zhasnout zařízení Květináček?

Uživatel: Ano.

Systém: Zařízení Květináček není správného typu.

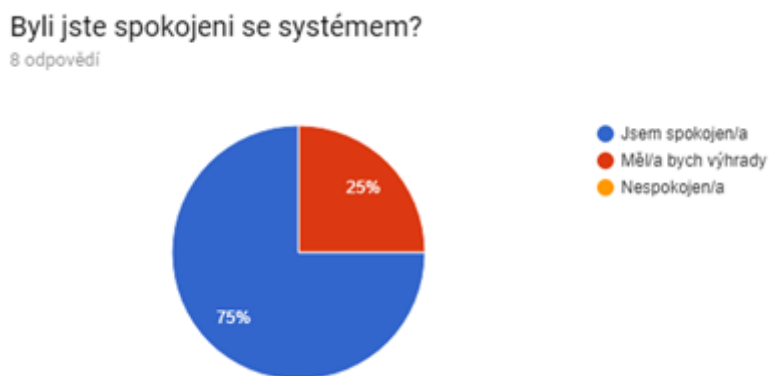
Schéma celého dialogu je vyobrazeno na obrázku 7.9.



Obrázek. 7.9. Struktura dialogového systému

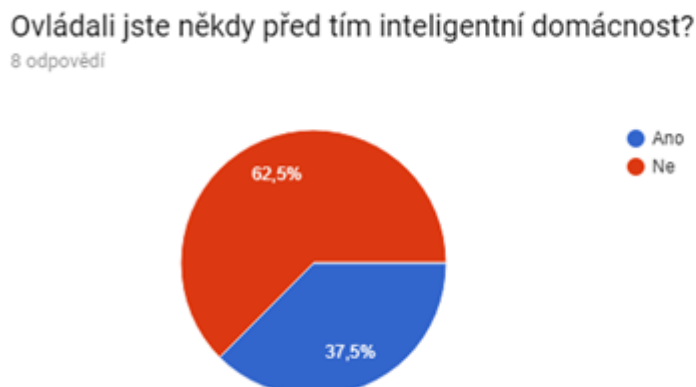
8. Testování

Testování se zúčastnilo celkem 8 subjektů, kde každý z nich testoval již vytvořenou domácnost. Bylo zapotřebí otestovat připojená zařízení pomocí grafického rozhraní a následně pomocí hlasového dialogu. Ve dvou případech byl problém s rozpoznáváním řeči, který byl způsoben špatným nastavením mikrofonu. Výsledky těchto testů dopadly vskutku dobře, jelikož většina uživatelů byla se systémem plně spokojena, jak uvádí graf na obrázku 8.1. Na základě promluv od uživatelův průběhu testování, byla přidána některá slova pro ovládání zařízení.



Obrázek. 8.1. Spokojenost uživatelů

Na základě grafu (viz obrázek 8.2.) bylo zjištěno, že pouze 3 jednotlivci měli zkušenosti s předchozím používáním inteligentní domácnosti.

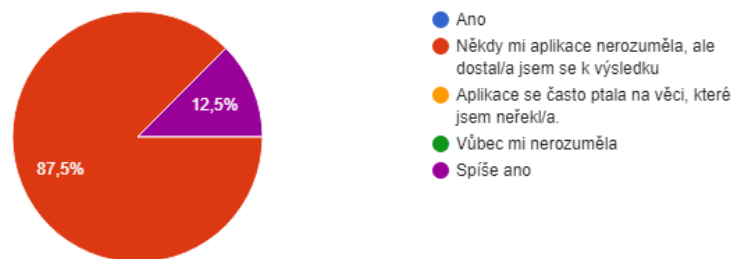


Obrázek. 8.2. Předchozí testování

Důraz byl hlavně na hlasové rozpoznávání a interakci s uživatelem. Tento test nedopadl na sto procent, ale můžeme ho i tak prohlásit za úspěšný (viz obrázek 8.3.). Velkou roli v tom hrála občasné nerozpoznání slov. V jednom případě se stalo, že si uživatel pojmenoval virtuální zařízení typu žaluzie názvem *Tma*, ale rozpoznávač mu ve většině času nerozuměl. Při řešení tohoto problému bylo zjištěno, že systém slovo promluvu slova *Tma* ve většině času klasifikuje jako dvě slova, konkrétně *To má*.

Bylo hlasové rozpoznávání v pořádku?

8 odpovědí



Obrázek. 8.3. Úspěšnost hlasového rozpoznání

Dále bylo nutno odpovědět, zda uživatel považuje rozdělení a ovládání prvků pomocí vytvořených místností za užitečné. Všechny odpovědi byly v tomto případě kladné (viz obrázek 8.4.). Uživatelé by tedy tuto funkci rádi využívali, pokud by u nich byl systém nainstalován.

Považujete za užitečné možnost ovládání celých místností?

