Univerzita Hradec Králové
Fakulta informatiky a managementu
Katedra informatiky a kvantitativních metod

Automatizované end-to-end testování SMS a MMS center

Diplomová práce

Autor: **Dominik Plašil**
Studijní obor: Aplikovaná informatika

Vedoucí práce: Ing. Pavel Kříž, Ph.D.
Odborný konzultant: Ing. Pavel Dejmek

Hradec Králové duben 2020

Prohlášení:

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 27. dubna 2020 Dominik Plašil

Poděkování:

Děkuji vedoucímu diplomové práce Ing. Pavlu Kříži, Ph.D. za metodické vedení práce, cenné připomínky a rady během tvorby této práce.

Anotace

V této diplomové práci je popsána architektura mobilních sítí, možnosti připojení a proces standardizace jejich funkcí. Následuje popis SMS a MMS zpráv zejména zaměřený na mobilní centra, jejich funkce a poskytované služby. U obou center je popsáno jejich fungování, rozhraní pro komunikaci s ostatními prvky sítě nebo externími systémy a proces odesílání i doručování textových a multimediálních zpráv. Tento proces je popsán z pohledu mobilního telefonu i externí specializované aplikace poskytující přidanou službu.

V druhé části práce jsou popsány možnosti automatizovaného testování těchto center a rešerše některých existujících řešení. Následuje analýza zadání a vývoj vlastního systému umožňujícího automatizované end-to-end testování mobilních center pro společnost O2 Czech Republic a.s. V závěru práce jsou uvedeny některé problémy, které vznikly při vývoji systému a celkové shrnutí dosažených výsledků.

Annotation

Title: Automated end-to-end testing of SMS and MMS centers

This Diploma Thesis describes the architecture of mobile networks environment, connectivity options and the process of standardizing their functions. The following is a description of SMS and MMS messages mainly focused on mobile centers, their functions and provided services. There is a description of both centers' operations, interfaces for communication with other network elements or external systems and the process of sending and delivering text and multimedia messages. This process is described from the perspective of the mobile phone and external specialized application providing added service.

The second part describes the possibilities of automated testing of these centers and research of some existing solutions. The following is an analysis of the assignment and development of own system for automated end-to-end testing of mobile centers for company O2 Czech Republic a.s. At the end of the thesis are mentioned some problems that arose during application development and overall summary of achieved results.

Obsah

1 Úvod ... 1

2 Cíl práce ... 2

3 SMS, MMS a související technologie ... 3

3.1 Standardizace ... 3

3.2 Připojení k mobilní síti ... 4

3.2.1 HLR a VLR ... 6

3.2.2 SIM karta ... 6

3.3 SMS zprávy a architektura SMS sítě ... 7

3.4 SMS centra ... 8

3.5 Protokol SMPP ... 8

3.5.1 Připojení a odpojení od SMSC ... 9

3.5.2 Odeslání SMS zprávy ... 10

3.5.3 Přijetí SMS zprávy ... 10

3.5.4 Reprezentace čísel ... 11

3.5.5 TLV ... 12

3.5.6 Formát PDU ... 12

3.6 MMS zprávy a architektura MMS sítě ... 13

3.7 MMS centra ... 14

3.7.1 MM1 ... 16

3.7.2 MM4 ... 17

3.7.3 MM7 ... 18

3.7.4 Reprezentace adres ... 21

3.7.5 Další vlastnosti mobilní center ... 22

4 Automatizované testování ... 24

4.1 Rizika pro mobilní centra ... 24

4.2 Rešerše existujících řešení 25
4.2.1 TestMySMS 26
4.2.2 SIGOS 26
4.2.3 NowSMS 26
4.3 Shrnutí existujících řešení 27
5 Vývoj systému pro end-to-end testování SMS a MMS center 28
5.1 Zadání a seznam požadavků 28
5.1.1 Příklady testovacích scénářů 28
5.2 Analýza požadavků 29
5.2.1 Diagram případů užití 29
5.3 Návrh realizace systému 31
5.4 Aplikace MMS Params 32
5.5 Programovací jazyk a použité technologie 32
5.6 Komunikace mezi komponentami systému 33
6 Vývoj systému 35
6.1 Moduly aplikace 35
6.1.1 MmsParamsAPI 35
6.1.2 MmsParamsClientLib 36
6.1.3 MmsParamsSMSC 36
6.1.4 MmsParamsMMSC 36
6.1.5 MmsParamsServer 37
6.1.6 MmsParamsAndroid 37
6.2 Komunikace přes WebSocket 38
6.2.1 Proces deserializace zpráv 40
6.3 Testovací scénář z pohledu klienta 42
6.3.1 Rozhraní ITestInstance 44

6.4 Testovací scénář z pohledu serveru 47
6.5 Testovací scénář z pohledu webového rozhraní 48
6.6 Testovací scénář z pohledu telefonu 49
6.7 Další funkce systému 50
6.7.1 Dokumentace 50
6.7.2 Chyby v testovacích scénářích 51
6.7.3 Zabezpečení 51
6.7.4 Logování 53
6.7.5 Vzdálené připojení telefonu 54
6.7.6 Firebase 54
7 Testování hotového řešení 55
7.1 Ukázkové testovací scénáře 55
7.2 Rozšiřitelnost systému 58
7.3 Problémy při vývoji systému 59
7.3.1 Odpojování mobilní aplikace 59
7.3.2 Rozpoznávání příchozích zpráv 59
7.4 Jak systém získat a nainstalovat 60
8 Závěr 61

Seznam obrázků

Obrázek 1 Schéma BSS [7] 5
Obrázek 2 Schéma PLMN [7] 5
Obrázek 3 Struktura sítě při přenosu SMS zpráv [9] 7
Obrázek 4 Diagram SMPP sítě [12] 9
Obrázek 5 Průběh komunikace přes SMPP [12] 11
Obrázek 6 Základní formát PDU v SMPP [12] 12
Obrázek 7 Schéma prostředí MMSE [14] 14
Obrázek 8 Architektura MMS sítě [15] 15
Obrázek 9 MM rozhraní MMS centra [14] 16
Obrázek 10 Odeslání a přijetí MMS zprávy pomocí PDU z rozhraní MM1 [12] 17
Obrázek 11 Schéma přenosu zpráv pomocí rozhraní MM4 [14] 18
Obrázek 12 Odeslání MMS zprávy přes MM7 [14] 20
Obrázek 13 Přijetí MMS zprávy přes MM7 [14] 21
Obrázek 14 MMS zpráva doručená na email 23
Obrázek 15 Use Case diagram testování SMS zpráv 30
Obrázek 16 Use Case diagram testování MMS zpráv 30
Obrázek 17 Komunikace mezi komponentami systému 34
Obrázek 18 Diagram modulů systému 35
Obrázek 19 UML diagram tříd pro základní komunikaci 40
Obrázek 20 Proces deserializace zprávy na serveru 41
Obrázek 21 Rozhraní ITestInstance 44
Obrázek 22 Diagram čekaní na přijetí zprávy 47
Obrázek 23 Webové rozhraní – seznam testovacích scénářů 48
Obrázek 24 Webové rozhraní – detail scénáře 49
Obrázek 25 Mobilní aplikace – nastavení připojení 50
Obrázek 26 Autentizace pomocí JWT tokenu [24] 52
Obrázek 27 Diagram testovacího scénáře SmsTest1 56

Seznam tabulek

Tabulka 1 Příklady použití TON a NPI 12
Tabulka 2 Atributy hlavičky zprávy 38
Tabulka 3 Přehled služeb pro testovací scénáře 44
Tabulka 4 Typy chyb testovacích scénářů 51

Seznam ukázek kódu

Ukázka kódu 1 Proces deserializace a rozpoznání zprávy 41
Ukázka kódu 2 Metoda runTest v třídě RunnableTest 43
Ukázka kódu 3 Šablona testovacího scénáře 43
Ukázka kódu 4 Odeslání požadavku se SMS zprávou 46
Ukázka kódu 5 Řetězec pro vzdálené připojení mobilní aplikace 54
Ukázka kódu 6 Zdrojový kód testovacího scénáře SmsTest1 57

Použité zkratky a pojmy

SMS	*Short Message Service* služba krátkých textových zpráv
SMSC	*Short Message Service Center* prvek mobilní sítě umožňující ukládat, převádět a doručovat SMS zprávy
MMS	*Multimedia Messaging Service* služba multimediálních zpráv
MMSC	*Multimedia Messaging Service Center* prvek mobilní sítě umožňující ukládat, převádět a doručovat MMS zprávy
3GPP	*3rd Generation Partnership Project* projekt sedmi organizací vytvářejících telekomunikační standardy
OMA	*Open Mobile Alliance* nezisková organizace vytvářející otevřené standardy pro průmysl mobilních telefonů a telekomunikačních služeb
W3C	*World Wide Web Consortium* mezinárodní standardizační organizace
GSM	*Global System for Mobile Communications* standard popisující protokoly pro mobilní sítě druhé generace
GPRS	*General Packet Radio Service* paketově orientovaný mobilní datový standard v systému mobilní komunikace
LTE	*Long-Term Evolution* telekomunikační technologie pro vysokorychlostní přenos dat v mobilních sítích
PLMN	*Public Land Mobile Network* kombinace bezdrátových komunikačních služeb nabízených konkrétním mobilním operátorem v určité zemi
MCC	*Mobile country code* třímístné číslo identifikující oblast nebo zemi

MNC *Mobile network code*
dvou nebo třímístné číslo identifikující mobilního operátora v dané oblasti nebo zemi

BTS *Base Transceiver Station*
základní vysílací zařízení, které umožňuje bezdrátovou komunikaci mezi uživatelským zařízením a mobilní sítí

BSC *Base Station Controller*
zařízení, které spravuje jednu nebo více BTS; vyměňuje mezi nimi informace a předává je do MSC

BSS *Base Station Subsystem*
skupina zařízení v mobilní síti, která jsou zodpovědná za přenos a řízení dat mezi mobilními zařízeními a MSC

MSC *Mobile Switching Center*
zařízení zodpovědné za směrování a spojování telefonních hovorů a přenos SMS zpráv

SIM *Subscriber Identity Module*
karta nebo čip uchovávající hodnoty uživatele určené k jeho autentizaci v mobilní síti

IMSI *International mobile subscriber identity*
číslo, které jednoznačně identifikuje každého uživatele mobilní sítě

MSISDN *Mobile Subscriber ISDN Number*
celosvětově unikátní číslo, které identifikuje účastníka v mobilní síti

ISDN *Integrated Services Digital Network*
komunikační standard pro simultánní digitální přenos zvuku, videa nebo jiných dat

ICCID *Integrated Circuit Card Identifier*
mezinárodní identifikátor SIM karty

MSIN *Mobile Subscription Identification Number*
číslo přidělované mobilním operátorem, které identifikuje uživatele mobilní sítě

PIN *Personal Identification number*
číselné nebo alfanumerické heslo sloužící pro autentizaci uživatele

APN *Access Point Name*
název brány mezi mobilní sítí a internetem nebo jinou počítačovou sítí

UCS-2 *Universal Coded Character Set*
standard kódování znaků ve kterém jsou znaky reprezentované fixní délkou 16 bitů

ETSI *European Telecommunications Standards Institute*
nezávislá nezisková organizace vytvářející standardizace pro telekomunikační průmysl v Evropě

SS7 *Signaling System No. 7*
skupina signalizačních protokolů používaných pro uskutečňování telefonních hovorů, doručování SMS zpráv, účtování poplatků a dalších služeb

SMPP *Short Message Peer-to-Peer*
otevřený průmyslový standardní protokol pro přenos krátkých textových zpráv mezi externími zařízeními a SMS centry

TLV *Tag-length-value*
schéma kódování dodatečné informace v určitém protokolu

SME *Short Message Entities*
aplikace nebo zařízení, které umožňuje odesílat a přijímat SMS zprávy

ESME *External Short Message Entities*
externí aplikace, které se připojují k SMS centru a odesílají nebo přijímají přes něj SMS zprávy

PDU *Protocol Data Unit*
jednotka informace pro přenos dat mezi zařízeními v počítačové nebo mobilní síti

MMSE *Multimedia Messaging Service Environment*
prostředí MMS sítě obsahující skupinu prvků nebo komponent spravované mobilním operátorem

VASP *Value-added service provider*
externí systém nebo aplikace poskytující přidanou hodnotu; v oblasti telekomunikací je to systém, který se připojuje k MMS centrům

SMTP *Simple Mail Transfer Protocol*
komunikační protokol pro přenos elektronických zpráv

XML *Extensible Markup Language*
značkovací jazyk, který definuje několik pravidel pro ukládání dokumentů v podobě, která snadno čitelná člověkem i strojem

SMIL *Synchronized Multimedia Integration Language*
rozšíření jazyka XML pro popis prezentace obsahu; umožňuje animace, rozložení, časování a další

MIME *Multipurpose Internet Mail Extensions*
internetový standard, který umožňuje přenášet texty v různých kódováních, binární data a vícedílné zprávy

SOAP *Simple Object Access Protocol*
protokol pro výměnu zpráv založených na XML

RDF *Resource Description Framework*
specifikace pro návrh metadat datového modelu

JSON *JavaScript Object Notation*
otevřený standard formátu dat, který definuje několik pravidel pro ukládání dokumentů v podobě, která snadno čitelná člověkem i strojem

DDoS *Denial-of-service*
typ útoku na internetové stránky nebo služby, jehož cílem je službu znefunkčnit nebo znepřístupnit ostatním uživatelům

JWT *JSON Web Token*
internetový standard pro vytváření přístupových tokenů založených na formátu JSON

1 Úvod

V oblasti vývoje softwaru jsou metody testování poměrně rozšířené a propracované. Existuje řada knihoven a nástrojů pro vytváření, spouštění a integraci testů v rámci celého procesu vývoje softwaru. Podobně je na tom i testování klíčových systémů a infrastruktury firem. V oblasti mobilních sítí jsou to zejména SMS a MMS centra. Každý mobilní operátor musí mít naprostou jistotu, že jejich servery a mobilní centra fungují zcela bezchybně a že poskytují téměř stoprocentní dostupnost všem koncovým zákazníkům.

Při odesílání SMS a MMS zpráv lze nastavit široké množství parametrů definujících jejich vlastnosti a případně i chování. Mobilní centra pak musí tyto parametry správně vyhodnocovat a zpracovávat, aby bylo dosaženo požadovaného chování, které zákazník očekává. Avšak při aktualizacích softwaru těchto center může potencionálně docházet k chybám způsobujících špatné vyhodnocování některých parametrů zpráv. To může ohrozit spolehlivost systému a následně i důvěryhodnost společnosti poskytující tyto služby. Tomuto problému lze předejít procesem testování, který však může být poměrně zdlouhavý a náročný právě z důvodu rozsáhlých možností nastavení parametrů.

Tato práce navazuje na bakalářskou práci [1], jejímž cílem bylo vytvořit mobilní aplikaci pro podporu testování SMS a MMS center. Cílem této aplikace bylo umožnit nastavit široké množství parametrů zpráv, které jsou v běžných aplikacích pro komunikaci většinou nedostupné. Po odeslání nebo přijetí zprávy bylo možné analyzovat všechny parametry a zjistit tak, jestli byla zpráva zpracována korektně dle očekávání. Nicméně toto řešení bylo značně omezené na nutnost provádět toto testování manuálně technikem mobilního centra. Cílem této práce je tuto aplikaci rozšířit a proces testování automatizovat.

2 Cíl práce

Cílem této práce je popsat princip a fungování SMS a MMS center z pohledu mobilních zařízení a externích systémů, které je využívají. Dalším cílem je rozšířit již existující aplikaci a vytvořit tak systém pro automatizované end-to-end testování SMS a MMS center, který umožní administrátorům mobilních center vytvářet různé testovací scénáře, které budou poskytovat možnost rychle a snadno otestovat velké množství funkcionalit nebo chování naráz. Testovací scénáře budou umožňovat odesílat SMS a MMS zprávy pomocí fyzických mobilních zařízení, ale také se připojovat přímo k SMS a MMS centrům a zprávy odesílat přes ně. Parametry přijatých zpráv budou následně analyzovány a ověřovány, zdali mají odpovídající hodnoty a tedy, že centrum funguje správně.

3 SMS, MMS a související technologie

Jedněmi z nejpoužívanějších komunikačních služeb jsou SMS a MMS zprávy. Prvotní koncept těchto služeb vznikl v osmdesátých letech dvacátého století a od té doby zaznamenal velký pokrok jak ve vývoji, tak v počtu uživatelů využívajících tyto služby každý den. Snížení poplatků a zlepšení dostupnosti zařadilo tuto technologii mezi nejpopulárnější a nejpoužívanější komunikační technologie vůbec. [2] Nicméně v posledních letech jsou tyto služby na ústupu hlavně kvůli masivnímu rozšiřování mobilního internetu a zlepšování jeho dostupnosti. Díky tomu začali uživatelé více využívat jiné komunikační nástroje (Email, WhatsApp, Facebook), které jsou většinou zdarma a nabízejí další služby než pouhé odesílání zpráv. [3]

3.1 Standardizace

V oblasti telekomunikačních technologií se pohybuje několik různých subjektů, kteří se zabývají vývojem a výrobou zařízení nebo poskytováním služeb. Všechny tyto subjekty se musejí řídit společnými pravidly, aby byla zajištěna co nejvyšší míra kompatibility mezi jednotlivými zařízeními a produkty. Tato pravidla jsou definována ve specifikacích a standardizacích pro jednotlivá odvětví komunikačních technologií. Může se jednat o způsob komunikace mezi mobilními zařízeními a mobilními centry, uskutečňování telefonních hovorů, odesílání zpráv nebo mobilního připojení k internetu. Standardizace musí také platit v celosvětovém měřítku, aby bylo zajištěno, že zákazník bude moci využívat všechny služby, i když překročí hranice státu nebo bude například cestovat na jiný kontinent.

Cílem standardizací je tedy určení základních konceptů a doporučení, kterými by se měli řídit všichni, kdo chtějí poskytovat služby nebo výrobky na tomto společném trhu. Hlavní standardizace SMS a MMS technologií vytvářejí organizace 3GPP a OMA, přičemž často vycházejí z již existujících technologií vyvinutých konsorciem W3C. [2]

3GPP (3rd Generation Partnership Project) vznikl roku 1998 jako projekt sedmi organizací vytvářejících telekomunikační standardy. Projekt vznikl s cílem vytvořit specifikace pro mobilní síť třetí generace založenou na již existující síti GSM. 3GPP také následně stál za vytvořením specifikací pro sítě GPRS, LTE a 5G. Při vytváření

bylo definováno několik nízko i vysokoúrovňových požadavků služeb, určena základní architektura sítě a vybráno několik vhodných streamovacích protokolů, formátů i kodeků. 3GPP také definuje způsoby účtování poplatků za použití služeb nebo zabezpečení systémů a komunikace. Důležitým aspektem je také zpětná kompatibilita potřebná pro komunikaci mezi staršími a novějšími částmi systému nebo sítě. [2, 4]

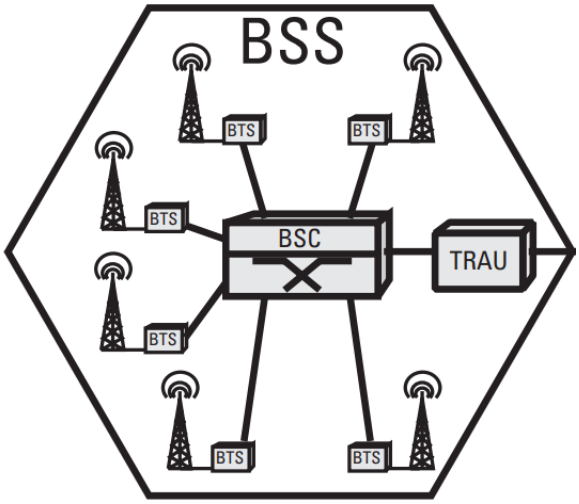
OMA (Open Mobile Alliance) je nezisková organizace s více než 220 členy. OMA byla založena nejvýznamnějšími výrobci a dodavateli mobilních zařízení a také některými mobilními operátory. Cílem organizace je vytvářet otevřené standardy a technické specifikace pro průmysl mobilních telefonů a telekomunikačních služeb. Vše má fungovat bez ohledu na operační systém, geografickou lokalitu, typ sítě nebo poskytovatele služeb. Jedná se tedy o konsolidaci různých přístupů ke konkrétním problémům. [5] Specifikace OMA stejně jako 3GPP reagují na aktuální požadavky trhu a průmysl rychle se vyvíjejících a měnících technologií a produktů. Dalším cílem je modularita, která umožňuje rozšiřitelnost, konzistenci a celkové snížení nákladů. Dodržování standardů OMA není povinné, nicméně je doporučeno se jimi řídit, aby nevznikaly problémy např. s kompatibilitou mezi zařízeními různých dodavatelů, což by mohlo způsobit horší uživatelský zážitek a přívětivost služby. [6]

3.2 Připojení k mobilní síti

PLMN (Public Land Mobile Network) je kombinací bezdrátových komunikačních služeb nabízených konkrétním mobilním operátorem v určité zemi. PLMN se většinou skládá z několika bezdrátových technologií jako jsou GSM/2G, UMTS/3G nebo LTE/4G. Každé PLMN je globálně identifikované pomocí tzv. PLMN kódu, který se skládá z MCC (Mobile Country Code), který identifikuje konkrétní zemi a MNC (Mobile Network Code), který identifikuje mobilního operátora. [2]

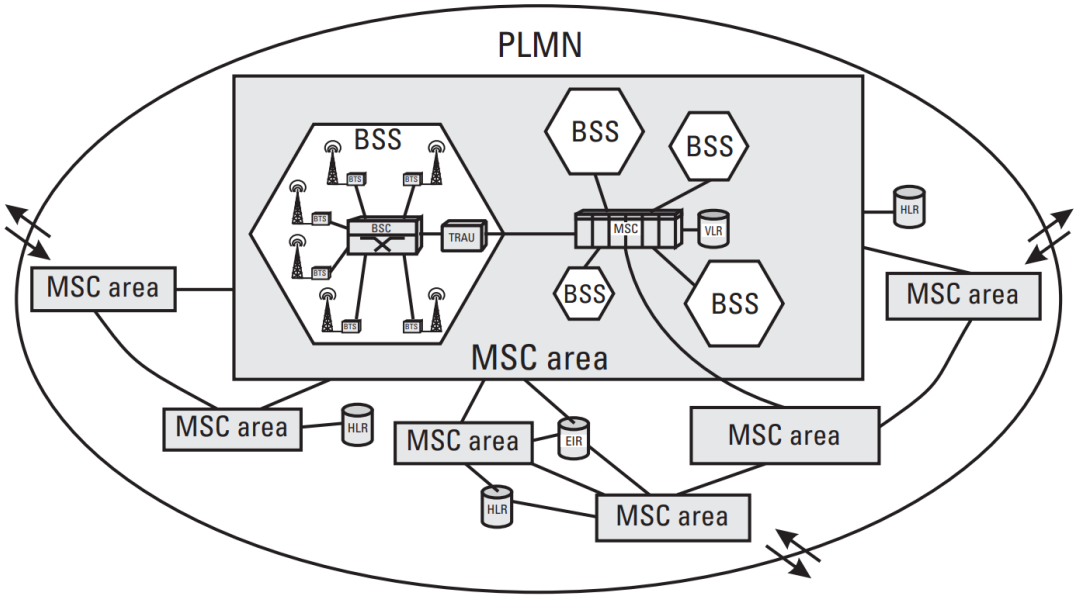
Zařízení uživatele, které umožňuje připojení k mobilní síti se označuje jako MS (Mobile Station). Pro připojení k mobilní síti musí MS používat SIM kartu, která jej registruje v síti daného mobilního operátora. Zařízení MS neustále komunikuje s nejbližším vysílačem (oz. BTS – Base Transceiver Station) a posílá mu informaci, že se nachází v jeho okruhu, resp. v dané části podsítě. BTS dále předává informaci BSC (Base Station Controller). Obě tyto části dohromady tvoří BSS (Base Station

Subsystem), což je označení pro podsíť mobilní sítě a skládá se vždy z BSC a několika BTS. Obrázek 1 ukazuje příklad celého BSS graficky. [7]



Obrázek 1 Schéma BSS [7]

Další komponentou mobilní sítě PLMN je MSC (Mobile Switching Center), kterému je přiřazena nějaká oblast s jedním nebo více BSS. MSC je mimo jiné zodpovědné za směrování a spojování telefonních hovorů a přenos SMS zpráv, ať už v rámci dvou BSS nebo mezi různými MSC oblastmi. MSC se také stará o účtování poplatků za použité služby a umožňuje sledovat stav kreditu předplacených karet v reálném čase. [7] Obrázek 2 následně ukazuje možné schéma celé sítě PLMN.



