

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačního inženýrství



Diplomová práce

Automatizovaná ekvitermní regulace kotle

Bc. Pavel Tůma

© 2023 ČZU v Praze

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Pavel Tůma

Systémové inženýrství a informatika

Informatika

Název práce

Automatizovaná ekvitermní regulace kotle

Název anglicky

Automated equithermal boiler regulation

Cíle práce

Cílem práce je vytvoření automatizovaného systému ekvitermní regulace kotle. Navržený automatizovaný systém bude sbírat a vyhodnocovat data ze senzorů a regulovat výkon kotle. Nasbíraná systémová data budou odeslána pomocí internetu, a následně uložena do MySQL databáze na PHP webové stránce. Webové rozhraní bude také autorizovanému uživateli nabízet vzdálenou správu a zobrazování hodnot. Cílem automatizovaného systému je úspora paliva a zvýšení efektivity kotle.

Metodika

Teoretická část bude vypracována na základě literárních zdrojů zabývajících se tématem automatizace, chytré domácnosti a programování mikrořadičů.

V praktické části práce bude zpracována analýza a návrh architektury řešeného problému. Implementací řešení vznikne automatizovaný systém ekvitermní regulace kotle naprogramovaný pomocí Arduino IDE. Systém bude vytvořen propojením mikrořadičů a odpovídajících senzorů dle návrhu řešení. Automatizovaný systém bude odesílat a přijímat data pomocí Wi-Fi do sítě internet. K systému vznikne PHP webové rozhraní s databází MySQL, které bude ukládat, a následně interpretovat hodnoty. Také umožní přihlášenému uživateli ovládat automatizovaný systém na dálku.

Doporučený rozsah práce

50-60 stran

Klíčová slova

Automatizace, mikrořadiče, chytrá domácnost, senzory, programování, PHP, ekvitermní regulace, IoT

Doporučené zdroje informací

BELL, C A. *Beginning sensor networks with Arduino and Raspberry Pi*. [New York, New York]: Apress, 2013. ISBN 1430258241.

GERON, D. *Arduino Programming*. Tiger Gain LTD, 2021. ISBN 1914306708

SELECKÝ, M. *Arduino*. Computer press 2016, ISBN 978-80-251-4840-2

VODA, Z. *Průvodce světem Arduina*. HW Kitchen 2015, ISBN 978-80-87106-90-7

Předběžný termín obhajoby

2022/23 LS – PEF

Vedoucí práce

Ing. Marek Pícka, Ph.D.

Garantující pracoviště

Katedra informačního inženýrství

Elektronicky schváleno dne 20. 3. 2023

Ing. Martin Pelikán, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 28. 3. 2023

doc. Ing. Tomáš Šubrt, Ph.D.

Děkan

V Praze dne 28. 03. 2023

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Chytrá domácnost" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne datum odevzdání

Poděkování

Rád bych touto cestou poděkoval Ing. Marku Píckovi, Ph.D. za vstřícnost, ochotu a odborné vedení při zpracování diplomové práce.

Automatizovaná ekvitermní regulace kotle

Abstrakt

Diplomová práce se zabývá naprogramováním a sestavením automatizované ekvitermní regulace kotle a posouzením, zda bude mít aplikace tohoto řešení pozitivní vliv na spotřebu paliva.

Teoretická část práce obsahuje obecné seznámení s technologiemi a trendy v oblasti automatizace a chytré domácnosti. Dále je obecně popsána technologie mikrokontrolérů a programování systému Arduino. Následně jsou podrobně analyzovány jednotlivé komponenty, které budou použity v praktické části této práce, a také způsob výpočtu ekvitermní rovnice a stanovení křivky.

Praktická část práce se zabývá analýzou současného stavu a návrhem možného řešení automatizovaného systému. Navržená varianta disponující síťovým přístupem byla následně naprogramována a sestavena. K automatizovanému systému vznikla webová stránka umožňující autorizovanému uživateli interpretaci dat a dálkové řízení automatizovaného systému.

Po implementaci a spuštění řešeného systému byla analyzována teplotní data v závislosti na spotřebě paliva pro automatizovaný a manuální režim regulace výkonu. Výsledkem měření je 10,71% úspora paliva při použití automatizovaného systému ekvitermní regulace výkonu kotle oproti manuální regulaci výkonu, nicméně výsledná data jsou spíše ilustračního charakteru z důvodu nemožnosti zaručit stejné podmínky pro měření.

Klíčová slova: Automatizace, mikrořadiče, chytrá domácnost, senzory, programování, PHP, ekvitermní regulace, IoT

Automated equithermal boiler regulation

Abstract

The diploma thesis deals with the programming and assembly of the automated equithermal regulation of the boiler and the assessment of whether the application of this solution has a positive effect on fuel consumption.

The theoretical part of the thesis contains a general introduction to technologies and trends in the field of automation and smart home. Furthermore, microcontroller technology and programming of the Arduino system are generally described. Subsequently, the individual components that will be used in the practical part of this work are analysed in detail, as well as the method of calculating the equithermal equation and determining the curve.

The practical part of the work deals with the design of possible solutions for the automated system, in variants with and without network connection. For the next step, a variant with network access was selected, which was programmed and assembled. A website was created for the automated system, allowing the authorized user to interpret the data and control remotely the automated system.

After the implementation and start-up of the solved system, the temperature data was analysed depending on the fuel consumption for the automated and manual mode of power regulation. The result of the measurement is a 10,71 % fuel saving when using the automated equithermal boiler power regulation system compared to manual power regulation, however, the resulting data are rather illustrative due to the impossibility of guaranteeing the same conditions for measurement.

Keywords: automation, microcontrollers, smart home, sensors, programming, PHP, equithermal regulation, IoT

Obsah

1	Úvod	11
2	Cíl práce a metodika.....	12
2.1	Cíl práce	12
2.2	Metodika	12
3	Teoretická východiska.....	13
3.1	Přehled bezdrátových komunikačních technologií.....	13
3.1.1	Wi-Fi	13
3.1.2	Bluetooth	14
3.1.3	Zigbee.....	15
3.1.4	Z-Wave.....	16
3.2	Internet věcí (IoT).....	16
3.3	Mikrokontroléry	17
3.3.1	Popis.....	17
3.3.2	Využití v systému chytré domácnosti	17
3.3.3	Příklady mikrokontrolérů	18
3.4	Arduino	19
3.4.1	Popis.....	19
3.4.2	Dostupný hardware	19
3.5	Webové a databázové technologie	26
3.5.1	PHP	26
3.5.2	MySQL.....	27
3.5.3	Http protokol.....	28
3.5.4	Basic autentizace.....	30
3.6	Automatizovaná regulace kotle na tuhá paliva	31
3.6.1	Ekvitermní regulace	31
3.6.2	Výpočet ekvitermní rovnice a křivky.....	31
3.6.3	Regulátor výkonu.....	33
4	Vlastní práce	34
4.1	Analýza	34
4.1.1	Současný stav.....	34
4.1.2	Požadavky.....	35
4.2	Návrh řešení	36
4.2.1	Seznam použitých komponent.....	37

4.2.2	Hardwarový popis návrhu řešení	37
4.2.3	Webové rozhraní.....	40
4.2.4	MySQL databáze	45
4.3	Implementace řešení	46
4.3.1	Stanovení rovnice ekvitermní křivky.....	47
4.3.2	Programování vývojové desky	49
4.3.3	Programování webové stránky	53
5	Zhodnocení výsledků	59
6	Závěr	61
7	Bibliografie.....	63
8	Seznam obrázků, tabulek, grafů a zkratk.....	67
8.1	Seznam obrázků	67
8.2	Seznam tabulek	67

1 Úvod

Tato diplomová práce se zaměřuje zejména na popsání a výběr technologií a hardwarových komponent používaných k sestrojení systémů automatizace, a následné vybudování systému automatizované ekvitermní regulace kotle dle nejvhodnějšího návrhu.

Problematika výběru, či sestrojení systémů automatizace částí domácnosti se s přibývajícím počtem různorodých technologií a uživatelů neustále navyšuje. Je tak čím dál tím obtížnější, zvolit optimální technologie a jejich následnou konfiguraci.

Teoretická část práce bude zaměřena na popis bezdrátových technologií a trendů v systémech automatizace. Poté bude provedena analýza technologie mikrořadičů obecně s následným zaměřením na Arduino, ESP8266 a použité senzory a motory. Následně bude obecně popsána technologie pro tvorbu PHP webových stránek, a také způsob výpočtu ekvitermní regulace a stanovení ekvitermní křivky.

Praktická část práce se bude zabývat zkoumáním současného stavu topného systému, na kterém bude navrhované řešení použito. Dále zpracováním funkčního návrhu vhodného a ekonomicky efektivního řešení automatizace ekvitermní regulace kotle. Po analýze, návrhu a výběru vhodného řešení, bude následovat sestrojení samotného systému ve fyzické podobě a jeho následná aplikace a spuštění na topném systému.

K automatizovanému systému vznikne zabezpečená webová stránka v jazyce PHP s databázovým úložištěm MySQL. Pomocí wifi modulu ESP8266 bude systém komunikovat do sítě internet na příslušnou webovou stránku, která bude soužit pro řízení systému na dálku, a také pro zobrazování naměřených dat.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem práce je vytvoření automatizovaného systému ekvitermní regulace kotle. Navržený automatizovaný systém bude sbírat a vyhodnocovat data ze senzorů a regulovat výkon kotle. Násbíraná systémová data budou odeslána do sítě internet, a následně uložena do MySQL databáze na PHP webové stránce. Webové rozhraní bude také autorizovanému uživateli nabízet vzdálenou správu či interpretaci hodnot. Cílem automatizovaného systému je úspora paliva a zvýšení efektivity kotle.

2.2 Metodika

Teoretická část bude vypracována na základě literárních zdrojů zabývajících se tématem automatizace a programování mikrořadičů. Dále bude obecně popsán systém Arduino, a také technologie PHP a MySQL pro tvorbu webových stránek. Důležitý bude také popis a způsob výpočtu ekvitermní regulace.

V praktické části práce bude zpracována analýza a návrh architektury řešeného problému. Implementací řešení vznikne automatizovaný systém ekvitermní regulace kotle naprogramovaný pomocí Arduino IDE. Systém bude vytvořen propojením vývojové desky, motoru a odpovídajících senzorů dle návrhu řešení. Automatizovaný systém bude měřit teplotní data ze tří senzorů, a následně bude pomocí servomotoru regulovat vzduchovou klapku kotle. Regulace vzduchové klapky bude vyhodnocována pomocí výpočtu ekvitermní rovnice, která stanoví hraniční teplotu regulace. Dále bude automatizovaný systém odesílat a přijímat data pomocí Wi-Fi modulu do sítě internet. K systému vznikne PHP webová stránka s databází MySQL, která bude ukládat, a následně interpretovat hodnoty. Také umožní přihlášenému uživateli ovládat automatizovaný systém na dálku.

3 Teoretická východiska

V této kapitole bude zpracována literární rešerše zvolené problematiky. Budou zde popsány bezdrátové technologie, které jsou v automatizovaných systémech používány. Následně bude obecně popsána technologie Iot, mikrořadičů a platformy Arduino. Poté budou podrobně zanalyzovány jednotlivé komponenty tvořeného automatizovaného systému. Následovat bude ještě popis technologií pro tvorbu webu a komunikace v síti.

3.1 Přehled bezdrátových komunikačních technologií

3.1.1 Wi-Fi

Wi-Fi je bezdrátová komunikační technologie, která umožňuje přenos dat mezi zařízeními. Funguje na principu bezdrátového přenosu dat mezi zařízeními pomocí modulů, které jsou v nich integrovány. Jsou jimi například mobilní telefony, počítače, tablety, televize a další. Tyto moduly komunikují se základnovým stanovištěm, které disponuje internetovým připojením a umožňuje přenos dat. Tyto moduly také umožňují zařízením komunikovat mezi sebou pomocí Wi-Fi sítě.

Wi-Fi využívá frekvence 2,4 a 5 GHz, a poskytuje širokopásmový přenos dat s rychlostmi stovek MB/s. Tato technologie se stala jednou z nejpoužívanějších bezdrátových sítí a klíčovým prvkem pro mnoho aplikací, jako jsou internetové služby, hry, streamování videa, videohovory a další.

Jsou podporovány různé standardy, které umožňují zařízením připojení k Wi-Fi síti, a umožňují různým typům aplikací využívat Wi-Fi. Tyto standardy se od sebe odlišují rychlostí, pokrytím, frekvencí a bezdrátovými protokoly.

Hlavní Wi-Fi standardy:

- 802.11a: první standard, který umožňuje vysokou rychlost a používá frekvenci 5 GHz.
- 802.11b: nižší rychlost než 802.11a, ale širší pokrytí na frekvenci 2,4 GHz.
- 802.11g: stejné frekvence jako 802.11b, ale vyšší rychlost a lepší kompatibilita s 802.11a.
- 802.11n: zlepšuje rychlost a pokrytí předchozích standardů, podporuje frekvence 2,4 a 5 GHz.
- 802.11ac: vysokorychlostní standard, který používá pouze frekvenci 5 GHz.

- 802.11ax: poslední a nejmodernější standard, který zvyšuje rychlost, efektivitu a pokrytí.

Výhody Wi-Fi zahrnují vysokou rychlost přenosu dat, možnost připojení k internetu bez nutnosti fyzického kabelu, schopnost komunikace s mnoha zařízeními současně a další. Navzdory svým výhodám má Wi-Fi také určité omezení, jako je omezený dosah, závislost na kvalitě signálu a potenciální rušení s jinými bezdrátovými sítěmi pracujícími na podobných frekvencích.

V budoucnosti se očekává, že Wi-Fi bude nadále významnou součástí bezdrátového připojení k internetu a že se bude dále vylepšovat a rozšiřovat. (Řehák, 2003)

3.1.2 Bluetooth

Bluetooth je bezdrátová komunikační technologie, která umožňuje přenos dat mezi zařízeními v krátkém dosahu. Byla vynalezena v 90. letech, a od té doby se stala standardem pro bezdrátové připojení v mnoha odvětvích, včetně automatizace a chytré domácnosti.

Bluetooth se v průběhu let vyvíjel, přičemž bylo vydáno několik různých verzí, které se odlišují rychlostí přenosu dat, dosahem a kompatibilitou s jinými technologiemi.

Verze rozhraní Bluetooth:

- Bluetooth 1.0 a 1.0B: První verze Bluetooth, která byla uvedena v roce 1999. Měla nízkou rychlost přenosu dat a krátký dosah.
- Bluetooth 2.0 + EDR (Enhanced Data Rate): Tato verze byla uvedena v roce 2004 a zlepšila rychlost přenosu dat až na 2 Mbps.
- Bluetooth 3.0 + HS (High Speed): Tato verze umožňuje přenos dat rychlostí až 24 Mbps pomocí Wi-Fi připojení.
- Bluetooth 4.0: Tato verze byla uvedena v roce 2010 a přidala podporu pro technologie Low Energy (LE), což umožňuje delší výdrž baterie pro IoT zařízení.
- Bluetooth 5.0: Tato verze byla uvedena v roce 2016 a zlepšila rychlost přenosu dat až na 2 Mbps a dosah až na 4x více než u předchozí verze.

V současné době se používá nejčastěji verze Bluetooth 5.0 a vývoj pokračuje dále s cílem zlepšit rychlost přenosu dat, dosah a spolehlivost.

Výhody technologie Bluetooth:

- Nízké náklady na implementaci
- Nízká spotřeba energie
- Snadné použití a propojení zařízení
- Schopnost propojit více zařízení
- Kompatibilita s širokým spektrem zařízení

Ačkoliv Bluetooth nabízí mnoho výhod, má i několik nedostatků, jako je krátký dosah, možnost rušení jinými bezdrátovými zařízeními a omezení rychlosti přenosu dat. Tyto nedostatky se však s postupem času zlepšují v rámci vývoje nových verzí technologie. (Čepička, 2009)

3.1.3 Zigbee

Zigbee je bezdrátová komunikační technologie určená pro Internet věcí (IoT) a automatizaci domácnosti. Je to jedna z nejrozšířenějších technologií pro chytrou domácnost, která poskytuje vysokou účinnost a spolehlivost.

Dále umožňuje bezdrátové připojení mezi různými zařízeními, jako jsou čidla, ovladače, termostaty a další, což umožňuje snadné a efektivní řízení všech funkcí v domácnosti. Navíc Zigbee podporuje vysokou hustotu zařízení, což umožňuje použití všech funkcí v celém domě bez nutnosti použití dalšího hardwaru.

Tato technologie má nízkou spotřebu energie, což umožňuje dlouhou životnost baterií a nižší cenu zařízení. Navíc je celkem účinně schopna přenášet data přes překážky, jako jsou stěny, stropy a podobně.

Mimo jiné podporuje šifrování a zabezpečení proti neoprávněnému přístupu, což zaručuje bezpečnost dat a zařízení. Také je kompatibilní s širokou škálou zařízení, a má široké spektrum použití včetně inteligentního osvětlení, termostatů, bezpečnostního systému a dalších. (J. Koton, 2006)

3.1.4 Z-Wave

Z-Wave je bezdrátová síť určená pro domácí automatizaci a internet věcí (IoT). Je to jedna z nejrozšířenějších technologií pro chytrou domácnost, která poskytuje vysokou kvalitu a spolehlivost.

