



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

AUTOMATIZOVANÉ ŘÍZENÍ VYTÁPĚNÍ RODINNÉHO DOMU

AUTOMATED CONTROL OF FAMILY HOUSE HEATING

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Petr Slouka

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Petr Blažek

BRNO 2023

Bakalářská práce

bakalářský studijní program **Telekomunikační a informační systémy**

Ústav telekomunikací

Student: Petr Slouka

ID: 227255

Ročník: 3

Akademický rok: 2022/23

NÁZEV TÉMATU:

Automatizované řízení vytápění rodinného domu

POKYNY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce je realizace automatizovaného ovládání topných jednotek v rodinném domě pomocí WiFi bezdrátových jednotek napájených z akumulátorů. Celý systém se bude skládat ze čtyř dílčích částí – server pro sběr dat a ovládání, jednotka elektromagnetického ventilu pro ovládání průtoku topné vody, jednotka pro snímání teploty na přívodu topné vody a jednotka pro snímání teploty v dané místnosti. V rámci teoretické části se seznámte s programováním jednotek založených na čipech ESP32 a ESP8266. Dále proveďte analýzu a návrh napájení s požadavkem na nízkou spotřebu a dlouhodobý provoz. Dílčím cílem práce je realizace webového/mobilního rozhraní pro kontrolu a ovládání jednotlivých zařízení. V rozhraní bude možné nastavit jednotlivé místnosti v domácnosti, do kterých bude možné přiřadit el. mag. ventil a snímač teploty. Dále bude rozhraní obsahovat algoritmus, který bude efektivně ovládat vytápění jednotlivých místností na základě dat ze snímačů. Součástí rozhraní bude také databázový systém, který bude uchovávat informace ze všech jednotek.

Výstupem bakalářské práce bude automatizovaný systém pro ovládání vytápění rodinného domu s konfiguračním webovým/mobilním rozhraním. Dílčím výstupem bude testování funkčnosti systému (záznam a rozbor komunikace mezi zařízeními, měření cyklů akumulátorů, rozbor a implementace algoritmu pro správu vytápění).

DOPORUČENÁ LITERATURA:

- [1] MONK, S. Programming Arduino: Getting Started with Sketches, 1. vyd. New York: McGraw-Hill, 2011.
- [2] MANN, B. C pro mikrokontroléry. BEN - technická literatura, Praha 2003

Termín zadání: 6.2.2023

Termín odevzdání: 26.5.2023

Vedoucí práce: Ing. Petr Blažek

prof. Ing. Jiří Mišurec, CSc.
předseda rady studijního programu

ÚPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Práce se zabývá návrhem a realizací automatizovaného řízení vytápění rodinného domu. Systém je založen na programovatelných jednotkách, jejichž základními prvky jsou zařízení, která jsou založena na programovatelných čipech ESP32 nebo ESP8266. Hlavní částí je návrh systému a realizace jednotlivých jeho částí jako je webová aplikace a jednotlivé senzory. Dále se práce zabývá komunikací mezi jednotlivými částmi, jejím rozbořením a analýzou výdrže akumulátorů jednotlivých zařízení. Poslední kapitolu tvoří návod k instalaci a spuštění celého systému.

KLÍČOVÁ SLOVA

Automatizace, vytápění, Arduino, Laravel, webová aplikace, ESP8266, ESP32.

ABSTRACT

The thesis deals with the design and implementation of automated control of family house heating. The system is based on programmable units whose basic elements are devices based on ESP32 or ESP8266 programmable chips. The main part is the design of the system and the implementation of its individual parts such as the web application and individual sensors. Furthermore, the thesis deals with the communication between the individual parts, its analysis and the analysis of the battery life of the individual devices. The last chapter consists of instructions for installation and start-up of the whole system.

KEYWORDS

Automatization, heating, Arduino, Laravel, web application, ESP8266, ESP32.

SLOUKA, Petr. *Automatizované řízení vytápění rodinného domu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2023, 69 s. Bakalářská práce. Vedoucí práce: Ing. Petr Blažek

Prohlášení autora o původnosti díla

Jméno a příjmení autora: Petr Slouka
VUT ID autora: 227255
Typ práce: Bakalářská práce
Akademický rok: 2022/23
Téma závěrečné práce: Automatizované řízení vytápění rodinného domu

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Petru Blažkovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci. Dále bych rád poděkoval svojí rodině a přítelkyni za podporu a motivaci při studiu.

Obsah

Úvod	21
1 Teoretický úvod	23
1.1 Model TCP/IP	23
1.1.1 Vrstva síťového rozhraní – Ethernet/WiFi	23
1.1.2 Síťová vrstva – IPv4	24
1.1.3 Transportní vrstva – TCP	25
1.1.4 Aplikační vrstva – DHCP/HTTPS	25
1.2 Zařízení s čipy ESP8266	26
1.2.1 ESP-WROOM-02	27
1.2.2 LOLIN D1 mini	28
1.3 Programování jednotek ESP8266	29
2 Návrh systému a popis jeho částí	33
2.1 Návrh systému	33
2.1.1 Komunikace mezi jednotlivými částmi	34
2.2 Dílčí části systému a jejich popis	35
2.2.1 Ovládání průtoku vody	35
2.2.2 Jednotka pro snímání teploty na přívodu teplé vody	36
2.2.3 Snímač teploty v dané místnosti	36
2.2.4 Snímač venkovní teploty	37
3 Webové rozhraní	39
3.1 Architektura MVC a framework Laravel	39
3.2 Nastavení webového rozhraní	41
3.2.1 Konfigurace webového serveru	41
3.3 Ovládání ventilů	43
4 Analýza a testování systému	47
4.1 Výdrž akumulátorů	47
4.1.1 Výdrž akumulátoru snímačů teploty	47
4.1.2 Výdrž akumulátoru pro zařízení nastavující průtok ventilů	48
4.2 Kapacita záznamů databáze	49
5 Prvotní spuštění a instalace systému	51
5.1 Koncept instalace systému	51
5.2 Konfigurace jednotlivých komponent	51

Závěr	55
Literatura	57
A Přílohy	59
A.1 Obsah elektronické přílohy	59
A.2 Použité komponenty	60
A.3 Zjištění klíče zabezpečení	61
A.4 Schéma čipu ESP WROOM 02	62
A.5 Srovnání protokolů HTTP a HTTPS	63
A.6 Webová aplikace	64

Seznam obrázků

1.1	TCP/IP Zapouzdření	24
1.2	IPv4 adresa	25
1.3	Wemos LOLIN D1 mini	29
1.4	Výběr desky a připojeného portu	31
2.1	Návrh systému	33
2.2	Přehled metod kontroléru	35
2.3	Ovládání ventilů vytápění	35
2.4	Ventil topného okruhu	36
2.5	Termočlánek s převodníkem	36
2.6	Snímání teploty v místnosti	37
3.1	Architektura MVC	40
3.2	Ukázka hodnot pro otevření a zavření ventilu	45
4.1	Správa úložiště	49
5.1	Vzor seznamu místností	51
A.1	Postup pro zobrazení SSL klíče	61
A.2	Ověření platnosti certifikátu	61
A.3	Exportování klíče	62
A.4	Popis výstupů ESP WROOM 02	62
A.5	HTTP pakety	63
A.6	Detail HTTP paketu	63
A.7	HTTPS pakety	64
A.8	Detail HTTPS paketu	64
A.9	Domovská stránka webové aplikace	65
A.10	Seznam místností	66
A.11	Seznam uživatelů	67
A.12	Náhled hodnot pro správce aplikace	68
A.13	Náhled místnosti	69

Seznam tabulek

1.1	Základní parametry ESP8266EX	27
1.2	Přiřazení pinů ESP-WROOM-02	28
1.3	Základní parametry ESP-WROOM-02	28
1.4	Technická specifikace LOLIN D1 mini	29

Seznam výpisů

3.1	Vzor nastavení Apache serveru	41
3.2	Vzor nastavení kořenové adresáře	41
3.3	Konfigurační soubor přihlašování	42
3.4	Konfigurační soubor přihlašování	43
3.5	Vzor nastavení kořenové adresáře	43
5.1	Příklad úvodní konfigurace pro komponenty	52
5.2	Přiřazení pinů v zařízení ESP WROOM 02	53
5.3	Oprava chyby v importované knihovně	53

Úvod

Bakalářská práce je zaměřena na realizaci automatizovaného řízení vytápění rodinného domu, které je ovládáno bezdrátovými jednotkami založených na čipech ESP8266 a ESP32. První kapitola teoreticky popisuje přenos dat po síti a jednotlivé vrstvy modelu TCP/IP i jeho protokoly, které jsou využity při přenosu dat mezi jednotlivými částmi. Dále jsou v kapitole popsány zařízení s čipy ESP8266 a ESP32, jejichž programování je vysvětleno v poslední podkapitole. Druhou kapitolu tvoří návrh realizace systému a popis jednotlivých částí. První podkapitola pak popisuje zejména samotný návrh systému a způsob, jakým mezi sebou zařízení komunikují, druhá je pak zaměřena na popis jednotlivých částí a způsob jejich napájení. Třetí kapitola popisuje samotné webové rozhraní, přičemž v úvodu je popsán způsob architektury a hrubě vysvětlen framework Laravel, dále je zde vysvětleno nastavení webového rozhraní a je rozebrán algoritmus pro nastavení vytápění. Čtvrtá kapitola je zaměřena na analýzu a testování systému, kdy první podkapitola vysvětluje popis měření a teoretickou výdrž akumulátorů jednotlivých částí systému. Druhá podkapitola popisuje na praktickém příkladu rozdíl mezi HTTPS a HTTP komunikací a třetí vysvětluje algoritmus pro nastavení jednotlivých ventilů topných okruhů z pohledu zařízení Wemos ESP WROOM 02. V poslední podkapitole je uvedena teoretická kapacita použité databáze, respektive náročnost na obsazené místo na pevném disku z pohledu zaplnění databáze. V páté kapitole této práce je popsána prvotní instalace do rodinného domu nebo bytu. Je zde dále popsán koncept instalace systému automatizovaného řízení vytápění do objektu, čtenáři je tak podán ucelený postup pro implementaci systému do jakéhokoli objektu.

1 Teoretický úvod

Práce řeší automatizované řízení vytápění rodinného domu. K tomu, aby celá aplikace složená z komponent, které jsou vysvětleny dále v textu, správně fungovala, je využito několik komunikačních protokolů, které zabezpečují přenos dat pomocí určitého média (vzduchem či kabelem).

Všechna zařízení, která jsou předmětem této práce, komunikují zpravidla prostřednictvím WiFi (Wireless Fidelity) a dále je pro komunikaci mezi nimi klíčový síťový protokol IP (Internet Protocol), transportní protokol TCP (Transmission Control Protocol) a aplikační protokol HTTP (Hypertext Transfer Protocol), resp. jeho zabezpečená verze protokolem TLS (Transport Layer Security) protokol HTTPS (Hypertext Transfer Protocol Secure).

Automatizované řízení vytápění rodinného domu využívá domácí lokální síť LAN (Local Area Network) a síť WAN (Wide Area Network), resp Internet pro spojení s webovou aplikací. Lokální sítě umožňují přenos dat v prostorově omezeném měřítku na vzdálenost v řádu maximálně několika jednotek kilometrů. Pro přenos dat se nejčastěji na vrstvě síťového rozhraní používají technologie Ethernet a WiFi. Síť WAN se využívají k propojení LAN sítí např. prostřednictvím protokolu FR (Frame Relay). Jsou to globální sítě pokrývající rozlehlou oblast v řádu až tisíců kilometrů, např. města anebo menší státy. Propojením několika WAN sítí vzniká tzv. Internetworking, zkráceně Internet. [1]

1.1 Model TCP/IP

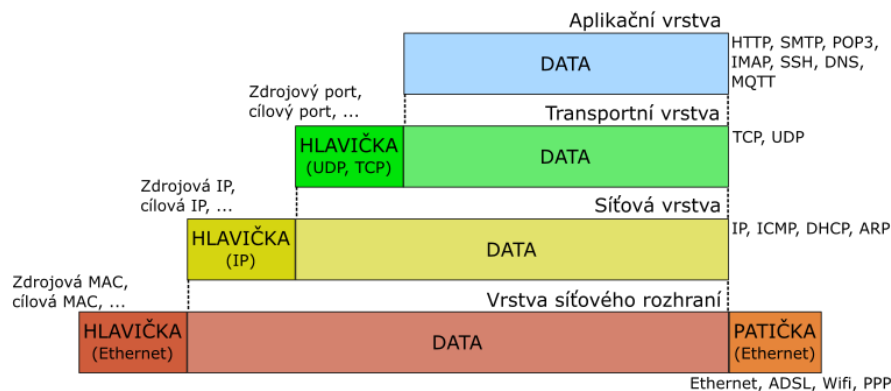
Soustava protokolů TCP/IP je velmi často využívaný model pro komunikaci mezi zařízeními v sítích. Na rozdíl od modelu ISO/OSI využívá 4 vrstvy: aplikační, transportní, síťovou a vrstvu síťového rozhraní.

Na obrázku 1.1 vidíme schéma zapouzdření dat na jednotlivých vrstvách a máme tak alespoň grafickou představu o tom, jaké protokoly mohou být v rámci jednotlivých vrstev použity.

Při komunikaci mezi jednotlivými zařízeními rozlišujeme, zda zařízení zprávu odesílá nebo přijímá. Pokud zařízení zprávu odesílá postupujeme podle obrázku 1.1 shora od dané aplikace. Pokud je zařízení příjemcem dat, postupujeme odspodu od síťového rozhraní k dané aplikaci.

1.1.1 Vrstva síťového rozhraní – Ethernet/WiFi

Jedná se o nejnižší vrstvu síťové komunikace a má na starosti vše, co je spojeno s ovládáním dané přenosové cesty. [1] Vrstva je rovněž závislá na přenosové techno-



Obr. 1.1: Schéma zapouzdření aplikačních dat na vrstvách TCP/IP [3]

logii.