Obrázek 2 Schéma PLMN [7]

3.2.1 HLR a VLR

HLR (Home Location Register) je centrální databáze, která obsahuje informace o všech uživatelích mobilní sítě. V databázi HLR jsou uloženy informace o každé SIM kartě vydané mobilním operátorem a jednotlivé záznamy tvoří dva primární klíče IMSI a MSISDN. Je důležité, aby tato databáze měla co nejkratší přístupový čas, protože na tom závisí rychlost služeb sítě (např. spojení telefonního hovoru). Databáze také musí být dobře zajištěná proti ztrátě dat anebo výpadkům, protože na ni závisí dostupnost celé služby. [7]

VLR (Visitor Location Register) je další databáze uživatelů mobilní sítě. Oproti HLR je ale tato databáze velmi dynamická a údaje se mění v závislosti na poloze uživatele (resp. jeho zařízení). V databázi VLR jsou pouze ti uživatelé, kteří jsou dostupní v oblasti konkrétního vysílače a poslali mu o tom informaci. Pokud uživatel například vypne svůj mobilní telefon, je z této databáze odstraněn nebo pokud cestuje a vyskytuje se v dosahu jiného z vysílačů, je nutné záznam aktualizovat. [7]

3.2.2 SIM karta

Pro připojení jakéhokoli zařízení k síti mobilního operátora je zapotřebí tzv. SIM karta. Tato karta nebo integrovaný čip obsahuje hodnoty uživatele potřebné pro jeho autentizaci a registraci do mobilní sítě. Každá SIM karta obsahuje několik identifikačních údajů, kterými jsou např. ICCID, IMSI, MSISDN a PIN kód. Unikátní číslo SIM karty ICCID nemůže být změněno a slouží k její jednoznačné identifikaci. Číslo IMSI o maximální délce 15 znaků identifikuje uživatele v síti GSM. IMSI je tvořeno čísly MCC, MNC a MSIN. MSIN (Mobile Subscription Identification Number) je přidělováno mobilním operátorem a identifikuje uživatele mobilní sítě. MSISDN (Mobile Station ISDN Number) je celosvětově unikátní číslo, které identifikuje účastníka v mobilní síti. Toto číslo je zároveň telefonním číslem daného uživatele a obvykle začíná mezinárodním přestupným znakem „+“ nebo „00“, následovaným kódem země a číslem samotným. [7, 8]

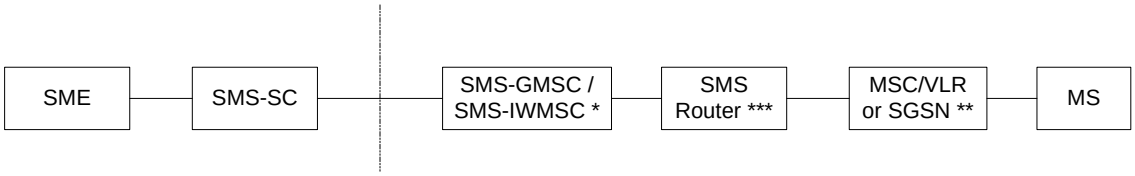
Pokud chce uživatel mobilní sítě odesílat MMS zprávy nebo používat mobilní internet, musí mít v telefonu správně nastavené APN (Access Point Name). To je většinou nastaveno automaticky po vložení SIM karty do telefonu a načtení čísla

IMSI. APN tvoří adresa MMS centra, IP adresa proxy serveru a jeho port. Dále může obsahovat uživatelské jméno a heslo pro přihlášení nebo typ autentizace. [7]

3.3 SMS zprávy a architektura SMS sítě

Služba SMS umožňuje odesílání a přijímání krátkých textových zpráv mezi mobilními zařízeními. Jedna SMS zpráva může obsahovat pouze text s maximální délkou 160 znaků, pokud jsou použity znaky z abecedy GSM (7bit) a při použití jiných speciálních znaků nebo diakritiky se délka zkracuje na 70 znaků a je použito kódování USC-2 (16bit). Standardizaci pro SMS dříve vytvářela organizace ETSI, nyní je však plně pod záštitou 3GPP. [2]

Prvky mobilní sítě, které jsou schopny odeslat a přijmout SMS zprávy, jsou označovány jako SME (Short Message Entities). SME může být softwarová aplikace v mobilním zařízení, FAX nebo server. Mobilní zařízení musí být správně nastaveno, aby mohlo správně fungovat v mobilní síti, přičemž jeho nastavení většinou probíhá již ve fázi výroby, ale může být provedeno i manuálně přímo uživatelem. Obrázek 3 ukazuje prvky sítě, přes které je SMS zpráva odesílána a následně doručována. [2]



Obrázek 3 Struktura sítě při přenosu SMS zpráv [9]

Jak je popsáno v kapitole 3.2, zařízení uživatele se vyskytuje v dosahu nějakého vysílače, resp. MSC. Když uživatel odešle novou SMS zprávu, je nejprve odeslána na nejbližší vysílač a odtud do MSC. Dále zpráva putuje do IWMSC (Inter Working Mobile Switching Centre), což je brána SMS centra. SMS centrum zprávu zpracuje a určí jejího příjemce. Při doručování zprávy komunikuje SMS centrum přes bránu GMSC (Gateway Mobile Switching Centre) se SMS Routerem, jehož úkolem je určit, v jaké podsíti (v dosahuje jakého vysílače) se příjemce nachází. SMS Router může také využít seznam HRL. Následně je zpráva odeslána do konkrétní podsítě a vysílače, ze kterého je doručena příjemci. Pokud je zařízení příjemce nedostupné, neexistuje o něm záznam v žádném seznamu VLR a SMS Router nedokáže určit, kam

zprávu doručit. V takové případě je SMS zpráva uložena na SMS centru a je doručena až ve chvíli, kdy je zařízení příjemce opět aktivní. [2, 9]

3.4 SMS centra

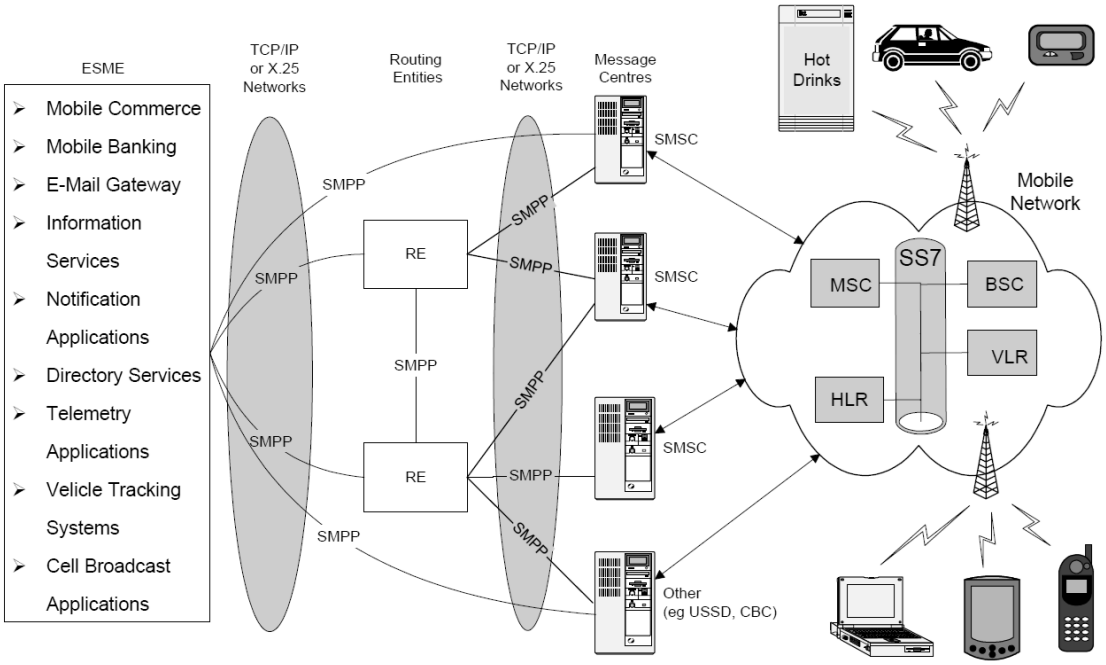
SMSC (SMS Center) někdy také nazývané SC (Service Center) je komponentou v architektuře sítě mobilního operátora a hraje klíčovou roli při odesílání a doručování SMS zpráv. Hlavní funkcí SMSC je přenos krátkých textových zpráv mezi koncovými mobilními zařízeními, ale také jejich uložení a pozdější doručení, pokud zrovna není příjemce dostupný. SMS centrum může být integrováno jako součást mobilní sítě (MSC – Mobile Switching Center) nebo jako samostatná síťová jednotka. Centrum může být dokonce umístěno mimo celou síť a být spravováno nějakou třetí stranou. Je velice běžné, že mobilní operátor má jedno nebo více SMS center, a naopak je teoreticky možné, aby bylo jedno SMSC využíváno více operátory. [2, 10] SMS zpráva odeslaná z jednoho SMSC v jedné síti může být doručena příjemci v jiné mobilní síti (provozované jiným operátorem). SMS centrum se v takovém případě snaží doručit zprávu pomocí protokolu SS7 (Signaling System No. 7) přímo na zařízení příjemce. [2]

Se SMS centry mohou komunikovat i externí systémy, což významně rozšiřuje možnosti jejich využití. Lze tak snadno hromadně odesílat velké množství SMS zpráv nebo použít tento systém pro různé registrace, dvoufázová přihlašování nebo např. zakoupení jízdenky do městské hromadné dopravy. [11]

3.5 Protokol SMPP

SMPP (Short Message Peer-to-Peer) je otevřený průmyslový standardní protokol, který poskytuje flexibilní komunikační rozhraní pro přenos krátkých textových zpráv mezi externími zařízeními nebo službami a SMS centry. V praxi to často vypadá tak, že je vytvořena aplikace, která přes SMPP komunikuje se SMS centrem a posílá mu příkazy k odeslání SMS zpráv. Pokud naopak SMS centrum nějakou zprávu obdrží, je přesměrována do externí aplikace. SMS zprávy odeslané s pomocí protokolu SMPP mají většinou adresu odesílatele v alfanumerickém nebo „short code“ formátu. Uživatel může následně použít tuto adresu jako adresu příjemce pro odeslání svého požadavku nebo odpovědi. [12]

SMPP spolu s externími aplikacemi mají široké spektrum možných využití a může se jednat například o: upozornění na zprávy v hlasové schránce; upozornění studenta, že byla lekce zrušena; informování administrátorů o problémech při provozu serveru; zprávy informačních služeb, např. o dopravních zácpách, předpovědi počasí, stavu akcií na burze; informace založené na poloze uživatele; dvoufázové přihlašování do systému. [12] Obrázek 4 pak ukazuje různé možnosti komunikace v síti. V levé části jsou externí ESME aplikace (External Short Message Entities), které přes SMPP komunikují se SMS centry. Ty mohou přijímat nebo doručovat zprávy několika různým zařízením, které jsou k mobilní síti připojeny.



Obrázek 4 Diagram SMPP sítě [12]

3.5.1 Připojení a odpojení od SMSC

Komunikaci se SMS centrem zahajuje externí aplikace, tím že naváže tzv. SMPP session spojení. Navázání probíhá přes protokol TCP/IP, kde SMSC poslouchá na nějaké adrese a portu (nejčastěji 2775). Po připojení ještě nebyly odeslány žádné identifikační údaje, takže je nutné, aby klient (externí aplikace) poslat tzv. bind požadavek. Bind požadavek slouží pro registraci klienta, která je nutná pro další odesílání nebo přijímání zpráv. Bind požadavek může být tedy požadován za jakousi formu přihlášení do systému a obsahuje identifikační údaje system_id a password.

Dále obsahuje parametr system_type, který nabývá hodnot Transmitter, Receiver nebo Transceiver. Podle tohoto parametru je určeno, jestli chce klient zprávy pouze odesílat, přijímat nebo obojí. Pokud proběhla registrace klienta v pořádku, centrum vrátí PDU s potvrzením, v opačném případě PDU s chybou. [12]

3.5.2 Odeslání SMS zprávy

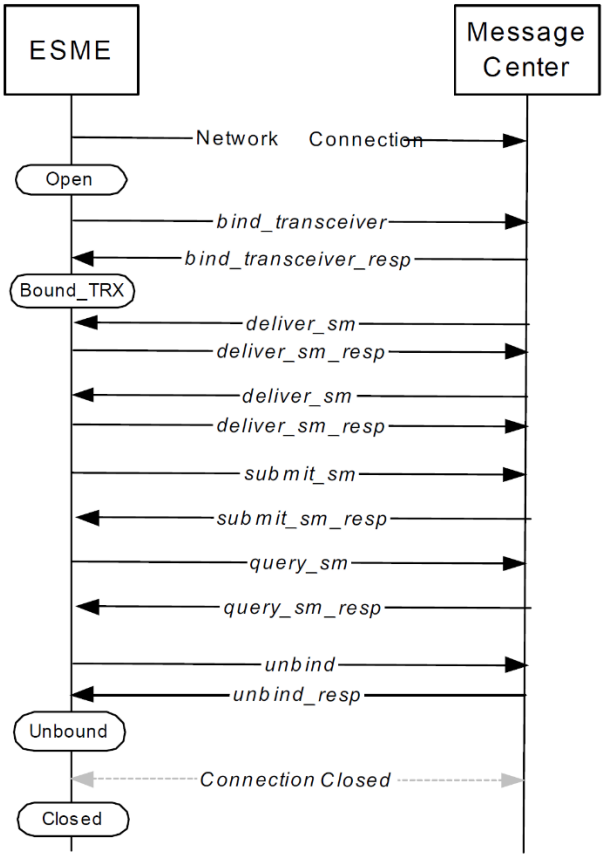
Po připojení k SMS centru a registraci může externí systém odesílat požadavky pro odeslání nových SMS zpráv. Nová SMS zpráva se odešle vytvořením PDU balíčku typu submit_sm, vyplněním požadovaných parametrů a odesláním centru. Mezi klíčové parametry patří číslo odesílatele a příjemce, text zprávy, priorita, doba platnosti, naplánovaný čas doručení nebo požadavek na zprávu o doručení. Další možnosti poskytují hodnoty TLV popsané v kapitole 3.5.5. V případě kladného přijetí požadavku na novou zprávu centrem je klientovi doručen PDU balíček typu submit_sm_resp, který obsahuje informaci o stavu vyřízení. [12]

3.5.3 Přijetí SMS zprávy

Doručení SMS zprávy je realizováno obdržením PDU balíčku typu deliver_sm. Přijaté PDU obsahuje podobné parametry jako PDU pro odeslání nové zprávy a pokud se jedná o zprávu přijatou z mobilního telefonu, SMSC automaticky doplní různé TLV parametry, čímž příjemce dostane dodatečné informace o zprávě.

Pokud klientská aplikace úspěšně přijala a zpracovala novou zprávu, měla by SMS centru odeslat PDU typu deliver_sm_resp, kterým mu potvrdí přijetí. Pokud by klient toto PDU neodeslal, může se stát, že se mu bude centrum snažit zprávu doručovat znovu a znovu až do úspěšného potvrzení.

Zpráva o doručení je také doručena jako PDU typu deliver_sm. To znamená, že klient musí sám rozlišit o jaký typ se jedná. Pokud je hodnota atributu esm_class rovna čtyřem, jedná se právě o zprávu o doručení. [12] Obrázek 5 ukazuje průběh celé komunikace při připojení, odeslání SMS zprávy, přijetí zprávy o doručení a následné odpojení od SMS centra.



Obrázek 5 Průběh komunikace přes SMPP [12]

3.5.4 Reprezentace čísel

V protokolu SMPP se adresy nebo telefonní čísla odesílatele i příjemce zprávy reprezentují pomocí tří čísel, resp. jednoho čísla a dvou konstant. Jedná se o TON (Type of Number), NPI (Number Plan Indicator) a Address.