8 odpovědí



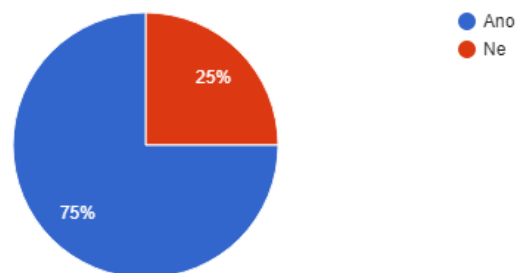
Obrázek. 8.4. Užitečnost místností

Dalším krokem bylo vyzkoušení rozpoznání nevhodného prostředí, kde dialog probíhá. V tomto prostředí by se měl systém zachovat tím, že uživatele upozorní a vypne hlasové

rozpoznávání. Tuto funkci vyzkoušelo, jak můžeme vidět na obrázku 8.5., 6 lidí na doporučení zapnout televizi nebo hudbu někde v blízkosti promluvy. Všem se systém choval, jak by v této situaci měl. Odpovědi na otázku *Pocíteli jste nějakou změnu při změně prostředí nebo mikrofonu*, jsou v tabulce 8.1.

Zkoušeli jste více mikrofonů nebo prostředí, kde jste systém ovládali hlasem?

8 odpovědí



Obrázek. 8.5. Více prostředí dialogu

Na doporučení zapnout televizi a vyzkoušet promlouvat – opravdu to nešlo. Pořád jsem byl upozorňován systémem na špatné rozpoznávání nebo paní povídala nesmysly.
Určitě záleží na hlučnost pozadí v místnosti, kde se právě nacházíte.
Při zapnuté televizi rozpoznání hlasu nefungovalo.
Když jsem byla v prostředí, které bylo jen trošičku hlučnější, tak se stalo, že mi aplikace ne vždy rozuměla.
Na tabletu slečna moc nerozuměla.

Tabulka. 8.1. Odpovědi na změnu prostředí

Poslední otázky byly vyhrazeny k funkcím, které mohou být v budoucnu do systému přiřazeny. Výsledky pro otázku *Je něco, co byste na systému změnili?* Jsou znázorněny v tabulce 8.2.

Možnost změnit hlas, na paní / pána jako je to u GPS.
Určitě odezva slečny, která odpovídá na ovládání hlasem. Nemusela by se tolik ptát.
Rychlejší odezva. Když řeknu "zamkni" a existuje jen 1 zámek, tak by mohl systém buď rovnou zamknout, nebo se zeptat na ten konkrétní zámek a až pak po potvrzení uživatele zamknout
Rychlost odezvy při hlasovém dialogu, upozornění, pokud zařízení neodpoví
Ne, se systémem jsem byla spokojená.
Abych si mohla vybrat, co mi má slečna říct. Například po každém přihlášení říct, jestli mám něco v poštovní schránce.

Tabulka. 8.2. Doplnění systému

Dále následovalo místo pro pochvaly (tabulka 8.3.) a problémy, se kterými se uživatelé setkali (tabulka 8.4.).

Grafická stránka
Na druhou stranu je hezké, že se slečna ptá na věci, které nebyly řečeny. Grafické ovládání je super!
Grafické zpracování
Grafický design
Grafický rozhraní
Jeho přehlednost
Určitě jednoduchost

Tabulka. 8.3. Pochvaly systému

Rozpoznávání hlasu
Jen u hlasového ovládání v hlučné místnosti (s TV nebo venku na mobilu).
1/5 příkazů mi domácnost nerozuměla
V rozpoznávání hlasu
Problém při rozpoznávání
Kromě občasného špatného rozpoznávání, nic.

Tabulka. 8.4. Problémy systému

9. Závěr

Výsledkem práce je systém inteligentní domácnosti, který byl v praxi nasazen a nadále je používán. Inteligentní prvky komunikují pomocí MQTT protokolu a jsou rozděleny na vstupní a výstupní zařízení, přičemž u výstupních je kladen důraz na spotřebu a u vstupních na rychlou odezvu na požadavky od uživatele. Všechna zařízení mají možnost nastavení v podobě možného výběru Wi-Fi sítě, do které chce uživatel zařízení připojit. Zařízení si toto nastavení pamatují do doby, než je uživatel vymaže pomocí tlačítka, které uvede zařízení do původního nastavení.

Systém nabízí možnost ovládání zařízení skrz grafické rozhraní, pomocí hlasového dialogu nebo kombinací těchto dvou technologií. Tento systém byl otestován osmi subjekty, kde dostal velice kladné hodnocení. Uživatelé ocenili grafické zpracování systému a upozornění na možnost špatného rozpoznávání kvůli ruchu v pozadí.

