



ZADÁNÍ BAKALÁ SKÉ PRÁCE

Název:	Informa ní systém pro správu klient poskytovatele WiFi p ipojení
Student:	Lukáš Korel
Vedoucí:	Ing. Mgr. Pavla Vozárová, Ph.D., M.A.
Studijní program:	Informatika
Studijní obor:	Informa ní systémy a management
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2017/18

Pokyny pro vypracování

1. Navrh te informa ní systém pro management p ipojení a plateb klient poskytovatele sí ového p ipojení na platform MikroTik, který bude obsahovat uživatelské rozhraní, ízení p ipojení klient, databázi uživatel, sledování plateb zákazník, analýzu stability a vytížení jednotlivých spoj a ízení rychlostí p ipojení.
2. Navrh te podobu automatických report generovaných systémem, které budou podkladem pro rozší ování a obnovu sít a usnadní ekonomické vyhodnocení investic.
3. P ipravte projektovou dokumentaci a studii proveditelnosti pro tento systém.
4. Odhadn te náklady na implementaci a provoz navrženého nástroje a vyhodno te limity takového ešení.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
d kan

V Praze dne 31. ledna 2017

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Bakalářská práce

Informační systém pro správu klientů poskytovatele WiFi připojení

Lukáš Korel

Vedoucí práce: Ing. Mgr. Pavla Vozárová, Ph.D., M.A.

11. května 2017

Poděkování

Děkuji paní Ing. Mgr. Pavle Vozárové, Ph.D., M.A. za cenné rady a odborné vedení práce. Děkuji také mé rodině a přátelům za podporu po celou dobu studia.

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 mé 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.

V Praze dne 11. května 2017

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2017 Lukáš Korel. 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, je nezbytný souhlas autora.

Odkaz na tuto práci

Korel, Lukáš. *Informační systém pro správu klientů poskytovatele WiFi připojení*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2017.

Abstrakt

Tato bakalářská práce se zabývá návrhem informačního systému pro správu klientů poskytovatele WiFi připojení. Cílem práce je analýza běžných požadavků správců a techniků menších sítí na takový systém, návrh systému s ověřením proveditelnosti a ekonomické vyhodnocení systému. Cílem práce je také ověření možnosti získávání informací o stabilitě a propustnosti bezdrátových spojů. Navrhovaný systém je určen pro kombinaci s produkty od společnosti MikroTik, které jsou menšími poskytovateli internetového připojení často používány. V první části práce je provedena analýza současných řešení a vyhodnocení jejich limitů. Následuje část s analýzou požadavků na informační systém. Další částí je návrh systému. V něm je vytvořen datový model pro ukládání dat do relační databáze a návrh grafického uživatelského rozhraní aplikace pro přenosný osobní počítač techniků. V této části jsou také vytvořeny testy pro ověření proveditelnosti komunikace mezi osobním počítačem a síťovými zařízeními. Systém je navrhován pro implementaci v programovacím jazyce Java. V části ekonomické vyhodnocení navrhovaného systému je vypočítána návratnost investice v případě, že se poskytovatel připojení rozhodne nechat si systém vytvořit. V závěru jsou vyhodnoceny limity navrhovaného řešení a uvedeny možnosti dalšího rozšíření systému. V příloze lze nalézt vytvořené zdrojové kódy pro grafické uživatelské rozhraní. Nacházejí se v ní také zdrojové kódy a konfigurace zařízení, pomocí kterých byla ověřena proveditelnost systému. Hlavním přínosem této práce je ověření proveditelnosti systému a vytvoření návrhu, který je připraven pro budoucí implementaci.

Klíčová slova Informační systém, analýza a návrh, monitoring spojení, správa plateb, poskytovatel internetového připojení, zjednodušení správy, RouterOS, Java

Abstract

This bachelor thesis contains design of an information system for client management of a WiFi connection provider. The aim of this bachelor thesis is analysis of smaller networks administrators and technicians' common requirements for the system, design of the system and its feasibility verification and economic evaluation. The aim of the thesis is also to verify the possibility of obtaining information of stability and channel capacity of wireless connections. The system is focused on combination of MiktoTik company products which are frequently used by smaller internet providers. First chapter of the thesis contains analysis of the solutions which are currently available as well as the evaluation of their limitations. This chapter is followed by analysis chapter which includes information system requirements. Next chapter is connected to the design of the system where the data model for storing data into relation database is presented as well as the graphical user interface aimed for a technicians' portable computer. This chapter contains the verification testing of the communication between the personal computer and the network. The system is designed for implementation in the Java programming language. In the economic evaluation chapter the economic return of the investments is calculated, in case the connection provider will be interested in the system. The conclusion acquaints with the evaluated limitations of the designed solution and suggested possibilities of the system improvement. In the attachment the source code and the device configurations used for system feasibility verification are available. The main contribution of the thesis is feasibility analysis and creation the design which is ready to future implementation.

Keywords Information system, analysis and design, monitoring connection, payment management, internet service provider, simplification management, RouterOS, Java

Obsah

Úvod	1
1 Cíl práce	3
2 Kontext zkoumané problematiky	5
2.1 Důležité základní pojmy	5
2.2 MikroTik a jeho síťová řešení	6
3 Současná řešení	9
3.1 ISPadmin	9
3.2 LMS	9
3.3 LoginPortal	10
3.4 TheDude	10
4 Analýza	11
4.1 Požadavky	11
4.2 Možnosti řešení	12
4.3 Zvolené řešení	13
4.4 Business process model	13
4.5 Doménový model	16
4.6 Případy užití	19
5 Návrh	23
5.1 Datový model	23
5.2 Grafické uživatelské rozhraní	25
5.3 Skripty pro síťová zařízení	32
6 Ekonomické vyhodnocení	37
6.1 Náklady na vývoj	37
6.2 Náklady na provoz a údržbu	38

6.3	Návratnost	39
6.4	Vedlejší benefity	40
6.5	Limity řešení a možnosti rozšíření	41
6.6	Shrnutí	41
	Závěr	43
	Literatura	45
	A Seznam použitých zkratk	49
	B Obsah příloženého média	51

Seznam obrázků

4.1	Proces získání nového klienta	14
4.2	Proces při problému s dostupností služeb u klienta	15
4.3	Doménový model	16
4.4	Případy užití	19
5.1	Datový model	23
5.2	Přihlašovací obrazovka	27
5.3	Přihlašovací obrazovka se špatně zadanými přihlašovacími údaji	27
5.4	Obrazovka se seznamem klientů	28
5.5	Obrazovka se seznamem síťových zařízení	29
5.6	Obrazovka pro vytvoření, nebo úpravu záznamu o platbě	30
5.7	Obrazovka pro měření odezvy spoje	30
5.8	Vzhled grafu při měření odezvy s periodou 1000 ms	31
6.1	Náklady na vývoj vypočítané v ProjectLibre	38
6.2	NPV různých variant	40

Seznam tabulek

5.1	Atributy entity Adresa	24
5.2	Atributy entity Klient	24
5.3	Atributy entity Zarizeni	25
5.4	Atributy entity Platba	26
6.1	Náklady na zaměstnance	37
6.2	Očekávaná úspora času	39

Úvod

V dnešní době je v řadě případů třeba rychle vyhodnocovat dostupnost a kvalitu služeb, aby klienti byli s poskytovatelem spokojeni a neodcházeli ke konkurenci. S rozvojem informačních technologií rostou nároky na kvalitu služeb. Aby podniky mohly rychle zpracovávat potřebná data, nasazují informační systémy. Takové systémy mohou sloužit také pro správu plateb za tyto služby. Platby je třeba kontrolovat a na jejich základě řídit dostupnost poskytovaných služeb klientům. Informační systémy také pomáhají k měření dostupnosti a kvality služeb. Díky rozvoji informačních technologií je možné ušetřit finanční náklady na zpracování získaných informací.

Téma práce jsem zvolil z toho důvodu, že existují malí lokální poskytovatelé síťového připojení, kteří nemají ani základní systém pro sledování plateb a centrální přehled o stavu sítě, a proto jim uniká část příjmů. Jedná se především o poskytovatele, kteří využívají pro komunikaci bezdrátovou technologii WiFi v příměstském prostředí a okolních vesnicích. Mnoho takovýchto sítí vzniklo jako náhlý projekt před deseti a více lety bez řízení a centrálního systému. Síť se postupem času rozrostly až na stovky klientů v síti a v této situaci přestává být únosné ruční sledování stavu sítě a plateb bez alespoň základního systému. Existuje i případ systému, ve kterém správce vlastní síťovou infrastrukturu, jeho kolegové se starají o klienty, ale již nemají dostatek znalostí pro konfiguraci zařízení, kterou pak řeší hlavní administrátor. Systém by tak měl právě pomoci v detekci místa problému, aby byl problém snadněji a rychleji odstraněn.

V práci také aplikuji mé zkušenosti a znalosti, které jsem získal od roku 2012 až do současnosti v prostředí menších poskytovatelů bezdrátového internetového připojení. Účastnil jsem se zejména rozšiřování sítí, posilování propustnosti spojů a řešení problémů, které se často vyskytují při provozu bezdrátových sítí.

Tato práce se zabývá proveditelností systému pro řízení přístupu klientů do sítě poskytovatele internetového připojení a souvisejícími záležitostmi. Součástí práce je analýza a návrh informačního systému pro management při-

pojení a plateb klientů, pro analýzu stability a vytížení jednotlivých spojů. Jedním z výstupů je také návrh podoby automatických reportů generovaných systémem, které budou sloužit poskytovateli jako podklad pro rozšiřování a obnovu sítě a usnadní ekonomické vyhodnocení investic do infrastruktury sítě. V práci jsou odhadnuty náklady na implementaci a provoz navrženého nástroje a vyhodnocené limity navrhovaného řešení. Systém je určen pro správu v síti se zařízeními od společnosti MikroTik a operačním systémem RouterOS, který tato firma pro svá zařízení vyvíjí. U malých poskytovatelů bezdrátového internetového připojení je řešení od této společnosti velmi často používané. Systém je určen zejména malým lokálním poskytovatelům síťového připojení, kteří budují bezdrátové sítě WiFi v menších městech a vesnicích.

Základem práce je tvorba analytické dokumentace pro informační systém. V této části je provedena podrobnější analýza potřeb menších správců sítě. Součástí analýzy jsou modely aktuálních podnikových procesů u poskytovatelů internetového připojení. Součástí dokumentace jsou popisy častých podnikových procesů při managementu sítě, vytvořen doménový model, model funkčních i nefunkčních požadavků na aplikaci a model případů užití.

Z analytické dokumentace vychází návrh systému. Základní částí je návrh databázového modelu pro evidenci zařízení, uživatelů, jejich plateb a adres, kde se klienti a zařízení nacházejí. Databáze obsahuje veškeré důležité informace, které poskytovatelé často potřebují evidovat. Dále je navržena podoba grafického uživatelského rozhraní, ke kterému patří návrh podoby automatických reportů generovaných systémem, které poslouží k efektivnímu rozšiřování a obnově sítě a usnadní ekonomické vyhodnocení případných investic. Součástí je podrobnější popis skriptů, které budou v aplikaci využity k získávání dat ze zařízení s operačním systémem RouterOS. Skripty budou zaslány do zařízení, kde se následně spustí a výsledek bude systémem automaticky zpracován. Komunikace a získávání výsledků je ověřena pomocí testování na skutečných zařízeních s RouterOS. Výsledkem je návrh, který prokazuje proveditelnost takového systému.

V ekonomickém vyhodnocení je vytvořen hrubý odhad nákladů na implementaci a provoz aplikace. Jelikož se jedná o aplikaci pro menší poskytovatele, je aplikace navržena pro implementaci pomocí technologií, které lze zdarma využít i pro komerční účely. Díky tomu údržba a provoz nebude větší finanční zátěží pro provozovatele systému. Zde jsou také vyhodnoceny limity navrhovaného řešení a popsány možnosti budoucího rozšíření systému.

Cíl práce

Cílem bakalářské práce je analýza a návrh informačního systému pro menší poskytovatele internetového připojení pomocí bezdrátové technologie WiFi. Cílem práce je také vyhodnocení proveditelnosti takového systému. Systém je určen pro správu informací o zákaznících, jejich platbách a síťových zařízeních tak, aby při jeho nasazení nebylo nutné provádět zásadní změny v jednotlivých zařízeních a systém mohl být jednoduše uveden do provozu. Systém jsem navrhl tak, aby bylo možné jej přizpůsobit a rozšířit podle detailních požadavků správce sítě. Mým záměrem je navrhnout aplikaci pro osobní počítač, která splňuje základní požadavky správců menších bezdrátových WiFi sítí, které jsem stanovil na základě získaných zkušeností u menších poskytovatelů. Mezi tyto požadavky patří základní správa zařízení v síti, jako je kontrola stability a vytížení bezdrátových spojů a řízení jejich rychlosti připojení. Dalším požadavkem je správa informací o klientech poskytovatele, se kterým souvisí sledování plateb a řízení rychlosti jejich připojení. Aplikaci by měl být schopen použít i technik v terénu, který nemá dostatek znalostí pro získávání informací ze systému pomocí příkazového řádku.

Kontext zkoumané problematiky

V této kapitole jsou popsány důležité pojmy, které se vyskytují v práci. Kapitola je rozdělena na pojmy základní a pojmy z oblasti síťových řešení od společnosti MikroTik. V práci se zabývám návrhem systému pro použití v kombinaci se síťovými prvky právě od společnosti MikroTik, jelikož řešení od této společnosti je často nasazované u malých poskytovatelů internetového připojení.