Hodnota Address může být krátké nebo standardní dlouhé číslo nebo alfanumerická hodnota. Alfanumerické adresy mohou obsahovat všechna čísla a malá nebo velká písmena anglické abecedy. TON neboli typ čísla reprezentuje hlavně typ jeho prefixu nebo zdali se jedná o alfanumerickou adresu. U TON se nejčastěji používají hodnoty 1, 3 nebo 5. Zvláštní případ je hodnota 0, která reprezentuje neznámý TON. V takové případě je na SMS centru, aby určilo, o jaký typ čísla se jedná. NPI reprezentuje standard podle kterého bylo telefonní číslo vytvořeno. Tabulka 1 obsahuje příklady kombinací TON a NPI pro různé druhy telefonních čísel. Hodnoty všech konstant a možností lze nalézt v dokumentaci protokolu SMPP. [12, 13]

Tabulka 1 Příklady použití TON a NPI

Formát adresy	TON	NPI
Krátké číslo	3 (Network Specific)	0 (Unknown)
Dlouhé číslo (standardní)	1 (International)	1 (ISDN, E163/E164)
Alfanumerická	5 (Alphanumeric)	0 (Unknown)
Pokud začíná „+“	1 (International)	1 (ISDN E163/E164)
Prázdná adresa	0 (Unknown)	1 (ISDN E163/E164)
Ostatní	0 (Unknown)	0 (Unknown)

3.5.5 TLV

SMPP PDU balíčky mohou obsahovat libovolný počet tzv. TLV (Tagged Length Value). TLV jsou doplňkové volitelné parametry zprávy, které jsou tvořeny jejich klíčem (tag), hodnotou (value) a délkou hodnoty (length). TLV se používají pro definování dalších parametrů zpráv jako je adresa SMS centra, platnost zprávy nebo informace o odesílateli nebo příjemci. TLV jsou využívány pouze SMS centry, mobilní zařízení k nim většinou přístup nemají. [12]

3.5.6 Formát PDU

Obrázek 6 ukazuje základní formát, ze kterého se skládá každé PDU v protokolu SMPP. Na začátku je vždy hlavička, která obsahuje informaci o délce a typu požadavku, následuje jeho stav a číslo sekvence. Hlavička PDU je povinná a musí být vyplněna u každého požadavku. Za hlavičkou následuje samotný obsah požadavku, který obsahuje parametry vyplněné dle typu požadavku ze specifikace SMPP pro dané PDU. [12]

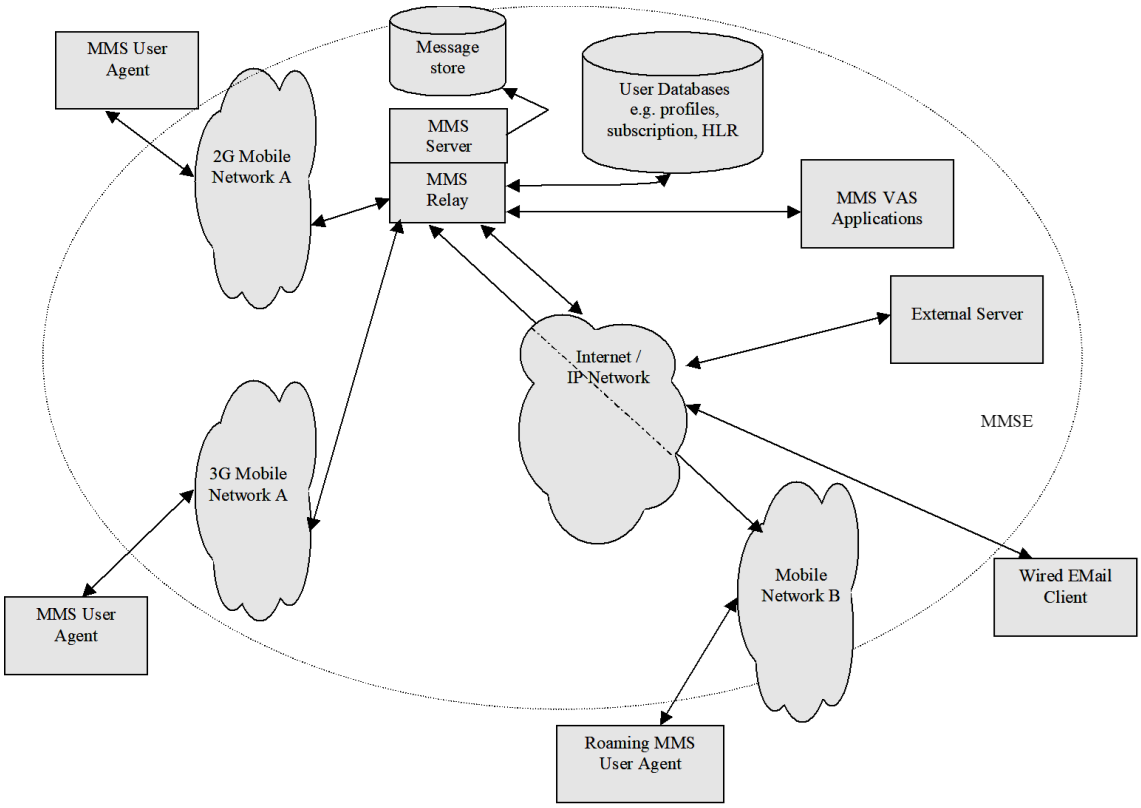
SMPP PDU				
PDU Header (mandatory)				PDU Body (Optional)
Command length	Command id	Command status	Sequence number	PDU Body
4 octets	4 octets	4 octets	4 octets	Length = (Command Length value - 16) octets
4 octets	Command Length - 4			

Obrázek 6 Základní formát PDU v SMPP [12]

3.6 MMS zprávy a architektura MMS sítě

MMS (Multimedia Messaging Service) je služba umožňující výměnu multimediálních zpráv mezi dvěma nebo více uživateli, které mohou obsahovat text spolu s dalšími multimediální soubory jako jsou obrázky, videa nebo např. zvukové nahrávky. Povolná velikost zprávy záleží na pravidlech a nastavení konkrétního mobilního operátora, avšak běžně se uvádí, že velikost celé zprávy včetně všech příloh může být maximálně 300 kB. Pro odeslání nebo přijetí MMS zprávy musí mít uživatel zařízení, které tuto službu umožňuje používat a také musí být připojeno k mobilní síti pomocí technologie GPRS nebo novější. [2]

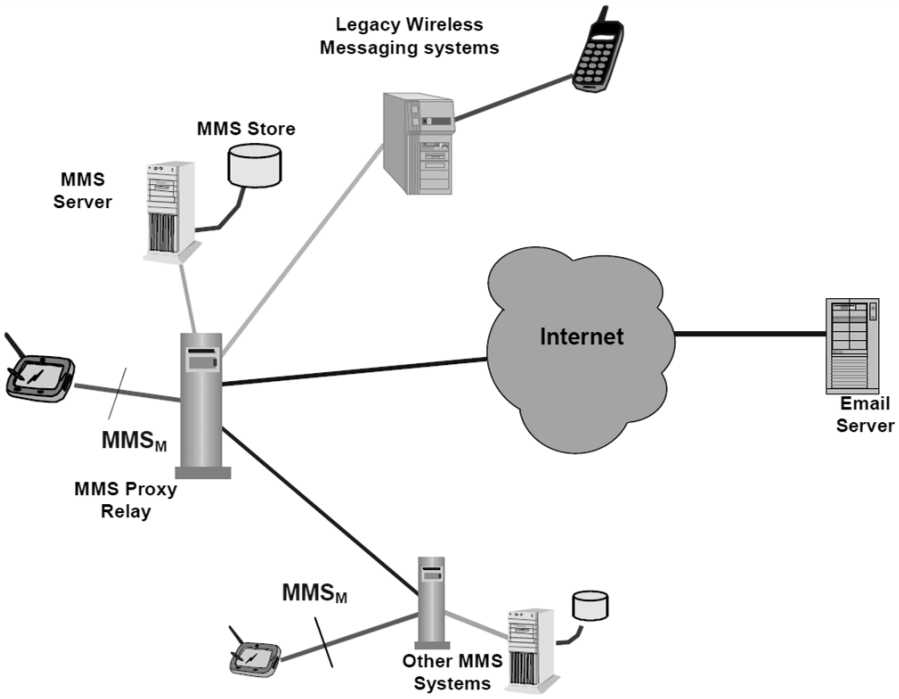
Výměna MMS zpráv probíhá v tzv. MMS síti nebo také MMSE (MMS Environment), které obsahuje několik různých komponent nejčastěji provozovaných mobilním operátorem. Obrázek 7 ukazuje několik prvků, které se mohou v MMSE vyskytovat. Hlavními prvky jsou MMS Server a MMS Relay více popsané v kapitole 3.7. Z obrázku je také patrné, že právě MMS Relay umožňuje komunikovat s dalšími prvky sítě, jako jsou konkrétní podsítě, do kterých jsou připojeny mobilní zařízení uživatelů, ale také s různými databázemi uživatelů, externími aplikacemi VASP (více v kapitole 3.7.3) nebo sítěmi jiných mobilních operátorů. [14]



Obrázek 7 Schéma prostředí MMSE [14]

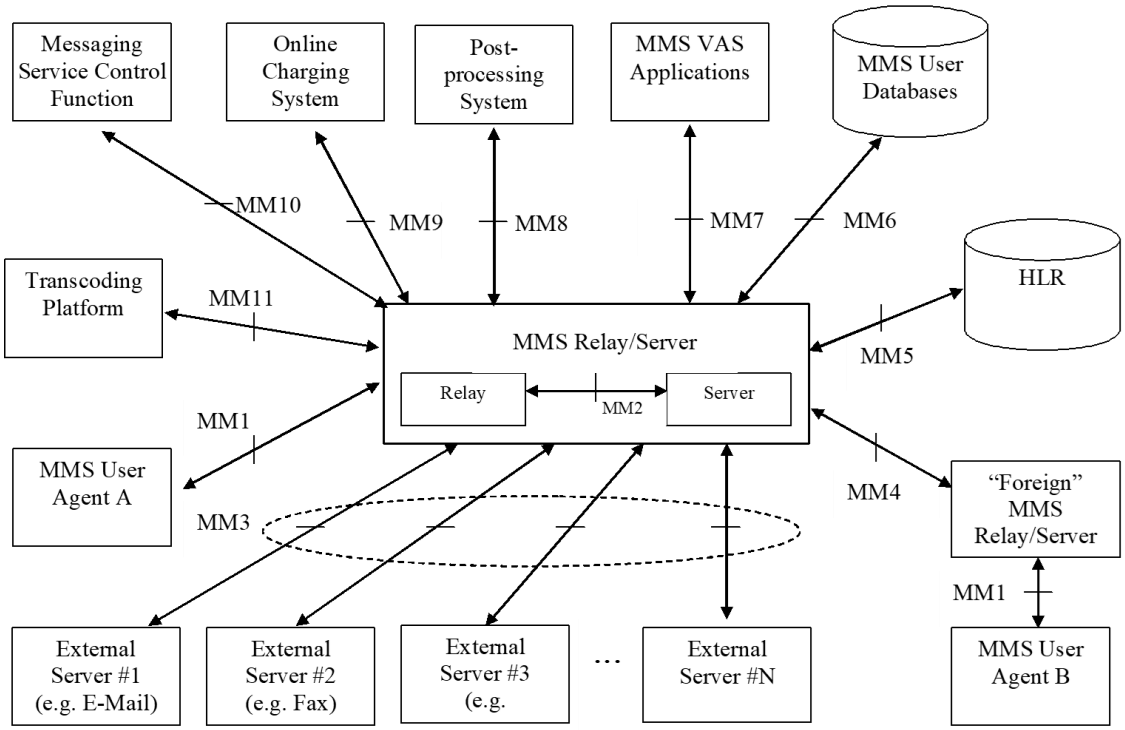
3.7 MMS centra

MMS centra (MMSC) jsou dalším klíčovým prvkem v architektuře mobilní sítě. Každé MMS centrum má dvě logické části: MMS Relay a MMS Server. MMS Relay přenáší zprávy mezi komponentami jedné sítě nebo do okolních sítí, kterými mohou být síť internet nebo síť dalšího mobilního operátora. MMS Server zprávy ukládá, zpracování a také informuje příjemce o zprávách nových. Když příjemce příjme oznámení o nové MMS zprávě, stahuje si její obsah právě z MMS Serveru. MMS Server může používat další specializované komponenty např. pro správu uživatelů nebo překódování zprávy. Popis dalších funkcí je uveden v kapitole 3.7.5. Obrázek 8 ukazuje další pohled na MMS Server/Relay a jejich propojení s ostatními částmi sítě. [15]



Obrázek 8 Architektura MMS sítě [15]

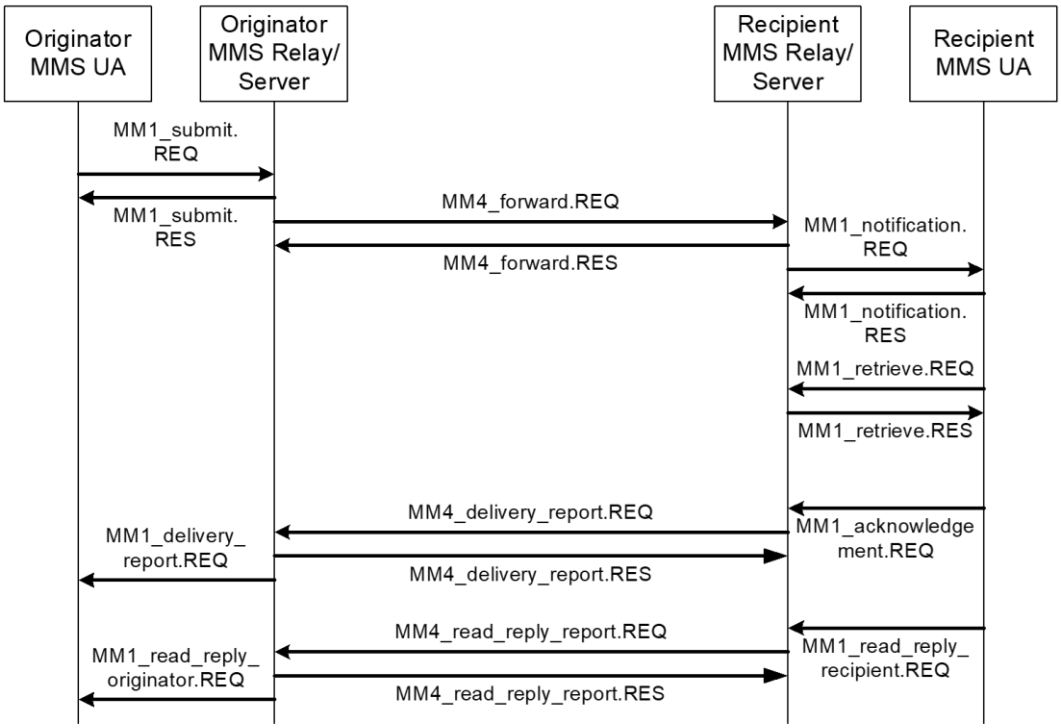
Ve specifikace 3GPP [14] je definováno jakým způsobem mohou MMS centra komunikovat s okolními systémy, komponentami nebo sítěmi. Komunikace probíhá pomocí rozhraní nazývaných MM, které mají další číselné označení, které rozděluje, k jakému jsou určeny účelu a jaké typy PDU používají. Tato práce se zabývá hlavně komunikací v rámci rozhraními MM1, které je určeno pro komunikaci mezi centrem a mobilním zařízením a rozhraním MM7, které slouží zejména pro komunikaci s externími systémy. [14, 15] Obrázek 9 ukazuje všechna rozhraní MM popsaná specifikací 3GPP a jejich konkrétní účely.



Obrázek 9 MM rozhraní MMS centra [14]

3.7.1 MM1

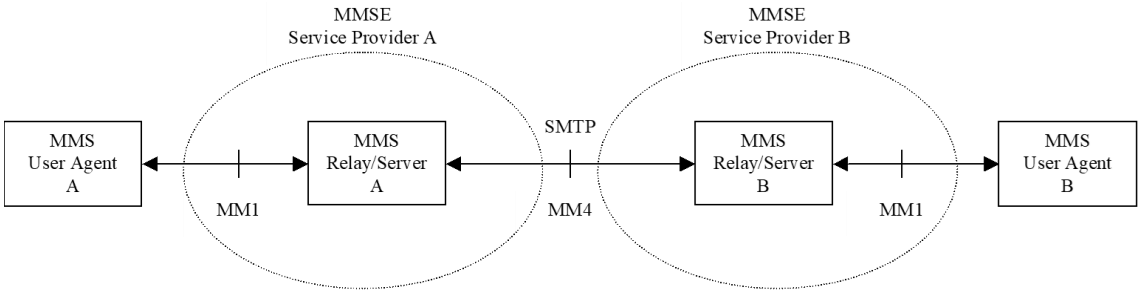
Rozhraní MM1 se používá pro přenos multimediálních zpráv mezi mobilním telefonem a MMS centrem. Může se jednat o odeslání nebo přijetí, ale také o zprávy o doručení nebo přečtení. Obrázek 10 pohází ze specifikace 3GPP a ukazuje komunikaci mezi zařízením uživatele a MMS centrem právě pomocí PDU balíčků rozhraní MM1. Odesílatel zprávy vytvoří PDU balíček MM1_submit.REQ a odešle jej mobilnímu centru. To pak informuje přímo příjemce o nové zprávě nebo přepošle zprávu jinému centru pomocí MM4. Příjemce si následně zprávu může stáhnout v rámci balíčku MM1_retrieve.RES a odesílatel je informován o jejím doručení a případně přečtení. [14]



Obrázek 10 Odeslání a přijetí MMS zprávy pomocí PDU z rozhraní MM1 [12]

3.7.2 MM4

Rozhraní MM4 slouží pro výměnu zpráv mezi dvěma MMS centry, které jsou buď ve stejné nebo různé mobilní síti. Toto rozhraní je použito zejména v případě, kdy příjemce používá jiného mobilního operátora než odesílatel a je tedy nutné přeposlat zprávu do jiné sítě, protože se o doručení zprávy stará jiné MMS centrum. Toto je další aspekt, ve kterém se liší SMS a MMS zprávy, protože SMS centrum odesílatele doručuje zprávu přímo příjemci. Obrázek 11 ukazuje schéma přenosu MMS zprávy z jednoho mobilního zařízení do MMS centra a následné přeposlání přes MM4 do MMSE příjemce a její doručení. Z obrázku je také patrné, že MM4 používá protokol SMTP. Před odesláním MM4 požadavku musí MMSC odesílatele nejdříve získat IP adresu MMS centra příjemce. Pro získání IP adresy použije MMSC odesílatele telefonní číslo příjemce a předdefinovanou tabulku, jež jej mapuje na doménové jméno cílového MMS centra. Následně je použita služba DNS, pro získání IP adresy. [14]



Obrázek 11 Schéma přenosu zpráv pomocí rozhraní MM4 [14]

3.7.3 MM7

Rozhraní MM7 je využíváno pro komunikaci mezi MMS centrem a externím systémem VASP. VASP (Value Added Service Provider) je externí systém nebo aplikace, která komunikuje přímo se MMS centrem a dokáže odeslat a přijmout MMS zprávy a následně obdržet zprávy o doručení nebo přečtení. VASP lze použít pro velké množství scénářů, kdy se může například jednat o odesílání zpráv většímu počtu příjemců zároveň nebo zpracování speciální zprávy přijaté z mobilního telefonu. [14]

Každý VASP systém je nutné přidat do konfigurace MMS centra. Tato konfigurace obsahuje jeho číslo (nebo rozsah čísel), na které mohou následně klienti služby odesílat MMS zprávy, které mají být systému doručeny. Při odeslání MMS zprávy VASP systémem se toto číslo zobrazí jako číslo odesílatele přijaté zprávy. Také je nutné nastavit jeho adresu v síti, na kterou budou doručovány požadavky od MMSC.

Připojení přes MM7

Komunikace mezi MMSC a VASP je bezstavová a využívá HTTP jako transportní protokol, konkrétně metodu POST. V případě, že je odesílán požadavek z VASP na MMS centrum, figuruje VASP jako klient a MMSC jako server, který vyřizuje jeho požadavky a vrací PDU s odpověďmi. Avšak pokud je jakýkoli požadavek odesílán opačným směrem, MMS centrum vystupuje jako klient. Tento fakt vyplývá z použitého typu komunikace mezi oběma prvky a stěžuje konfiguraci centra, protože to musí dopředu znát adresu VASP systému. Také musí být správně nakonfigurováno sítové připojení, aby bylo možné požadavky odesílat.

Podle specifikace [14] by mělo být MMS centrum zabezpečeno nějakou formou autentizace. V praxi se používá tzv. Basic access authentication, kde je v hlavičce

požadavku posíláno přihlašovací jméno a heslo jako hodnota atributu Authorization. Specifikace dále navrhuje použití autentizace pomocí certifikátu nebo zavedení šifrované komunikace pomocí SSL nebo TLS.

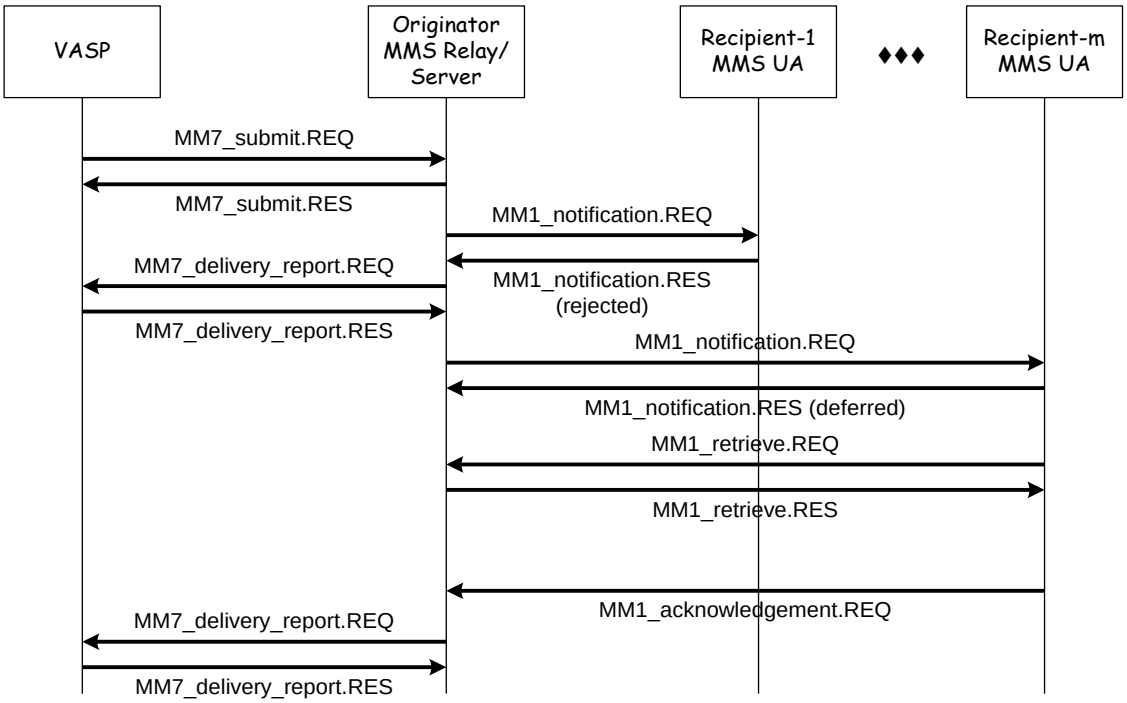
MMS centra také využívají autorizaci, kde je každému VASP systému předem nastaveno, jaké operace může po centru požadovat. Autorizace se provádí pomocí dvou atributů obsažených v požadavcích, konkrétně VASPID a VASID. Atribut VASPID identifikuje organizaci poskytující VASP a VASID identifikuje samotnou službu (resp. systém). Tento způsob označení je však pouze doporučením a záleží pouze na administrátorech MMSC jaké hodnoty pro identifikaci zvolí. [2]

Odeslání a přijetí zprávy přes MM7

Obsahem HTTP požadavků (PDU) jsou zprávy typu MIME, které obsahují SOAP obálku a zakódované přílohy. SOAP obálka je ve formátu XML a využívá tagy specifikované pro MM7. V obálce zprávy se nachází definice typu konkrétního PDU s požadavkem nebo odpovědí a jeho parametry. Mezi základní parametry patří používaná verze MM7, atributy VASPID a VASID a identifikátor zprávy. Dle typu PDU zde může být číslo odesílatele a příjemce, datum odeslání nebo informace o stavu vyřízení.