Seznam literatury

- [1] PECHÁČKOVÁ, Kateřina. *Bezpečnost Internetu věcí* [online]. 1.12.2017 [cit. 16.5.2018]. Dostupný na WWW: <https://medium.com/edtech-kisk/bezpe%C4%8Dnost-internetu-v%C4%9Bc%C3%AD-28af18f69e21>
- [2] PSUTKA, Josef. *Mluvíme s počítačem česky*. 1.Praha: Academia, 2006, 746. ISBN 80-200-1309-1.
- [3] MALÝ, Martin. *Protokol MQTT: komunikační standard pro IoT* [online]. 29. 6. 2016 [cit. 16.5.2018]. Dostupný na WWW: <https://www.root.cz/clanky/protokol-mqtt-komunikacni-standard-pro-iot/>
- [4] MALÝ, Martin. *Web Sockets* [online]. 14.12.2009 [cit. 16.5.2018]. Dostupný na WWW: <https://www.zdrojak.cz/clanky/web-sockets/>
- [5] Jan, Švec. *SpeechCloud API* [elektronická skripta]. ©2016 [cit. 16.5.2018]. Dostupný na WWW: https://docs.google.com/presentation/d/1bV4nedeZOjfgwbJlbJf7tR8GpZfMCAWoywfpQGJKVPA/edit#slide=id.gc6a7a5523_1_0
- [6] AUTOR NEUVEDEN. *Venkovní žaluzie zajistí stínění, ochranu i tepelnou izolaci* [online]. 26.11.2012 [cit. 16.5.2018]. Dostupný na WWW: <http://okna-dvere.bydleniprokazdeho.cz/rolety-a-zaluzie/venkovni-zaluzie-zajisti-stineni-ochranu-i-tepelnou-izolaci.php>
- [7] M, Luboš. *ESP8266 - vývojová deska WeMos D1* [online]. 21. 2. 2017 [cit. 16.5.2018]. Dostupný na WWW: <http://navody.arduino-shop.cz/navody-k-produktum/esp8266-vyvojova-deska-wemos-d1.html>
- [8] AUTOR NEUVEDEN. *Montáž alarmů Kladno – specialisté na alarmy Jablotron* [online]. 23. 2. 2018 [cit. 16.5.2018]. Dostupný na WWW: <http://www.izpravy.com/montaz-alarmu-kladno-specialiste-alarmy-jablotron/>

Seznam příloh

Příloha A Kompletní seznam slov sloužící jako gramatika

Příloha A Kompletní seznam slov sloužící jako gramatika (tabulka)

Pozitivní odpověď	ano, jo, chci, určitě, správně, jasně, jasnačka, přijmout, potvrzují, potvržu
Negativní odpověď	ne, nechci, vůbec, zamítnout, nechceme, zapomeň
Zapnutí světla (příkaz)	rozsvítit, rozsviť, zapni, zapnout
Vypnutí světla (příkaz)	zhasnout, zhasni, vypni, vypnout
Pozitivní stav světla	rozsvíceno, rozsvjeceno
Negativní stav světla	zhasnuto, zhaslé, zhasnuté
Pozitivní stav květináče	zalité, zalitý, zalito, zalít, vlhký, mokrý
Negativní stav květináče	vyprahlý, suché, suchý
Poštovní schránka stav	noviny, dopis, dopisy, poště, poštu, pošta
Odemknout zámek (příkaz)	odemknout, odemkni
Zamknout zámek (příkaz)	zamknout, zamkni
Negativní stav zámku	zamknuto, zamčené, zamknuté, zamknutý, zamčený, zamčená
Pozitivní stav zámku	odemčeno, odemčené, odemknuté, odemčený, odemknuto
Odrhnout rolety (příkaz)	odhrnout, odhrň, vytáhnout, vyhrnout, odroluj
Zahrnout rolety (příkaz)	zahrnout, zahrň, zatahnout, zaroluj
Pozitivní stav rolet	otevřeno, odtaženo, odtáhnuto
Negativní stav rolet	zataženo, zataženo, zatahnuto
Zájmena	já, mi, tě, to, ten, ta
Bezkontextová gramatika	chci, chtěl, bych, rád, prosím, prosim, můžeš, řekni, mám, někde, říct, říkat, přeji, přeju, je, hlína, dostatečně, nějaké, voda, žaluzie, rolety, roleta, můžeš, prosím, řekni, myslím, nemáme, vložka, jako, nespočetně, ježiš, marja, nevím, proč, ptá, co, cože, nerozumím, nikdy, zkus, ptáci, kdy, kam, kudy, proč, teta, strejda, strýc, matka, dcera, syn, zpívá, kouká, televize, noviny, zprávy, vydělává, nevydělává, zpívat, sušit, vyprat, vypracovat, pracovat, nebude, nedám, půjdu, půjdeš, nepůjdeš, nefunguje, funguje, poslední, první, se, nebav, vid', mrzí, netuším, rád, můžeme, nemůžeme, odlétám, jestli, nevíš, nevím, zařízení, dáváš, dobrý, dobrá, dobré, zavíral, okno, panty, dřevěné, v, u, o