Na této vrstvě probíhá komunikace zpravidla na základě MAC (Media Access Control) adresy jednotlivých zařízení, kterým je u osobních počítačů síťová karta, u síťových prvků to mohou být jednotlivá rozhraní. Najdeme zde protokoly pro dané přenosové technologie jako Ethernet (pro kabelové spojení) i WiFi (pro bezdrátové přenosy). V případě využití technologií WiFi mluvíme o WLAN (Wireless LAN), resp. o bezdrátové lokální síti. Zmíněnou technologii pro bezdrátový přenos dat využívají všechna zařízení automatizovaného vytápění, která se nachází v dané domácnosti. [1]

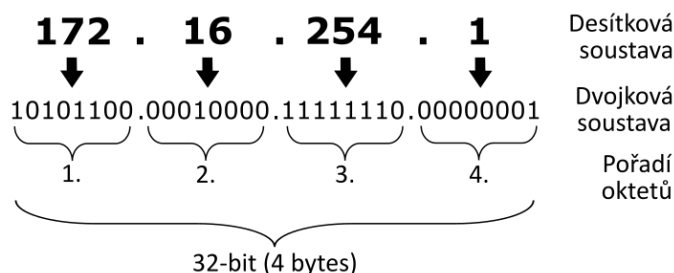
1.1.2 Síťová vrstva – IPv4

Síťová vrstva není závislá na použité přenosové technologii a používá protokol IP verze 4 nebo 6. Zprávy se v této vrstvě nazývají paketem a úkolem této vrstvy je, aby se dané pakety dostaly ke svému správnému příjemci. [1] Protokolem verze 6 se práce nezabývá, jelikož je v rámci lokální sítě využita pouze 4. verze IP protokolu.

Každé zařízení v síti má pro danou síť svoji unikátní IP adresu, jejíž příklad můžeme vidět na obrázku 1.2. IPv4 adresa se tedy skládá ze čtyř oktétů oddělených tečkou, což dává dohromady 32 bitů. Síť bývá rovněž označena svojí IP adresou, ke které se přidává maska sítě, která udává velikost sítě, resp. podsítě.

Maska sítě IPv4

Maska sítě má stejný zápis jako IPv4 adresa, nicméně nám udává kolik míst v binárním vyjádření můžeme změnit. Například maska sítě 255.255.0.0 (zkráceně zapsáno



Obr. 1.2: Struktura a popis IPv4 adresy

jako /16 – součet dvou oktetů) říká, že můžeme u IP adresy měnit jen třetí a čtvrtý oktet, jelikož maska sítě je v této části nulová.

1.1.3 Transportní vrstva – TCP

Třetí transportní vrstva je realizována protokolem TCP (Transmission Control Protocol). Hlavním úkolem vrstvy je zajistit přenos mezi dvěma koncovými účastníky a podle požadavků může tato vrstva regulovat tok dat oběma směry nebo zajišťovat spolehlivost přenosu. [1]

1.1.4 Aplikační vrstva – DHCP/HTTPS

DHCP

DHCP (Dynamic Host Configuration Protocol) je protokol aplikační vrstvy pro dynamické nastavení parametrů v síti, zejména nastavení parametrů koncových zařízení. Jedná se primárně o parametry jako adresa zařízení, maska sítě a výchozí brána, případně parametry DNS a další. [1]

Tento protokol funguje na principu klient-server, kdy se server je v našem případě na domácím routeru. DHCP server funguje tak, že zařízení parametry sítě propůjčuje a zaznamenává si jejich platnost (lease time). Po uplynutí této doby musí stanice požádat DHCP server o dané parametry znovu. [1] Je možné přiřadit zařízením jejich adresu trvale, tzn. že žádné jiné zařízení tuto adresu nedostane. Trvalé přiřazování adres probíhá na základě MAC adresy zařízení (síťové karty).

HTTPS

Protokol HTTPS je zjednodušeně řečeno zabezpečený protokol HTTP zpravidla protokolem TLS. Dříve bylo k zabezpečení protokolu HTTP využito protokolu SSL (Secure Sockets Layer), který je zjednodušeně řečeno předchůdcem protokolu TLS, který z něho vzešel, konkrétně z SSL verze 3.0.

Protokol HTTP je protokol na aplikační vrstvě, který slouží ke komunikaci mezi klientem (prohlížeč v počítači) a webovým serverem, pomocí kterého můžeme přenášet různá data ve formě textu, obrázků apod. Na straně server je zpravidla využíván TCP port 80. [1] Pokud bychom zachytávali pakety programem Wireshark¹, mohli bychom vidět přímo psané části nějaké webové stránky, hesla a další údaje. Takto nezabezpečený web může představovat v případě bezdrátového přenosu dat v sítích bezpečnostní riziko, protože není problém sledovat bezdrátový provoz a z něj získat potřebná data. Porovnání nezabezpečeného a zabezpečeného přenosu je vidět na obrázcích v příloze A.5.

Při sloučení protokolu HTTP a TLS vzniká protokol HTTPS. Data přenášená tímto protokolem jsou šifrována, což umožňuje zabezpečenou komunikaci mezi zařízeními. Tím, že je šifrování prováděno na vyšší vrstvě modelu TCP/IP, jsou zašifrovaná data bez zásahu do nich přenesena celou sítí mezi koncovými zařízeními. [2]

Pro názorný příklad se můžeme podívat na obrázky v příloze A.5. Na obr. A.5 a A.6, kde v detailu zachyceného přenosu mezi zařízeními lze vidět i celou URL adresu, kterou zařízení sítí odesílá. Z adresy tak lze vyčíst i využívaný klíč, kterým webová aplikace ověřuje oprávněnost k zápisu dané hodnoty do databáze. Na obr. A.8 vidíme šifrování přenosu protokolem TLS verze 1.3, který data šifruje do nečitelné podoby, která je vidět na řádku „Encrypted Application Data“.

Na obr. A.7 lze pak vidět komunikaci mezi senzorem pro snímání teploty a webovou aplikací jako v odstavci výše, která je již šifrovaná. Na obrázku si dále můžeme všimnout sestavení spojení na úrovni transportní vrstvy protokolem TCP (řádek 1505 a 1510). Následně pak probíhá výměna klíčů pro zabezpečenou komunikaci protokolem TLS verze 1.3 a na dalších následujících řádcích pak probíhá zabezpečená komunikace protokolem HTTPS.

1.2 Zařízení s čipy ESP8266

Série zařízení, která je založena na čipech ESP8266, respektive na jeho vylepšené verzi ESP8266EX, od společnosti Espressif Systems obsahuje zabudovaný WiFi mikročip s integrovaným softwarem pro komunikaci v sítích TCP/IP. Dále implementuje správu výkonu a disponuje 32 piny, z čehož je 20 pinů vstupně-výstupních (I/O – Input/Output), 4 vstupní (Input) a 8 napájecích (Power). [8] Základní parametry čipu jsou uvedeny v tabulce 1.1.

Zařízení, která jsou založena na čipech ESP32, disponují více piny. To znamená, že k zařízení je možné připojit více periférií, protože je možné programovat více

¹<https://www.wireshark.org/>

vstupně-výstupních portů. Dále jsou čipy ESP32 na rozdíl od čipů ESP8266 vybaveny Bluetooth modulem.

Tab. 1.1: Základní parametry ESP8266EX [8]

Procesor	Tensilica L106 32-bit RISC
Frekvence procesoru	160 MHz
Pracovní napětí	2,5 - 3,5 V
Pracovní proud	průměrně 80 mA
Paměť	32 kB instruction RAM 32 kB instruction cache RAM 80 kB user-data RAM 16 kB ETS system-data RAM
Podporované rozhraní	UART, SDIO, GPIO, ADC, PWM, SPI, I2C/I2S, IR ²
WiFi protokol	802.11 b/g/n
Síťové protokoly	IPv4, TCP, UDP, HTTP
Zabezpečení	WPA, WPA2
Šifrování	WEP, TKIP, AES

Neuvedené parametry jako například kapacita flash paměti³, ve které se ukládají programy, záleží vždy na aktuálním provedení. ESP8266EX podporuje až 16 MB externí flash paměti. Minimální velikost externí paměti je 512 kB, když je OTA⁴ vypnuto, nebo 1 MB, když je OTA zapnuto. [8]

Deep-sleep mód

V případě uvedení zařízení do tzv. deep-sleep módu klesne jeho pracovní proud na 20 μ A. V takovém případě je v provozu pouze RTC (Real Time Clocks)⁵ a všechny ostatní prostředky jsou vypnuty. Tohoto je využito při realizaci zařízení v této práci. [8]

1.2.1 ESP-WROOM-02

Zařízení můžeme popsat jako vývojovou desku využívající čip ESP8266, který je popsán v kapitole 1.2. Využívá například stejnou procesorovou jednotku, mírně se liší uváděné pracovní napětí. Na rozdíl od základního modelu ESP8266 má zařízení

³Flash paměť – elektricky programovatelná paměť s libovolným přístupem.

⁴Over-the-air (OTA) upgrade umožňuje bezdrátový přenos softwaru, firmwaru a dalších dat.

⁵RTC – Digitální čas jehož jediným úkolem je zajistit reálný čas zařízení, zatímco ostatní prostředky jsou vypnuty.

implementován protokol FTP (File Transfer Protocol)⁶. Základní parametry zařízení jsou uvedeny v tabulce 1.3.

Tab. 1.2: Přiřazení pinů čipu k portům na desce ESP-WROOM-02

Port	D1	D2	D3	D4	D5	D6	D7	D8
IO Pin	5	4	0	2	14	12	13	15

Na obrázku A.4 jsou zobrazeny jednotlivé výstupní piny čipu, který je v kapitole 2.2.1 vidět na daném zařízení. V dokumentaci k danému zařízení, které je použito při realizaci této práce, nebylo možné vyhledat přiřazení pinů k výstupním portům. Měřením bylo následně zjištěno, že pořadí jednotlivých portů neodpovídá daným pinům. Zjištěné přiřazení jednotlivých pinů k portům na desce je uvedeno v tabulce 1.2.

Tab. 1.3: Základní parametry ESP-WROOM-02 [9]

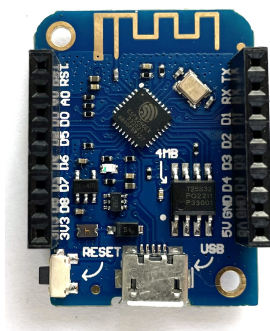
Procesor	Tensilica L106 32-bit RISC
Frekvence procesoru	160 MHz
Pracovní napětí	2,7 - 3,6 V
Pracovní proud	průměrně 80 mA
Paměť	32 kB instruction RAM 32 kB instruction cache RAM 80 kB user-data RAM 16 kB ETS system-data RAM
Podporované rozhraní	UART, HSPI, I2C/I2S, GPIO, PWM, IR
WiFi protokol	802.11 b/g/n
Síťové protokoly	IPv4, TCP, UDP, HTTP, FTP
Zabezpečení	WPA, WPA2
Šifrování	WEP, TKIP, AES

1.2.2 LOLIN D1 mini

Vývojová deska LOLIN D1 mini od firmy Wemos je založena na čipu ESP8266EX. V závislosti na verzi tohoto zařízení se můžeme setkat s různými velikostmi flash

⁶FTP – Komunikační protokol pro přenos dat mezi koncovými zařízeními.

paměti a různým druhem USB konektoru pro připojení například k počítači pro nahrání potřebného programu. Rovněž se liší počet použitelných pinů zařízení. V aktuálně používané verzi 3.0.0, viz obr. 1.3, je například k dispozici mikro USB, v novější verzi 4.0.0 je již implementováno USB typu C. Dále je na zařízení tlačítko reset, které restartuje zařízení, a tím opětovně spustí běh nahraného programu.



Obr. 1.3: Wemos LOLIN D1 mini

Tab. 1.4: Technická specifikace LOLIN D1 mini

Pracovní napětí	3,3 V
Digital I/O Pins	11
Analog Input Pins	1 (max. 3,2 V
Flash paměť	4 MB
Velikost zařízení	34,2 x 25,6 mm
Hmotnost	3 g

1.3 Programování jednotek ESP8266

Čip ESP8266 je tvořen tak, aby do něj bylo možné nahrávat svůj vlastní software. Použitelnými programovacími jazyky je např. Lua, Micropython anebo Wiring (Arduino IDE). Poslední zmíněný programovací jazyk je využíván i při realizaci jednotlivých komponent automatizovaného vytápění.

Arduino je programovací jazyk vytvořený na základech jazyka C++, který využívá vlastní software pro vývoj jednotlivých programů – Arduino IDE⁷ (Arduino Integrated Development Environment) [6] Prvním krokem k programování jednotek

⁷<https://www.arduino.cc/>

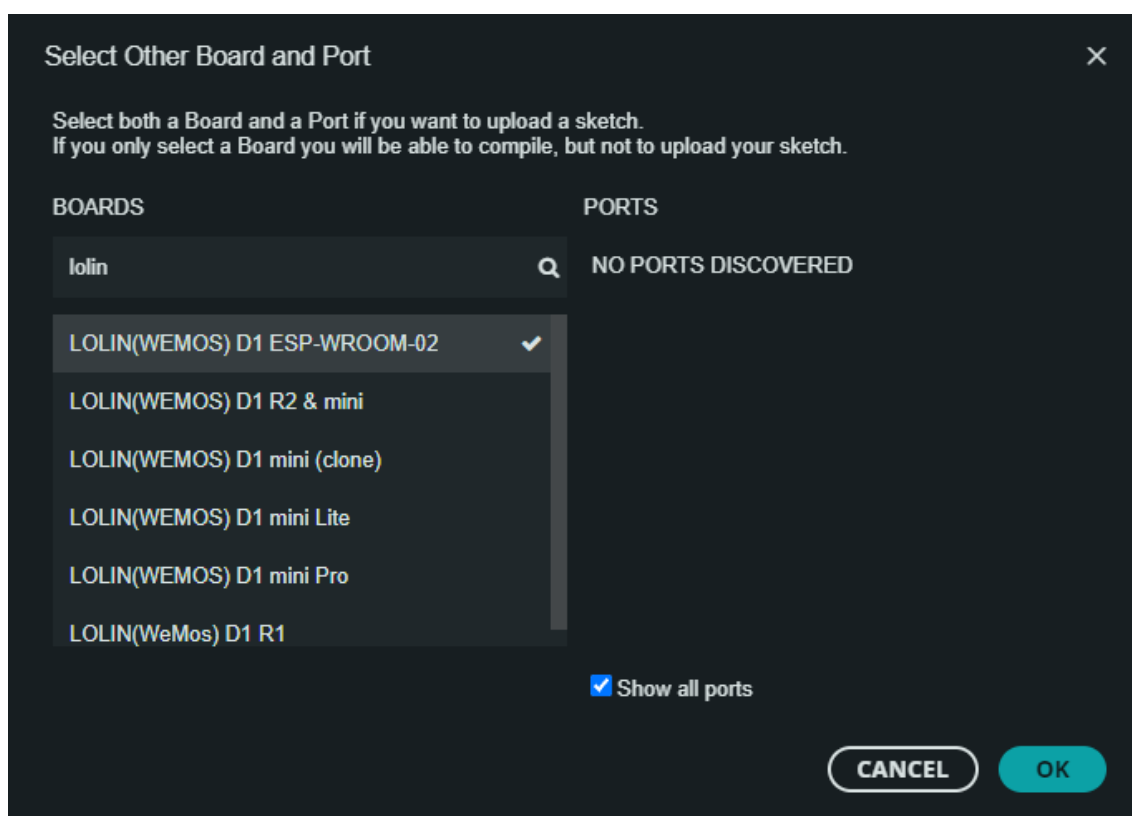
pomocí Arduina je instalace uvedeného softwaru. Na webové stránce pro Arduino IDE je nutné vybrat příslušný instalační soubor podle používaného operačního systému. Po jeho instalaci je dále nutné doinstalovat balíček pro ESP8266. Je nutné tedy otevřít vlastnosti aplikace a do pole „URL pro správu dalších desek“ vložit odkaz na příslušný balíček⁸. Nakonec přes možnost „Nástroje“ vybereme příslušnou vývojovou desku.