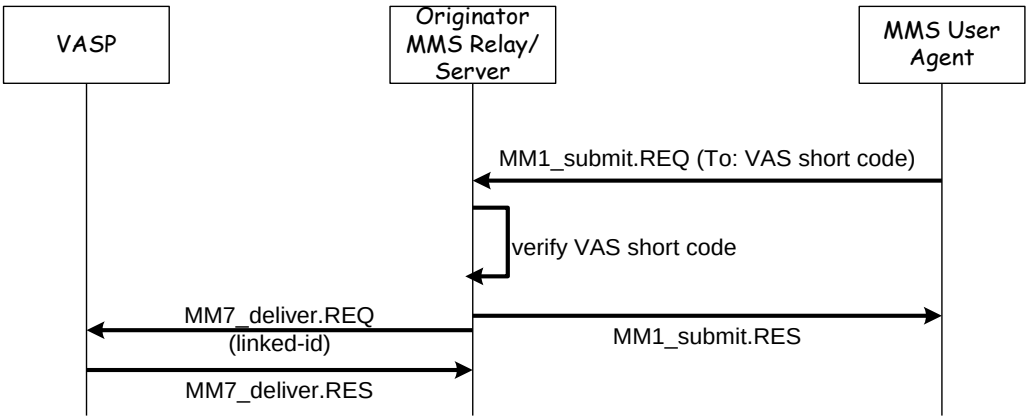
Pro odeslání nové zprávy VASP vytvoří PDU typu MM7_submit.REQ, připojí se k MMS centru pomocí HTTP a odešle PDU metodou POST. Centrum následně v odpovědi na požadavek vrací PDU typu MM7_submit.RES, které obsahuje informaci, zda byl požadavek vyřízen správně nebo z nějakého důvodu zamítnut. Následuje doručení zprávy všem příjemcům, kteří byli v požadavku uvedeni. VASP službě jsou také doručovány přijaté zprávy a zprávy o doručení nebo přečetní. Pokud je zpráva doručena v pořádku, je do VASP odesláno PDU typu MM7_delivery_report.REQ s informací o výsledku. Povinností VASP je potvrdit, že toto PDU v pořádku přijala a zpracovala.

Obrázek 12 ukazuje průběh a typy PDU balíčků, které jsou použity při odeslání MMS zprávy z VASP na mobilní zařízení. Lze si všimnout, že VASP komunikuje s MMS centrem pomocí MM7, ale to dále komunikuje s telefonem pomocí MM1. Je zde také zobrazen scénář, kdy je zpráva odesílána více příjemcům současně a jeden z příjemců zprávu odmítne a ostatní ji přijmou. [14]



Obrázek 12 Odeslání MMS zprávy přes MM7 [14]

Obrázek 13 ukazuje průběh komunikace při přijetí zprávy od mobilního zařízení. Mobilní zařízení opět využívá MM1, přes které je zpráva odeslána MMS centru. Centrum podle adresy příjemce vyhodnotí, kterému VASP systému se má zpráva doručit a odešle mu požadavek (PDU) typu MM7_deliver.REQ. Požadavek informuje o doručení nové zprávy a zároveň obsahuje všechny její parametry a přílohy. Při úspěšném zpracování musí VASP odeslat odpověď typu MM7_deliver.RES. Pokud by služba žádnou odpověď neodeslala nebo požadavek skončil s chybou, může se stát, že by MMS centrum zkoušelo zprávu opakovaně doručit, což by mohlo způsobit jeho zacyklení nebo dokonce dočasnou nedostupnost. [14]



Obrázek 13 Přijetí MMS zprávy přes MM7 [14]

PDU, která slouží jako požadavky pro odeslání nebo přijetí zprávy mohou obsahovat přílohy. Tyto přílohy nejčastěji tvoří textová zpráva nebo např. obrázek. Speciálním typem přílohy je SMIL, kde jsou definovány typy všech ostatních příloh a jejich další nastavení, které slouží zejména pro informování zařízení příjemce, jak má zprávu správně zobrazit. [14]. Tyto přílohy jsou kódovány dle specifikace [16] a nejčastěji se jedná o formáty 7bit, 8bit, binary nebo Base64.

3.7.4 Reprezentace adres

Při používání rozhraní MM1 se používají celkem tři adresy. První adresou je adresa samotného MMS Relay/Server, která by měla být ve formátu URI a musí být nastavena přímo v mobilním zařízení uživatele. Druhou adresou je adresa odesílatele zprávy, která může být číslo (E.164 MSISDN) nebo adresa dle standardu RFC 2822 (emailová adresa). Třetí je adresa příjemce, která může být číslo, krátký kód nebo adresa dle standardu RFC 2822. Odesílat zprávy musí vždy před odesláním zajistit, že je adresa správně naformátovaná.

Rozhraní MM7 používá opět tři adresy, a to adresu MMS centra, odesílatele a příjemce. MMS centrum musí podporovat všechny tři možné formáty adres a také musí být schopno přeložit adresu VASP (číslo) na její URL, aby bylo možné zprávu doručit. [14]

3.7.5 Další vlastnosti mobilní center

MMS centra mohou obsahovat další specializované komponenty a součásti, jejichž cílem je zlepšit uživatelskou přívětivost celé služby, a hlavně nabídnout zákazníkům další rozšířené možnosti a funkce. Následující kapitoly se věnují popisu některých doplňkových služeb a komponent, které mohou být součástí MMS centra.

Transformace obsahu MMS zprávy

Při odesílání multimediálního obsahu z jednoho telefonu na druhý může dojít k situaci, že zařízení příjemce není schopné zobrazit přijatý obsah. Nejčastěji se jedná o obrázky nebo videa, která mohou mít různé formáty a vlastnosti (parametry). K zajištění větší uživatelské přívětivosti a zamezení rizika špatného zobrazení může MMS centrum obsah zprávy různě transformovat. Jedná se například o změnu formátu obrázku nebo videa, případně o úpravu rozlišení nebo velikosti. Tyto operace provádí právě MMS centrum, protože je při odesílání MMS zprávy centrálním prvkem architektury a zároveň má informaci o telefonu odesílatele i příjemce. Když příjemce odešle požadavek na stažení obsahu zprávy je součástí požadavku také tzv. User Agent Profile (UAPROF), který popisuje schopnosti zařízení ve formátu RDF (Resource Description Framework). Jelikož ale může být RDF poměrně rozsáhlé, odesílá se pouze URL této definice. Když následně MMS centrum zpracovává tento požadavek na stažení zprávy, načte si její obsah a zároveň si stáhne RDF schéma. Tak lze zjistit vlastnosti a schopnosti zařízení příjemce a případně provést požadovanou transformaci. Transformace může být náročná na výpočetní výkon a paměť, proto je vhodnější ji provádět na jiném serveru, a dokonce lze použít i specializovaný hardware, který je více přizpůsobený pro tyto účely. [17]

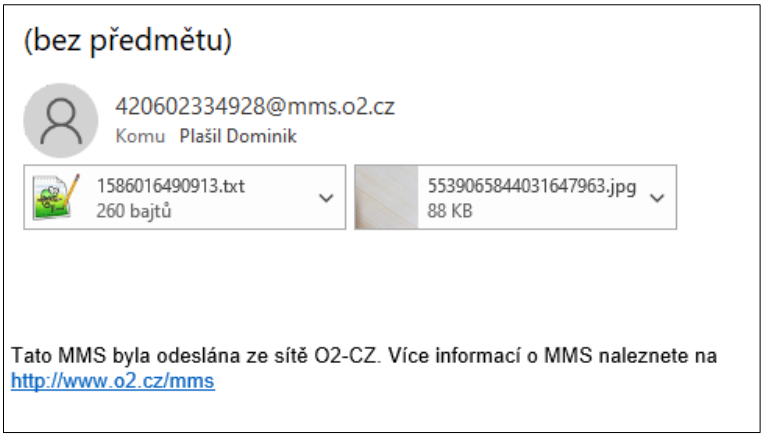
MMS alba

MMS album je specializovaná komponenta připojená k MMS Serveru rozšiřující jeho možnosti. Pokud MMS zpráva nemůže být příjemci doručena nebo si ji nestáhne před vypršením platnosti, může ji mobilní operátor uložit do tzv. MMS alba. Příjemci zprávy je následně doručena SMS zpráva s instrukcemi, jak si tuto zprávu zobrazit na webové stránce. SMS zpráva také často obsahuje heslo, pro přístup do alba

příjemce. [18] Administrátoři MMS center mají možnost nastavit, že se na některá mobilní zařízení nebo dokonce mobilní čísla nebudou MMS zprávy doručovat vůbec a budou rovnou umístěny do MMS alba.

Odeslání MMS na email

Odeslání MMS na email je speciálním typem doručení zprávy, kdy odesílatel vyplní emailovou adresu místo čísla příjemce. MMS centrum následně musí správně vyhodnotit, že se jedná o email a odeslat jej na patřičnou adresu. Specifikace [14] uvádí, jak se transformují balíčky MM1_submit.REQ a MM7_forward.REQ, které jsou určeny pro doručení na email. Je nutné podotknout, že některé atributy jsou při této transformaci vynechány, protože buď nedávají v kontextu emailu smysl nebo nejsou podporovány. [14] Mobilní operátor může následně definovat, jak bude vypadat doručená emailová zpráva. Obrázek 14 ukazuje, jak vypadá MMS zpráva doručená na email mobilním operátorem O2. Je zde patrné, že telefonní číslo odesílatele je uvedeno jako emailová adresa a text a obrázek zprávy jsou doručeny jako příloha.



Obrázek 14 MMS zpráva doručená na email

4 Automatizované testování

Technologie spojené s mobilními sítěmi a SMS a MMS centry jsou již dlouho dobu jasně standardizované a předpokládá se, že je všechny aplikace implementují správně. Nicméně software fungující jako SMSC a MMSC může být poměrně různorodý a nabízet další funkce přidávající hodnotu celé službě. Jako každý jiný software i tato centra je nutné pravidelně aktualizovat a testovat. Při testování je nutné věnovat pozornost nejen základním funkcionalitám, ale také různým zranitelnostem a například i spolehlivosti doručování zpráv nebo zátěžovým testům. Klíčová je správná reakce center na jednotlivé parametry zpráv. [19] Jako příklad mohou posloužit zprávy (potvrzení) o doručení nebo přečtení. V parametru MMS zprávy lze uvést, zdali je toto potvrzení vyžadováno nebo povoleno a MMS centrum se pak musí zachovat přesně tak, jak je v parametru uvedeno. Za zmínku také stojí parametr o zobrazení nebo skrytí odesílatele zprávy nebo čas nejdřívější doručení zprávy. Všechny tyto možnosti vyžadují rozšířenou režii od MMS centra a je nutné otestovat všechny tyto varianty před nasazením nové verze do reálného produkčního prostředí. [14, 19]

Administrátoři tedy potřebují spolehlivý systém pro testování všech funkcí SMS a MMS center. Existuje základní dokumentace a doporučení [19], které části a funkce systému by měly být otestovány před uvedením do produkčního provozu. V této dokumentaci je mimo jiné uvedeno vhodné prostředí pro testování, doporučené typy testů a parametry jednotlivých zpráv. V základu lze uvažovat tři typy scénářů. Prvním je odeslání zprávy z jednoho mobilního zařízení na druhé. Dalšími dvěma jsou odeslání zprávy ze SMS/MMS centra na mobilní zařízení a naopak. Nabízí se i možnost odesílat zprávu z jednoho centra na druhé, avšak to se v praxi téměř nevyužívá.

4.1 Rizika pro mobilní centra

Jedním z důvodů, proč testovat SMS a MMS centra může být bezpečnost celého systému. Článek [20] se zabývá bezpečností koncových zařízení v mobilní síti. Uživatelé mobilních zařízení mohou používat různé aplikace umožňující odesílání a přijímání MMS zpráv. Tím může vzniknout potenciální zranitelnost jak na straně

klienta, tak pro MMS centrum. MMS zprávy mohou obsahovat různé multimediální soubory, ale také binární i spustitelné soubory nebo skripty. MMS centrum pak může tyto soubory analyzovat nebo dokonce transformovat. To je však potenciální bezpečnostní riziko, protože pokud odesílatel odešle soubor nakažený virem, může se tento vir dostat i na server mobilního operátor. Článek také upozorňuje na fakt, že uživatel většinou neví, jak aplikace funguje a nemusí si být ani vědom, že odesílá nějaké zprávy na pozadí. Jedním z dalších rizik může být útok typu DDoS. Útočník může využít nainstalovanou aplikaci na koncových zařízeních uživatelů a vytvořit z nich aktivní prvky při svém útoku.

Kapitola v knize [21] se naopak zabývá zranitelností spojenou se SMS zprávami a SMS centry. Opět je zde diskutován problém DDoS útoků. SMSC centra používají metodu „store-and-forward“, což může způsobit jistou limitaci v počtu vyřízených požadavků, protože kapacita paměti SMSC není nekonečná. Autoři provedli test, kde byl jeden telefon vypnutý a druhý na něj odeslal řádově stovky SMS zpráv. Protože je SMS centrum nemohlo doručit, muselo je držet v paměti až do opětovné dostupnosti zařízení příjemce. Po zapnutí druhého telefonu porovnávali jednotlivá doručení na různých softwarech použitých pro SMSC. Systém AT&T doručil maximálně 400 zpráv v pořadí, ve kterém byly odeslány. Naopak systém Verizon ukládal pouze 100 zpráv a následně tedy doručil pouze posledních sto, které byly odeslány a ty předchozí ignoroval, resp. smazal. [21]

4.2 Rešerše existujících řešení

Jak je popsáno v předchozích kapitolách, SMS a MMS centra mohou tvořit poměrně rozsáhlé systémy a služby, které musí být spolehlivé a správně vyřizovat požadavky uživatelů a zároveň správně reagovat na nastavené parametry zpráv. Manuální testování administrátory center může být náročné, zdlouhavé a může se stát, že bude nějaká chyby přehlédnuta kvůli lidskému faktoru. Je tedy lepší provádět automatizované testování počítačem z důvodu větší rychlost, přesnosti a spolehlivosti. Následující kapitoly se věnují rešerši některých existujících řešení, které umožňují takovéto testování provádět.

4.2.1 TestMySMS

Jedním z možných řešení pro testování je služba TestMySMS, která se zabývá testováním spolehlivosti doručení zpráv. [22] Uživatel si vytvoří předpřipravené scénáře se SMS zprávami, které jsou následně odeslány na koncová zařízení služby pomocí mobilního operátora ve zvolené zemi. Pokud je zpráva na koncové zařízení doručena, služba informuje uživatele. TestMySMS umožňuje odesílat SMS zprávy složené z více částí (Concatenated SMS) a také podporuje integraci přes SMPP nebo REST API. Uživatel může dokonce použít své vlastní zařízení jako telefon příjemce a je připravována možnost odeslat zprávu zpět, aby se otestovaly oba směry doručení.

4.2.2 SIGOS

Komerční služba SIGOS [23] funguje na podobném principu jako TestMySMS, nicméně nabízí další funkce a rozšíření. Je zde podpora pro SMS a MMS testy, stejně jako služby VASP. Navíc lze naprosto odděleně testovat dostupnost mobilní sítě, IoT zařízení nebo například telefonování a tísňová volání. Uživatel může použít předem připravené scénáře nebo si vytvořit vlastní. Následně vše přehledně vidí na jedné stránce včetně různých grafů a lze si zapnout i notifikace pro případ chyby. Z výsledků testů je možné vytvořit přehledný protokol shrnující všechny testy.

4.2.3 NowSMS

Při testování se lze také zaměřit na samotné externí aplikace, které se SMS nebo MMS centrem přímo komunikují, protože i ty mohou obsahovat různé zranitelnosti a mohou tak případně negativně ovlivnit chod produkčních systémů. Jednou z možností je použít oddělená testovací centra nebo dokonce centrum virtuální. K tomuto účelu lze použít službu NowSMS [13], která poskytuje software pro plnohodnotné SMS a MMS centrum, avšak lze ji použít i jako náhražku skutečného centra. Takto lze emulovat celý systém, aniž by vznikala nějaká rizika. Systém umožňuje komunikaci pomocí všech rozhraní pro MMS zprávy (definovaných standardizací 3GPP a OMA) a také SMPP. Systém tedy může sloužit jako brána do dalších systémů nebo pouze jako emulátor, který přijatou zprávu předá do jiného výstupu.

4.3 Shrnutí existujících řešení

Protože se jedná o poměrně úzkou problematiku, kterou řeší pouze mobilní operátoři, existuje jen omezené množství dostupných řešení. Často se jedná o formu nějaké služby, která odesílá předdefinované zprávy a kontroluje jejich doručení. Uživatel také musí platit za každou odeslanou zprávu nebo paušální poplatek. [22, 23] Tato řešení lze použít pro základní analýzy a testování, avšak často není možné vytvořit naprosto libovolné scénáře, které by umožňovaly testovat všechny parametry PDU balíčků nebo provádět nějaké komplexnější úkony či validace.

5 Vývoj systému pro end-to-end testování SMS a MMS center

Jedním z cílů této práce bylo vytvořit a popsat systém pro automatizované testování SMS a MMS center. Projekt byl zadán společností O2 Czech Republic a.s., která plánuje výsledné řešení používat pro testování vlastních center a zdokonalování poskytovaných služeb. Tomuto vývoji se budou věnovat zbývající kapitoly.

5.1 Zadání a seznam požadavků

Zadáním bylo vytvořit systém pro end-to-end testování SMS a MMS center. Testování bude probíhat pomocí předdefinovaných testovacích scénářů, které bude možné spouštět opakovaně. Testovací scénáře musí být schopny odeslat SMS a MMS zprávu z fyzického mobilního zařízení s operačním systémem Android nebo pomocí SMS a MMS centra. Následně musí systém umět odeslanou zprávu přijmout a zpracovat. Důležitá je validace výsledků, zdali je doručená zpráva totožná s odeslanou, případně jestli se shodují vybrané parametry zprávy. Je nutné také monitorovat průběh celého testovacího scénáře a například kontrolovat, že se dokončí v daném časovém limitu.

5.1.1 Příklady testovacích scénářů

Od zadavatele projektu bylo také upřesněno několik základních testovacích scénářů, které by měl být systém schopen vytvořit a provést. Základním kritériem byly směry komunikace v jednotlivých scénářích. Musí být možné odeslat zprávu mezi dvěma telefony, mezi SMS nebo MMS centrem a telefonem a naopak. Systém musí umět odeslat SMS zprávy s různým obsahem, konkrétně se jedná o krátkou nebo dlouhou SMS zprávu s různými typy kódování. Jedná se o zprávy bez diakritiky s kódováním 7-bit a délkou až 320 znaků nebo zprávy s diakritikou nebo speciálními znaky a délkou až 140 znaků. Pro každou zprávu může být vyžadováno potvrzení o doručení. Při odesílání zprávy ze SMS centra musí být možné nastavit všechny dodatečné parametry, které specifikace SMPP obsahuje, včetně zprávy v binárním formátu nebo další TLV parametry. U MMS zpráv musí být možné nastavit různé parametry, které PDU odesílané MMS zprávy obsahuje. MMS zprávy mohou

obsahovat různý multimediální obsah jako je obrázek, zvuk, video nebo např. vizitka. V případě odesílání ze MMS centra musí být možné nastavit všechny dodatečné parametry.

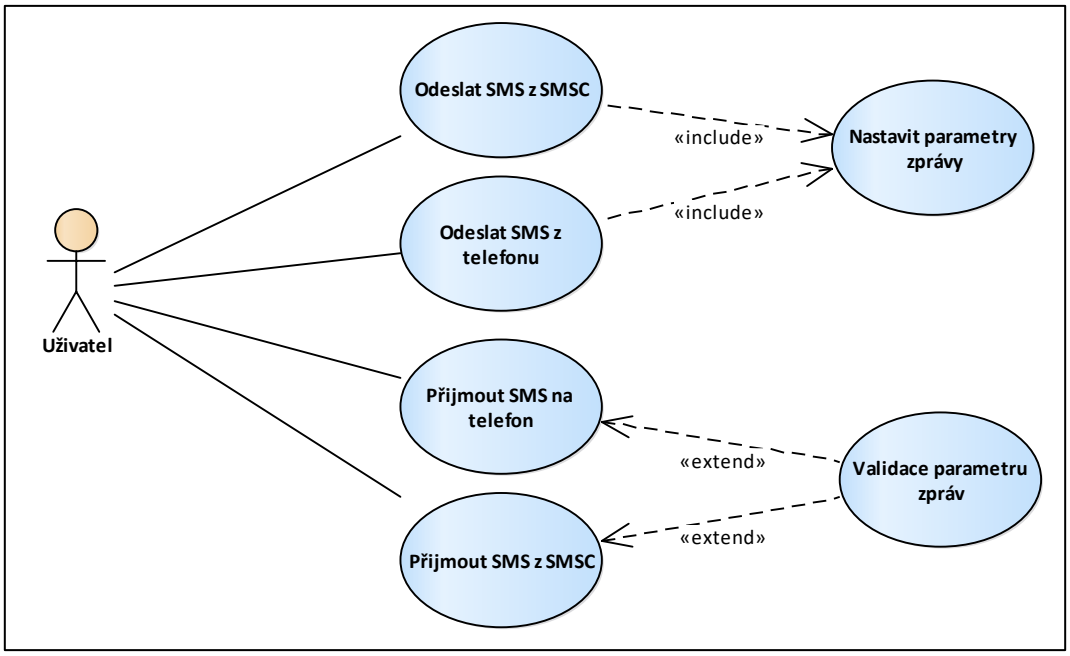
5.2 Analýza požadavků

V kapitole 4.2 bylo popsáno několik již existujících řešení, avšak žádné plně nesplňuje všechny požadavky od zadavatele. Hlavním nedostatkem je omezená možnost nastavování různých parametrů zpráv a také komunikace se SMS a MMS centry, které musí být dobře zabezpečené a není vhodné k nim umožňovat přístup aplikacím z internetu.