2.1 Důležité základní pojmy

2.1.1 Informační systém

Informační systém (IS) má mnoho významů a definic. Jedna z mnoha definic říká, že „*Informační systém definujeme jako uspořádání vztahů mezi lidmi, datovými a informačními zdroji a procedurami jejich zpracování za účelem dosažení stanovených cílů.*“ [1, str. 14].

V dnešní době je velmi obtížné rychle zpracovávat větší množství dat bez informačního systému. K efektivnímu řízení firmy je třeba data zpracovat systémem. Výsledkem zpracování by měly být pouze informace, které jsou pro uživatele systému užitečné. IS může sloužit pro správu vnitřních dat, pro efektivní řízení firmy, může pomoci při rozhodování vedení firmy, může sloužit pro řízení vztahů se zákazníky nebo dodavateli a pomáhat při mnoha dalších firemních procesech.

2.1.2 ISP a WISP

Jedna z mnoha definic pojmu ISP stanovuje, že „*ISP (Internet Service Provider) je organizace poskytující přístup do Internetu (buď přímo nebo zprostředkovává službu jiného – nadřazeného – ISP)*“ [2, str. 2].

Pojem WISP přidává informaci, že infrastruktura sítě ISP využívá převážně bezdrátové technologie spojů, nejčastěji bezdrátovou technologii WiFi.

2.2 MikroTik a jeho síťová řešení

Podle [3], je MikroTik lotyšská společnost, která vyvíjí síťová zařízení a software pro ISP. Pro síťová zařízení vyvíjí systém RouterOS umožňující jejich řízení. Svá vlastní síťová zařízení, která jsou schopna pracovat samostatně, označuje jako RouterBoard.

2.2.1 RouterOS

RouterOS je operační systém pro síťová zařízení založená na mnoha architekturách počítačových čipů. Kromě mnoha jiných architektur je podporovaná architektura x86. Ta umožňuje sestavit síťové zařízení z běžných počítačových komponent i v domácích podmínkách. Takové zařízení může být v budoucnu rozšiřováno o další komponenty, jako jsou síťové karty, moduly pro bezdrátové spoje a různá další zařízení kompatibilní se systémem. [4]

K používání RouterOS je třeba, aby na zařízení, kde je instalován systém, byla platná licence, jinak bude systém funkční pouze 24 hodin. Pokud je zakoupeno zařízení RouterBoard, licence je již jeho součástí. Pro provoz na sestaveném síťovém zařízení či pro upgrade na vyšší verzi licence je třeba licenci zakoupit zvlášť. Licence se váže na paměťové médium, takže je možné ji z jednoho zařízení přenést na jiné, pokud médium není fyzicky spojené se zbytkem zařízení. [5]

RouterOS je založen na Linuxovém jádře verze 3.3.5. Systém v základní konfiguraci poskytuje mnoho funkcí. Například je možné přiřadit IP adresy k ethernetovým portům, tvořit virtuální LAN, NAT (překlad síťových adres), nastavovat pravidla ve firewallu, směrování a mnoho dalších funkcí. [6]

Základní systém obsahuje jen část balíčků a funkcí, které jsou součástí každé instalace. Systém je možné rozšířit o mnoho dalších funkcí pomocí rozšiřujících balíčků, které vytváří společnost MikroTik společně s RouterOS. Systém je možné rozšířit o různé nástroje pro správu zařízení, zabezpečovací prvky, podporu protokolů NTP, DHCP, IPv6 a mnoho dalších. [7]

Ke vzdálenému ovládní zařízení je možné využít terminál nebo nástroj s GUI zvaný Winbox. Použití Winboxu je nejjednodušší možností, neboť nevyžaduje pokročilejší znalosti práce v terminálu. Nástroj je vytvořen pro operační systém Windows, přesto jej lze spustit na Linuxových systémech pomocí programu Wine. Pomocí aplikace Winbox je možné nakonfigurovat a ovládat téměř všechny funkce RouterOS, až na několik výjimek, které jsou dostupné pouze pomocí terminálu. [8]

2.2.2 RouterBoard

Společnost MikroTik vyrábí síťové prvky pro vnitřní i venkovní použití. Většina prvků je modulárních, aby si ISP mohli sestavit síťový uzel podle potřeb. Rozšiřující prvky nemusejí být pouze od společnosti MikroTik. Často jedinou

podmínkou při stavbě síťového uzlu je, aby bylo možné na vybraném zařízení provozovat systém RouterOS. Mezi jednotlivé prvky, ze kterých se uzly skládají patří síťové směrovače, přepínače, WiFi vysílače, antény, integrované směrovače označované jako RouterBoard a mnoho dalších rozšiřujících prvků. Zařízení RouterBoard je třeba těmito rozšiřujícími prvky často doplnit pro přizpůsobení požadavkům síťové infrastruktury. [9]

Jednotlivé RouterBoardy se od sebe liší zejména výkonem integrovaného procesoru, propustností ethernetových portů typu RJ45, počtem slotů pro rozšiřující bezdrátové nebo optické moduly, velikostí operační paměti a dalšími vlastnostmi. Tato zařízení se využívají zejména v menších síťových uzlech. K nim se doplňují zejména bezdrátové rádiové karty pro rozšíření funkčnosti. Pro bezdrátové karty mají některé RouterBoardy rozšiřující sloty typu Mini PCI, nebo Mini PCI Express. Rozšiřující prvky se rozlišují frekvencí, podporou protokolů, výkonem a typem konektoru. K bezdrátovým kartám je třeba ještě doplnit antény pro vysílání a přijímání signálu.

Jako klientské bezdrátové zařízení se využívá integrované bezdrátové zařízení, které je složeno z antény, vysílacího modulu a směrovače v jednom obalu.

Pro větší síťové uzly se zátěž rozkládá do více RouterBoardů vzájemně propojených metalickým kabelem, druhou možností je sestavení výkonného zařízení na architektuře x86 s použitím rozšiřujících desek s konektory pro bezdrátové moduly. V některých případech se také používá jeden hlavní výkonný směrovač s mnoha konektory typu RJ45, ke kterým se připojí jen venkovní zařízení, která se nestarají o směrování, ale pouze o přijímání a vysílání bezdrátového signálu, a zbytek zařídí hlavní směrovač.

Současná řešení

V současné době existuje několik řešení pro management sítí a klientů ISP. Velká většina řešení je tvořena jako komerční nebo jako systém, který nepodporuje RouterOS. Z osobní zkušenosti vím, že pro malé ISP by bylo pořízení komerčního systému finančně nevýhodné. Z mého průzkumu také vyplývá, že menší ISP má jiné požadavky na systém, kterým některá řešení nevyhovují. Nejzajímavější systémy pro ISP jsou v této kapitole podrobněji popsány.

3.1 ISPadmin

Jedním z největších systémů pro ISP je ISPadmin [10]. Tento IS představuje komplexní řešení. ISPadmin je komerční systém a není snadné jej nasadit, což většinou provádí placená technická podpora. Systém přináší finanční úspory až u větších ISP, kdy dokáže nahradit část kancelářské pracovní síly, což vyplývá z [11]. Pracovní tým menších ISP se skládá jen z techniků, kteří veškerou práci zastanou sami bez kancelářské pracovní síly. Z důvodu uzavřenosti systému jej není možné libovolně upravit a správci sítě jsou závislí na oficiálních aktualizacích systému.

3.2 LMS

LMS je polský systém pro správu síťových zařízení a zejména správu klientů ISP. Z [12] vyplývá, že systém je vyvíjen pod licencí GNU GPL v2. Systém je vyvíjen v jazyce PHP, Perl a C. Z dokumentace systému [13] vyplývá, že systém nepodporuje zařízení se systémem RouterOS. Systém je spíše určen pro nasazení na server s OS Linux.

3.3 LoginPortal

System LoginPortal je komerční IS. Není sice tak rozsáhlý jako ISPadmin, ale nabízí také velké množství možností [14, 15]. System podporuje zařízení MikroTik, Ubiquity a fineBSD. Z ceníku [16] lze zjistit, že roční náklady na system pro menší ISP, kde se počet klientů pohybuje v řádu stovek vychází na 57 600 Kč.

3.4 TheDude

System The Dude byl vytvořen společností MikroTik. Slouží pro řízení všech zařízení se systémem RouterOS v síti poskytovatele z jednoho místa v síti. Má mnoho funkcí pro správu všech zařízení v síti, aby technik nemusel některé úkony provádět na každém zařízení zvlášť. V dokumentaci [17] jsou popsány detailně možnosti tohoto systému. Patří mezi ně aktualizace systému v zařízeních, správa zařízení pomocí připravených nástrojů a mnoho dalších. The Dude není určeno pro evidenci klientů a jejich plateb. Je vhodné jej použít jako další system, který technici pro svou práci využívají.

Analýza

4.1 Požadavky

Kapitola obsahuje popis požadavků na navrhovaný systém. Požadavky jsem vytvořil z vlastních zkušeností v prostředí menších ISP, kde tyto funkce při správě sítě často chybí. Požadavky jsou rozděleny na funkční a nefunkční.

4.1.1 Funkční požadavky

- F1 Evidence klientů** Systém bude schopen vytvářet, upravovat a mazat záznamy o klientech. Bude možné vyhledat klienta podle zadaných filtrů, zobrazit jeho platby a adresu jeho pobytu, kde klient přijímá síťové služby poskytovatele.
- F2 Evidence plateb klientů** Systém bude schopen vytvářet, upravovat a mazat záznamy o přijatých platbách od klientů. Bude možné vyhledat platbu podle zadaných filtrů a vyhledat záznam o klientovi z vybrané platby. Systém bude také podporovat import plateb ze souboru CSV ve specifikovaném formátu.
- F3 Evidence zařízení** Systém bude vytvářet, upravovat a mazat záznamy o zařízeních v síti. Bude možné vyhledat adresu, kde se zařízení nachází, zobrazit další zařízení připojená přímo k němu a umožní připojit se k zařízení pomocí aplikace Winbox.
- F4 Informace o spojích** Systém bude schopen provádět základní analýzu spojů v síti. Bude schopen analyzovat výpadky komunikace ve spojích a sledovat vytížení jednotlivých spojů.
- F5 Řízení rychlostí připojení** Systém bude schopen nastavovat omezení propustnosti spoje.
- F6 Grafické uživatelské rozhraní** Systém bude poskytovat GUI, pro ovládání systému myší a klávesnicí. Bude lokalizované do českého jazyka.

4.1.2 Nefunkční požadavky

N1 Systém přizpůsobitelný potřebám Systém bude možné upravit a rozšířit o další funkce.

N2 Provoz na osobním počítači Aplikaci bude možné provozovat na operačním systému Windows verze 7 Service Pack 1, 8, 8.1 a 10.

N3 Přenosná aplikace bez nutnosti instalace Aplikaci bude možné přenášet bez nutnosti instalace na přenosném datovém zařízení (např. USB flash disk) a spouštět ji z tohoto zařízení.

N4 Zabezpečený přístup do aplikace Přístup do aplikace bude zabezpečen pomocí uživatelského jména a hesla, aby nemohla neoprávněná osoba cokoli v systému měnit.

N5 Zabezpečená komunikace Při komunikaci po veřejném kanále budou posílaná data chráněna proti odposlechu šifrováním.

N6 Použitelná bez internetového připojení Aplikaci bude možné spustit a použít funkce, které nevyžadují připojení do sítě i při výpadku internetového připojení.

4.2 Možnosti řešení

Možností řešení požadovaného systému je několik. Pro nejjednodušší evidenci malého strukturovaného množství informací může být použita tabulková aplikace z libovolného balíku kancelářských aplikací. Ze zástupců komerčních uzavřených aplikací například Microsoft Excel z kancelářského balíku Microsoft Office. Ze zástupců aplikací s licencí volného užití je možné využít například aplikaci Calc z kancelářského balíku Libre Office. Při použití tohoto řešení není možné propojit evidované informace s informacemi ze síťových zařízení. Zpočátku může být takové řešení dostačující, ale nehodí se pro velké množství dat. Při větším množství klientů a zařízení v síti není řešení časově zvládnutelné a časem se může stát, že bude dokument s velkým množstvím dat velmi nepřehledný. Při řešení, ve kterém jsou odděleny záznamy klientů a zařízení, je pak nutné využívat další aplikaci (v síti se zařízeními se systémem RouterOS aplikaci WinBox) pro management síťových zařízení a ručně synchronizovat data mezi tabulkami a zařízeními. Takové řešení není pružné vzhledem k možnému rozšiřování sítě.

Pokročilejším řešením může být centrální systém, který může být provozován na centrálním serveru v síti. Tam se správci sítě mohou přihlásit a řídit jednotlivé prvky z jednoho místa. Nevzniká pak problém způsobený různě aktuálními daty. Nevýhodou ale je, že pokud dojde k výpadku spoje, na kterém

je správce, do systému se nedostane. V praxi jsou evidované informace o zařízeních nejčastěji potřebné v nestandardních situacích, kdy nějaká část sítě vykazuje problémy.

Další možností je vytvořit aplikaci, která umožňuje provoz na osobní stanici správce sítě a může být i přenositelná mezi zařízeními. Správce ji může mít vždy s sebou u klientů, kterým aktuálně nejde spojení. Nevýhodou tohoto řešení je možná nekonzistence dat, která mohou být uložena na různých stanicích v různých verzích. Řešením může být jedna centrální stanice v síti s průběžně aktualizovanými daty a mnoho pracovních stanic správců sítě, kteří si při zjištění změny stáhnou vždy aktuální data.