Před připojením fyzického zařízení je nutné nainstalovat ovladač CH340⁹ umožňující programování těchto jednotek přes USB. Instalace je nutná, jelikož zařízení založené na čipu ESP8266 odesílá a přijímá data sériově a nikoli paralelně, jako je tomu standardně při využívání USB. Nyní je možné zařízení programovat.

Po připojení zařízení k počítači a otevření softwaru pro jeho programování – Arduino IDE. Na obrázku 1.4 je možné vidět seznam nabízených desek společnosti Wemos. Tímto si můžeme ověřit správnou instalaci programu a knihovny, zobrazená nabídka by nebyla dostupná, pokud jsme v postupu instalace programu a knihovny udělali chybu. V pravé části by pak byly zobrazeny porty (COM), do kterých je příslušné zařízení připojeno.

⁸http://arduino.esp8266.com/stable/package_esp8266com_index.json

⁹https://www.wemos.cc/en/latest/ch340_driver.html



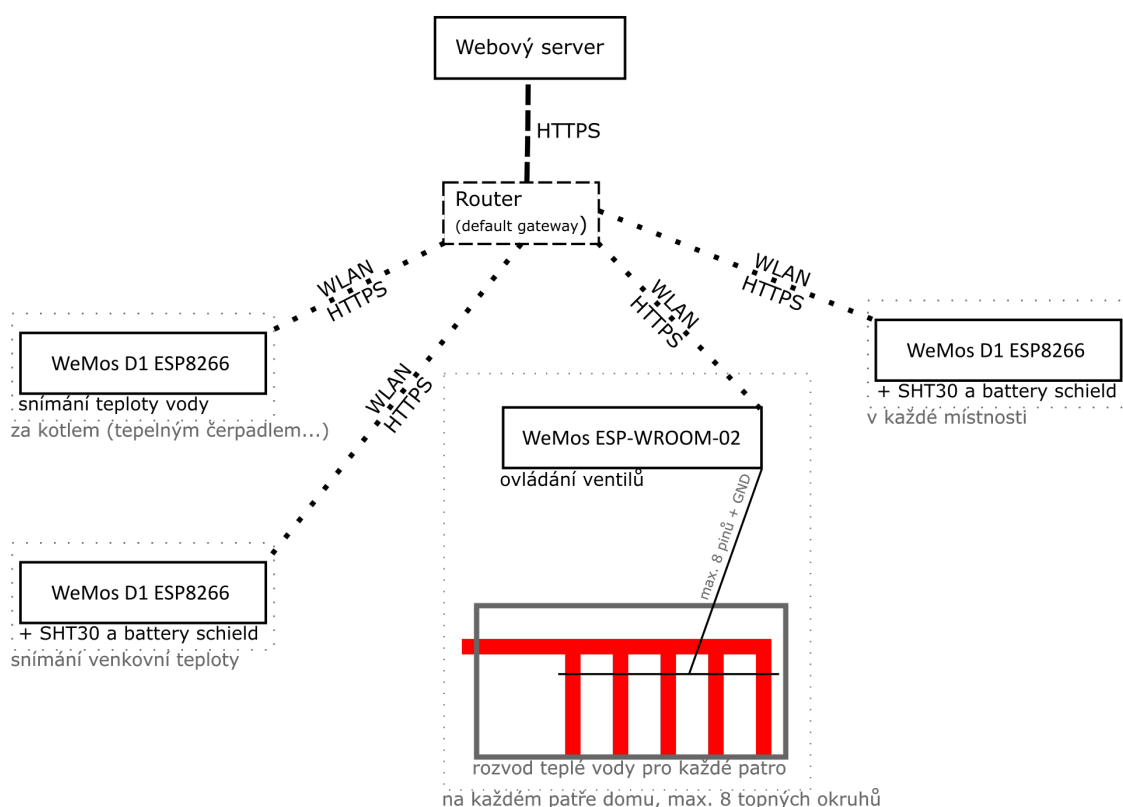
Obr. 1.4: Výběr desky a připojeného portu v programu Arduino IDE

2 Návrh systému a popis jeho částí

V první podkapitole je popsán návrh systému a je popsána komunikace mezi jednotlivými komponentami. Následuje kapitola s popisem jednotlivých komponent.

2.1 Návrh systému

Systém, se skládá z částí, které vidíme na obrázku 2.1, na němž jsou uvedeny i komunikační protokoly mezi jednotlivými částmi. Plná čára značí kabelové propojení mezi zařízením pro ovládání ventilů a jednotlivými ventily topných okruhů. Přerušovaná (tečkovaná) čára pak značí bezdrátové spojení (WiFi) mezi jednotlivými komponentami a směrovačem (routerem), čárkovaná čára pak značí spojení mezi směrovačem a webovou aplikací přes Internet.



Obr. 2.1: Blokové schéma návrhu systému pro vytápění

Stěžejním prvkem celého systému je webové rozhraní pro zobrazování hodnot ze snímačů a ovládání průtoku vody v jednotlivých topných okruzích. Pro zajištění optimálního vytápění v daném objektu jsou umístěny snímače teploty na nezbytná místa jako je např. obývací pokoj, ložnice, chodba, koupelna a další místnosti, které jsou

vytápěny topnými okruhy. Neméně důležitými prvky v systému je snímač venkovní teploty, který snímá aktuální venkovní teplotu, a snímač teploty vody v topném okruhu, který předává webovému rozhraní aktuální hodnotu topné vody.

Části systému můžeme tedy shrnout následovně:

- Webové aplikace – Rozhraní pro sběr dat a ovládání (kap. 3).
- Zařízení pro nastavení průtoku vody jednotlivých topných okruhů (kap. 2.2.1).
- Snímač teploty na přívodu topné vody (kap. 2.2.2).
- Snímač teploty a vlhkosti v místnostech (kap. 2.2.3).
- Snímač venkovní teploty (kap. 2.2.4).

Při navrhování systému bylo plánováno zařízení Raspberry, které by se nacházelo v lokální síti a fungovalo by jako prostředník mezi jednotlivými zařízeními v domácnosti a webovým serverem. Z důvodu finanční úspory a spotřeby elektrické energie je zařízení vynecháno. Dále by zařízení ani nemělo zvlášť velký vliv na provoz v síti, respektive na přenos dat mezi domácnostmi a webovým serverem. Nebyly tedy shledány žádné kladné důvody pro ponechání tohoto zařízení ve schématu. Dalším důvodem jeho vynechání byla neexistence navrhovaného systému, který řeší tato práce.

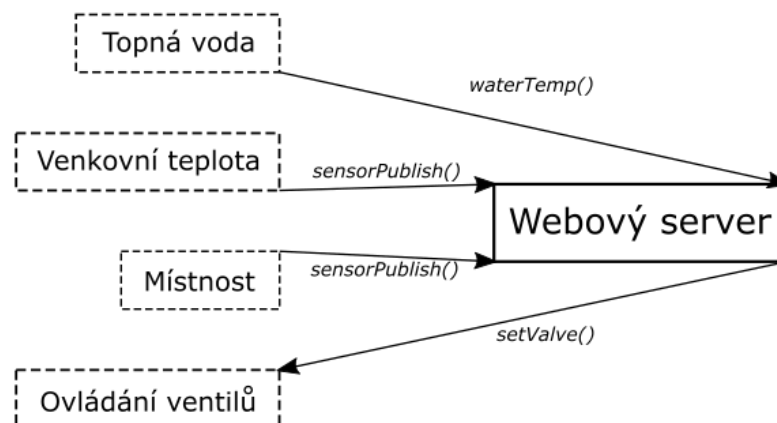
2.1.1 Komunikace mezi jednotlivými částmi

Vzhledem k rozšířenosti modelu TCP/IP je i na tomto modelu založena komunikace mezi jednotlivými částmi systému a webovým rozhraním. Každá část systému komunikuje s webovým serverem prostřednictvím protokolu HTTPS díky URL dotazům. Jednotlivé protokoly dle modelu TCP/IP by se jednotlivé komunikační protokoly v jednotlivých vrstvách modelu daly popsat následovně:

- Aplikační vrstva – HTTPS (TLS)
- Transportní vrstva – TCP
- Síťová vrstva – IP (IPv4/v6)
- Vrstva síťového rozhraní – WiFi/Ethernet

Protokol na aplikační vrstvě je vybrán na základě námi vybrané komunikace mezi zařízeními. V nižších vrstvách modelu TCP/IP jsou dané protokoly vybrány automaticky dle vhodnosti k dané technice přenosu.

Na obrázku 2.2 jsou uvedeny velmi zjednodušeně jednotlivé komponenty systému a metody kontroléru `app/Http/Controllers/HomeHeatingController.php` webové aplikace, prostřednictvím kterých jednotlivé komponenty komunikují s webovou aplikací. Uvedené metody třídy (kontroléru) `HomeHeatingController` nejprve ověří oprávněnost volání této metody přes HTTPS dotaz kontrolou klíče, který je nastaven v souboru `config/custom.php`. Po ověření klíče již následuje samotný cyklus dané metody.



Obr. 2.2: Přehled metod kontroléru webové aplikace

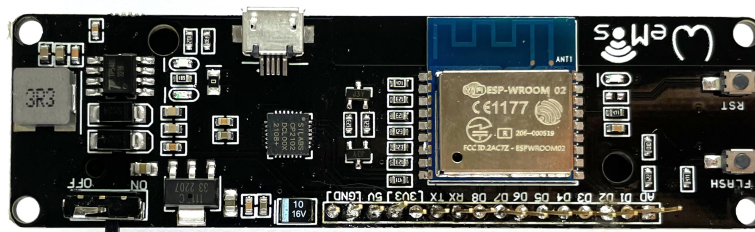
2.2 Dílčí části systému a jejich popis

V následující části bude popsány spíše hardwarové vlastnosti jednotlivých částí systému pro automatizované řízení vytápění. Webovému rozhraní je vzhledem k jeho obsáhlosti věnována kapitola 3.

2.2.1 Ovládání průtoku vody

Ovládání průtoku teplé vody je řešeno zařízením ESP-WROOM-02, které bude napájeno baterií Panasonic NCR18650B. Zařízení je vyfoceno na obrázku 2.3.

V práci je dále řešena výdrž baterie zařízení, která je dále popsána v kapitole 4.1.



Obr. 2.3: Zařízení pro ovládání průtoku vody ESP-WROOM-02

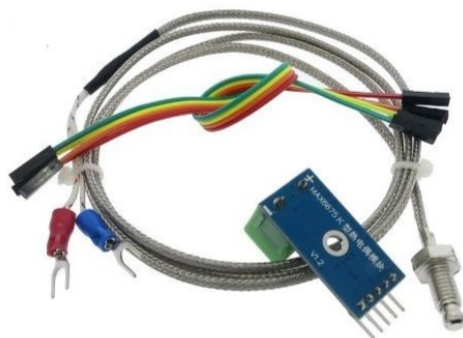
Ovládání jednotlivých ventilů bude řešeno kabelovým připojením ventilů (obr. 2.4) do portů D1-D8 (viz kapitolu 1.2.1). Kabely s černou izolací vedoucí z ventilu je nutné spojit např. přes můstek tak, aby je bylo možné následně jedním kabelem zapojit do portu „GND“ (ground – zem) na desce. Zařízení pak dle údaje z algoritmu z webové aplikace nastaví ventil do požadované hodnoty, viz kapitolu 3.3.



Obr. 2.4: Ventil pro nastavení průtoku vody

2.2.2 Jednotka pro snímání teploty na přívodu teplé vody

Jednotka pro snímání teploty vodu v topném okruhu je řešena zařízením WeMos Lolin D1 mini s termočlánkem typu K a převodníkem MAX6675 k tomuto termočlánku, viz obr. 2.5. Termočlánek bude zašroubován do trubky vedoucí topnou vodu ihned za kotlem (nebo jiným topným prvkem) a zapojen do převodníku MAX6675, který již předává potřebná data do zařízení WeMos Lolin D1, které je pak odesílá HTTPS požadavkem webové aplikaci. Zařízení je opět napájeno akumulátorem GeB LiPol Baterie 603048 900mAh přes WeMos Battery Shield.



Obr. 2.5: Termočlánek typu K s převodníkem MAX6675

2.2.3 Snímač teploty v dané místnosti

Důležitá část systému nacházející se v každé místnosti, ve které je nasazeno automatizované řízení vytápění. Skládá se z následujících částí:

- WeMos Lolin D1 mini
- WeMos Battery Shield
- GeB LiPol Baterie 603048 900mAh
- WeMos SHT30 Shield
- Solární panel 6V 1W

Výše uvedené součásti vidíme na obrázku 2.6, přičemž snímač teploty není přímo spojen jako je tomu u zařízení Lolin D1 a jeho battery shieldem, ale vzhledem k tomu, že při snímání teploty byl snímač teploty SHT30 ovlivněn teplotou zařízení Lolin D1 mini, byl snímač teploty k zařízení Lolin D1 mini připojen tzv. DuPont kabely.



Obr. 2.6: Zařízení pro snímání teploty v místnosti Lolin D1

2.2.4 Snímač venkovní teploty

Poslední součástí tvořící systém automatizovaného vytápění je snímač venkovní teploty. Snímač poskytuje webové aplikace aktuální venkovní teplotu. Jeho realizace je obdobná jako v předchozí kapitole, tedy pomocí zařízení WeMos Lolin D1, teplotního shieldu SHT30, battery shieldu, solárního panelu a baterie. Celé zařízení by bylo vhodné umístit do pouzdra, které by dosahovalo určité odolnosti vůči povětrnostním podmínkám, ale které by neovlivňovalo měřenou teplotu.

3 Webové rozhraní

Tato část systému je jedna z nejdůležitějších částí pro automatizované vytápění a díky svému konceptu nezáleží na tom, zda je v domě, kde je zřízeno automatizované vytápění, připojení k Internetu s veřejnou IP adresou.

Pokud se v objektu nachází veřejná adresa je možné provozovat webovou aplikaci na např. Raspberry serveru, které by kompletně nahradilo nutnost pronajímat si webový hosting.

Pokud se tedy v objektu nenachází připojení s veřejnou IP adresou a budoucí uživatel bude požadovat přístup k automatizovanému vytápění i z jiných míst než jen z daného objektu (z jiné než lokální sítě), bude potřeba provozovat aplikaci na webovém serveru, kde je realizováno i rozhraní pro nastavení aplikace pro vytápění. Uvedenou možnost řeší i tato bakalářská práce.

Rozhraní pro automatizované řízení vytápění bude tedy provozován na webhostingové službě, ke které se lze díky připojení k internetu připojit odkudkoli. Pro prezentaci a testování byl vybrán hosting webglobe.cz (dříve onebit.cz), který pro danou práci stačí a mj. nabízí i SSL certifikáty zdarma, čehož bude využito při připojení prostřednictvím protokolu zabezpečeného protokolu HTTPS.

Vytvořené webové rozhraní pro ovládání vytápění rodinného domu je dostupné na mém githubu ¹, kde je uveden i postup pro jeho prvotní zprovoznění. Funkční verze webové aplikace je dostupná na doméně sloukapetr.cz ². Aplikace, která vytvořena pomocí PHP frameworku Laravel, je na této doméně dostupná prostřednictvím protokolu HTTPS.

3.1 Architektura MVC a framework Laravel

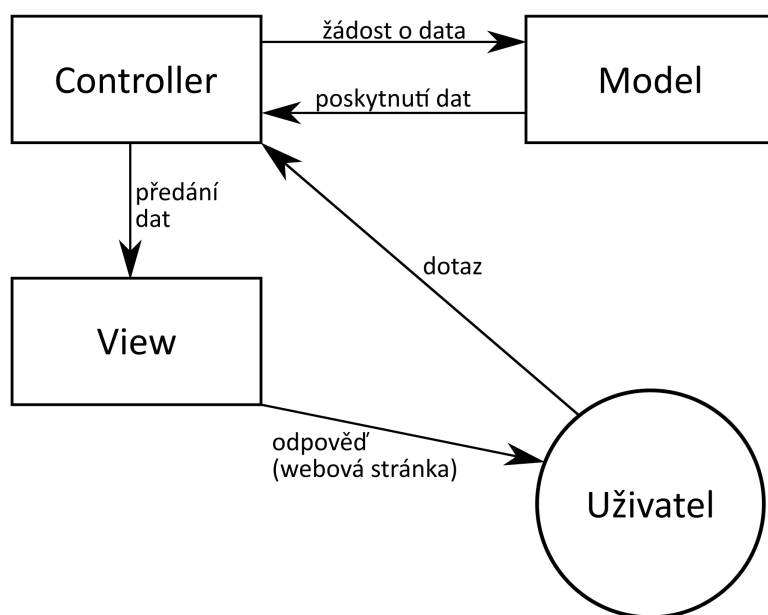
Laravel je framework (jednoduše řečeno softwarová struktura pro podporu a vývoj programů) pro tvorbu webových aplikací postaven na programovacím jazyce PHP a obsahuje přehlednou dokumentaci. Po úvodní instalaci je vývojáři poskytnut tzv. výstupní bod, který zahrnuje základní strukturu Laravelu. Vývojář se tak může soustředit přímo na řešení svého problému. [7]

Jedná se MVC framework, to je architektonický vzor, který umožňuje oddělit logickou část programu od jeho výstupů. MVC (Model, View, Controller - česky model, pohled, kontrolér) architektura funguje tak, že uživatel zasílá svůj dotaz serveru, ten jej předává programu (v tomto případě webové aplikaci) a následně jej aplikace zpracovává pomocí controlleru, ten si pomocí modelu vyžádá potřebná data

¹https://github.com/sloukapetr/bachelors_thesis-web_application

²<https://bachelors-thesis.sloukapetr.cz>

z databáze a následně tato data předá do pohledu. Pohled je zaslán uživateli jako výstup (webová stránka) jeho dotazu. [7]



Obr. 3.1: Architektura MVC

Laravel je možné rozšířit různými balíčky (packages) pro rozšíření svých funkcí jako např. balíček pro kompletní správu uživatelů, jejich registraci a přihlašování (Fortify), balíček pro implementaci přihlašování pomocí sociálních sítí (Socialite), rozhraní pro správu předplatného (Cashier) a mnoho dalších užitečných komponent. [7]

Při následné instalaci webové aplikace jsou některé soubory zejména důležité, je nutné se seznámit s obsahem elektronické přílohy v kapitole A.1. Zaměřte se na i obsah složky `config` a v ní obsažené soubory a dále na soubor `.env` v kořenové složce webové aplikace.

Při nastavování konfiguračních souborů je možné aktivovat dvoufázové ověření, které je realizováno prostřednictvím mobilní aplikace **Google Authenticator**. Tento způsob ověření přidává webové aplikaci možnost lépe zabezpečit přístup uživatelů do aplikace. [4]

Webová aplikace využívá databázi MySQL pro uchovávání dat. Framework Laravel komunikuje s databází prostřednictvím svých modelů, které využívají rozšíření PDO (PHP Data Objects), což je rozhraní pro přístup k databázím v programovacím jazyce PHP. [7]

3.2 Nastavení webového rozhraní

Při instalaci celého systému je nejdříve nutné zajistit správné fungování webového serveru nebo webhostingu. Následně je pak důležité zaměřit se na samotný koncept vytápění podle kapitoly 5. V příloze A.6 je pak ukázka jednotlivých stránek webové aplikace.

3.2.1 Konfigurace webového serveru

Pro správné fungování webové aplikace je nutné dle dokumentace Laravel nastavit webový server Apache. V nastavení webového serveru se dostaneme do složky `apache\conf\extra`, kde vyhledáme soubor `httpd-vhosts.conf`, v tomto souboru je nutné nastavit položku `DocumentRoot` tak, aby cesta vedla do složky `public` v Laravel projektu.

Výpis 3.1: Vzor nastavení Apache serveru

```
<VirtualHost *:80>
    DocumentRoot "C:/XAMPP/htdocs/web_app/public"
    ServerName web-app.cz
</VirtualHost>
```

V případě využití nějakého webového hostingu je nutné zjistit, zda umožňují individuální úpravu výše uvedeného nastavení nebo zdali je toho možné docílit souborem `.htaccess`, který by se musel nacházet v kořenové složce webové aplikace a nikoli ve složce `public`. Individuální úpravu kořenového adresáře umožňuje hosting `webglobe.cz`.

Výpis 3.2: Vzor nastavení kořenové adresáře

```
<IfModule mod_rewrite.c>
    RewriteEngine On
    RewriteRule ^(.*)$ public/$1 [L]
</IfModule>
```

Pro správné fungování webové aplikace je nutné rozbalit soubory webové aplikace anebo je získat z GitHubu, který je uveden v dřívějších kapitolách. Pro stažení všech balíčků je potřebné nainstalovat program **Composer**³ a pokud bychom chtěli zasahovat do vzhledu aplikace, je nutné nainstalovat i program **npm**⁴. Pro stažení všech balíčků stačí již jen zadat příkaz z příkazové řádky ze složky se soubory k webové aplikaci `composer update`. Aby webová aplikace správně fungovala je také nutné

³<https://getcomposer.org/>

⁴<https://www.npmjs.com/>

soubor `.env.example` přejmenovat na `.env` a nastavit jednotlivé proměnné dle použitého webového serveru a domény. Rovněž je doporučeno nastavit SMTP server pro odesílání e-mailů, aby bylo možné zapnout možnost pro obnovu hesla uživatelů, a vytvořit klíč aplikace příkazem `php artisan key generate`. Nyní je webová aplikace připravena k nahrání na webový server.

Konfigurace přihlašování

O přihlašování se uživatelů a jejich registraci se stará balíček Fortify pro framework Laravel. Tento balíček je tzv. backendovou nástavbou pro Laravel a implementuje formou cest (routes) a kontrolérů funkce frameworku Laravel jako je přihlášení uživatele, jeho registraci, obnovu hesla, emailové ověření a dále například dvoufázové ověřování uživatele.

Fortify je v základu nakonfigurován tak, jak je vidět ve výpisu 3.3, který se nachází v souborech webové aplikace `config/fortify.php`, kde je povolena pouze registrace uživatelů a změna hesla. Ověření emailové adresy je zakázáno z toho důvodu, že by bylo potřeba konfigurovat emailový server. Po nastavení např. emailového serveru je možné povolení těchto funkcí pouhým odkomentováním řádku (odstranění obou lomítek).

Po prvotní instalaci webového serveru je doporučeno vypnout registraci uživatelů a povolit dvoufázové ověření (na toto nastavení upozorňuje sama webová aplikace správce. Přidávat nové uživatele může pak pouze správce systému, kterému je umožněno jednotlivým uživatelům obnovit (nastavit) heslo, změnit jejich údaje a udělovat oprávnění pro jednotlivé místnosti.

Výpis 3.3: Konfigurační soubor přihlašování

```
'features' => [
    Features::registration(),
    // Features::resetPasswords(),
    // Features::emailVerification(),
    //Features::updateProfileInformation(),
    Features::updatePasswords(),
    //Features::twoFactorAuthentication([
    //     'confirm' => true,
    //     'confirmPassword' => true,
    //     'window' => 0,
    // ]),
],
```

Konfigurační soubor

Ve výpisu 3.4 jsou vidět jednotlivé řádky pole (array), které je v aplikaci voláno příkazem `custom('custom.*')`, kde místo symbolu `*` je uveden klíč pro daný záznam (např. „`app_key`“). Tyto hodnoty je možné upravit, zejména je pak důležité vytvořit nebo vygenerovat⁵ nový klíč pro aplikaci. Hodnotu pro požadovanou teplotu vody v topném okruhu („`required_water_temp`“) je možné změnit, avšak aktuálně nastavená hodnota je dle doporučení [5] v hodnotě pro vytápění v jarních a podzimních měsících.

Výpis 3.4: Konfigurační soubor přihlašování

```
return [
    'app_key' => 'XYgXUjEmS1ZjwCZ5fMDQBSc3fapU07ji',
    'required_water_temp' => '32',
    'weather_api_address' => 'Brno, Czech',
    'weather_api_key' => '8CE9QGZ825NL9KF2G4V59SS3R',
    'show_floors' => (bool) false,
    'show_admin_messages' => (bool) false,
    'sensor_last_update' => 900, // limit,
];
```

3.3 Ovládání ventilů

Zařízení pro ovládání ventilů jednotlivých topných okruhů odesílá ve výchozím nastavení každých 15 minut HTTPS dotaz, kterým zjistí požadovanou hodnotu pro otevření nebo uzavření ventilu v topném okruhu.

Hodnota ventilů se mění vždy při zapsání nové hodnoty teploty v místnosti nebo při změně požadované hodnoty v místnosti. Tato proměnná může nabývat hodnot v rozsahu od 0 do 1 včetně s přesností na 2 desetinná místa. Nastavenou hodnotu ventilů ve webové aplikaci vidí pouze správce systému.

Výpočet hodnoty pro nastavení průtoku

O výpočet hodnot otevření ventilů se stará funkce, která se nachází ve webové aplikaci v souboru `App/Http/Controllers/HomeHeatingController.php`. Tuto funkci je možné vidět ve výpisu 3.5.

Výpis 3.5: Vzor nastavení kořenové adresáře

```
if ($room->sensors()->count() > 0) {
```

⁵<https://randomkeygen.com/>

```

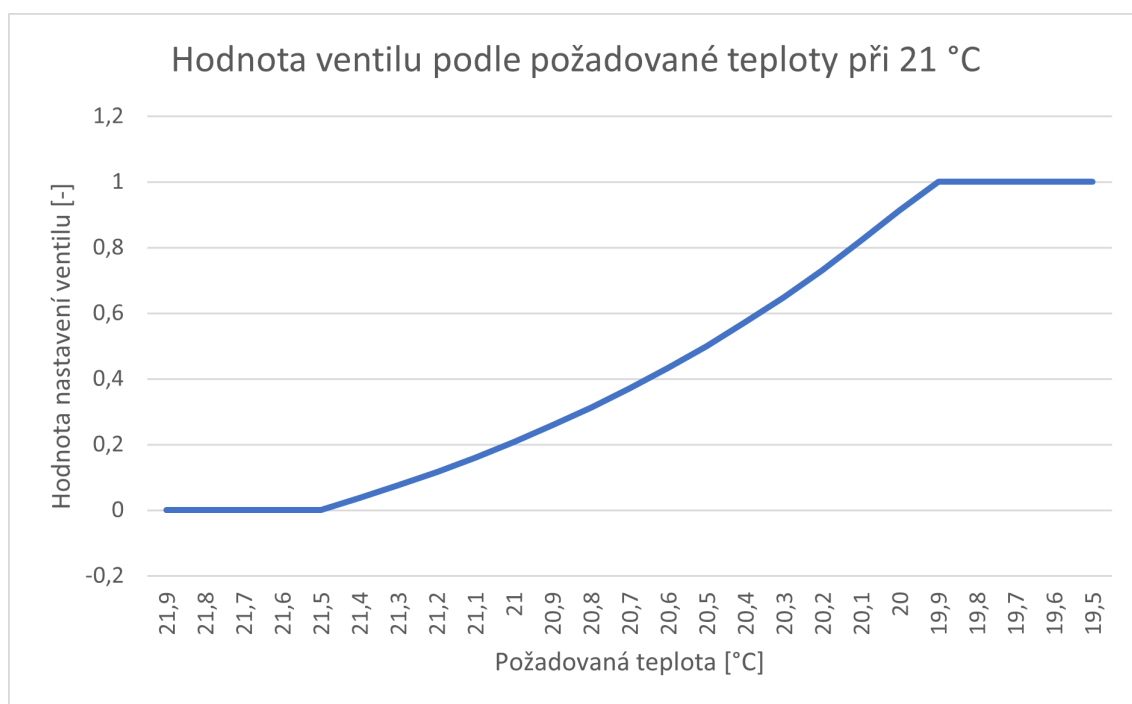
    $actualTemp = $room->sensors()->latest()->first()->temp;
    $waterTemp = Setting::where('name', 'water_temp')->first
        ()->value;
    $requiredWaterTemp = Setting::where('name', '
        required_water_temp')->first()->value;
    $tempDiff = $room->goal_temp - $actualTemp;
    if (($waterTemp > $requiredWaterTemp) AND ($room->
        goal_temp > 0)) {
        if ($tempDiff > 1) {
            $valveValue = 1;
        } elseif ($tempDiff < - 0.5) {
            $valveValue = 0;
        } else {
            $valveValue = round(pow(2, ($tempDiff - 0.5)) -
                0.5, 2);
        }
        $room->update([
            'valve_value' => $valveValue,
        ]);
    }
}

```

Lze ji jednoduše popsat tak, že pokud je požadovaná teplota vyšší než „1“, je výstupní hodnota nastavena rovněž do hodnoty „1“. Pokud je požadovaná hodnota menší o „-0,5“, je výstupní hodnota nastavena na hodnotu „0“. Ve všech ostatních případech je využita rovnice 3.1, kdy za x je dosazen rozdíl mezi požadovanou teplotou a aktuální teplotou v místnosti.

$$x = 2^{x-0,5} - 0,5 \quad (3.1)$$

Výše uvedenou a popsanou funkci lze vidět na grafu 3.2 při požadované teplotě 21 °C. Zařízení pro ovládání ventilů pak odebírá tuto hodnotu a dle svého vlastního nastavení otevírá daný ventil topného okruhu, nebo jej uzavře. Výchozí referenční hodnotou pro jeho funkci je „0“, tzn. že pokud je hodnota menší nebo rovna referenční hodnotě, ventil se uzavře, pokud je větší, ventil bude otevřený. Referenční hodnota se nastavuje v programu pro zařízení ESP-WROOM-02.



Obr. 3.2: Ukázka hodnot pro otevření a zavření ventilu

4 Analýza a testování systému

V kapitole je popsáno testování systému, které zahrnuje primárně výdrž akumulátorů jednotlivých komponent a rozebírá komunikaci mezi částmi systému a jejich zabezpečení.

4.1 Výdrž akumulátorů

V kapitole je popsána a vypočtena z reálných měření výdrž jednotlivých zařízení, která jsou nutná pro správné fungování automatizovaného řízení vytápění rodinného domu.

4.1.1 Výdrž akumulátoru snímačů teploty

Do snímače pro měření teploty v místnosti byl v prvním kroku nahrán finální software pro měření teploty v místnost, respektive jeho upravená verze, kdy jeho programová část trvala 8 sekund a dále bylo zařízení v režimu „deep-sleep“ dalších 15 sekund. Testování a měření zařízení proběhlo bez zapojeného solárního panelu a ve třech krocích, kdy byl akumulátor nabit tak, že vykazoval v nezapojeném stavu napětí 3,4 V, 3,0 V a 2,8 V.

Zařízení Wemos ESP8266 D1 mini bylo spojeno s rozšířením pro akumulátor (battery shield) a se senzorem teploty a vlhkosti SHT30. Dále byl k zařízení připojen akumulátor.

Měření proudů probíhalo multimetrem „UNI-T UT131A“ a klešťovým multimetrem „UNI-T UT210E“, který byl využit pouze pro měření proudů při provozu zařízení, jelikož není určen pro měření stejnosměrného proudu v řádech miliampér a mikroampér. Měření bylo nahráno na záznamové zařízení a z jeho přehrávání byly zapisovány hodnoty z měřících přístrojů. Režim „deep-sleep“ nevykazoval žádné výchylky při měření a proud nabýval hodnot 248 uA, naopak měření zařízení při provozu vykazovalo výchylky v řádu 10 mA, a tak byly zaznamenávány hodnoty každou sekundu po celou dobu běhu programu. Zaznamenáno tak bylo ve 3 cyklech po 10 měření celkem 30 hodnot.

Z těchto 30 hodnot byla mediánem vypočítána hodnota proudu při provozu zařízení, která činí 67,5 mA. Z uvedených hodnot proudů a známé doby trvání cyklu, kdy program na zařízení běží 8 sekund a režim „deep-sleep“ trvá 8 minut (900 sekund) lze vypočítat, že za hodnu je spotřebováno 0,84 mAh, za celý den zařízení spotřebuje 20,17 mAh. Při těchto hodnotách a znalosti kapacity akumulátoru, která činí 850 mAh vypočítáme, že zařízení vydží při plném nabití fungovat 42 dnů.

Pokud bychom chtěli výdrž zařízení zvýšit, museli bychom k zařízení přidat solární článek, např. Solární panel 6V 1W. Uvedený solární článek by měl dle údajů výrobce být schopen dodávat až 167 mAh. Při proudu, který článek dodával do zařízení jsem však zjistil, že jeho maximální dobíjecí proud byl ve standardně osvětlené místnosti přibližně 1 mAh. Za předpokladu, že takové denní světlo je v místnosti třetinu dne, tj. 8 hodin, dodá do zařízení 8 mAh, celková spotřeba za jeden den tak klesne na 12,17 mAh. Toto zařízení poté vydrží fungovat minimálně 69 dnů, než bude potřeba jej nabít.

Stejnou hardwarovou skladbou a obdobným programovým vybavením bude je řešen i snímač venkovní teploty a snímač teploty vody v topném okruhu, a tak nebylo nutné provádět stejná měření.

4.1.2 Výdrž akumulátoru pro zařízení nastavující průtok ventilů

Testování zařízení Wemos ESP-WROOM-02 pro ovládání průtoku ventilů bylo provedeno na upraveném programu tak, že byla zkrácena doba trvání „deep-sleep“ módu na 10 sekund, které dostačovaly pro změření odběru proudu z baterie. Následoval samotný běh programu a měření proudů, které bylo zaznamenáno na video, ze kterého by následně po celou dobu běhu programu odečítány hodnoty po 0,5 sekundách.

Program pro nastavování teploty byl upraven tak, že probíhalo všech osm URL dotazů, kdy v databázi byly nastaveny postupně hodnoty ventilů od 0 do 1 (0; 0,10 ; 0,25; 0,40; 0,50; 0,75; 0,90; 1), přičemž cyklus probíhal tak, že pin D1 nastavil do požadované hodnoty na 10 vteřin, poté jej vynuloval a vyžádal si další hodnotu.

Celkem proběhly 3 měření po 10 cyklech programu pro napětí akumulátoru na 2,8 V, 3,0 V a 3,3 V. Měření bylo nahráno na záznamové zařízení především z důvodu zachycení přesného času běhu samotného programu. Záznam byl následně postupně přehráván a po 0,5 sekundách byl zaznamenán aktuální odběr proudu z akumulátoru. Použité měřicí přístroje byly stejné jako v předešlé kapitole, tedy multimetr „UNI-T UT131A“ a klešťový multimetr „UNI-T UT210E“.

Výpočtem bylo zjištěno, že zařízení v době svého provozu odebírá průměrně 61,9 mA a v době „deep-sleep“ klesne jeho spotřeba na 0,25 mA. Dále bylo zjištěno, že doba, po kterou zařízení běží je 80 sekund a dalších 900 sekund (15 minut) stráví zařízení v režimu „deep-sleep“. Z uvedených hodnot, že spotřeba energie za jednu hodinu je 5,28 mAh, tj. 126,8 mAh za den. Při použitém typu akumulátoru¹ vydrží zařízení 26 dnů.

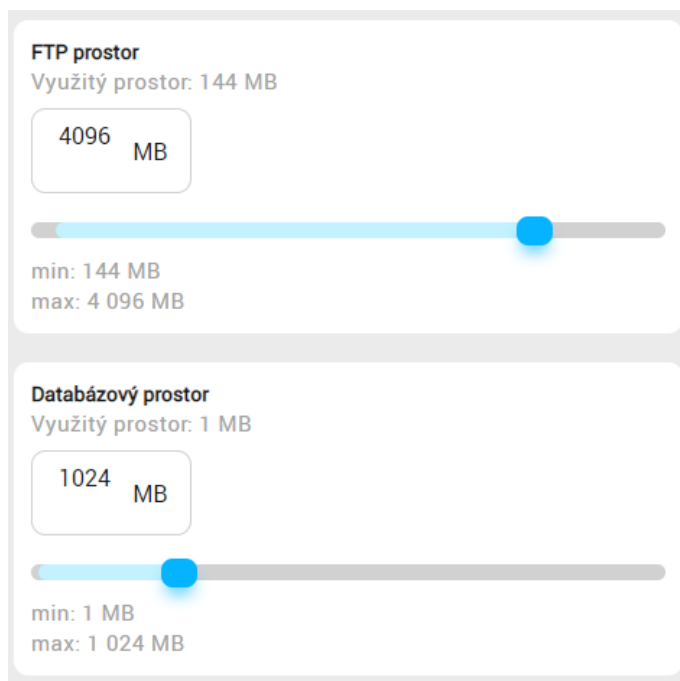
Z uvedeného je na každém případném uživateli, aby zvážil možnost připojení zařízení k elektrické síti např. USB kabelem, aby nebylo nutné zařízení hlídat. Využití

¹Panasonic NCR18650B 3350mAh 18650 Li-Ion Baterie

solárního článku v tomto případě vzhledem k umístění daného zařízení je zbytečné, jelikož se počítá s využitím např. v technické místnosti domu, která nemusí být vždy dobře osvětlena.

4.2 Kapacita záznamů databáze

V kapitole je s ohledem na praktické záznamy webové aplikace vypočítána předpokládaná kapacita databáze, resp. její náročnost na kapacitu pevného disku, který využívá databáze. Nemusí se však jednat o fyzickou kapacitu pevného disku, ale v případě využití webhostingových služeb může jít o poskytovanou kapacitu úložiště apod. V případě webhostingu webglobe.cz se jedná o nastavitelnou kapacitu úložiště, kterou lze dynamicky rozdělovat mezi FTP úložiště, databázový prostor a prostor pro e-mailový prostor, viz obr. 4.1.



Obr. 4.1: Přerozdělení kapacity úložiště webhostingu webglobe.cz

Prostřednictvím frameworku Laravel a jeho funkcím pro práci s databází byla testována její kapacity z pohledu náročnosti na úložiště. Databáze byla v několika fázích nejdříve pouze vytvořena a ponechána prázdná. Následně byla plněna daty pro 4, 8 a 16 místností, všechny 3 možnosti počítaly i se senzorem venkovní teploty. V cyklech byla databáze plněna daty, kdy nejdříve se jednalo o 200 záznamů na místnost, nakonec byla databáze plněna 300000 záznamy na jednu místnost, což

bylo přibližně 4800000 záznamů. Po každém cyklu byla databáze vyexportována a vyčištěna před dalším plněním.

Testem tak bylo zjištěno, že jeden záznam v databázi zabírá přibližně 81 bajtů. Pokud bychom měli senzory pro 8 místností, které by publikovaly měřené hodnoty každých 5 minut, za rok by bylo obsazeno 72 MB, pokud by publikování probíhalo každých 15 minut, obsazeno by bylo 24 MB paměti databáze. Pro 16 místností by pak platilo, že při publikování záznamů každých 15 minut by za rok bylo obsazeno 48 MB, v tomto případě by pak databáze s kapacitou 1 GB vydržela bez zásahu více než 21 let.

5 Prvotní spuštění a instalace systému

Kapitola popisuje instalaci a kompletní spuštění celého systému. V první podkapitole je obecně popsán koncept a v následujících kapitolách je popis instalace a konfigurace dále jednotlivých komponent. Instalaci a konfiguraci webové aplikace je věnována podkapitola 3.2.1.

5.1 Koncept instalace systému

Dále je popsán koncept pro instalaci automatizovaného řízeného vytápění do domu a instalace jednotlivých komponent, webové aplikaci je věnována samostatná kapitola 3. V úvodu je třeba se zamyslet a podle cílového domu či bytu, kde má být automatizované vytápění instalováno si vytvořit například tabulku nebo seznam místností, u kterých bude uvedeno i podlaží, pokud jsou.

Po vytvoření nějaké pomocné tabulky např. v programu MS Excel, viz obr. 5.1, je doporučeno dle kapitoly 3.2.1 provést instalaci a prvotní spuštění webové aplikace a jednotlivé místnosti v ní vytvořit. Po jejich vytvoření je dobré si do pomocné tabulky poznačit identifikační čísla (ID) jednotlivých místností, které nám identifikují jednotlivé snímače teploty. Rovněž je výhodné si vyznačit podlaží, ve které

	A	B	C	D
1	ID	Místnost	Podlaží	Piny
2	5	Obývací pokoj	1	D1
3	2	Kuchyně	1	D2
4	4	Koupelna	1	D3
5	1	Toaleta 1	1	D4
6	3	Pokoj	1	D5
7	7	Ložnice	2	D1
8	6	Toaleta 2	2	D2

Obr. 5.1: Vzor seznamu místností s identifikačními čísly a přiřazenými výstupy

5.2 Konfigurace jednotlivých komponent

Při konfiguraci komponent Wemos Lolin D1 mini a Wemos ESP-WROOM-02 je úvodní část programu zpravidla shodná. Je tedy nutné si zjistit po instalaci webové aplikace a zprovoznění spojení přes HTTPS (implementace SSL klíče) klíč certifikátu pro šifrovanou komunikaci, viz kapitolu 5.2.

Pro každé zařízení je nutné nastavit parametry uvedené ve výpisu kódu 5.1. Je tedy potřeba si vytvořit nebo nechat vygenerovat nějaký např. 32 znaků dlouhý „klíč“, který je nutný pro ověření oprávnění pro volání jednotlivých funkcí webové aplikace, a tento klíč vložit jako parametr APP_KEY. Dále je nutné upravit dle vlastní doménové adresy pole WEB_URL.

Podle konfigurace vlastní domácí WiFi sítě je dále nutné nastavit parametry WIFI_SSID (název WiFi sítě) a WIFI_PASSWORD (heslo pro připojení k WiFi síti).

Parametry PUBLISH_RATE a SLEEP_MODE definují pauzu mezi jednotlivými smyčkami programu, respektive jak pravidelně bude program vykonávat svoji funkci. A parametr DEBUG typu bool (pravda, lež) pak nastavuje možnost výpisu přes sériový port. V případě, že budeme chtít číst data přes sériový port zapojeného do počítače do USB portu, musíme nainstalovat potřebný převodník¹. Před samotným spuštěním potřebné aplikace, např. Tera Term², ještě upravíme rychlost posílání dat, resp. jejich příjem, tímto portem na hodnotu 115200, aby byla data čitelná.

Výpis 5.1: Příklad úvodní konfigurace pro komponenty

```
// Root certificate
const char IRG_Root_X1 [] PROGMEM = R"CERT(
-----BEGIN_CERTIFICATE-----
-----your_certificate-----
-----END_CERTIFICATE-----
)CERT";

// Global setting
#define APP_KEY          "app_key"    //Our key from web
    application
#define WEB_URL          "https://domain.com" //URL include
    protocol without ending slash
// WiFi setup
#define WIFI_SSID        "SSID"        //your wifi name - SSID
#define WIFI_PASSWORD    "password"    //your WiFi password
// Sleep and loop setup
#define PUBLISH_RATE     10*60          // publishing rate in
    seconds (minute*60), def.: 10*60 - 10 minut
#define SLEEP_MODE       true           // deep sleep then reset
    or idling (DO must be connected to RST)
// Debug true/false
#define DEBUG            false          // debug to serial port
```

¹<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers>

²<https://ttssh2.osdn.jp/index.html.en>

Ve výpisu 5.2, který pochází z programu pro ovládání ventilů topných okruhů je v poli s názvem `pinOutput` postupně definován výstup za výstupem od pinu D1 až po pin D2, do tohoto pole je doporučeno nezasahovat vzhledem ke správnému fungování celého systému. V poli s názvem `roomId` je pak definováno vždy identifikační číslo dané místnosti, které se vyplňuje podle tabulky uvedené na obr. 5.1 postupně tak, jak požadujeme přiřadit jednotlivé výstupy k topným okruhům.

Výpis 5.2: Přiřazení pinů v zařízení ESP WROOM 02

```
// Define outputs pins to valve - in order output D1-D8
int pinOutput[8] = {5, 4, 0, 2, 14, 12, 13, 15}; // DO NOT
TOUCH THIS
int roomId[8] = {0, 0, 0, 0, 0, 0, 0, 0}; //change it by room
to pin in order D1 - D8; 0 (zero) will be ignored
```

Zjištění klíče certifikátu

Pro zjištění klíče certifikátu pro šifrovanou komunikaci stačí danou doménu načíst ve webovém prohlížeči. Pro následující postup je využít prohlížeč Chrome od společnosti Google.

Po načtení dané domény je nutné kliknout na ikonu zámku vlevo od pole s doménou následně na možnost „Připojení je zabezpečené“ a následně na „Certifikát je platný“, viz obr. A.1 a A.2. Po otevření okna (viz obr. A.3) je nutné vybrat **ISRG Root X1** a tuto možnost následně vyexportovat (soubor uložit). Po uložení otevřeme soubor v jakémkoli textovém editoru a celý klíč (vč. řádků **BEGIN CERTIFICATE** a **END CERTIFICATE**) zkopírujeme. Tento certifikát nakopírujeme do každé varianty programu pro komponenty, viz např. výpis programu 5.1.

Potencionální chyba v knihovně

Z důvody potřeby kompilace upraveného kódu programu pro senzory teploty v místnostech a venku je potřeba upravit kód importované knihovny **WEMOS_SHT3x**, kde se v souboru **src/WEMOS_SHT3X.cpp** na 30. řádku vyskytuje chyba, která je uvedena ve výpisu 5.3 (zakomentovaný druhý řádek). Pod tímto výrazem je uveden správný zápis. Uvedená chyba je v příložených souborech již opravena.

Výpis 5.3: Oprava chyby v importované knihovně

```
// Request 6 bytes of data
// Wire.requestFrom(_address, 6); - chybný řádek
Wire.requestFrom(_address, (uint8_t) 6);
```


Závěr

V úvodu bakalářské práce je teoreticky popsán model TCP/IP a jednotlivé protokoly pro dané vrstvy modelu. Následuje podkapitola popisující čipy ESP8266 a ESP32, je zde rovněž zmíněno programování jednotek založených na těchto čipech. Jsou zde uvedeny i jednotlivé technické parametry čipů a pro vývojovou desku ESP-WROOM-02 popisuje i zjištěné přiřazení pinů čipu k portům na vývojové desce. Druhá kapitola bakalářské práce je věnována návrhu systému pro automatizované řízení vytápění rodinného domu, popisuje jednotlivé části systému a komunikaci mezi nimi. Jeden odstavec je věnován důvodům, proč bylo vynecháno zařízení Raspberry, které bylo původně plánováno pro systém, další odstavec se pak věnuje třídě, se kterou pomocí HTTPS dotazů komunikují jednotlivá zařízení. Tato třída nacházející se ve webové aplikaci pracuje s přijatými daty a předává je dále jiným třídám webové aplikace nebo je ukládá do databáze. V dalších podkapitolách jsou popsány podrobněji jednotlivé části systému společně s jejich obrázky. Třetí kapitola práce je věnována webové aplikaci. Dílčí části kapitoly jsou věnovány architektuře MVC a frameworku Laravel, na kterém je postavena webová aplikace. Práce dále popisuje konfiguraci webového serveru a konfiguraci přihlašování do aplikace. V další podkapitole je popsán algoritmus, který nastavuje hodnoty ventilů, které odebírá zařízení ESP-WROOM-02. Čtvrtá kapitola popisuje měření výdrže akumulátorů jednotlivých zařízení. Měřením bylo zjištěno, že v při dodržení periody pro komunikaci mezi zařízeními a webovou aplikací vydrží zařízení pro snímání teploty až 60 dnů, zařízení pro nastavení hodnot ventilů pouze 21 dnů. V další části kapitoly je popsána kapacita databáze a její paměťová náročnost. Testováním webové aplikace, resp. databáze, jsem došel k závěru, že databáze s kapacitou 1 GB by vydrželo více než 21 let bez zásahu. Poslední kapitola práce popisuje koncept instalace a čtenář je tak proveden i instalací celého systému. Dále je zde popsán postup nastavení programu jednotlivých komponent.

Literatura

- [1] JEŘÁBEK, Jan. *Komunikační technologie*. verze 2022. Brno: Vysoké učení technické v Brně, 2014. ISBN 978-80-214-4713-4.
- [2] F. WIJITRISNANTO, SUHARDI AND P. YUSTIANTO. HTTPS Contribution in Web Application Security: A Systematic Literature Review. *2020 International Conference on Information Technology Systems and Innovation (ICITSI)*. 2020, 347–356. Dostupné z: 10.1109/ICITSI50517.2020.9264971.
- [3] MUDRÁK, David. Schéma zapouzdření aplikačních dat na vrstvách TCP/IP [vektorová grafika]. 2008 [cit. 22. 11. 2022]. Dostupné z: <https://upload.wikimedia.org/wikipedia/commons/c/c0/Tcpip_zapouzdeni.svg>.
- [4] M. PAPATHANASAKI, L. MAGLARAS, I. KANTZAVELOU Security Revisited: Honeytokens meet Google Authenticator. *2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*. 2022, 1–8. Dostupné z: 10.1109/SEEDA-CECNSM57760.2022.9932907.
- [5] FUKS, Miroslav. Topná voda — levná kapalina, která se může prodrazit. *Topná voda* [online]. 2021 [cit. 2023-04-15]. Dostupné z: <https://www.av-equen.cz/topna-voda-levna-kapalina-ktera-se-muze-prodrazit/>
- [6] Arduino Reference. *Arduino* [online]. Arduino, 2023 [cit. 2023-04-18]. Dostupné z: <https://www.arduino.cc/reference/en/>
- [7] Laravel Documentation - 10.x. *Laravel* [online]. 2023 [cit. 2023-05-20]. Dostupné z: <https://laravel.com/docs/10.x>
- [8] *ESP8266 Arduino Core Documentation* [online]. 2023 [cit. 2023-05-16]. Dostupné z: <https://arduino-esp8266.readthedocs.io/en/latest/>
- [9] ESPRESSIF SYSTEMS. *ESP-WROOM-02 Datasheet*. Verze 3.4. 2022, 22 s.

A Přílohy

První částí příloh tvoří popis obsahu elektrické přílohy bakalářské práce, druhou část pak tvoří seznam všech komponent, následuje příloha s grafickým zobrazením

A.1 Obsah elektronické přílohy

Ve složce pro webovou aplikaci nejsou uvedeny všechny soubory a složky, ale jsou vypsány a okomentovány důležité soubory a složky pro instalaci webové aplikace. Složka „Arduino IDE“ obsahuje všechny vytvořené programy pro jednotlivé komponenty, složka „Webová aplikace“ obsahuje zdrojové soubory pro spuštění webové aplikace a nezařazenými soubory jsou pak vzorové soubory pro databázi.

```
/.....kořenový adresář příloženého archivu
├── Arduino IDE/.....složka s jednotlivými programy pro komponenty
│   ├── ESP-WROOM-02/.....složka s programem pro dané zařízení
│   │   └── valves_controller/.....složka programem kontrolér ventilů
│   ├── libraries/.....složka s použitými knihovnami pro Arduino IDE
│   └── Lolin D1 mini/.....složka s jednotlivými programy pro dané zařízení
│       ├── temp_hum_sensor/.....složka programem pro teplotní senzor
│       └── water_temp_sensor/.....složka programem pro snímač teploty vody
├── Webová aplikace.....složka s webovou aplikací
│   ├── app/
│   │   ├── Http/
│   │   │   ├── Controllers/.....složka s kontrolérem pro komponenty
│   │   │   ├── Livewire/.....složka kontroléry pro aplikaci
│   │   │   └── Middleware/.....složka s pomocnými funkcemi
│   │   ├── Models/.....složka se soubory pro přístup k databázi
│   │   └── Policies/.....složka se soubory pro úpravu práv
│   ├── config/.....složka s konfiguračními soubory
│   │   ├── app.php.....soubor pro nastavení lokalizace a časového pásma
│   │   ├── custom.php ....složka se souborem pro konfiguraci např. aplikačního klíče
│   │   └── fortify.php .....složka s nastavením přihlašování a registrací
│   ├── database/.....složka se soubory k databázi
│   │   ├── seeders/.....složka s plniči databáze
│   │   │   └── DatabaseSeeder.php .....soubor pro konfiguraci plničů
│   ├── public/.....přístupná složka webové aplikace
│   ├── lang/.....složka s lokalizacemi aplikace
│   │   ├── cs/.....složka se soubory překládající aplikaci do českého jazyka
│   │   └── cs.json .....soubor s dalším českým překladem
│   ├── resources/.....složka se zdroji např. pro pohledy
│   ├── routes/
│   │   └── web.php.....soubor s definovanými URL cestami
│   ├── .env.example ..... konfigurační soubor např. pro databázi a emailový server
│   └── composer.json .....soubor s požadovanými komponentami pro composer
```

└─ package.json soubor s požadovanými balíčky pro npm
└─ database_example.sql výchozí stav databáze k importování (prázdná)
└─ database_test.sql vzorový stav databáze pro testování k importování

A.2 Použité komponenty

V této příloze jsou uvedeny jednotlivé komponenty využité při realizaci bakalářské práce.

LOLIN D1 Mini ESP8266 v3.0.0 WiFi modul

www.laskakit.cz/lolin-d1-mini-esp8266-v3-1-0-wifi-modul/

WeMos D1 mini lithium battery shield V1.2

www.laskakit.cz/wemos-d1-mini-lithium-battery-shield/

Solární panel 6V 1W

www.laskakit.cz/solarni-panel-6v-1w/

WeMos D1 mini SHT30 shield, I2C

www.laskakit.cz/wemos-d1-mini-sht30-shield--i2c/

Převodník MAX6675 + Termočlánek typu K 800C M6 0.5m

www.laskakit.cz/prevodnik-max6675-termoclanek-typu-k-800c-m6-0-5m/

WeMos ESP-WROOM-02 vývojová deska

www.laskakit.cz/wemos-esp-wroom-02-vyvojova-deska/

LilyGO TTGO T5-4.7"E-Paper ESP32 WiFi Modul

www.laskakit.cz/lilygo-ttgo-t5-4-7--e-paper-esp32-wifi-modul/

GeB LiPol Baterie 603048 900mAh 3.7V JST-PH 2.0

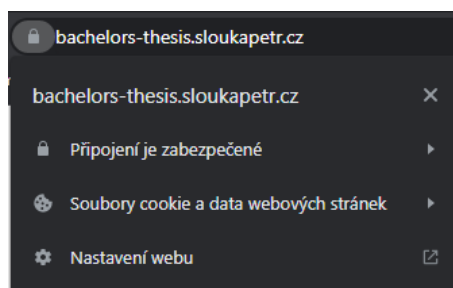
www.laskakit.cz/ehao-lipol-baterie-603048-900mah-3-7v/

Panasonic NCR18650B 3350mAh 18650 Li-Ion Baterie

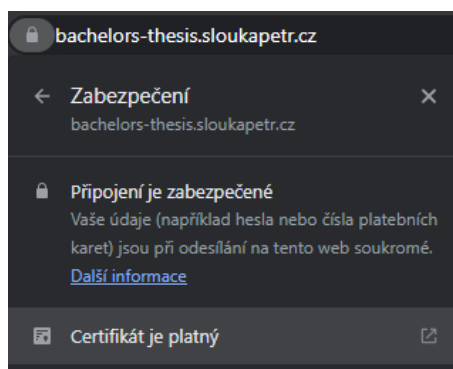
www.laskakit.cz/3350mah-panasonic-ncr18650b-18650-li-ion-baterie/

A.3 Zjištění klíče zabezpečení

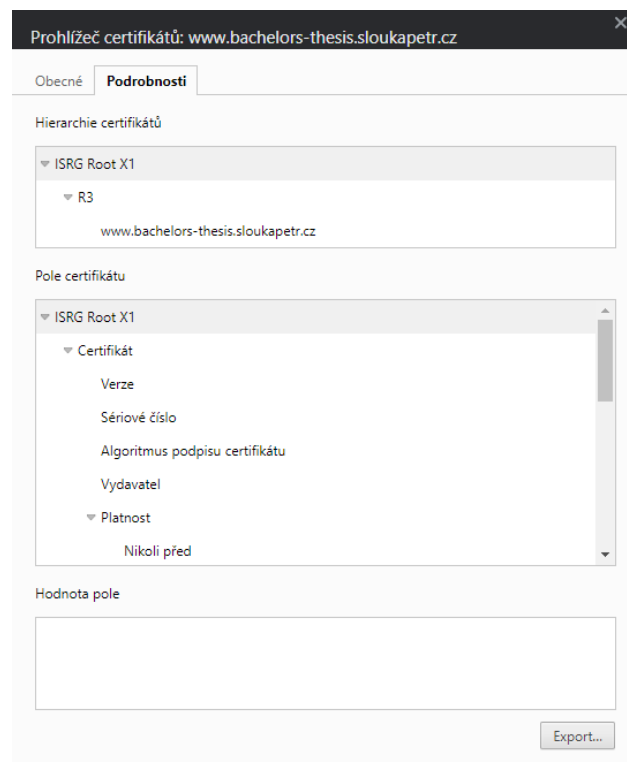
Na obrázcích je uveden zevrubný postup dle kapitoly 5.2 pro zjištění klíče k jeho následné implementaci do jednotlivých programů automatizovaného vytápění.



Obr. A.1: Postup pro zobrazení SSL klíče

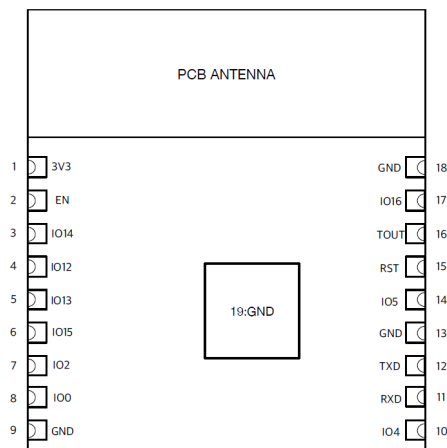


Obr. A.2: Ověření platnosti certifikátu



Obr. A.3: Exportování klíče pro šifrovanou komunikaci

A.4 Schéma čipu ESP WROOM 02



Obr. A.4: Popis výstupů ESP WROOM 02 [8]

A.5 Srovnání protokolů HTTP a HTTPS

V příloze jsou uvedeny obrázky srovnávající protokoly HTTP a HTTPS z pohledu zařízení v domácnosti.

No.	Time	Source	Destination	Protocol	Length	Info
219	3.308696	192.168.1.3	62.109.154.113	TCP	62	58063 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
222	3.321054	62.109.154.113	192.168.1.3	TCP	64	80 → 58063 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1452 SACK_PERM=1
223	3.321230	192.168.1.3	62.109.154.113	TCP	54	58063 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
226	3.321476	192.168.1.3	62.109.154.113	HTTP	1324	GET /sensor-inside-publish/XYgXUjEmS1ZjwCZ5fMDQBS3fapU07ji/1-1/20/50 HTTP/1.1
232	3.352141	62.109.154.113	192.168.1.3	TCP	56	80 → 58063 [ACK] Seq=1 Ack=1271 Win=41910 Len=0
244	3.535174	62.109.154.113	192.168.1.3	HTTP	1298	HTTP/1.1 200 OK (text/html)
277	3.584232	192.168.1.3	62.109.154.113	TCP	54	58063 → 80 [ACK] Seq=1271 Ack=1245 Win=65340 Len=0
285	3.595964	192.168.1.3	62.109.154.113	HTTP	1287	GET /favicon.ico HTTP/1.1

Hypertext Transfer Protocol

> GET /sensor-inside-publish/XYgXUjEmS1ZjwCZ5fMDQBS3fapU07ji/1-1/20/50 HTTP/1.1\r\n

Host: www.bachelors-thesis.sloukapetr.cz\r\n

Connection: keep-alive\r\n

Upgrade-Insecure-Requests: 1\r\n

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Safari/537.36\r\n

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7\r\n

Accept-Encoding: gzip, deflate\r\n

Accept-Language: cs-CZ,cs;q=0.9,pl;q=0.8,en;q=0.7\r\n

> [truncated]Cookie: XSRF-TOKEN=eyJpdIi6IjQ3dmI2QVY1RjE5a3V3R6tHUFJGfXc9PSIsInZhbnV1IjoiwG9SSfBXUjdqNmVQ1NWRUa0dElwksiSVY2Q0VteVhYkdVS0lFZThTaFhhclZc\r\n

[Full request URI: <http://www.bachelors-thesis.sloukapetr.cz/sensor-inside-publish/XYgXUjEmS1ZjwCZ5fMDQBS3fapU07ji/1-1/20/50>]

[HTTP request 1/9]

[Response in frame: 244]

[Next request in frame: 285]

Obr. A.5: Pakety při spojení HTTP protokolem

No.	Time	Source	Destination	Protocol	Length	Info
296	3.615776	62.109.154.113	192.168.1.3	HTTP	367	HTTP/1.1 200 OK
305	3.662284	192.168.1.3	62.109.154.113	TCP	54	58063 → 80 [ACK] Seq=2504 Ack=1558 Win=65027 Len=0
388	4.124223	192.168.1.3	62.109.154.113	HTTP	1350	GET /sensor-inside-publish/XYgXUjEmS1ZjwCZ5fMDQBS3fapU07ji/1-1/20/50 HTTP/1.1
390	4.152742	62.109.154.113	192.168.1.3	TCP	56	80 → 58063 [ACK] Seq=1558 Ack=3800 Win=41910 Len=0
399	4.286698	62.109.154.113	192.168.1.3	HTTP	1298	HTTP/1.1 200 OK (text/html)
402	4.330719	192.168.1.3	62.109.154.113	TCP	54	58063 → 80 [ACK] Seq=3800 Ack=2802 Win=65340 Len=0
415	4.539466	192.168.1.3	62.109.154.113	HTTP	1350	GET /sensor-inside-publish/XYgXUjEmS1ZjwCZ5fMDQBS3fapU07ji/1-1/20/50 HTTP/1.1
418	4.570961	62.109.154.113	192.168.1.3	TCP	56	80 → 58063 [ACK] Seq=2802 Ack=5096 Win=41910 Len=0

Transmission Control Protocol, Src Port: 58063, Dst Port: 80, Seq: 2504, Ack: 1558, Len: 1296

Source Port: 58063

Destination Port: 80

[Stream index: 9]

[Conversation completeness: Incomplete, DATA (15)]

[TCP Segment Len: 1296]

Sequence Number: 2504 (relative sequence number)

Sequence Number (raw): 3961532580

[Next Sequence Number: 3800 (relative sequence number)]

Acknowledgment Number: 1558 (relative ack number)

Acknowledgment number (raw): 1771676788

0101 = Header Length: 20 bytes (5)

Obr. A.6: Detail paketu při spojení HTTP protokolem

No.	Time	Source	Destination	Protocol	Length	Info
1505	104.413898	192.168.1.3	62.109.154.113	TCP	62	57839 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM
1510	104.430726	62.109.154.113	192.168.1.3	TCP	62	443 → 57839 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1452 SACK_PERM
1512	104.430819	192.168.1.3	62.109.154.113	TCP	54	57839 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
1514	104.431206	192.168.1.3	62.109.154.113	TLSv1.3	705	Client Hello
1516	104.444643	62.109.154.113	192.168.1.3	TCP	54	443 → 57839 [ACK] Seq=1 Ack=652 Win=41689 Len=0
1518	104.448817	62.109.154.113	192.168.1.3	TLSv1.3	1506	Server Hello, Change Cipher Spec, Application Data
1519	104.448817	62.109.154.113	192.168.1.3	TCP	1506	443 → 57839 [PSH, ACK] Seq=1453 Ack=652 Win=41689 Len=1452 [TCP segment of a reassembled PDU]
1520	104.448817	62.109.154.113	192.168.1.3	TCP	1246	443 → 57839 [PSH, ACK] Seq=2905 Ack=652 Win=41689 Len=1192 [TCP segment of a reassembled PDU]
1521	104.448817	62.109.154.113	192.168.1.3	TLSv1.3	614	Application Data, Application Data, Application Data
1522	104.448878	192.168.1.3	62.109.154.113	TCP	54	57839 → 443 [ACK] Seq=652 Ack=4657 Win=65340 Len=0
1523	104.450693	192.168.1.3	62.109.154.113	TLSv1.3	134	Change Cipher Spec, Application Data
1524	104.450833	192.168.1.3	62.109.154.113	TLSv1.3	152	Application Data
1525	104.450959	192.168.1.3	62.109.154.113	TLSv1.3	1200	Application Data
1528	104.462446	62.109.154.113	192.168.1.3	TCP	54	443 → 57839 [ACK] Seq=4657 Ack=732 Win=41689 Len=0
1529	104.462446	62.109.154.113	192.168.1.3	TLSv1.3	357	Application Data
1530	104.462446	62.109.154.113	192.168.1.3	TCP	54	443 → 57839 [ACK] Seq=4960 Ack=830 Win=41689 Len=0
1531	104.462446	62.109.154.113	192.168.1.3	TLSv1.3	357	Application Data
1532	104.462446	62.109.154.113	192.168.1.3	TLSv1.3	125	Application Data
1533	104.462522	192.168.1.3	62.109.154.113	TCP	54	57839 → 443 [ACK] Seq=1976 Ack=5334 Win=64663 Len=0
1534	104.462760	192.168.1.3	62.109.154.113	TLSv1.3	85	Application Data
1535	104.483334	62.109.154.113	192.168.1.3	TCP	54	443 → 57839 [ACK] Seq=5334 Ack=1976 Win=41689 Len=0

Obr. A.7: Pakety při spojení HTTPS protokolem

No.	Source	Destination	Protocol	Length	Info
1505	192.168.1.3	62.109.154.113	TCP	62	57839 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
1510	62.109.154.113	192.168.1.3	TCP	62	443 → 57839 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1452 SACK_PERM=1
1512	192.168.1.3	62.109.154.113	TCP	54	57839 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
1514	192.168.1.3	62.109.154.113	TLSv1.3	705	Client Hello
1516	62.109.154.113	192.168.1.3	TCP	54	443 → 57839 [ACK] Seq=1 Ack=652 Win=41689 Len=0
1518	62.109.154.113	192.168.1.3	TLSv1.3	1506	Server Hello, Change Cipher Spec, Application Data

```

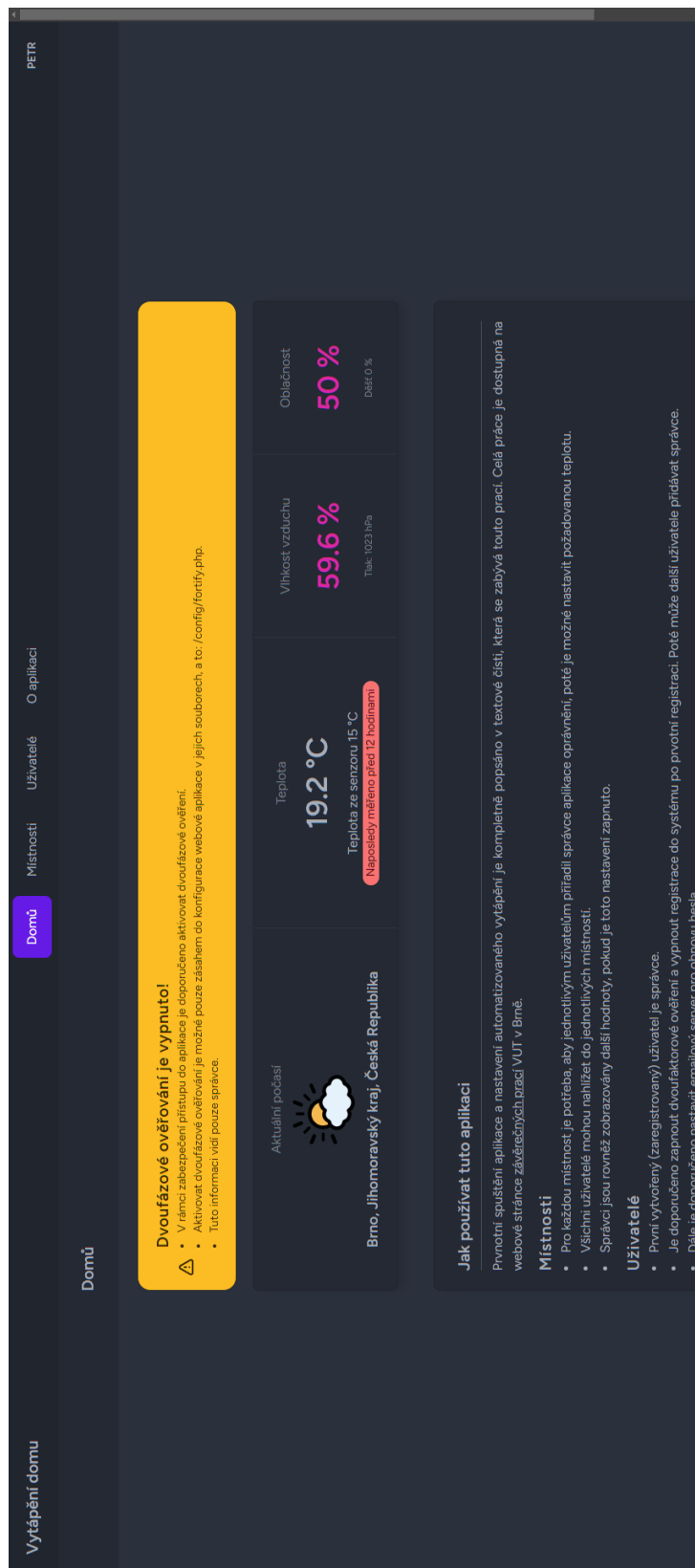
> Frame 1518: 1506 bytes on wire (12048 bits), 1506 bytes captured (12048 bits) on interface \Device\NPF_{B8B1461-DEF2-408E-8DB8-F9D09F0865B1}, id 0
> Ethernet II, Src: TP-Link db:d6:6a (c0:06:c3:db:d6:6a), Dst: IntelCor_5d:0a:7a (4c:79:6e:5d:0a:7a)
> Internet Protocol Version 4, Src: 62.109.154.113, Dst: 192.168.1.3
> Transmission Control Protocol, Src Port: 443, Dst Port: 57839, Seq: 1, Ack: 652, Len: 1452
▼ Transport Layer Security
  ▼ TLSv1.3 Record Layer: Handshake Protocol: Server Hello
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 122
    ▼ Handshake Protocol: Server Hello
      Handshake Type: Server Hello (2)
      Length: 118
      Version: TLS 1.2 (0x0303)
      Random: 43a3a7ccb69dd94dbc20d476cc1ff3e5cb6c999187e5a454cc01ffaf54c2a88e
      Session ID Length: 32
      Session ID: 4d47d5d71d6c8e7c24f2cf47845f0e0dcce8778256402d008f168caf907f9b4e
      Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
      Compression Method: null (0)
      Extensions Length: 46
      > Extension: supported_versions (len=2)
      > Extension: key_share (len=36)
        [JA3S Fullstring: 771,4866,43-51]
        [JA3S: 15af977ce25de452b96affa2addb1036]
    ▼ TLSv1.3 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
      Content Type: Change Cipher Spec (20)
      Version: TLS 1.2 (0x0303)
      Length: 1
      Change Cipher Spec Message
    ▼ TLSv1.3 Record Layer: Application Data Protocol: http-over-tls
      Opaque Type: Application Data (23)
      Version: TLS 1.2 (0x0303)
      Length: 36
      Encrypted Application Data: 77d467cbb1bc436fb9fd05c21b61d09e708790db8fbbac823c3aa140379b69e5cc11c970
      [Application Data Protocol: http-over-tls]

```

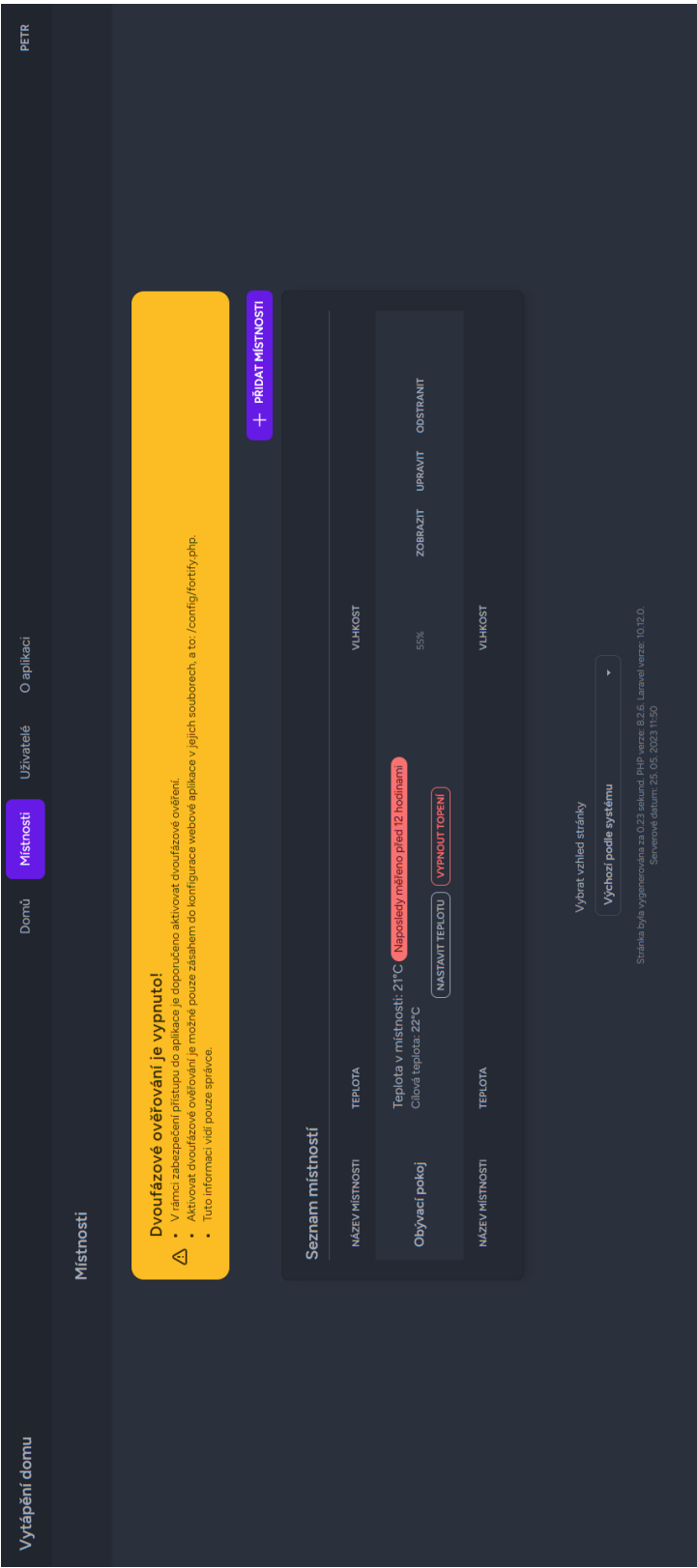
Obr. A.8: Detail paketu při spojení HTTPS protokolem

A.6 Webová aplikace

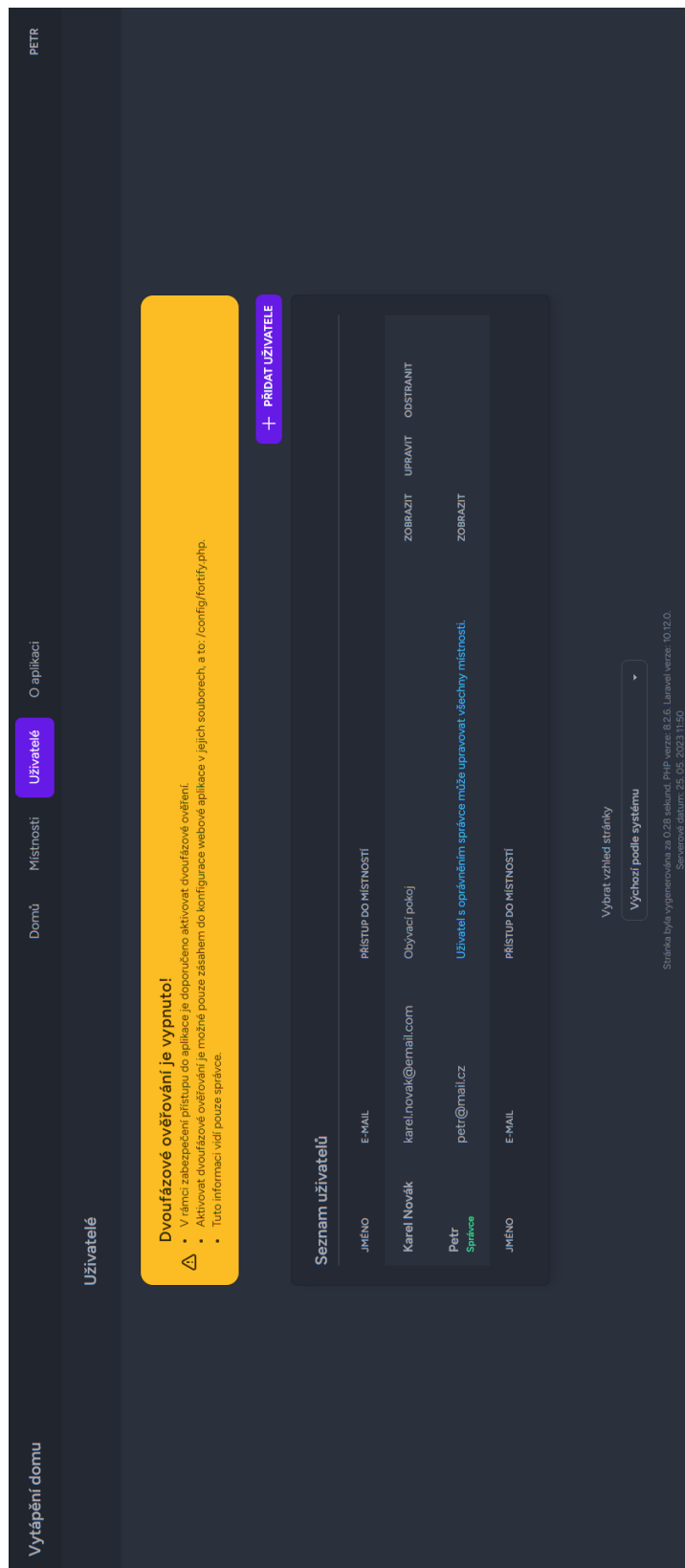
Příloha poskytuje náhled jednotlivých stránek systému, aby čtenář měl i vizuální představu o tom, jak webová aplikace vypadá.



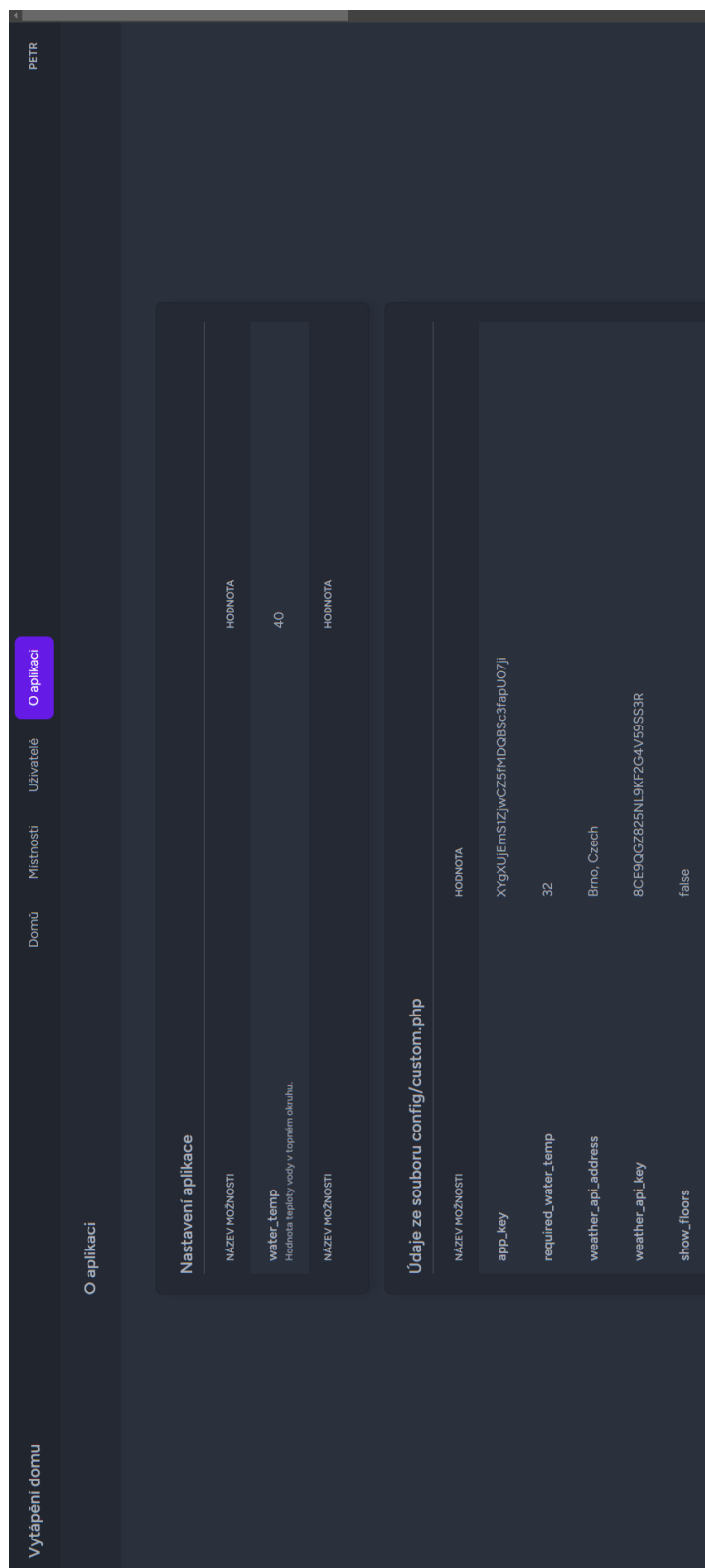
Obr. A.9: Domovská stránka webové aplikace



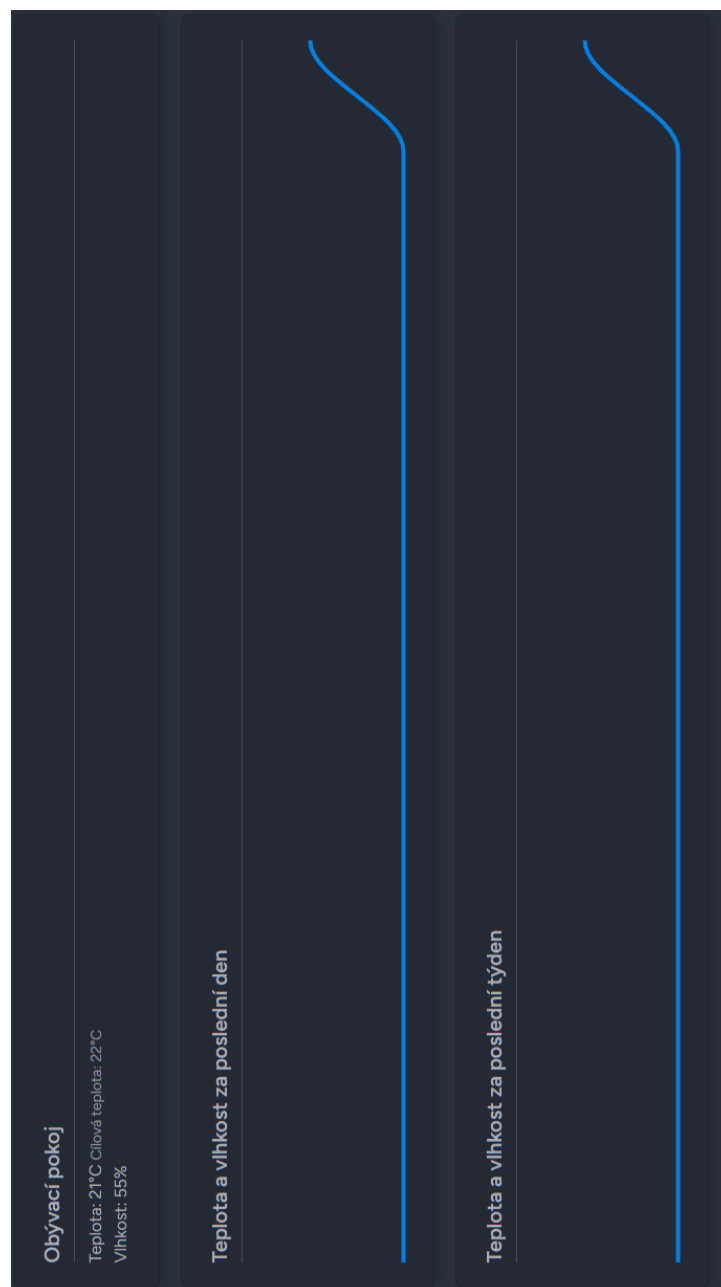
Obr. A.10: Seznam místností



Obr. A.11: Seznam uživatelů



Obr. A.12: Náhled hodnot pro správce aplikace



Obr. A.13: Náhled místnosti