Po prostudování zadání a konzultaci se zadavatelem projektu bylo rozhodnuto, že nejlepší možností bude vytvořit vlastní systém, kde si uživatel bude moci vlastní testovací scénáře naprogramovat s pomocí předpřipravené knihovny. Jinou uvažovanou variantou bylo vytvořit systém s uživatelským rozhraním, kde by si uživatel scénář pouze „naklikal“ a následně by se provedl jako celek. Od tohoto řešení bylo upuštěno z důvodu větší komplexnosti řešení a možnému snížení flexibility a možností při vytváření testovacích scénářů. Nově vytvořený systém se bude jmenovat MMS Params a bude založený na již existující mobilní aplikaci vytvořené v rámci bakalářské práce [1].

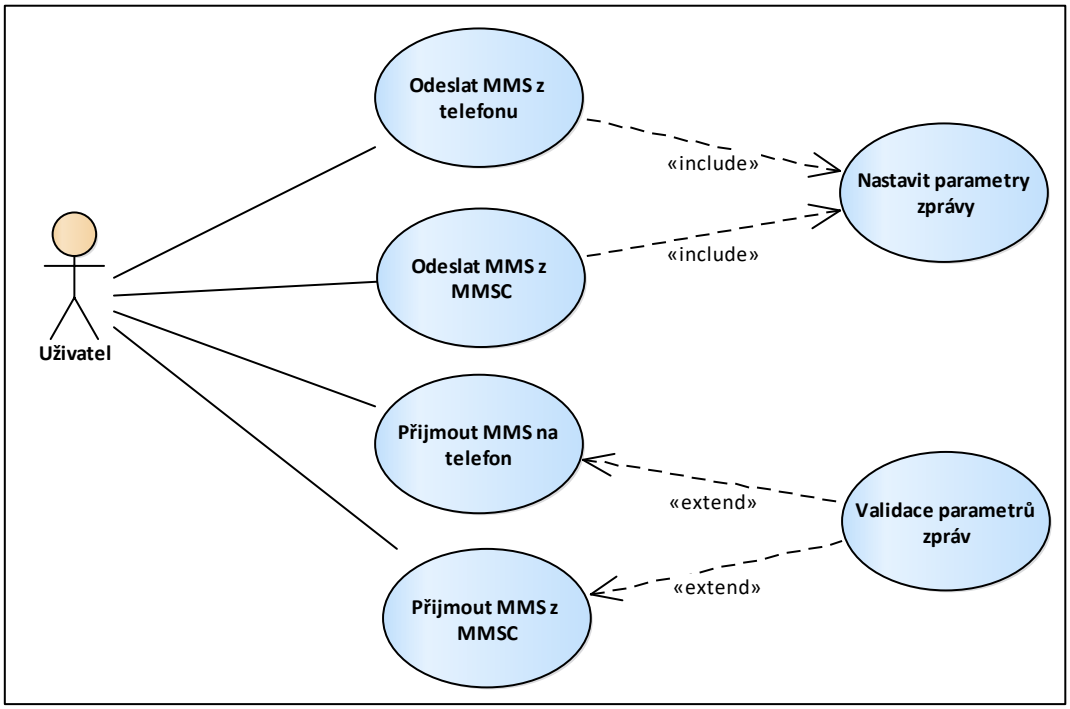
5.2.1 Diagram případů užití

Z požadavků od zadavatele a příkladů testovacích scénářů byly následně vytvořeny diagramy případů užití. Diagramy obsahují přehled nejdůležitější funkcí, které může uživatel po systému požadovat a systém je musí být schopný realizovat. Obrázek 15 je diagramem případů užití zaměřený na funkce spojené se SMS zprávami. Je zde modelováno odeslání i přijetí zprávy, a to buď přes mobilní telefon nebo SMS centrum. U každé odesílané zprávy je možné nastavit všechny její parametry a přijaté zprávy je možné validovat, zdali jejich parametry odpovídají parametrům předpokládaným.



Obrázek 15 Use Case diagram testování SMS zpráv

Následující Obrázek 16 obsahuje případy užití pro MMS zprávy. Obdobně jako u SMS zpráv je zde odeslání i přijetí, a to buď přes mobilní telefon nebo pomocí MMS centra.



Obrázek 16 Use Case diagram testování MMS zpráv

Systém musí být také schopný zobrazit uživateli výsledky jednotlivých testovacích scénářů. Důležité je, jestli byl testovací scénář splněn úspěšně nebo zdali skončil chybou. V případě chyby je nutné rozlišit, zda byla způsobena během validací parametrů zpráv, chybou v testovacím scénáři nebo komunikací mezi komponentami systému, případně jestli nenastala nějaká jiná neočekávaná chyba.

5.3 Návrh realizace systému

Z analýzy vyplynulo, že je nutné mít několik oddělených komponent systému, které mezi sebou budou komunikovat a společně realizovat požadavky zadané v testovacím scénáři. Klíčovou částí systému bude serverová aplikace, která bude fungovat jako centrální prvek mezi testovacím scénářem vytvořeným uživatelem a mobilní aplikací odesílající a přijímající zprávy. Server bude také zajišťovat připojení s SMS a MMS centrům a umožňovat tak odesílání i příjem zpráv. Server bude mít přehled o všech probíhajících testovacích scénářích a až bude scénář ukončen, uloží o něm veškeré informace pro pozdější zobrazení a vyhodnocení uživatelem. Toto zobrazení bude realizováno pomocí webové aplikace v prohlížeči uživatele.

Uživatel bude testovací scénáře programovat v běžném programovacím jazyce. K tomuto účelu je nutné vytvořit knihovnu, která bude obsahovat konkrétní funkce a příkazy pro vytváření testovacích scénářů. Tato knihovna bude implementovat připojení k serverové aplikaci a následně bude umožňovat odesílat příkazy do mobilní aplikace a také serveru (např. pro připojení a odeslání zprávy do SMS nebo MMS centra).

Knihovna bude dále obsahovat podpůrné funkce, umožňující snadné provádění validací nebo získání informací o průběhu testu. Uživatel si tedy pomocí příkazů vytvoří vlastní kroky testovacího scénáře a následné validace výsledků sám a podle vlastní potřeby. Při návrhu funkcí knihovny musí být dbáno zejména na její obecnost, tedy že bude možné vytvořit co nejširší spektrum testovacích scénářů s různými parametry, aniž by bylo nutné přidávat větší množství vlastních funkcí.

5.4 Aplikace MMS Params

Základem mobilní aplikace bude již existující aplikace MMS Params vytvořená v rámci bakalářské práce [1]. Existující mobilní aplikace umožňuje odesílat a přijímat SMS a MMS zprávy a nastavovat jim téměř všechny parametry dle specifikace 3GPP. Aplikace pracuje přímo s PDU balíčky, což umožňuje řídit celý průběh při odesílání i zpracovávání přijatých MMS zpráv.

Použitá aplikace dále umožňuje vytvoření tzv. profilů, které obsahují předdefinované parametry odesílaných zpráv. Tyto profily lze následně přepínat podle potřeby a odesílat tak MMS zprávy s různým nastavením. Aplikace dále umožňuje předdefinovat MMS šablony, které lze odesílat opakovaně. Jednou z předností je možnost zobrazit PDU balíčky MMS zpráv. To uživateli umožňuje analyzovat hodnoty jejich parametrů a zjistit tak, zdali odpovídají předpokladům a mobilní centra se chovají správně.

Tyto již implementované funkcionality poslouží jako dobrý základ pro novou aplikaci. Jediný rozdíl bude v možnosti využívat tyto funkce vzdáleně z počítače, konkrétně uživatelem naprogramovaného testovacího scénáře. Uživatel v rámci vytváření testovacího scénáře nastaví všechny požadované parametry odesílaných zpráv a knihovna pouze vyšle příkaz do mobilní aplikace, která odešle nebo přijme SMS nebo MMS zprávu. Po této operaci pošle všechny informace zpátky klientovi, resp. naprogramovanému testovacímu scénáři. Cílem je tedy vykonávat testovací scénáře, aniž by uživatel musel vzít mobilní telefon do ruky.

5.5 Programovací jazyk a použité technologie

Od zadavatele projektu bylo také zadáno, že by celý systém měl být vytvořen pouze s použitím open source nebo freeware technologií. Byl tedy vybrán programovací jazyk Java, který bude použit pro všechny komponenty systému, kromě webové části, která bude vytvořena pomocí frameworku Angular. Mobilní aplikace je vytvořena pro operační systém Android. Systém může využívat jakoukoli databázi, ideálně MySQL nebo MariaDB. Předpokládá se, že serverová aplikace bude hostována na serveru s operačním systémem Linux u zadavatele projektu. Zadavatel

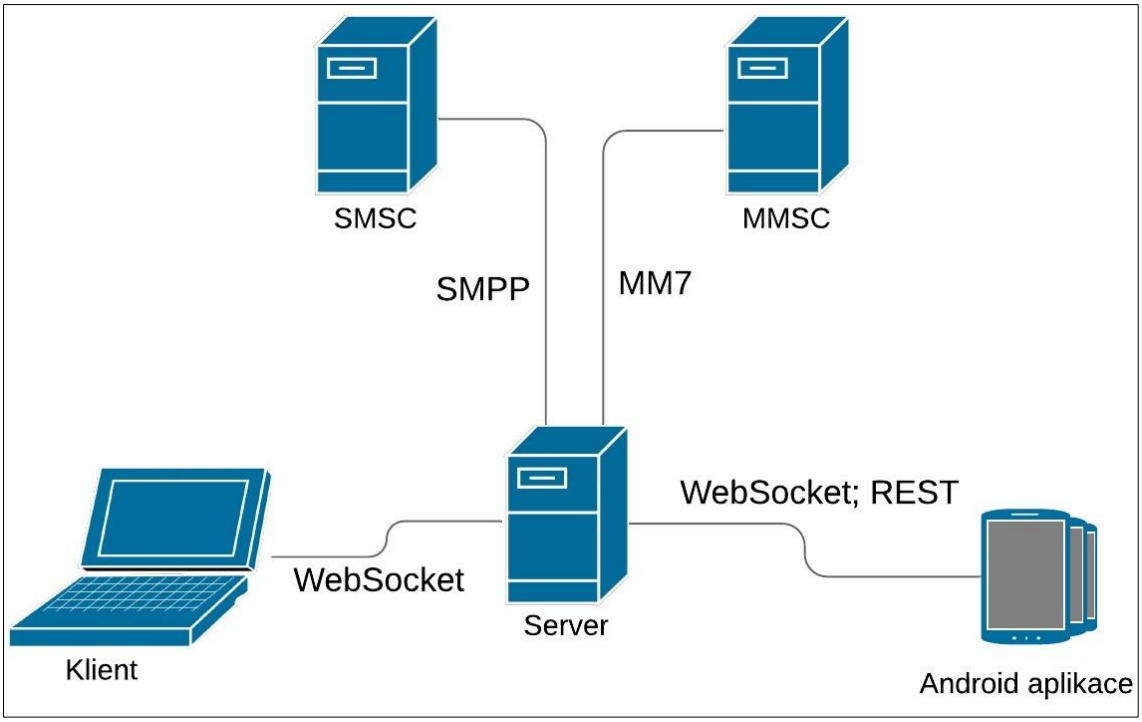
si sám zajistí veškerou konfiguraci prostředí pro aplikaci včetně nastavení serverů, firewallu a přístupů do ostatních částí sítě.

5.6 Komunikace mezi komponentami systému

Jelikož bude testovací scénář spouštěn jako běžný program, je nutné, aby komunikace s ostatními komponentami systému probíhala bez jakýchkoli zpoždění a co možná nejplynuleji. Při analýze byly uvažovány dva způsoby komunikace mezi jednotlivými moduly. Prvním byla komunikace pomocí REST protokolu. Jednotlivé moduly by se serverem komunikovaly pomocí GET a POST požadavků a ten by následně prováděl požadovanou činnost. Druhým uvažovaným způsobem byla komunikace přes WebSocket připojení. Uživatelský testovací scénář a mobilní telefony by se připojily k serveru a udržovaly spojení tak dlouho, jak by bylo potřeba.

Pro samotnou komunikaci byla vybrána druhá možnost, protože WebSocket umožňuje okamžité odeslání a zpracování informací mezi všemi komponentami a nevzniká žádná zbytečná prodleva nebo zatěžování sítě kvůli opakovaným požadavkům na zjištění stavu. V případě jakékoliv chyby při zpracování je možné okamžitě informovat klienta a testovací scénář ukončit jako neúspěšný.

Nicméně ani první možnost nezůstane stranou a komunikace přes REST bude implementována pro načítání dat ze serveru pro webového rozhraní. Tato část však bude sloužit pouze pro zobrazování informací a nebude možné testovací scénáře vytvářet nebo ovlivňovat. Obrázek 17 ukazuje navržený způsob komunikace mezi jednotlivými komponentami systému. Je zde také zobrazena komunikace se SMS a MMS centrem, která musí být realizována pomocí protokolu SMPP a rozhraní MM7.



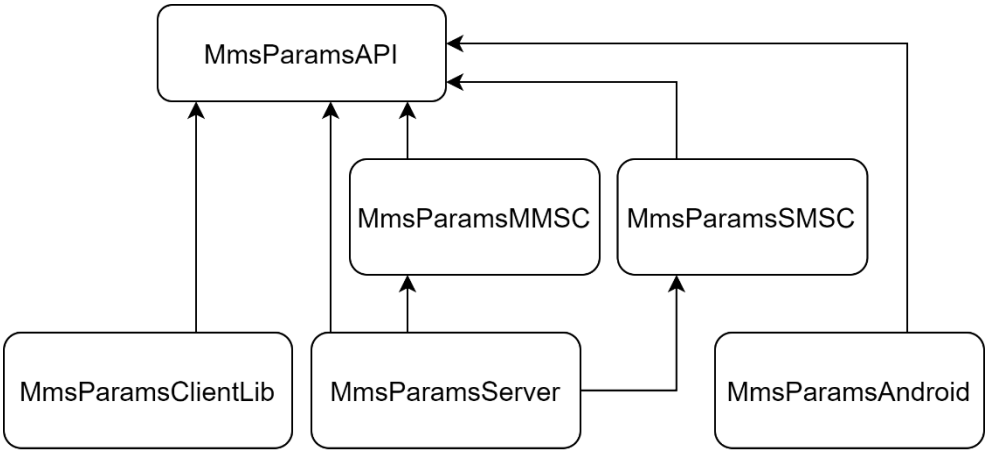
Obrázek 17 Komunikace mezi komponentami systému

6 Vývoj systému

Z analýzy požadavků vyplynulo, že je potřeba vytvořit systém, který bude mít tři hlavní komponenty: serverovou aplikaci, knihovnu pro psaní testovacích scénářů uživatelem a mobilní aplikaci pro telefony s operačním systémem Android. Následující kapitoly se věnují samotnému vývoji systému, konkrétně popisu vytvořených modulů aplikace, komunikace mezi nimi a následnému popisu funkcí jednotlivých komponent při vytváření a spouštění testovacích scénářů.

6.1 Moduly aplikace

Z pohledu Javy tvoří systém několik modulů, kde každý z nich má určitou funkci. Obrázek 18 ukazuje navržené moduly aplikace a jejich vzájemné závislosti, které jsou detailněji popsány v následujících kapitolách.



Obrázek 18 Diagram modulů systému

6.1.1 MmsParamsAPI

Společný modul pro všechny další moduly se nazývá MmsParamsAPI. V tomto modulu jsou třídy, které jsou sdílené pro všechny ostatní moduly a poskytují jim společné funkce. Modul obsahuje zejména definice objektů zpráv pro komunikaci mezi komponentami systému a také třídy, které představují obsahy jednotlivých zpráv. Jsou zde i třídy pro podporu rozpoznávání příchozích zpráv a jejich serializaci a deserializaci. Modul také obsahuje řadu konstant a typů enum spolu s definicemi

základních výjimek (Exception) systému. Dále obsahuje třídy pro autentizaci, HTTP požadavky, validace nebo logování.
Funkce tohoto modulu je klíčová pro správné fungování celého systému, protože je nezbytné, aby všechny komunikující komponenty pracovaly s totožnými definicemi objektů, které jsou přenášeny. Sekundárním účelem je sdílení různých funkcí, takže jednotlivé moduly nemusí mít svoji vlastní implementaci, čím by vznikal duplicitní kód a také by bylo těžší udržovat všechny funkce totožné nebo kompatibilní.

6.1.2 MmsParamsClientLib

Modul MmsParamsClientLib je určen přímo pro uživatele vytvářejícího testovací scénáře a poskytuje všechna potřebná rozhraní pro vytváření konkrétních kroků, resp. příkazů. Modul obsahuje implementaci připojení k serveru a zajišťuje veškerou komunikaci s ním. Uživatel tedy jednoduše volá metody dané rozhraním systému a modul se postará o správné vytvoření a odeslání požadavků na server. Modul navíc obsahuje další podpůrné funkce pro vytváření testovacích scénářů jako jsou validace, načítání souborů pro přílohy MMS zpráv nebo odchytávání výjimek. Průběh vykonávání scénáře je detailně logován a uživatel může následně dohledat, kde například vznikla chyba.

6.1.3 MmsParamsSMSC

Modul MmsParamsSMSC zajišťuje komunikaci se SMS centrem pomocí protokolu SMPP. Modul funguje jako návrhový vzor adaptér nad knihovnou třetí strany, která implementuje protokol SMPP. Poskytuje tak serveru jednoduché rozhraní, které umožňuje připojení a odpojení od SMSC a také odesílání a přijímání SMS zpráv. Při přijetí SMS zprávy informuje o této skutečnosti modul serveru a ten pak rozhoduje, co se má s přijatou zprávou stát.

6.1.4 MmsParamsMMSC

Modul MmsParamsMMSC obsahuje funkcionalitu pro komunikaci s MMS centrem pomocí rozhraní MM7. Modul umožňuje odeslat MMS zprávy na MMSC a také implementuje funkce pro rozpoznání zpráv příchozích. Přijatá zpráva je opět

předána serveru, který ji přepošle do daného testovacího scénáře. Tento modul také využívá knihovnu třetí strany, která obsahuje kompletní implementaci MM7.

6.1.5 MmsParamsServer

Modul MmsParamsServer tvoří základ celého systému a je vytvořen pomocí frameworku Spring Boot. K serveru se připojují telefony s mobilní aplikací a uživatelské testovací scénáře, které mu odesílají příkazy, na které reaguje. Přes server probíhá komunikace mezi vytvořeným testovacím scénářem a mobilní aplikací. Server také zajišťuje připojení a komunikaci se SMS a MMS centry pomocí již popsaných modulů MmsParamsSMSC a MmsParamsMMSC.

Server má tedy přehled o všech připojených zařízeních a o všech spuštěných i ukončených testovacích scénářích. U každého testovacího scénáře eviduje informace o klientovi, použitých telefonech, připojeních k SMSC nebo MMSC a také o všech odeslaných nebo přijatých příkazech vzniklých v průběhu celého testovacího scénáře. Po ukončení testovacího scénáře ukládá všechny tyto informace do databáze.

Webové rozhraní serveru

Serverová aplikace obsahuje grafické uživatelské rozhraní v podobě webové aplikace vytvořené pomocí frameworku Angular. Webová aplikace umožňuje zobrazit informaci o stavu serveru a také o připojených mobilních telefonech nebo klientech vykonávajících testovací scénáře. Webové rozhraní umožňuje uživateli zobrazit seznam všech aktivních i ukončených testovacích scénářů a u každého scénáře je možné zobrazit jeho detail. V detailu scénáře jsou nejen informace o názvu a popisu, ale také o všech použitých mobilních telefonech. Uživatel si může prohlédnout seznam všech validací a jejich výsledků a v případě chyby analyzovat všechny odeslané i přijaté požadavky použité při komunikaci mezi komponentami systému.

6.1.6 MmsParamsAndroid

Jak již bylo zmíněno dříve, modul MmsParamsAndroid tvoří upravená aplikace MMS Params, která běží na mobilních telefonech s operačním systémem Android.

Mobilní aplikace se připojuje k řídícímu serveru a čeká na příkazy od testovacího scénáře. Po obdržení příkazu na odeslání SMS nebo MMS zprávy vytvoří aplikace konkrétní SMS zprávu nebo PDU pro odeslání MMS zprávy, nastaví všechny potřebné parametry uvedené v požadavku a zprávu odešle. Po odeslání zprávy odešle zpět testovacímu scénáři informaci o výsledku, konkrétně zdali bylo odeslání zprávy, resp. vyřízení požadavku, úspěšné. Pokud je telefon připojen k serveru a obdrží novou SMS nebo MMS zprávu, je vytvořen nový požadavek, který je odeslán serveru ke zpracování.

