

Sem vložte zadání Vaší práce.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA ČÍSLICOVÉHO NÁVRHU



Diplomová práce

Nadřazený systém pro správu garáže

Bc. Ondřej Červenka

Vedoucí práce: Ing. Martin Daňhel

5. listopadu 2017

Poděkování

Děkuji panu Ing. Martinu Daňhelovi za čas, který mi věnoval a zejména za cenné rady a odborné vedení mé bakalářské práce.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 5. listopadu 2017

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2017 Ondřej Červenka. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.

Odkaz na tuto práci

Červenka, Ondřej. *Nadřazený systém pro správu garáže*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2017.

Abstrakt

V několika větách shrňte obsah a přínos této práce v češtině. Po přečtení abstraktu by se čtenář měl mít čtenář dost informací pro rozhodnutí, zda chce Vaši práci číst.

Klíčová slova Nahradte seznamem klíčových slov v češtině oddělených čárkou.

Abstract

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

Keywords Nahradte seznamem klíčových slov v angličtině oddělených čárkou.

Obsah

Úvod	1
1 Analýza	3
1.1 Struktura systému	3
1.2 Výběr komunikačního protokolu	3
1.3 Výběr platformy	5
2 Návrh	7
3 Implementace	9
4 Testování	11
Závěr	13
Literatura	15
A Seznam použitých zkratk	17
B Obsah přiloženého CD	19

Seznam obrázků

1.1	Konečná podoba zařízení	3
-----	-----------------------------------	---

Úvod

Cílem této práce je vytvořit nadřazený systém pro monitorování garážového komplexu. Výsledná aplikace bude komunikovat pomocí WiFi či Ethernetu s podřízenými systémy (zasílajícími údaje z čidel v garážích). Na základě získaných dat pak bude udržován stav jednotlivých garáží a vytvářena historie událostí.

Systém bude poskytovat webového rozhraní pro administraci. V tom bude možné přidávat a odebírat podřízené systémy, zobrazovat jejich stav a zaznamenané události.

Vzhledem k povaze zadání je nutné systém navrhnout s ohledem na zabezpečení přenášených informací před odposloucháváním či manipulací. Též je nutné autorizovat uživatele přistupující do webového rozhraní.

Dalším důležitým požadavkem je snadná rozšiřitelnost o nové funkce. Systém by mělo být možné v budoucnu doplnit o možnost správy rozdílných podřízených systémů (například subsystemy pro sledování skladových zásob) či integraci s mobilní aplikací. Bude tedy potřeba navrhnout vhodné API pro předávání informací mezi systémem a jeho klienty.

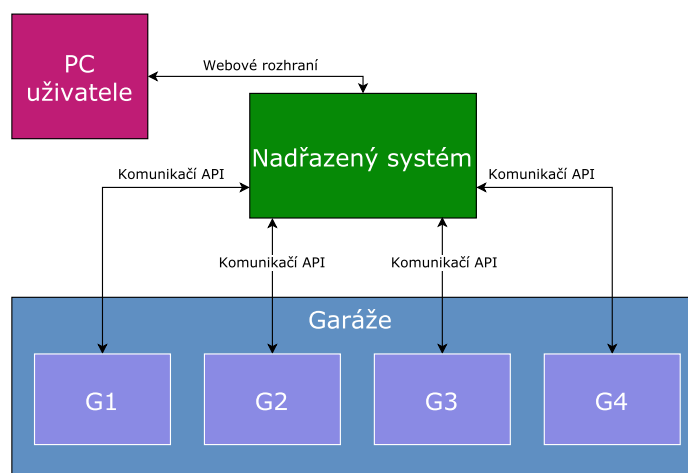
V této práci se chci zaměřit na tvorbu aplikace na jedné konkrétní hardwarové platformě, jako například *Raspberry Pi*. Aplikace spolu s touto platformou by pak měla tvořit kompletní zařízení, které bude možné po základní konfiguraci (připojení do WiFi sítě, nastavení hesla) hned nasadit.

Výsledné řešení by však mělo být dostatečně nezávislé na zvolené platformě. Tudíž by neměl být problém spustit systém například na osobním počítači či virtuálním serveru.

V analytické části práce tedy přiblížím proces výběru vhodné platformy, komunikačního protokolu a dalších prvků systému. Také stručně popíšu podřízený systém (garážové čidlo), se kterým budu dále pracovat. Další části mapují návrh zařízení na základě této analýzy, jeho implementaci a testování.

Analýza

1.1 Struktura systému



Obrázek 1.1: Základní struktura systému

1.1.1 Podřízený systém

1.1.2 Událost

1.2 Výběr komunikačního protokolu

Nadřazený systém bude se svými klienty (monitorovací zařízení v jednotlivých garážích) komunikovat přes WiFi nebo Ethernet. Základem komunikace bude TCP/IP protokol, je však potřeba zvolit vhodný protokol z aplikační vrstvy OSI modelu, který na něm bude stavět.

1.2.1 Vlastní protokol

Jedna z možností je implementovat vlastní protokol pomocí TCP/IP socketů. Toto řešení se mi však nezdá příliš vhodné, neboť nepřináší žádné významné výhody, naopak se s ním pojí řada komplikací.

Pro vlastní protokol by bylo nutné vytvořit robustní server, který zvládá obsluhu více klientů najednou. Dále by vzhledem k citlivosti přenášených dat bylo nutné implementovat nějakou formu šifrování. Tyto velmi obsáhlé problémy přitom řeší většina dnešních protokolů.

Další nevýhodou je nutnost implementace klientské části protokolu při vytváření nových zařízení spravovaných nadřazeným systémem. To do jisté míry omezuje jeho rozšiřitelnost.

1.2.2 HTTPS

Další možnost je využít ke komunikaci protokol HTTPS. V tomto případě by klienti komunikovali se systémem pomocí HTTP metod jako například `get` nebo `post`.

Jelikož součástí požadavků na systém je i webové uživatelské rozhraní, bude v každém případě nutné použít webový server pro jeho provoz. Ten by pak bylo možné využít i k poskytnutí API pro komunikaci systému s garážovými čidly.

Vhodný webový server (jako například *Nginx*) zajistí vícevláknovou obsluhu všech klientů. HTTPS se také postará o kryptografické zabezpečení přenášených dat, je však nutné získat certifikát k ověření pravosti serveru.

Lze použít například certifikáty nadace Let's Encrypt, které jsou poskytovány zdarma. Kromě toho dodává Let's Encrypt také automatizačního klienta *Certbot* [1] pro snadné nasazení a aktualizaci jejich certifikátů.

Certifikát pro provoz HTTPS bude potřeba zajistit i v případě, že komunikace s klienty nebude postavena na tomto protokolu. Je totiž nutné také zabezpečit webové rozhraní, například kvůli ověření identity uživatele. Nutnost pořízení certifikátu tedy nepředstavuje nevýhodu oproti jiným protokolům.

API realizované pomocí HTTPS je poměrně snadno rozšiřitelné. Pro nově implementovanou operaci stačí definovat URL a případně formát přenášených dat.

Další výhodou je snadná implementace na straně klienta, tedy garážového čidla. Knihovny umožňující vytvářet HTTP požadavky jsou dostupné na většině populárních platforem jako například *Arduino* (s Ethernet shieldem, oficiální knihovna *EthernetClient* [2]) nebo *ESP8266* (knihovna *esp8266wifi* [3]).

1.2.3 MQTT

[4]

1.3 Výběr platformy

Pro realizaci systému je nutné zvolit vhodnou platformu. Jelikož je cílem práce vytvořit fyzické zařízení, rozhodl jsem se jako základ použít některý z jednoduchých počítačů, které jsou v dnešní době na trhu. Tyto počítače bývají cenově velmi dostupné a zároveň poskytují dostatečný výkon a podporu pro provoz systému.

Při výběru počítače byla nejdůležitějším kritériem podpora softwaru potřebného k implementaci monitorovacího systému.

Návrh

Implementace

```
# ... code here ...

import numpy as np

def foo(a):
    print(a)

class FooBar:
    def __init__(self):
        self.b = 10

a = [1, 2, 3, 4]
for i in a:
    if i == 2:
        print("hello world")
```

Ukázka 1: Testovací listing

Testování

Závěr

Literatura

- [1] Electronic Frontier Foundation: Certbot – About. [cit. 2017-10-18]. Dostupné z: <https://certbot.eff.org/about/>
- [2] Arduino: Web Client. 2015, [cit. 2017-10-25]. Dostupné z: <https://www.arduino.cc/en/Tutorial/WebClient>
- [3] Grokhotkov, I.: esp8266wifi – Client Example. 2017, [cit. 2017-10-25]. Dostupné z: <https://github.com/esp8266/Arduino/blob/master/doc/esp8266wifi/client-examples.rst>
- [4] Lampkin, V.: What is MQTT and how does it work with WebSphere MQ? 2012, [cit. 2017-10-25]. Dostupné z: https://www.ibm.com/developerworks/mydeveloperworks/blogs/aimsupport/entry/what_is_mqtt_and_how_does_it_work_with_websphere_mq?lang=en

Seznam použitých zkratek

API

HTTP Graphical user interface

HTTPS Graphical user interface

MQTT Graphical user interface

OSI

TCP/IP Graphical user interface

URL

Obsah přiloženého CD

	readme.txt.....	stručný popis obsahu CD
	exe	adresář se spustitelnou formou implementace
	src	
	impl.....	zdrojové kódy implementace
	thesis	zdrojová forma práce ve formátu L ^A T _E X
	text	text práce
	thesis.pdf	text práce ve formátu PDF
	thesis.ps	text práce ve formátu PS