Tato technologie podporuje mnoho různých typů zařízení a nabízí širokou škálu funkcí, jako je například ovládání světel, zámků, topení a klimatizace pomocí chytrého telefonu nebo hlasového asistenta. Navíc Z-Wave podporuje šifrování a zabezpečení proti neoprávněnému přístupu, což zaručuje bezpečnost dat a zařízení.

Síť se vyznačuje nízkou spotřebou energie a nízkou rychlostí přenosu dat, což umožňuje dlouhou životnost baterií a nižší cenu zařízení. Navíc je schopen přenášet data přes překážky, jako jsou stěny a stropy, což umožňuje snadné použití v celém domě bez nutnosti použití dalšího hardwaru. (Resler, 2019)

3.2 Internet věcí (IoT)

IoT (Internet of Things) je síť fyzických zařízení, která jsou propojena se sítí internet a jsou schopna mezi sebou komunikovat a odesílat data bez nutnosti asistence člověka.

Hlavním cílem IoT je zlepšit efektivitu a umožnit automatizaci procesů, jako je například řízení energetické spotřeby, sledování zdravotního stavu pacientů a optimalizace výrobního procesu. IoT také umožňuje lepší monitorování a řízení složitých systémů, jako je například infrastruktura měst a městského dopravního systému.

Výhody IoT zahrnují zlepšení efektivity a automatizaci procesů, lepší monitorování a řízení složitých systémů a schopnost získávat a analyzovat velké množství dat. Nevýhody zahrnují bezpečnostní rizika, kdy nezabezpečená zařízení mohou být napadnuta a zneužita, a také otázky související s ochranou osobních údajů a soukromí.

V budoucnu se očekává, že IoT bude hrát klíčovou roli při řešení složitých problémů a usnadnění každodenního života. Nicméně k zajištění plného potenciálu této technologie je důležité, aby byly zajištěny bezpečnostní opatření a záruky ochrany soukromí. (Kodůusková, 2022)

3.3 Mikrokontroléry

3.3.1 Popis

Mikrokontroler je malý integrovaný čip, který se skládá z mikroprocesoru, paměti, vstupů/výstupů a dalších periferních zařízení, které umožňují řízení a řešení různých úkolů v elektronických zařízeních, jako jsou například mobilní telefony, automobilové systémy, průmyslová zařízení, roboti a další.

Mikrokontroléry jsou k dispozici v různých architektonických a výkonových specifikacích. Tyto jednotky používají jednoduchý hardware pro řízení funkcí zařízení, jako jsou například napájení, vstupy, výstupy, komunikace s jinými zařízeními a řízení paměti.

Mikrokontroléry jsou řízeny pomocí software (firmware¹), který se nahrává do paměti. Tento řídicí software může být naprogramován v mnoha různých programovacích jazycích jako je například C, C++ nebo assembler², a používají se pro kontrolu funkcí zařízení, jako je například řízení vstupů, výstupů a komunikace s jinými zařízeními. (Wostl, 2019)

3.3.2 Využití v systému chytré domácnosti

Mikrokontroléry jsou často používány v chytrých domácnostech k řízení spotřebičů a zařízení pomocí počítačového programování. Tyto mikrokontroléry lze programovat pomocí jazyků, jako je C a C++ a jsou schopné přijímat a vykonávat instrukce z jiných zařízení, jako jsou chytré telefony nebo počítače.

Chytrá domácnost může využívat mikrokontroléry k řízení různých spotřebičů, jako jsou světla, termostaty, bezpečnostní systémy, automatizaci výroby elektrické energie a mnoho dalšímu. Tyto mikrokontroléry mohou být také použity ke sběru a analýze dat o využití energií a výrobních zdrojů, což může pomoci v optimalizaci spotřeby energie a snížení nákladů.

Mikrokontroléry také mohou být snadno aktualizovány pomocí firmware a software, což umožňuje, aby se jejich funkce a schopnosti v průběhu času rozšiřovaly.

¹ Firmware je typ softwaru, který je uložen přímo v paměti zařízení, jako jsou mikrokontroléry, čipy, senzory a další. Jedná se o programové vybavení, které zajišťuje správné fungování zařízení, kontroluje jeho vstupy a výstupy a umožňuje interakci s ostatními zařízeními. (Eldin, 2023)

² Assembler je programovací jazyk, který umožňuje programátorům psát instrukce pro počítačové procesory přímo v nízkoúrovňovém strojovém kódu. (Thakkar, 2022)

V souhrnu hrají mikrokontroléry důležitou roli v chytrých domácnostech jako základní technologie pro řízení a automatizaci spotřebičů a zařízení. Tyto mikrokontroléry umožňují uživatelům využívat chytrou domácnost efektivně a optimalizovat spotřebu energie a náklady. (Wostl, 2019)

3.3.3 Příklady mikrokontrolérů

- AVR – mikrokontroléry od společnosti Atmel, používané ve vývojových deskách Arduino
- PIC – mikrokontroléry od společnosti Microchip Technology, používané v mnoha aplikacích
- ARM Cortex-M - rodina mikrokontrolérů od společnosti ARM, používaná v mobilních zařízeních, automobilech a průmyslových systémech
- MSP430 – mikrokontroléry od společnosti Texas Instruments, používané pro nízkopříkonové aplikace
- 8051 - starší rodina mikrokontrolérů od společnosti Intel, stále používaná v některých systémech
- STM32 – mikrokontroléry od společnosti STMicroelectronics, používané v mnoha systémech včetně průmyslu, automobilového průmyslu a spotřební elektroniky
- ESP8266/ESP32 – mikrokontroléry s Wi-Fi moduly od společnosti Espressif, používané v IoT aplikacích a projektech s připojením k internetu (Team Tesca, 2021)

3.4 Arduino

3.4.1 Popis

Arduino je open-source platforma pro vývoj snadno použitelného hardwaru a softwaru. Umožňuje programování vlastních vývojových desek opatřených mikrokontrolérem pomocí jednoduchého softwaru a vlastního vývojového prostředí Arduino IDE. Toto vývojové prostředí obsahuje editor kódu, sériový monitor a nástroje pro nahrávání kódu na vývojovou desku.

Desky Arduino jsou navrženy tak, aby byly snadno použitelné a přizpůsobitelné pro různé typy projektů. Jsou nejčastěji používány pro vývoj a prototypování projektů, jako například robotů, senzorů, domácí automatizace, řízení osvětlení a mnoha dalších řešení. Vývojové desky Arduino obsahují mikrokontroléry, které mohou být naprogramovány pomocí jazyka Wiring nebo jazyka C++.

Jednou z výhod platformy Arduino je jeho open-source³ licence, což znamená, že kód a design desek je volně dostupný, a tak je velmi často modifikován. Tento otevřený přístup umožňuje komunitě vývojářů vytvářet a sdílet nové projekty, a rozšiřovat tak funkce a možnosti Arduino platformy. (Voda, 2019)

3.4.2 Dostupný hardware

V následující kapitole bude rozebrán technický detail jednotlivých dostupných hardwarových komponent, které budou použity pro sestavení výsledného automatizovaného systému.

3.4.2.1 Arduino Uno R3

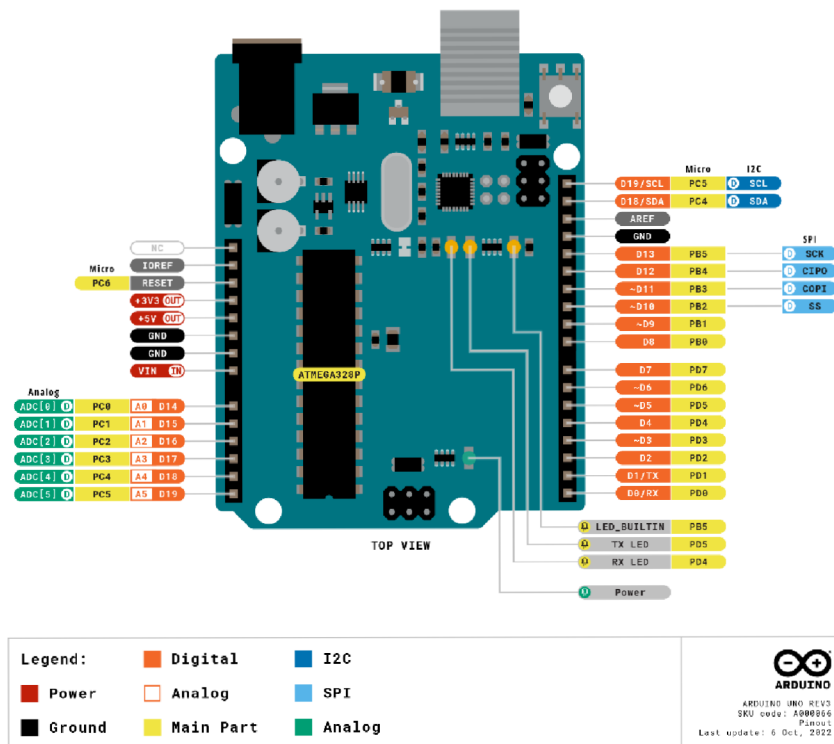
Arduino UNO R3 je jedna z nejpoužívanějších vývojových desek na světě, určená pro vývoj a realizaci projektů v oblasti automatizace a řízení. Je vybavena čipem ATmega328P, který pracuje na frekvenci 16 MHz, poskytuje výkon a paměť potřebnou pro implementaci řady funkcí.

Arduino Uno R3 obsahuje 14 digitálních vstupně/výstupních pinů. Z toho 6 mohou být použity jako PWM výstupy a 6 jako analogové vstupy. Kromě toho obsahuje také 6 pinů pro připojení k UART rozhraní, 2 pinů pro přerušení a 1 pin pro připojení I2C rozhraní.

³ Otevřená licence pro použití softwaru a hardwaru, který je tak možné používat a modifikovat bez porušení autorských práv. (Štráfelda, 2023)

Deska obsahuje USB konektor pro připojení k počítači a napájení, ale také může být napájena z vnějšího zdroje pomocí napájecího konektoru. Arduino Uno R3 také obsahuje integrovaný stabilizátor napětí a resetovací tlačítko.

UNO R3 také podporuje mnoho programovacích jazyků, včetně C++ a vysokoúrovňových jazyků jako je Processing. (Arduino.cc, 2023)



Obrázek 1: Diagram rozložení pinů desky Arduino Uno R3 (Arduino.cc, 2023)

Tabulka 1: Specifikace Arduino Uno R3 (Arduino.cc, 2023)

Mikrokontroler	ATmega328P
Pracovní napětí	5 V
Napájecí napětí (limit)	6–20 V
Počet digitálních I/O pinů	14
Počet digitálních PWM I/O pinů	6
Počet analogových pinů	6
Maximální DC proud na I/O pin	20 mA
Maximální DC napětí pro 3.3V pin	50 mA
Flash paměť	32 KB (ATmega328P)
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Frekvence	16 MHz
Délka	68.6 mm
Šířka	53.4 mm
Hmotnost	25 g

3.4.2.2 Wemos D1 Uno WiFi ESP8266

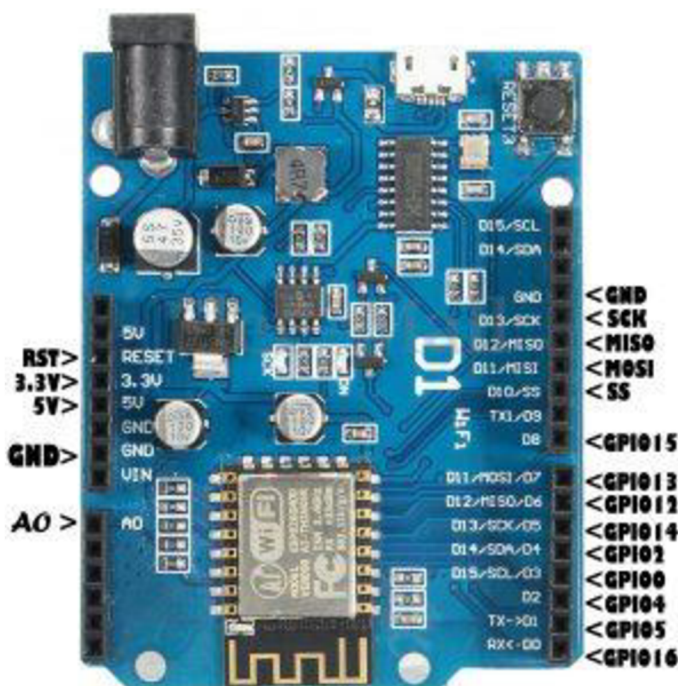
WEMOS D1 UNO WiFi ESP8266 je malá vývojová deska založená na čipu ESP8266, který obsahuje modul umožňující připojení k Wi-Fi síti. Má podobný formát rozložení pinů, jako deska Arduino UNO, což znamená, že je kompatibilní s mnoha různými Arduino shieldy⁴ a moduly.

⁴ Jedná se o rozšiřující karty, které doplňují funkčnost základní desky. Např Podpora Wi-Fi, Bluetooth atp. (Webster, 2015)

Deska obsahuje 11 digitálních vstupů/výstupů (I/O) a 1 analogový vstup. Dále 1 UART⁵ rozhraní, 1 SPI⁶ rozhraní a 1 I²C⁷ rozhraní. K programování a nahrávání kódu na desku se využívá Micro USB rozhraní, které je také používáno k napájení desky.

Deska WEMOS D1 UNO WiFi ESP8266 je vhodná pro vývoj a prototypování IoT projektů, které vyžadují připojení k internetu. Deska je také podporována v prostředí Arduino IDE, což znamená, že může být programována pomocí jazyka C/C++. Další nespornou výhodou vývojové desky je možnost OTA⁸ uploadu, čímž odpadá nutnost připojovat desku k počítači pro nahrání kódu.

Celkově je WEMOS D1 UNO WiFi ESP8266 velmi užitečnou vývojovou deskou pro projekty zaměřené na IoT, které vyžadují připojení k internetu a zároveň využívají široké možnosti rozšiřitelnosti a kompatibility s moduly a shieldy pro desky Arduino UNO. (faranux.com, 2023)



Obrázek 2: Diagram rozložení pinů desky Wemos D1 Uno WiFi ESP8266 (faranux.com, 2023)

⁵ UART se dá specifikovat, jako univerzální asynchronní přijímač a vysílač složící k asynchronnímu přenosu dat. (Admin, 2014)

⁶ SPI je sériové periferní rozhraní, které se používá pro komunikaci mezi řídicími mikroprocesory. (Grusin, 2023)

⁷ Jedná se o sériovou počítačovou sběrnici používanou pro připojení nízkorychlostních periférií k základní, nebo vývojové desce. (Redakce, 2000)

⁸ Zkratka z anglického „Over the air“, znamenající možnost aktualizace softwaru pomocí bezdrátového připojení. (Čermák, 2021)

Tabulka 2: WEMOS D1 UNO WiFi ESP8266 (*faranux.com, 2023*)

Mikrokontroler	ESP-8266EX
Pracovní napětí	3,3 V
Napájecí napětí (limit)	9–24 V
Počet digitálních I/O pinů	11
Počet analogových pinů	1
Flash paměť	4Mb
Frekvence	80MHz/160MHz
Délka	68,6 mm
Šířka	53,4 mm
Hmotnost	25 g

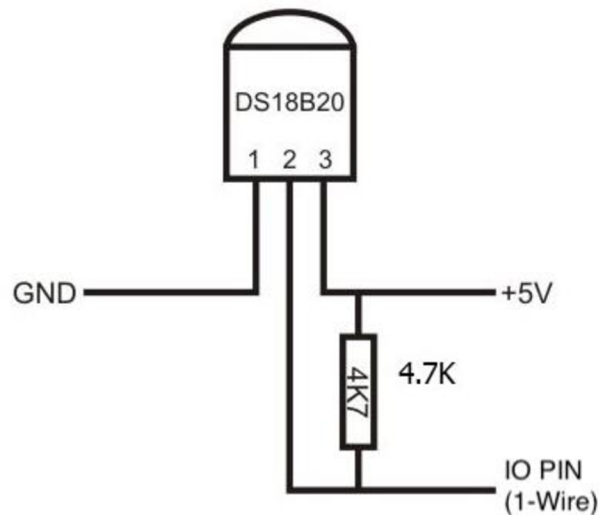
3.4.2.3 Dallas DS18B20

Senzor Dallas DS18B20 je digitální teplotní senzor, který umožňuje měření teploty s přesností až $\pm 0,5$ °C v rozsahu od -55 °C do +125 °C.

Senzor je navržen tak, aby byl jednoduše použitelný s mikrokontroléry nebo jinými digitálními obvody, přičemž je připojen k jedinému datovému vodiči a zdroji napájení. Senzor komunikuje pomocí jednoduchého sériového protokolu, který umožňuje čtení teplotních dat zpět do řídicího mikrokontroléru.

DS18B20 má malé rozměry, což umožňuje jeho snadnou integraci do různých aplikací, jako je například měření teploty v průmyslových procesech, v domácí automatizaci, v řízení klimatizace a topení, v meteorologických stanicích a v lékařských přístrojích.

Další vlastností senzoru DS18B20 je jeho schopnost připojení k většímu počtu senzorů na jediném datovém vodiči. Tento protokol se nazývá 1 – Wire a umožňuje propojení více senzorů do sítě, což je výhodné pro měření teploty v rozličných místnostech bez nutnosti použití více datových vodičů. Každý senzor má unikátní adresu, pomocí které ho lze snadno identifikovat v rámci 1 – Wire sběrnice. Na jedné sběrnici je možno připojit až 50 teplotních senzorů v maximální vzdálenosti až 100 metrů od zdroje napájení. (Faltus, 2023)



Obrázek 3: Diagram rozložení pinů senzoru DS18b20 (pajenicko.cz, 2023)

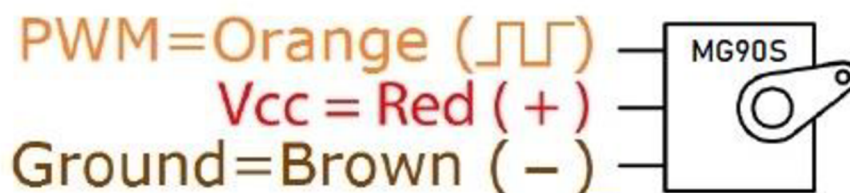
3.4.2.4 Servomotor MG90S

Servomotor MG90S je malý, ale výkonný servomotor, který se často používá v různých elektronických systémech a řešeních.

MG90S váží pouze 13,4 g a má rozměry 22,8 x 12,2 x 28,5 mm. Obsahuje kuličková ložiska, která umožňují rychlé a přesné pohyby s rychlostí otočení 0,1 sekundy za 60 stupňů. Servomotor MG90S má také vysoký točivý moment a může snadno pohybovat menšími předměty.

Servomotor MG90S má tři vodiče, z nichž jsou dva určeny pro napájení a jeden pro řízení polohy a natočení ramene. Napětí pro pohon motoru se obvykle pohybuje v rozmezí 4,8 až 6 V, a maximální moment točivé síly je 1,8 – 2,2 kg/cm.

Servomotor MG90S je kompatibilní s mnoha mikrokontroléry a vývojovými deskami, jako je například Arduino. Díky své malé velikosti a výkonu je MG90S ideální pro různé aplikace, jako jsou například modelářství, robotika, automatizace a další. (Hareendran, 2021)



Obrázek 4: Diagram zapojení vodičů servomotoru MG90S (Hareendran, 2021)

3.4.2.5 LED

LED je zkratka pro anglický výraz "Light Emitting Diode", tedy diodu vysílající světlo. LED je elektronická součástka, která se využívá k vytvoření systému signalizace a světelných zdrojů s vysokou účinností a nízkou spotřebou energie.

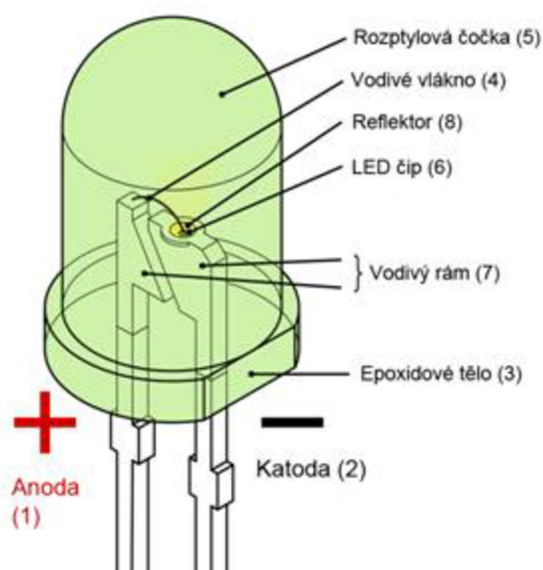
LED mají mnoho výhod oproti klasickým žárovkám nebo zářivkám, jako jsou vysoká účinnost, dlouhá životnost, rychlá reakční doba a nízká emise tepla. Jsou také velmi odolné vůči otřesům a vibracím, což z nich dělá ideální volbu pro použití v různých systémech, kde jsou méně náchylné na poškození, oproti klasickým žárovkám.

LED jsou k dispozici v mnoha různých tvarech a velikostech, včetně kulatých, čtvercových, obdélníkových a SMD⁹ (Surface Mount Device) variant. Svítivost LED se vyjadřuje v lumenech.

Díky svému vysokému výkonu a nízké spotřebě energie jsou LED diody ideální pro použití v osvětlení domů a kanceláří, reklamních a výstavních tabulích, jakož i v automobilovém a průmyslovém osvětlení. LED diody se také používají v displejích, signalizacích a senzorech.

V současné době existuje mnoho různých druhů LED, jako jsou RGB (Red Green Blue) diody, které umožňují vytvořit různé barevné efekty a LED s vysokým CRI (Color Rendering Index), které dokážou reprodukovat barvy věrněji. LED diody se také používají v kombinaci s mikroprocesory a dalšími technologiemi pro vytvoření chytrých osvětlení, které umožňují ovládání z mobilních zařízení a dalších zařízení s připojením k internetu. (Dvořáček, 2009)

⁹ Jedná se o elektronickou součástku, která je přímo umístěná na povrchu plošného spoje. (Rouse, 2016)



Obrázek 5: Diagram LED (Chuděj, 2023)

3.5 Webové a databázové technologie

Kombinace technologií PHP a MySQL umožňuje vytvořit interaktivní webové stránky, které mohou zobrazovat a aktualizovat data uložená v databázi.

3.5.1 PHP

PHP (Hypertext Preprocessor) je skriptovací programovací jazyk používaný k tvorbě dynamických webových stránek a webových aplikací. Byl vyvinut v roce 1995 Rasmusem Lerdorfenem.

Jazyk PHP obsahuje mnoho funkcí pro práci s webovými stránkami, jako jsou například práce s databázemi nebo řízení toku stránky. Skripty v jazyce PHP jsou vykonávány na serverové straně a výsledky jsou posílány na klientskou část v podobě vygenerovaného HTML kódu.

Jazyk PHP podporuje mnoho různých operačních systémů, včetně Windows, Linux a macOS. Pro vývoj v PHP je nutné mít na serveru nainstalovaný PHP interpreter, který pracuje jako samostatný proces na serveru. Kód PHP se spouští při každém požadavku na server, což umožňuje generování dynamických webových stránek.

Jazyk PHP umožňuje práci s různými typy databází, jako jsou například MySQL, PostgreSQL nebo Oracle. To umožňuje tvorbu webových aplikací, které mohou ukládat a zobrazovat informace z databází. Dále PHP obsahuje mnoho knihoven a rozšíření, které

usnadňují práci s různými webovými technologiemi, jako jsou například XML, SOAP nebo REST.

Další výhodou PHP je jeho široká podpora a velké množství dokumentace a komunitních fór. (Achour, a další, 2023)

3.5.1.1 SQL Injection

SQL injection je druh škodlivého softwarového útoku, který umožňuje útočnickovi získat neoprávněný přístup k databázi prostřednictvím nezabezpečeného vstupního pole na webové stránce. Běžně se jedná o vyhledávací pole nebo formulář, kam uživatel může zadávat určité dotazy pro vyhledávání nebo editaci dat. (Hranický, 2023)

Útok SQL injection je možný, pokud webová aplikace nedostatečně ověřuje vstupní data od uživatele, která mohou obsahovat SQL kód. Pokud útočník do vstupního pole zadá škodlivý SQL kód, aplikace ho vyhodnotí jako součást svého SQL příkazu a spustí ho. Tím umožní útočnickovi získat citlivá data z databáze, nebo dokonce ovládnout celou webovou aplikaci.

SQL injection útoky jsou nebezpečné, protože mohou být použity k odcizení citlivých dat a zneužití webových aplikací. Proti SQL injection útokům lze zabránit například použitím parametrizovaných dotazů nebo kontrolou vstupních dat od uživatele před tím, než se použijí v SQL dotazu. (Kulman, 2014)

3.5.2 MySQL

MySQL je open-source relační databázový systém, který umožňuje ukládání a správu dat. Byl vyvinut společností MySQL AB, která byla později zakoupena společností Oracle Corporation. MySQL je jedním z nejpobulárnějších databázových systémů na světě a často se používá pro webové aplikace a webové stránky.

Základními vlastnostmi MySQL jsou rychlost, spolehlivost, jednoduchost použití a škálovatelnost. Podporuje mnoho různých programovacích jazyků, jako je například PHP, Python, Perl a Java. MySQL je multiplatformní a může fungovat na různých operačních systémech, jako je Windows, Linux a macOS.

MySQL umožňuje ukládání dat do relačních tabulek, které jsou propojeny pomocí klíčů. Podporuje standardní dotazovací jazyk SQL (Structured Query Language) pro manipulaci s daty. MySQL také obsahuje rozsáhlou sadu funkcí pro zabezpečení, zálohování a obnovu dat, správu uživatelů a práv a replikaci dat.

Výhodou MySQL je jeho škálovatelnost a schopnost zpracování velkého objemu dat. Lze také snadno vytvářet a spravovat relační databáze a databázové aplikace. Mnoho vývojářů a organizací používá MySQL jako svou primární databázi pro webové aplikace a systémy.

Nevýhodou MySQL může být jeho omezená podpora některých funkcí a jazyků v porovnání s jinými databázovými systémy. Také může být poměrně náročný na správu a údržbu, zejména v případě velkých databázových systémů. (Moore, 2023)

3.5.3 Http protokol

HTTP (Hypertext Transfer Protocol) je základní protokol používaný pro komunikaci mezi webovými prohlížeči a webovými servery. Tento protokol umožňuje klientům (webovým prohlížečům) přistupovat k různým zdrojům (webovým stránkám) na serveru a provádět s nimi různé akce (číst, zapisovat, mazat apod.).

HTTP protokol se skládá ze tří hlavních částí:

1. Základní funkce HTTP protokolu: HTTP protokol je založen na jednoduchých požadavcích a odpovědích mezi klientem a serverem. Klient posílá požadavek na server pomocí HTTP metody (např. GET, POST, PUT, DELETE) a server odpovídá HTTP odpovědí s kódem odpovědi a obsahem odpovědi.
2. Struktura HTTP požadavku: HTTP požadavek se skládá z několika částí, včetně řádku požadavku, hlaviček a těla požadavku. Řádek požadavku obsahuje HTTP metodu, URI a verzi protokolu. Hlavičky obsahují různé informace o požadavku, jako například typ obsahu, jazyk, délka obsahu atd. Tělo požadavku obsahuje data, která se mají poslat na server.
3. Struktura HTTP odpovědi: HTTP odpověď serveru obsahuje kód odpovědi a obsah odpovědi. Kód odpovědi vrací informaci o výsledku operace, např. 200 OK pro úspěšný požadavek, 404 Not Found pro neexistující zdroj, nebo 500 Internal Server Error pro chybu na serveru. Obsah odpovědi může obsahovat různé typy dat, jako jsou HTML stránky, obrázky, JSON nebo XML soubory.

Výhodou HTTP protokolu je jeho jednoduchost a snadná implementace. Navíc je protokol nezávislý na platformě a umožňuje komunikaci mezi různými systémy. Nevýhodou

může být pomalost při přenosu větších objemů dat a zranitelnost vůči různým druhům útoků, jako jsou např. útoky typu Denial of Service (DoS). (Kučera, 2011)

HTTP protokol nabízí komunikaci mezi klientem a serverem pomocí nezašifrovaného textu. To znamená, že při použití protokolu HTTP jsou data přenášena v otevřené podobě, což může způsobit rizika, jako je například odposlech dat třetí stranou.

Na druhé straně HTTPS je zabezpečená verze protokolu HTTP. V této verzi jsou data šifrována pomocí SSL nebo TLS protokolu, což znamená, že jsou přenášena ve formě zakódovaného textu, který je těžší pro třetí strany odposlouchávat. To zlepšuje bezpečnost komunikace mezi klientem a serverem, protože data jsou chráněna proti útokům typu man-in-the-middle (útok, při kterém útočník odposlechne a modifikuje komunikaci mezi klientem a serverem).

Při použití HTTPS musí mít webový server digitální certifikát, který potvrzuje jeho identitu. Certifikát obsahuje informace o serveru, jako je jeho název, certifikační autorita, která certifikát vydala, a platnost certifikátu. Klient (webový prohlížeč) poté ověří platnost certifikátu a pokud je platný, naváže se šifrovaná komunikace mezi klientem a serverem. (Kadousková, 2021)

3.5.3.1 REST

REST (Representational State Transfer) je architektura, která se používá pro komunikaci mezi klientem a serverem pomocí webových služeb. REST definuje sadu návrhových principů, které umožňují klientovi provádět operace na serveru pomocí HTTP protokolu.

Restový požadavek se skládá z několika částí:

1. Metoda požadavku – definuje, jakou akci chceme na serveru provést. Nejčastěji používanými metodami jsou GET, POST, PUT a DELETE.
2. URI (Uniform Resource Identifier) - identifikuje zdroj na serveru, se kterým chceme komunikovat. URI se skládá z adresy serveru a cesty ke zdroji na serveru.
3. Hlavičky – obsahují metadata o požadavku, jako například typ obsahu, délka obsahu nebo autentizační údaje.
4. Tělo požadavku – obsahuje data, která chceme odeslat na server. Tělo požadavku může být prázdné, nebo může obsahovat různé typy dat, jako jsou JSON, XML nebo obrázky.

Restová odpověď serveru pak obsahuje informace o výsledku operace, kterou jsme na serveru provedli. Odpověď může obsahovat HTTP kód odpovědi (např. 200 OK nebo 404 Not Found), hlavičky a tělo odpovědi. (Malý, 2009)

3.5.4 Basic autentizace

Basic autentizace je metoda ověřování identity, která se často používá při přístupu k webovým stránkám nebo službám, jako jsou například webová API¹⁰. Při použití základní autentizace je uživatel požádán o zadání uživatelského jména a hesla, které jsou poté posílány v otevřené podobě v HTTP hlavičce.

Když uživatel zkusí přistoupit k chráněnému obsahu, server mu odešle HTTP požadavek s přípojením "WWW-Authenticate", což vyvolá v prohlížeči dotaz směrem k uživateli na jeho přihlašovací údaje. Pokud uživatel poskytne své uživatelské jméno a heslo, tak prohlížeč opět odešle požadavek na server, tentokrát s přidáním hlavičky "Authorization". Server ověří zaslané údaje a pokud jsou správné, umožní uživateli přístup k chráněnému obsahu. Pokud jsou údaje nesprávné, nebo uživatel odmítne vyplnění přístupových údajů, tak je komunikace přerušena.

Basic autentizace je velmi jednoduchá a snadno implementovatelná, ale má několik nedostatků, které ji dělají méně bezpečnou. Největším problémem je to, že uživatelské jméno a heslo jsou posílány v otevřené podobě, což znamená, že mohou být snadno zachyceny útočníkem, který odposlouchává komunikaci.

Z tohoto důvodu se základní autentizace často nepoužívá jako samostatná metoda ověřování identity, ale spíše jako doplňková metoda k dalším zabezpečovacím metodám, jako je například HTTPS nebo tokenová¹¹ autentizace. (GÜBÜR, 2022)

¹⁰ Api je aplikační programové rozhraní, které slouží k předávání dat mezi softwarovými aplikacemi definovaným způsobem. (Kodůusková, 2021)

¹¹ Tokenová autentizace je proces ověření identity uživatele pomocí tokenu namísto přímého poskytnutí přihlašovacích údajů (uživatelské jméno a heslo). Token je krátký kód nebo řetězec, který se generuje na základě přihlašovacích údajů uživatele a poté se používá jako identifikátor pro přístup k určitému zdroji nebo službě. (Doupal, 2022)

3.6 Automatizovaná regulace kotle na tuhá paliva

Pro automatizaci procesu regulace výkonu topidla obecně se nejčastěji používají termostaty, senzory teploty a elektromagnetická relé. Termostat zde slouží jako řídicí zařízení, které se zapíná a vypíná na základě naměřené teploty, a tím spíná, či rozepíná elektromagnetické relé. Senzory teploty se používají k měření aktuální teploty a relé umožňuje samotné ovládání regulace výkonu zařízení na základě signálu od termostatu. (Buildex, 2023)

3.6.1 Ekvitermní regulace

Ekvitermní regulace je typ regulace, která se používá pro udržení stálé teploty v prostředí. Je obecně tvořen jedním, nebo více regulátory, které přizpůsobují teplotu topné vody do soustavy v závislosti na momentální venkovní teplotě.

Ekvitermní regulace je velmi užitečná při udržování stálé teploty v prostředí pomocí klimatizace, nebo vytápění. Proces regulace zahrnuje měření venkovní teploty a přizpůsobení tepelného výstupu tak, aby se udržela konstantní teplota.

V současné době se ekvitermní regulace používá ve velkém množství různých systémů, jako jsou například inteligentní domácnosti, chytrá města, průmyslové procesy nebo zemědělství. Využití ekvitermní regulace má mnoho výhod, jako jsou nižší provozní náklady, snížení emisí CO₂ a zvýšení produktivity topné soustavy. (Valter, 2003)

3.6.2 Výpočet ekvitermní rovnice a křivky

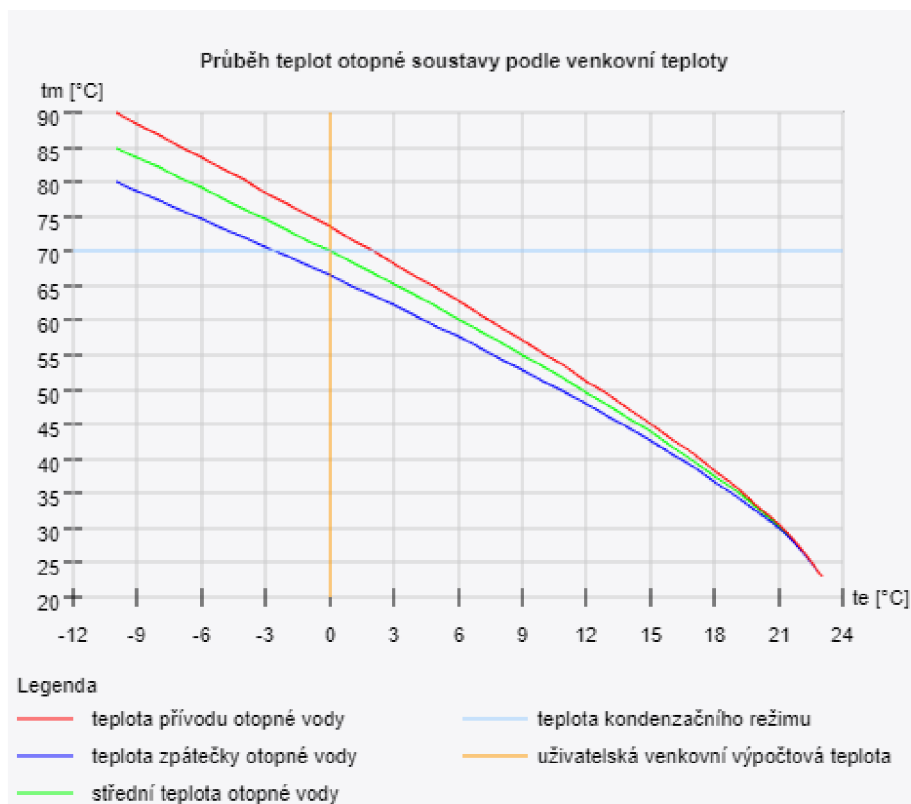
Základem ekvitermní regulace je vztah mezi aktuální teplotou venkovního vzduchu, tepelně technickými vlastnostmi topné soustavy, která teplo předává do místnosti, teplotou topné vody, která do tohoto prvku vtéká, a teplotou topné vody z prvku vytékající. Při klesající venkovní teplotě se v dané topné soustavě teplota dodávané topné vody zvyšuje, aby se do místností dostalo více tepla, které kompenzuje tepelné ztráty místnosti. (Reinberk, 2017)

$$t_m = t_i + \left(\frac{t_{w1,max} + t_{w2,max}}{2} - t_i \right) \cdot \left(\frac{t_e - t_i}{t_{e,min} - t_i} \right)^{\frac{1}{n}} \quad [^{\circ}\text{C}]$$

Obrázek 6: Rovnice pro výpočet střední teploty teplotnosné látky (Reinberk, 2017)

Proměnné vstupující do rovnice:

- t_i – vnitřní výpočtová teplota
 - Cílová teplota sledovaného objektu, nebo místnosti, ve které je umístěn teplotní senzor.
- t_e – cílová venkovní teplota
 - Cílová teplota venkovního vzduchu pro výpočet rovnice.
- $t_{w1,max}$ – maximální teplota přívodu topné vody
 - Hraniční teplota vody vtékající do topného prvku.
- $t_{w2,max}$ – maximální teplota výstupu topné vody
 - Maximální teplota vody vytékající z topného prvku.
- n – teplotní exponent soustavy
 - Exponent je určen dle typu a technických vlastností topné soustavy.



Obrázek 7: Příklad ekvitermní křivky (Reinberk, 2017)

3.6.3 Regulátor výkonu

Regulátory výkonu jsou přímočinná zařízení, které umožňují automatizovanou regulaci výkonu kotlů na tuhá paliva. Regulace výkonu je prováděna v závislosti na teplotě topné vody v kotli. Dle naměřené teploty je následně regulována vzduchová klapka tak, že při stoupající teplotě topné vody je klapka uzavírána, a při klesající teplotě topné vody je otevírána. Toto zařízení představuje prakticky jedinou možnost automatizované regulace výkonu kotle na tuhá paliva, která dokáže ušetřit v průměru až 10 % paliva. (FME VUTBR, 2016)

4 Vlastní práce

V této kapitole bude nejprve zpracována analýza řešeného problému, která stanoví současný stav systému, a vyplynou z ní požadavky na vytvoření budovaného automatizovaného systému. Následně z analýzy vznikne návrh řešení automatizované ekvitermní regulace kotle pomocí vývojové desky s mikrořadičem, servomotoru a definovaných senzorů. Dle návrhu bude naprogramován a sestaven automatizovaný systém ve fyzické podobě, který bude sbírat a odesílat data na přidruženou webovou stránku.

Webová stránka bude naprogramována v kombinaci webových technologií HTML, CSS, JavaScript, REST a PHP, a bude zobrazovat nasbíraná data do grafů. Bude obsahovat i ovládací část, pomocí které bude možné řídit automatizovaný systém na dálku. Data pro systém a webovou stránku budou uložena v MySQL databázi na serveru.

Součástí webové stránky bude API rozhraní, které bude přijímat data z teplotních senzorů a ukládat je do databáze. V odpovědi bude API endpoint odesílat případné změny v řídicím formuláři tak, aby na ně mohl automatizovaný systém reagovat. Přístup na webovou stránku bude možný pouze po autentizaci příchozího uživatele pomocí šifrovaného https připojení.

4.1 Analýza

V následující kapitole proběhne analýza současného stavu ekvitermní regulace kotle, ze kterého vyjdou požadavky pro navrhovaný automatizovaný systém.

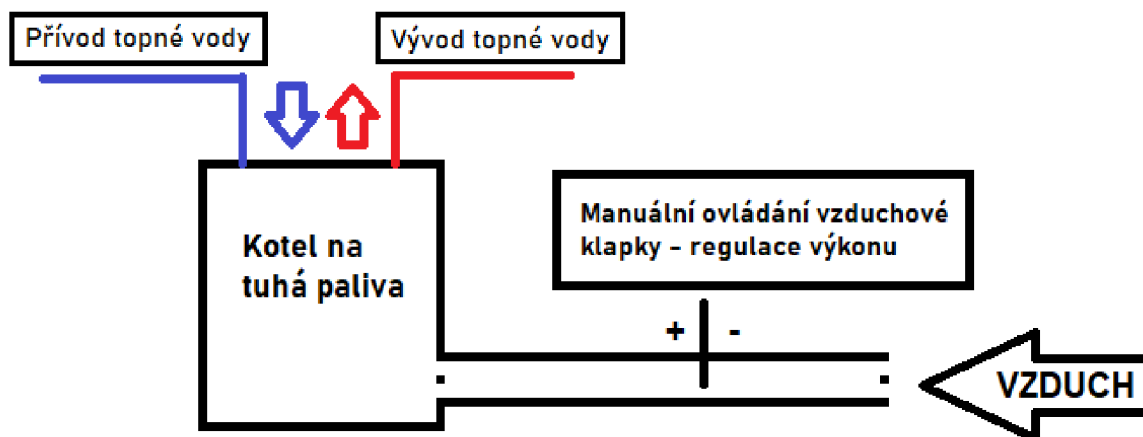
4.1.1 Současný stav

Dle níže zobrazeného diagramu můžeme vidět současný stav analyzovaného kotle. Jedná se o kotel na tuhá paliva s výměníkem topné vody. To znamená, že při spalování paliva dochází k zahřívání topné vody v kotli, a následně k její cirkulaci do systému propojených deskových topných těles. (Lyčka, 2018)

Kotle na tuhá paliva mají obecně jedinou možnost regulace topného výkonu, kterou je ovládání vzduchové klapky, která reguluje množství vzduchu vstupujícího do procesu spalování. (FME VUTBR, 2016)

V současném stavu dochází k manuální regulaci výkonu, čímž dochází k výkyvům teplotních rozsahů v topném systému. Manuální regulace je plně závislá na lidském faktoru, a jen velmi omezeně zohledňuje faktory jako jsou vnitřní a venkovní teplota, nebo teplota

topného systému. Největším omezením v manuální regulaci výkonu je časová náročnost pro člověka. K manuální regulaci tak dochází jen nahodile, čímž může docházet k nadbytečnému spalování paliva v topném systému, což má za následek zvýšené náklady na topení. V analyzovaném topném systému dochází k průměrné spotřebě 0,02 m³ za 1 hodinu provozu.



Obrázek 8: Diagram současného stavu

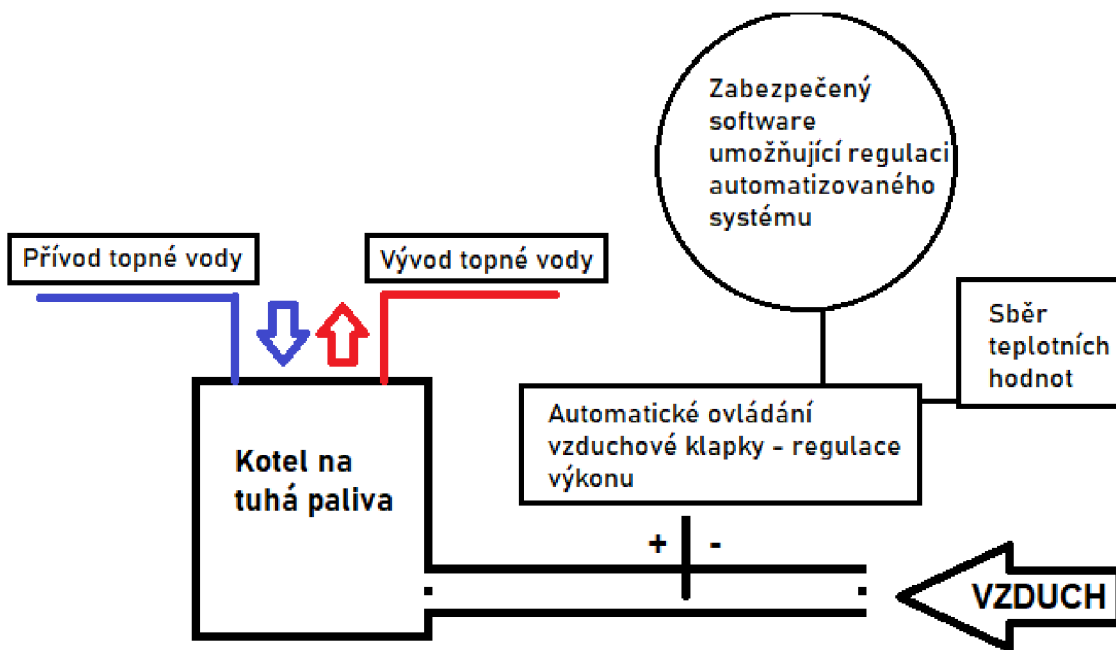
4.1.2 Požadavky

Z předcházející kapitoly je patrné, že manuální regulace výkonu kotle je neefektivní vzhledem ke spotřebě paliva, jelikož dochází k výkyvům teploty topné vody a spalování nadbytečného množství paliva.

Z toho plyne hlavní požadavek na úsporu spotřeby paliva v topném systému, čímž dojde k zefektivnění procesu spalování. Další plynoucí požadavek je na úsporu času člověka, jakožto současného regulátora výkonu. Odstraněním nutnosti manuální regulace výkonu kotle odpadne také nutnost kontrolovat, jak stavy teploty vody v topném systém, tak stavy teplot uvnitř a vně vytápěného objektu. Tím se zvyšuje i bezpečnost, jelikož nemůže dojít k přehřátí systému ohřevu topné vody z důvodu nastavení možného bezpečnostního limitu pro maximální možnou teplotu topné vody.

Při odstranění manuální regulace výkonu vyvstane požadavek na zasahování do automatizovaného systému pomocí softwarového řešení. Například při potřebě krátkodobého otevření vzduchové klapky při rozdělávání, nebo úplném uzavření vzduchové klapky například při poruše systému. Dalším požadavkem je regulace cílové teploty uvnitř

objektu z důvodu maximalizace komfortu osob. Vzniknout by tak měla softwarová možnost řízení systému na dálku. Takový softwarový přístup by pak měl poskytovat možnost zabezpečení, aby nedocházelo k neoprávněné regulaci.



Obrázek 9: Diagram požadovaného stavu

4.2 Návrh řešení

V následující kapitole bude zpracován návrh řešení automatizovaného systému. Grafický návrh řešení bude zpracován ve volně dostupné webové aplikaci Tinkercad. (Tinkercad, 2023)

Tento návrh bude zpracován pro místa s dostatečným pokrytím Wi-Fi signálem, a možností připojení do sítě internet, nicméně fungovat může i v místech bez možnosti síťového připojení, čímž odpadne možnost systém uživatelsky ovládat na dálku. V tomto modelu bude možné ovládat vývojovou desku přes síťové připojení, a také vyčítat a interpretovat nasbíraná data.

4.2.1 Seznam použitých komponent

- Vývojová deska Wemos D1 Uno WiFi ESP8266
- 3x Digitální teplotní senzor Dallas DS18B20
- Servomotor MG90S
- 1x červená LED
- 1x zelená LED
- Propojovací vodiče
- 2x 220 Ω rezistor
- 1x 4.7k Ω rezistor

4.2.2 Hardwarový popis návrhu řešení

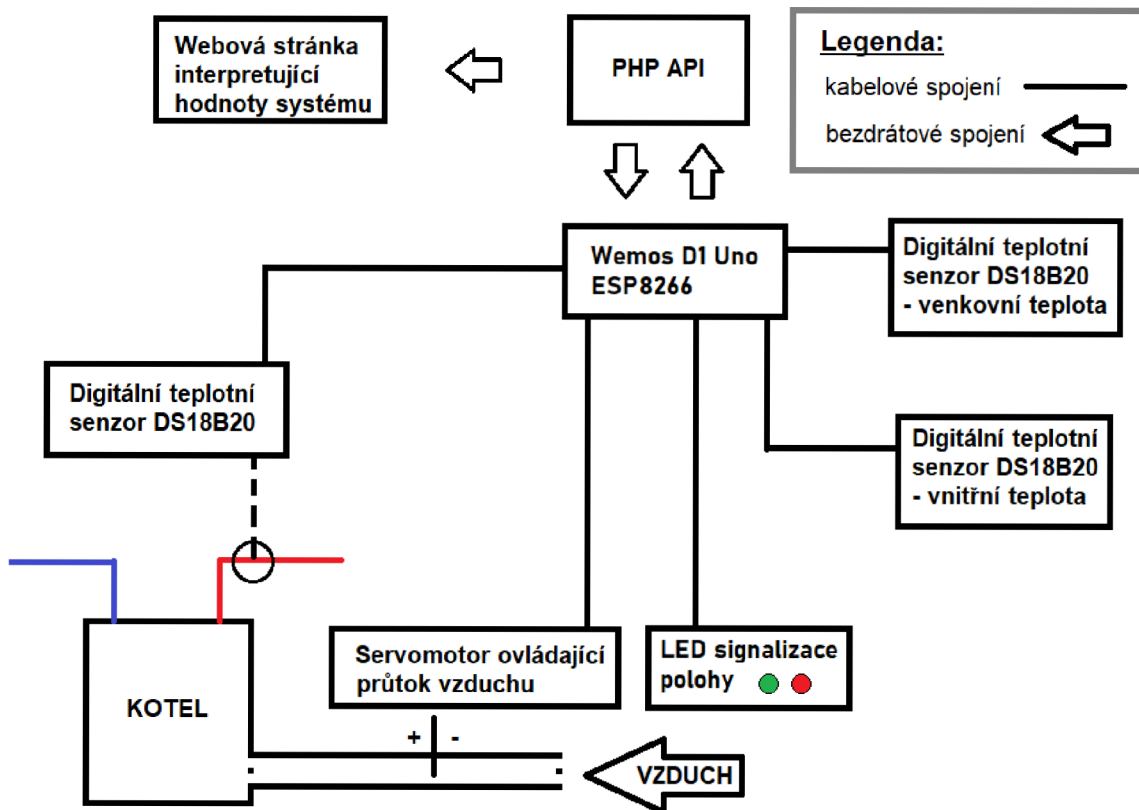
Návrh řešení se síťovou konektivitou obsahuje vývojovou desku Wemos D1 Uno WiFi s mikrokontrolérem ESP8266 pracující s provozním napětím 3,3 V na digitálních pinech. Veškeré použité komponenty jsou dle příslušných specifikací kompatibilní s řídicím digitálním signálem v rozsahu 0 – 3,3 V. Jedinou výjimku, a to pouze v napájení nikoliv řízení, má použitý servomotor. Servomotoru je kvůli splnění specifikací a maximalizaci výkonu použito napájení 5 V, přičemž řídicí signály z vývojové desky budou probíhat v rozmezí 0 – 3,3 V.

K vývojové desce s mikrokontrolérem je na digitálním pinu 4 pomocí 1 – Wire sběrnice připojena soustava digitálních teplotních senzorů s jedinečnou adresou. Unikátní adresy senzorů jsou použity pro přístup a vyčítání teplotních hodnot ze všech senzorů do globálních proměnných, čímž bude současně naměřená teplota dostupná v celé aplikaci. V zapojení je použit jeden 4,7k Ω „pull-up“ rezistor, který je vyžadován 1 – Wire sběrnici.

Mikrokontroler vyhodnotí načtené teploty z teplotních senzorů, a posléze otevře, nebo uzavře vzduchovou klapku na kotli pomocí servomotoru připojeného na digitálním pinu 9 vývojové desky. K vyhodnocení hraniční teploty pro přepnutí je použita příslušná ekvitermní rovnice soustavy.

Zelená a červená dioda signalizují polohu natočení ramena servo motoru, respektive polohu vzduchové klapky kotle. Zelená LED, připojená na digitálním pinu 6, značí polohu „otevřeno“. Červená LED, připojená na digitálním pinu 7, signalizuje polohu „zavřeno“. Obě LED mají mezi digitálním pinem a anodou (+ pól) jeden 220 Ω rezistor, kvůli snížení proudu v obvodu, a ochraně tranzistorů desky. (Voda, 2016)

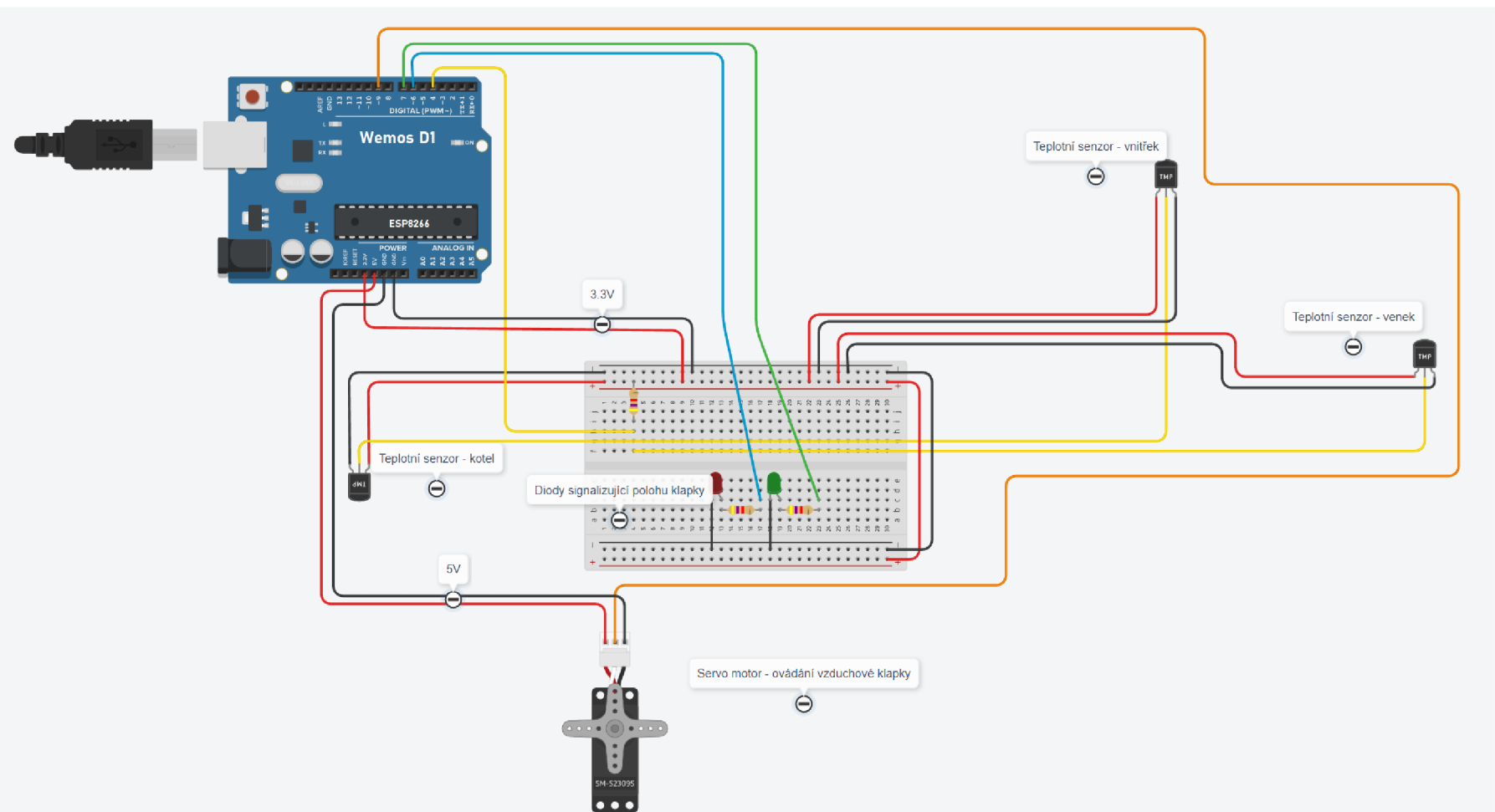
Vývojová deska umožňuje připojení k síti internet pomocí mikrokontroléru ESP8266 s podporou Wi-Fi. Nasbírané teplotní údaje budou odeslány do sítě internet na příslušný API endpoint a zpět budou vráceny řídicí informace od uživatele. Pomocí nich bude moci autorizovaný uživatel na přidružené webové stránce řídit cílovou vnitřní teplotu, otevírání a zavírání vzduchové klapky, délku cyklu měření teplot, a také délku cyklu, po kterou má platit otevření, či uzavření klapky nastavené uživatelem.



Obrázek 10: Vizualizace návrhu řešení

Obrázek 10: Vizualizace návrhu řešení obsahuje vizualizaci schématu celého navrhovaného systému.

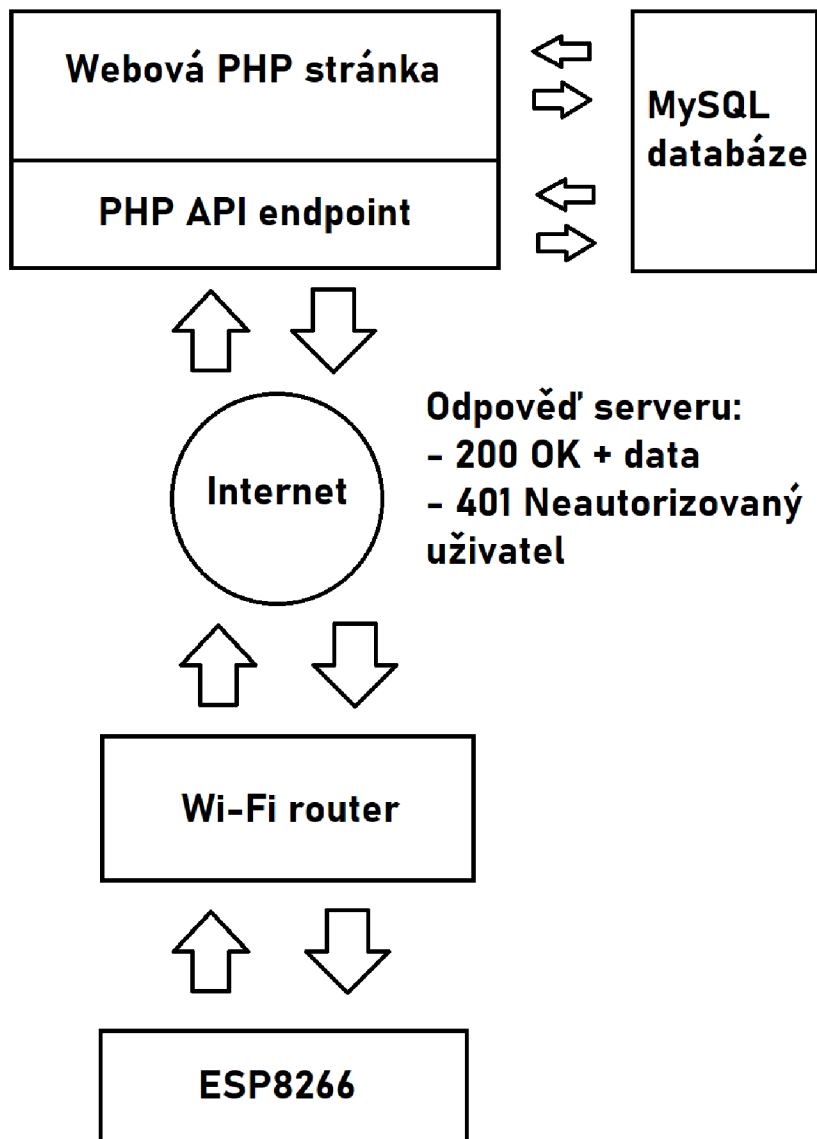
Technický detail zapojení vývojové desky, teplotních senzorů a servomotoru je dostupný na Obrázek 11: Diagram zapojení návrhu. Ten popisuje předpokládané skutečné zapojení jednotlivých prvků, včetně jejich popisu, a také napájecích vodičů, včetně vizualizace zapojení jednotlivých vodičů do digitálních pinů vývojové desky.



Obrázek 11: Diagram zapojení návrhu

4.2.3 Webové rozhraní

Pro tento návrh vznikne webová stránka v kombinaci technologií PHP, HTML, JavaScript, REST a MySQL.



Obrázek 12: Diagram informačního toku v síti

4.2.3.1 Webová stránka

Webová stránka bude obsahovat autentizaci příchozích uživatelů, a bude tak zabraňovat neautorizovaným uživatelům přistupovat do systému. Autentizace bude probíhat vůči přidružené tabulce uživatelů v databázi. Formulář přihlášení je také jediným veřejně

přístupným formulářem webové stránky, a proto je zabezpečen proti útokům typu SqlInjection použitím parametrizovaného dotazu do databáze.

Dále bude autorizovanému uživateli zobrazovat a interpretovat naměřené a zaslané teplotní hodnoty z vývojové desky. Uživateli s úrovní oprávnění pro zápis umožní také ovládat vývojovou desku na dálku změnou příslušných řídicích hodnot ve formuláři. Tato řídicí data jsou poté odeslána do vývojové desky přes API endpoint.

Webová stránka bude vytvořena na základě struktury a designu definovanému v níže uvedeném wireframu¹² obrazovek.

Přihlášení

Uživatelské jméno

Heslo

Přihlásit

Obrázek 13: Wireframe přihlašovací obrazovky

¹² Jedná se o obrazovou definici kostry webu definující jednotlivé komponenty. (Dubinská, 2022)

Přihlášený uživatel: xxx
Odhlásit

Řídicí formulář

Pozice servo motoru:

Vnitřní cílová teplota:

Teplota změny pozice servo motoru:

Časový limit platnosti změny hodnot:

Časový limit cyklu měření a odesílání dat:

Uložit změny

Vývoj teploty

GRAF

Vývoj teploty vody

GRAF

Vývoj teploty uvnitř

GRAF

Vývoj teploty venku

GRAF

Obrázek 14: Wireframe hlavní obrazovky po přihlášení

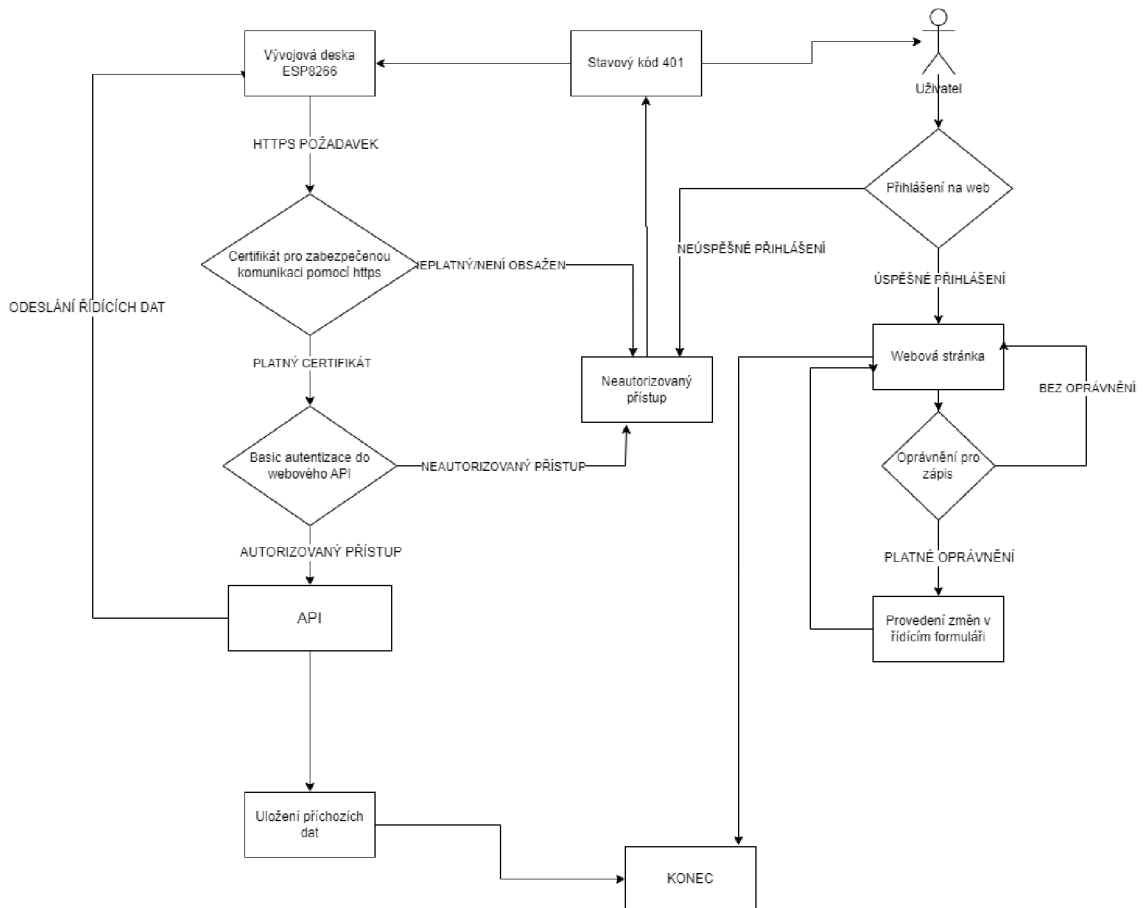
4.2.3.2 API

Webové aplikační rozhraní bude očekávat teplotní data ze všech 3 senzorů, teplotní úroveň změny polohy servomotoru, kterou systém získá výpočtem ekvitermní rovnice. V případě, že se jedná o první cyklus, po ukončení řídicího cyklu, dojde také k odeslání hodnoty polohy servomotoru z důvodu přepsání hodnoty v řídicím formuláři. Pokud by k tomu nedošlo, tak by řídicí formulář neustále vyřazoval systém automatizované ekvitermní regulace.

Obdržené hodnoty bude ukládat do databáze. Jako odpověď bude odesílat hodnoty z řídicího formuláře, které vývojová deska očekává. Endpoint bude vytvořen podle swagger¹³ dokumentace uvedené na Obrázek 16.

¹³ Swagger je open source nástroj pro návrh a tvorbu dokumentace API. (Ellis, 2022)

Součástí http požadavku na API endpoint musí být autentizační hlavička typu basic. Pokud není, či obsahuje uživatelská data, která nejsou nalezena v příslušné tabulce uživatelů v databázi, tak je daný požadavek serverem vyhodnocen jako neautorizovaný. V případě neautorizovaného požadavku bude v odpovědi vrácen pouze stavový kód odpovědi – 401, což bude značit neautorizovaný přístup.



Obrázek 15: Vývojový diagram navrhovaného komunikačního schématu

GET /api Uloží teploty a získá řídicí hodnoty systému

Endpoint uloží naměřené teplotní hodnoty a jako odpověď získá řídicí hodnoty systému

Parameters Try it out

Name	Description
hotWaterTemp * required number(\$float) (query)	Teplota vody na výstupu z kotle <input type="text" value="hotWaterTemp"/>
outsideTemp * required number(\$float) (query)	Venkovní teplota <input type="text" value="outsideTemp"/>
insideTemp * required number(\$float) (query)	Vnitřní teplota <input type="text" value="insideTemp"/>
switchTemperature * required number(\$float) (query)	Teplota, při které dochází ke změně polohy vzduchové klapky <input type="text" value="switchTemperature"/>
targetTemperature * required integer(\$int64) (query)	Cílová vnitřní teplota <input type="text" value="targetTemperature"/>
servoPosition integer(\$int32) (query)	Pozice servo motoru <input type="text" value="servoPosition"/>

Responses

Code	Description	Links
200	successful operation	No links
	Media type <input type="text" value="application/json"/> Controls Accept header. Example Value Schema <pre>{ "servoPosition": 1, "switchTemperature": 56.98, "cycleTime": 5, "masterCommandDelay": 15 }</pre>	
401	Neautorizovaný požadavek	No links

Obrázek 16: Swagger definice API endpointu (swagger, 2023)

4.2.4 MySQL databáze

V tomto návrhu bude použita databáze MySQL z důvodu vysoké kompatibility s PHP skripty přidružené webové stránky.

Databázová struktura bude obsahovat celkem tři tabulky pro ukládání dat z webového formuláře, autentizaci uživatelů, a hlavně ukládání naměřených hodnot z vývojové desky.

The image shows three screenshots of MySQL database table structures. Each screenshot displays a table name at the top, followed by its columns and their data types.

Table Name	Column Name	Data Type
temperature	timestamp	timestamp
	hotWaterTemp	float
	outsideTemp	float
	insideTemp	float
admin_z2577	id	int
	username	varchar(100)
	password	varchar(100)
	canWrite	tinyint(1)
controlvalues	id	int
	servoPosition	int
	switchTemperature	float
	cycleTime	int
	masterCommandDelay	int
	targetRoomTemperature	int

Obrázek 17: Databázová struktura

Tabulka „admin“ bude obsahovat data o uživateli s oprávněním přistupovat do webové aplikace. Sloupec „id“ obsahuje automaticky generovaný primární klíč. Sloupce „username“ a „password“ obsahují uživatelské jméno a heslo uživatele. Poslední sloupec tabulky „canWrite“ obsahuje pravdivostní hodnotu, která značí, zda má uživatel právo na úpravu řídicího formuláře, či nikoliv. Tato tabulka je zdrojová pro autentizaci uživatelů na webové stránce i v API.

Tabulka „temperature“ bude obsahovat data z teplotních senzorů, které byly odeslány do webového API. Tabulka obsahuje sloupec „timestamp“, který má automaticky generovanou časovou značku. Hodnota udává čas zápisu dat do tabulky. Sloupec „hotWaterTemp“ obsahuje údaj o teplotě výstupní vody z kotle. Sloupec „outsideTemp“ zastupuje hodnotu venkovní teploty a sloupec „insideTemp“ obsahuje údaj o vnitřní teplotě v místnosti. Sloupce obsahující teplotní hodnoty jsou datového typu „float“, jelikož hodnoty jsou ukládány s přesností na dvě desetinná místa. (Busbee , a další, 2023)

Třetí tabulka slouží pro ukládání řídicích dat systému. Sloupec „id“ obsahuje automaticky generovaný primární klíč. Sloupec „servoPosition“ reprezentuje stav servo motoru. Přípustné hodnoty jsou 0 pro zavřeno, a 1 pro otevřeno. Sloupec „switchTemperature“ obsahuje teplotní hodnotu, při které systém otevírá, nebo uzavírá vzduchovou klapku kotle. Tento údaj je neměnný z webového řídicího formuláře, jelikož je vypočítáván pomocí ekvitermní rovnice přímo v softwaru vývojové desky, takže má pouze informativní charakter. Sloupec „cycleTime“ udává dobu cyklu v minutách, při kterém dochází ve vývojové desce k měření a vyhodnocování teplot, změnu polohy servo motoru, a také k odesílání a přijímání dat z API endpointu. Sloupec „masterCommandDelay“ udává dobu v minutách, po kterou vývojová deska musí držet hodnotu servomotoru nastavenou v řídicím formuláři na webu. Po uplynutí této doby se stav vývojové desky vrátí zpět do normálního režimu ekvitermní regulace.

4.3 Implementace řešení

Důvodem pro výběr této varianty vývojové desky je její podpora Wi-Fi konektivity, jelikož je žádoucí budovaný automatizovaný systém ekvitermní regulace připojit do sítě internet, a vytvořit k němu webové rozhraní. Implementace bude rozdělena do třech kapitol.

V první kapitole bude stanovena rovnice ekvitermní křivky pro cílové řešení. Tato rovnice bude použita při programování automatizované regulace.

Ve druhé kapitole bude dle návrhu řešení naprogramována vývojová deska s mikrokontrolérem do cílového stavu. K vývojové desce budou připojeny teplotní senzory, diody a servomotor dle diagramu v návrhu Obrázek 11. Celý systém bude umístěn na kotel, kde bude připojeným servomotorem ovládat vzduchovou klapku, a tím regulovat jeho výkon. Dle návrhu pak bude automatizovaný systém vyhodnocovat data z teplotních

senzorů, která bude odesílat prostřednictvím Wi-Fi na přidružené API rozhraní webové stránky.

Ve třetí kapitole bude naprogramována webová stránka dle návrhu. Bude použita kombinace technologií HTML, JavaScript a PHP pro tvorbu webové stránky s podporou databáze MySQL. Webová stránka bude zabezpečena autentizací uživatelů, jak na straně webu, tak na straně API. Pro přístup k webové stránce bude nutné zadat přihlašovací jméno a heslo, které bude následně ověřeno v databázi v tabulce „admins“. API endpoint je zabezpečen basic autentizací. Pokud https požadavek neobsahuje autentizační hlavičku typu Basic s uživatelským jménem a heslem, tak je v požadavku vrácen status kód 401 – neautorizovaný uživatel.

Webová stránka bude navíc podporovat pouze šifrovanou HTTPS komunikaci. Bez platného certifikátu tak nebude možné na webovou stránku přistoupit.

4.3.1 Stanovení rovnice ekvitermní křivky

Pro stanovení ekvitermní křivky bude využit formulář a generátor křivky na webové stránce viz zdroj. (Reinberk, 2017)

Výpočet bude proveden dosazením níže definovaných proměnných do rovnice v kapitole Výpočet ekvitermní rovnice a křivky.

Proměnné vstupující do rovnice:

- Vnitřní výpočtová teplota: 22 °C
- Minimální venkovní výpočtová teplota: -10 °C
- Maximální teplota přívodu otopné vody: 90 °C¹⁴
- Maximální teplota zpátečky otopné vody 90 °C¹⁵
- Teplotní exponent soustavy: 1,4¹⁶
- Cílová venkovní výpočtová teplota: 0 °C

Výpočet ekvitermní rovnice pro venkovní teplotu 0 °C:

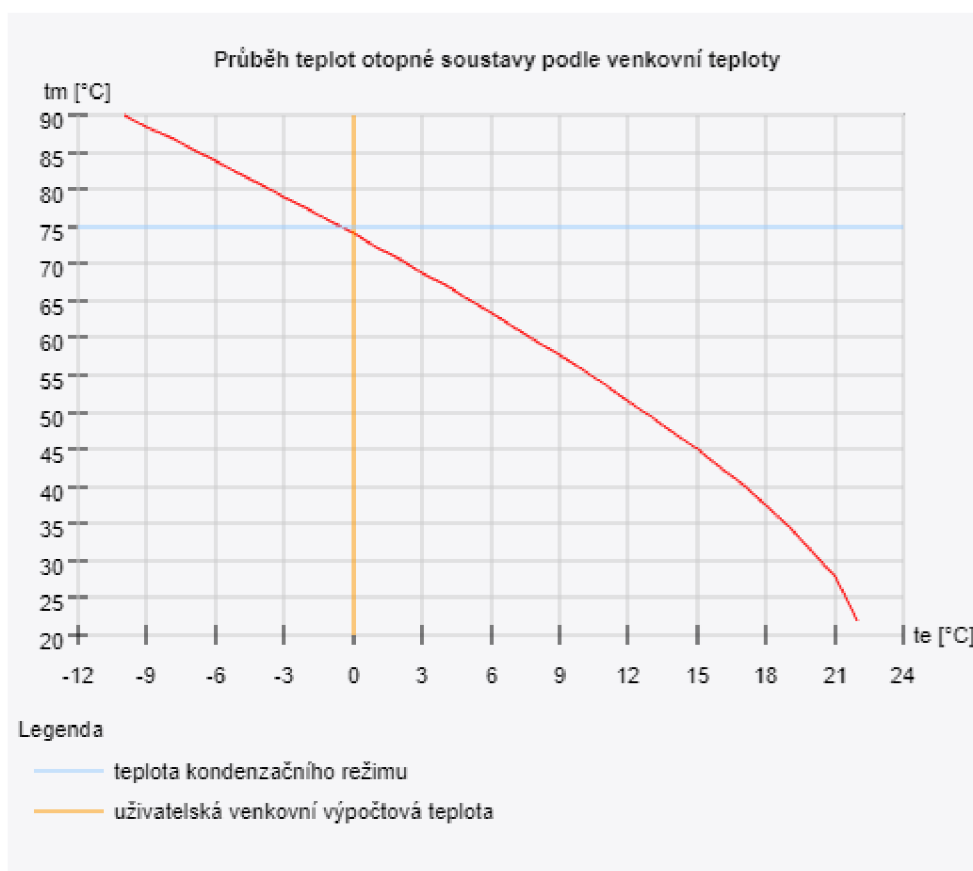
$$t_m = 22 + \left(\frac{90+90}{2} - 22 \right) * \left(\frac{0-22}{-10-22} \right)^{\frac{1}{1,3}}$$

¹⁴ Hraniční teplota topného systému, aby voda nedosáhla bodu varu 100 °C.

¹⁵ Zpátečka otopné vody může být rovna přívodu otopné vody až do jejího maxima 90 °C, jelikož v systému pracujeme pouze s jedním teplotním čidlem na kotli.

¹⁶ Stanoven dle zdroje (Reinberk, 2017). V cílovém systému jsou použita desková topná tělesa, tzn. koeficient je stanoven na 1,3.

$$t_m = 72,9 \text{ } ^\circ\text{C}$$



Obrázek 18: Stanovená ekvitemní křivka (Reinberk, 2017)

Tabulka 3: Tabulka průběhu teplot (Reinberk, 2017)

Tabulka průběhu teplot [°C]											$t_{e,user}$	
t_e	-10	-7	-4	-1	2	5	8	11	14	23		0
t_m	90	85.4	80.6	75.7	70.6	65.3	59.7	53.7	47.3	22		90

Dosažením hodnot do rovnice dle kapitoly Výpočet ekvitemní rovnice a křivky, byla stanovena ekvitemní křivka a ilustrační tabulka průběhu teplot. Z daného výpočtu vyplývá tvar křivky a hraniční teploty topné vody závislé na venkovní teplotě. Rovnice bude transformována do funkce ve zdrojovém kódu v následující kapitole.

4.3.2 Programování vývojové desky

Pro programování vývojové desky bude zvoleno prostředí open-source vývojářského prostředí Arduino IDE¹⁷ ve verzi 2.0.4¹⁸. Jedná se o software vytvořený na programování mikrokontrolérů, který má také, po doinstalování balíčků, plnou podporu zvolené vývojové desky Wemos D1 Uno WiFi ESP8266.

Vývojová deska bude naprogramována dle návrhu v odstavci **Chyba! Nenalezen zdroj odkazů.** Vznikne automatizovaný systém, který se bude skládat z vývojové desky Wemos D1 Uno, třech teplotních senzorů DS18B20, dvou LED a servo motoru MS90s. Systém bude vyhodnocovat teploty ze třech senzorů: teplotu venku, teplotu uvnitř objektu a teplotu topné vody vycházející z kotle. Na základě stanovené ekvitermní rovnice bude upravovat hraniční teplotu, při které dochází k regulaci výkonu kotle ovládním přístupu vzduchu (v tomto případě ovládním vzduchové klapky). Vybraná data budou také odeslána https požadavkem na API endpoint, kde proběhne jejich uložení do databáze. Výsledkem požadavku může být řídicí příkaz, který na zvolenou dobu odstaví ekvitermní rovnici a nastaví vzduchovou klapku do zadané polohy. Po uplynutí této doby se systém vrátí do normálního režimu fungování s ekvitermní rovnicí.

Popis vybraných softwarových metod:

```
void setTimeNtp()
{
  if (wifiMulti.run() == WL_CONNECTED) {
    delay(1000);
    configTime(3 * 3600, 0, "pool.ntp.org",time.nist.gov");
    time_t now = time(nullptr);
    while (now < 8 * 3600 * 2) {
      delay(500);
      now = time(nullptr);
    }
    struct tm timeinfo;
```

¹⁷ IDE, z anglického Integrated Development Environment, znamená vývojářské prostředí neboli software pro vývoj programů. (Strelec, 2023)

¹⁸ Arduino IDE je open-source software dostupný na adrese <https://www.arduino.cc/en/software>

```
    gmtime_r(&now, &timeinfo);
}
}
```

Po prvním spuštění při inicializaci programu na vyvíjeném systému je třeba nastavit vnitřní hodiny. Hlavním důvodem pro nastavení vnitřních hodin je synchronizace systému s přesným časem, bez něhož by nemohla fungovat validace certifikátu v průběhu HTTPS komunikace s webovým aplikačním rozhraním. Získání a nastavení času probíhá ze 2 nezávislých veřejných domén definovaných ve funkci „configtime“. Samozřejmostí je zde kontrola Wi-Fi připojení, bez něhož by přesný čas nešlo získat.

```
void serverCommunication() {
    if (wifiMulti.run() == WL_CONNECTED) {
        WiFiClientSecure client;
        HTTPClient httpsClient;
        client.setTrustAnchors(&cert);
        httpsClient.setAuthorization("diplomovaPrace1",
"dHiEpSlLoOмоваPrace1");
        httpsClient.addHeader("Content-Type", "application/json");
        String url = "?hotWaterTemp=" +
String(hotWaterTempSensorValue) + "&outsideTemp=" +
String(outdoorTempSensorValue) + "&insideTemp=" +
String(indoorTempSensorValue) + "&switchTemperature=" +
String(switchTemperature);
        if(sendControlValues)
        {
            url += "&servoPosition=" + String(setPos);
            sendControlValues = false;
        }
        httpsClient.begin(client, baseUrl + url);
    }
}
```

```

int statusCode = httpsClient.GET();
if (statusCode > 0) {
    if (statusCode == HTTP_CODE_OK) {
        handleSuccessWebServerResponse(httpsClient.getString())
;
    } else {
        Serial.printf("Got HTTP status: %d", statusCode);
        String payload = httpsClient.getString();
        Serial.println(payload);
    }
} else {
    Serial.printf("Error occurred while sending HTTP Get:
%s\n", httpsClient.errorToString(statusCode).c_str());
}
httpsClient.end();
}
}

```

Výše uvedená metoda „serverCommunication“ je používána k volání API webového rozhraní. Po ověření funkčního Wi-Fi připojení probíhá definice instancí tříd pro síťovou komunikaci. Instanci třídy WiFiClientSecure je nastaven HTTPS komunikační certifikát pro zabezpečenou komunikaci s webovým rozhraním. Do proměnné url jsou nastaveny příslušné hodnoty systému, dle swagger definice API z Obrázek 16.

Po nastavení cílové url je pomocí instance třídy HTTPClient vyvoláno spojení s webovým serverem. Po dokončení HTTPS požadavku je vrácen stavový kód, který obsahuje informaci o úspěchu, nebo neúspěchu požadavku. Při navrácení stavového kódu 200 je obdržena i odpověď ve formě JSON objektu, který obsahuje hodnoty z řídicího formuláře z webové stránky. JSON objekt, který je součástí úspěšné odpovědi serverového požadavku je definován ve výše zmíněné Swagger definici.

```

void handleSuccessWebServerResponse(String httpResponse) {
    StaticJsonDocument<500> doc;
    deserializeJson(doc, httpResponse);

    if (doc["servoPosition"].as<int>() != setPos) {
        switchServoPosition(doc["servoPosition"].as<int>());
        if (masterCommand == false)
        {
            masterCommand = true;
        }
    }
    if (doc["targetRoomTemperature"].as<int>() !=
targetTemperature) {
        targetTemperature =
doc["targetRoomTemperature"].as<int>();
    }
    if (doc["masterCommandDelay"].as<int>() * 60000 !=
masterCommandDelay) {
        masterCommandDelay = doc["masterCommandDelay"].as<int>() *
60000;
    }
    if (doc["cycleTime"].as<int>() * 60000 != actionDelay) {
        actionDelay = doc["cycleTime"].as<int>() * 60000;
    }
}

```

Metoda „handleSuccessWebServerResponse“ zpracovává úspěšnou odpověď HTTPS požadavku na webový server. Odpověď obdržena v odpovědi požadavku je datového typu string, tudíž proběhne její deserializace do statického JSON dokumentu, aby bylo možné snadno zpracovat jednotlivé hodnoty. Z nově vytvořeného JSON objektu jsou poté vyčteny jednotlivé hodnoty, a ty jsou poté nastaveny do příslušných globálních proměnných. Pokud

je hodnota nastavení servomotoru v odpovědi ze serveru (servoPosition) odlišná od současně nastavené polohy servomotoru, pak je motor nastaven na pozici definovanou v odpovědi ze serveru. Dojde tím k vyřezení ekvitermní regulace na dobu stanovenou v globální proměnné „masterCommandDelay“. Po uplynutí této doby se systém vrátí do standardního režimu provozu s ekvitermní regulací.

```
void computeSwitchTempValue()
{
    int firstFractionResult = 90 - targetTemperature;
    float secondFraction = (outdoorTempSensorValue -
targetTemperature)/(-10- targetTemperature);
    float secondFractionExponent = 1/1.4;
    switchTemperature = (targetTemperature + (firstFractionResult
* pow(secondFraction, secondFractionExponent)));
}
```

Metoda „computeSwitchTempValue“ je programová interpretace ekvitermní funkce. Metoda v každé iteraci cyklu vypočítává cílovou teplotu, při které dochází k regulaci výkonu topné soustavy. Do rovnice vstupuje globální proměnná „targetTemperature“, která obsahuje cílovou vnitřní teplotu, které má být dosaženo. Tuto teplotu nastavuje uživatel ve webovém formuláři. Pokud je nezměněna, tak v základní konfiguraci programu má hodnotu 22. Výsledkem metody je nastavení zmíněné cílové teploty, při které dochází k regulaci vzduchové klapky.

4.3.3 Programování webové stránky

Pro programování webové stránky je využita kombinace webových technologií HTML, JavaScript a PHP. Jako vývojové prostředí je použit open-source software VisualStudio Code. Webová stránka je umístěna na bezplatném webhostingu¹⁹ webzdarma.cz.

Přihlašovací obrazovka obsahuje formulář, který po zadání uživatelského jména a hesla, buď přihlásí uživatele, nebo vypíše chybovou hlášku. Autentizace uživatele probíhá vůči

¹⁹ Webhosting je pronájem prostoru pro webové stránky na serveru poskytovatele. (Sikora, 2022)

tabulce „admins“ v databázi. Pokud jsou zadané údaje shodné s údaji v tabulce, tzn. uživatel byl nalezen v databázi, proběhne přesměrování na hlavní obrazovku. Pokud na přihlašovací obrazovku přistoupí již dříve přihlášený uživatel s platným přihlášením, je automaticky přesměrován na hlavní stránku.

API endpoint obsahuje ochranné opatření proti neautorizovanému vniknutí. Každý příchozí HTTPS požadavek musí obsahovat dva zabezpečovací prvky: platný kořenový certifikát pro zabezpečenou komunikaci mezi vývojovou deskou a serverem, a autentizační hlavičku typu basic s platným uživatelským jménem a heslem. Uživatelské údaje jsou ověřeny v databázi, stejně jako při přihlašování na přihlašovací obrazovce. Pokud byl uživatel úspěšně autentizován, proběhne uložení teplotních hodnot a případně pozice servomotoru do databáze. Jako odpověď je pak odeslán objekt se současným stavem řídicího formuláře. Pokud není uživatel úspěšně autentizován, jako odpověď bude zaslán stavový kód 401 značící neautorizovaný přístup.

Hlavní obrazovka obsahuje hlavičku stránky se zobrazeným uživatelským jménem a tlačítko pro odhlášení. Pod tímto prvkem následuje řídicí formulář, jež uživateli s oprávněním pro zápis²⁰ umožňuje na dálku ovládat automatizovaný systém změnou hodnot ve formuláři. Uživatel bez oprávnění pro zápis nemá možnost ukládat případné změny ve formuláři. Pod řídicím formulářem jsou umístěny čtyři grafy, jejichž křivky reprezentují teplotní hodnoty, naměřené automatizovaným systémem. První graf je souhrnný a obsahuje křivky všech třech teplotních hodnot. Ostatní grafy obsahují pouze jednu křivku s hodnotou daného senzoru.

²⁰ Příznak ‚canWrite‘ v tabulce ‚admins‘.

Přihlášený uživatel: xxx

[Odhlásit](#)

Řídící hodnoty systému

Pozice servo motoru [0/1]:

0

Vnitřní cílová teplota [°C]:

22

Teplota změny pozice servo motoru [°C]:

42,15

Časový limit cyklu změny hodnot [minuty]:

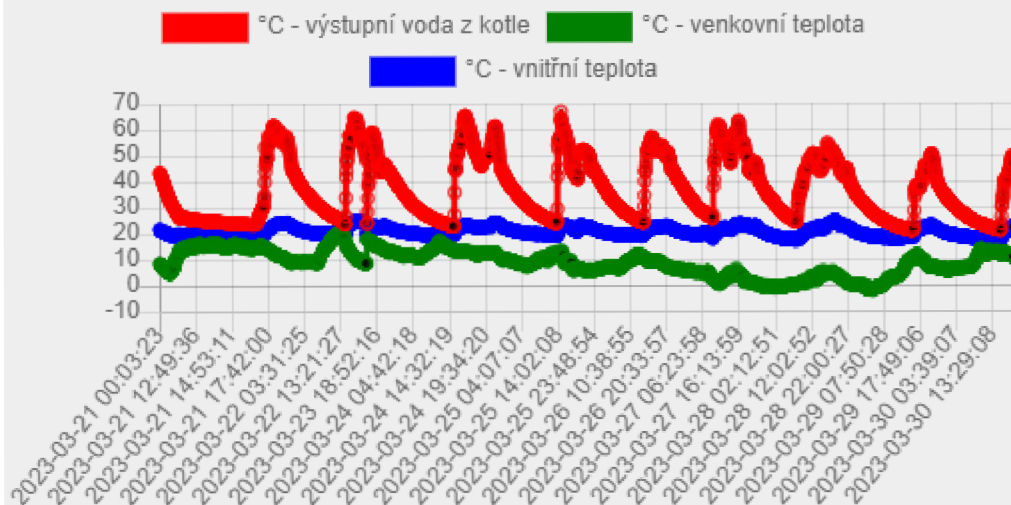
60

Časový limit cyklu měření a odesílání dat [minuty]:

5

[Uložit změny](#)

Vývoj teploty



Obrázek 19: Skutečná podoba části hlavní stránky

```

if (!isset($_SERVER['PHP_AUTH_USER']))
    header('WWW-Authenticate: Basic realm="My Realm"');
header('HTTP/1.0 401 Unauthorized');
echo '401 Unauthorized';
exit;
}

```

Výše uvedený úryvek kódu zobrazuje počáteční kontrolu webového API rozhraní, zda je v příchozím HTTPS požadavku obsažena autentizační hlavička typu basic. Pokud není, tak je uživateli vrácen stavový kód 401 – neautorizovaný přístup.

```

if (isset($_GET['hotWaterTemp']) && isset($_GET['outsideTemp'])
&& isset($_GET['insideTemp']))
    {
        $hotWaterTemp = $_GET['hotWaterTemp'];
        $outsideTemp = $_GET['outsideTemp'];
        $insideTemp = $_GET['insideTemp'];
        $connect = OpenCon();
        $query = mysqli_query($connect, "INSERT INTO
`temperature` (`hotWaterTemp`, `outsideTemp`, `insideTemp`)
VALUES('$hotWaterTemp', '$outsideTemp', '$insideTemp')");
    }

```

Výše zmíněná ukázka kódu ve webovém API slouží pro kontrolu příchozího HTTPS požadavku na to, zda obsahuje potřebná data zasláná automatizovaným systémem. Pokud ano, tak jsou následně uložena do příslušné databázové tabulky.

```

$query = $connect->prepare('SELECT * FROM `admin` where username = ?');
$query->bind_param('s', $username);
$query->execute();
$result = $query->get_result();

```

Úryvek kódu, zobrazený výše, je proveden po pokusu o přihlášení uživatele na přihlašovací stránce. Je vytvořen parametrizovaný dotaz do databáze, zda uživatel se zadaným jménem existuje. Parametrizovaný dotaz chrání databázi před škodlivými útoky typu SqlInjection.

```
<canvas class="row" id="generalChart"
style="width:100%;"></canvas>
<script>
  const c1 = document.getElementById('generalChart');
  new Chart(c1, { type: "line",
data: {
  labels: <?php echo json_encode($timeStampArray); ?>,
  datasets: [{
    label: '°C - výstupní voda z kotle',
    data: <?php echo json_encode($hotWaterTempArray); ?>,
    borderColor: "red", fill: false
  }, {
    label: '°C - venkovní teplota',
    data: <?php echo json_encode($outsideTempArray); ?>,
    borderColor: "green", fill: false
  }, {
    label: '°C - vnitřní teplota',
    data: <?php echo json_encode($insideTempArray); ?>,
    borderColor: "blue", fill: false
  }
  ]
},
options: {
  legend: {display: true}
}
});
</script>
```

Pro vyobrazení grafů na hlavní stránce je použita JavaScriptová knihovna „Chart“. Ve výše uvedené ukázce hlavního grafu stránky je vypsána definice grafu, do nějž vstupují hodnoty ze všech tří teplotních senzorů automatizovaného systému. Jedná se o liniový typ grafu, kde je v parametru „data“ předáno pole teplotních hodnot odpovídajících danému senzoru. V nastavení „options“ je aktivováno zobrazení popisků jednotlivých os pro přehlednost.

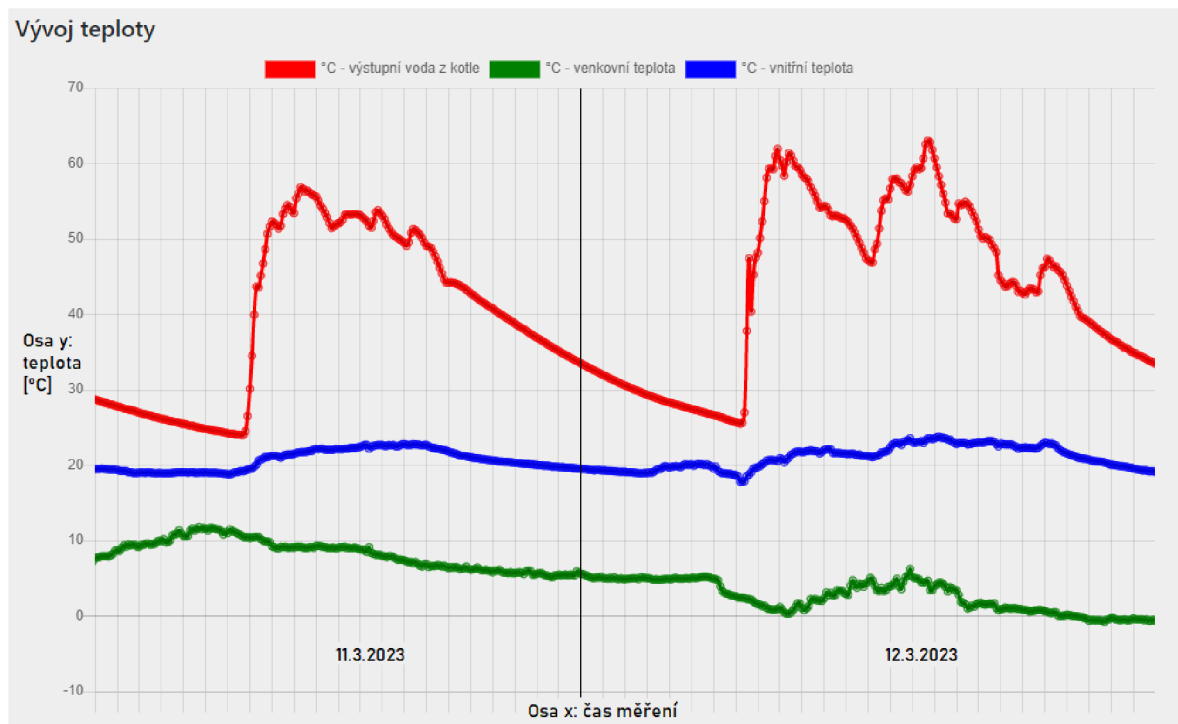


Obrázek 20: Fotodokumentace skutečného umístění servomotoru, který pomocí táhla reguluje pozici vzduchové klapky na topné soustavě

5 Zhodnocení výsledků

Hlavním cílem práce je zjistit, zda má automatizované řešení ekvitermní regulace kotle pozitivní vliv na spotřebu paliva a účinnost topné soustavy.

Pro ilustraci naměřených dat použijeme Obrázek 21 grafu, vygenerovaný na naprogramované webové stránce. Data pro tento graf jsou k dispozici v externí příloze této práce.



Obrázek 21: Graf vývoje teplot²¹

Z grafu vyplývá, že při použití automatizované regulace docházelo méně často k výkyvům teplot topné vody. Výkyvy teplot byly také v mnohem menší intenzitě pohybující se mezi 3–6 °C. Při manuální regulaci docházelo k výkyvům teplot topné vody 5–15 °C.

Také lokální extrémní hodnoty při automatizované regulaci nedosahovaly tak vysokých hodnot. Maximální hodnota extrému teploty topné vody dosahovala dle dat 56,88 °C. Při manuální regulaci se maximální hodnota extrému dle dat vyšplhala na 63,06 °C.

²¹ Levá část grafu odpovídá datům ze dne 11. 3. 2023 – manuální regulace výkonu. Pravá část grafu zobrazuje křivku pro data z 12. 3. 2023 – automatizovaná regulace výkonu.

Vývoj vnitřní teploty u automatizované regulace můžeme dle dat posoudit jako lineárnější, s minimálními výkyvy teploty, oproti datům z manuální regulace. Při automatizované regulaci docházelo k udržování nastavené cílové teploty 22 °C, s odchylkou do maxima 0,81 °C a bez klesající tendence. Zato u dat vnitřní teploty při manuální regulaci docházelo k větším odchylkám obsahujícím i klesající tendenci. Docházelo k výkyvům teplot od 20,38 °C až do 23,81 °C.

Tabulka 4: Spotřeba paliva v topné soustavě

<i>Typ regulace</i>	<i>Automatizovaná regulace výkonu</i>	<i>Manuální regulace výkonu</i>
<i>Datum</i>	11. 3. 2023	12. 3. 2023
<i>Spotřeba paliva za 1 hodinu²²</i>	0,018 m ³	0,02m ³

$$U = 100 - \left(\left(\frac{0,018}{0,02} \right) * 100 \right)^{23}$$

$$U = 10,71 \%$$

Dle výše uvedené Tabulka 4: Spotřeba paliva v topné soustavě, obsahující naměřené údaje o spotřebě paliva²⁴ za 8 hodin a rovnice výpočtu, došlo při automatizované regulaci k úspoře paliva 12,5 % oproti manuální regulaci.

Výsledkem implementace automatizovaného systému ekvitermní regulace na kotel bylo podle naměřených a vypočítaných hodnot dosaženo úspory paliva, nicméně při měření spotřeby paliva musíme brát v úvahu nemožnost přesného stanovení hodnot. Proto jsou výsledky spíše orientační.

²² Měření bylo prováděno v co možná nejméně teplotně rozdílné dny.

²³ Výpočet podle (Ozomová, 2022)

²⁴ Palivem je zde myšleno neštípané borové dřevo v tloušťce průměru 6-9 cm.

6 Závěr

Tato práce se věnovala sestavení automatizovaného systému ekvitermní regulace kotle. Cílem bylo zjistit, zda má automatizovaná regulace výkonu kotle pomocí vzduchové klapky pozitivní vliv na spotřebu paliva, a tím i zvýšení efektivity kotle. Práce je rozdělena na teoretickou a praktickou část.

Teoretická část se zabývala obecným popisem technologií využívaných v oblasti automatizace a chytré domácnosti. Byly popsány jednotlivé v praxi používané technologie bezdrátové komunikace a technologie Iot a mikrořadičů. Dále byly popsány hardwarové komponenty, a také webové a databázové technologie použité v praktické části práce. Definován byl také způsob výpočtu ekvitermní rovnice potřebné pro vytvoření automatizovaného systému.

Praktická část se zabývala sestavením a naprogramováním automatizovaného systému ekvitermní regulace kotle. Byla sestavena analýza současného řešení, z níž vzniklo několik funkčních požadavků. Následně byl vytvořen návrh řešení, který obsahoval vývojovou desku Wemos D1 UNO. Ta obsahuje v základní konfiguraci síťový modul, jež byl předpokladem pro zajištění funkční síťové komunikace.

Vývojová deska byla naprogramována ve vývojovém prostředí Arduino IDE pomocí programovacího jazyka C++. Zdrojový kód pro vývojovou desku byl vložen do externí přílohy této práce, přičemž důležité úryvky kódu byly uvedeny v příslušné kapitole.

Následně byla dle návrhu naprogramována webová stránka, pomocí které lze zobrazovat a interpretovat naměřené teplotní hodnoty. Dále je možné ovládat automatizovaný systém na dálku pomocí změn v řídicím formuláři. Webová stránka má vystaven jeden API endpoint sloužící pro komunikaci mezi automatizovaným systémem a webovým serverem. Mezi systémy jsou data předávána obousměrně. Z automatizovaného systému jsou odesílána data o teplotách a pozici vzduchové klapky na kotli. Tato, a veškerá ostatní data důležitá pro chod systému jsou uložena v databázi. Zpět jsou odesílána data a případné změny v řídicím formuláři, na které automatizovaný systém v případě změny reaguje.

Webová stránka je zabezpečena nutností použití zabezpečené https komunikace, a kontrolou proti neautorizovanému přístupu, což má za cíl zabránit neoprávněné manipulaci se systémem. Webová část obsahuje přihlašovací formulář, kde je kontrolováno jméno a heslo uživatele vůči uživatelům uloženým v databázi. Api část je zabezpečena vyžadováním

povinné autentizační hlavičky typu basic, kde probíhá stejná kontrola jako v předchozím případě. Zdrojový kód webové stránky byl umístěn do externí přílohy této práce.

Po spuštění automatizovaného systému ekvitermní regulace na kotli, a publikaci webové stránky, byla nasbírána data o chování systému. Pro zjištění, zda aplikování automatizovaného systému na kotel zapříčiní úsporu paliva bylo provedeno měření spotřeby topidla během provozu s automatizací, a bez automatizace. Výsledkem měření byla 10,71% úspora paliva při použití automatizovaného systému ekvitermní regulace výkonu kotle, vůči manuální regulaci výkonu. Z toho vyplývá, že použití definované automatizované ekvitermní regulace na kotli má vliv na úsporu paliva a je ekonomicky výhodné ho použít. Nicméně provedené měření je nutné brát pouze ilustrativně, jelikož není možné dosáhnout vždy stejného vývoje venkovní teploty pro jednotlivé dny. Hlavním úspěchem této práce byla úspěšná implementace automatizované ekvitermní regulace na kotel.

7 Bibliografie

- Admin. 2014.** UART (USART) – komunikujte sériově po dvou vodičích. *zavavov.cz*. [Online] 23. 4 2014. [Citace: 26. 3 2023.] <http://www.zavavov.cz/cz/elektrotechnika/komunikacni-sbernice/67-uart-usart-komunikujte-seriove-po-dvou-vodicich/>.
- Achour, Mehdi , a další. 2023.** PHP Manual. *php.net*. [Online] PHP Group, 2023. [Citace: 25. 3 2023.] <https://www.php.net/manual/en/>.
- Arduino. 2018.** What is Arduino? *arduino.cc*. [Online] 5. 2 2018. [Citace: 22. 3 2023.] <https://www.arduino.cc/en/Guide/Introduction>.
- Arduino.cc. 2023.** Arduino Uno Rev3. *store.arduino.cc*. [Online] 2023. [Citace: 22. 3 2023.] <https://store.arduino.cc/products/arduino-uno-rev3>.
- Buildex. 2023.** Automatizace kotlů na tuhá paliva: jak zefektivnit vytápění kotlem TT? *buildex.techinfus.com*. [Online] 2023. [Citace: 26. 3 2023.] <https://buildex.techinfus.com/cs/kotly/gazovjy-kotel-perepady-temperatury-vody.html>.
- Busbee , Kenneth Leroy a Braunschweig, Dave . 2023.** Floating-Point Data Type. *press.rebus.community*. [Online] 2023. [Citace: 22. 3 2023.] <https://press.rebus.community/programmingfundamentals/chapter/floating-point-data-type/>.
- Čepička, David. 2009.** Základy technologie Bluetooth: původ a rozsah funkcí. *pcworld.cz*. [Online] 10. 2 2009. [Citace: 19. 3 2023.] <https://www.pcworld.cz/clanky/zaklady-technologie-bluetooth-puvod-a-rozsah-funkci/>.
- Čermák, Ladislav. 2021.** Co je OTA? Jak funguje? Co umí a co neumí? Proč jej chtít? *fdrive.cz*. [Online] 21. 10 2021. [Citace: 26. 3 2023.] <https://fdrive.cz/clanky/co-je-ota-jak-funguje-co-umi-a-co-neumi-proc-jej-chtit-podrobne-informace-7914>.
- Datart. 2021.** VŠE, CO JSTE CHTĚLI VĚDĚT O CHYTRÉ DOMÁCNOSTI. *datart.cz*. [Online] 15. 8 2021. [Citace: 19. 3 2023.] <https://www.datart.cz/novinky/technologie/chytra-domacnost>.
- Doupal, František. 2022.** Spočívá budoucnost autentizace v tokenech? *rmol.cz*. [Online] 5. 9 2022. [Citace: 26. 3 2023.] <https://www.rmol.cz/novinky/spociva-budoucnost-autentizace-v-tokenech>.
- Dubinská, Lída. 2022.** CO JE WIREFRAME WEBU, PROČ HO POTŘEBUJETE A JAK HO VYTVOŘIT? *rascasone.com*. [Online] 10. 11 2022. [Citace: 23. 3 2023.] <https://www.rascasone.com/cs/blog/co-je-wireframe-predstavujeme-5-duvodu-proc-je-pro-klienty-drateny-model-dulezity>.
- Dvořáček, Ing. Vladimír. 2009.** LED je zkratka pro anglický výraz "Light Emitting Diode", tedy diodu vysílající světlo. LED je elektronický součástka, která se využívá k vytvoření světelných zdrojů s vysokou účinností a nízkou spotřebou energie. *odbornecasopisy.cz*. [Online] 5 2009. [Citace: 22. 3 2023.] <http://www.odbornecasopisy.cz/res/pdf/39810.pdf>.
- Eldin. 2023.** Co dělá firmware? *soubory.info*. [Online] 2023. [Citace: 26. 3 2023.] <https://soubory.info/app/973/co-dela-firmware>.
- Ellis, Danielle. 2022.** What is Swagger? A Beginner's Guide. *blog.hubspot.com*. [Online] 26. 7 2022. [Citace: 23. 3 2023.] <https://blog.hubspot.com/website/what-is-swagger>.
- Faltus, Bc. Robert. 2023.** Zapojení dvou a více teplotních senzorů DS18B20. *navody.drateg.cz*. [Online] 2023. [Citace: 22. 3 2023.] <https://navody.drateg.cz/technikuv-blog/zapojeni-teplotniho-senzoru-ds18b20.html>.
- faranux.com. 2023.** OTA WeMos D1 CH340 WiFi Development Board ESP8266 ESP-12E For Arduino BRD55 , R23. <https://www.faranux.com/>. [Online] 2023. [Citace: 22. 3

2023.] <https://www.faranux.com/product/ota-wemos-d1-ch340-wifi-development-board-esp8266-esp-12e-arduino/>.

FME VUTBR. 2016. Možnosti regulace vytápění v rodinných domcích. *web.archive.org*. [Online] 5. 3 2016. [Citace: 26. 3 2023.] <https://web.archive.org/web/20070604092320/http://dt.fme.vutbr.cz/enviro/RegVytapeni/tpreg.htm>.

Grusin, Mike. 2023. Serial Peripheral Interface (SPI). *learn.sparkfun.com*. [Online] 2023. [Citace: 26. 3 2023.] <https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all>.

GÜBÜR, Koray Tuğberk. 2022. HTTP Authentication: Basic Authentication. *holisticseo.digital*. [Online] 16. 3 2022. [Citace: 26. 3 2023.] <https://www.holisticseo.digital/technical-seo/http-header/http-authentication/>.

Hareendran, T. K. 2021. Tower Pro MG90S Micro Servo. *electroschematics.com*. [Online] 18. 9 2021. [Citace: 25. 3 2023.] <https://www.electroschematics.com/micro-servo/>.

Hranický, Jan. 2023. Lekce 2 - Technika útoku SQL injection. *itnetwork.cz*. [Online] 2023. [Citace: 26. 3 2023.] <https://www.itnetwork.cz/php/bezpecnost/technika-utoku-sql-injection>.

Chuděj, Martin. 2023. Popis a sestavení funkce LED diod. *wulfstan.cekuj.net*. [Online] 2023. [Citace: 22. 3 2023.] <http://wulfstan.cekuj.net/2019/01/popis-a-sestaveni-funkce-led-diod/>.

J. Koton, P. Číka, V. Křivánek. 2006. Standard nízkorychlostní bezdrátové komunikace ZigBee. *fel.cvut.cz*. [Online] 18. 4 2006. [Citace: 19. 3 2023.] <http://access.fel.cvut.cz/rservice.php?akce=tisk&cislocikanku=2006032001>.

Jelič, Pavel. 2023. Porovnání hlasových asistentů: Siri, Alexa a Google Assistant. *zbozi.cz*. [Online] 2. 2 2023. [Citace: 2023. 3 22.] <https://www.zbozi.cz/magazin/c/srovnani-hlasovych-asistentu/>.

Kad'ousková, Barbora. 2021. HTTPS V KOSTCE: CO TO JE, JAK FUNGUJE A JAK NA NĚJ PŘEJÍT. *rascasone.com*. [Online] 17. 11 2021. [Citace: 26. 3 2023.] <https://www.rascasone.com/cs/blog/co-je-https-http-ssl-tls>.

Kod'ousková, Barbora. 2021. CO JE TO API A JAKÉ JSOU MOŽNOSTI JEHO VYUŽITÍ? *rascasone.com*. [Online] 17. 11 2021. [Citace: 23. 3 2023.] <https://www.rascasone.com/cs/blog/co-je-api>.

— . 2022. INTERNET VĚCÍ (IOT): DEFINICE, PŘÍKLADY VYUŽITÍ, PRODUKTY. *ascasone.com*. [Online] 1. 7 2022. [Citace: 22. 3 2023.] <https://www.rascasone.com/cs/blog/iot-internet-veci-definice-produkty-historie>.

Kučera, František. 2011. Protokol HTTP. *zdrojak.cz*. [Online] 12. 9 2011. [Citace: 26. 3 2023.] <https://zdrojak.cz/clanky/protokol-http/>.

Kulman, Igor. 2014. SQL Injection pre každého. *zdrojak.cz*. [Online] 18. 7 2014. [Citace: 23. 3 2023.] <https://zdrojak.cz/clanky/sql-injection-pre-kazdeho/>.

Lyčka, Ing. Zdeněk. 2018. Pohledem znalce – kde je hranice mezi kamny s výměníkem a teplovodním kotlem? *vytapeni.tzb-info.cz*. [Online] 9. 10 2018. [Citace: 26. 3 2023.] <https://vytapeni.tzb-info.cz/18020-pohledem-znalce-kde-je-hranice-mezi-kamny-s-vymenikem-a-teplovodnim-kotlem>.

Malý, Martin. 2009. REST: architektura pro webové API. *zdrojak.cz*. [Online] 3. 8 2009. [Citace: 26. 3 2023.] <https://zdrojak.cz/clanky/rest-architektura-pro-webove-api/>.

Moore, Lindsay. 2023. MySQL. *techtarger.com*. [Online] 2023. [Citace: 22. 3 2023.] <https://www.techtarger.com/searchoracle/definition/MySQL>.

Ozomová, Adéla. 2022. Jak na výpočet procent? Pomůže třeba online kalkulačka procenta. *vimvic.cz*. [Online] 17. 5 2022. [Citace: 24. 3 2023.] <https://www.vimvic.cz/clanek/jak-vypocitat-procenta-priklady>.

pajenicko.cz. 2023. Digitální čidlo teploty Dallas DS18B20. *pajenicko.cz*. [Online] 2023. [Citace: 22. 3 2023.] <https://pajenicko.cz/digitalni-cidlo-teploty-dallas-ds18b20>.

Pavel Kasík, Václav Nývlt. 2017. Velký test digitálních asistentek: překvapily inteligenci i neschopností. *idnes.cz*. [Online] 13. 4 2017. [Citace: 22. 3 2023.] https://www.idnes.cz/technet/technika/test-digitalni-asistentky-amazon-echo-alexa-google-home.A170410_074844_tec_technika_pka.

Redakce. 2000. Stručný popis sběrnice I2C a její praktické využití k připojení externí eeprom 24LC256 k mikrokontroléru PIC16F877. *vyvoj.hw.cz*. [Online] 20. 5 2000. [Citace: 26. 3 2023.] <https://vyvoj.hw.cz/navrh-obvodu/strucny-popis-sbernice-i2c-a-jeji-prakticke-vyuziti-k-pripojeni-externi-eeprom-24lc256>.

Reinberk, Ing. Zdeněk. 2017. Výpočet a graf ekvitemní křivky. *vytapeni.tzb-info.cz*. [Online] 8. 3 2017. [Citace: 23. 3 2023.] <https://vytapeni.tzb-info.cz/tabulky-a-vypocty/50-vypocet-a-graf-ekvitemni-krivky>.

Reschke, J. 2015. The 'Basic' HTTP Authentication Scheme. *rfc-editor.org*. [Online] 9 2015. [Citace: 22. 3 2023.] <https://www.rfc-editor.org/rfc/rfc7617>.

Resler, Bc. Tomáš. 2019. VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ. *vut.cz*. [Online] 16. 5 2019. [Citace: 19. 3 2023.] https://www.vut.cz/www_base/zav_prace_soubor_verejne.php?file_id=190795.

Rouse, Margaret. 2016. Surface-Mount Device. *technopedia.com*. [Online] 27. 12 2016. [Citace: 26. 3 2023.] <https://www.techopedia.com/definition/12371/surface-mount-device-smd>.

Řehák, Jan. 2003. Co je to WiFi - úvod do technologie. *vyvoj.hw.cz*. [Online] 21. 6 2003. [Citace: 19. 3 2023.] <https://vyvoj.hw.cz/produkty/ethernet/co-je-to-wifi-uvod-do-technologie.html>.

Sikora, Marek. 2022. Co je to webhosting. *maxsico.cz*. [Online] 14. 9 2022. [Citace: 23. 3 2023.] <https://maxsico.cz/co-je-to-webhosting/>.

Strelec, Ing. Michal. 2023. Co je to IDE - vývojové prostředí při vývoji softwaru na míru. *strelc.pro*. [Online] 2023. [Citace: 23. 3 2023.] <https://www.strelc.pro/slovník-vyvojare/co-je-to/ide-vyvojove-prostredi>.

swagger. 2023. Swagger editor. *swagger.io*. [Online] 23. 3 2023. [Citace: 23. 3 2023.] <https://editor-next.swagger.io/>.

Štráfelda, Jan. 2023. Open source. *strafelda.cz*. [Online] 2023. [Citace: 26. 3 2023.] <https://www.strafelda.cz/open-source>.

Team Tesca. 2021. What is a Microcontroller, Types, Applications, and How does it Work? *tescaglobal.com*. [Online] 29. 4 2021. [Citace: 26. 3 2023.] <https://www.tescaglobal.com/blog/what-is-a-microcontrollers-and-how-does-it-work/>.

Thakkar, Megha. 2022. What Is Assembly Language? A Quick Overview. *sectigostore.com*. [Online] 3. 5 2022. [Citace: 26. 3 2023.] <https://sectigostore.com/blog/what-is-assembly-language/>.

Tinkercad. 2023. Tinkercad. *tinkercad.com*. [Online] 2023. [Citace: 22. 3 2023.] <https://www.tinkercad.com/dashboard?type=circuits&collection=designs>.

Václavík, Miroslav. 2021. Co je Apple HomeKit? *czc.cz*. [Online] 20. 9 2021. [Citace: 22. 3 2023.] <https://www.czc.cz/geek/co-je-apple-homekit/clanek>.

Valter, Jaroslav. 2003. Ekvitemní regulace. *valter.byl.cz*. [Online] 29. 10 2003. [Citace: 22. 3 2023.] https://valter.byl.cz/sites/default/files/soubory/ekvitem_0.pdf.

Voda, Zbyšek. 2019. ARDUINO POD POKLIČKOU: JAK FUNGUJE MIKROKONTROLÉR. *hwkitchen.cz*. [Online] 21. 3 2019. [Citace: 26. 3 2023.] <https://bastlirna.hwkitchen.cz/arduino-pod-poklickou-jak-funguje-mikrokontroler/>.

— . **2016.** PROČ MUSÍM U LED POUŽÍT REZISTOR? *bastlirna.hwkitchen.cz*. [Online] 17. 2 2016. [Citace: 22. 3 2023.] <https://bastlirna.hwkitchen.cz/proc-musim-u-led-pouzit-rezistor/>.

Webster, K. 2015. 4. díl o Arduinu – Shieldy. *phgame.cz*. [Online] 5. 3 2015. [Citace: 26. 3 2023.] https://phgame.cz/PHGame_serialy/serialy/zaciname-s-arduinem/4-dil-o-arduinu-shieldy/.

Wostl, Marek. 2019. Co je to mikrokontrolér? *blog.zonepi.cz*. [Online] 27. 9 2019. [Citace: 22. 3 2023.] <https://blog.zonepi.cz/maker-uno-ve-vyuce-1-5-co-je-to-mikrokontroler/>.

8 Seznam obrázků, tabulek, grafů a zkratk

8.1 Seznam obrázků

Obrázek 1: Diagram rozložení pinů desky Arduino Uno R3 (Arduino.cc, 2023)	20
Obrázek 2: Diagram rozložení pinů desky Wemos D1 Uno WiFi ESP8266 (faranux.com, 2023).....	22
Obrázek 3: Diagram rozložení pinů senzoru DS18B20 (pajenicko.cz, 2023)	24
Obrázek 4: Diagram zapojení vodičů servomotoru MG90S (Hareendran, 2021)	24
Obrázek 5: Diagram LED (Chuděj, 2023).....	26
Obrázek 6: Rovnice pro výpočet střední teploty teplotnosné látky (Reinberk, 2017)	32
Obrázek 7: Příklad ekvitemní křivky (Reinberk, 2017).....	32
Obrázek 8: Diagram současného stavu.....	35
Obrázek 9: Diagram požadovaného stavu	36
Obrázek 10: Vizualizace návrhu řešení	38
Obrázek 11: Diagram zapojení návrhu.....	39
Obrázek 12: Diagram informačního toku v síti	40
Obrázek 13: Wireframe přihlašovací obrazovky	41
Obrázek 14: Wireframe hlavní obrazovky po přihlášení	42
Obrázek 15: Vývojový diagram navrhovaného komunikačního schématu.....	43
Obrázek 16: Swagger definice API endpointu (swagger, 2023).....	44
Obrázek 17: Databázová struktura	45
Obrázek 18: Stanovená ekvitemní křivka (Reinberk, 2017)	48
Obrázek 19: Skutečná podoba části hlavní stránky.....	55
Obrázek 20: Fotodokumentace skutečného umístění servomotoru, který pomocí táhla reguluje pozici vzduchové klapky na topné soustavě	58
Obrázek 21: Graf vývoje teplot.....	59

8.2 Seznam tabulek

Tabulka 1: Specifikace Arduino Uno R3 (Arduino.cc, 2023)	21
Tabulka 2: WEMOS D1 UNO WiFi ESP8266 (faranux.com, 2023)	23
Tabulka 3: Tabulka průběhu teplot (Reinberk, 2017)	48
Tabulka 4: Spotřeba paliva v topné soustavě.....	60