6.2 Komunikace přes WebSocket

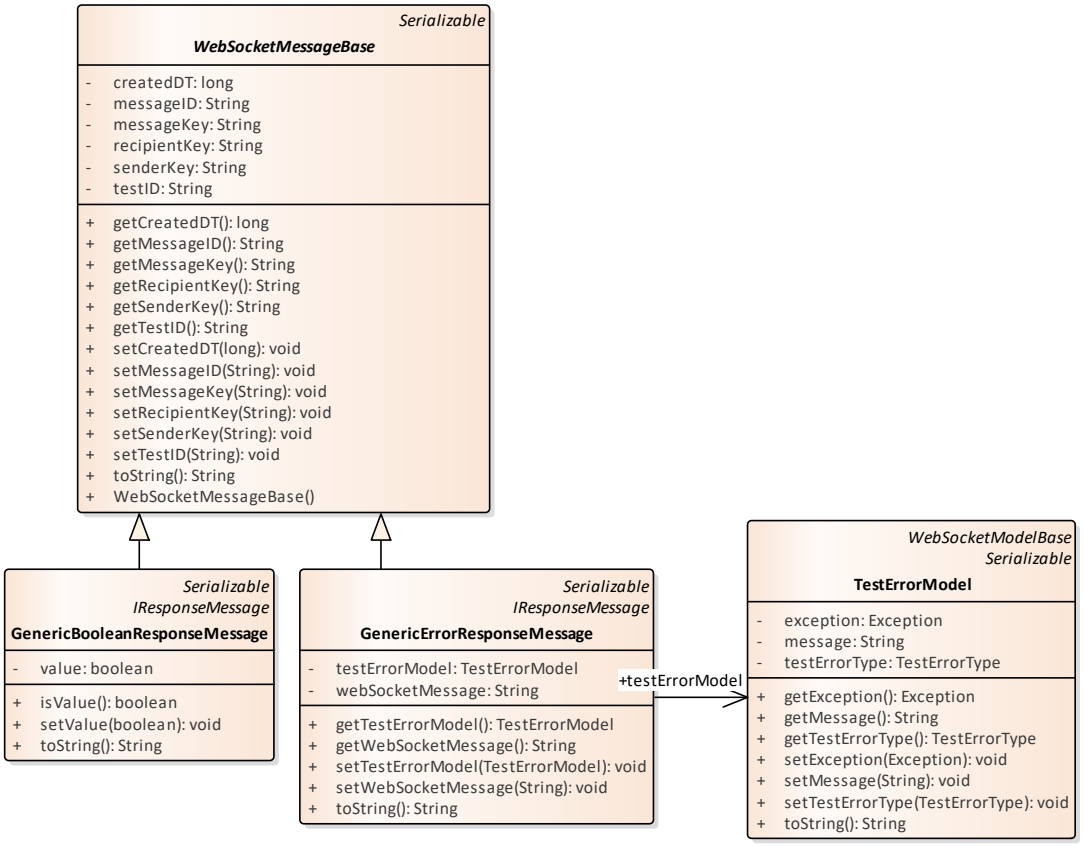
Při komunikaci mezi komponentami (resp. moduly) je nutné rozlišit jednotlivé typy požadavků a jejich parametry. Každý typ požadavku (zprávy) tvoří samostatná třída, která obsahuje potřebné parametry pro jeho uskutečnění a správné vyřízení. Jelikož použitá knihovna pro komunikaci přes WebSocket podporuje pouze odesílání a příjem obsahu v textové podobě (String), je každá zpráva putující mezi komponentami systému před odesláním serializována do formátu JSON a při přijetí je deserializována. Tento přístup vyžaduje, aby byl při přijetí zprávy správně určen její typ a následně při deserializaci proběhlo vytvoření instance správného typu.
Abstraktní třída WebSocketMessageBase je předkem všech zpráv v systému, které mohou být přenášeny přes WebSocket. Z této třídy dědí již konkrétní třídy pro požadavky, které představují například jednotlivé příkazy testu. Třída WebSocketMessageBase obsahuje atributy (viz Tabulka 2), které jsou společné pro každou zprávu a tvoří tak její hlavičku.

Tabulka 2 Atributy hlavičky zprávy

Název atributu	Popis
senderKey	Unikátně vygenerovaný klíč odesílatele požadavku.
recipientKey	Unikátně vygenerovaný klíč příjemce požadavku.
messageKey	Informace o typu požadavku, který slouží pro jeho správné rozpoznání při deserializaci.

messageID	Unikátně vygenerovaný identifikátor požadavku, který je jedinečný pro každý požadavek. Výjimkou jsou odpovědi na požadavky, které mají messageID stejné jako požadavek, aby je bylo možné jednoznačně propojit.
testID	Unikátně vygenerovaný identifikátor testovacího scénáře, který slouží pro spárování každého požadavku s daným scénářem.
createdDT	Datum a čas vytvoření požadavku.

Komunikace je navržena tak, aby na každý požadavek následovalo odeslání odpovědi o výsledku zpracování. Tyto odpovědi lze rozdělit na tři typy. Prvním je obecná odpověď nazvaná GenericBooleanResponseMessage, která obsahuje pouze logickou hodnotu true nebo false a jednoduše informuje, že byl požadavek přijat a úspěšně zpracován. Dalším typem je GenericErrorResponseMessage. Tato odpověď je odeslána v případě vzniku jakékoliv chyby a obsahuje informaci o typu chyby, místě výskytu a případně další doplňkové informace. Pokud například vznikne chyba při odesílání SMS zprávy, je klientu doručena tato odpověď a scénář je ihned ukončen jako neúspěšný. Posledním typem odpovědi jsou všechny ostatní zprávy. Tyto odpovědi se liší podle konkrétního typu příkazu a mohou obsahovat další informace a parametry, které je nutné v odpovědi zahrnout. Obrázek 19 obsahuje UML diagram tříd a ukazuje společnou bázovou třídu WebSocketMessageBase a třídy představující odpovědi na běžné požadavky.



Obrázek 19 UML diagram tříd pro základní komunikaci

6.2.1 Proces deserializace zpráv

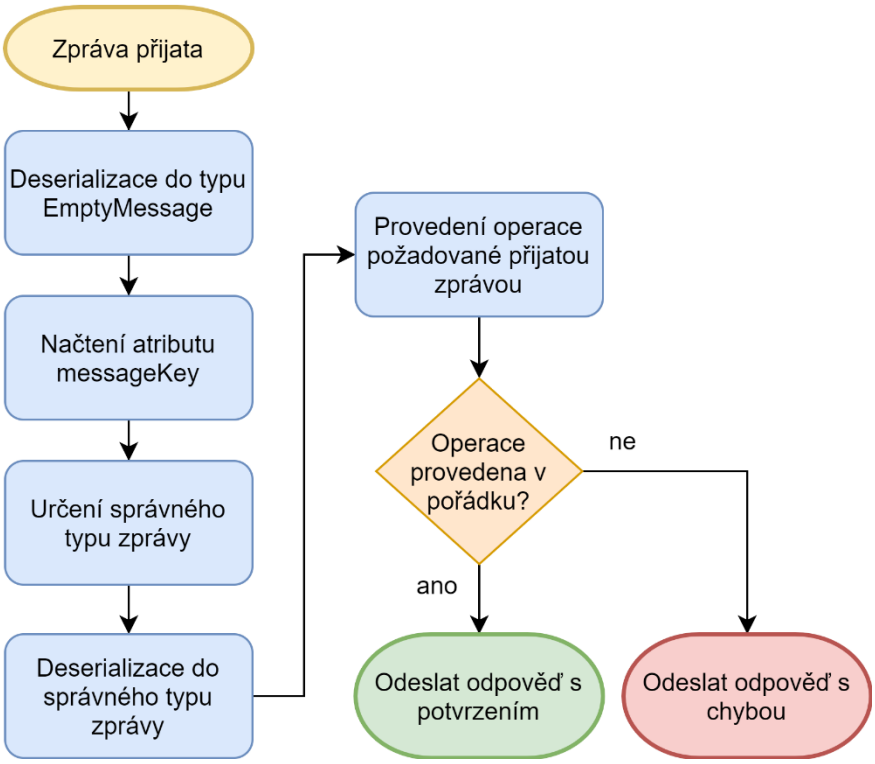
Jak již bylo zmíněno dříve, přijatá zpráva je ve formátu JSON a je nutné určit její typ a deserializovat ji do instance správné třídy. Přijatá zpráva ve formátu JSON je nejdřív deserializována do pomocného typu (třídy) EmptyMessage, který obsahuje pouze hlavičku. Po deserializaci je hlavička zprávy validována, zdali je správně vyplněna a zdali byla doručena správnému příjemci. Po validaci je z hlavičky načten atribut messageKey, který obsahuje informaci o skutečném typu zprávy. Po zjištění toho typu je znovu provedena deserializace, tentokrát již do správného typu a vznikne tak instance třídy reprezentující konkrétní požadavek nebo odpověď.

Ukázka kódu 1 ukazuje proces deserializace zprávy probíhající v klientské části aplikace a mobilní aplikaci. Parametrem metody je instance třídy implementující generické rozhraní IMessageReceiveSub a samotná zpráva ve formátu JSON. Toto rozhraní obsahuje metody onReceive s parametrem pro všechny typy zpráv použité v systému. Implementací tohoto rozhraní může klient snadno rozlišit, která zpráva byla přijata a podle toho určit další chování.

```
public static <T> T process(IMessageReceiveSub<T> iMessageReceiveSub,
              String message)
{
    EmptyMessage emptyMessage = JsonUtilsSafe.fromJson(message,
              EmptyMessage.class);
    if (emptyMessage == null || !MessageUtils.isMessageValid(emptyMessage)) {
        return iMessageReceiveSub.onReceiveUnknown(message);
    }

    String key = emptyMessage.getMessageKey();

    if (MessageType.Register_Phone_Message.equals(key)) {
        RegisterPhoneMessage msg = JsonUtilsSafe.fromJson(message,
              RegisterPhoneMessage.class);
        return iMessageReceiveSub.onReceive(msg);
    }
    else if (MessageType.Phone_List_Request_Message.equals(key)) {
        PhoneListRequestMessage msg = JsonUtilsSafe.fromJson(message,
              PhoneListRequestMessage.class);
        return iMessageReceiveSub.onReceive(msg);
    }
    // Zde pokračují else-if pro každý typ zprávy
```

Ukázka kódu 1 Proces deserializace a rozpoznání zprávy

Obrázek 20 graficky znázorňuje proces deserializace a zpracování zprávy v serverové aplikaci. Úkolem serveru je požadavky zpracovat nebo přeposlat dále, proto se zde hledí i na to, zdali bylo zpracování úspěšné a v případě neúspěchu je odesílatel informován o selhání požadavku.



Obrázek 20 Proces deserializace zprávy na serveru

6.3 Testovací scénář z pohledu klienta

Pro vytváření testovacích scénářů uživatelem slouží modul MmsParamsClientLib, který obsahuje velké množství funkcí, které lze využít pro jednotlivé kroky a příkazy scénáře. Každý testovací scénář by měl být vytvořen jako jedna třída, která musí dědit z třídy RunnableTest. Při použití této abstraktní třídy je třeba implementovat dvě metody a konstruktor. Metoda init požaduje po uživateli vytvoření instance třídy TestSettings. Třída TestSettings obsahuje základní parametry pro spuštění a průběh testu, tedy adresu serveru, přihlašovací jméno a heslo a defaultní hodnotu platnosti každé operace.

Druhá metoda, kterou je nutno v každém testu implementovat se nazývá run. V parametru této metody dostane uživatel instanci třídy implementující rozhraní ITestInstance. Toto rozhraní obsahuje všechny dostupné funkce a služby, které knihovna poskytuje pro vytváření testovacího scénáře. V konstruktoru se bázové třídě předá název scénáře a jeho popisek.

Ukázka kódu 2 obsahuje metodu runTest v třídě RunnableTest, která se používá pro spuštění celého testovacího scénáře. Metoda runTest se postará o správné vytvoření objektu ITestInstance, včetně inicializace všech služeb dostupných pro psaní scénáře a připojení k serveru pomocí WebSocketu. Systém je navíc navržen tak, že každý testovací scénář vytváří vlastní připojení k serveru a po ukončení nebo chybě jej uzavírá. Ihned po připojení k serveru je odeslána první zpráva, která obsahuje informace o vytvořeném scénáři a klientovi. Poté je zavolána metoda run, kde uživatel nadefinoval vlastní operace testu. Na konci testovacího scénáře je zavolána metoda testFinished, která rozhodne o výsledku scénáře a informuje server o jeho ukončení. Celý testovací scénář se spouští v try-catch bloku, aby byly správně odchyceny případné chyby. Pokud kdykoli během testovacího scénáře vznikne chyba, je o této skutečnosti informován server a test je okamžitě ukončen jako neúspěšný.

```
public boolean runTest() {
    ClientLogFacade.clearAll();
    final TestSettings ts = init();
    ITestInstance test = null;
    try {
        test = TestInstance.initNewTest(ts, this.testName, this.testDesc);
        run(test);
        boolean result = test.testFinished();
        afterTestFinished(test.TestInfo());
        if (!result) {
            onTestFailed(test, test.TestInfo(), null);
        }
        return result;
    }
    catch (Exception e) {
        if (test != null) {
            test.testFinishedWithError(e);
            afterTestFinished(test.TestInfo());
            onTestFailed(test, test.TestInfo(), e);
        }
        else {
            onTestFailed(null, null, e);
        }
        return false;
    }
}
```

Ukázka kódu 2 Metoda runTest v třídě RunnableTest

Ukázka kódu 3 je základní šablonou pro každý testovací scénář. Je zde uvedena metoda main, pomocí níž lze vytvořit nový testovací scénář a následně jej spustit.

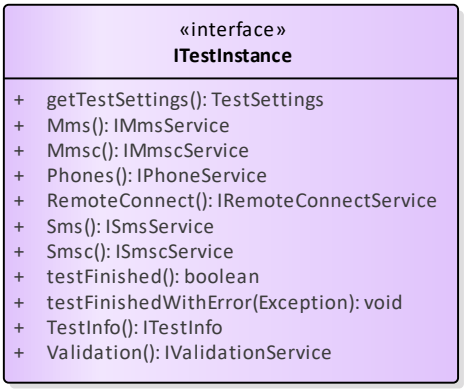
```
public class TestCase1 extends RunnableTest
{
    public static void main(String[] args) {
        new TestCase1().runTest();
    }

    public TestCase1() {
        super("TestCase1", "Test case template");
    }

    @Override
    protected TestSettings init() {
        return new TestSettings(
            new ServerAddressProvider(HttpProtocolEnum.HTTP, "localhost:4301"),
            100,
            "username",
            "password");
    }

    @Override
    protected void run(ITestInstance test) throws Exception {
        // Zde se píšou příkazy testovacího scénáře
        test.testFinished();
    }
}
```

Ukázka kódu 3 Šablona testovacího scénáře

V principu vytváření testovacích scénářů lze nalézt jistou analogii s klasickými jednotkovými nebo integračními testy. V těch se také inicializuje prostředí testu, otestuje nějaká funkce nebo chování, provedou validace a test je ukončen.

6.3.1 Rozhraní ITestInstance

Jak již bylo řečeno, rozhraní ITestInstance (Obrázek 21) poskytuje přístup k dalším službám, které je možné použít při vytváření testovacích scénářů. V podstatě se jedná o oddělené služby, které vytvářejí a odesílají specifické typy požadavků určené pro každou požadovanou operaci.



Obrázek 21 Rozhraní ITestInstance

Tabulka 3 obsahuje základní informace o jednotlivých službách, jejichž příkazy lze využít při vytváření testovacích scénářů. Kompletní seznam všech funkcí včetně detailního popisu lze najít v dokumentaci projektu[1] nebo v rámci zdrojových kódů.

Tabulka 3 Přehled služeb pro testovací scénáře

Název služby	Popis funkce
IPhoneService	• načtení seznamu všech připojených telefonů k serveru • vyhledání připojeného telefonu podle identifikátoru • zamčení a odemčení telefonu
ISmsService	• odeslání SMS zprávy z telefonu • přijetí SMS zprávy z telefonu • přijetí potvrzení o doručení SMS zprávy

[1] https://mmsparams.cz

IMmsService	• odeslání MMS zprávy z telefonu • nastavení profilu příjemce • přijetí MMS zprávy z telefonu • přijetí zprávy o doručení MMS zprávy • přijetí zprávy přečtení MMS zprávy
ISmscService	• připojení k SMS centru • odeslání SMS ze SMSC • přijetí SMS ze SMSC • přijetí zprávy o doručení ze SMSC • odpojení od SMSC
IMmscService	• odeslání MMS z MMSC • určení čísla příjemce pro test • přijetí MMS z MMSC • přijetí zprávy o doručení z MMSC • přijetí zprávy o přečtení z MMSC
IValidationService	• provedení vytvořených validací
ITestInfo	• základní informace o testu • délka trvání testu • seznam přijatých a odeslaných požadavků
IRemoteConnectService	• vzdálené připojení telefonu k serveru pomocí SMS z jiného telefonu nebo SMSC

Většina jednotlivých služeb funguje obdobně a poskytuje základní dva typy operací, konkrétně odeslání nějakého požadavku a čekání na jeho vyřízení nebo doručení nějaké zprávy. Odeslání požadavku se provede vytvořením instance třídy reprezentující daný typ zprávy požadavku a nastavením všech požadovaných parametrů. Zde se může jednat například o obsah SMS zprávy, parametry zprávy MMS nebo informace o připojení k SMS centru. Následuje automatické vyplnění hlavičky zprávy, serializace do formátu JSON a odeslání na server.

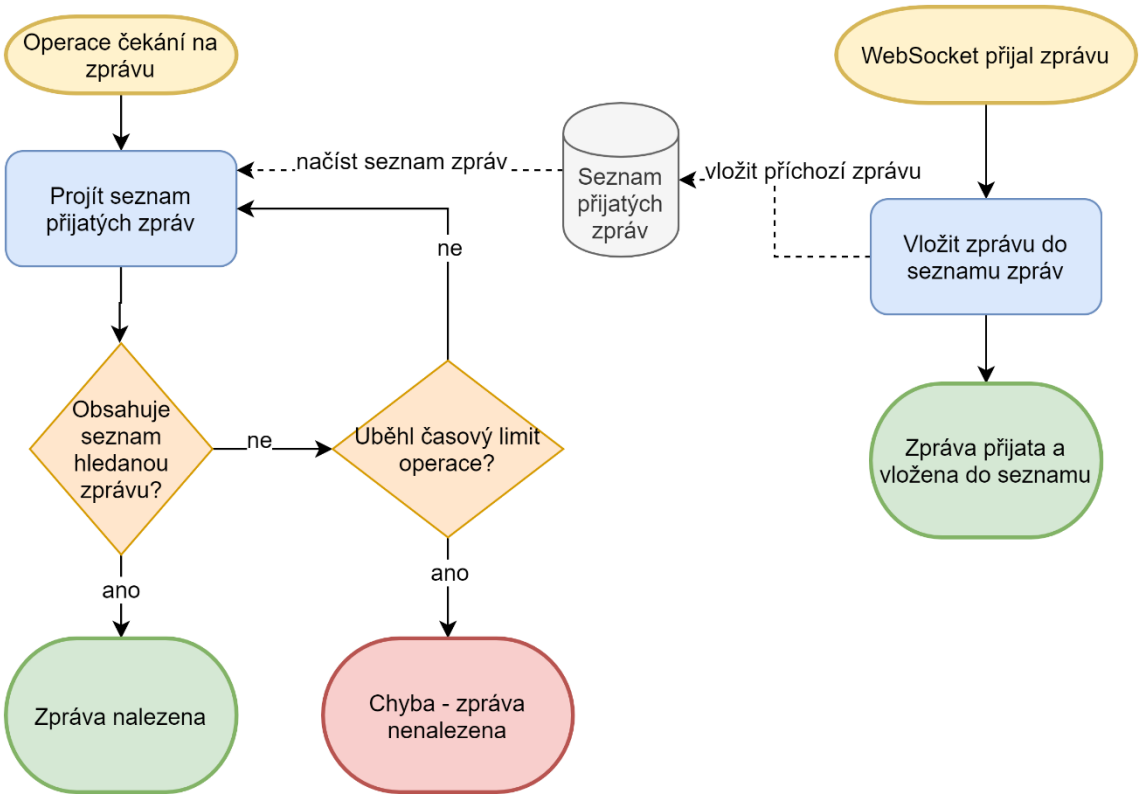
Ukázka kódu 4 obsahuje metodu odesílající požadavek na odeslání nové SMS zprávy z mobilní aplikace. Parametry zadané uživatelem jsou nejprve validovány. Následně je vytvořena instance třídy SmsSendPhoneRequestMessage a je nastaven její atribut, který obsahuje parametry pro vytvoření nové SMS zprávy telefonem. Po vytvoření je požadavek odeslán na server ke zpracování a následuje čekání na odpověď o úspěšném vyřízení. Metoda vrací instanci třídy SmsSendMessageId,

která obsahuje jednoznačný identifikátor odeslaného požadavku, který může uživatel využít pro další operace.

```
@Override
public SmsSendMessageId sendSms(SmsSendModel sms, PhoneInfoModel phoneSender)
{
    SmsSendModelValidator.validate(sms);
    PhoneInfoModelValidator.validate(phoneSender, GenericConstants.PHONE_SENDER);

    SmsSendPhoneRequestMessage request =
        MessageFactory.create(SmsSendPhoneRequestMessage.class);
    request.setSmsSendModel(sms);
        request.setRecipientKey(phoneSender.getPhoneKey());

    SmsSendMessageId messageID = new
        SmsSendMessageId(webSocketSender.sendMessage(request));
    webSocketReceiver.waitForGenericBooleanResponseMessage(messageID);
    return messageID;
}
```

Ukázka kódu 4 Odeslání požadavku se SMS zprávou

Po odeslání každého požadavku se vždy čeká na odpověď o úspěšném zpracování, případně na informaci o chybě. Jelikož příjem zpráv přes WebSocket běží na jiném vlákně, je příchozí zpráva pouze uložena do kolekce přijatých zpráv. Když pak služba čeká na přijetí nějaké zprávy nebo odpovědi na požadavek, pouze ve smyčce prochází tuto kolekci a kontroluje, zdali se v ní daná přijatá zpráva již vyskytuje. Pokud zpráva není v kolekci ani po uběhnutí času platnosti operace, je vyhozena výjimka a test je ukončen s chybou. Pokud je však přijatá zpráva nalezena, je službou zpracována a uživateli je vrácen její obsah. Obsahy zpráv lze následně validovat nebo v některých případech použít jako vstup do dalších příkazů. Obrázek 22 ukazuje průběh čekání na přijetí zprávy graficky.