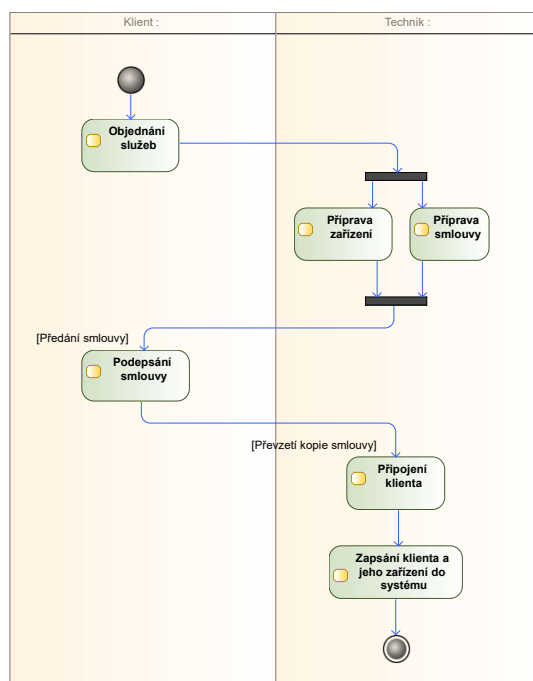
Pokročilejší systém by kombinoval jak výhody centrálního systému, tak lokální aplikace. Lokální by tvořila rozhraní pro testování spojů, stability a dalších vlastností sítě na žádost správce z osobního počítače. Nepřetržitou analýzu s hlášeními a nepřetržitými záznamy by zajišťoval systém na centrální stanici v síti. Centrální systém pak může vycházet z vlastností lokální aplikace, rozšířit ji o webový portál s informacemi pro klienty s využitím rozšířené databáze z lokální aplikace. Takové řešení ale vyžaduje mít centrální stanici v síti s webovým serverem a je již potřeba mít pokročilejší znalosti jak systém nasadit.

4.3 Zvolené řešení

Zvolil jsem následující řešení — navrhnout informační systém pro pracovní stanice techniků a správců sítě, bez centrálního systému, o který může být v budoucnu řešení rozšířeno. Hlavním důvodem výběru je jednoduchost nasazení aplikace v již existující síti. Technik také bude moci mít aplikaci vždy sebou v terénu na přenosném počítači. Navrhované řešení nebude vyžadovat složitější konfiguraci síťových zařízení před jejím nasazením. Bude využito vestavěné databáze pro uchování informací v aplikaci. Výsledkem by měl být návrh, který bude v budoucnu rozšiřitelný, snadno zhotovitelný, jednoduše použitelný a ušetří správcům sítě čas.

4.4 Business process model

V této kapitole jsou popsány nejdůležitější procesy, se kterými jsem se setkal při běžném provozu sítě. Jedná se o uzavření smlouvy s novým klientem, kontrolu plateb od klientů a o proces při řešení problému s dostupností služby u klienta. Diagramy jsem vytvořil v aplikaci Modelio. Celý projekt s modely je na příloženém datovém médiu.



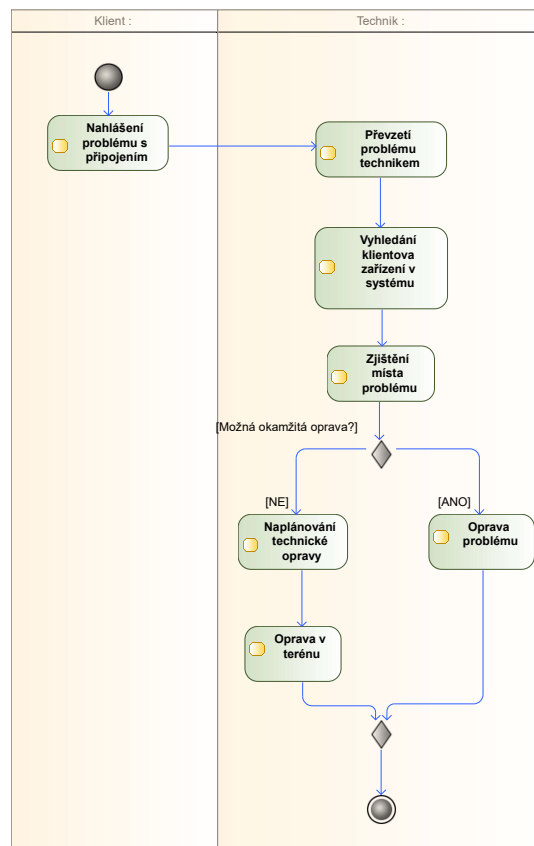
Obrázek 4.1: Proces získání nového klienta

4.4.1 Uzavření smlouvy s klientem

Uzavření smlouvy s klientem je jeden z procesů, které nastávají při běžném provozu sítě. Tento proces znázorňuje obrázek 4.1. Proces nejčastěji začíná objednáním služby osobním či telefonním rozhovorem. Technik či administrátor připraví smlouvu a zařízení podle dohody, u klienta je dále provedeno podepsání smlouvy, instalace zařízení, která může být doplněna doplňkovou konfigurací podle přání klienta. Na konci je proveden záznam do systému o novém zákazníkovi a jeho zařízení.

4.4.2 Kontrola plateb

Z mých zkušeností vyplývá, že někteří malí poskytovatelé příchozí platby kontrolují jen náhodně, nebo při odvodu daní ze zisku. Někdy si ani nemusejí všimnout, že klient některé služby nezaplatil. Pokud důkladně kontrolují veškeré příchozí platby, tak to dělají technici či administrátoři sítě na úkor jiné práce. Jelikož malé firmy se snaží vyjít co nejvíce vstříc přání klienta, je jim umožněno libovolně pozastavit dodávky služeb, klienti platí v různě dlouhých periodách (např. měsíčně, čtvrtletně, ročně), a z toho vzniká nepřehlednost v platbách. Z uvedených důvodů unikají některým poskytovatelům zisky. Proto je třeba, aby systém evidoval veškeré příchozí platby, umožňoval zobrazení plateb za různá období a mohl filtrovat platby podle klientů. V praxi jsem také zjistil,



Obrázek 4.2: Proces při problému s dostupností služeb u klienta

že některým přichozím platbám chybějí identifikační údaje jako je variabilní symbol a poté je nutné dohledávat původ platby podle čísla účtu či dalších atributů.

4.4.3 Řešení nedostupnosti služby

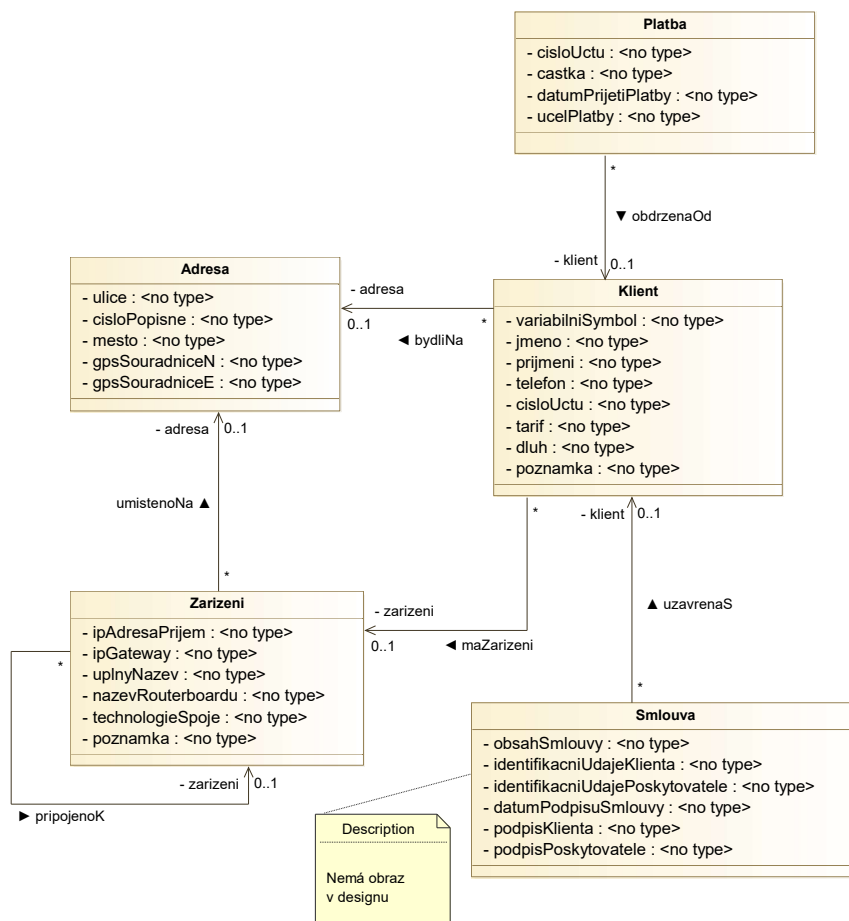
V některých situacích může nastat problém s dostupností poskytovaných služeb. Proces je znázorněn na obrázku 4.2.

Nejčastěji problém nahlásí klient telefonickým hovorem či formou SMS. Technik převezme problém. V této situaci může postupovat každý technik jinak, ale obecně každý potřebuje dohledat klientovo zařízení v síti, aby mohl provádět testy v dostupnosti zařízení, propustnosti spojů k tomuto zařízení a měření stability spoje. Pokud technik nepoužívá žádný systém na správu klientů a zařízení, řeší tento problém pomocí příkazového řádku, či grafickým klientem Winbox. Postupně prochází síťovou strukturou až ke spoji, který vykazuje nějaké problémy. Technici, kteří mají nějaký systém s evidencí všech zařízení a klientů, tak mohou tento proces urychlit. Nejlépe jsou na tom ti,

kteří mají částečně automatizovaný systém, umožňující měření stability, propustnosti spojů a detekci problémů. Po nalezení místa problému je třeba nalézt možnosti jeho řešení. Některé závady lze vyřešit ihned z prostředí pro správu zařízení, některé vyžadují fyzický zásah v terénu. Složitější zásahy se musejí naplánovat tak, aby nebyl způsoben výpadek služeb dalším uživatelům.

4.5 Doménový model

Kapitola popisuje entity, které souvisejí s analyzovanou doménou. Diagram doménového modelu je na obrázku 4.3. Jednotlivé entity jsou detailně popsány, aby bylo zřejmé, jaké všechny objekty a informace je nutné v systému uchovávat. V této části není přiřazen k atributům datový typ. Některé entity nejsou přeneseny do designu, jelikož nejsou potřeba uchovávat v systému, jsou jen zde znázorněny pro úplnost analyzované domény.



Obrázek 4.3: Doménový model

4.5.1 Zařízení

Třída představuje adresovatelné síťové zařízení (v tomto případě RouterBoard), které je zapojené do infrastruktury sítě.

ipAdresaPrijem IP adresa zařízení, pomocí které je možné se přihlásit do konkrétního zařízení. Tato adresa je na ethernetovém portu, přes který probíhá komunikace se zařízením blíže ke kořeni ve stromové topologii sítě.

ipGateway IP adresa přímého předchůdce tohoto zařízení směrem ke kořeni ve stromové topologii.

uplnyNazev Nezkrácený název zařízení v síti.

nazevRouterboardu Kódové označení typu zařízení udávané výrobcem zařízení.

technologieSpoje Název technologie, pomocí které je zařízení připojeno ke svému přímému předchůdci směrem ke kořeni v topologii sítě.

poznamka Doplňující informace k zařízení.

4.5.2 Klient

Osoba využívající služby poskytovatele.

variabilniSymbol Identifikační symbol osoby, pod kterým jsou prováděny platby. Klient jej získá při podpisu smlouvy.

jmeno Křestní jméno klienta.

prijmeni Příjmení klienta.

telefon Telefonní číslo klienta.

cisloUctu Číslo účtu, ze kterého bude klient platit za služby. Eviduje se pro případ, kdy klient při placení udělá chybu a lze pomocí tohoto údaje snadněji dohledat původ platby.

tarif Částka, kterou klient aktuálně měsíčně platí za služby.

dluh Finanční částka, kterou by měl klient zaplatit. Obsahuje součet všech dosud nezaplacených částek.

poznamka Obsahuje dodatečné poznámky ke klientovi, jako jsou informace o přerušení poskytovaných služeb a další potřebné informace.

4.5.3 Adresa

Adresa nemovitosti, ve které se zařízení nachází, či ve které klient bydlí.

ulice Název ulice, ve které se nemovitost nachází.

cisloPopisné Číslo popisné nemovitosti.

mesto Název města či vesnice, ve kterém se nemovitost nachází.

gpsSouradniceN Geografická souřadnice rovnoběžky severní polokoule.

gpsSouradniceE Geografická souřadnice poledníku východní polokoule.

4.5.4 Platba

Platba od klienta, která je účtována podle aktuálního tarifu klienta.

cisloUctu Číslo účtu, ze kterého byla platba přijata.

castka Velikost částky, která byla přijata.

datumPrijetiPlatby Datum, ve kterém byla platba přijata.

ucelPlatby Informuje o tom, k jaké službě se platba vztahuje.

4.5.5 Smlouva

Entita smlouvy není již dále zobrazována v designu, ale je zde popsána, jelikož je součástí procesu získání nového klienta. Vyskytuje se pouze v papírové podobě a je uchovávána jak klientem, tak poskytovatelem služeb.

obsahSmlouvy Obsahuje informace a podmínky poskytování služeb.

identifikacniUdajeKlienta Údaje sloužící k jednoznačné identifikaci klienta.

identifikacniUdajePoskytovatele Údaje o dodavateli služeb.

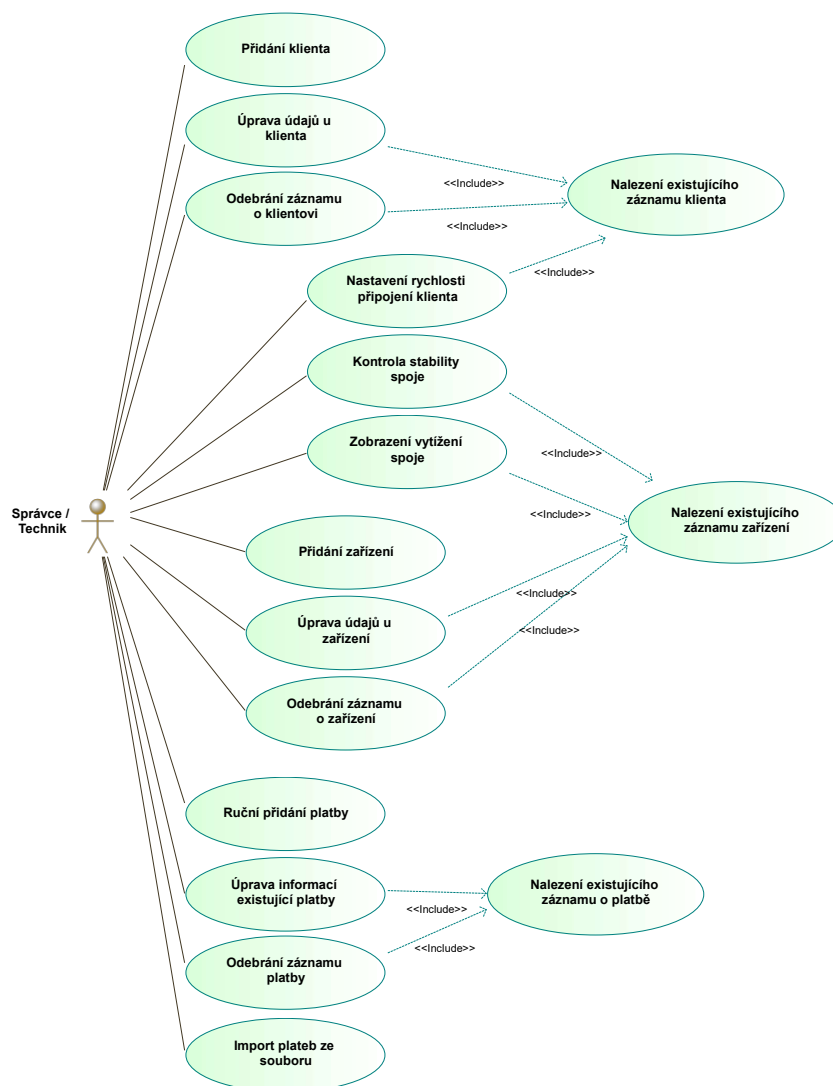
datumPodpisuSmlouvy Datum, kdy byla smlouva klientem podepsána.

podpisKlienta Ruční podpis osobou klienta.

podpisPoskytovatele Ruční podpis poskytovatele služeb či jeho oprávněného zástupce.

4.6 Případy užití

Kapitola popisuje případy užití (use cases) IS pro navrhovaný systém. Uvedené funkcionality bude navrhovaný systém uživatelům poskytovat. Jelikož je systém pouze pro techniky a správce, kteří potřebují přístup ke všem uvedeným funkcionalitám, nachází se na diagramu 4.4 pouze jeden typ uživatele systému. Jednotlivé funkcionality jsou dále detailněji rozepsány. Podle těchto funkcionalit budou při implementaci vytvořeny testy k ověření jejich splnění.



Obrázek 4.4: Případy užití

4. ANALÝZA

- UC1 Přidání klienta** Přidá nového klienta do systému. Bude využíváno při procesu uzavření smlouvy s klientem.
- UC2 Úprava údajů u klienta** Umožňuje upravit údaje o klientovi. Tento případ obsahuje část nalezení existujícího záznamu klienta, který má být upravován.
- UC3 Odebrání záznamu o klientovi** Umožňuje uživateli odebrat existující záznam o klientovi. Tento případ obsahuje část nalezení existujícího záznamu klienta, který má být odebrán.
- UC4 Nastavení rychlosti připojení klienta** Umožňuje nastavit rychlost připojení klienta do sítě. Tento případ obsahuje část nalezení existujícího záznamu klienta, kterému má být rychlost nastavena.
- UC5 Kontrola stability spoje** Umožňuje spustit kontrolu stability spoje. Zobrazí odezvu na spoji a případné výpadky v komunikaci. Tento případ obsahuje část nalezení existujícího záznamu zařízení, ze kterého bude kontrola prováděna.
- UC6 Zobrazení vytížení spoje** Umožňuje spustit kontrolu vytížení spoje. Zobrazí datový tok na vybraném spoji. Tento případ obsahuje část nalezení existujícího záznamu zařízení, ze kterého bude vytížení zobrazováno.
- UC7 Přidání zařízení** Přidá nové zařízení do systému. Bude využíváno při procesu, kdy je do infrastruktury sítě nainstalováno nové síťové zařízení.
- UC8 Úprava údajů o zařízení** Umožňuje upravit záznam o zařízení. Případ obsahuje část nalezení existujícího záznamu zařízení, který má být upravován. Nastává při úpravě nastavení zařízení, výměně jeho části nebo celého zařízení za nové.
- UC9 Odebrání záznamu o zařízení** Umožňuje odebrat existující záznam o zařízení. Případ obsahuje část nalezení existujícího záznamu zařízení, který má být odebrán. Využije se v případě fyzického odstranění zařízení ze síťové infrastruktury, například při ukončení poskytování služeb klientovi, kdy není využíváno dalšími klienty.
- UC10 Ruční přidání platby** Umožňuje ručně přidávat provedenou platbu. Použije se v případě, kdy není platba provedena převodem na účet či u plateb, které není možné importovat z důvodu chybějících platebních údajů.
- UC11 Úprava informací existující platby** Umožňuje upravit informace o platbě. Případ obsahuje část nalezení existujícího záznamu o platbě, která má být upravena. Nastává při možném chybném záznamu o platbě.

UC12 Odebrání záznamu platby Umožňuje uživateli odebrat existující záznam o platbě. Případ obsahuje část nalezení existujícího záznamu o platbě, která má být odebrána. Využije se při chybném vložení platby.

UC13 Import plateb ze souboru Umožňuje import nových plateb z CSV souboru, který bude splňovat vyžadovaný formát.

Nalezení existujícího záznamu klienta Umožňuje vyhledat existující záznam klienta. Buďto ručně vyhledat v seznamu všech klientů nebo po zadání filtrů z jejich podmnožiny. Využito v některých případech užití.

Nalezení existujícího záznamu zařízení Umožňuje vyhledat existující záznam zařízení. Buďto ručně vyhledat v seznamu všech zařízení nebo po zadání filtrů z jejich podmnožiny. Využito v některých případech užití.

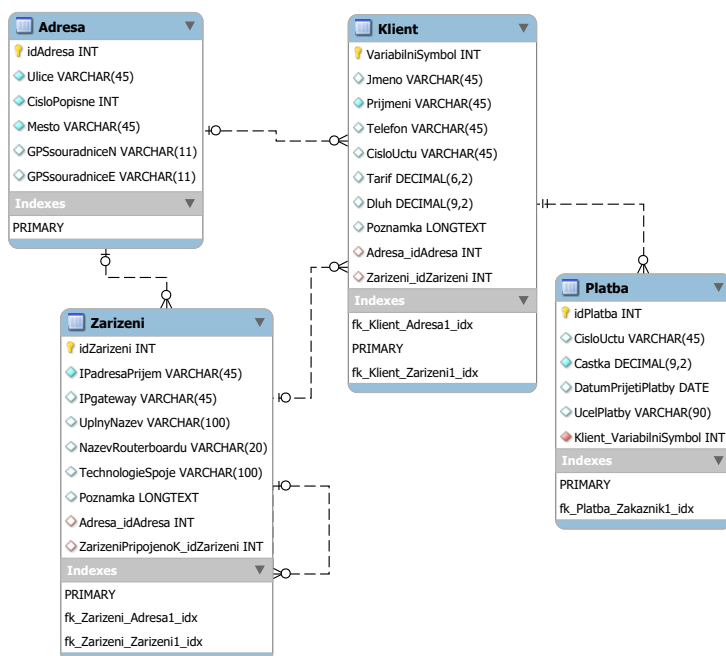
Nalezení existujícího záznamu o platbě Umožňuje vyhledat existující záznam o platbě. Buďto ručně vyhledat v seznamu všech plateb nebo po zadání filtrů z jejich podmnožiny. Využito v některých případech užití.

Návrh

V této kapitole popisují navržený datový model pro ukládání záznamů do relační databáze, návrh vzhledu GUI aplikace a popis skriptů, které budou použity v aplikaci pro získávání dat ze síťových zařízení.

5.1 Datový model

Datový model relační databáze je vyobrazen na diagramu 5.1, níže popisují jednotlivé entity, jejich atributy a omezení.



Obrázek 5.1: Datový model

5.1.1 Tabulka „Adresa“

Entita představuje adresu, která určuje místo, kde klient přijímá služby nebo místo, kde se nachází síťové zařízení. Její atributy jsou popsány v tabulce 5.1.

Tabulka 5.1: Atributy entity Adresa

Atribut	Datový typ	Poznámka
idAdresa	INT	Primární klíč adresy Autoincrement
Ulice	VARCHAR(45)	Ulice nemovitosti Povinný atribut
CisloPopisne	INT	Číslo popisné nemovitosti Povinný atribut
Mesto	VARCHAR(45)	Město, nebo vesnice nemovitosti Povinný atribut
GPSsouradniceN	VARCHAR(11)	GPS souřadnice - sever
GPSsouradniceE	VARCHAR(11)	GPS souřadnice - východ

5.1.2 Tabulka „Klient“

Entita představuje klienta, který využívá služby a má uzavřenou smlouvu s poskytovatelem. Její atributy jsou popsány v tabulce 5.2.

Tabulka 5.2: Atributy entity Klient

Atribut	Datový typ	Poznámka
VariabilniSymbol	INT	Primární klíč klienta
Jmeno	VARCHAR(45)	Křestní jméno klienta
Prijmeni	VARCHAR(45)	Příjmení klienta Povinný atribut
Telefon	VARCHAR(45)	Telefonní číslo klienta
CisloUctu	VARCHAR(45)	Číslo účtu klienta
Tarif	DECIMAL(6,2)	Paušální měsíční částka
Dluh	DECIMAL(9,2)	Nesplacený dluh
Poznamka	LONGTEXT	Poznámka ke klientovi
Adresa_idAdresa	INT	Cizí klíč z entity Adresa Označuje adresu, kde klient přijímá služby
Zarizeni_idZarizeni	INT	Cizí klíč z entity Zarizeni Označuje zařízení, které je konečné, před zařízeními klienta

5.1.3 Tabulka „Zarizeni“

Entita představuje síťové zařízení zapojené do infrastruktury sítě, ke kterému je možné se vzdáleně připojit. Její atributy jsou popsány v tabulce 5.3.

Tabulka 5.3: Atributy entity Zarizeni

Atribut	Datový typ	Poznámka
idZarizeni	INT	Primární klíč zařízení Autoincrement
IPadresaPrijem	VARCHAR(45)	IP adresa zařízení Je možné se pomocí adresy do zařízení připojit IP adresa směrem ke kořeni sítě Povinný atribut
IPgateway	VARCHAR(45)	IP adresa výchozí brány
UplnyNazev	VARCHAR(90)	Úplný název zařízení
NazevRouterboardu	VARCHAR(20)	Kódové označení udávané vý- robceem zařízení
Technologie spoje	VARCHAR(100)	Označení technologie spoje ke svému přímému předchůdci směrem ke kořeni sítě
Poznamka	LONGTEXT	Poznámka k zařízení
Adresa_idAdresa	INT	Cizí klíč z entity Adresa Adresa, kde se zařízení nachází
ZarizeniPripojenoK_ idZarizeni	INT	Cizí klíč z entity Zarizeni Zařízení, které je přímým před- chůdcem akruálního směrem ke kořeni sítě

5.1.4 Tabulka „Platba“

Entita představuje platbu za poskytované služby. Její atributy jsou popsány v tabulce 5.4.

5.2 Grafické uživatelské rozhraní

Z důvodu možné potřeby přenositelnosti aplikace mezi různými platformami jsem navrhl grafické uživatelské rozhraní (GUI) pro programovací jazyk Java. Aplikace vyvinuté v tomto jazyce je možné mezi platformami přenášet bez změn v kódu. Tento jazyk také umožňuje tvořit GUI pomocí frameworků.

Dříve byl velmi oblíbený framework Swing [18]. Umožňuje jednoduchou tvorbu formulářových aplikací. Obsahuje sadu připravených komponent pro

Tabulka 5.4: Atributy entity Platba

Atribut	Datový typ	Poznámka
idPlatba	INT	Primární klíč platby Autoincrement
CisloUctu	VARCHAR(45)	Číslo účtu příchozí platby
Castka	DECIMAL(9,2)	Velikost přijaté částky Povinný atribut
DatumPrijetiPlatby	DATE	Datum, kdy byla částka přijata Pokud je atribut prázdný, klient částku dluží
UcelPlatby	VARHCAR(90)	K jakému účelu se platba vztahuje
Klient_ Variabilni-Symbol	INT	Cizí klíč z entity Klient Určuje, ke kterému klientovi se platba vztahuje Povinný atribut

většinu situací. Je možné také tvořit vlastní komponenty nebo upravovat existující. Swing je postaven nad AWT (Abstract Window Toolkit), což je ještě starší framework, který obaluje rozhraní samotného operačního systému. Swing oproti AWT poskytuje bohatší a komfortnější komponenty a navíc podporuje MVC (Model-View-Controller) architekturu.

V současné době existuje spolu se Swingem ještě jeden modernější framework a to JavaFX. Obsahuje lepší oddělení logiky a výstupu, podporu rychlejšího vykreslování, animací, multimédií a dalších nových technologií. V praxi se u formulářových aplikací používají oba frameworky, u novějších aplikací JavaFX, existující aplikace z většiny používají Swing. Oficiálně bylo společností Oracle potvrzeno, že JavaFX Swing nahrazuje. V praxi ještě stále existuje mnoho aplikací napsaných ve Swingu, ale nové aplikace prakticky nemá smysl v něm vytvářet a doporučuje se jen JavaFX.

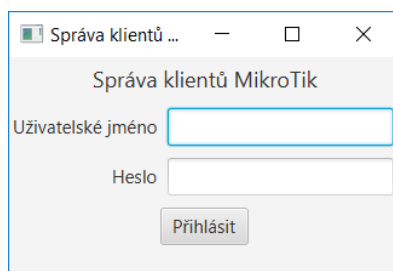
Podle [19] je JavaFX moderní framework pro tvorbu bohatých okenních aplikací. JavaFX přináší podporu obrázků, videa, hudby, grafů, CSS stylů a dalších technologií, které zajistí, že výsledná aplikace má moderní vzhled. Zároveň je kladen důraz na jednoduchost tvorby. Všechny uvedené prvky jsou v JavěFX v základu. JavaFX se hodí jak pro desktopové aplikace, tak pro webové applety nebo mobilní aplikace.

V JavaFX je možné vyvíjet dvěma styly. První je podobný Swingu. Tvoří se instance formulářových prvků, které se vkládají do kontejnerů na formulářové komponenty. Druhý způsob je pomocí jazyka pro návrh formulářů FXML. Tímto způsobem jsem vytvořil návrh vzhledu GUI v této práci. FXML je jazyk odvozený z jazyka XML. JavaFX zde přenáší principy z HTML a CSS do desktopových aplikací. Soubory ve formátu FXML pro GUI je možné tvo-

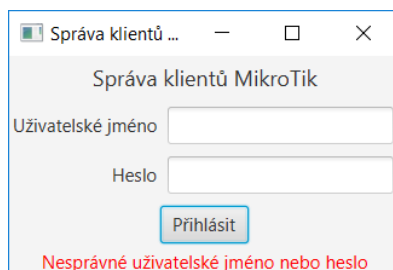
řit pomocí grafického designeru JavaFX Scene Builder. Díky této aplikaci se eliminuje riziko lidské chyby v FXML kódu a vývoj je snadnější. Veškeré vytvořené obrázky vzhledu aplikace a zdrojové kódy GUI jsou na přiloženém médiu v adresáři „GUI“ a „src/GUI“.

Aplikace bude rozdělena na dvě části. První část bude umožňovat přístup do aplikace, která bude provozuschopná i v režimu bez internetového připojení. Bude umožňovat manipulaci s lokálními záznamy v lokální databázi. Druhá část, která bude vyžadovat připojení alespoň k těm síťovým prvkům, na kterých bude měření či nastavení propustnosti spoje s klientským zařízením probíhat.

5.2.1 Základní aplikace



Obrázek 5.2: Přihlašovací obrazovka

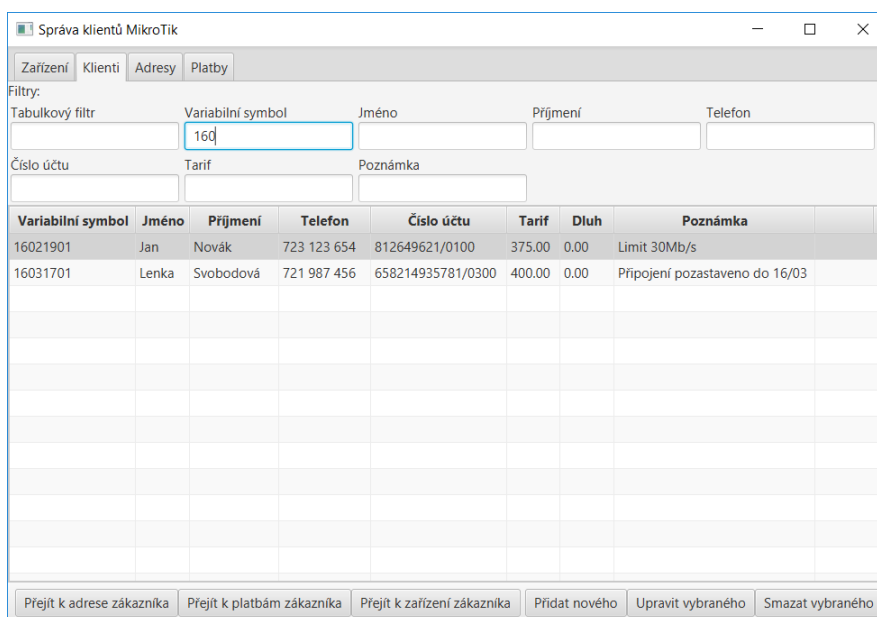


Obrázek 5.3: Přihlašovací obrazovka se špatně zadanými přihlašovacími údaji

Jedním z požadavků na navrhovaný IS je zabezpečený přístup do aplikace. Po spuštění aplikace se nejdříve zobrazí okno s požadavkem na zadání přihlašovacích údajů ve formě uživatelského jména a hesla. Výzva na zadání ověřovacích údajů je znázorněna na obrázku 5.2. Pokud uživatel zadá nesprávné přihlašovací údaje, zobrazí se upozornění, které je na obrázku 5.3. Pokud uživatel zadá správné přihlašovací údaje, umožní se uživateli přístup k datům, se kterými bude moci manipulovat.

Základní část aplikace má čtyři záložky. Záložka „Zařízení“ pro práci se síťovými zařízeními, „Klienti“ pro práci s klienty, „Adresy“ se záznamy adres

5. NÁVRH



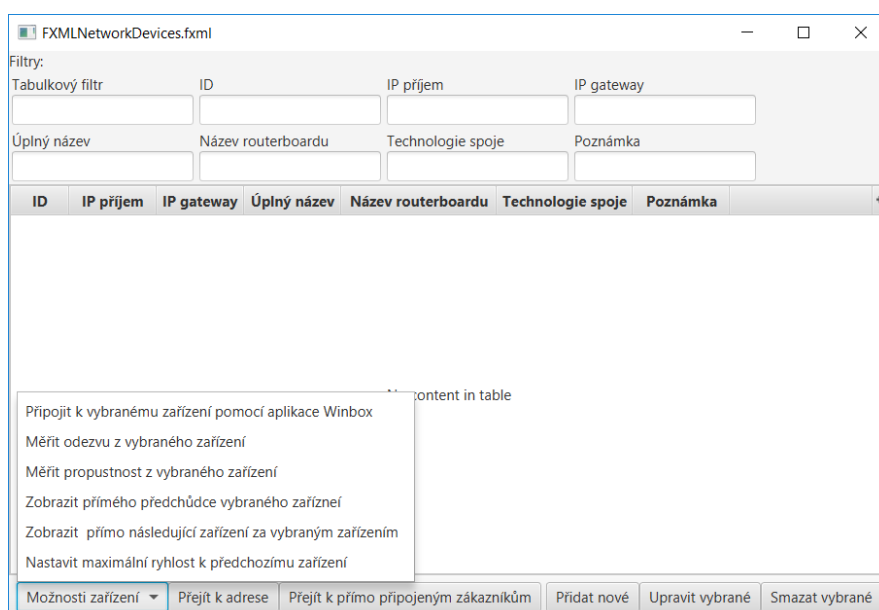
Obrázek 5.4: Obrazovka se seznamem klientů

pro zařízení i klienty a „Platby“ pro evidenci plateb klientů. Na obrázku 5.4 je zobrazená záložka s klienty. Všechny další záložky mají podobný vzhled.

V každé záložce bude možné filtrovat zobrazený obsah. Pole tabulkový filtr je určeno pro filtry nad libovolným sloupcem, budou tedy zobrazeny řádky, kde alespoň jeden sloupec obsahuje ve své hodnotě zadaný text. Ostatní filtry slouží pro filtrování textu nad každým sloupcem zvlášť. Tato funkce je znázorněná také na obrázku 5.4, kde je zadaná hodnota v poli „Variabilní symbol“ a v tabulce jsou zobrazeny pouze záznamy s variabilním symbolem, který obsahuje zadaný řetězec. Filtry bude možné libovolně kombinovat, což bude fungovat jako logická funkce AND, aby v zobrazeném výsledku bylo méně záznamů a uživatel v nich mohl snadněji vyhledávat.

Pod tabulkou jsou tlačítka s možnostmi v aktuální záložce. V levé části tlačítka vyžadují vybrat jeden záznam z tabulky, pak lze přecházet k záznamům v různých tabulkách v závislosti na vybraném záznamu. V pravé části jsou tlačítka pro přidání záznamu do databáze, úprava vybraného záznamu a smazání vybraného záznamu. Podobné funkce jsou i v ostatních záložkách. V záložce „Platby“ je navíc tlačítko s možností import plateb ze souboru. Tato funkce umožní import plateb například z bankovního výpisu přijatých plateb.

Rozšířené možnosti oproti ostatním záložkám má záložka „Zařízení“, která je na obrázku 5.5. Ve spodní části je tlačítko možnosti zařízení, které obsahuje další možnosti, které lze na vybraném zařízení ze seznamu provádět. Možnost „Zobrazit přímého předchůdce vybraného zařízení“ zobrazí v tabulce pouze



Obrázek 5.5: Obrazovka se seznamem síťových zařízení

takové zařízení, ke kterému je vybrané zařízení přímo připojeno a vede přes něj cesta ven ze sítě. Další možností je „Zobrazit přímo následující zařízení za vybraným zařízením“, která zobrazí zařízení, která jsou přímo připojena k vybranému. Tato možnost usnadní přehled o zařízeních, která mohou být ovlivněna změnou konfigurace vybraného zařízení a může dojít k přerušení jejich komunikace. Ovlivněna mohou být i další zařízení, která se nacházejí za zobrazenými.

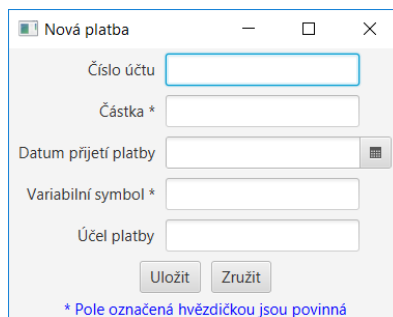
Ostatní možnosti v záložce „Zařízení“ jsou již závislá na vzdálenému přístupu k vybranému zařízení, se kterým bude tento systém komunikovat. Takové operace nelze provádět v režimu offline.

První možnost je „Připojit k vybranému zařízení pomocí aplikace Winbox“. Tato možnost pouze předá aplikaci Winbox parametry nutné pro připojení vybranému zařízení. Zbytek operací provádí aplikace Winbox sama a není již závislá na této aplikaci.

Možnosti pro měření a nastavení rychlosti budou využívat skripty popsané v další části tohoto dokumentu. Možnost „Nastavit maximální rychlost k předchozímu zařízení“ si vyžádá od uživatele aplikace jakou rychlost chce nastavit a pomocí skriptu se rychlost ve vybraném zařízení nastaví. Možnost „Měřit odezvu z vybraného zařízení“ a „Měřit propustnost z vybraného zařízení“ vytvoří nové okno poskytující rozhraní pro takové měření, které bude v další části detailněji popsáno.

Pro vytváření a úpravu záznamů bude sloužit formulářové okno. Jedno z nich je na obrázku 5.6, které slouží pro ruční přidání záznamu o platbě.

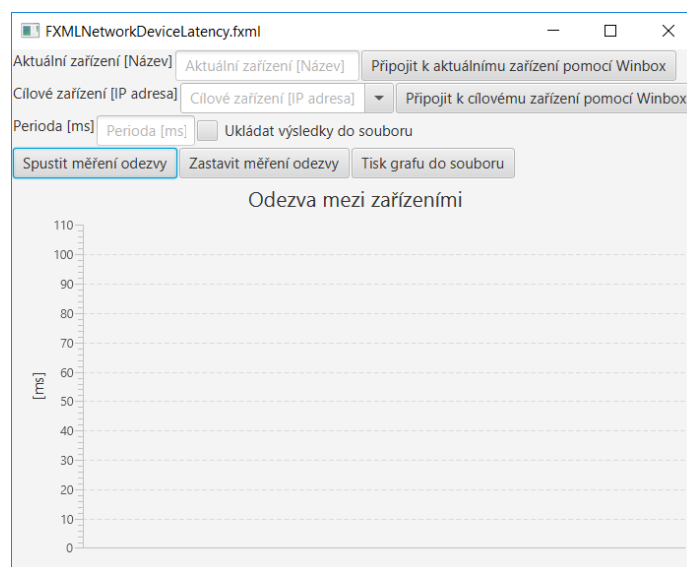
5. NÁVRH



Obrázek 5.6: Obrazovka pro vytvoření, nebo úpravu záznamu o platbě

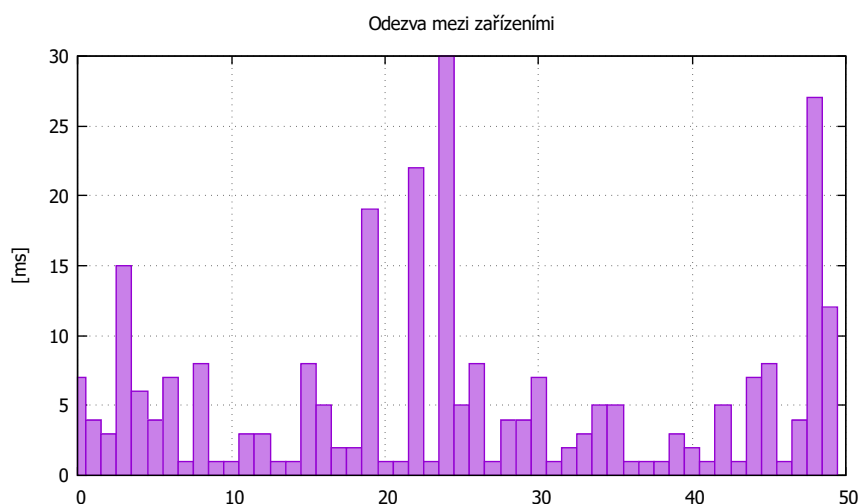
Zadané hodnoty v textových polích budou před uložením validovány pomocí regulárních výrazů v aplikaci. Pokud uživatel nevyplní nějaké povinné pole, přesáhne maximální délku řetězce nebo zadá jinak nesprávný vstup, který lze před uložením ověřit, bude na to upozorněn a takové pole bude označeno, že obsahuje nesprávnou hodnotu. V takovém případě nebude záznam v databázi nijak narušen. Dokud neprojde validace celého záznamu, nehrozí poškození záznamu, či uložení pouze části.

5.2.2 Analýza spojení



Obrázek 5.7: Obrazovka pro měření odezvy spoje

Při analýze spojení bude využito rozhraní pro měření odezvy spoje a měření propustnosti spoje. Pokud bude na vybraném zařízení ze seznamu zvolena



Obrázek 5.8: Vzhled grafu při měření odezvy s periodou 1000 ms

možnost „Měřit odezvu z vybraného zařízení“, zobrazí se okno pro zadání přihlašovacích údajů do zařízení a poté se zobrazí okno, které je na obrázku 5.7. V prvním textovém poli bude název vybraného zařízení, ze kterého bude měření probíhat. Do pole „Cílové zařízení“ může uživatel zadat libovolnou IP adresu zařízení, které zvolí jako cílové pro měření odezvy. V tomto poli bude možné vybrat ze zařízení, jejichž záznamy bude mít aplikace k dispozici z databáze. Pokud se na takovém zařízení bude nacházet operační systém RouterOS, bude možné se k němu připojit pomocí aplikace Winbox.

Pro měření odezvy bude muset uživatel zadat periodu v milisekundách, ve které bude test odezvy probíhat. Perioda také představuje hodnotu, kdy vyprší vyhrazený čas na odezvu a takový pokus bude vyhodnocen jako výpadek v komunikaci. Pokud bude zaškrtnuta položka pro ukládání výsledku do souboru, pak před spuštěním měření bude uživatel dotázán na zadání cílové cesty k souboru. Graf bude možné také vytisknout do souboru jako obrázek. Zobrazený graf se bude průběžně obnovovat, dokud nebude pomocí tlačítka „Zastavit měření propustnosti“ měření zastaveno nebo okno zavřeno. Výsledný graf může vypadat jako na obrázku 5.8, který jsem sestavil z měřených hodnot na uměle vytvořeném spoji v zarušeném prostředí při nastavení periody 1000 ms. Z grafu zkušený technik zjistí, že spoj nemá ideální podmínky pro provoz. Díky takovému grafu se pak technik dozví, který spoj je potřeba optimalizovat.

Pokud bude na vybraném zařízení ze seznamu zvolena možnost „Měřit propustnost z vybraného zařízení“, zobrazí se okno pro zadání přihlašovacích údajů do zařízení a poté se zobrazí okno podobné oknu pro měření odezvy. Jediným rozdílem je, že uživatel bude moci zvolit, zda chce měřit download,

nebo upload. Ostatní možnosti jsou stejné. Po kliknutí na tlačítko pro spuštění měření bude ještě dotázán na přihlašovací údaje do cílového zařízení. Je to z toho důvodu, že možnost měření propustnosti vyžaduje komunikaci s cílovým zařízením. Proto je možné takové měření provádět jen mezi zařízeními, na kterých je operační systém RouterOS nebo kde je spuštěna aplikace BTest, která je volně dostupná z webových stránek firmy MikroTik. Tato aplikace má jednoduché grafické rozhraní. Pokud je na cílové straně nastavena jako server, pak je k takové stanici možné měřit propustnost.

Vzhled grafu při měření propustnosti bude také podobný jako v případě měření odezvy. Jediný rozdíl je, že na svislé ose budou jednotky Mb/s. Analýza spoje pomocí propustnosti může pomoci technikovi odhalit některé problémy ve spoji. V některých případech se stává, že odezva spoje nevykazuje žádné problémy, ale při vytížení spoje na jeho maximální propustnost může docházet k takovým problémům, jako je výrazný pokles propustnosti, výrazně se zvýší odezva nebo ve výjimečných případech může docházet až k výpadkům spojení. Pro odhalení takového problému bude vhodné použít měření propustnosti.

5.3 Skripty pro síťová zařízení

Aby aplikace byla schopna zobrazit informace o zařízeních a spojích mezi nimi, je nutné zajistit její komunikaci s cílovým zařízením. Po připojení do zařízení může aplikace provádět předem připravené skripty pro RouterOS a výsledky zobrazovat uživateli.

Pro komunikaci se vzdáleným zařízením jsem se zabýval třemi možnostmi provedení. První možností byla ruční implementace komunikace se zařízením. Aby komunikace byla zabezpečená proti odposlechnutí, musel jsem využít knihovnu JSch [20], která poskytuje rozhraní pro vytvoření zabezpečeného komunikačního kanálu. Pro testování jsem se pokoušel vytvořit testovací třídu. Při prvních testech jsem zjistil nedostatek tohoto řešení. Výsledky ze zasláných skriptů byly vráceny v podobě, která je přizpůsobena v RouterOS pro pohodlné čtení v terminálu, ale je již problematické z výsledků vyfiltrovat užitečná data pro další zpracování v aplikaci.

Druhou možností bylo využít API v jazyce Java [21] vytvořené přímo pro komunikaci s RouterOS. Dostupné API je možné volně použít i pro komerční účely. Na stránkách s dokumentací tohoto API jsou zveřejněny příklady, jak se zařízením komunikovat. Poslední zveřejněná verze je ale z roku 2012 a není zaručena kompatibilita s novými verzemi RouterOS.

Třetí možností bylo využít nového API, které je stále aktualizované [22], a tím se bude následující část práce zabývat. Jeho poslední stabilní verze vyšla v říjnu 2016. Takové řešení jsem zvolil jako vhodné pro navrhovanou aplikaci. API je naprogramováno tak, aby výsledky ze skriptů vracelo v podobě, která je jednoduše programově zpracovatelná. Zvolené API je vydáváno pod licencí Apache 2.0 [23], která umožňuje komerční použití a neomezuje výslednou apli-

kaci. Vybrané API je možné získat jako zdrojové soubory, které mohou být zakomponovány do aplikace. Je možné také použít připravený Java Archive nebo nastavit závislost v aplikaci pomocí Apache Maven. API také umožňuje vytvořit zabezpečené spojení pro komunikaci se vzdáleným zařízením. Po navázání spojení a přihlášení do zařízení je možné zasílat příkazy pro RouterOS a výsledky zpracovat v aplikaci. Díky možnosti asynchronního spouštění příkazů ve vzdáleném zařízení je možné průběžně získávat výsledky z měření. Díky této vlastnosti může být graf s výsledky průběžně aktualizován. Autor API také zveřejnil příklady, ze kterých je možné vycházet a jednoduše v aplikaci použít.

Abych ověřil proveditelnost navrhovaného systému, sestavil jsem v domácích podmínkách bezdrátový spoj. Použil jsem k tomu zařízení RouterBoard RB433 a RB133. Konfiguraci těchto zařízení jsem vložil na přiložené datové médium. Abych mohl důkladně otestovat níže popisované příkazy, vytvořil jsem testovací program, ve kterém jsem implementoval metody pro testování. Zdrojový kód tohoto testovacího programu je také na přiloženém médiu.

V následujících částech popisuji příkazy, pomocí kterých bude aplikace získávat požadované výsledky ze zařízení. Tyto části mají mnoho společného. Jednotlivé části popíši na zde uvedeném příkladu, pomocí kterého jsem testoval jednotlivé příkazy pro cílová zařízení. Předtím, než je možné zaslat pomocí některé testovací metody příkaz do zařízení, je nutné mít navázané spojení se zařízením.

```
public String ukazkovaMetoda(String parametr, ...)
    throws MikrotikApiException {
    ResultListener rl = new ResultPrinter();
    return con.execute("prikazProZarizeni", rl);
}
```

Vytvořené metody pro testování mají parametry podle možností příkazu, který bude vykonán v zařízení. Metoda využívá navázané spojení, které se jmenuje *con*. Pro zaslání příkazu nebo celého skriptu je využito metody *execute* na navázaném spojení. Prvním parametrem jsou příkazy pro zařízení a druhým parametrem instance třídy, která implementuje rozhraní *ResultListener*. Díky této třídě je možné průběžně získávat průběžné výsledky ze zařízení. Metoda vrací řetězec, který identifikuje vykonávaný příkaz, který je možné pomocí něj kdykoli ukončit. Metody, které nejsou určeny k měření a nepotřebují možnost asynchronního běhu, nebudou *ResultListener* využívat a budou vykonávány sekvenčně. Pokud nelze příkaz vykonat, bude vyhozena výjimka *MikrotikApiException*. V následujících částech budou popsány pouze předávané parametry v *con.execute(...)*, které se u konkrétních metod mění.

5.3.1 Měření odezvy

Měření odezvy bude probíhat pomocí nástroje *Ping* [24], který je součástí RouterOS.

```
con.execute( "/ping address="+destinationIPAddress+"  
interval="+timeout+"ms", rl );
```

Nástroji *ping* budou předány dva parametry. První je *destinationIPAddress* typu *String*, který bude obsahovat IP adresu cílového zařízení. Druhý parametr *timeout* typu *int* nastavuje jak maximální dobu čekání na odezvu, tak délku periody, ve které bude měření opakováno. Druhý parametr bude v milisekundách.

5.3.2 Měření maximální propustnosti

Měření maximální propustnosti zajistí nástroj *Bandwidth Test* [25]. Testy je možné provádět nejen mezi zařízeními se systémem RouterOS, na kterém je spuštěn *BTest Server*. Pokud je na cílovém osobním počítači spuštěna aplikace *BTest* ve stavu *server*, pak je možné test spustit také mezi síťovým zařízením a osobním počítačem.

```
con.execute( "/tool/bandwidth-test protocol=tcp address=  
"+destiantionIPAddress+" tcp-connection-count="+  
tcpConnectionCount+" direction="+direction+" user="+  
username+" password="+password, rl );
```

Nástroji *bandwidth-test* bude v metodě předáno pět parametrů. Test bude probíhat na protokolu TCP, jelikož významná většina přenesených dat při běžném provozu využívá protokol TCP. Prvním předávaným parametrem je *destinationIPAddress*, který představuje IP adresu cílového zařízení. Druhým parametrem je *tcpConnectionCount*, který udává počet TCP spojení při testu. Třetí parametr *direction* bude obsahovat řetězec *receive* pro měření rychlosti stahování, nebo řetězec *transmit* pro měření odesílání z pohledu zařízení. Poslední dva parametry slouží pro předání přihlašovacích údajů do zařízení, na kterém je spuštěn *BTest* ve stavu *server*.

Měření maximální zátěže značně zatěžuje procesor, proto není vhodné nechávat test spuštěn příliš dlouho, aby nebyl příliš omezen provoz všech spojů, které komunikují přes zařízení, na kterém probíhá měření maximální propustnosti.

5.3.3 Měření aktuálního datového toku

Pro měření aktuálního datového toku na bezdrátovém rozhraní bude využito nástroje *monitor* [26]. Tento nástroj umožňuje sledovat dění na bezdrátovém rozhraní včetně datového toku.

```
con.execute("/interface/wireless/monitor_".id=" +
  wirelessInterfaceName, rl);
```

V metodě bude předán jeden parametr označen jako *wirelessInterfaceName*, ve kterém bude název bezdrátového rozhraní, na kterém má měření probíhat. Díky tomuto nástroji bude možné detekovat nejvytíženější články v síťové infrastruktuře a pomohou při rozhodování, kterou část by bylo vhodné posílit.

5.3.4 Řízení rychlosti připojení klientů

Pro nastavení maximální propustnosti spoje bude využito nástroje *Queue* [27]. Nástroj je vhodný pro řízení maximální propustnosti, omezení provozu podle dní v týdnu, hodin a podobně. V navrhované aplikaci bude využit k omezení rychlosti připojení u klientů podle tarifu, nebo při omezení služeb z důvodu nplacení závazků či z důvodu přerušování odběru služeb.

```
con.execute("/queue/simple/add_name=" + limitName + "_
  target=" + interfaceName + "_max-limit=" + upload + "
  k/" + download + "k");
```

Pro nastavení limitu bude požadováno několik parametrů. Jméno omezení *limitName* vyžaduje unikátní název v rámci jednoho zařízení. Pro přehlednost bude vhodné jako název omezení předávat variabilní symbol klienta, kterého se omezení týká, nebo jeho kombinaci s jinými atributy. Aby se nějakým omezením klienta nezpůsobilo nechtěné omezení nějakého jiného spoje, je třeba jako atribut *interfaceName* zvolit jméno rozhraní, na kterém je připojen pouze jeden konkrétní klient. Hodnoty atributů *upload* a *download* jsou nezáporná celá čísla. Pomocí těchto hodnot bude nastavena maximální rychlost na danou hodnotu v jednotkách kb/s. Směr toku je přiřazován z pohledu zařízení. Tedy pokud bude nastavena rychlost *download* na rozhraní směrem dál od kořene sítě, z pohledu klienta bude omezena rychlost *upload* a naopak.

Odstranění omezení bude na základě zadání názvu omezení, který je při vytváření omezení v parametru *limitName*. Pokud takový název v seznamu omezení v zařízení existuje, pak bude omezení odstraněno.

5.3.5 Skenování bezdrátového pásma

WISP často řeší problémy se zarušením bezdrátového frekvenčního pásma okolními vysílači. Při analyzování možností RouterOS jsem se pokoušel nalézt možnost skenování pásma pro snadnější hledání volné vysílací frekvence v bezdrátovém pásmu. Z dokumentace [26] plyne, že možnosti pro skenování pásma existují, ale žádná z nich není univerzálně použitelná. Problém spočívá v tom, že při klasickém skenování pásma dojde k přerušování navázaného spoje. Takové řešení není vhodné pro použití na spojích, na kterých je připojeno mnoho klientů. Navíc je nutné výsledky ukládat do souboru v zařízení, jehož obsah po

opětovném navázání spoje bude aplikací zpracován. Od verze RouterOS 6.35 existuje funkce zvaná *Background scan*, která umožňuje skenování frekvenčního pásma, aniž by bylo nutné přerušit spojení. Tato funkce by představovala vhodné řešení situace, ale podporuje pouze bezdrátový protokol 802.11. Z tohoto důvodu není možné do aktuálně navrhované aplikace vložit tuto funkci, jelikož mnoho spojů je provozováno na protokolu Nstreme, nebo NV2. Pokud v některé z budoucích verzí RouterOS bude možno provádět skenování pásma bez nutnosti přerušit spoje i na ostatních bezdrátových protokolech, může být aplikace o tuto funkci rozšířena.

Ekonomické vyhodnocení

V této kapitole provádím odhad nákladů na vývoj, provoz a údržbu aplikace. Zabývám se také tím, kolik času může ušetřit technik při hledání závady v síťové infrastruktuře, což může ušetřit i finanční náklady. Při výpočtu návratnosti investice jsem zohlednil fakt, že někteří klienti neplatí za poskytované služby, když platby nejsou kontrolovány. Tento problém je bez IS pro sledování a uchovávání přehledu plateb odhalen jen při náhodné ruční kontrole plateb, která je časově náročná. V kapitole se také zabývám limity tohoto řešení a možnostmi budoucího rozšíření systému.

6.1 Náklady na vývoj

Při odhadu nákladů na vývoj jsem použil program *ProjectLibre*, který slouží pro plánování projektů. Jedná se o podobnou aplikaci jako je *MS Project*. Náklady jsem rozdělil do několika částí. Jedná se pouze o hrubý odhad časových a finančních nákladů. Pro výpočet jsem použil průměrné mzdy na jednotlivých pozicích v České republice [28]. Průměrné hrubé měsíční mzdy jsem přepočítal na hodinové náklady na jednotlivé zaměstnance. Při výpočtu jsem hrubou mzdu zvedl o další náklady zaměstnavatele na zdravotní a sociální pojištění (9 % a 25 %). Počítám, že průměrný měsíc má 168 pracovních hodin. Výsledek jsem poté zaokrouhlil na desítky nahoru. Výsledné částky jsou v tabulce 6.1.

Tabulka 6.1: Náklady na zaměstnance

Pozice	Hodinový náklad
SW architekt	570 Kč
Analytik	340 Kč
GUI designér	250 Kč
Programátor Java	360 Kč
Tester	260 Kč
IT specialista	240 Kč

6. EKONOMICKÉ VYHODNOCENÍ

	⊞	Jméno	Trvání	Předchůdci	Jména zdrojů	Náklady
1		IS pro správu klientů poskytovatele WiFi připojení	47 dní			174720 Kč
2		Zahájení	1 den		IT specialista	1920 Kč
3		Analýza	5 dní			19040 Kč
4		Sběr funkčních a nefunkčních požadavků	3 dní	2	Analytik	8160 Kč
5		Případy užití	1 den	2	Analytik	2720 Kč
6		Analýza business procesů	1 den	2	Analytik	2720 Kč
7		Doménový model	2 dní	4;5;6	Analytik	5440 Kč
8		Design	7 dní			32720 Kč
9		Datový model	2 dní	3	SW architekt	9120 Kč
10		GUI	7 dní	3	GUI designér	14000 Kč
11		Skripty pro komunikaci s RouterOS	5 dní	3	IT specialista	9600 Kč
12		Implementace	27 dní			82800 Kč
13		Class diagram	3 dní	8	SW architekt	13680 Kč
14		Datová vrstva	3 dní	13	Programátor Java	8640 Kč
15		Business logická vrstva	14 dní	14	Programátor Java	40320 Kč
16		Prezentační vrstva	7 dní	15	Programátor Java	20160 Kč
17		Testování	5 dní			24800 Kč
18		Automatické testy	5 dní	8	Programátor Java	14400 Kč
19		Manuální testování	5 dní	8	Tester	10400 Kč
20		Dokumentace	5 dní			9600 Kč
21		Zdrojového kódu	2 dní	12	IT specialista	3840 Kč
22		Uživatelská	3 dní	21	IT specialista	5760 Kč
23		Nasazení do provozu	2 dní	3;8;12;17;20	IT specialista	3840 Kč

Obrázek 6.1: Náklady na vývoj vypočítané v ProjectLibre

Náklady na jednotlivé části jsou v tabulce 6.1. Tu jsem získal jako obrázek z programu *ProjectLibre*, ve kterém jsem provedl odhad nákladů na projekt. Z něj vyplývá, že celkové odhadované náklady jsou 175 000 Kč. Z hrubého odhadu vyhází, že časová náročnost projektu je 66 md (man-days). Celková doba projektu může tedy trvat až 66 dní, pokud systém bude vyvíjet jedna osoba a nebude možné využít možnosti souběžného průběhu některých částí. Jelikož je možné mnoho částí tvořit souběžně, pak se při dostatečném množství lidských zdrojů celková délka projektu zkrátí. Jelikož některé části jsou již v této práci vytvořené, pak náklady mohou klesnout až na 130 000 Kč.

6.2 Náklady na provoz a údržbu

Při návrhu systému jsem se snažil o co nejnižší provozní náklady. Je to také jedna z předností tohoto systému. Systém by měl být téměř bezúdržbový. Pouze pokud bude potřeba provádět nějaké úpravy z důvodu nekompatibility s nově vydanými verzemi RouterOS, bude potřeba provést údržbu systému. Takový náklad by ale neměl přesáhnout 5 000 Kč za rok, v průměru odhaduji 2 000 Kč za rok. Jelikož systém bude vytvořen pomocí nástrojů a aplikací, které umožňují bezplatné komerční použití, nevznikají žádné jednorázové ani paušální náklady při provozu. Aplikaci bude možné provozovat na běžném osobním počítači, který každý technik běžně vlastní, proto nevznikají ani náklady na pořízení nového zařízení.

6.3 návratnost

Menší poskytovatelé síťového připojení mají 100–500 klientů. Průměrnou cenu za připojení k internetu v ČR jsem převzal z [29], která v prosinci 2012 činila 430 Kč. Cena za připojení pomocí WiFi se pohybuje o několik desítek Kč níže, než je průměr, jelikož připojení má horší kvalitu než drátové. Proto budu počítat s částkou 400 Kč/měsíc za standardní služby. Z osobní zkušenosti vím, že počet neplatících uživatelů se pohybuje mezi 3–5 %.

K posouzení návratnosti jsem použil metodu NPV. Předpokládaná doba provozu aplikace bez zásadních úprav je pět let, proto jsem NPV počítal pro období pět let. Vyhodnocení návratnosti jsem provedl pro maximální výši investice 175 000 Kč. Jak ale uvádím v části s výpočty nákladů na vývoj, investice může klesnout až na 130 000 Kč díky již vytvořeným některým částem. Výnosy jsou složeny ze dvou položek. První část výnosu je tvořena platbami původně neplatících klientů.

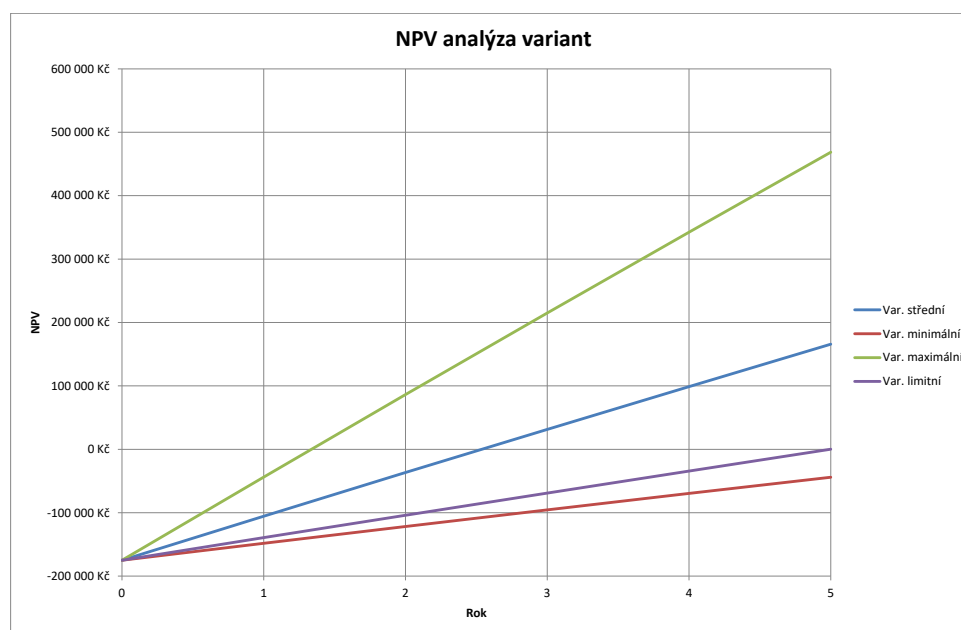
Druhou část výnosu tvoří úspora času technika, která vznikne zrychlením odstraňováním problémů ve spojích. Některé operace při hledání a řešení problému v infrastruktuře sítě mohou být díky aplikaci zkráceny. Tím ušetří čas při opravách technikům a klientům bude dodávka služeb rychleji obnovena. Podle mé zkušenosti technik v menší síti bez podpůrného IS stráví hodinu denně při hledání a řešení problémů s bezdrátovými spoji. Z toho přibližně třetinu času hledá místo problému a analyzuje jak se problém projevuje. Tato aplikace by měla pomoci zkrátit tuto dobu na polovinu. Toho dosáhne díky ušetření stráveného času při hledání klienta zařízení v infrastruktuře sítě a dalším podpůrným nástrojům pro analýzu spojů. Z tabulky 6.2 vyplývá, že roční úspora času technika je 3650 minut, což je přibližně 61 hodin. Sazba technika je stejná jako na IT specialistu, takže roční úspora je 14 600 Kč.

Tabulka 6.2: Očekávaná úspora času

Název atributu	Hodnota
Denní čas strávený s opravami problémů	60 minut
Současná časová náročnost detekce příčiny problému	20 minut
Očekávaná časová náročnost detekce příčiny problému	10 minut
Očekávaná roční úspora času	3650 minut
Hodinový náklad technika	240 Kč
Očekávaná roční úspora	14 600 Kč

V metodě NPV jsem použil diskontní sazbu 1 %. Jedná se o jedno z nejvyšších možných zhodnocení vkladu na bankovním spořicí účtu (Air Bank) [30]. Jiné způsoby zhodnocení vkladu (akcie, fondy, apod.) jsem neuvažoval. Vyhodnocení návratnosti investice jsem provedl v několika variantách. Střední varianta odpovídá průměrnému počtu klientů (300) a průměrnému počtu neplatících klientů (4 %). Minimální varianta odpovídá spodní hranici počtu kli-

6. EKONOMICKÉ VYHODNOCENÍ



Obrázek 6.2: NPV různých variant

entů (100) a nejnižšímu odhadu neplaticích klientů (3 %). Maximální varianta odpovídá horní hranici počtu klientů (500) a nejvyššímu odhadu neplaticích klientů (5 %). Doplnil jsem také variantu limitní, t.j. variantu s nejnižším počtem klientů, pro který vyjde kladná hodnota NPV v uvažovaném období pěti let.

Výpočty jsou provedené v příloze „NPV.xlsx“. Výsledky z výpočtů jsou na obrázku 6.2. Z výsledků plyne, že pro limitní variantu je potřeba 163 klientů při 3 % neplaticích klientů. Pro střední variantu se investice vrátí po 2,5 letech. Pro minimální variantu se investice nevrátí ani po pěti letech. V takovém případě se investice tohoto systému s hlediska NPV nevyplatí. Pro maximální variantu se investice vrátí přibližně za 1,5 roku.

6.4 Vedlejší benefity

Používání navrhovaného IS může přinést další benefity. Například při rychlejších opravách jsou zákazníci spokojenější se službami a sníží se pravděpodobnost odchodu klientů ke konkurenci. Ušetřený čas z efektivnější detekce problémů může technik využít pro analýzu kritických nebo často problémových spojů a předcházet tak problémům před jejich projevením na kvalitě spojení.

6.5 Limity řešení a možnosti rozšíření

Žádný IS není univerzální a každý má svá omezení. V tomto navrhovaném systému jsou omezení, která plynou už z principu zvolených technologií, jiná z limitujících možností RouterOS atd. Jelikož jsem zvolil řešení pro osobní počítač, je aplikace omezena tím, že není schopna zajistit neustálou kontrolu bezdrátových spojů. Návrh je tvořen tak, že aplikace bude jednouuživatelská a bude ji mít každý technik nebo správce sítě na osobním počítači či přenosném disku. Z toho vyplývá omezení, že každý technik bude mít v aplikaci vlastní data a synchronizace bude poměrně náročná.

Jak jsem již zmínil v části, kde jsem se zabýval příkazy pro RouterOS, aplikace nebude schopna provádět měření vysílacího pásma. Tato funkce je omezena v RouterOS a očekávám, že s příchodem podpory pro všechny vysílací protokoly v nějaké budoucí verzi doplním i tuto funkci do tohoto IS. Aplikace také nebude schopna sama vytvořit záznamy o zařízeních jen tím, že by se připojila do sítě a sama si vytvořila záznamy o existujících zařízeních v podsíti. Veškeré operace bude nutné provádět ručním přidáním záznamů. V případě importu plateb bude možné načíst data ze souboru, který může být výstupem internetového bankovníctví s příchozími platbami.

Aktuálně je projekt připraven k implementaci aplikace pro zařízení techniků. Do budoucna plánuji aplikaci rozšiřovat o další funkce, které více usnadní řízení provozu v síti. Fáze rozšiřování možností bude až v době po zhotovení tohoto základního systému a řádném otestování v provozu reálné sítě. Během používání systému mohou vznikat další konkrétní požadavky na rozšíření.

V delším časovém horizontu plánuji rozšířit systém o webovou aplikaci, která odstraní limit navrhovaného systému. Bude možné průběžně kontrolovat stabilitu, tvořit automatické reporty a při výpadku spoje bude technik ihned informován o místě, ve kterém došlo k problému. Taková aplikace pak může sloužit i pro klienty poskytovatele, kde mohou získat informace o službách a plánovaných údržbách v síti. Při kombinaci aplikace pro osobní počítač a webové aplikace bude možné zajistit průběžnou synchronizaci záznamů.

6.6 Shrnutí

Vytvořil jsem SWOT analýzu, která umožňuje rychle získat přehled o silných stránkách, slabých stránkách, příležitostech a hrozbách projektu. Díky ní lze jednoduše zjistit, co můžeme od tohoto projektu očekávat a jaké jsou jeho možnosti.

- Strengths (Silné stránky)
 - Nízké náklady na provoz
 - Ověřená funkčnost skriptů
 - Jednoduchá nasaditelnost

6. EKONOMICKÉ VYHODNOCENÍ

- Přenositelnost
- Evidence záznamů a ovládání zařízení v jedné aplikaci
- Weaknesses (Slabé stránky)
 - Úzce zaměřené řešení
 - Nezkušenost s vývojem aplikací
- Opportunities (Příležitosti)
 - Rozšiřitelnost o další funkce
 - Rozšiřitelnost o webovou aplikaci
- Threats (Hrozby)
 - Nedostatek financí na vývoj
 - Konkurenční produkty

Ze SWOT analýzy vyplývá, že silné stránky převažují nad slabými. Slabé stránky mohou být časem eliminovány. Jelikož je systém možné libovolně rozšiřovat, může časem z tohoto projektu vzniknout ucelený informační systém pro malé poskytovatele internetového připojení.

Závěr

Cílem práce bylo navrhnout IS, který zefektivní práci správcům a technikům menších ISP. Navrhovaný systém je zaměřen na zařízení s operačním systémem RouterOS. Cílem systému je také usnadnění sledování plateb, aby bylo možné jednoduše odhalit neplatící klienty.

Nejprve jsem provedl analýzu existujících řešení s podobným zaměřením. Zjistil jsem že, takové systémy jsou většinou komerční a nejsou vhodné pro malé ISP.

Při analýze jsem stanovil funkční a nefunkční požadavky na IS. Požadavky převážně vycházejí z mých zkušeností se správou síťových zařízení u menšího ISP. Poté jsem se v analýze zabýval volbou řešení, které splní požadavky na systém. Také jsem vytvořil modely nejčastějších procesů, které se v prostředí ISP vyskytují. Výsledkem analýzy je doménový model. Na konci analýzy jsem vytvořil diagram s případy užití požadovaného systému, které budou využity pro testování v době, kdy bude aplikace implementována.

V návrhu jsem vytvořil datový model pro vytvoření relační databáze v aplikaci. Také jsem vytvořil návrh vzhledu GUI aplikace v nástroji JavaFX Scene Builder, který bude sloužit při implementaci jako view v prezentační vrstvě aplikace. Pro komunikaci se síťovými zařízeními jsem se rozhodl pro využití API pro jazyk Java, které zprostředkovává komunikaci a výsledky vrací v jednoduše zpracovatelném formátu. Pro ověření proveditelnosti navrhovaného IS jsem vytvořil třídu pro testování komunikace prostřednictvím vybraného API. Z výsledků při testování jsem vyhodnotil, že navrhovaná aplikace je proveditelná. Během testování jsem se zabýval i dalšími funkcemi, které RouterOS poskytuje, z čehož jsem vyhodnotil, že aplikaci bude možné rozšiřovat.

V ekonomickém vyhodnocení jsem vycházel z mých zkušeností u reálného ISP. Provedl jsem výpočet nákladů na realizaci systému. Zabýval jsem se také náklady na provoz a údržbu. Jelikož jsem zvolil technologie, které je možné využít bezplatně i pro komerční využití, jsou náklady na provoz a údržbu velmi nízké. Pomocí metody NPV jsem spočítal návratnost investice do tohoto systému a ověřil, kdy se systém vyplatí vytvořit a nasadit do provozu. Z výsledků

vyplývá, že systém se vyplatí ve většině uvažovaných variant. Také jsem vyhodnotil limity navrhovaného řešení a možnosti rozšíření systému. Na konci ekonomického vyhodnocení jsem vytvořil shrnutí ve formě SWOT analýzy.

Z provedené analýzy, návrhu a ekonomického vyhodnocení vyplývá, že IS je proveditelný, z hlediska nákladů na implementaci a údržbu není finančně náročný a investice má ve většině případů dobrou návratnost. Jelikož systém může přinést i další benefity, pak každý ISP bez systému pro kontrolu plateb by měl systém nasadit do provozu.

Jelikož jsem systém vyhodnotil jako proveditelný, následuje po této studii implementace navrhovaného systému. Šance na úspěšnou implementaci je díky detailnímu ověření proveditelnosti velmi vysoká.

Systém budu implementovat v nejbližší možné době. Díky kontaktům na techniky ISP mohu systém testovat v reálném prostředí a postupem času podle dalších požadavků systém rozšiřovat o další funkce. Do budoucna plánuji rozšířit systém v rámci diplomové práce o webovou aplikaci, která spolu s touto již navrženou aplikací bude tvořit ucelené řešení managementu klientů a síťových prvků menších ISP.

Literatura

- [1] Vymětal, D.: *Informační systémy v podnicích*. Praha: Grada Publishing, a.s., 2009, ISBN 978-80-247-3046-2, 14 s.
- [2] Šárka Vavrečková: Počítačová síť a internet [online]. [cit. 2017-02-07]. Dostupné z: <http://vavreckova.zam.slu.cz/obsahy/pocsit/skripta/sitebc.pdf>
- [3] MikroTik: About [online]. [cit. 2017-02-07]. Dostupné z: <https://mikrotik.com/aboutus>
- [4] MikroTik: Software [online]. [cit. 2017-03-19]. Dostupné z: <https://mikrotik.com/software>
- [5] MikroTik: Manual:License [online]. [cit. 2017-03-19]. Dostupné z: <https://wiki.mikrotik.com/wiki/Manual:License>
- [6] MikroTik: Manual:RouterOS features [online]. [cit. 2017-03-19]. Dostupné z: https://wiki.mikrotik.com/wiki/Manual:RouterOS_features
- [7] MikroTik: Manual:System/Packages [online]. [cit. 2017-03-19]. Dostupné z: <https://wiki.mikrotik.com/wiki/Manual:System/Packages>
- [8] MikroTik: Manual:Winbox [online]. [cit. 2017-03-19]. Dostupné z: <https://wiki.mikrotik.com/wiki/Manual:Winbox>
- [9] MikroTik: Products [online]. [cit. 2017-03-19]. Dostupné z: <https://routerboard.com/products/group/0>
- [10] ISPadmin: Co je ISPadmin? [online]. [cit. 2017-04-18]. Dostupné z: <https://www.ispadmin.eu/cz/>
- [11] i4wifi Martin: Komplexní informační systém pro poskytovatele [online]. [cit. 2017-04-19]. Dostupné z: <http://i4wifi.blog.cz/1008/komplexni-informacni-system-pro-poskytovatele>

- [12] LMS developers: About LMS [online]. [cit. 2017-04-19]. Dostupné z: <http://lms.rulez.pl/about.php>
- [13] LMS developers: LMS - LAN Management System 1.11-cvs [online]. [cit. 2017-04-19]. Dostupné z: http://lms.rulez.pl/doc_en/index.html
- [14] Anthill s.r.o.: Základní části informačního systému [online]. [cit. 2017-04-19]. Dostupné z: <http://www.loginportal.cz/o-systemu/casti-systemu>
- [15] Anthill s.r.o.: Moduly informačního systému LoginPortal [online]. [cit. 2017-04-19]. Dostupné z: <http://www.loginportal.cz/o-systemu/moduly>
- [16] Anthill s.r.o.: LoginPortal - cenová kalkulace [online]. [cit. 2017-04-19]. Dostupné z: <http://www.loginportal.cz/cena>
- [17] MikroTik: Manual:The Dude [online]. [cit. 2017-04-19]. Dostupné z: https://wiki.mikrotik.com/wiki/Manual:The_Dude
- [18] Čápka, D.: 1. díl - Úvod do formulářových aplikací v Java Swing [online]. [cit. 2017-03-21]. Dostupné z: <http://www.itnetwork.cz/java/swing/java-tutorial-swing-okenni-aplikace-uvod>
- [19] Čápka, D.: 1. díl - Úvod do JavaFX [online]. [cit. 2017-03-21]. Dostupné z: <http://www.itnetwork.cz/java/javafx/java-tutorial-uvod-do-javafx>
- [20] JCraft, Inc.: JSch - Java Secure Channel [online]. [cit. 2017-04-11]. Dostupné z: <http://www.jcraft.com/jsch/>
- [21] Janisk: API in Java [online]. [cit. 2017-04-14]. Dostupné z: https://wiki.mikrotik.com/wiki/API_in_Java
- [22] LeGrange, G.: MikroTik-Java [online]. [cit. 2017-04-15]. Dostupné z: <https://github.com/GideonLeGrange/mikrotik-java>
- [23] The Apache Software Foundation: Apache Licence version 2.0 [online]. [cit. 2017-04-15]. Dostupné z: <https://www.apache.org/licenses/LICENSE-2.0>
- [24] MikroTik: Manual:Tools/Ping [online]. [cit. 2017-04-18]. Dostupné z: <https://wiki.mikrotik.com/wiki/Manual:Tools/Ping>
- [25] MikroTik: Manual:Tools/Bandwidth Test [online]. [cit. 2017-04-18]. Dostupné z: https://wiki.mikrotik.com/wiki/Manual:Tools/Bandwidth_Test

- [26] MikroTik: Manual:Interface/Wireless [online]. [cit. 2017-04-18]. Dostupné z: <https://wiki.mikrotik.com/wiki/Manual:Interface/Wireless>
- [27] MikroTik: Manual:Queue [online]. [cit. 2017-04-21]. Dostupné z: <https://wiki.mikrotik.com/wiki/Manual:Queue>
- [28] Profesia CZ, spol. s r.o.: Platy v kategorii: Informační technologie [online]. [cit. 2017-04-22]. Dostupné z: <http://www.platy.cz/platy/informacni-technologie>
- [29] Kolesárová; Tomek: Využívání počítače a internetu [online]. [cit. 2017-04-23]. Dostupné z: <https://www.scribd.com/doc/133579677>
- [30] Air Bank, a. s.: Přehled úročení [online]. [cit. 2017-05-06]. Dostupné z: <https://www.airbank.cz/produkty/sporici-ucet/>

Seznam použitých zkratek

IS Informační systém

GUI Grafické uživatelské rozhraní

UC Příklad užití

ISP Poskytovatel internetového připojení

WISP Poskytovatel bezdrátového internetového připojení

MVC Model-View-Controller architektura

API Aplikační programové rozhraní

NPV Čistá současná hodnota

Obsah přiloženého média

	readme.txt	stručný popis obsahu média
	config	konfigurace testovacích zařízení
	obr	obrázky
	GUI	obrázky GUI
	src	
	GUI	zdrojové kódy obrazovek
	impl	zdrojové kódy s testy skriptů
	thesis	zdrojová forma práce ve formátu L ^A T _E X
	text	text práce
	BP_Korel_Lukas_2017.pdf	text práce ve formátu PDF
	BP_Korel_Lukas_2017_zadani.pdf	zadání práce ve formátu PDF