Obrázek 22 Diagram čekaní na přijetí zprávy

6.4 Testovací scénář z pohledu serveru

K serveru se mohou připojovat testovací scénáře i aplikace v mobilních zařízeních pomocí připojení WebSocket. Server udržuje seznam všech aktuálně připojených zařízení a může tak mezi jimi přeposílat různé zprávy nebo od nich přijímat zprávy s příkazy k vykonání.

Když server obdrží novou zprávu s požadavkem, rozhodne se podle její hlavičky, komu je zpráva určena. Pokud je zpráva určena jinému příjemci, zprávu pouze přepošle na konkrétní telefon nebo klienta. Pokud je zpráva určena serveru (atribut recipientKey = SERVER), zprávu zpracovává server. Hlavními příkazy, které server zpracovává je zaregistrování nového testovacího scénáře, ukládání veškerých informacích o běžících nebo ukončených scénářích a v neposlední řadě také komunikace se SMS a MMS centry.

Při registraci nového testovacího scénáře si server vytvoří novou instanci, do které ukládá všechny parametry testu. Jedná se o všechny telefony použité pro daný scénář, ale také o všechny zprávy, které byly odeslány a přijaty v rámci jednoho scénáře nebo seznam všech validací a chyb, které v průběhu vznikly. Po ukončení

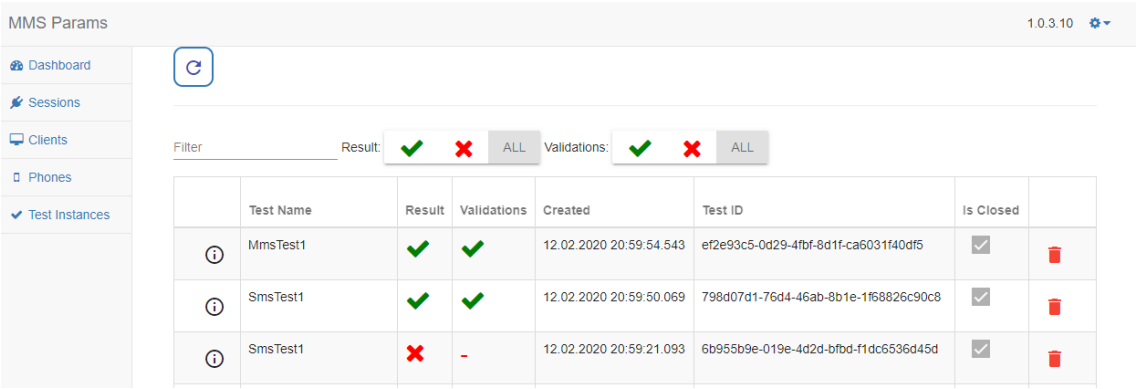
testovacího scénáře jsou všechny tyto informace uloženy do databáze a následně jsou dostupné přes webové rozhraní, kde si uživatel může prohlédnout celý průběh a případně analyzovat chyby, které vznikly nebo validace atributů, které nebyly úspěšné.

Pokud je v klientském scénáři požadováno připojení k SMSC, toto připojení realizuje a udržuje právě server. Následně pomoci něj odesílá požadavky na SMSC k odeslání zprávy. Když je server připojen k SMSC, obdrží také všechny přijaté SMS zprávy a zprávy o doručení.

Jelikož je připojení k MMSC bezstavové, server s tímto centrem neudržuje žádné aktivní spojení a pouze posílá požadavky metodou POST. Server také obsahuje servlet, na který jsou z MMSC přijímány příchozí MMS zprávy stejně jako zprávy o doručení a přečtení.

6.5 Testovací scénář z pohledu webového rozhraní

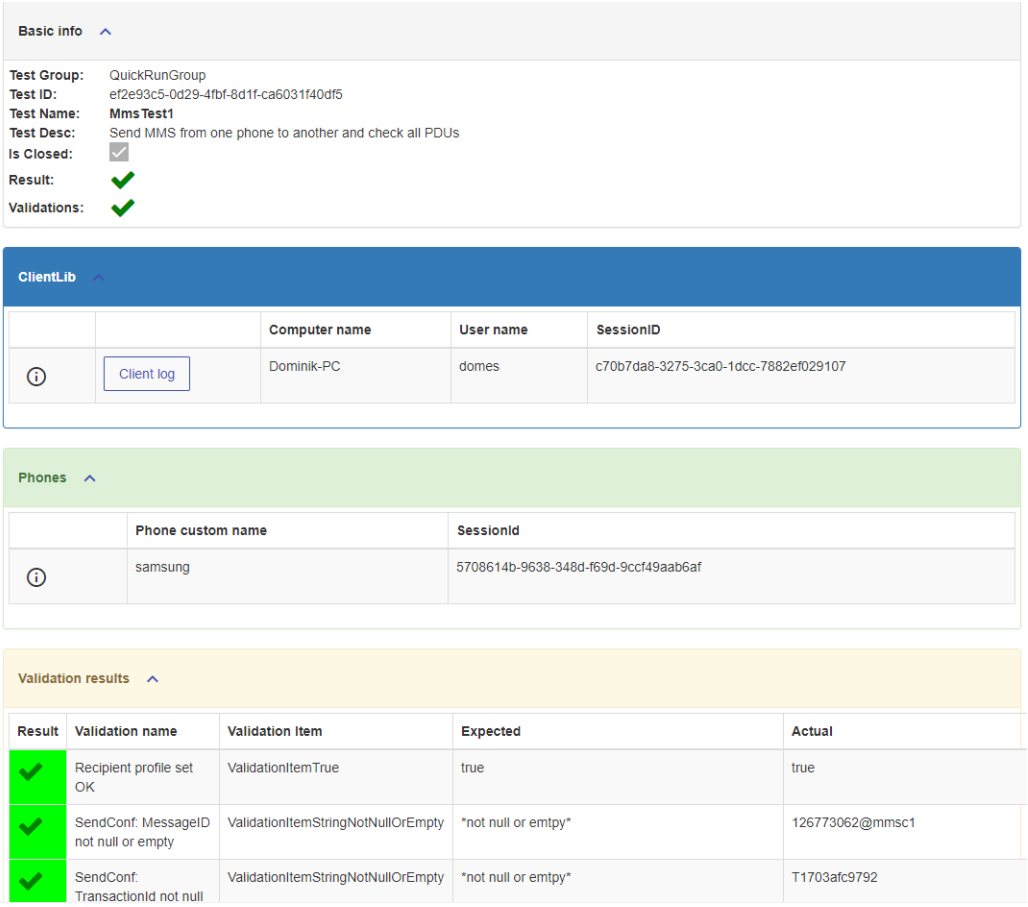
Webové rozhraní vytvořené pomocí frameworku Angular je uživateli poskytováno přímo serverovou aplikací. Zde si lze zobrazit seznam všech připojených mobilních zařízení a seznam běžících testovacích scénářů. Obrázek 23 ukazuje, jak zde vypadá přehled všech ukončených testovacích scénářů.



Obrázek 23 Webové rozhraní – seznam testovacích scénářů

Obrázek 24 následně ukazuje, jak vypadá část detailu jednoho konkrétního ukončeného scénáře. Tento detailní pohled obsahuje základní informace o testovacím scénáři, údaje o počítači, na kterém byl scénář spuštěn a seznam všech použitých mobilních zařízení. Následuje seznam provedených validací a zpráv odeslaných v průběhu testu. V případě vzniku chyby při vykonávání testovacího

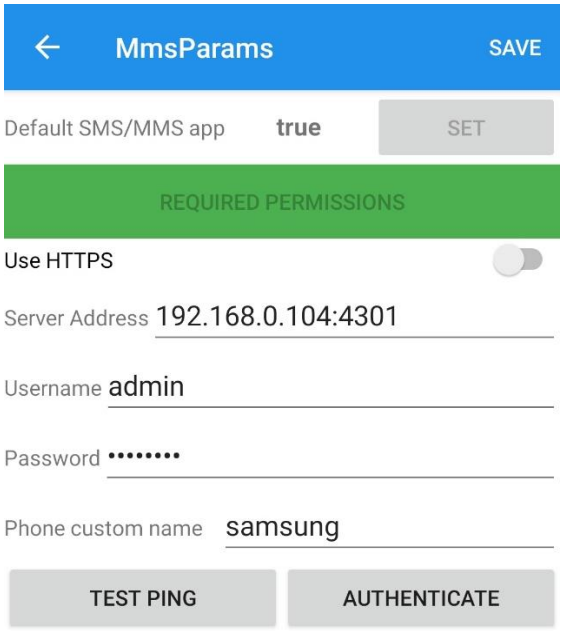
scénáře je zde uveden její detail včetně místa vzniku. Pomocí tohoto přehledu může uživatel analyzovat, v čem přesně se stala chyba a následně upravit testovací scénář nebo hledat další řešení tohoto problému.



Obrázek 24 Webové rozhraní – detail scénáře

6.6 Testovací scénář z pohledu telefonu

Jak již bylo zmíněno v analýze projektu, aplikace pro mobilní telefon je rozšířená existující aplikace z bakalářské práce. Obrázek 25 ukazuje nové nastavení, ve kterém uživatel zadá adresu serveru a přihlašovací údaje. Je zde také kontrola potřebných oprávnění a zdali je aplikace nastavena jako výchozí pro odesílání a přijímání SMS a MMS zpráv.



Obrázek 25 Mobilní aplikace – nastavení připojení

Po tomto nastavení se může telefon připojit k serveru a čekat na příkazy z testovacího scénáře. Aplikace musela být rozšířena právě o toto připojení k serveru a také vyřizování požadavků. Pokud přijde požadavek na odeslání nové SMS nebo MMS zprávy, je tato zpráva patřičně vytvořena a odeslána. Následně je zpět do testovacího scénáře odeslána informace o výsledku odeslání. Pokud telefon přijme novou SMS nebo MMS zprávu, případně zprávu o doručení nebo přečtení a zároveň je připojen k serveru, odešle všechny informace serveru, který se postará o jejich zpracování a případné předání do testovacího scénáře.

6.7 Další funkce systému

Následující kapitoly popisují některé další funkce systému, které nebyly při zadání projektu zmíněny. Tyto funkce vznikly jako nutnost pro lepší fungování systému, jeho správu a zabezpečení nebo jako další požadavek od zadavatele.

6.7.1 Dokumentace

V rámci vývoje systému vznikla dokumentace[2], která má sloužit jako návod pro uživatele systému (administrátory SMS a MMS center) k jejímu používání.

[2] https://mmsparams.cz/

Dokumentace obsahuje základní informace o systému a vytváření testovacích scénářů, návod, jak nastavit mobilní aplikaci pro připojení k řídícímu serveru a také jak ze zdrojových kódů sestavit celý funkční systém včetně správné konfigurace parametrů. Na stránce dokumentace lze také přímo stáhnout soubor s knihovnou pro vytváření testovacích scénářů, serverovou aplikaci nebo soubory pro nainstalování mobilní aplikace.

6.7.2 Chyby v testovacích scénářích

Při spouštění testovacích scénářů může dojít k řadě různých chyb, které musí být možné jednoznačně rozlišit, aby bylo snadné najít jejich příčinu. K tomuto účelu byly možné chyby rozděleny na jednotlivé typy, které vznikají v různých situacích. Tabulka 4 obsahuje výčet jednotlivých typů chyb, místo, kde chyba vzniká a její stručný popis.

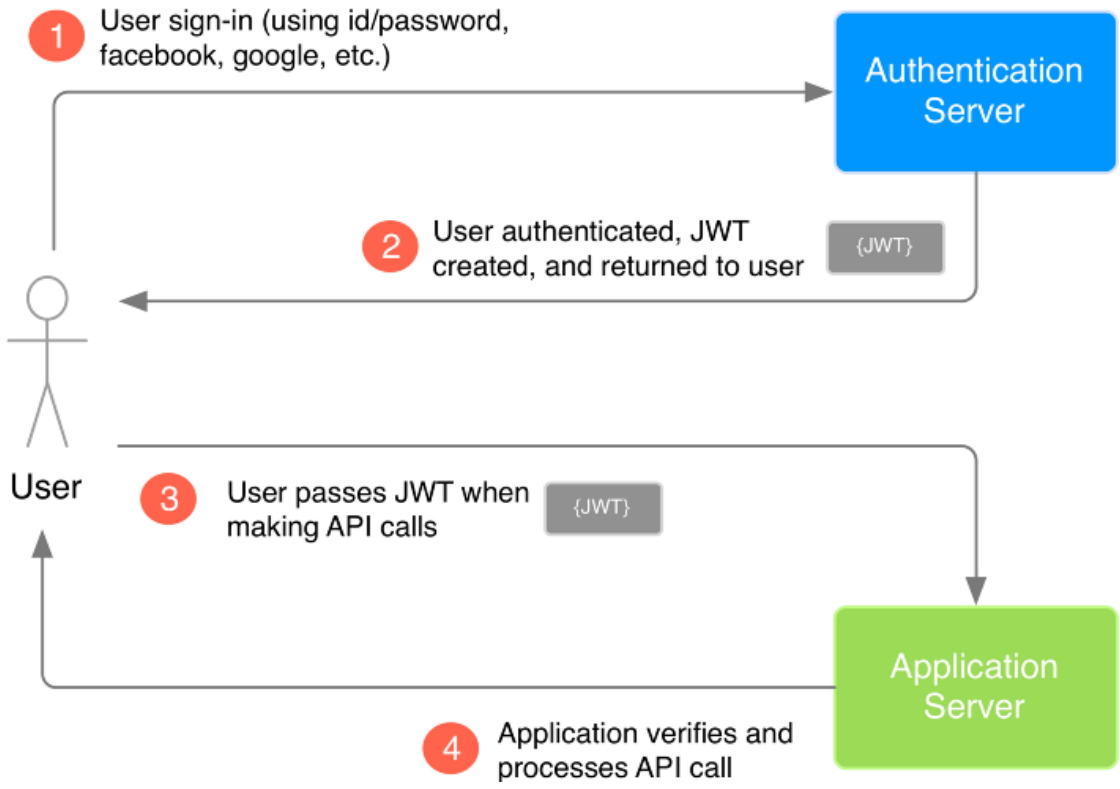
Tabulka 4 Typy chyb testovacích scénářů

Typ chyby	Místo vzniku	Popis
SYSTEM_ERROR	Server	Obecná chyba; test byl neočekávaně ukončen – např. přerušení spojení.
TEST_ERROR	Server	Chyby průběhu testu – např. odeslání zprávy v rámci uzavřeného testu.
WRONGLY_WRITTEN_TEST	Klient	Při volání služby testu byl použit špatný parametr.
VALIDATIONS	Klient	Uživatelské validace nejsou validní.
TIMEOUT	Klient	Vypršel časový limit při čekání na přijetí zprávy.
TEST_FORCE_CLOSE	Server	Vynucené ukončení testu přes webové rozhraní.
SMSC	Server	Chyby spojené s SMS centrem (připojení, odpojení, odeslání, přijetí).
MMSC	Server	Chyby spojené s MMS centrem (připojení, odpojení, odeslání, přijetí).
ANDROID	Mobilní aplikace	Jakákoli chyba v mobilní aplikaci.

6.7.3 Zabezpečení

Zabezpečení aplikace je jednou z nejdůležitějších vlastností celého systému. Očekává se, že serverová aplikace bude mít přímý přístup k SMS a MMS centrům a je tedy nutné ji patřičně zabezpečit, zejména pokud bude její rozhraní přístupné přímo

ze sítě internet. Pro jakoukoli komunikaci se serverem je nutné se nejprve přihlásit pomocí uživatelského jména a hesla. Přihlašování probíhá přes POST požadavek na server, který obsahuje přihlašovací jméno a heslo. Server jméno a heslo ověří a pokud jsou správné, vygeneruje JWT token, který vrátí v odpovědi. Tento token musí být následně uveden v každém dalším požadavku odeslaném na server. Při připojení WebSocketu je tento token vložen do hlavičky při vytváření spojení, tzv. handshake. Pokud není atribut hlavičky Authorization vyplněn, server požadavek odmítne a vrátí chybu 401 (Unauthorized). Obrázek 26 ukazuje obecný princip používání JWT autentizace v praxi.



Obrázek 26 Autentizace pomocí JWT tokenu [24]

JWT token

JWT (JSON Web Token) token je ověřený standard RFC 7519, sloužící k reprezentaci bezpečné komunikace mezi dvěma stranami a lze jej využít k autentizaci uživatele. JWT token je textový řetězec, který se skládá ze tří hlavních částí: hlavička, obsah a podpis. Tyto části jsou odděleny tečkou a jsou ve formátu Base64.

Hlavička obsahuje informace o typu tokenu a algoritmu pro výpočet jeho podpisu. Obsah tokenu obsahuje seznam tzv. Claim, což mohou být informace o uživateli a případně další libovolné údaje nebo data. Typicky se vyplňuje vydavatel tokenu, čas expirace a identifikátor uživatele. Častým obsahem je také seznam oprávnění, které daný uživatel má. Podpis JWT tokenu se vytvoří tak, že se spojí jeho zakódovaná hlavička a obsah (ve formátu Base64) spolu s tajným klíčem. Následně je použit algoritmus uvedený v hlavičce pro vytvoření podpisu.
Hlavičku i obsah je jednoduché dekódovat zpátky do čitelné podoby, takže je možné je použít v klientské aplikaci. Podpis tokenu však vznikl s přidáním tajného klíče, takže jej není možné zfalšovat, a proto se používá pro následné ověření validity celého tokenu serverem. [25]

6.7.4 Logování

Při vytváření systému byl také kladen velký důraz na logování každé důležité akce. Serverová část loguje do souboru všechny akce, které vykoná včetně příchozích a odchozích zpráv. Do dalšího souboru nazvaného FATAL se logují pouze závažné chyby v systému, typicky neočekávané výjimky. Akce spojené se SMS a MMS centry se logují zvlášť do oddělených souborů.
Hlavním důvodem pro takto rozdělené logování je zejména možnost zpětně dohledat konkrétní akce systému a případné chyby. Zprávy spojené se SMS a MMS centry se logují zvlášť, aby mohl administrátor systému jednoznačně najít každou přijatou nebo odeslanou zprávu a případně ji porovnat se záznamy přímo v systému center.
Zvlášť oddělené logování má knihovna pro vytváření testovacích scénářů. Zde se loguje do konzole aplikace, ale je také možné zaregistrovat uživatelem vytvořenou třídu pro vlastní logování. To může být použito v případě, že by uživatel potřeboval např. logování do souboru nebo chtěl logy odesílat do nějakého dalšího systému.
V knihovně pro psaní testovacích scénářů je také speciální třída, která ukládá logy pouze do paměti a po dokončení testu je odešle na server, který je uloží v databázi. Následně je možné na webovém rozhraní zobrazit konkrétní klientské logy ke každému testovacímu scénáři. Tato funkce byla vytvořena z důvodu, že se logy klienta a serveru podstatně liší tím, že klientské logy jsou detailnější

a zaznamenávají konkrétní provedené akce a případné chyby. Pokud test běží automaticky a skončí chybou, může si uživatel ve webovém rozhraní zpětně dohledat, co přesně se v době chyby stalo a případně chybu opravit.

6.7.5 Vzdálené připojení telefonu

Zadavatel projektu uvedl požadavek, že musí být možné telefony vzdáleně připojit k řídícímu serveru a následně je využívat v testovacích scénářích. Tento požadavek vznikl hlavně z důvodu, že většina scénářů bude spouštěna opakovaně a automaticky například každý den. Proto bylo nutné implementovat funkci pro vzdálené připojení, protože se mobilní aplikace může po nějakém čase sama odpojit od serveru nebo může dojít k dočasnému výpadku internetového připojení. Po konzultaci a analýze byla mobilní aplikace upravena, aby se po přijetí SMS zprávy v předem definovaném formátu sama automaticky připojila k serveru.

Ukázka kódu 5 ukazuje příklad takové zprávy, která obsahuje adresu serveru, zašifrované přihlašovací jméno a heslo a další informace pro nastavení mobilní aplikace. V uživatelské knihovně pro vytváření testovacích scénářů existuje služba, která tuto SMS zprávu s příkazem umožňuje odeslat pomocí jiného mobilního telefonu nebo přes SMS centrum.

```
MMSPARAMS_CONNECT_WS;192.168.0.101:4301;true;samsung_phone;YWRtaW51c2VybmFtZTpteXN
lY3JldHBhc3N3b3Jk
```

Ukázka kódu 5 Řetězec pro vzdálené připojení mobilní aplikace

6.7.6 Firebase

Systém také nepřímo využívá platformu Firebase. Pokud v mobilní aplikaci vznikne chyba, je automaticky odeslána na tuto platformu, kde je možné zjistit více informací a případně ji následně opravit. Firebase je také použit pro hostování webové stránky s dokumentací projektu.

7 Testování hotového řešení

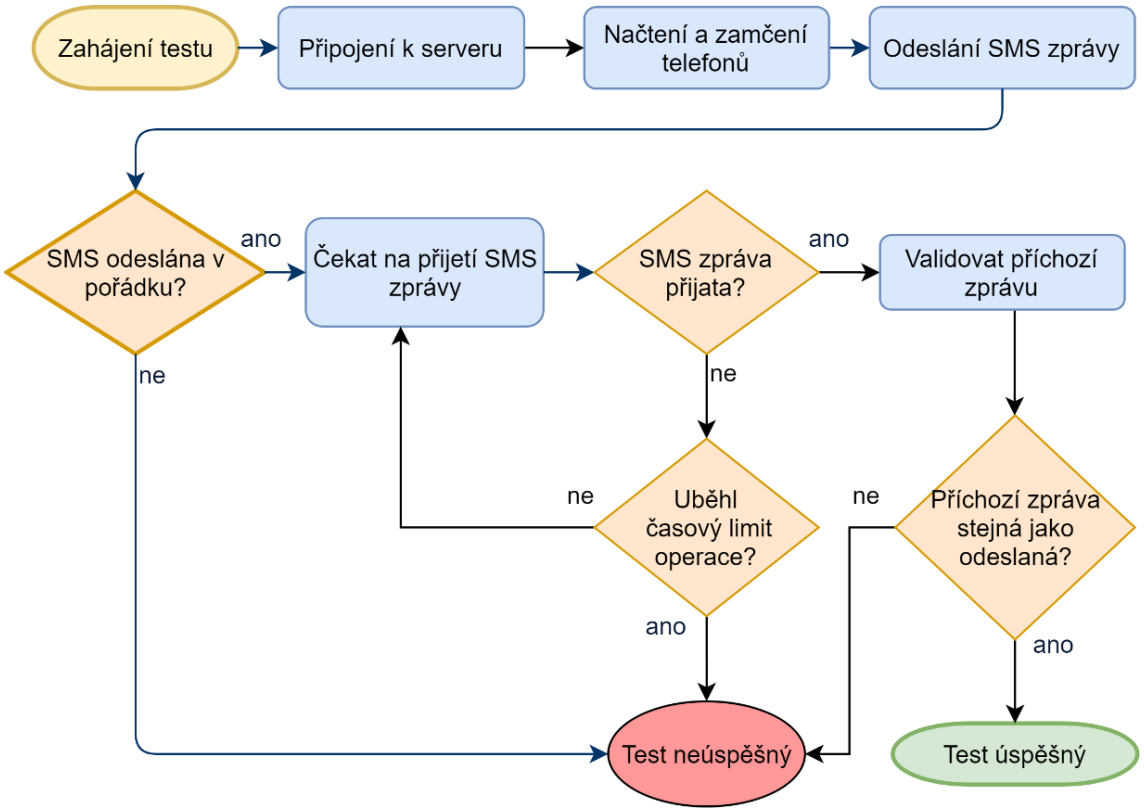
V průběhu celého vývoje probíhalo testování jednotlivých funkcí a bylo opraveno několik vzniklých chyb. V rámci projektu také vzniklo více než pět set jednotkových testů, které mají za úkol testovat různé komponenty systému a jejich správné fungování již během vývoje. Existuje také několik integračních testů, které testují větší celky systému a snižují tak pravděpodobnost vzniku chyb při pozdějších úpravách nebo při přidávání nových funkcí.

7.1 Ukázkové testovací scénáře

Automatizované end-to-end testování SMS a MMS center probíhá pomocí předpřipravených testovacích scénářů, které si uživatel může vytvářet dle vlastní potřeby. Pro ukázku bylo vytvořeno několik ukázkových testovacích scénářů pro různé situace a parametry, aby uživatel lépe pochopil princip a možnosti jednotlivých funkcí knihovny. Tyto scénáře jsou součástí zdrojových kódů aplikace[3]. V následujícím odstavci je popsán testovací scénář nazvaný SmsTest1. Tento scénáře je také znázorněn grafiky jako Obrázek 27 a ve formě kompletního zdrojového kódu jako Ukázka kódu 6.

Testovací scénář SmsTest1 má za úkol otestovat odeslání a přijetí SMS zprávy mezi dvěma telefony. Nedříve jsou načteny oba telefony určené pro test a jsou tzv. zamčeny, aby je v průběhu testu nemohl použít žádný jiný scénář. Následně je odeslán příkaz pro odeslání SMS zprávy z jednoho telefonu a čeká se na potvrzení úspěšného odeslání. Po odeslání testovací scénář čeká na doručení SMS zprávy druhým telefonem. Pokud je zpráva doručena v pořádku, jsou provedeny validace, že je text přijaté zprávy shodný s textem zprávy odeslané. Jestliže je vše v pořádku, test je ukončen jako úspěšný a testovací scénář se odpojí od serveru. V opačném případě je test neúspěšný a na server je odeslána chyba.

[3] https://bitbucket.org/domestos99/mmsparams2/src



Obrázek 27 Diagram testovacího scénáře SmsTest1

```
public class SmsTest1 extends RunnableTest {
    public SmsTest1() {
        super("SmsTest1", "Zprava delky 160 znaku");
    }

    @Override
    protected TestSettings init() {
        return AppConfig.getDefaultTestSettings();
    }

    @Override
    protected void run(ITestInstance test) throws Exception {
        PhoneInfoModel phoneA =
           test.Phones().getByCustomNameAndLock(AppConfig.CustomNameA);
        PhoneInfoModel phoneB =
           test.Phones().getByCustomNameAndLock(AppConfig.CustomNameB);

        SmsSendModel sms = new SmsSendModel();
        sms.setTo(AppConfig.NumberToB);
        sms.setServiceCenterSender(AppConfig.SMS_CENTER_ADDRESS);
        sms.setDeliveryReport(true);

        // Generate random text with prefix and length 160 characters total
        String smsText = SampleTextUtils.generateText("SmsTest1", 160);
        sms.setText(smsText);

        // Send SMS
        SmsSendMessageId smsSendID = test.Sms().sendSms(sms, phoneA);
        SmsSendResponseModel sendOk =
           test.Sms().waitFor().smsSentSuccessfully(smsSendID);

        // Receive SMS
        SmsReceiveModel smsReceiveModel =
           test.Sms().waitFor().anySmsReceived(phoneB);

        // Validations
        ValidationBuilder vb = new ValidationBuilder();
        vb.withEquals(SmsConstants.RESULT_OK, sendOk.getResult(), "SMS send ok");
        vb.withEquals(smsText, smsReceiveModel.getMessage(), "Sms text");
        test.Validation().validatePrintThrow(vb);
    }
}
```

Ukázka kódu 6 Zdrojový kód testovacího scénáře SmsTest1

V ukázkovém zdrojovém kódu testovacího scénáře je také použita třída AppConfig. Tato třída byla vytvořena jako součást ukázkových testovacích scénářů a obsahuje různé konstanty pro jednotlivé testy. Konkrétně se jedná o telefonní čísla používaných telefonů nebo adresu serveru či přihlašovací údaje. Pokud se uživatel rozhodne používat ukázkové scénáře pro reálné testovaní, stačí mu upravit hodnoty konstant v této třídě a může rovnou začít s testováním. Použití této třídy tedy umožňuje mít např. telefonní čísla pouze na jednom místě a následně není nutné je vkládat a později upravovat v každém testovacím scénáři zvlášť.

7.2 Rozšiřitelnost systému

Při návrhu a vývoji systému byl kladen důraz na co největší flexibilitu a možnosti nastavování parametrů jednotlivých zpráv. Již při vývoji a konzultacích se zadavatelem vzniklo několik doplňkových požadavků na další funkce, přičemž mnoho z nich bylo úspěšně implementováno. Následující kapitoly se věnují dalším možnostem, jak by bylo možné systém v budoucnu rozšířit.

Pravidelné spouštění testovacích scénářů

Jednou z myšlenek při zadávání systému bylo, že by byly testovací scénáře pouštěny pravidelně například každý den nebo týden. Tato myšlenka byla uskutečněna jen částečně a to tak, že je každý testovací scénář samostatná třída v Javě, kterou je možné spustit pomocí metody main. Pro pravidelné spouštění scénářů je tedy možné vytvořit spustitelné soubory pro každý testovací scénář a následně ho spouštět např. skriptem přímo v operačním systému. Možným rozšířením by zde mohlo být vytvoření nějaké služby, která by byla spuštěná pořád a v daný čas by spustila testovací scénář nebo jejich skupinu. Tato služba by případně mohla být přímo součástí serverové aplikace, kde by se testovací scénáře spouštěly v nějaké naplánované úloze. Systém by tak nebyl závislý na operačním systému a byl by i snáze přenositelný např. na jiný fyzický server.

Odeslání MMS zprávy na email

SMS a MMS centra umožňují doručení zprávy na emailovou adresu. To lze provést tak, že se místo čísla příjemce vyplní emailová adresa. Současná verze systému toto doručení umožňuje, avšak pro účely testování by bylo nutné přijatou zprávu načíst přímo z emailové schránky, resp. emailového serveru. Toto řešení by vyžadovalo rozšíření o funkce načítající emailové zprávy a následné zpracování a validování. Mohl by ale vzniknout problém v určování, která emailová zpráva patří současně běžícímu testovacímu scénáři, protože při spuštění více takovýchto scénářů naráz nemusí být zřejmé jaké odeslané zprávě odpovídá jaký email. Mohlo by se také stát, že by email skončil v jiné složce, než by se předpokládalo (např. spam) a testovací scénář by skončil s chybou.

Notifikování o neúspěšných testovacích scénářích

Další funkcí systému by mohlo být lepší informování o testovacích scénářích, které skončily s chybou. Tato funkce přímo souvisí s pravidelným spouštěním scénářů v nějakém intervalu. Při tomto spouštění by uživatel musel pravidelně kontrolovat stav všech provedených testovacích scénářů na webovém rozhraní a ověřovat, jestli byly úspěšné. Bylo by tedy možné systém rozšířit o odesílání notifikací např. emailem nebo přímo SMS zprávou. Uživatel by následně kontroloval stav ukončených testovacích scénářů pouze při obdržení této notifikace.

7.3 Problémy při vývoji systému

Závěrečné kapitoly popisují některé problémy, které vznikly při vytváření systému. Problémy jsou doplněny popisem možných řešení, které mohou přispět ke snížení pravděpodobnosti jejich vzniku.

7.3.1 Odpojování mobilní aplikace

Aplikace v mobilním telefonu je připojena k řídícímu serveru pomocí WebSocketu. Pro správné fungování je nezbytné, aby byl telefon připojen během průběhu celého testovacího scénáře. Nicméně se může stát, že operační systém Android toto připojení uzavře z důvodu bezpečnosti, uvolňování paměti nebo jiných, předem neznámých důvodů. Při testování aplikace tento problém nenastal přímo v průběhu spuštěného testovacího scénáře, ale bylo běžné, že se telefon odpojil v době nečinnosti. Pro částečné zamezení vzniku tohoto problému byla v mobilní aplikaci implementována funkce, která po připojení pravidelně odesílá na server prázdné požadavky, které jsou ignorovány. Tím je vytvořena občasná aktivita a připojení tak není nečinné. Dalším řešením může být zavolání funkce pro vzdálené připojení telefonu před spuštěním každého testovacího scénáře nebo skupiny scénářů, aby se zajistilo, že telefon bude připojen.

7.3.2 Rozpoznávání příchozích zpráv

Když je mobilní aplikace připojena k serveru, přeposílá mu všechny doručené SMS a MMS zprávy. Pokud je spuštěn testovací scénář, ve kterém se čeká na doručení zprávy na daný telefon a ve stejný okamžik obdrží telefon novou zprávu z jiného

zdroje, může dojít k chybě testovacího scénáře. Příčina je v tom, že testovací scénář většinou čeká na první doručenou zprávu a pokud tato zpráva není součástí testovacího scénáře, bude obsahat jiné parametry a text zprávy. To následně způsobí chybu při validacích a tím i selhání celého scénáře. Tomuto problému se dá předejít používáním mobilních telefonů, u kterých je malá pravděpodobnost, že budou přijímat jiné zprávy než zprávy testovacích scénářů. Také je možné pro každý scénář vytvořit vlastní unikátní prefix textu zprávy. To zajistí, že při kontrole neúspěšných validací bude rychle odhalen rozdíl v tomto textu a bude zřejmé, že došlo ke kolizi zpráv při čekání na přijetí zprávy.

7.4 Jak systém získat a nainstalovat

Zdrojové kódy systému lze nalézt ve veřejném repositáři verzovacího systému Bitbucket[4]. Na stránce dokumentace projektu[5] lze najít podrobný návod, jak systém sestavit a následně používat včetně popisu základních funkcí a doporučení. Jsou zde také ke stažení soubory s knihovnou pro vytváření testovacích scénářů a serverová aplikace ve formátu jar.

[4] https://bitbucket.org/domestos99/mmsparams2
[5] https://mmsparams.cz/

8 Závěr

Hlavním cílem této práce bylo popsat princip fungování SMS a MMS center z pohledu mobilních zařízení a externích systémů, které je využívají. Práce také obsahuje popis standardizace celého odvětví, informace o fungování mobilních sítí a připojování k nim. Následoval popis SMS a MMS center a jejich fungování. Další část práce se zabývala automatizovaným testováním těchto center a rešerší existujících řešení.

Dalším cílem bylo rozšířit již existující mobilní aplikaci a vytvořit tak systém pro automatizované end-to-end testování SMS a MMS center, který umožní administrátorům mobilních center vytvářet různé testovací scénáře, které budou poskytovat možnost rychle a snadno otestovat velké množství funkcionalit nebo chování naráz. Tento cíl byl splněn a byl vytvořen plně funkční a otestovaný systém, který toto testování umožňuje. Součástí zdrojových kódů jsou ukázkové testovací scénáře připravené pro okamžité používání nebo další rozšiřování administrátory SMS a MMS center. Toto řešení ušetří spoustu času a nákladů spojených s pravidelným preventivním testováním nebo testováním před nasazením nové verze konkrétního centra.

Seznam použité literatury

[1] PLAŠIL, Dominik. *Mobilní aplikace pro podporu testování MMS centra* [online]. B.m.: Univerzita Hradec Králové, Fakulta informatiky a managementu, Hradec Králové, 2018. Dostupné z: https://theses.cz/id/f9d43u/

[2] LE BODIC, Gwenaël. *Mobile messaging technologies and services: SMS, EMS, and MMS*. 2nd ed. Chichester, West Sussex, England ; Hoboken, NJ: J. Wiley & Sons, 2005. ISBN 978-0-470-01143-0.

[3] SMS and MMS: messages per day in Germany 2014. *Statista* [online]. [vid. 2020-02-28]. Dostupné z: https://www.statista.com/statistics/461700/number-of-sms-and-mms-sent-per-day-germany/

[4] 3GPP. *About 3GPP* [online]. [vid. 2020-02-01]. Dostupné z: https://www.3gpp.org/about-3gpp

[5] 3GPP. Open Mobile Alliance and 3GPP. In: [online]. B.m. Dostupné z: ftp://www.3gpp.org/PCG/PCG_09/DOCS/PDF/PCG9_22.pdf

[6] SPECWORKS, O. M. A. About OMA SpecWorks. *OMA SpecWorks* [online]. [vid. 2020-02-01]. Dostupné z: https://www.omaspecworks.org/about/

[7] HEINE, Gunnar. *GSM networks: protocols, terminology, and implementation* [online]. Boston, Mass: Artech House, 1999. Mobile communications series. ISBN 978-0-89006-471-9. Dostupné z: https://ss7.at.ua/_ld/0/13_GSMM.pdf

[8] ŠIMKO, Andrej. *Forenzní analýza SIM karet [online]* [online]. B.m.: Masarykova univerzita, Fakulta informatiky, Brno, 2012. Dostupné z: Dostupné z WWW <https://is.muni.cz/th/ghcai/>

[9] 3GPP. *3GPP TS 23.040* [online]. B.m.: http://www.3gpp.org. 2019. Dostupné z: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=747

[10] MAHMOUD, Tarek M a Ahmed M MAHFOUZ. SMS Spam Filtering Technique Based on Artificial Immune System [online]. 2012, 9(2), 9. ISSN 1694-0814. Dostupné z: doi:10.1.1.402.2510

[11] SMPP DEVELOPERS FORUM. *Short Message Peer to Peer Protocol Specification v3.4* [online]. 1999. Dostupné z: http://docs.nimta.com/SMPP_v3_4_Issue1_2.pdf

[12] SMS FORUM. *Short Message Perr-to-Peer Protocol Specification Version 5.0* [online]. B.m.: www.smsforum.net. 2003. Dostupné z: http://opensmpp.org/specs/smppv50.pdf

[13] *NowSMS | SMS Gateway, SMS Server Software, MMS Gateway & MMSC | NowSMS* [online]. [vid. 2019-11-28]. Dostupné z: https://www.nowsms.com/

[14] 3GPP. *3GPP TS 23.140* [online]. B.m.: http://www.3gpp.org. 2009. Dostupné z: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=794

[15] OPEN MOBILE ALLIANCE. *Multimedia Messaging Service Client Transactions* [online]. 2011. Dostupné z: http://www.openmobilealliance.org/release/MMS/V1_3-20110913-A/OMA-TS-MMS_CTR-V1_3-20110913-A.pdf

[16] WORLD WIDE WEB CONSORTIUM. *SOAP Messages with Attachments* [online]. [vid. 2020-01-31]. Dostupné z: https://www.w3.org/TR/SOAP-attachments

[17] VATSA, R. a V. KUMAR. Role of media transformation in multimedia messaging. In: *2005 IEEE International Conference on Personal Wireless Communications, 2005. ICPWC 2005.* [online]. 2005, s. 258–262. Dostupné z: doi:10.1109/ICPWC.2005.1431344

[18] MMS. *O2* [online]. [vid. 2020-01-31]. Dostupné z: https://www.o2.cz/osobni/203283-sms_mms/21384-mms.html

[19] GSM ASSOCIATIAN. *MMS Inter-working Tests* [online]. 2012. Dostupné z: https://www.gsma.com/newsroom/wp-content/uploads/IR.53-v3.1.pdf

[20] MULLINER, Collin a Giovanni VIGNA. Vulnerability Analysis of MMS User Agents. In: *2006 22nd Computer Security Applications Conference*: *2006 22nd Annual Computer Security Applications Conference (ACSAC'06)* [online]. Miami Beach, FL: IEEE, 2006, s. 77–88 [vid. 2019-11-28]. ISBN 978-0-7695-2716-1. Dostupné z: doi:10.1109/ACSAC.2006.55

[21] TRAYNOR, Patrick, Patrick MCDANIEL a Thomas La PORTA. Vulnerabilities in the Short Messaging Service (SMS). In: Patrick TRAYNOR, Patrick MCDANIEL a Thomas La PORTA, ed. *Security for Telecommunications Networks* [online]. Boston, MA: Springer US, 2008 [vid. 2019-11-28], Advances in Information Security, s. 65–108. ISBN 978-0-387-72442-3. Dostupné z: doi:10.1007/978-0-387-72442-3_5

[22] *Test My SMS* [online]. [vid. 2019-11-28]. Dostupné z: https://testmysms.com/

[23] SIGOS - No. 1 in mobile experience and services. *SIGOS* [online]. [vid. 2019-11-28]. Dostupné z: https://www.sigos.com/

[24] *JSON Web Tokens (JWT) – zooom.cz* [online]. [vid. 2020-02-25]. Dostupné z: https://zooom.cz/json-web-tokens-jwt/

[25] AUTH0.COM. *JWT.IO - JSON Web Tokens Introduction* [online]. [vid. 2020-02-25]. Dostupné z: http://jwt.io/

Přílohy

1. Obsah elektronické přílohy
    a. Zdrojové kódy systému
2. Veřejný repositář v systému Bitbucket s poslední verzí aplikace
    a. https://bitbucket.org/domestos99/mmsparams2/src
3. Dokumentace projektu a instalační soubory systému
    a. https://mmsparams.cz/

Oskenované zadání práce

UNIVERZITA HRADEC KRÁLOVÉ
Fakulta informatiky a managementu
Akademický rok: 2018/2019

Studijní program: Aplikovaná informatika
Forma studia: Prezenční
Obor/kombinace: Aplikovaná informatika (ai2-p)

Podklad pro zadání DIPLOMOVÉ práce studenta

Jméno a příjmení:	**Bc. Dominik Plašil**
Osobní číslo:	**I1800118**
Adresa:	Mírová 61, Velké Meziříčí, 59401 Velké Meziříčí, Česká republika
Téma práce:	Automatizované end-to-end testovaní SMS a MMS center
Téma práce anglicky:	Automated end-to-end testing of SMS and MMS centers
Vedoucí práce:	Ing. Pavel Kříž, Ph.D. Katedra informatiky a kvantitativních metod

Zásady pro vypracování:

Cíl: Vytvořit systém pro podporu end-to-end testování SMS a MMS center. Systém se bude skládat z webového serveru, mobilní aplikace pro operační systém Android a knihovny pro vytváření testovacích scénářů. Systém bude komunikovat s aplikací v mobilním telefonu, SMS centrem přes protokol SMPP a MMS centrem přes rozhraní MM7.

Osnova:

1. Úvod
2. Cíl
3. SMS a MMS centra
4. Automatizované testování
5. Rešerše existujících řešení
6. Analýza a návrh vlastního řešení
7. Implementace
8. Výsledky
9. Závěr

Seznam doporučené literatury:

1) LE BODIC, Gwenael. Mobile messaging technologies and services: SMS, EMS and MMS. 2nd ed. Hoboken, NJ: J. Wiley, c2005. ISBN 0470011432. 2) https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=794 3) http://www.openmobilealliance.org/release/MMS/V1_3-20050927-C/OMA-TS-MMS-CTR-V1_3-20050927-C.pdf 4) http://www.qtc.jp/3GPP/Specs/23140-6g0.pdf

Podpis studenta: Datum:

Podpis vedoucího práce: Datum: