



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

## **APLIKACE PRO REZERVACI VSTUPENEK V IMS**

APPLICATION FOR THE RESERVATION OF TICKETS IN THE IMS

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. PAVEL OTOUPALÍK**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. L'UBOŠ NAGY**

BRNO 2010



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Diplomová práce

magisterský navazující studijní obor  
**Telekomunikační a informační technika**

**Student:** Bc. Pavel Otoupalík

**ID:** 78437

**Ročník:** 2

**Akademický rok:** 2009/2010

## NÁZEV TÉMATU:

### **Aplikace pro rezervaci vstupenek v IMS**

#### **POKYNY PRO VYPRACOVÁNÍ:**

Cílem projektu je prostudovat a popsat architekturu IMS (IP Multimedia Subsystem) se zaměřením na vývoj aplikací pro tuto technologii. Na základě získaných poznatků navrhnete systém sestávající z klientského modulu a z modulu pro přístup a správu databáze, který umožní uživatelům aplikace online rezervaci vstupenek prostřednictvím IMS sítě. Aplikaci klienta bude určena pro mobilní terminály. Aplikaci realizujte v prostředí SDS od firmy Ericsson. Následně proveďte podrobnou analýzu SIP komunikace v navrženém projektu.

#### **DOPORUČENÁ LITERATURA:**

[1] RUSSELL, Travis. The IP Multimedia Subsystem (IMS): Session Control and Other Network Operations. V. Británie: Mc Graw-Hill OSBOURNE, 2008. 242 s. ISBN 0071488537.

[2] POIKSELKA, Miikka, MAYER, Gregor, KHARTABIL, Hisham, NIEMI, Aki. The IMS: IP Multimedia Concepts and Services. V. Británie: WILEY, 2006. 466 s. Second edition. ISBN 978-0-470-01906-1.

**Termín zadání:** 29.1.2010

**Termín odevzdání:** 26.5.2010

**Vedoucí práce:** Ing. Ľuboš Nagy

**prof. Ing. Kamil Vrba, CSc.**

*Předseda oborové rady*

#### **UPOZORNĚNÍ:**

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

# Anotace

Tato práce se zabývá popisem IMS (Internet Protocol Multimedia Subsystem) a možnostmi tvorby aplikací využívajících služeb tohoto systému.

Obsahem teoretické části této práce je podrobný popis síťové architektury IMS a její jednotlivých prvků. V práci je dále popsána struktura protokolu SIP, který je v IMS používán pro přenos signalizace.

V praktické části projektu je popsán návrh, tvorba a testování systému umožňující rezervaci vstupenek do kina prostřednictvím IMS. Pro vývoj aplikací využívajících služeb IMS bylo použito vývojové prostředí SDS 4.1 FD1 od firmy Ericsson, které umožňuje pro účel testování aplikací simulovat IMS síť. Vytvořené aplikace jsou na bázi klient-server. Klientská aplikace je určena mobilním zařízeníům s podporou technologie J2ME. Serverová část systému je realizována jako SIP Servlet aplikace, která má přístup do MySQL databáze, v níž jsou uloženy záznamy o registrovaných uživateli, nabízených vstupenkách a provedených rezervacích.

Aby mohla být práce nad databází MySQL uživatelsky příznivá a snadná, byla vytvořena aplikace na bázi platformy Java SE 6, která je určena právě pro účel správy databáze systému pro rezervaci vstupenek. Díky této aplikaci může obsluha systému snadno přidávat nebo odebírat filmy, dobu jejich promítání a na základě znalosti identifikační čísla rezervací provádět výdej lístků.

V závěru práce je provedena SIP analýza registrace klienta do IMS a komunikace mezi klientskou aplikací a SIP Servletem.

**Klíčová slova:** IMS, rezervace vstupenek, SIP, Servlet, J2ME, Java, MySQL

# Abstract

This master's thesis deals description of the IMS (Internet Protocol Multimedia Subsystem) and shows the possibility for design applications that using services this system.

The theoretical part of the thesis is focused on a detailed description of the IMS architecture, its key components and the SIP protocol that is main signaling protocol in the IMS.

The practical part of the work describes the design, implementation and test of the system for reservation of tickets to cinema in the IMS. The project was created as client-server application in the development studio SDS Ericsson 4.1 FD1 with built-in IMS simulator. The client application is designed for mobile devices with support of the J2ME platform. The server part of the system is realized as SIP Servlet with access to the MySQL database. In the database are saves records about registered users, offered tickets and successful reservation.

For easy work over MySQL database of the system for reservation of tickets was created the application for its administration in the Java programming language (Java SE 6). With the assistance of this application, the operators of system can add and erase the records about films, time projection and disburse tickets on the basis of knowledge of reservation ID.

In the end of the work is realized SIP signaling analysis of the client registration to IMS and communication between the client application and the SIP Servlet.

**Keywords:** IMS, reservation of tickets, SIP, Servlet, IJCU, Java, MySQL

OTOUPALÍK, P. *Aplikace pro rezervaci vstupenek v IMS*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 85 s. Vedoucí diplomové práce Ing. Ľuboš Nagy.

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma Aplikace pro rezervaci vstupenek v IMS jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením tohoto projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce Ing. Ľubošovi Nagyovi, za velmi užitečnou metodickou pomoc a cenné rady při zpracování diplomové práce.

## Seznam zkratek

AMR	Adaptive Multi-Rate
API	Application Programming Interface
AS	Application Servers
AuC	Authentication Center
AUTN	Network Authentication Token
AV	Autentizační Vektor
BGCF	Breakout Gateway Control Functions
BSD	Berkeley Software Distribution
	Customized Applications for Mobile network Enhanced
CAMEL	Logic
CAP	CAMEL Application Part
CK	Ciphering Key
CLDC	Connected Limited Device Configuration
CS	Circuit Switching
CSCF	Call/Session Control Functions
DNS	Domain Name System
DOM	Document Object Model
DSL	Digital Subscriber Line
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
GUI	Graphical User Interface
HLR	Home Location Register
HSS	Home Subscriber Servers
HTTP	Hypertext Transfer Protocol
I-CSCF	Interrogating Call Session Control Function
IARI	IMS Application Reference Identifier
IDE	Integrated Development Environment
IETF	Internet Engineering Task Force
IJCU	IMS Java Client Utility
IK	Integrity Key
IMAP	Internet Message Access Protocol
IMS	Internet Protocol Multimedia Subsystem
IM-SSF	IP Multimedia Service Switching Function
IP	Internet Protocol
IP	Internet Protocol
IPsec	Internet Protocol Security
ISIM	IP Multimedia Service Identity Module
J2ME	Java Micro Edition
Java SE 6	Java Standart Edition 6
JDBC	Java Database Connectivity
JSR	Java Specification Requests



MD5	Message-Digest algorithm 5
MGCF	Media Gateway Controller Function
MGW	Media Gateway
MIDP	Mobile Information Device Profile
MRF	Media Resource Functions
MRFC	Multimedia Resource Function Controller
MRFP	Multimedia Resource Function Processor
MySQL	My Structured Query Language
OSA	Open Service Access
OSA-SCS	Open Service Access - Service Capability Server
PCM	Pulse-code Modulation
P-CSCF	Proxy Call Session Control Function
PDF	Policy Decision Function
PLMN	Public Land Mobile Network
POP3	Post Office Protocol version 3
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RAND	Random Challenge
RES	Response
RMS	Record Management System
RTP	Real-time Transport Protocol
S-CSCF	Serving Call Session Control Function
SDK	Software development kit
SDP	Session Description Protocol
SDS	Service Development Studio
SGW	Signaling Gateway
SIP	Session Initiation Protocol
SIP AS	Session Initiation Protocol Application Server
SLF	Subscriber Location Functions
SMTP	Simple Mail Transfer Protocol
SQL	Structured Query Language
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
UML	Unified Modeling Language
UMTS	Universal Mobile Telecommunications System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WLAN	Wireless Local Area Network
XML	eXtensible Markup Language
XRES	Expected Result

# Obsah

<b>Seznam zkratek</b> .....	<b>8</b>
<b>Seznam obrázků</b> .....	<b>12</b>
<b>Seznam tabulek</b> .....	<b>13</b>
<b>Úvod</b> .....	<b>14</b>
<b>1. Architektura IMS</b> .....	<b>16</b>
1.1. Prvky architektury IMS.....	17
1.2. HSS (Home Subscriber Servers).....	17
1.3. SLF (Subscriber Location Functions).....	18
1.4. CSCF (Call/Session Control Functions).....	18
1.4.1. P-CSCF (Proxy Call Session Control Function).....	18
1.4.2. I-CSCF (Interrogating Call Session Control Function).....	18
1.4.3. S-CSCF (Serving Call Session Control Function).....	19
1.5. AS (Application Servers).....	19
1.6. MRF (Media Resource Functions).....	20
1.7. BGCF (Breakout Gateway Control Functions).....	21
1.8. PSTN/CS brány.....	21
<b>2. SIP protokol</b> .....	<b>23</b>
2.1. SIP adresy.....	24
2.2. Formát SIP zprávy.....	25
2.2.1. Záhlaví SIP zprávy.....	25
2.2.2. Pole hlaviček SIP zprávy.....	26
2.2.3. Tělo zprávy.....	27
2.3. SIP registrace v IMS.....	27
<b>3. Ericsson Service Development Studio (SDS)</b> .....	<b>29</b>
3.1. IMS Java Client Utility (IJCU).....	29
3.2. Java SIP Servlet.....	30
<b>4. Aplikace pro IMS</b> .....	<b>31</b>
4.1. Návrh systému.....	31
4.2. Použité programy a knihovny třetích stran.....	33
4.3. Testování aplikací.....	34
4.4. Databáze MySQL.....	36

4.4.1.	Struktura databáze.....	36
4.4.2.	Uživatelský přístup .....	37
4.5.	Klientská aplikace.....	39
4.5.1.	Požadavky klientské aplikace na mobilní zařízení .....	39
4.5.2.	Struktura klientské aplikace.....	40
4.5.3.	Popis ovládání klientské aplikace.....	43
4.6.	SIP Servlet aplikace .....	45
4.6.1.	Struktura SIP Servlet aplikace .....	45
4.7.	Aplikace pro správu databáze .....	48
4.7.1.	Struktura aplikace pro správu databáze .....	48
4.7.2.	Popis ovládání aplikace pro správu databáze .....	51
4.8.	HTTP Servlet aplikace.....	53
4.8.1.	Struktura HTTP Servlet aplikace.....	53
<b>5.</b>	<b>Analýza komunikace.....</b>	<b>54</b>
5.1.	Registrace klienta do IMS.....	54
5.2.	Komunikace mezi klientskou aplikací a SIP Servletem.....	55
5.2.1.	Rozbor záhlaví a pole hlaviček SIP zprávy .....	59
5.2.2.	Žádost o vygenerování nového hesla.....	60
5.2.3.	Stahování filmové ukázky z webového serveru.....	61
<b>6.</b>	<b>Závěr .....</b>	<b>63</b>
	<b>Literatura .....</b>	<b>65</b>
	<b>Seznam příloh.....</b>	<b>67</b>
A.	Nastavení SDS .....	68
B.	Ukázky klientské aplikace .....	74
C.	Ukázky aplikace pro správu databáze.....	79
D.	Obsah příloženého DVD.....	85

## Seznam obrázků

Obr. 1.1: Architektura IMS [1] .....	16
Obr. 1.2: Typy aplikačních serverů [1].....	20
Obr. 1.3: Propojení IMS a PSTN [1] .....	22
Obr. 2.1: Formát SIP zprávy [14] .....	25
Obr. 2.2: Registrace klienta do IMS [14].....	27
Obr. 4.1: Realizace systému pro rezervaci vstupenek .....	31
Obr. 4.2: Experimentální síť pro testování aplikací.....	35
Obr. 4.3: Diagram modelu databáze .....	37
Obr. 4.4: UML diagram klientské aplikace .....	42
Obr. 4.5: Klientská aplikace v emulátoru mobilního telefonu Sony Ericsson W810i....	43
Obr. 4.6: UML diagram SIP Servlet aplikace.....	47
Obr. 4.7: UML diagram aplikace pro správu databáze.....	50
Obr. 4.8: Aplikace pro správu databáze.....	51
Obr. 4.9: Nastavení aplikace pro správu databáze.....	52
Obr. 4.10: UML diagram HTTP Servlet aplikace .....	53
Obr. 5.1: Registrace klienta do IMS .....	54
Obr. 5.2: SIP zpráva REGISTER.....	55
Obr. 5.3: Záznamy Registrar serveru.....	55
Obr. 5.4: Zachycená komunikace mezi klientskou aplikací a SIP Servletem .....	56
Obr. 5.5: SIP zpráva MESSAGE určená pro přihlášení uživatele.....	57
Obr. 5.6: SIP zpráva MESSAGE s nabídkou filmů k promítání .....	58
Obr. 5.7: SIP zpráva MESSAGE s žádostí o vygenerování nového heslo .....	60
Obr. 5.8: Emailová zpráva s nově vygenerovaným heslem.....	61
Obr. 5.9: Stažení filmové ukázky .....	62
Obr. A.1: Nastavení adresy DNS serveru .....	68
Obr. A.2: Nastavení adres CSCF serverů .....	69
Obr. A.3: Nastavení autentizace pro registraci do IMS .....	70
Obr. A.4: Konfigurace DNS serveru .....	71
Obr. A.5: Přiřazení filtračních kritérií AS .....	72
Obr. A.6: První filtrační pravidlo – typ zprávy MESSAGE .....	72
Obr. A.7: Druhé filtrační pravidlo – vyvolání spojení.....	72

Obr. A.8: Definice uživatelských profilů.....	73
Obr. B.1: Hlavní menu klientské aplikace.....	74
Obr. B.2: Nastavení parametrů pro registraci do IMS sítě a adresy AS.....	74
Obr. B.3: Přihlášení uživatele do systému a žádost o zaslání nového hesla na email ....	75
Obr. B.4: Nová registrace uživatele.....	75
Obr. B.5: Hlavní menu aplikace a menu pro výběr filmu.....	76
Obr. B.6: Zobrazení informací o filmu a přehrávání ukázky k filmu.....	76
Obr. B.7: Výběr data a času promítání a volba počtu vstupenek k rezervaci.....	77
Obr. B.8: Výběr míst v promítacím sále a potvrzení rezervace vstupenek .....	77
Obr. B.9: Přiřazení identifikačního kódu rezervace a historie provedených rezervací ..	78
Obr. C.1: Sekce programu pro správu filmů.....	79
Obr. C.2: Sekce programu pro správu promítání.....	80
Obr. C.3: Sekce programu pro správu sálů.....	80
Obr. C.4: Sekce programu pro správu uživatelských účtu .....	81
Obr. C.5: Rezervované vstupenky vybraného uživatele.....	81
Obr. C.6: Sekce programu pro prodej lístků.....	82
Obr. C.7: Vymazání filmové ukázky z webového serveru.....	82
Obr. C.8: Nastavení aplikace pro správu databáze .....	83
Obr. C.9: Emailová zpráva informující o zrušeném promítání.....	84

## Seznam tabulek

Tab. 2.1: Metody SIP žádostí.....	23
Tab. 2.2: Třídy SIP odpovědí.....	24
Tab. 4.1: Síťové adresy serverů .....	35
Tab. 4.2: Síťové adresy aplikací .....	35
Tab. 4.3: Privilegia k účtu ticketservlet.....	38
Tab. 4.4: Privilegia k účtu ticketscontrol.....	38
Tab. 4.5: Požadavky klientské aplikace na standardy JSR.....	39
Tab. 4.6: Nastavení atributů pro přístup do MySQL databáze .....	46
Tab. 4.7: Parametry emailového účtu na serveru gmail.com.....	46
Tab. 5.1: Vyhrazené SIP adresy pro komunikaci .....	56

# Úvod

Díky globálnímu rozvoji Internetu, který umožňuje výměnu textových, zvukových a obrazových informací v reálném čase, se multimediální přenos dat stal perspektivním a atraktivním odvětvím moderních komunikačních technologií. V dnešní době se nejpoužívanější formou komunikace stala komunikace prostřednictvím mobilních telefonů v sítích GSM. Tato síť už ale není dimenzována pro rozvoj nových technologií. Z tohoto důvodu vznikl nový systém, který je pro takovou formu komunikace připraven. Tímto systémem je The Internet Protocol Multimedia Subsystem, známý též jako IMS.

IMS je platformově nezávislou architekturou určenou především pro konvergenci mobilních a pevných sítí. IMS je založena na velkém množství síťových protokolů, z nichž většina byla vyvinuta organizací IETF (Internet Engineering Task Force). Na jejich základě poskytuje IMS služby v reálném čase. Při vývoji IMS byly kladeny požadavky zejména na nezávislost přístupové sítě, tak aby bylo možné poskytovat multimediální služby přes všechny sítě založené na protokolu IP (např. WLAN, GPRS, DSL). Architektura IMS obsahuje propracované prostředky pro zajištění kvality služeb QoS, což vede k efektivnímu využití síťových prostředků pro přenos jak obecných dat, tak hlasu či videa.

První kapitola této práce je zaměřena na seznámení se s architekturou systému IMS. Jsou v ní popsány jednotlivé prvky tohoto systému, jejich funkce a vzájemná komunikace.

Druhá kapitola se zabývá popisem signalizačního protokolu SIP. Signalizace v IMS je založena právě na tomto protokolu. V práci je podrobně popsána funkce protokolu SIP, formát SIP zpráv a způsob registrace uživatelů do systému IMS.

Třetí kapitola slouží k obeznámením se s vývojovým prostředím SDS 4.1 FD1 od firmy Ericsson. Toto prostředí umožňuje tvorbu aplikací vyžívajících služeb IMS. SDS obsahuje vestavěný simulátor IMS sítě a nabízí podporu tvorby jak klientských, tak i serverových aplikací v rámci této architektury.

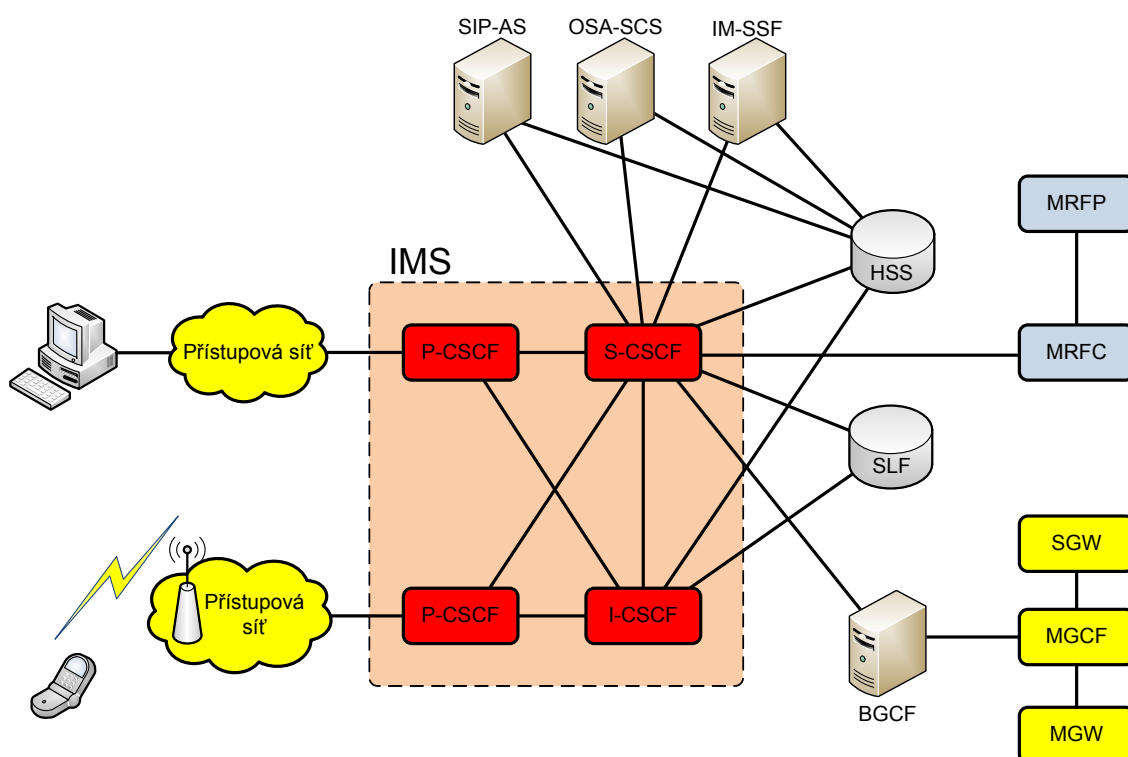
Čtvrtá kapitola se zabývá návrhem, popisem a testováním systému pro rezervaci vstupenek do kina prostřednictvím architektury IMS. Vytvořený projekt je realizován aplikacemi na bázi klient-server, kde klientská aplikace je určena mobilním zařízením s podporou technologie J2ME. Serverová část je realizována jako SIP Servlet aplikace,

která slouží k obsluze klientských požadavků a je jí umožněn přístup do databáze systému. Aby bylo možné nějakým způsobem uložené záznamy v databázi upravovat, byla pro tento účel vytvořena aplikace pro správu databáze založená na platformě Java SE 6.

Poslední pátá kapitola se zabývá analýzou registrace klienta do IMS sítě a komunikací mezi klientskou aplikací a SIP Servletem prostřednictvím signalizačního protokolu SIP. Sledování průběhu komunikace a zachytávání paketů bylo provedeno za pomoci analyzátoru síťového provozu Wireshark 1.2.2.

# 1. Architektura IMS

Důvodem, proč IMS (The Internet Protocol Multimedia Subsystem) vznikl, bylo sjednocení přenosu hlasových a datových služeb do jediné ucelené sítě. IMS tvoří architekturu přenosové sítě, která je dostupná globálně a nezávislá na použité přístupové technologii. Přístup do IMS je možný nejen z různých paketově orientovaných sítí (WLAN, GPRS, DSL, UMTS...), ale také ze sítí založených na spojování okruhů, jakými jsou například klasické telefonní systémy (PSTN). Cílem systému IMS je poskytnout uživatelům širokou sadu služeb pro přenos multimediálních dat. Signalizace v IMS je založena na signalizačním protokolu SIP. Samotný přenos dat sítí je paketový a je realizován síťovým protokolem IP. Architektura systému IMS je znázorněna na obr. 1.1 [1].



Obr. 1.1: Architektura IMS [1]



## 1.1. Prvky architektury IMS

IMS se skládá z několika různých a navzájem spolupracujících prvků. Každý typ prvku nemusí být v rámci systému IMS zastoupen pouze jedenkrát, ale může se vyskytovat i ve vyšším počtu. Mezi tyto prvky patří [1]:

- databáze uživatelů **HSS** (Home Subscriber Servers) a **SLF** (Subscriber Location Functions),
- skupina SIP serverů **CSCF** (Call/Session Control Functions),
- aplikační servery **AS** (Application Servers),
- sada funkcí pro multimediální zdroje **MRF** (Media Resource Functions),
- prvky pro přístup do okruhově spínaných sítí **BGCF** (Breakout Gateway Control Functions), **SGW** (Signaling Gateway), **MGCF** (Media Gateway Controller Function) a **MGW** (Media Gateway).

## 1.2. HSS (Home Subscriber Servers)

Databáze HSS je centrálním úložištěm uživatelských dat a služeb v IMS. Jsou v ní uložena přístupová práva uživatelů k poskytovaným službám IMS, údaje o identitě uživatelů a jejich profily, bezpečnostní informace sloužící k autentizaci a autorizaci uživatelů, informace o poloze, kde se uživatel právě nachází, nebo údaje o S-CSCF, který je uživateli právě přidělen. [1]

Identity uživatelů jsou v databázi HSS rozděleny na dva typy: soukromé (privátní) a veřejné. Soukromé identity jsou takové identity, které přiřadí uživateli operátor domovské sítě a jsou využívány pro účely registrace a autorizace uživatelů. Zatímco k veřejným identitám mají přístup i ostatní uživatelé, kteří je pak mohou využít pro navázání vzájemné komunikace. [1]

HSS je rozdělen na podskupiny HLR (Home Location Register) a AuC (Authentication Center). HLR je domovský lokalizační registr a nabízí podporu jak pro paketově, tak i okruhově spojované sítě. V AuC jsou uloženy tajné klíče, které jsou používány pro autentizaci mobilních uživatelů. Bezpečnostní údaje jsou také využívány k poskytnutí integrity a k šifrování komunikace mezi uživatelskými zařízeními a sítí IMS. [16]

### **1.3. SLF (Subscriber Location Functions)**

SLF je pouze jednoduchou databází, která váže adresu uživatele k adrese databáze HSS. Uzel (I-CSCF, P-CSCF nebo AS) pošle dotaz s adresou uživatele k SLF a jako odpověď dostane od SLF adresu HSS, ke které je uživatel vázán. Pokud se v síti IMS vyskytuje pouze jedna databáze HSS, není už databáze SLF potřebná. Naopak, jsou-li v síti alespoň dvě databáze HSS nebo více, je použití SLF nezbytné. SLF a HSS mezi sebou komunikují za pomoci protokolu Diameter. [1]

### **1.4. CSCF (Call/Session Control Functions)**

CSCF je základním prvkem architektury IMS. Jeho hlavním úkolem je poskytnutí signalizaci prostřednictvím SIP protokolu. V podstatě se jedná o skupinu SIP serverů. CSCF můžeme rozdělit na tři podtypy: I-CSCF, P-CSCF a S-CSCF. [1]

#### **1.4.1. P-CSCF (Proxy Call Session Control Function)**

P-CSCF je prvním místem kontaktu mezi uživatelským zařízením a IMS sítí. Jedná se o proxy server, který přeposílá žádosti a odpovědi od uživatelů do IMS sítě a naopak. O bezpečný přenos SIP signalizace se stará protokol IPsec (Internet Protocol security), díky kterému je možné detekovat případnou změnu zprávy od její původní podoby. P-CSCF také umožňuje kompresi a následnou dekompresi SIP zpráv pro snížení datového toku mezi uživatelem a IMS sítí (využití pro GPRS). Součástí P-CSCF může být také PDF (Policy Decision Function) pro přidělení šířky pásma přenosu konkrétnímu uživateli. Díky tomu je IMS síti umožněné řídit datové toky a poskytovat uživatelům služby s danou kvalitou (Quality of Service, QoS). [1]

#### **1.4.2. I-CSCF (Interrogating Call Session Control Function)**

Jedná se o SIP server, který slouží jako kontaktní bod ve vnitřní síti operátora. Jeho hlavním úkolem je přiřazení konkrétního S-CSCF serveru uživateli na základě jeho

registračních údajů. Jeho dalším úkolem je směrování SIP požadavků směrem k vhodnému cíli (typicky k některému S-CSCF). [1]

### 1.4.3. S-CSCF (Serving Call Session Control Function)

S-CSCF je SIP server, který slouží jako centrální uzel pro SIP signalizaci. V domovské síti operátora se může nacházet více S-CSCF prvků a každý z nich může plnit rozdílné funkce. Hlavní funkcí S-CSCF je zpracování registračních požadavků a uchování si záznamu o přiřazení IP adresy uživatelova zařízení se SIP adresou. Spojení mezi S-CSCF a databází HSS probíhá za pomoci protokolu Diameter a umožňuje ověření přístupu uživatele do IMS. S-CSCF slouží také ke zjišťování, k jakému AS mají být SIP zprávy dále směrovány. S-CSCF umožňuje také překlad SIP URI adres na telefonní čísla a naopak. [1]

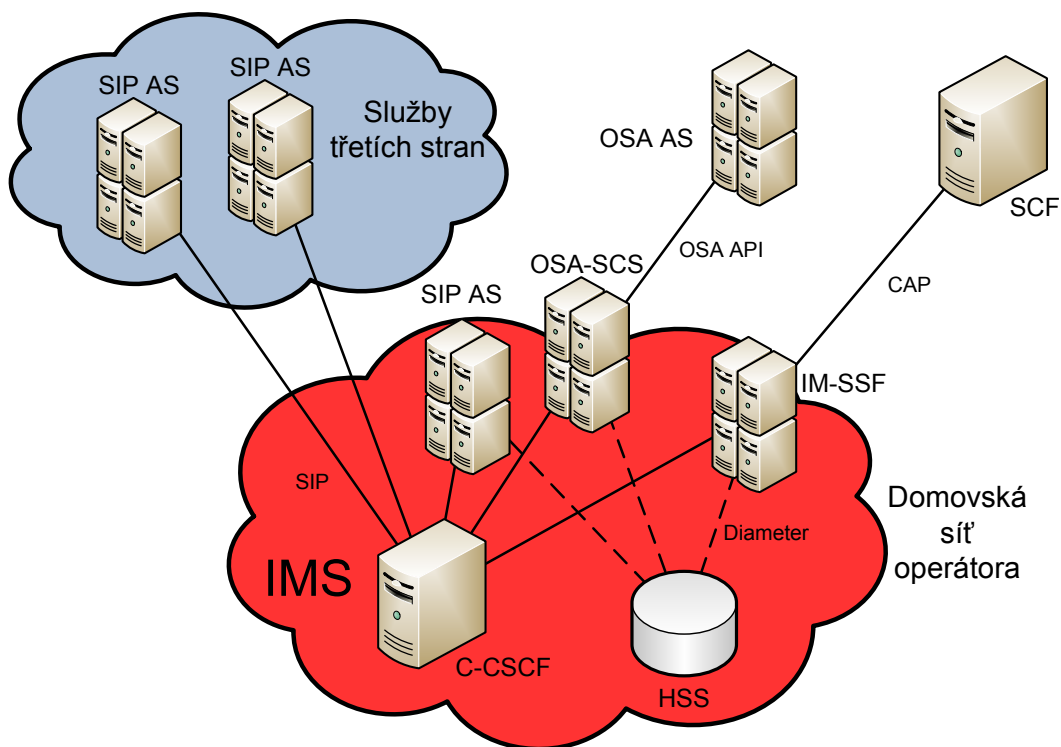
## 1.5. AS (Application Servers)

Aplikační servery se používají v IMS sítích jako zdroje nabízených služeb. Komunikace s prvky S-CSCF probíhá prostřednictvím signalizačního protokolu SIP. S databází HSS komunikují AS za pomoci protokolu Diameter. AS se může nacházet jak v domovské síti operátora, tak i mimo něj. V takovém případě ale neexistuje spojení mezi AS a HSS. Existují tři typy AS [1]:

- **SIP AS** (Session Initiation Protocol Application Server) – jedná se o implicitní aplikační server v IMS poskytující multimediální služby založené na protokole SIP,
- **OSA-SCS** (Open Service Access - Service Capability Server) - za pomoci standardizovaného rozhraní OSA API zprostředkovává komunikaci mezi sítí IMS a OSA aplikačním serverem, který je tak možné umístit i mimo síť operátora,
- **IM-SSF** (IP Multimedia Service Switching Function) – tento server dovoluje v IMS využít platformu CAMEL (Customized Applications for Mobile network Enhanced Logic) pro GSM služby inteligentní sítě. Propojením GSM inteligentní sítě a IMS, lze pak např. nabídnout službu, kde si uživatel bude moci

vybrat buď GSM nebo IMS koncové zařízení, na které mu budou směřovány příchozí volání. Další variantou by mohl být případ, kdy inteligentní síť sama rozhodne na základě toho, v které síti je uživatel přihlášen (GSM, IMS) o koncovém zařízení. Pro komunikaci mezi IM-SSF a řídicí logiky inteligentní sítě GSM (SCF – Service Switching Function) je využíván protokol CAP (CAMEL Application Part). [8]

Umístění různých typů v IMS jejich zapojení je znázorněno na obrázku 1.2 [1].



Obr. 1.2: Typy aplikačních serverů [1]

## 1.6. MRF (Media Resource Functions)

MRF slouží k poskytování multimediálních služeb v domovské síti operátora. Nabízí možnost oznamování informací o zdrojích přehrávání, slučování multimediálních proudů, převod mezi odlišnými formáty dat (kodeky), vytváření statistik a na jejich základě provádět analýzy.

MRF se skládá se dvou prvků [1]:

- **MRFC** (Multimedia Resource Function Controller) - řídí multimediální tok dat a poskytuje služby konference,
- **MRFP** (Multimedia Resource Function Processor) - poskytuje funkce pro práci s multimediálním obsahem jako je přehrávání, mixování multimediálních proudů, překódování nebo přizpůsobení obsahu přenosu.

## 1.7. BGCF (Breakout Gateway Control Functions)

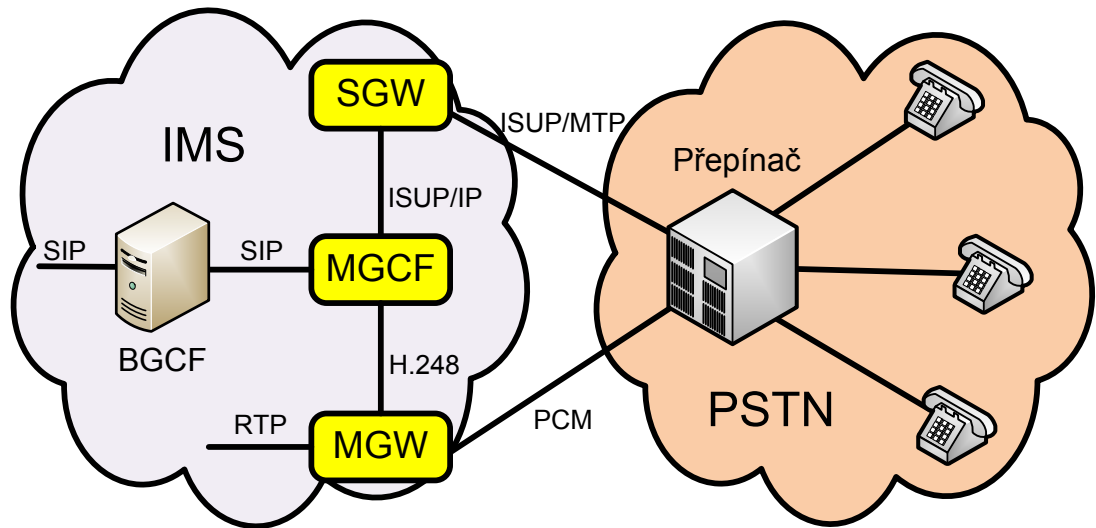
Jedná se o SIP server, který zajišťuje směrování na základě telefonních čísel. BGCF je v IMS zapotřebí pouze tehdy, vyskytuje-li se uživatel v některé síti založené na spojování okruhů (PSTN nebo PLMN). [1]

## 1.8. PSTN/CS brány

Tyto brány poskytují rozhraní umožňující přístup k síti IMS uživatelům ze standardní telefonní sítě (PSTN), nebo jiných okruhově spojovaných sítí CS. Funkce PSTN brány jsou rozděleny na tři části [1]:

- **SGW** (Signaling Gateway) – umožňuje převod signalizace na nižších vrstvách,
- **MGCF** (Media Gateway Controller Function) - provádí konverzi mezi SIP signalizací a signalizací použitou v CS síti,
- **MGW** (Media Gateway) – poskytuje konverzi mediálního obsahu přenášeného pomocí protokolu RTP do časových slotů PCM a naopak. Dále také umožňuje překódování obsahu. MGW najde uplatnění zejména v takovém případě, kdy IMS terminál kóduje hlas za pomocí kodeku AMR, ale PSTN terminál vyžaduje data kódovaná kodekem G.711.

Propojení IMS sítě s klasickou telefonní sítí (PSTN) vystihuje obrázek 1.3 [1].



Obr. 1.3: Propojení IMS a PSTN [1]

## 2. SIP protokol

Pro přenos signalizačních údajů v systémech IMS byl zvolen protokol SIP (Session Initiation Protocol). Jedná se o textově orientovaný protokol aplikační vrstvy sloužící k sestavení, modifikaci a ukončení spojení mezi dvěma nebo i více účastníky v IP sítích. Je schopný pracovat nad protokolem transportní vrstvy TCP, tak i nad protokolem UDP. Standardně protokol SIP využívá síťový port 5060 nebo 5061, který je určen pro zabezpečenou komunikaci prostřednictvím protokolu TLS. Funkce protokolu SIP je založena na bázi klient-server s podporou dvou typů zpráv – žádosti (request) a odpovědi (response).

SIP žádosti jsou formulované v textové podobě jako metody. Prvních šest metod v tabulce 2.1 [2] jsou definovány v základní specifikaci SIP RFC3261 [14]. Zbylé metody byly vytvořeny jako rozšiřující.

SIP odpovědi jsou reprezentovány jako trojčíselná čísla z rozsahu 100 až 699 rozdělená do šesti tříd. První číslice vždy udává, do jaké třídy odpověď patří. Zbylé dvě číslice pak upřesňují význam odpovědi. Odpovědi označené kódem 1xx jsou pouze dočasné, zatímco odpovědi od kódu 200 a výše jsou konečné. Třídy SIP odpovědí jsou uvedeny v tabulce 2.2 [2].

**Tab. 2.1: Metody SIP žádostí**

Metody	Popis
REGISTER	Registrace klienta
INVITE	Žádost o sestavení spojení
ACK	Potvrzení volajícího, že obdržel odpověď na svou zprávu INVITE
CANCEL	Ukončení nesestaveného spojení
BYE	Ukončení spojení
OPTIONS	Dotaz na možnosti a schopnosti serveru
MESSAGE	Přenos krátkých zpráv
NOTIFY	Informace o mimořádné události
SUBSCRIBE	Žádost o oznámení specifické události
UPDATE	Aktualizace informací o spojení
PUBLISH	Nahrání informací o uživateli na server
INFO	Přenos PSTN signalizace
PRACK	Potvrzení přijetí dočasné odpovědi (1xx)
REFER	Nařízení serveru o poslání žádosti

**Tab. 2.2: Třídy SIP odpovědí**

<b>Třída</b>	<b>Popis</b>
1XX	Informační zprávy - např. 100 Trying, 180 Ringing
2XX	Kladná odpověď / úspěšné vyřízení - např. 200 OK
3XX	Přesměrování, dotaz je třeba směřovat jinam - např. 302 Moved Temporarily, 305 Use Proxy
4XX	Chybný požadavek - např. 403 Forbidden
5XX	Chyba serveru - např. 500 Server Internal Error, 501 Not Implemented
6XX	Globální (zásadní) chyba - např. 606 Not Acceptable

## 2.1. SIP adresy

Koncoví uživatelé, aplikační servery, brány, ale i další zařízení využívající služeb SIP jsou identifikováni na základě SIP URI (Uniform Resource Identifikátor), neboli SIP adresy, které jsou podobné adresám emailovým. Kompletní formát SIP adresy je následující [13]:

**sip:**[uživatel[:heslo]@]**host**[:port][;uri-parametry][?hlavičky]

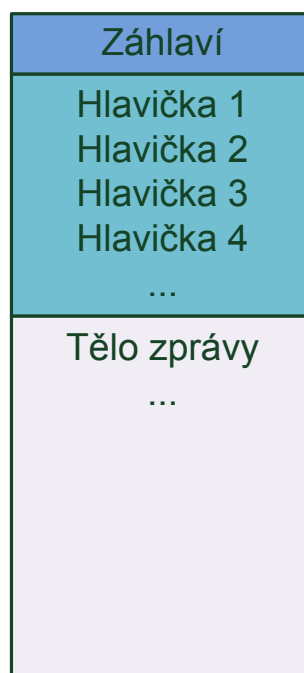
SIP adresa se skládá ze dvou částí. První část (před zavináčem) se nazývá lokální a slouží k identifikaci uživatele v rámci domény, ke které je registrován. Uživatel může být identifikován jak podle jména, tak také podle telefonního čísla. Pak však namísto předpony sip: mívá adresa předponu tel:. Druhá část (za zavináčem) udává jméno domény nebo její IP adresu. Adresa může obsahovat množství dalších nepovinných parametrů, jako je například číslo portu. Pokud není uvedeno, předpokládá se použití všeobecně známého portu 5060. Modře vyznačené části SIP adresy jsou povinné. Hranaté závorky mají význam nepovinnosti. [13]



## 2.2. Formát SIP zprávy

Formát SIP zprávy je textově orientovaný, tím pádem je její obsah jednoduše čitelný i pro člověka. Jak ukazuje obrázek 2.1 [14], formát SIP zprávy se skládá celkem ze tří částí [14]:

- **záhlaví** (start line),
- **pole hlaviček** (headers),
- **tělo zprávy** (body).



**Obr. 2.1: Formát SIP zprávy [14]**

### 2.2.1. Záhlaví SIP zprávy

Obsah záhlaví se liší podle typu SIP zprávy. Tedy, jedná-li se o SIP zprávu typu žádosti nebo odpovědi.

Záhlaví SIP zprávy typu žádost se skládá ze tří částí [14]:

- **Metoda** – určuje typ žádosti. Druhy metody byly popsány v tabulce 2.1,
- **URI adresa** – obsahuje SIP adresu toho, komu je žádost určena,
- **Verze protokolu** – určuje verzi SIP protokolu. V současnosti je nejběžnější používanou verzí SIP/2.0.

Také záhlaví SIP zprávy typu odpověď se skládá ze tří částí [14]:

- **Verze protokolu** – tak jako u SIP zprávy typu žádosti toto pole určuje verzi použitého protokolu SIP,
- **Třída** – jedná se o trojčíferné číslo určující druh odpovědi. Třídy SIP odpovědi byly popsány v tabulce 2.2,
- **Text odpovědi** – toto pole obsahuje krátký text k doplnění informací o vyslané SIP zprávě typu odpověď.

### 2.2.2. Pole hlaviček SIP zprávy

Pole hlaviček obsahuje data související ze SIP zprávou. Toto pole může informovat příjemce zprávy o tom, jakého typu je obsah těla zprávy. Například, bude-li hlavička *Content-Type* obsahovat hodnotu *application/sdp*, znamená to, že tělo zprávy ponese informaci o relaci ve formě protokolu SDP. Hlavička *Content-Length* udává velikost těla zprávy v bitech. Některé hlavičky jsou povinné a musí být obsaženy v každé SIP zprávě [14]:

- **To** – obsahuje jméno uživatele, pro koho je zpráva určena a jeho SIP adresu. Parametr *tag* je náhodný řetězec generovaný pro každý dialog.

To: "Pavel Otoupalík" <sip:pavel@ericsson.com>; tag=125675588062

- **From** - obsahuje jméno a SIP adresu odesílatele SIP zprávy.

From: "Pavel Otoupalík" <sip:pavel@ericsson.com>; tag=g1cb5ijh-1

- **Call-ID** – obsahuje jedinečný identifikátor pro danou relaci generovaný na straně klienta.

Call-ID: dsfsdf45gtfd5646fdg4fg45dfgs6@192.168.56.1

- **CSeq** – definuje sekvenční číslo žádosti a obsahuje také název SIP metody. Žádost i odpověď na ni mají stejné číslo Cseq.

CSeq: 923 INVITE

- **Via** - nese informace, jakou cestou zpráva prošla. Každý proxy server přidává k této hlavičce svoji adresu. Odpovědi potom putují stejnou cestou, jakou přišli žádosti na ně.

Via: SIP/2.0/udp 192.168.56.1;branch=dsfh6456hdf3456dfsgs4545gdg...

- **Max-Forwards** – používá se k nastavení maximálního počtu přeskoků zprávy. Standardně se počet přeskoků nastavuje na 70. Klesne-li hodnota k nule je vyslána SIP odpověď 483 (To Many Hops).

Max-forwards: 70

- **Contact** – jsou zde vypsány všechny adresy, na kterých je volající dostupný.

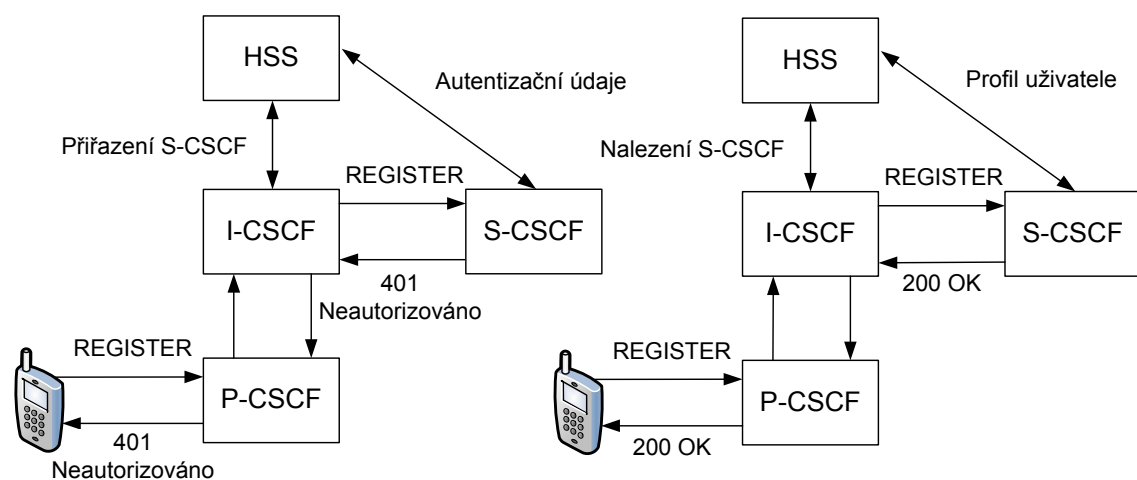
Contact: <192.168.56.101:51726;transport=tcp>

### 2.2.3. Tělo zprávy

Do těla zprávy může být vložena libovolná textová informace. Typickým příkladem použití je vložení zprávy ve formě protokolu SDP, který slouží k popisu přenosu dat mezi koncovými zařízeními. Dalším příkladem může být vyslání žádosti typu MESSAGE, kdy je do těla zprávy vložen obsah přenášené zprávy.

## 2.3. SIP registrace v IMS

Registrace je proces, který umožňuje koncovému uživatelskému zařízení přístup do IMS a povoluje mu využívání služeb v rámci této přenosové sítě. Pro ověření přístupu do IMS se na straně klienta využívá aplikace ISIM (IP Multimedia Service Identity Module), která slouží k výpočtu autentizačních a autorizačních údajů. Registrace se skládá ze dvou fází, jak je znázorněno na obrázku 2.2 [14].



Obr. 2.2: Registrace klienta do IMS [14]

V první fázi uživatelské zařízení nejdříve odešle SIP zprávu REGISTER směrem k proxy serveru P-CSCF, jehož adresu musí klient dopředu znát. REGISTER zpráva by měla obsahovat privátní a veřejnou identitu uživatele, URI jeho domácí domény a IP adresu terminálu, přes který se uživatel registruje. P-CSCF pak přepośle zprávu serveru I-CSCF, který kontaktuje HSS. Ten zjistí, zda už je uživateli přiřazen některý S-CSCF. Pokud ne, HSS mu některý podle požadovaných parametrů vybere. I-CSCF následně předá vybranému S-CSCF zprávu REGISTER. S-CSCF si uvědomuje, že uživatel ještě doposud nebyl zaregistrován, proto si z HSS stáhne data pro ověření uživatele, tzv. *Autentizační Vektor (AV)*, který se skládá ze čtyř položek [14]:

- **RAND** (random challenge) – náhodné číslo,
- **XRES** (expected result) – očekávaný výsledek,
- **AUTN** (network authentication token) – síťový autentizační symbol,
- **IK** (integrity key) – klíč integrity,
- **CK** (ciphering key) – šifrovací klíč.

S-CSCF následně odešle uživateli SIP zprávu třídy 401 (Unauthorized), do které přidá novou hlavičku pod názvem *WWW-Authenticate* s hodnotami RAND, AUTN, IK a CK. Hodnoty IK a CK jsou určeny proxy serveru P-CSCF, a proto budou ze zprávy tímto serverem odstraněny. [14]

Ve druhé fázi nejdříve uživatelské zařízení předá hodnoty RAND a AUTN aplikaci ISIM, která z těchto hodnot vypočítá za pomoci hašovacího algoritmu MD5 odpověď RES určenou serveru S-CSCF a také IK, který bude použit pro šifrovanou komunikaci mezi uživatelem a P-CSCF. Následně klient vyšle novou zprávu REGISTER společně s odpovědí RES, která bude vložena do hlavičky *Authorization* jako položka *response*. Tato zpráva putuje opět přes I-CSCF až k S-CSCF. Ten porovná přijatou odpověď RES s očekávanou hodnotou XRES. Pokud se zprávy shodují, S-CSCF si stáhne z databáze HSS uživatelův profil a uživateli pošle SIP odpověď 200 (OK), čímž je potvrzena jeho registrace v IMS. [14]

### **3. Ericsson Service Development Studio (SDS)**

Pro vývoj aplikací využívajících služeb IMS bylo zvoleno vývojové prostředí SDS ve verzi 4.1 FD1 od firmy Ericsson. Důvod k výběru tohoto vývojového prostředí byl ten, že SDS obsahuje vestavěný simulátor IMS sítě a nabízí programátorovi ucelenou sadu nástrojů pro tvorbu jak klientských, tak i serverových aplikací v rámci této architektury. SDS je založeno na platformě Eclipse IDE a programovacím jazyku Java verze 6. Součástí instalace SDS je i aplikační server GlassFish rozšířený o projekt SailFin, který přidává k aplikačnímu serveru SIP Servlet kontejner umožňující spouštět na severu SIP Servlet aplikace. SDS umožňuje vývoj klientských aplikací určených nejen osobním počítačům, ale také mobilním zařízením s operačním systémem Symbian nebo dnes běžným mobilním telefonům s podporovou technologií J2ME.

Po instalaci SDS je nutné provést nastavení vývojového prostředí podle postupu uvedeného v dokumentaci [4] a také doinstalovat některou z dostupných sad nástrojů pro vývoj aplikací pro mobilní zařízení zvaných Wireless Toolkit. Jednou z nich je například Sun Java Wireless Toolkit 2.5.2\_01 for CLDC. Jedná se o sadu nástrojů a technologií pro vývoj aplikací určených pro mobilní zařízení, v největší míře pak pro mobilní telefony.

#### **3.1. IMS Java Client Utility (IJCU)**

Pro vývoj IMS klientských aplikací, určených běžným mobilním telefonům bez podpory protokolu SIP, bylo v SDS navrženo API, které dostalo název IMS Java Client Utility (IJCU). Toto API implementuje do mobilního zařízení podporu protokolu SIP a umožňuje mu tak využívat služby v IMS. Díky tomuto API má mobilní zařízení možnost se zaregistrovat do IMS sítě a následně vytvořit relaci s jiným koncovým zařízením. Po vytvořeném spojení mohou uživatelé využívat podporovaných služeb IMS, jako je přenos textových zpráv, řeči, souborů, ale i streamovaného audia či videa. Postup vytvoření nového IJCU projektu je popsán v dokumentaci [3].

## 3.2. Java SIP Servlet

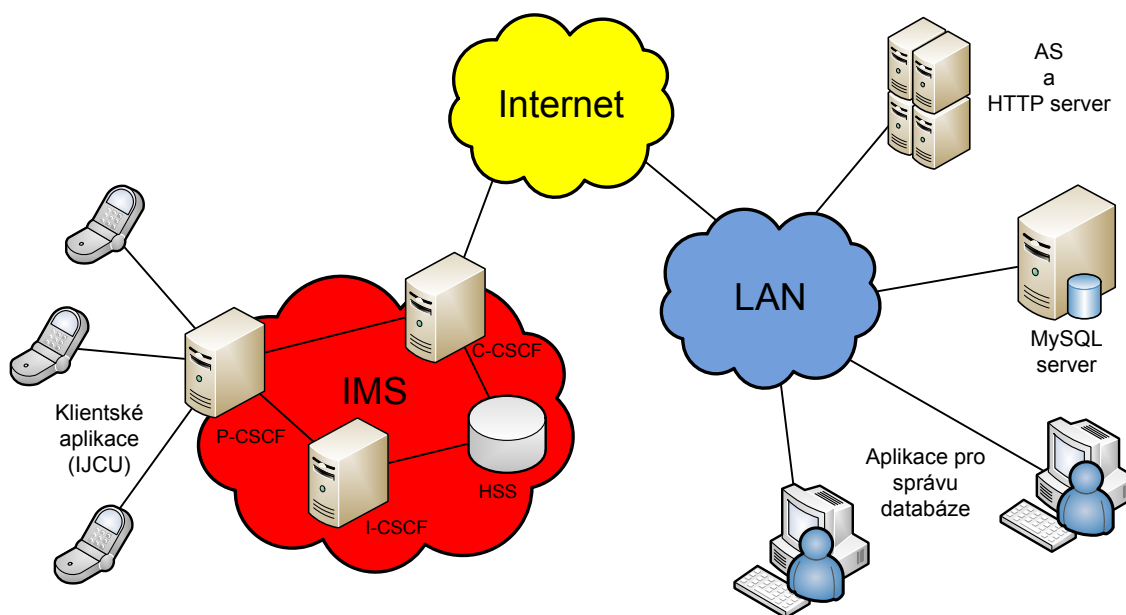
Java SIP Servlet je API určené k vývoji javovských aplikací založených na protokolu SIP běžících na straně serveru. Specifikace SIP Servlet byla vytvořena tak, aby se co nejvíce blížila specifikaci HTTP Servlet. Podobně, jako HTTP Servlet slouží k ošetření HTTP požadavků například vygenerováním nové webové stránky, tak SIP Servlet slouží ke zpracování SIP žádosti a vytvoření SIP odpovědi. SIP Servlety jsou na aplikačních serverech nejčastěji spouštěny jako reakce na příchozí požadavek od klienta. Na straně serveru může být vytvořeno i několik instancí SIP Servletu, jejichž počet je většinou dán počtem připojených klientů. V případě, kdy je nutné synchronizovat přístup více klientů ke sdíleným zdrojům, jako jsou například databáze, je lepší mít vytvořenou v daném okamžiku pouze jednu instanci SIP Servletu, aby nedocházelo ke kolizím. Bohužel toto řešení má za následek snížení výkonu serveru při větším množství přístupů, protože klienti musejí čekat ve frontě, než se na ně dostane řada. Postup vytvoření a spuštění SIP Servletu je popsán v dokumentaci [3].

## 4. Aplikace pro IMS

Úkolem této práce bylo vytvořit systém sestávající se z klientského modulu a z modulu pro přístup a správu databáze, který umožní uživatelům online rezervaci vstupenek prostřednictvím IMS sítě. Klientská aplikace by měla být určena pro mobilní terminály s podporou technologie J2ME.

### 4.1. Návrh systému

Aby měl vytvořený projekt z praktického hlediska nějaký smysl, bylo zvoleno, že navržený systém bude sloužit pro rezervaci vstupenek prostřednictvím IMS do smyšleného kina. Návrh, jak by takový systém mohl být realizován ve skutečnosti, ukazuje obrázek 4.1.



Obr. 4.1: Realizace systému pro rezervaci vstupenek

Uživatelé se mohou prostřednictvím mobilních terminálů s nainstalovanou klientskou aplikací registrovat do IMS sítě. Klientská aplikace je vytvořena jako IJCU projekt a je jí tedy možné spustit na běžném mobilním telefonu s podporou technologie J2ME. Po úspěšně provedené registraci pak klienti mohou navázat komunikaci

s aplikačním serverem (AS), který je umístěn mimo domovskou IMS síť operátora. Tím, že je aplikační server umístěn v síti poskytovatele služby pro rezervaci vstupenek, není mu umožněno přímo přistupovat do uživatelské databáze HSS. Tento způsob návrhu přináší nevýhodu v tom, že uživatel se musí nejdříve přihlásit do IMS sítě a následně i do systému poskytovatele služby pro rezervaci vstupenek. Bohužel, i kdyby se aplikační server nacházel uvnitř sítě IMS, nenabízí současná verze vývojového prostředí SDS, ve kterém byl projekt vytvořen, žádné programové rozhraní, jak by bylo možné databázi HSS spravovat. Z tohoto důvodu, aby se uživatelé mohli registrovat do IMS sítě a využívat jejich služeb, musejí být uživatelské účty v HSS databázi nadefinovány v perspektivě SDS pojmenované *Provisioning* (viz příloha A).

Komunikace mezi klientskou aplikací a aplikačním serverem probíhá prostřednictvím značkovacího jazyka XML (eXtensible Markup Language). Díky tomuto značkování je umožněno aplikacím si vyměňovat i složitější zprávy obsahující větší množství údajů. XML data jsou přenášena mezi aplikacemi v těle SIP zprávy typu MESSAGE.

V SIP kontejneru aplikačního serveru je vložen SIP Servlet, který slouží ke zpracování klientských požadavků a tvorby odpovědí na ně. SIP Servlet aplikaci je také umožněn přístup do MySQL (My Structured Query Language) databáze. V této databázi jsou uloženy údaje o registrovaných uživatelích, promítaných filmech, nabízených vstupenek, provedených rezervaci apod. Aby bylo možné tuto databázi nějakým způsobem obsluhovat, byla vytvořena aplikace pro správu databáze na bázi platformy Java SE 6 (Java Standard Edition 6).

Společně s aplikačním serverem musí být na tomtéž serveru nainstalován i HTTP server. Důvod je ten, že si klienti mohou právě z tohoto webového serveru stahovat ukázky filmů, které si mohou následně přehrát na svém mobilním zařízení. Jako HTTP server byl pro projekt zvolen softwarový webový server Apache. Na aplikačním serveru je kromě SIP Servlet aplikace umístěn v kontejneru také HTTP Servlet, který umožňuje aplikacím pro správu databáze nahrávat a mazat ze serveru filmové ukázky.



## 4.2. Použité programy a knihovny třetích stran

Při tvorbě projektu bylo využito dvou programových vývojových prostředí. Prvním z nich bylo již zmiňované SDS 4.1 FD1 od firmy Ericsson. Toto prostředí bylo využito při vývoji klientské IJCU aplikace, SIP Servlet aplikace a HTTP Servlet aplikace. SDS je k dispozici zdarma a je možné si jej stáhnout ze stránek firmy Ericsson na adrese: [http://www.ericsson.com/developer/sub/open/technologies/ims\\_poc/tools/sds\\_41](http://www.ericsson.com/developer/sub/open/technologies/ims_poc/tools/sds_41). Aplikace pro správu databáze byla vytvářena ve vývojovém prostředí NetBeans IDE 6.8. Důvodem k výběru tohoto prostředí byla snadnější tvorba grafického uživatelského rozhraní (GUI) než v prostředí Eclipse. NetBeans je možné zdarma stáhnout z internetu na adrese: <http://netbeans.org/downloads/index.html>.

Pro testování vytvořeného projektu bylo využito balíčku XAMPP 1.7.3, který obsahuje instalaci MySQL databáze ve verzi 5.1.41 a webového serveru Apache ve verzi 2.2.14. Balíček XAMPP je k dispozici zdarma na internetové adrese: <http://www.apachefriends.org/en/xampp-windows.html>.

Vytvořena klientská aplikace byla testována ve dvou emulátorech mobilního telefonu. Prvním z nich byl emulátor ze sady Wireless Toolkit 2.5.2\_01 for CLDC, který je volně stažitelný na internetové adrese: <http://java.sun.com/products/sjwtoolkit/download.html>. Druhým byl emulátor mobilního telefonu Sony Ericsson W810i ze sady Sony Ericsson SDK for the Java ME platform 2.5.0.6 dostupný na internetové adrese: <http://developer.sonyericsson.com/wportal/devworld/search-downloads/docstools/java/sdk?cc=gb&lc=en>.

Komunikace mezi klientskou aplikací a SIP Servletem probíhá prostřednictvím značkovacího jazyka XML. Jelikož není v platformě J2ME podpora XML standardně implementována, musela být do projektu klientské aplikace přidána knihovna kXML2 2.3.0. Tato knihovna patří do skupiny tzv. Pull parserů. Tyto parsery mají menší paměťové nároky, ale o to více zabírají místa ve výsledném balíku JAR. Jedná se o parser, který byl vytvořen přímo pro použití v MIDP (Mobile Information Device Profile) aplikacích. Knihovna kXML2 je licencována pod licencí BSD a je k dispozici zdarma ke stažení na internetové adrese: <http://sourceforge.net/projects/kxml/files/kxml2/>. [15]

Z důvodu, aby klientská aplikace neposílala heslo v otevřené formě, ale pouze jeho otisk získaný pomocí hašovací funkce MD5 (Message-Digest algorithm 5), byla

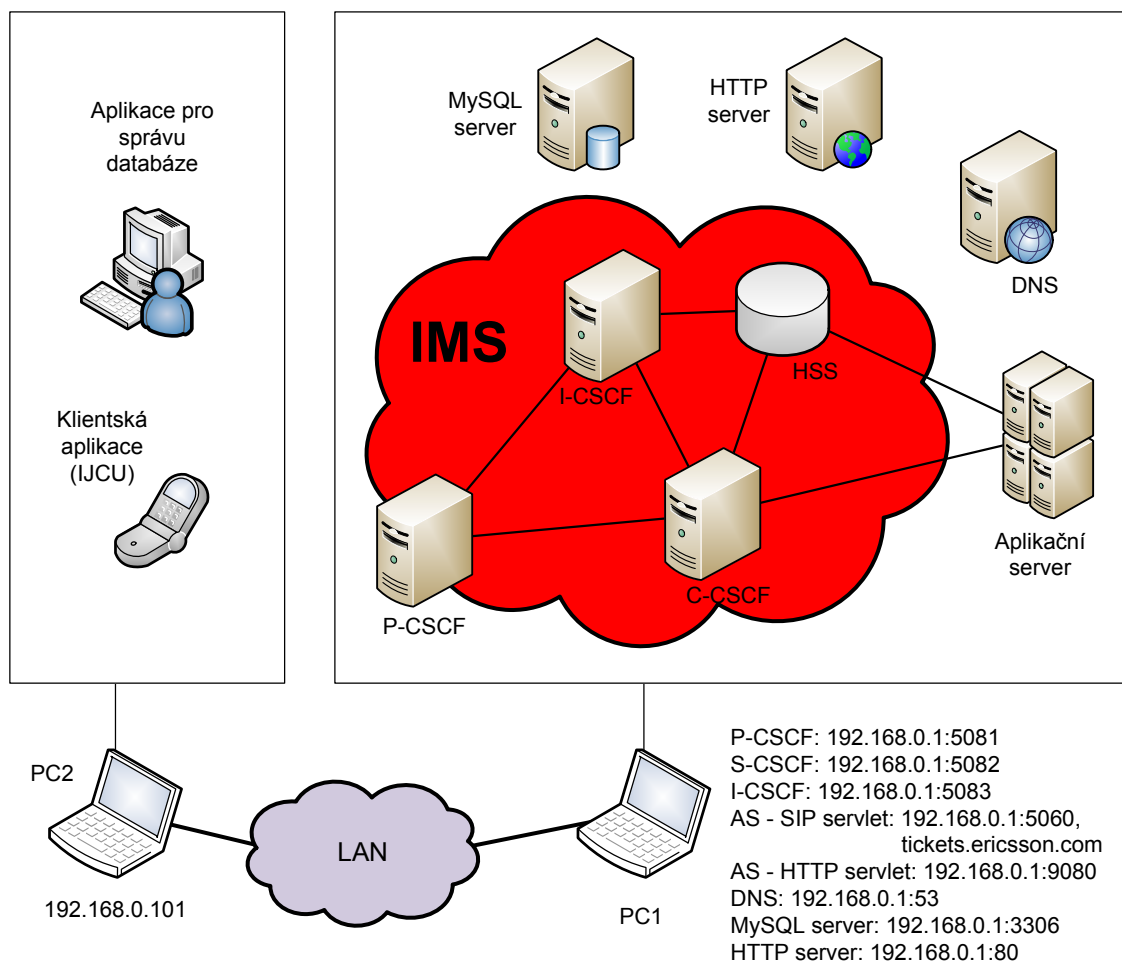
aplikace doplněna o aplikační programové rozhraní Bouncy Castle Crpyto 1.45. Tento balíček obsahuje mnoho kryptografických algoritmů, které nejsou standardně implementovány v J2ME. V projektu byla použita odlehčená verze tohoto API určená právě pro platformu J2ME. Bouncy Castle Crpyto 1.45 je zdarma dostupný na adrese: [http://www.bouncycastle.org/latest\\_releases.html](http://www.bouncycastle.org/latest_releases.html).

Aby byl SIP Servlet schopen přistupovat do databáze MySQL, byl k tomuto účelu použit JDBC ovladač MySQL Connector/J ve verzi 5.1.11. Tento ovladač je určený právě pro aplikace vyvíjené v Javě a je možné si ho zdarma stáhnout na internetové adrese: <http://dev.mysql.com/downloads/connector/j/>. Po stáhnutí ovladače je nutné jej zkopírovat do adresáře, kde je nainstalován aplikační server GlassFish. V případě výchozí instalace vývojového prostředí SDS 4.1 FD1 se jedná o adresář C:\Ericsson\SDS4.1FD1\Simulators\sailfin\lib.

Pro možnost posílání emailových zpráv klientům, bylo v projektu využito aplikačního programového rozhraní JavaMail 1.4.3. V implementaci tohoto API jsou obsažené nejpoužívanější protokoly umožňující přenos emailových zpráv, jako jsou např. SMTP, POP3 a IMAP. JavaMail API je zdarma k dispozici na internetové adrese: <http://java.sun.com/products/javamail/downloads/index.html>.

### 4.3. Testování aplikací

Pro možnost otestování projektu byla vytvořena experimentální síť (viz obr. 4.2) sestávající se ze dvou počítačů s operačním systémem Windows XP Professional SP3 a s nainstalovaným vývojovým prostředím SDS 4.1 FD1. Na prvním počítači (PC1) byl dále nainstalován balíček XAMPP ve verzi 1.7.3 s webovým serverem Apache verze 2.2.14 a s MySQL databází verze 5.1.41. Na PC1 byla spuštěna simulace IMS sítě, DNS server a aplikační server GlassFish, který umožňuje běh jak HTTP Servlet aplikací, tak i za pomoci projektu SailFin SIP Servlet aplikací. Na druhém počítači (PC2) byla spuštěna aplikace pro správu databáze a v emulátoru mobilního telefonu Sony Ericsson W810i také klientská aplikace. Nastavené síťové adresy jednotlivých síťových prvků, serverů a aplikací v experimentální síti jsou uvedené v tabulkách 4.1 a 4.2. Nastavení vývojového prostředí SDS a konfigurace simulované IMS je popsána v příloze A.



**Obr. 4.2: Experimentální síť pro testování aplikací**

**Tab. 4.1: Síťové adresy serverů**

Server	IP adresa	Port	Doménové jméno
P-CSCF	192.168.0.1	5081	
S-CSCF	192.168.0.1	5082	
I-CSCF	192.168.0.1	5083	
AS – SIP Servlet	192.168.0.1	5060	tickets.ericsson.com
AS – HTTP Servlet	192.168.0.1	9080	
DNS server	192.168.0.1	53	
MySQL server	192.168.0.1	3306	
HTTP server	192.168.0.1	80	

**Tab. 4.2: Síťové adresy aplikací**

Aplikace	IP adresa
Klientská aplikace (IJCUI)	192.168.0.101
Aplikace pro správu databáze	192.168.0.101

## 4.4. Databáze MySQL

Aby bylo možné uchovávat informace o registrovaných uživatelích, promítaných filmech, nabízených vstupenkách apod., byla do projektu zařazena MySQL databáze pod názvem *tickets* skládající se celkem ze sedmy tabulek (*film*, *promitani*, *sal*, *listky*, *rezervace*, *zakaznik*, *id\_kod*). Databáze byla navržena za pomoci [12, 17]. Diagram modelu databáze je znázorněn na obrázku 4.3.

### 4.4.1. Struktura databáze

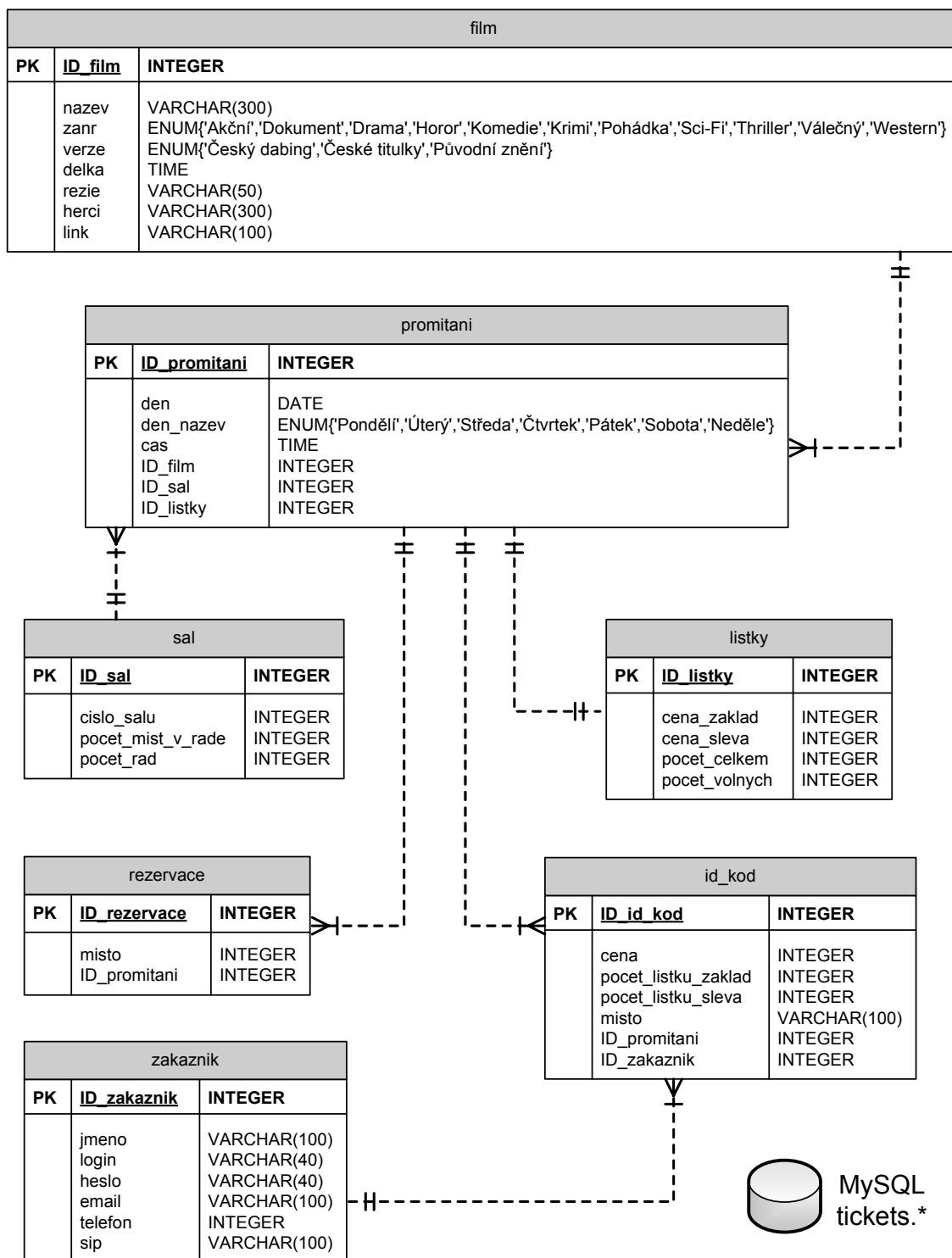
Tabulka *film* slouží k uchování informací o promítaných filmech, jako jsou název, žánr, verze, délka, jméno režiséra a herců, kteří hrají ve filmu. Poslední položka *link* obsahuje odkaz na webovou adresu, odkud je možné si stáhnout filmovou ukázkou.

Tabulka *promitani*, která je s tabulkou *film* v relaci 1:N, což znamená, že jednomu filmu může být přiřazeno několik promítání, uchovává informace o promítání konkrétního filmu. Těmito informacemi jsou zejména den a čas promítání. Tato tabulka je dále v relaci 1:1 s tabulkou *listky* a v relaci 1:N s tabulkou *rezervace*, *sal* a *id\_kod*.

V tabulce *sal* jsou ukládány informace o čísle promítacího sálu a jeho kapacitě, ze které je pak odvozen počet nabízených vstupenek k rezervaci. Při každém novém vytvořeném promítání jsou do tabulky *listky* přidány nové záznamy o cenách vstupenek ve dvou cenových skupinách (dospělý a dítě/student). Dále jsou v této tabulce uchovávány informace o celkovém počtu vstupenek určených k rezervaci a také počet vstupenek, kolik jich je ještě možné rezervovat. Tabulka *rezervace* obsahuje záznamy informující o obsazenosti konkrétního místa v promítacím sále pro dané promítání.

Do tabulky *zakaznik* jsou ukládány základní informace o uživatelích využívajících služeb systému pro rezervaci vstupenek. Těmito informacemi jsou jméno uživatele, přihlašovací údaje do systému, emailová adresa, telefonní číslo a SIP adresa.

Při každé nové rezervaci vstupenek je uživateli přiřazen identifikační kód. Tento kód je uložen v tabulce *id\_kod*, která je s tabulkou *zakaznik* v relaci 1:N, což znamená, že každý uživatel může provést několik rezervací. V tabulce *id\_kod* jsou zaznamenány informace o rezervaci jako je celková cena vstupenek, počet lístků a vybraná místa v promítacím sále. Přiřazení uskutečněné rezervace s promítaným filmem je umožněno díky relaci 1:N mezi tabulkami *promitani* a *id\_kod*.



Obr. 4.3: Diagram modelu databáze

#### 4.4.2. Uživatelský přístup

Aby mohla SIP Servlet aplikace a aplikace pro správu databáze přistupovat k záznamům databáze, byly k tomuto účelu vytvořeny dva uživatelské účty

*ticketsservlet* (heslo: *ticketsservlet*) a *ticketscontrol* (heslo: *ticketscontrol*). Pomocí těchto účtů je možno přistupovat s omezenými právy pouze do databáze *tickets* z jakéhokoliv počítače v síti. Nastavení privilegií k těmto účtům je uvedeno v tabulkách 4.3 a 4.4.

**Tab. 4.3: Privilegia k účtu *ticketsservlet***

Příkaz	Tabulky
INSERT	tickets.zakaznik, tickets.rezervace, tickets.id_kod
SELECT	tickets.*
UPDATE	tickets.zakaznik, tickets.listky

**Tab. 4.4: Privilegia k účtu *ticketscontrol***

Příkaz	Tabulky
INSERT	tickets.*
SELECT	tickets.*
UPDATE	tickets.*
DELETE	tickets.*

Význam jednotlivých příkazů je následující:

- INSERT (umožňuje uživateli přidávat do databáze nové záznamy),
- SELECT (umožňuje uživateli spouštět výběrové dotazy),
- UPDATE (umožňuje uživateli upravovat existující záznamy v databázi),
- DELETE (umožňuje uživateli odstraňovat záznamy z tabulek).

Vytvoření těchto uživatelských účtů a nastavení jejich práv bylo provedeno zadáním příkazů jazyka SQL:

```
GRANT INSERT ON tickets.id_kod TO ticketsservlet@%' IDENTIFIED BY 'ticketsservlet';
GRANT INSERT ON tickets.rezervace TO ticketsservlet@%' IDENTIFIED BY 'ticketsservlet';
GRANT INSERT ON tickets.zakaznik TO ticketsservlet@%' IDENTIFIED BY 'ticketsservlet';
GRANT UPDATE ON tickets.zakaznik TO ticketsservlet@%' IDENTIFIED BY 'ticketsservlet';
GRANT UPDATE ON tickets.listky TO ticketsservlet@%' IDENTIFIED BY 'ticketsservlet';
GRANT SELECT ON tickets.* TO ticketsservlet@%' IDENTIFIED BY 'ticketsservlet';
GRANT INSERT, SELECT, UPDATE, DELETE ON tickets.* TO ticketscontrol@%' IDENTIFIED BY 'ticketscontrol';
```

## 4.5. Klientská aplikace

Klientská aplikace je vytvořena jako IJCU projekt a je ji tedy možné spustit na běžném mobilním telefonu s podporou technologie J2ME. Tato aplikace umožňuje uživateli zaregistrovat se do IMS sítě. Po této události může mobilní zařízení navázat spojení se SIP Servletem a pomocí SIP zpráv navzájem komunikovat. Tímto způsobem je umožněn uživateli přístup do systému pro rezervaci vstupenek. Výměna datových informací mezi klientskou aplikací a SIP Servletem probíhá prostřednictvím značkovacího jazyka XML. Při tvorbě klientské aplikace bylo využito [7, 11, 18,19].

### 4.5.1. Požadavky klientské aplikace na mobilní zařízení

Klientská aplikace byla navržena tak, aby mohla být spuštěna na mobilních zařízeních s podporou standardu MIDP 2.0/CLDC 1.1. Konfigurace CLDC (Connected Limited Device Configuration) byla speciálně navržena tak, aby splňovala požadavky Java platformy, která běží na zařízeních s omezenou pamětí, výkonem procesoru a grafikou. Obsahuje knihovny podobné těm, jaké jsou známé ze standardní Javy, doplněné o knihovny specifické pro platformu Java ME. Profil MIDP přidává k CLDC další funkce umožňující například práci se sítí, tvorbu grafického uživatelské rozhraní nebo podporu přehrávání multimedií. [9]

Jelikož byla klientská aplikace vyvíjena jako IJCU projekt, jsou na mobilní zařízení kladeny další požadavky, zejména na standardy JSR (Java Specification Requests). Potřebné standardy JSR, které musí mobilní zařízení podporovat, jsou uvedeny v dokumentaci k IJCU API (viz tab. 4.5).

**Tab. 4.5: Požadavky klientské aplikace na standardy JSR**

Číslo	Název
75	FileConnection API
118	MIDP 2.0
135	Mobile Media API
139	CLDC 1.1
172	Web Services

#### 4.5.2. Struktura klientské aplikace

Na obrázku 4.4 je zobrazen UML diagram vytvořené klientské aplikace pro rezervaci vstupenek prostřednictvím IMS sítě. Třída *TicketsClient* je potomkem třídy *MIDlet*, díky čemuž je možné spustit aplikaci na mobilním zařízení podporující platformu J2ME. Tato třída slouží zejména ke komunikaci aplikace s uživatelem. Umožňuje zobrazení přijatých zpráv na displeji mobilního zařízení a přijímání příkazů od uživatele pomocí klávesnice.

Prostřednictvím třídy *RMSProcess* a jejich metod *loadSetting* a *saveSetting* může aplikace přistupovat do RMS databáze mobilního telefonu, do které si aplikace ukládá konfigurační parametry pro přístup do IMS sítě a také adresu aplikačního serveru.

Velice důležitou třídou je třída *XmlProcess*. Jak již bylo napsáno dříve, klientská aplikace komunikuje se SIP Servletem prostřednictvím značkovacího jazyka XML. Právě tato třída umožňuje zpracovávat data, která jsou reprezentována ve formě této struktury. Jelikož jsou zprávy posílány jako textové řetězce, musí být data ze struktury XML převedena právě na tento datový typ, což zajišťuje metoda *xmlToString*. Tato metoda tedy nejdříve vytvoří z dat pro odeslání XML dokument v DOM struktuře, který převede na textový řetězec tak, aby mohla být data odeslána v těle SIP zprávy typu MESSAGE. Metoda *stringToXmlToVector* slouží naopak pro získání dat z přijaté zprávy, kdy je potřeba data ve formě textového řetězce nejdříve převést na XML dokument v DOM struktuře a následně data od sebe oddělit na základě XML elementů. Metody *findNextName* a *findNextValue* jsou privátními metodami třídy a slouží pro vyhledání elementů v XML dokumentu a k získání jejich hodnot.

Třídy *RegisterThread* a *SendMessageThread* implementují rozhraní *Runnable*, což umožňuje jejich běh jako samostatná vlákna. První z těchto tříd slouží pro poslání SIP zprávy typu REGISTER, která zajistí přihlášení klienta do IMS sítě. Druhá třída slouží k vytvoření a poslání SIP zprávy typu MESSAGE. Do vstupních parametrů konstruktoru této třídy je potřeba zadat SIP adresu příjemce zprávy a obsah zprávy, která se má odeslat.

Pro příjem SIP zpráv je v projektu určena třída *CoreServiceClientAdapter*, která implementuje rozhraní *CoreServiceListener*. Z tohoto rozhraní je v aplikaci využita pouze jedna implementovaná metoda pro příjem SIP zpráv typu MESSAGE, a to *pageMessageReceived*.

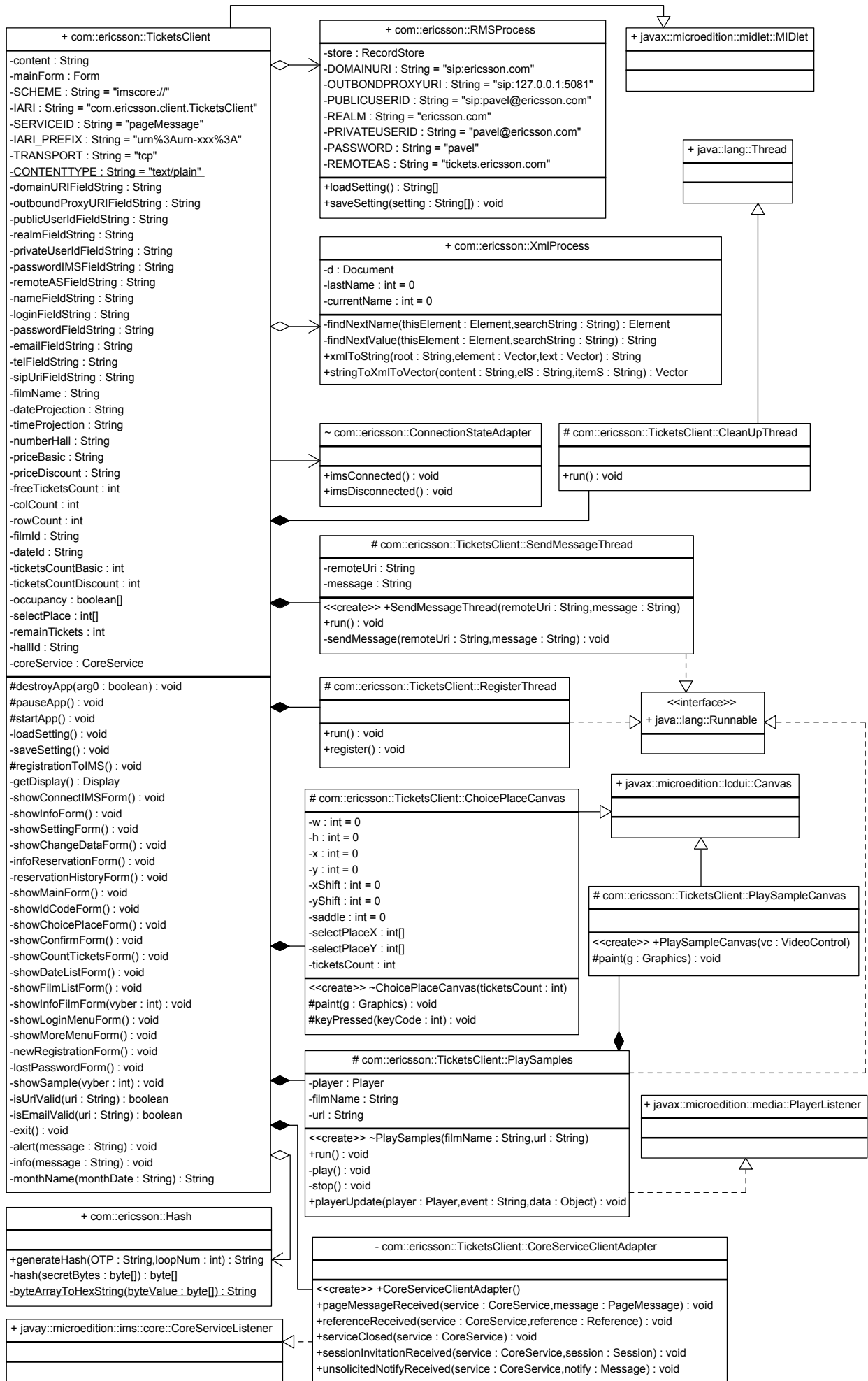


Třída *ChoicePlaceCanvas* je vytvořena jako potomek třídy *Canvas* a slouží k vykreslení struktury promítacího sálu na displeji mobilního zařízení. Uživatel si při rezervaci vstupenek díky tomu může vybrat, která místa v sále si chce rezervovat.

Pro stažení a přehrání filmových ukázek byla vytvořena třída *PlaySmample* implementující rozhraní *Runnable*, což umožní její běh jako samostatné vlákno. Ukázky jsou stahovány z webového serveru přes protokol HTTP. Do vstupních parametrů konstruktoru této třídy je potřeba zadat URL adresu, odkud se ukázka má stáhnout. Třída *PlaySampleCanvas*, která dědí vlastnosti a metody z třídy *Canvas*, pak zajistí zobrazení přehrávaného videa na displeji mobilního zařízení.

Třída *Hash* slouží k vytvoření otisku textového řetězce za pomoci hašovací funkce MD5. Aby uživatelovo heslo nebylo přenášeno v jeho otevřené formě, je prostřednictvím metody *generateHash* proveden výpočet otisku uživatelova hesla. Díky tomuto způsobu zabezpečení je zabráněno zachycení hesla útočníkem při odposlechu komunikace. Privátní metoda *byteArrayToHexString* zajistí převod výsledku hašovací funkce z pole bajtů na textový řetězec obsahující číslo v hexadecimální tvaru.

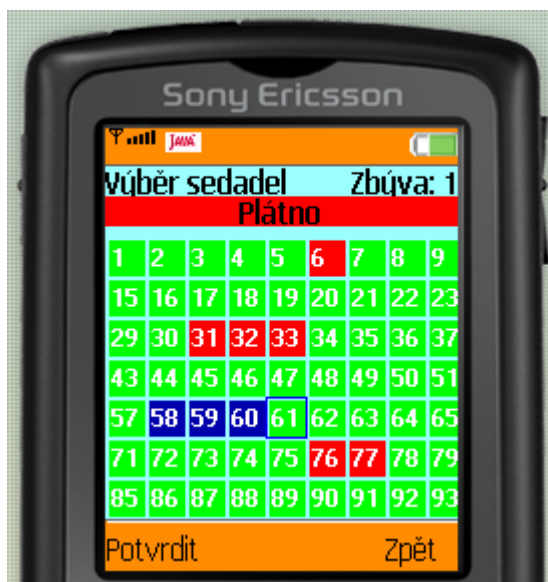
Poslední z vytvořených tříd je třída *CleanUpThread*, která je potomkem třídy *Thread*. Tato třída slouží k vytvoření a poslání SIP zprávy typu REGISTER pro odhlášení uživatele z IMS sítě.



Obr. 4.4: UML diagram klientské aplikace

### 4.5.3. Popis ovládání klientské aplikace

Na obrázku 4.5 je zobrazena ukázka klientské aplikace, která byla spuštěná na emulátoru mobilního telefonu Sony Ericsson W810i.



Obr. 4.5: Klientská aplikace v emulátoru mobilního telefonu Sony Ericsson W810i

Po spuštění klientské aplikace se na displeji mobilní zařízení objeví hlavní menu. Pod volbou *Nastavení* lze nakonfigurovat jednotlivé parametry pro registraci do IMS sítě. V nastavení je také možné zadat adresu aplikačního serveru, se kterým aplikace po registraci bude komunikovat. Nastavené parametry jsou uloženy v RMS databázi mobilního telefonu a při dalším spuštění aplikace jsou z této databáze opět načteny. Tím je docíleno, že uživatel nemusí po každém spuštění aplikace znovu nastavovat konfigurační parametry. Nastavené parametry při výchozím stavu aplikace jsou následující:

- **URI domácí domény** sip.ericsson.com,
- **URI P-CSCF** sip:192.168.0.1:5081,
- **Veřejná identita** sip:pavel@ericsson.com,
- **Oblast autorizace** ericsson.com,
- **Privátní identita** pavel@ericsson.com,
- **Heslo** pavel,
- **Aplikační server** tickets@ericsson.com.

Po volbě položky *Přihlášení* z hlavního menu dojde k registraci uživatele do IMS sítě. Proběhla-li registrace úspěšně, objeví se na displeji mobilního zařízení formulář pro přihlášení do systému pro rezervaci vstupenek. Pokud uživatel doposud nemá v systému vytvořen účet, může si z menu zobrazeného po stisku tlačítka více a výběrem položky *Nová registrace*, vytvořit nový uživatelský účet. Pro registraci do systému pro rezervaci vstupenek musí uživatel vyplnit tyto údaje:

- Jméno a příjmení,
- Přihlašovací jméno,
- Heslo,
- Opakovat heslo,
- Email,
- Telefonní číslo,
- SIP adresa (nelze měnit) – adresa je nastavena na SIP adresu použitou při registraci do IMS.

Pokud by se stalo, že by uživatel zapomněl svoje heslo pro přístup do systému, může si ho po zadání emailové adresy uvedené při registraci a svého přihlašovacího jména, po volbě položky *Zapomenuté heslo* z menu, nechat poslat emailem.

Po úspěšném přihlášení do systému pro rezervaci vstupenek, se na displeji mobilního zařízení zobrazí menu s položkami umožňující provedení nové rezervace vstupenek (*Vybrat film*), zobrazení již provedených rezervací (*Rezervované lístky*) nebo provést změnu registračních údajů (*Změna registračních údajů*). Vybere-li uživatel možnost nové rezervace vstupenek, zobrazí se mu na displeji mobilního zařízení seznam filmů. Projekt byl navržen tak, aby se klientovi poslala pouze nabídka filmů, které budou promítány od následujícího dne prováděné rezervace. Klient si může po výběru položky *Info* z menu nechat zobrazit informaci o vybraném filmu, nebo po výběru položky *Ukázka* nechat přehrát krátkou ukázkou k filmu.

Po výběru filmu si zákazník může na následující obrazovce zvolit z nabídky den a čas promítání. Po tomto kroku následuje volba počtu rezervovaných vstupenek. Při jedné rezervaci si uživatel může zamluvit maximálně 8 vstupenek (4 ks pro dospělé, 4 ks pro děti nebo studenty). Po potvrzení vybraného počtu vstupenek je uživateli umožněno vybrat si místa k sezení v promítacím sále. Na konci prováděné rezervace vstupenek se na displeji mobilního zařízení zobrazí shrnutí o této rezervaci. Tím, že ji

uživatel potvrdí, je proces rezervace vstupenek dokončen. Rezervaci je přiřazen identifikační kód, na základě jehož znalosti si uživatel může rezervované vstupenky vyzvednout. Ukázky klientské aplikace jsou uvedeny v příloze B.

## 4.6. SIP Servlet aplikace

Úkolem SIP Servlet aplikace je zprostředkovat přístup do MySQL databáze vzdáleným klientským aplikacím, s kterými SIP Servlet komunikuje skrz IMS síť. SIP Servlet je uložen v kontejneru aplikačního serveru Glassfish doplněný o projekt SailFin. Pokud dojde na port 5060 aplikačního serveru SIP zpráva typu žádosti, pak je tato zpráva předána SIP Servlet aplikaci, která tuto správu určitým způsobem zpracuje a vytvoří na ní odpověď. Při tvorbě SIP Servlet aplikace bylo použito [5, 7, 12].

### 4.6.1. Struktura SIP Servlet aplikace

UML diagram SIP Servlet aplikace je zobrazen na obrázku 4.6. Třída *TicketsServlet* je potomkem třídy *SipServlet*, což umožňuje aplikaci spustit na aplikačním serveru. Před vložením aplikace do kontejneru aplikačního serveru, je nezbytně nutné mít nastaveny atributy této třídy *DOMAIN* a *CSCF* na správné hodnoty. Pro vytvořenou experimentální síť byl doménový název aplikačního serveru [tickets.ericsson.com](http://tickets.ericsson.com) (může být použita i adresa 192.168.0.1:5060) a adresa *CSCF* serveru byla nastavena na 192.168.0.1:5082.

Důležitou metodou třídy *TicketsServlet* je metoda *doMessage*. Tato metoda přijímá SIP požadavky od klientských aplikací a na základě jména obsaženého v SIP URI příjemce rozhodne, které další metodě předá zprávu ke zpracování. K vytvoření odchozích SIP zpráv slouží metoda *createResponce*. Prvním vstupním parametrem této metody je příchozí SIP zpráva, ze které je získána SIP adresa klienta. Dalšími parametry této metody jsou SIP adresa odesilatele a obsah zprávy, který bude přenesen v těle vytvořené SIP zprávy. K vytvoření obsahu odchozí zprávy slouží zbylé metody třídy.

Přístup SIP Servletu do MySQL databáze zajišťuje třída *MySQLProcess*. Aby bylo možné přistupovat do vytvořené databáze *tickets*, musí být atributům této třídy přiřazeny hodnoty podle tabulky 4.6. Spojení aplikace s databází se vytváří již

v konstruktoru této třídy. Metoda *getConnection* předává vytvořené spojení nadřazenému objektu, který instanci této třídy vytvořil. Metoda *closeConnection* slouží k ukončení spojení s databází.

**Tab. 4.6: Nastavení atributů pro přístup do MySQL databáze**

Atribut	Hodnota
NAMEDB	tickets
DOMAIN	192.168.0.1
PORT	3306
USER	ticketsservlet
PASS	ticketsservlet

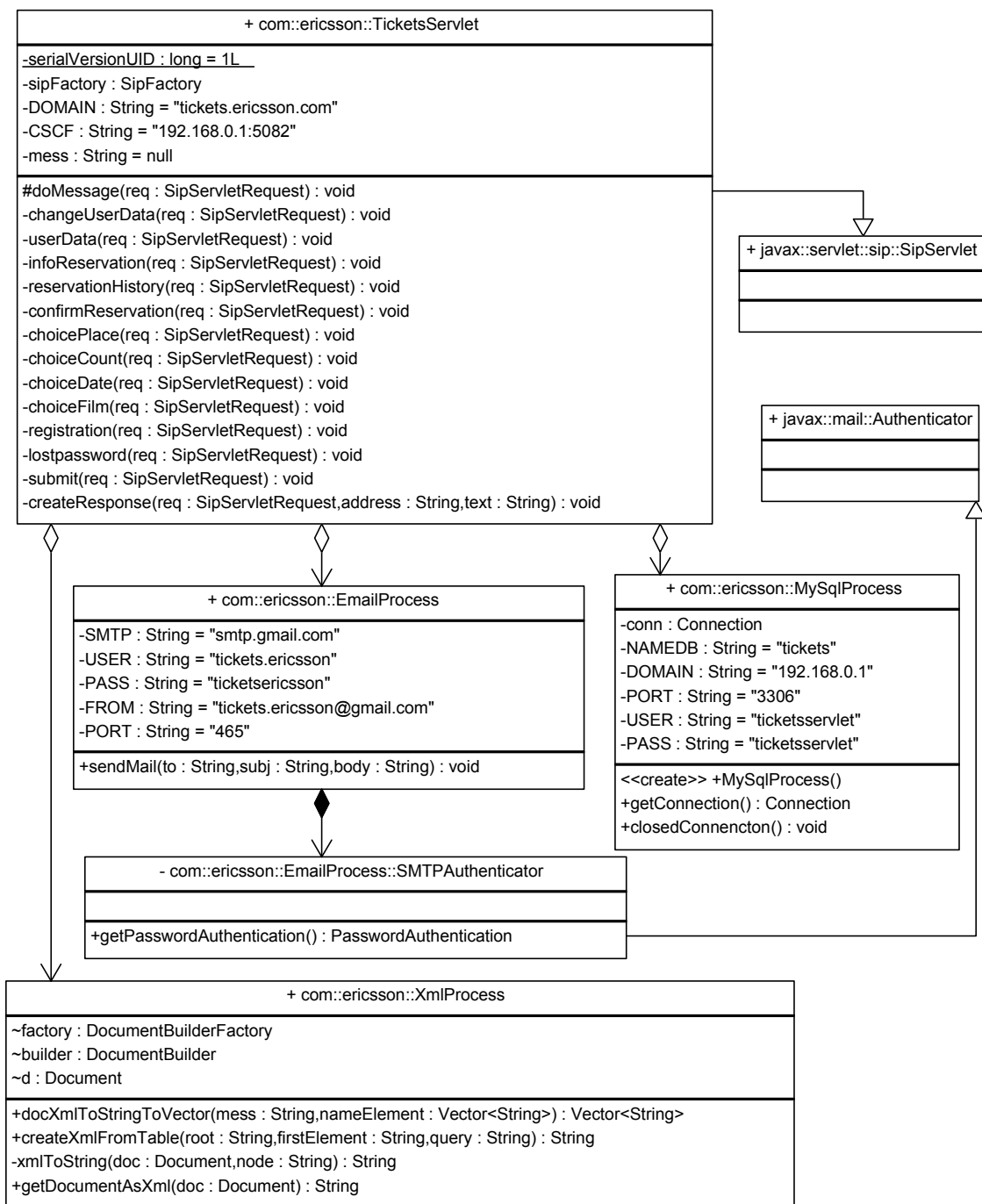
K posílání emailových zpráv slouží třída *EmailProcess*. Zprávy se zasílají na emailové schránky klientů v případě zapomenutého hesla. Pro možnost posílání emailových zpráv byl vytvořen účet na poštovním serveru [gmail.com](http://gmail.com). Aby byla aplikace schopná zasílat klientům emailové zprávy, musí být hodnoty atributů třídy nastaveny podle tabulky 4.7. Pro odeslání emailové zprávy slouží metoda *sendMail*. Její vstupními parametry jsou emailová adresa příjemce, předmět zprávy a obsah zprávy. Třída *SMTPAuthenticator* slouží k autorizaci připojení k SMTP serveru.

**Tab. 4.7: Parametry emailového účtu na serveru gmail.com**

Atribut	Hodnota
SMTP	smtp.gmail.com
USER	tickets.ericsson
PASS	ticketsericsson
FROM	tickets.ericsson@gmail.com
PORT	465

Jak již bylo napsáno dříve, SIP Servlet aplikace komunikuje s klientskou aplikací pomocí značkovacího jazyka XML. Pro zpracování XML dokument v DOM struktuře byla vytvořena třída *XMLProcess*. Metoda *docXMLToStringToVector* slouží k získání dat z příchozí zprávy. Metoda má dva vstupní parametry. Těmi jsou přijatá zpráva a vektor hledaných elementů v XML dokumentu. Metoda pracuje tak, že je zpráva nejdříve z formátu textového řetězce převedena na XML dokument v DOM struktuře. V ní jsou pak nalezeny hodnoty hledaných elementů zprávy.

Metoda *createXmlFromTable* je určena pro vytvoření XML dokumentu ve formě textového řetězce. Vstupním parametrem této metody je název kořenového elementu a dotaz na databázi. Z výsledku dotazu na databázi je vytvořen obsah zprávy ve formě XML dokumentu, který bude poslán klientovi. Pro převod z celé DOM struktury XML dokumentu na textový řetězec byla vytvořena metoda *getDocumentAsXml*. Pro převod pouze jednoho záznamu z XML dokumentu na textový řetězec slouží metoda *xmlToString*.



**Obr. 4.6: UML diagram SIP Servlet aplikace**

## 4.7. Aplikace pro správu databáze

Aby mohla být práce nad databází uživatelsky příznivá a snadná, byla vytvořena aplikace na bázi platformy Java SE 6, která je určena právě pro účel správy databáze systému pro rezervaci vstupenek. Díky této aplikaci může obsluha snadno přidávat nebo odebírat filmy, dobu jejich promítání a na základě znalosti identifikační čísla rezervací provádět výdej lístků. Při tvorbě této aplikace bylo čerpáno z [7, 12].

### 4.7.1. Struktura aplikace pro správu databáze

Na obrázku 4.7 je zobrazen UML diagram aplikace pro správu databáze. O grafické rozhraní a komunikaci aplikace s uživatelem se stará třída *TicketsControlGUI*, která je potomkem třídy *JFrame*. Aby byla třída spustitelná, implementuje metodu *main*. Z důvodu přehlednosti, nebyly do UML diagramu této třídy zahrnuty grafické komponenty. K inicializaci grafický komponent slouží metoda *initComponents*. Pro zobrazení načtených záznamů z databáze byly vytvořeny modely tabulek pojmenované jako *mdFilm* (filmy), *mdUser* (zákazníci), *mdHall* (sály) a *mdProjection* (promítání), které jsou všechny deklarovány jako typ *DefaultTableModel*. Obsah tabulek je plněn dvourozměrným polem textových řetězců pojmenovaným jako *v*. K naplnění pole *v* daty z databáze byly vytvořeny metody *showHall*, *showFilms*, *showUsers*, *showProjection*.

Hlavní třídou aplikace je třída *TicketsControl*, která umožňuje načítání, ukládání a mazání záznamů z databáze. Načtené záznamy jsou ukládány do dvourozměrného pole textových řetězců *v*. Pro snadnější práci nad databází jsou do dalšího pole pojmenované jako *vID* ukládány primární klíče tabulek. Pro možnost předávání načtených záznamů z databáze jiným objektům aplikace slouží metoda *getV*. Metody *timeConversion* a *dateConversion* byly vytvořeny pouze za účelem konverze formátu datumu a času. Například datum 2010-04-16 načtené z databáze je převedeno na formát 16. dubna 2010. Zbylé metody třídy slouží k načítání, ukládání a mazání záznamů z databáze tak, jak plyne z jejich názvu.

Přístup aplikace do MySQL databáze zajišťuje třída *MySQLProcess*. Důležitými metodami této třídy jsou metody *executeQuery* a *executeUpdate*, které jsou určeny k vykonávání SQL příkazů nad databází. První z těchto metod umožňuje použití příkazu



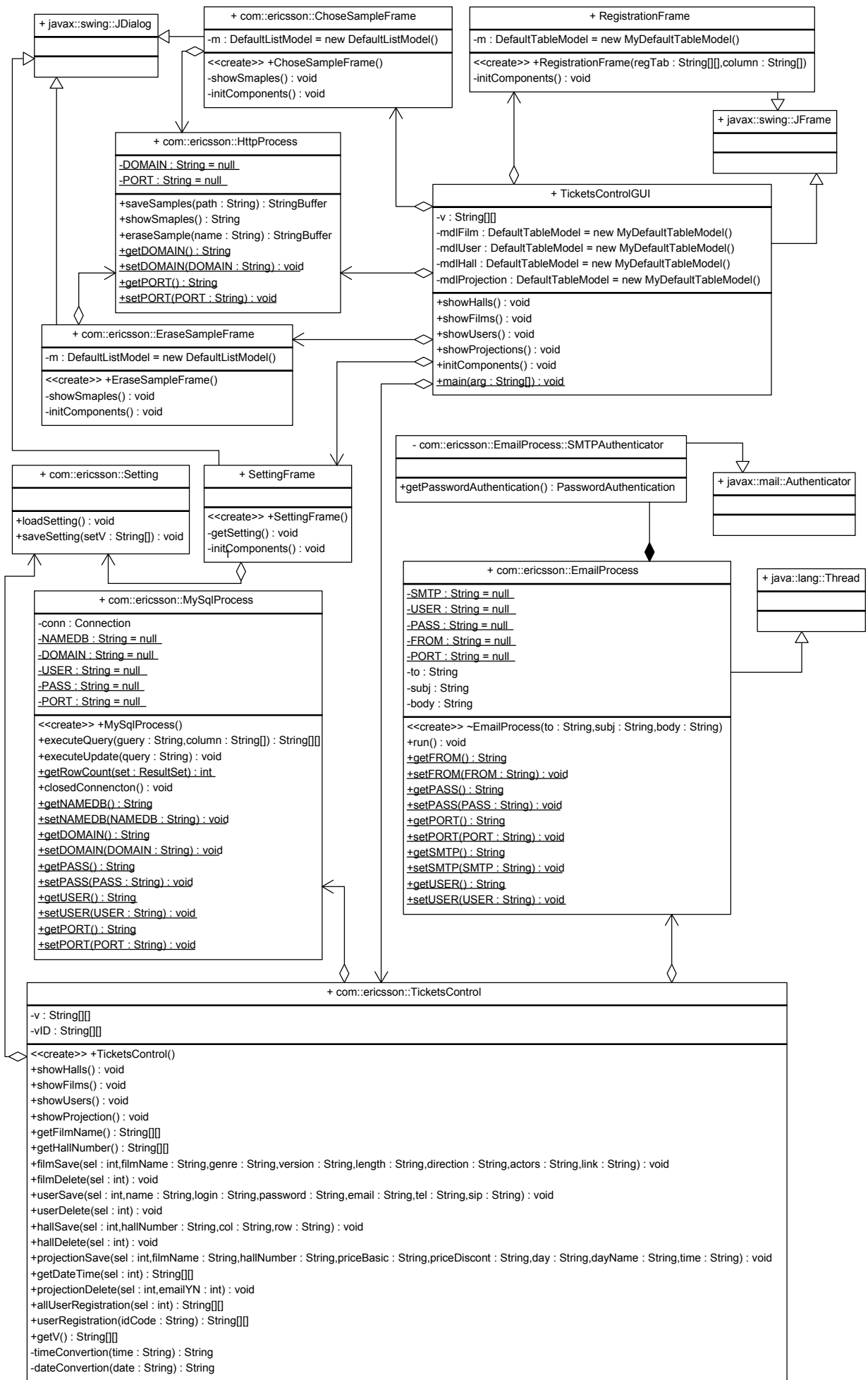
SELECT. Druhá metoda slouží k zadávání příkazů INSERT, UPDATE a DELETE. Metoda *closedConnection* slouží k ukončení spojení s databází. Další metody slouží k nastavení konfiguračních parametrů pro přístup do databáze. Tyto parametry jsou uloženy v attributech *NAMEDB* (název databáze), *DOMAIN* (jméno domény/IP adresa), *USER* (název uživatelského účtu), *PASS* (heslo) a *PORT* (číslo portu).

K posílání emailových zpráv slouží třída *EmailProcess*. Zprávy je možné v aplikaci zaslat na emailové schránky klientů v případě zrušeného promítání. Poslání emailu je zajištěno už v konstruktoru této třídy, kterému je potřeba předat tyto parametry: *to* (adresa příjemce), *subj* (předmět zprávy) a *body* (obsah zprávy). Třída *EmailProcess* je potomkem třídy *Thread*, čímž je zajištěno, že běží jako samostatné vlákno. Další metody třídy slouží k nastavení konfiguračních parametrů uživatelského účtu vytvořeného na poštovním serveru [gmail.com](http://gmail.com). Tyto parametry jsou uloženy v attributech *SMTP* (adresa SMTP serveru), *USER* (název uživatelského účtu), *PASS* (heslo), *FROM* (adresa odesilatele) a *PORT* (číslo portu). Třída *SMTPAuthenticator* slouží k autorizaci připojení k SMTP serveru.

Aby mohla aplikace ukládat, načítat a mazat filmové ukázky z webového serveru, byla vytvořena k tomuto účelu třída *HttpProcess*. Metoda *saveSample* umožňuje nahrání ukázky na webový server. Jediným vstupním parametrem této metody je parametr *path*, který obsahuje cestu k ukázce umístěnou na lokálním disku. Metoda *showSamples* slouží k vytvoření seznamu, který obsahuje názvy filmových ukázek uložených na webovém serveru. Metoda *eraseSample* vymaže ukázku z webového serveru. Zbylé metody slouží k nastavení doménového jména a čísla portu webového serveru. Pro uživatelský přístup k funkcím této třídy byly vytvořeny třídy *ChoseSampleFrame* a *EraseSampleFrame*, které jsou potomky třídy *JDialog*.

Aplikace si ukládá všechny nastavené konfigurační parametry do textového souboru *setting.txt*, který se nachází ve stejném adresáři jako samotná aplikace. Třída *Setting* zajišťuje právě ukládání a načítání těchto konfiguračních parametrů do tohoto souboru. Pomocí metody *loadSetting* jsou hodnoty ze souboru načítány a použitím metody *saveSetting* naopak do souboru ukládány. Aby mohla obsluha programu provádět nastavení konfiguračních parametrů přímo v puštěné aplikaci prostřednictvím jednoduchého formuláře, byla pro tento účel vytvořena třída *SettingFrame*, která je potomkem třídy *JDialog*.

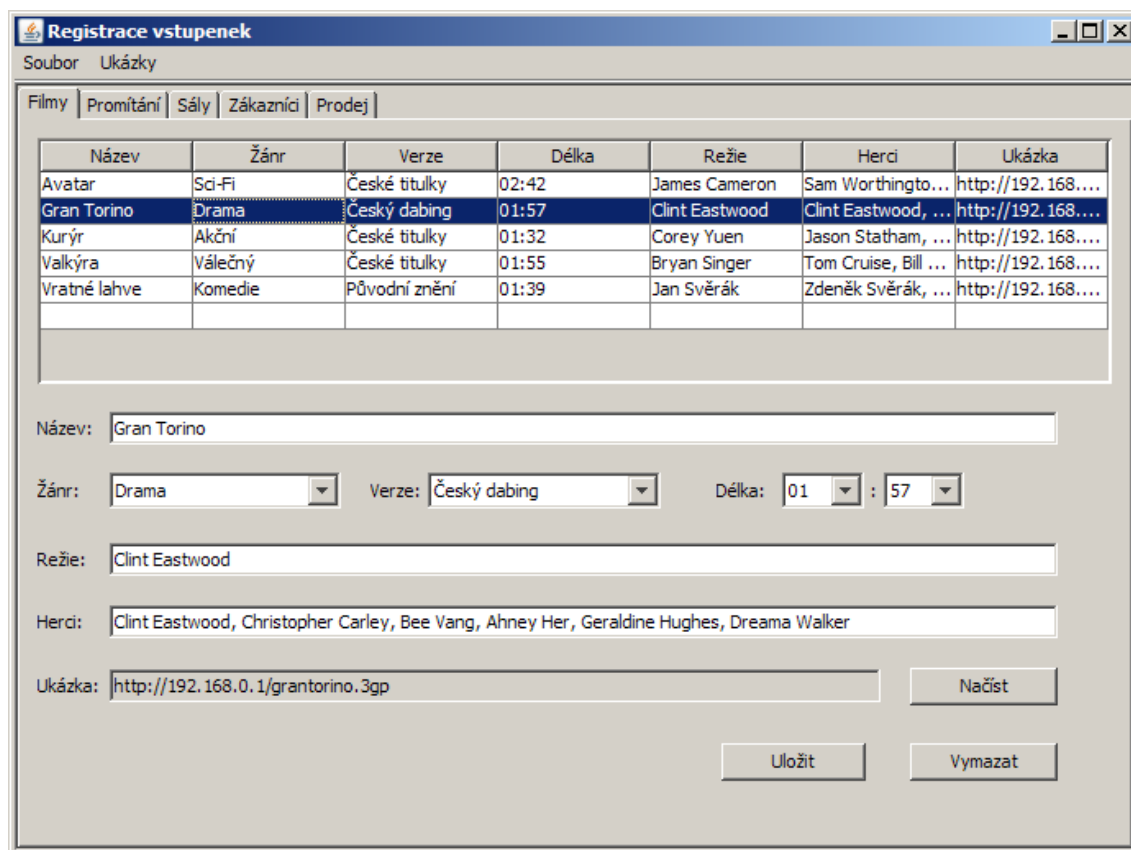
Poslední vytvořena třída *RegistrationFrame* slouží k prohlížení již provedených rezervací vstupenek zvoleného uživateli.



Obr. 4.7: UML diagram aplikace pro správu databáze

#### 4.7.2. Popis ovládání aplikace pro správu databáze

Ukázka aplikace pro správu databáze je zobrazena na obrázku 4.8. Aplikace byla spuštěna v prostředí operačního systému Windows XP SP3.



Obr. 4.8: Aplikace pro správu databáze

Pomocí záložek umístěných v horní části okna spuštěné aplikace se dá přepínat mezi jednotlivými částmi programu. V sekci programu, která se zobrazí po výběru první záložky pojmenované jako *Filmy*, lze přidávat, upravovat nebo mazat filmy z MySQL databáze. V další sekci programu, pod názvem záložky *Promítání*, lze k vytvořeným filmům přiřadit dobu jejich promítání, cenu lístků a číslo promítacího sálu. Jednotlivé promítací sály lze vytvořit v části programu, která je k dispozici pod záložkou *Sály*. Ke každému sálu lze přiřadit počet míst určujících kapacitu sálu. Po výběru záložky *Zákazníci*, si lze nechat v okně programu do tabulky vypsát seznam uživatelských účtů registrovaných zákazníků i jejich doposud provedené rezervace vstupenek. V poslední sekci programu, pod záložkou *Prodej*, lze na základě vložení identifikačního čísla rezervace provést výdej vstupenek.

Z hlavního menu, po výběru položky *Soubor -> Nastavení*, lze provést konfiguraci programu (viz obr. 4.9). Pomocí tohoto formuláře mohou být nastaveny parametry určené pro přístup aplikace do databáze MySQL, k webovému serveru a poštovnímu serveru. Prostřednictvím položky *Ukázky* z hlavního menu mohou být nahrávány a mazány filmové ukázky z webového serveru. Podrobnější popis ovládání aplikace a více ukázek se nacházejí v příloze C.

**Nastavení**

Databáze:

Adresa: 192.168.0.1

Port: 3306

Název: tickets

Uživatel: ticketscontrol

Heslo: ticketscontrol

Webový server:

Adresa: 192.168.0.1

Port: 9080

Email:

Adresa: tickets.ericsson@gmail.com

Uživatel: tickets.ericsson

Heslo: ticketsericsson

SMTP: smtp.gmail.com

Port: 465

Zpět OK

**Obr. 4.9: Nastavení aplikace pro správu databáze**

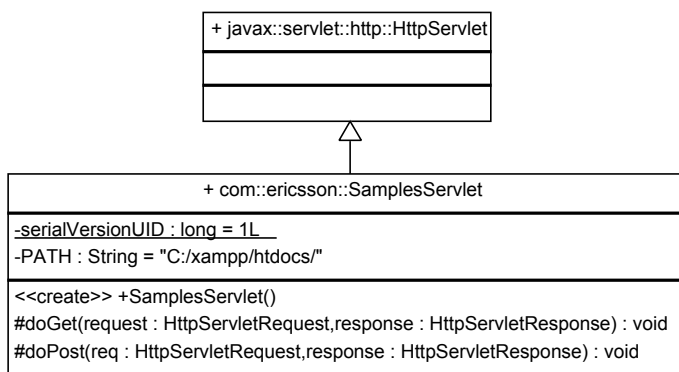
## 4.8. HTTP Servlet aplikace

Tato aplikace byla do projektu doplněna z toho důvodu, aby filmové ukázky, které si mohou uživatelé stáhnout a přehrát na svém mobilním zařízení, bylo možné na webový server nahrát nebo je naopak vymazat. HTTP Servlet je stejně jako SIP Servlet uložen v kontejneru aplikačního serveru Glassfish. Pokud dojde na port 9080 aplikačního serveru požadavek protokolu HTTP, pak je tato zpráva předána HTTP Servlet aplikaci, která tuto správu určitým způsobem zpracuje a vytvoří na ní odpověď.

### 4.8.1. Struktura HTTP Servlet aplikace

Jak je vidět na obrázku 4.10, na kterém je zobrazen UML diagram HTTP Servlet aplikace, jedná se o jednoduchou aplikaci vytvořenou pouze z jedné třídy *SamplesServlet*, která je potomkem třídy *HttpServlet*, což umožňuje aplikaci spustit na aplikačním serveru. Před vložením aplikace do kontejneru aplikačního serveru, je nutné mít nastaven atribut této třídy *PATH* na hodnotu, která obsahuje cestu k adresáři, v němž se nachází úložiště webového serveru (v případě standardní instalace balíku XAMP 1.7.3 obsahující webový server Apache se jedná o adresář *C:/xampp/htdocs/*).

Třída *SamplesServlet* obsahuje dvě metody. Metoda *doGet* slouží ke zpracování příchozího požadavku HTTP protokolu metody GET. Jako odpověď na tuto metodu se pošle odesílateli inicializačního požadavku seznam uložených filmových ukázek nacházejících se na webovém serveru. Metoda *doPost* se vykoná v takovém případě, kdy je příchozí požadavek HTTP protokolu typu metody POST. V aplikaci je této metody využito při nahrávání a mazání filmových ukázek na webový server.



Obr. 4.10: UML diagram HTTP Servlet aplikace

## 5. Analýza komunikace

Tato kapitola se zabývá analýzou registrace klienta do IMS sítě a komunikací mezi klientskou aplikací a SIP Servletem prostřednictvím SIP protokolu. Sledování průběhu komunikace a zachytávání paketů bylo provedeno za pomoci analyzátoru síťového provozu Wireshark 1.2.2.

### 5.1. Registrace klienta do IMS

Registrace klienta do IMS sítě (viz obr. 5.1) se skládá se dvou fází, tak jak bylo popsáno v kapitole 2.3. Díky tomuto procesu je provedena autorizace a autentizace uživatele v IMS.

Time	192.168.0.101	192.168.0.1	Comment
0,032	Request: REGISTER s (1050) → (5081)		SIP: Request: REGISTER sip:ericsson.com
0,270	Status: 401 Unautho (1050) ← (5081)		SIP: Status: 401 Unauthorized - Respond to challenge (0 bindings)
0,361	Request: REGISTER s (1050) → (5081)		SIP: Request: REGISTER sip:ericsson.com
0,496	Status: 200 OK (1050) ← (5081)		SIP: Status: 200 OK (1 bindings)

Obr. 5.1: Registrace klienta do IMS

Na obrázku 5.2 je zobrazen obsah SIP zprávy REGISTER, která byla vyslána klientem jako reakce na příchozí SIP zprávu třídy 401 (Unauthorized) od S-CSCF. Tato nová zpráva se liší od původní zprávy REGISTER pouze tím, že do ní byla přidána nová hlavička *Authorization*, která slouží k autentizaci uživatele v IMS. Nejdůležitější hodnotou této hlavičky je položka *response*. Tato položka obsahuje odpověď RES spočítanou z hodnot AUTN, RAND a uživatelova hesla pomocí hašovacího algoritmu MD5. S-CSCF porovná tuto spočtenou hodnotu s očekávanou hodnotou XRES a pokud se shodují, uživatele zaregistruje. Položka *nonce* obsahuje původní hodnoty AUTN a RAND, které byli klientem přijaty. [14]

Z hodnoty hlavičky *Expires* lze vyčíst, že klient bude v IMS přihlášen pod dobu 3 600 s. Správnou registraci klienta do IMS je možné si ověřit v záznamech Registrar serveru v perspektivě *Provisioning* na záložce *Registrar* (viz obr. 5.3).

```

REGISTER sip:ericsson.com SIP/2.0
Max-Forwards: 70
CSeq: 777 REGISTER
Expires: 3600
Content-Length: 0
Contact: <sip:192.168.0.101:59313;transport=tcp>;
        message;+g.3gpp.app-ref="urn%3Aurn-xxx%3Acom.ericsson.client.TicketsClient"
Authorization: Digest username="pavel@ericsson.com",
        realm="ericsson.com",
        nonce="ac3528d10801711996eb56bf886cb9f9",
        uri="sip:ericsson.com",qop=auth,nc=00000001,
        cnonce="7aef1a75",
        response="489ff698f5f9e9aeb29c38bbc431d986",
        opaque="b9006ac52c63443ca27face4121d73a0",
        algorithm=MD5
User-Agent: Ijcu Client
P-Preferred-Identity: <sip:pavel@ericsson.com>
From: <sip:pavel@ericsson.com>;tag=1257588639076
To: <sip:pavel@ericsson.com>
Route: <sip:192.168.0.1:5081;transport=tcp>
Call-ID: dcce53ead6e83116a4d0fe9448fed167@192.168.0.101
Via: SIP/2.0/TCP 192.168.0.201:1042;branch=z9hG4bK9fe991fcd492504de692c185209d2abd

```

**Obr. 5.2: SIP zpráva REGISTER**

The screenshot shows a web-based interface for a Registrar server. At the top, there are navigation tabs for DNS, HSS, BGCF, and Registrar. Below the tabs is a search bar with a dropdown menu set to 'User Agent Info' and an empty 'Value' input field. Below the search bar is a table with three columns: 'User Agent Info', 'Contact Addresses', and 'Expiration Time'. The table contains one entry for 'Ijcu Client' with a contact address of 'sip:192.168.0.1:53588;transport=tcp' and an expiration time of '00:59:02'. There are checkboxes next to the entries in the 'User Agent Info' column.

User Agent Info	Contact Addresses	Expiration Time
<input type="checkbox"/> sip:pavel@ericsson.com		
<input type="checkbox"/> Ijcu Client	sip:192.168.0.1:53588;transport=tcp	00:59:02

**Obr. 5.3: Záznamy Registrar serveru**

## 5.2. Komunikace mezi klientskou aplikací a SIP Servletem

Ukázka, jak může vypadat SIP komunikace mezi klientskou aplikací a SIP Servletem, který je spuštěn na aplikačním serveru, je znázorněna na obrázku 5.4. Obě aplikace si vzájemně vyměňují SIP zprávy typu MESSAGE, v jejichž těle jsou obsaženy přenášené informace. Zpráva je vždy potvrzena druhou stranou SIP odpovědí třídy 200 OK. Aby aplikace mohly rozlišit, co je obsahem zprávy, bylo pro tento účel vyhrazeno několik SIP adres domény [tickets.ericsson.com](http://tickets.ericsson.com) (viz tab. 5.1).

Time	192.168.0.101	192.168.0.1	Comment
13,802	Request: MESSAGE sip	(1127) (5081)	SIP: Request: MESSAGE sip:submit@tickets.ericsson.com (text/plain)
13,881	Status: 200 OK	(1127) (5081)	SIP: Status: 200 OK
13,907	Request: MESSAGE sip	(55127) (2133)	SIP: Request: MESSAGE sip:192.168.0.101:55127;transport=tcp (text/plain)
14,842	Status: 200 OK	(55127) (2133)	SIP: Status: 200 OK
15,956	Request: MESSAGE sip	(1127) (5081)	SIP: Request: MESSAGE sip:choiceFilm@tickets.ericsson.com
16,026	Status: 200 OK	(1127) (5081)	SIP: Status: 200 OK
16,088	Request: MESSAGE sip	(55127) (2133)	SIP: Request: MESSAGE sip:192.168.0.101:55127;transport=tcp (text/plain)
16,320	Status: 200 OK	(55127) (2133)	SIP: Status: 200 OK
22,362	Request: MESSAGE sip	(1127) (5081)	SIP: Request: MESSAGE sip:choiceDate@tickets.ericsson.com (text/plain)
22,418	Status: 200 OK	(1127) (5081)	SIP: Status: 200 OK
22,455	Request: MESSAGE sip	(55127) (2133)	SIP: Request: MESSAGE sip:192.168.0.101:55127;transport=tcp (text/plain)
22,602	Status: 200 OK	(55127) (2133)	SIP: Status: 200 OK
24,310	Request: MESSAGE sip	(1127) (5081)	SIP: Request: MESSAGE sip:choiceCount@tickets.ericsson.com (text/plain)
24,366	Status: 200 OK	(1127) (5081)	SIP: Status: 200 OK
24,551	Request: MESSAGE sip	(55127) (2133)	SIP: Request: MESSAGE sip:192.168.0.101:55127;transport=tcp (text/plain)
24,660	Status: 200 OK	(55127) (2133)	SIP: Status: 200 OK
32,084	Request: MESSAGE sip	(1127) (5081)	SIP: Request: MESSAGE sip:choicePlace@tickets.ericsson.com (text/plain)
32,149	Status: 200 OK	(1127) (5081)	SIP: Status: 200 OK
32,217	Request: MESSAGE sip	(55127) (2133)	SIP: Request: MESSAGE sip:192.168.0.101:55127;transport=tcp (text/plain)
32,414	Status: 200 OK	(55127) (2133)	SIP: Status: 200 OK
42,921	Request: MESSAGE sip	(1127) (5081)	SIP: Request: MESSAGE sip:confirmReservation@tickets.ericsson.com (text/plain)
42,996	Status: 200 OK	(1127) (5081)	SIP: Status: 200 OK
43,189	Request: MESSAGE sip	(55127) (2133)	SIP: Request: MESSAGE sip:192.168.0.101:55127;transport=tcp (text/plain)
43,375	Status: 200 OK	(55127) (2133)	SIP: Status: 200 OK

Obr. 5.4: Zachycená komunikace mezi klientskou aplikací a SIP Servletem

Tab. 5.1: Vyhrazené SIP adresy pro komunikaci

SIP URI	Použití
<a href="mailto:sip:registration@tickets.ericsson.com">sip:registration@tickets.ericsson.com</a>	Registrace nového uživatele
<a href="mailto:sip:lostpassword@tickets.ericsson.com">sip:lostpassword@tickets.ericsson.com</a>	Žádost o posláni nového hesla na email
<a href="mailto:sip:submit@tickets.ericsson.com">sip:submit@tickets.ericsson.com</a>	Přihlášení uživatele
<a href="mailto:sip:choiceFilm@tickets.ericsson.com">sip:choiceFilm@tickets.ericsson.com</a>	Výběr filmu
<a href="mailto:sip:choiceDate@tickets.ericsson.com">sip:choiceDate@tickets.ericsson.com</a>	Výběr doby promítání
<a href="mailto:sip:choiceCount@tickets.ericsson.com">sip:choiceCount@tickets.ericsson.com</a>	Výběr počtu vstupenek k rezervaci
<a href="mailto:sip:choicePlace@tickets.ericsson.com">sip:choicePlace@tickets.ericsson.com</a>	Výběr míst v sále
<a href="mailto:sip:confirmReservation@tickets.ericsson.com">sip:confirmReservation@tickets.ericsson.com</a>	Potvrzení provedené rezervace
<a href="mailto:sip:reservationHistory@tickets.ericsson.com">sip:reservationHistory@tickets.ericsson.com</a>	Prohlížení všech provedených rezervací
<a href="mailto:sip:infoReservation@tickets.ericsson.com">sip:infoReservation@tickets.ericsson.com</a>	Detailní informace o vybrané rezervaci
<a href="mailto:sip:userData@tickets.ericsson.com">sip:userData@tickets.ericsson.com</a>	Pošle registrační údaje pro jejich změnu
<a href="mailto:sip:changeUserData@tickets.ericsson.com">sip:changeUserData@tickets.ericsson.com</a>	Změna registračních údajů
<a href="mailto:sip:info@tickets.ericsson.com">sip:info@tickets.ericsson.com</a>	Posláni krátké zprávy, která se objeví klientovy ihned na displeji mobilního zařízení (chybové zprávy, potvrzení)



Na obrázku 5.5 je zobrazen obsah SIP zprávy, kterou vyslal klient směrem k SIP Servletu. Podle SIP adresy příjemce, která je [sip:submit@tickets.ericsson.com](mailto:sip:submit@tickets.ericsson.com), je zřejmé, že se jedná o zprávu určenou pro přihlášení uživatele do systému pro rezervaci vstupenek. Zpráva byla poslána uživatelem s přihlašovací jménem *pavel*, jehož SIP adresa je [sip:pavel@ericsson.com](mailto:sip:pavel@ericsson.com). SIP zpráva je typu MESSAGE a v jejím těle byla přenesena informace ve formátu značkovacího jazyka XML. Z obsahu zprávy lze vyčíst přihlašovací jméno do systému pro rezervaci vstupenek a otisk uživatelského hesla provedený pomocí hašovacího algoritmu MD5.

```
MESSAGE sip:submit@tickets.ericsson.com SIP/2.0
Max-Forwards: 70
CSeq: 529 MESSAGE
Content-Length: 136
Contact: <sip:192.168.0.101:55127;transport=tcp>
P-Preferred-Identity: <sip:pavel@ericsson.com>
Accept-Contact: *;+g.3gpp.app-ref="urn%3Aurn-xxx%3Acom.ericsson.client.TicketsClient";
    require
Accept-Contact: *;message
Route: <sip:192.168.0.1:5081;transport=tcp;lr>,<sip:orig@192.168.0.1:5082;lr>
To: <sip:submit@tickets.ericsson.com>
From: <sip:pavel@ericsson.com>;tag=1271747862580
Call-ID: 4e7d0db5ad700b6bc63a8cf26114b058@192.168.0.101
Via: SIP/2.0/TCP 192.168.0.201:1127;branch=z9hG4bKcc41795d69dce37c28606d77d868f3e5
Content-Type: text/plain

<?xml version='1.0' encoding='windows-1250' ?>
<submit>
  <login>pavel</login>
  <password>ef1652b79c940145b600de7a2fe0288e</password>
</submit>
```

**Obr. 5.5: SIP zpráva MESSAGE určená pro přihlášení uživatele**

SIP zpráva přenášená opačným směrem, tedy zpráva vyslána SIP Servletem směrem ke klientovi, je zobrazena na obrázku 5.6. Adresa odesílatele je [sip:choiceFilm@tickets.ericsson.com](mailto:sip:choiceFilm@tickets.ericsson.com), což znamená, že zpráva obsahuje nabídku filmů, které jsou v kině promítány. Aby mohlo mobilní zařízení, na kterém je spuštěna klientská aplikace správně dekodovat znaky s českou diakritikou, byla do SIP zprávy přidána hlavička *Content-Encoding* s hodnotou *windows-1250*, jež značí použitou znakovou sadu při kódování obsahu těla zprávy.

```

MESSAGE sip:192.168.0.101:55127;transport=tcp SIP/2.0
Max-Forwards: 68
CSeq: 1 MESSAGE
P-Called-Party-ID: <sip:pavel@ericsson.com>
Content-Length: 2213
Record-Route: <sip:192.168.0.1:5082;from-tag=g88dubkw-k;lr>
Accept-Contact: *;+g.3gpp.app-ref="urn%3Aurn-xxx%3Acom.ericsson.client.TicketsClient";
    require
P-Asserted-Identity: sip:choiceFilm@tickets.ericsson.com
To: <sip:pavel@ericsson.com>
From: <sip:choiceFilm@tickets.ericsson.com>;tag=g88dubkw-k
Call-ID: 192.168.0.1_10_7722394204933494933
Via: SIP/2.0/tcp 192.168.0.1:5082;branch=z9hG4bKbc23b24697474bdf619fb448ad767681,SIP/
    2.0/TCP
    192.168.0.1:5060;branch=z9hG4bKdaac68add150ab054dfdba74d2f138b3388e;receive
    d_port_ext=1969;received=192.168.0.1
Content-Type: text/plain
Content-Encoding: windows-1250

<?xml version="1.0" encoding="windows-1250" standalone="no"?>
<RezervaceVstupenek>
  <Film>
    <ID_film>42</ID_film>
    <nazev>Avatar</nazev>
    <zanr>Sci-Fi</zanr>
    <verze>.esk. titulky</verze>
    <delka>02:42:00</delka>
    <rezie>James Cameron</rezie>
    <herci>Sam Worthington, Zoe Saldana, Sigourney Weaver, Michelle Rodriguez, Giovanni
Ribisi, Joel Moore, Wes Studi, Laz Alonso</herci>
    <link>http://192.168.0.1/avatar.3gp</link>
  </Film>
  <Film>
    <ID_film>43</ID_film>
    <nazev>Gran Torino</nazev>
    <zanr>Drama</zanr>
    <verze>.esk. dabing</verze>
    <delka>01:57:00</delka>
    <rezie>Clint Eastwood</rezie>
    <herci>Clint Eastwood, Christopher Carley, Bee Vang, Ahney Her, Geraldine Hughes,
Dreama Walker</herci>
    <link>http://192.168.0.1/grantorino.3gp</link>
  </Film>
  <Film>
    <ID_film>44</ID_film>
    <nazev>Vratn. lahve</nazev>
    <zanr>Komedie</zanr>
    <verze>P.vodn. zn.n.</verze>
    <delka>01:39:00</delka>
    <rezie>Jan Sv.r.k</rezie>
    <herci>Zden.k Sv.r.k, Daniela Kol..ov., Tatiana Vilhelmov., Ji.. Mach..ek, Pavel Landovsk.,
Jan Buda., Nela Boudov., Miroslav T.borsk., V.ra Tich.nkov., Jana Plodkov., Martin Pechl.t</
herci>
    <link>http://192.168.0.1/vratne_lahve.3gp</link>
  </Film>
</RezervaceVstupenek>

```

**Obr. 5.6: SIP zpráva MESSAGE s nabídkou filmů k promítání**

### 5.2.1. Rozbor záhlaví a pole hlaviček SIP zprávy

Detailní rozbor záhlaví a pole hlaviček SIP zprávy je proveden pouze na jedné zvolené zprávě, která je zobrazená na obrázku 5.6. Podle struktury záhlaví SIP zprávy lze říci, že se jedná o SIP zprávu typu žádosti. Metoda SIP zprávy je nastavena na hodnotu MESSAGE, tedy je SIP zpráva určena pro přenos krátkých textových zpráv.

URI adresa v záhlaví obsahuje SIP adresu toho, komu je žádost určena. V zachycené SIP zprávě je hodnota této položky záhlaví nastavena na hodnotu [sip:192.168.0.101:55127;transport=tcp](mailto:sip:192.168.0.101:55127;transport=tcp). SIP adresa obsahuje IP adresu síťového rozhraní počítače, na kterém byl spuštěn emulátor mobilního zařízení a číslo portu, na kterém komunikovala spuštěná klientská aplikace. Nepovinný parametr transport udává, že je použit transportní protokol TCP. Verze protokolu SIP je SIP/2.0. [13]

Z hlavičky SIP zprávy *Max-Forwards* lze vyčíst, že počet možných přeskoků zprávy přes mezilehlé uzly v síti je 68 z původních 70. Hlavička *Cseq* udává sekvenční číslo žádosti 1 a typ SIP metody MESSAGE. Odpověď na tuto žádost musí mít uvedeno stejné číslo v hlavičce *Cseq*. [14]

Hlavička *P-Called-Party-ID* obsahuje SIP adresu příjemce, jejíž hodnota je [sip:pavel@ericsson.com](mailto:sip:pavel@ericsson.com). Tato hlavička má vhodné použití v takovém případě, kdy klient používá více než jednu SIP adresu. SIP Proxy server může změnit hodnotu SIP adresy příjemce na takovou adresu, pod kterou je klient právě zaregistrován. Hodnota v hlavičce *P-Called-Party-ID* však zůstane zachována. Díky této hlavičce, může klientské zařízení zjistit původně nastavenou SIP adresu příjemce. [16]

Hlavičky *Content-Length* obsahuje hodnotu 2 213, což je velikost těla zprávy v bajtech [14]. Hlavička *Record-Route* je používána pro striktní směrování uvnitř sítě využívajícího protokolu SIP. Každý SIP Proxy server, kterým zpráva prošla, vloží do této hlavičky svoji adresu [16]. V tomto případě hlavička obsahuje pouze jednu adresu, a to adresu S-CSCF serveru, která je 192.168.0.1:5082. Hlavička *Accept-Contact* specifikuje hodnotu IARI (IMS Application Reference Identifier), která slouží k identifikaci aplikace. IARI je užitečné pouze pro klienty IMS a je absolutně ignorováno entitami jádra sítě IMS [6]. Hlavička *P-Asserted-Identity* obsahuje SIP adresu odesílatele zprávy [sip:choiceFilm@tickets.ericsson.com](mailto:sip:choiceFilm@tickets.ericsson.com). S-CSCF zpracovává pouze ověřené žádosti. Bez této hlavičky by server S-CSCF žádost odmítl s odpovědí 403 Forbidden (zapovězeno). Tato hlavička prokáže, že žádost pochází z důvěryhodné domény [5].

Hlavičky *To* obsahuje SIP adresu příjemce [sip:pavel@ericsson.com](mailto:sip:pavel@ericsson.com) a hlavička *From* SIP adresu odesilatele [sip:choiceFilm@tickets.ericsson.com](mailto:sip:choiceFilm@tickets.ericsson.com). Hlavička *Call-ID* obsahuje jedinečný identifikátor pro danou relaci generovanou na straně klienta. Hlavička *Via* nese informaci o tom, jakou cestou zpráva prošla. V tomto případě jsou zde obsaženy dvě adresy: 192.168.0.1:5082 (S-CSCF) a 192.168.0.1:5060 (aplikační server). Hlavička *Content-Type* informuje příjemce, jakého typu je obsah těla SIP zprávy. Nastavená hodnota *text/plain* znamená, že se jedná o prostý, čitelný a nezakódovaný text. Nastavená hodnota *windows-1250* hlavičky *Content-Encoding* značí použitou znakovou sadu při kódování obsahu těla zprávy. [14]

### 5.2.2. Žádost o vygenerování nového hesla

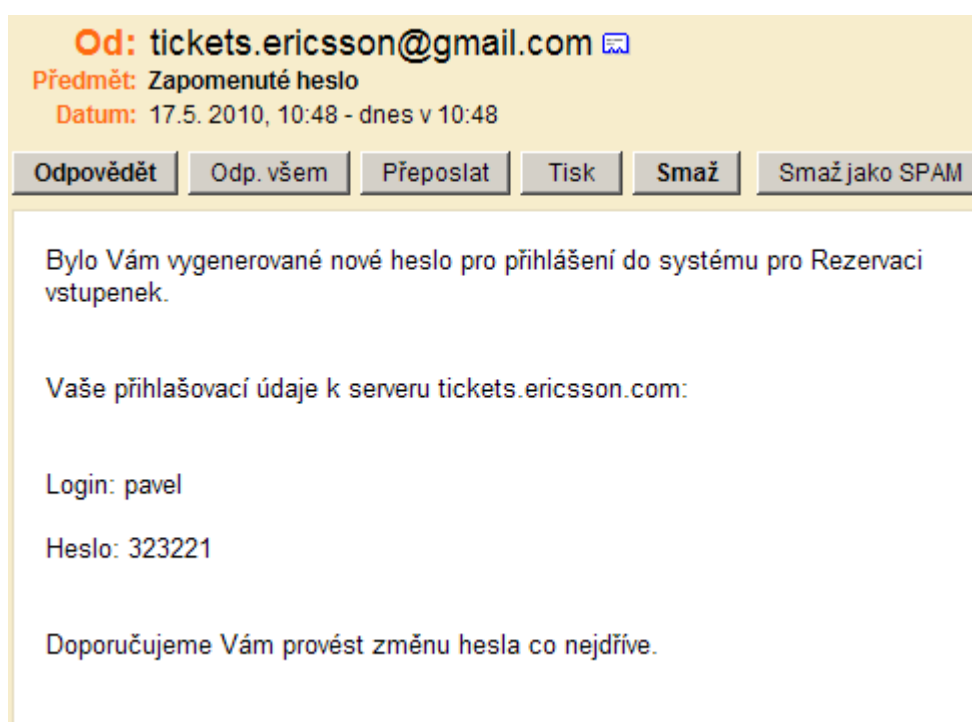
Pokud klient zapomene své přihlašovací heslo do systému pro rezervaci vstupenek, může si nechat vygenerovat heslo nové. SIP zpráva, kterou vyšle klient směrem k SIP Servlet aplikaci, bude obsahovat SIP adresu příjemce s hodnotou [sip:lostpassword@tickets.ericsson.com](mailto:sip:lostpassword@tickets.ericsson.com). V těle zprávy jsou přenášeny informace o uživatelově přihlašovacím jméně a jeho emailové adrese zadané při registraci (viz obr. 5.7).

```
MESSAGE sip:lostpassword@tickets.ericsson.com SIP/2.0
Max-Forwards: 70
CSeq: 337 MESSAGE
Content-Length: 139
Contact: <sip:192.168.0.1:59018;transport=tcp>
P-Preferred-Identity: <sip:pavel@ericsson.com>
Accept-Contact: *;+g.3gpp.app-ref="urn%3Aurn-xxx%3Acom.ericsson.client.TicketsClient";require
Accept-Contact: *;message
Route: <sip:192.168.0.1:5081;transport=tcp;lr>,<sip:orig@192.168.0.1:5082;lr>
To: <sip:lostpassword@tickets.ericsson.com>
From: <sip:pavel@ericsson.com>;tag=1274085714125
Call-ID: 7d97cd0cd6db8d5e92fc0689547e2f32@10.0.0.3
Via: SIP/2.0/TCP 192.168.0.201:2073;branch=z9hG4bKe8f47d02d28f8187fac1f359549a09c0
Content-Type: text/plain

<?xml version='1.0' encoding='windows-1250' ?>
<lostPassword>
  <login>pavel</login>
  <email>pavel.otoupalik&#64;seznam.cz</email>
</lostPassword>
```

Obr. 5.7: SIP zpráva MESSAGE s žádostí o vygenerování nového hesla

SIP Servlet podle přihlašovacího jména ověří, zda je uživatel již v systému zaregistrován a zda souhlasí emailová adresa s tou, která byla zadána při registraci a je uložena v MySQL databázi. Pokud údaje souhlasí, SIP Servlet vygeneruje nové heslo jako náhodné číslo od 0 do 999 999. Toto heslo pak SIP Servlet pošle na emailovou adresu klienta a haš hesla vypočítaný pomocí hašovacího algoritmu MD5 uloží do MySQL databáze. Ukázka emailové zprávy s novým vygenerovaným heslem je zobrazena na obrázku 5.8. Heslo je přenášeno ve zprávě v otevřené podobě a lze ho tak snadno zachytit. Proto je klientovi v textu emailové zprávy doporučeno, aby si heslo co nejdříve změnil.



**Obr. 5.8: Emailová zpráva s nově vygenerovaným heslem**

### 5.2.3. Stažení filmové ukázky z webového serveru

Klient si může na svém mobilním zařízení přehrát krátkou filmovou ukázkou k vybranému filmu. Ukázka se stáhne z webového serveru Apache, který naslouchá na portu 80. Komunikace mezi klientskou aplikací a webovým serverem je zachycena na obrázku 5.9. Navázání spojení začíná tak, že strana, která spojení iniciuje (klientská aplikace), zašle TCP segment s nastaveným příznakem SYN (požadována synchronizace číslování přenášených bajtů v dopředném směru) a v poli SEQ toto číslo

nastaví. Webový server odpoví nastavením příznaku ACK (synchronizace OK) a zároveň taktéž chce synchronizovat (nastaví SYN bit) číslování pro přenos dat ve zpětném směru a toto prvotní číslo nastaví do SEQ. Ještě potvrdí přijetí bajtu číslovaného jako 0 tím, že do pole potvrzovaného bajtu (ACK) dá 1. To vše je odesláno v jednom segmentu. Klient následně nastaví příznak ACK (synchronizace pro zpětný přenos OK), dále potvrdí příjem bajtu číslo 0 tím, že do pole potvrzovaného bajtu dá číslo 1. Číslo SEQ samozřejmě vzroste, protože musí být zvýšeno na základě počtu přenesených bajtů. Vzniklá jednotka se opět odešle a spojení je nyní navázáno. Tomuto způsobu navázání spojení se říká three-way handshake (třicestné podání rukou). Zkrácený zápis této komunikace: [SYN] > [SYN, ACK] > [ACK]. [10]

Následně klient prostřednictvím protokolu HTTP a jeho metody GET předá webovému serveru název filmové ukázky jako součást URL, která má být stažena. Server začne posílat klientovi data přes již vytvořené TCP spojení a klient jej potvrzuje nastaveným příznakem ACK. Jakmile dojde k úplnému stažení souboru, webový server potvrdí vyřízení požadavku GET odpovědí 200 OK.

Ukončení TCP spojení inicializuje webový server vysláním TCP segmentu s nastavenými příznaky FIN a ACK. Klient ukončení potvrdí pouze nastaveným příznakem ACK.

Time	192.168.0.101	192.168.0.1	Comment
0,002	(1070)		TCP: gmupdateserv > http [SYN] Seq=0 Win=64240 Len=0 MSS=1460
0,002		(180)	TCP: http > gmupdateserv [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
0,002	(1070)		TCP: gmupdateserv > http [ACK] Seq=1 Ack=1 Win=64240 Len=0
0,003	(1070)		HTTP: GET /vratne_lahve.3gp HTTP/1.1
0,003	(1070)		TCP: [TCP segment of a reassembled PDU]
0,003	(1070)		TCP: [TCP segment of a reassembled PDU]
0,004	(1070)		TCP: gmupdateserv > http [ACK] Seq=114 Ack=2921 Win=64240 Len=0
0,004	(1070)		TCP: [TCP segment of a reassembled PDU]
0,494	(1070)		TCP: [TCP segment of a reassembled PDU]
0,494	(1070)		TCP: gmupdateserv > http [ACK] Seq=114 Ack=2739420 Win=6896 Len=0
0,494	(1070)		TCP: gmupdateserv > http [ACK] Seq=114 Ack=2742056 Win=4260 Len=0
0,494	(1070)		HTTP: HTTP/1.1 200 OK (video/3gpp)
0,495	(1070)		TCP: gmupdateserv > http [ACK] Seq=114 Ack=2744692 Win=1624 Len=0
0,502	(1070)		TCP: gmupdateserv > http [ACK] Seq=114 Ack=2745870 Win=4206 Len=0
0,503	(1070)		TCP: [TCP Window Update] gmupdateserv > http [ACK] Seq=114 Ack=2745870 Win=
6,007	(1070)		TCP: http > gmupdateserv [FIN, ACK] Seq=2745870 Ack=114 Win=65422 Len=0
6,007	(1070)		TCP: gmupdateserv > http [ACK] Seq=114 Ack=2745871 Win=64240 Len=0

Obr. 5.9: Stažení filmové ukázky

## 6. Závěr

Teoretická část práce se zabývala popisem architektury IMS. Díky tomuto systému je umožněno ve stávajících mobilních sítích nasazení moderních komunikačních technologií bez nutnosti radikálního zásahu do jejich architektury. Důvodem vzniku IMS bylo sjednocení přenosu hlasových a datových služeb do jediné ucelené sítě. IMS tvoří architekturu přenosové sítě, která je dostupná globálně a nezávislá na použité přístupové technologii. Signalizace v IMS je zajištěna protokolem SIP. Tento signalizační protokol umožňuje uživatelům registrovat se do systému IMS a následně vytvořit spojení mezi koncovými zařízeními.

V rámci praktické části projektu byl vytvořen systém umožňující rezervaci vstupenek do kina prostřednictvím sítě IMS. Realizovaný projekt se skládá celkem ze čtyř aplikací a databáze MySQL.

První aplikace je určena běžným mobilním zařízení s podporou platformy J2ME, pomocí níž se mohou klienti přihlásit do IMS sítě a využívat služeb systému pro rezervaci vstupenek. Klientská aplikace byla testována na emulátoru mobilního zařízení Sony Ericsson W810i. Druhou aplikací je SIP Servlet aplikace, kterou je možné vložit do SIP Servlet kontejneru aplikačního serveru. SIP Servletu je umožněn přístup do MySQL databáze a na základě příkazů přijatých od klientů může nad touto databází vykonávat požadované operace. Tyto dvě aplikace byly naprogramovány v programovacím jazyce Java ve vývojovém prostředí SDS 4.1 FD1 od firmy Ericsson.

Pro ukládání záznamů o registrovaných uživatelích, promítaných filmech, nabízených vstupenkách a uskutečněných rezervacích, byla vytvořena databáze v databázovém systému MySQL. Aby mohla obsluha systému pro rezervaci vstupenek nějakým jednoduchým způsobem upravovat záznamy v databázi a vkládat do ní nové údaje, byla pro tento účel vytvořena aplikace pro správu databáze rovněž v programovacím jazyce Java.

Poslední aplikací, která byla vytvořena, je HTTP Servlet umožňující nahrávání a mazání krátkých filmových ukázek z webového serveru, odkud si jej klienti mohou stáhnout a přehrát na svém mobilním zařízení. Na závěr práce byla provedena analýza registrace klienta do IMS sítě a SIP komunikace mezi klientskou aplikací a SIP Servletem.

Vytvořený projekt slouží pouze k uskutečnění rezervace vstupenek do kina prostřednictvím mobilního zařízení za pomoci systému IMS. Není v něm žádným způsobem realizováno zpoplatnění rezervovaných vstupenek. Předpoklad návrhu byl takový, že platba za rezervované vstupenky se provede až při jejich převzetí. Klientům je na konci prováděné rezervace přiřazen jedinečný identifikační kód rezervace. Obsluha systému, jež má k dispozici aplikaci pro správu databáze, může na základě tohoto identifikačního čísla provést výdej rezervovaných lístků. Z praktických důvodů by bylo lepší, kdyby byla vytvořena ještě jedna aplikace, která by byla nainstalována na pokladnách a umožňovala by pouze výdej rezervovaných lístků. Tím by byla oddělena aplikace určena pro výdej rezervovaných lístků od aplikace pro správu databáze, která by sloužila pouze pro účely úprav záznamů v databázi MySQL.

Realizovaný systém, za pomoci něhož lze provést rezervaci vstupenek prostřednictvím mobilního zařízení a systému IMS, slouží jako ukázka a také možná alternativa k dnes nejpoužívanějším technologiím a síťovým protokolům určených pro přenos dat, jako je zejména v mobilních sítích technologie GPRS a internetový protokol HTTP. V dnešní době již většina mobilních zařízení obsahuje vestavěný internetový prohlížeč, nebo si jej lze nainstalovat ve formě Java aplikace (např. Opera Mini). V případě, kdyby byl projekt realizován ve formě webových stránek, byla by tvorba systému mnohem snazší a uživatel by se vyhnul nezbytné registraci do systému IMS.

Výhodou vytvořeného projektu je to, že je realizován v programovacím jazyku Java, který přináší programátorovi mnohem více možností při tvorbě programů, než kdyby byl systém realizován jen jako webové stránky. Programátor má větší přístup k hardwaru zařízení. Díky tomu může programátor definovat funkci jednotlivých tlačítek, využívat ukládacího prostoru zařízení a může přesně určit, kde a jak budou jednotlivé grafické objekty na displeji umístěny. Tím pádem může více ovlivnit, jak bude výsledná aplikace na displeji mobilního zařízení vypadat.



## Literatura

- [1] CAMARILLO, G., GACÍA-MARTÍN, M. A. *The 3G IP Multimedia Subsystem (IMS) – Merging the Internet and the Cellular Worlds*. V. Británie: WILEY, 2004. 407 s. ISBN 0-470-87156-3.
- [2] ČÍKA, P. *Multimediální služby: skripta*. Brno: FEKT VUT Brno, 2007. 106 s.
- [3] Ericsson. *Service Development Studio (SDS) 4.1 Developer's Guide* [online]. Poslední aktualizace 2009-02-13 [cit. 2009-11-14]. Ve formátu PDF. Dostupné z WWW: [http://www.ericsson.com/developer/sub/open/technologies/ims\\_poc/docs/sds\\_4\\_0\\_dev\\_guide](http://www.ericsson.com/developer/sub/open/technologies/ims_poc/docs/sds_4_0_dev_guide).
- [4] Ericsson. *Service Development Studio (SDS) 4.1 Installation Instructions* [online]. Poslední aktualizace 2009-02-16 [cit. 2009-11-14]. Ve formátu PDF. Dostupné z WWW: [http://www.ericsson.com/developer/sub/open/technologies/ims\\_poc/docs/sds\\_4\\_0\\_install\\_inst](http://www.ericsson.com/developer/sub/open/technologies/ims_poc/docs/sds_4_0_install_inst).
- [5] Ericsson. *Service Development Studio (SDS) 4.1 Tutorial* [online]. Poslední aktualizace 2009-02-13 [cit. 2009-11-14]. Ve formátu PDF. Dostupné z WWW: [http://www.ericsson.com/developer/sub/open/technologies/ims\\_poc/docs/sds40\\_tutorial](http://www.ericsson.com/developer/sub/open/technologies/ims_poc/docs/sds40_tutorial).
- [6] GOURRAUD, Ch. *3GPP Communication Services* [online]. Poslední aktualizace 2008-04-09 [cit. 2010-05-18]. Dostupné z WWW: <http://theimslantern.blogspot.com/2008/04/3gpp-communication-services.html>.
- [7] HEROUT, P. *Učebnice jazyka Java*. České Budějovice: Kopp, 2001. 349 s. ISBN 80-7232-115-3.
- [8] HORÁK, M. *Vývoj služeb v mobilních sítích* [online]. Poslední aktualizace 2008-03-17 [cit. 2010-03-25]. Dostupné z WWW: <http://access.feld.cvut.cz/rservice.php?akce=tisk&cisloclanku=2008030001>.
- [9] *Java ME* [online]. Poslední aktualizace 2009-05-11 [cit. 2010-04-23]. Dostupné z WWW: [http://kore.fi.muni.cz:5080/wiki/index.php/Java\\_ME](http://kore.fi.muni.cz:5080/wiki/index.php/Java_ME).
- [10] JEŘÁBEK, J. *Pokročilé komunikační techniky: skripta*. Brno: FEKT VUT Brno, 2009. 230 s.
- [11] MAHMOUD, Q. H. *The J2ME Mobile Media API* [online]. Poslední aktualizace Červen 2003 [cit. 2010-04-23]. Dostupné z WWW: <http://developers.sun.com/mobility/midp/articles/mmapioverview>.
- [12] MASLAKOWSKI, M. *Naučte se MySQL za 21 dní*. Brno: Computer Press, 2001. 478 s. ISBN 80-7226-448-6.

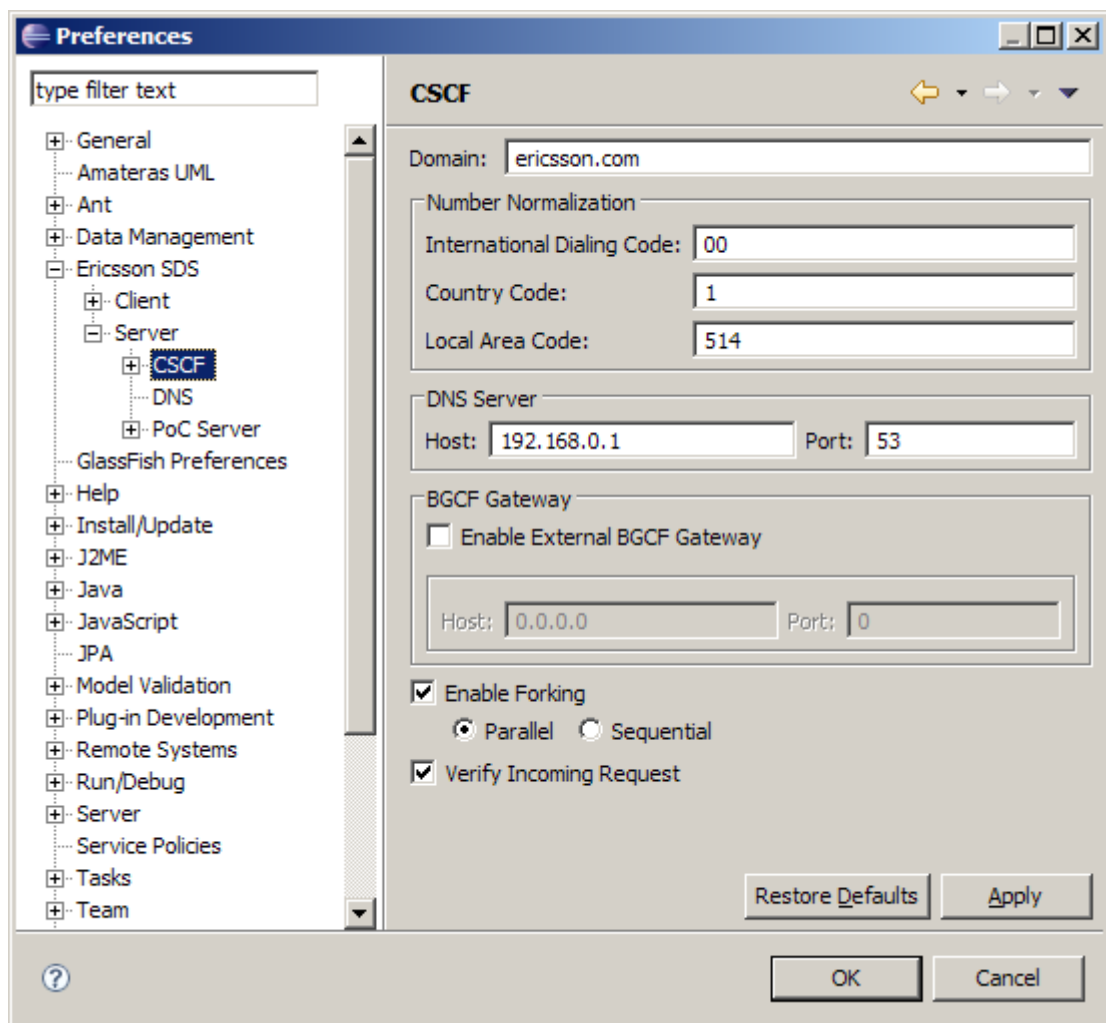
- [13] NOVOTNÝ, V. *Účastnická koncová zařízení: skripta*. Brno: FEKT VUT Brno, 2002. 122 s.
- [14] POIKSELKA, M., MAYER, G., KHARTABIL, H., NIEMI, A. *The IMS: IP Multimedia Concepts and Services in the Mobile Domain*. V. Británie: WILEY, 2004. 450 s. ISBN 0-470-87113-X.
- [15] PROCHÁZKA, J. *J2ME pro pokročilé – XML* [online]. Poslední aktualizace 2003-12-02 [cit. 2010-05-15]. Dostupné z WWW: <<http://interval.cz/clanky/j2me-pro-pokrocile-xml/>>.
- [16] RUSSELL, T. *Session Initiation Protocol (SIP) – Controlling Convergent Networks*. V. Británie: Mc Graw-Hill OSBOURNE, 2008. 284 s. ISBN 0071488529.
- [17] SCHNEIDER, D., R. *MySQL: oficiální průvodce tvorbou, správou a laděním databází*. Praha: Grada, 2006. 372 s. ISBN 80-247-1516-3.
- [18] WHITE, J. P., HEMPHILL, D. A. *Java 2 Micro Edition: Java in Small Things*. USA: Manning Publications Co, 2002. 501 s. ISBN 1-930110-33-2.
- [19] WILDING-MCBRIDE, D. *Java™ Development on PDAs: Building Applications for PocketPC and Palm Devices*. USA: Addison-Wesley, 2003. 256 s. ISBN 0-201-71954-1.

## Seznam příloh

<b>A.</b>	<b>Nastavení SDS .....</b>	<b>68</b>
<b>B.</b>	<b>Ukázky klientské aplikace.....</b>	<b>74</b>
<b>C.</b>	<b>Ukázky aplikace pro správu databáze .....</b>	<b>79</b>
<b>D.</b>	<b>Obsah přiloženého DVD.....</b>	<b>85</b>

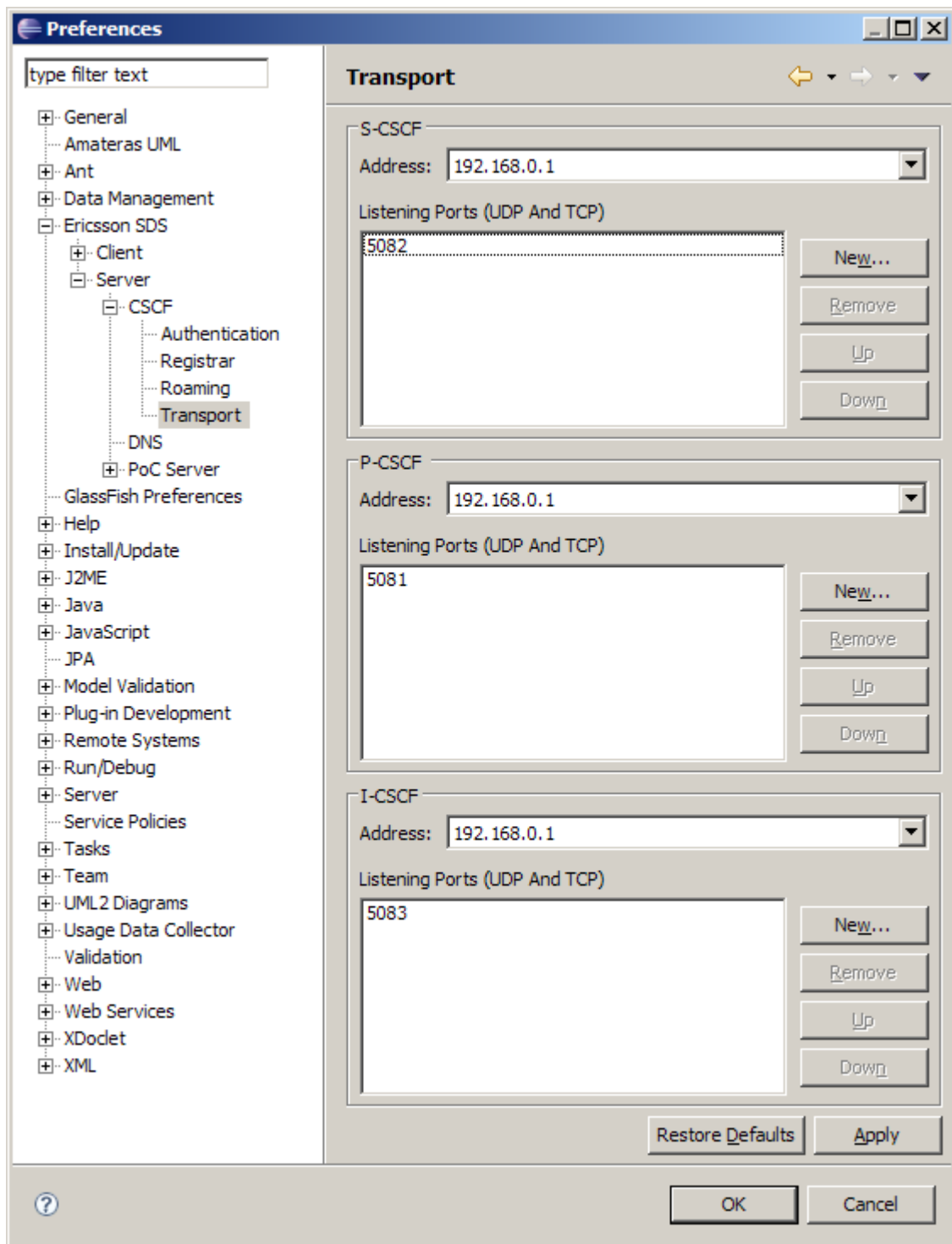
## A. Nastavení SDS

Pro správný chod aplikací je nutné v SDS nakonfigurovat simulovanou IMS síť. V okně *Preferences*, dostupného z hlavního menu ze záložky *Window* na kartě *Ericsson SDS > Server > CSCF*, musí být nastavena adresa DNS serveru na hodnotu IP adresy síťového rozhraní počítače (viz obr. A.1.). Port DNS serveru i doménové jméno CSCF zůstanou ponechané na původních hodnotách.



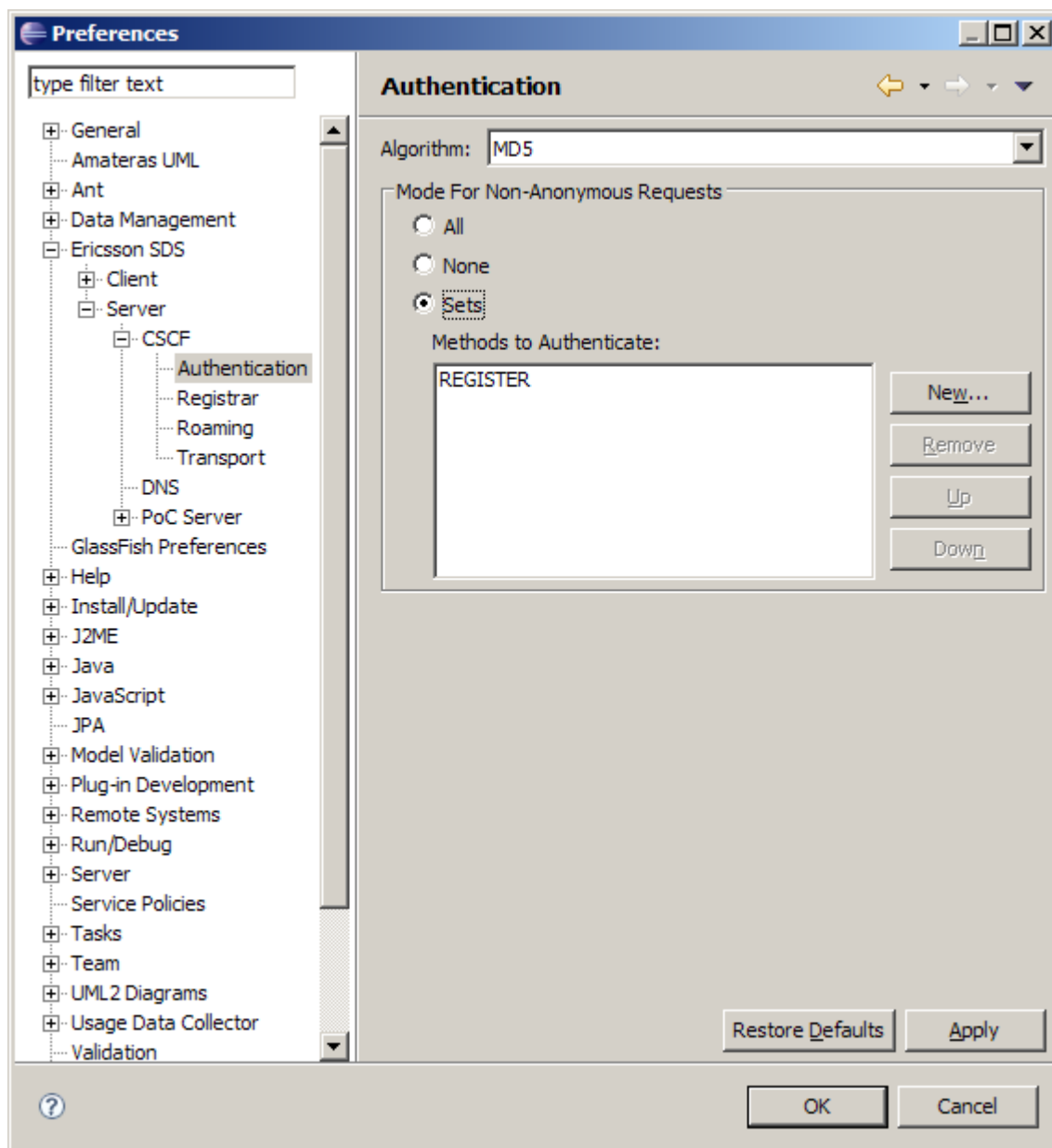
Obr: A.1: Nastavení adresy DNS serveru

Na kartě *Ericsson SDS > Server > CSCF > Transport* lze nastavit IP adresy a porty CSCF serverů. Všechny tyto servery musí mít opět nastavenou adresu na IP adresu síťového rozhraní počítače (viz obr. A.2).



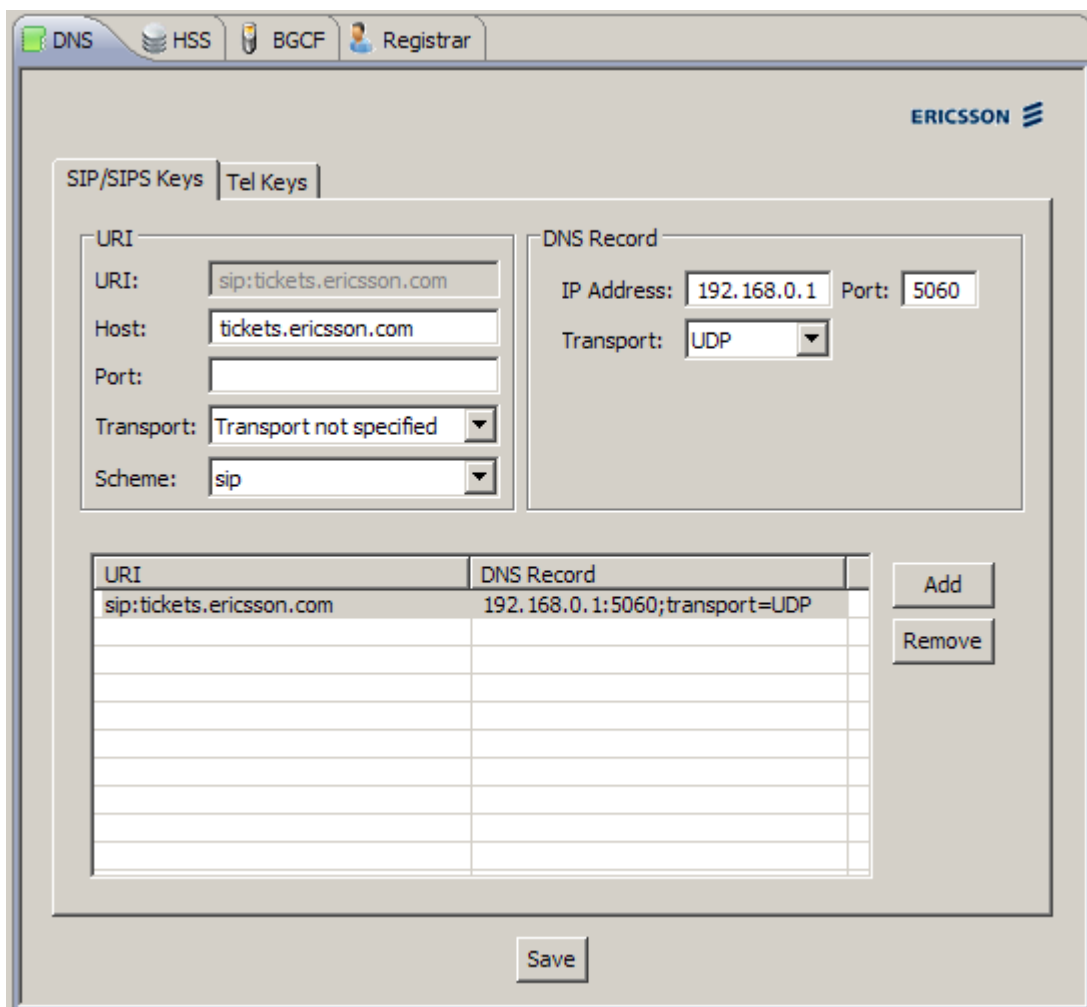
**Obr. A.2: Nastavení adres CSCF serverů**

Pro ověření totožnosti uživatele při registrace do IMS nastavíme na kartě *Ericsson SDS > Server > CSCF > Authentication* hašovací algoritmus na MD5. Autentizace musí být přiřazena jen zprávám typu REGISTER (viz obr. A.3).



**Obr. A.3: Nastavení autentizace pro registraci do IMS**

Pro účely konfigurace DNS serveru a databáze HSS je ve vývojovém prostředí SDS určena perspektiva *Provisioning*. V nastavení DNS serveru je potřeba vytvořit nový záznam, který přiřadí doménovému jménu [tickets.ericsson.com](http://tickets.ericsson.com) IP adresu 192.168.0.1 a port 5060, což je adresa aplikačního serveru. Toto nastavení lze provést v záložce *DNS*. (viz obr. A.4).



**Obr. A.4: Konfigurace DNS serveru**

Na kartě HSS lze nadefinovat filtrační kritéria SIP zpráv, které mají být přeposílány aplikačnímu serveru. V případě vytvořeného projektu je potřeba nastavit filtr SIP zpráv tak, aby byly aplikačnímu serveru, jehož URI je [sip:tickets.ericsson.com](mailto:sip:tickets.ericsson.com), posílány SIP zprávy typu MESSAGE. Na první záložce *Initial Filter Criteria* v kartě *Definition* lze přiřadit filtračnímu kritériu jeho prioritu a adresu aplikačního serveru, kam v případě splnění filtračních kritérií budou zprávy posílány (viz obr. A.5). Na druhé kartě *Service Point Trigger* lze nadefinovat posloupnost pravidel, které musí být splněny, aby byla zpráva AS přeposlána. Pro správnou funkčnost projektu, musí být definovány dvě pravidla (viz obr. A.6 a A.7), která zajistí přeposílání SIP zpráv typu MESSAGE aplikačnímu serveru. Tato pravidla musí být přiřazeny do stejné filtrační skupiny.

Initial Filter Criteria

Name:

Priority:

Trigger Point

Condition Type:

Application Server

Server Address:

Default Handling:

**Obr. A.5: Přiřazení filtračních kritérií AS**

Name:

Condition

Trigger when condition is:

Group:

Trigger Definition

Trigger Type:

Method:

**Obr. A.6: První filtrační pravidlo – typ zprávy MESSAGE**

Name:

Condition

Trigger when condition is:

Group:

Trigger Definition

Trigger Type:

Session Case:

**Obr. A.7: Druhé filtrační pravidlo – vyvolání spojení**



Vytvořený filtr musí být přiřazen na záložce *Service Profile* nově vytvořenému služebnímu profilu, pojmenovaného například *ticktes\_profile*. Tento služební profil je v poslední záložce *User Profile* přidělen vytvořeným uživatelským profilům. Každému uživateli je v této záložce navíc přiřazena veřejná identita (SIP adresa), soukromá identita a heslo (viz obr. A.8). SIP adresy a hesla pro vytvořené uživatele jsou následující:

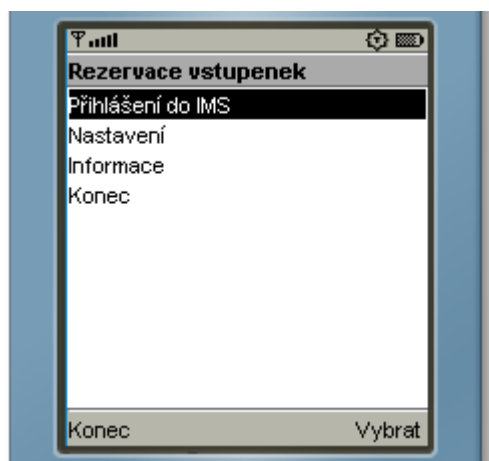
- **sip:pavel@ericsson.com** heslo: **pavel**
- **sip:jan@ericsson.com** heslo: **jan**
- **sip:franta@ericsson.com** heslo: **franta**

Public ID	Private ID	Password	Registration Status	Service Profile
sip:pavel@ericsson.com	pavel@ericsson.com	pavel	Not Registered	ticktes_profile
sip:jan@ericsson.com	jan@ericsson.com	jan	Not Registered	ticktes_profile
sip:franta@ericsson.com	franta@ericsson.com	franta	Not Registered	ticktes_profile

**Obr. A.8: Definice uživatelských profilů**

## B. Ukázky klientské aplikace

Po spuštění klientské aplikace se na displeji mobilní zařízení objeví hlavní menu (viz obr. B.1).



Obr. B.1: Hlavní menu klientské aplikace

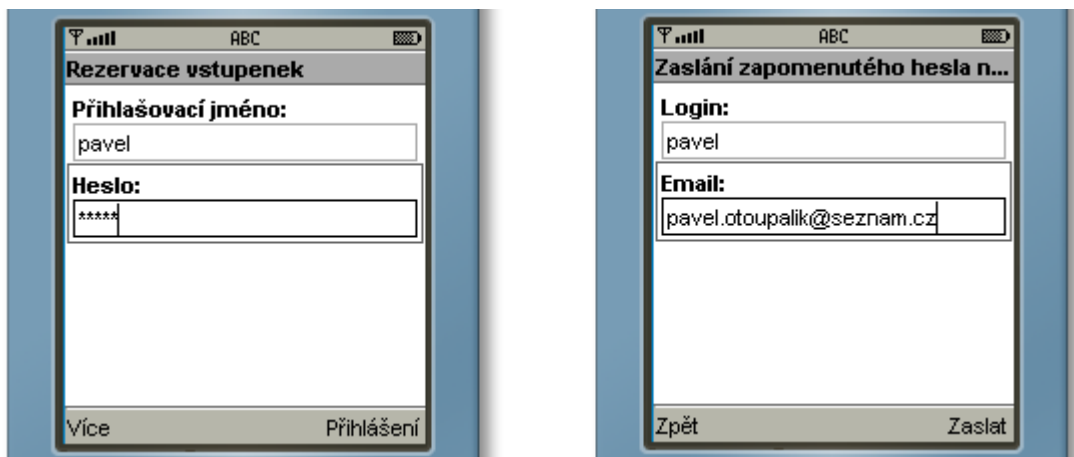
Na obrázku B.2 je zobrazeno nastavení konfiguračních parametrů klientské aplikace pro přihlášení se do IMS sítě a adresy aplikačního server, které je dostupné po volbě *Nastavení* z hlavního menu.



Obr. B.2: Nastavení parametrů pro registraci do IMS sítě a adresy AS

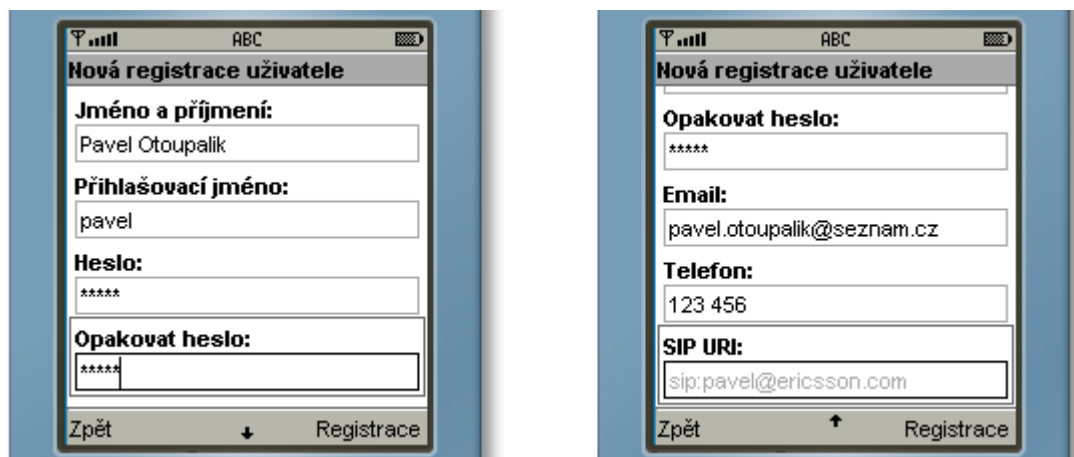
Po úspěšném přihlášení mobilního zařízení do IMS se uživatel může vyplněním formuláře, do kterého zadá své uživatelské jméno a heslo, přihlásit do systému pro rezervaci vstupenek. Zapomene-li uživatel své heslo, může si nechat po zvolení položky

z menu *Zapomenuté heslo*, vygenerovat heslo nové, které mu bude zasláno na emailovou schránku uvedenou při registraci. Formulář pro přihlášení a žádost o poslání nově vygenerovaného hesla je na obrázku B.3.



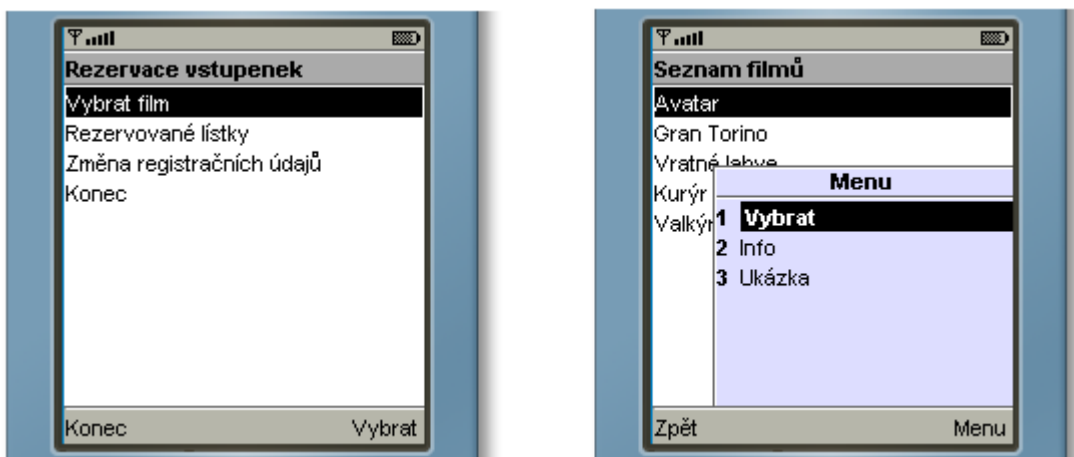
**Obr. B.3: Přihlášení uživatele do systému a žádost o zaslání nového hesla na email**

Není-li ještě uživatel registrován v systému pro rezervaci vstupenek, může tak učinit po volbě položky *Registrace* z menu a vyplněním formuláře, jak je ukázáno na obrázku B.4.



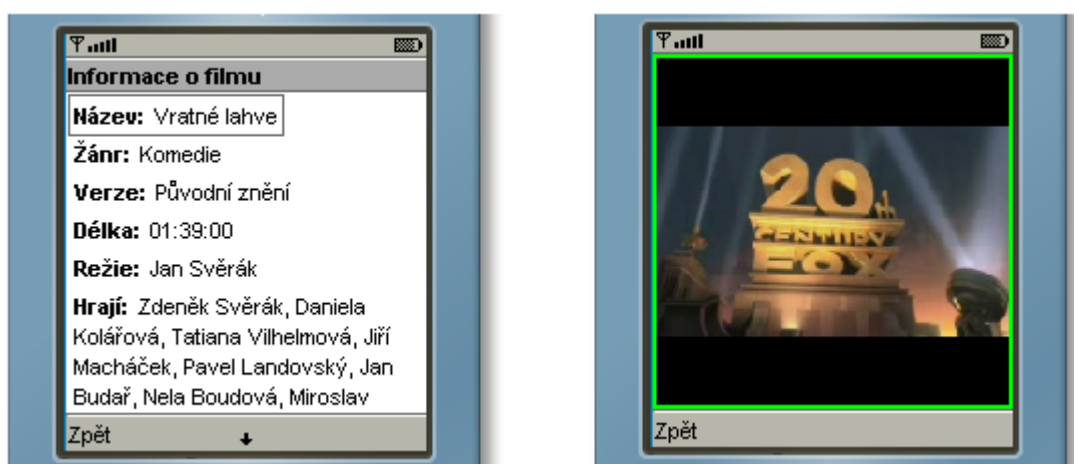
**Obr. B.4: Nová registrace uživatele**

Po úspěšném přihlášení do systému pro rezervaci vstupenek, si uživatel může ze zobrazené nabídky na displeji mobilního zařízení, po výběru položky *Vybrat film*, nechat zobrazit seznam promítaných filmů. (viz obr. B.5).



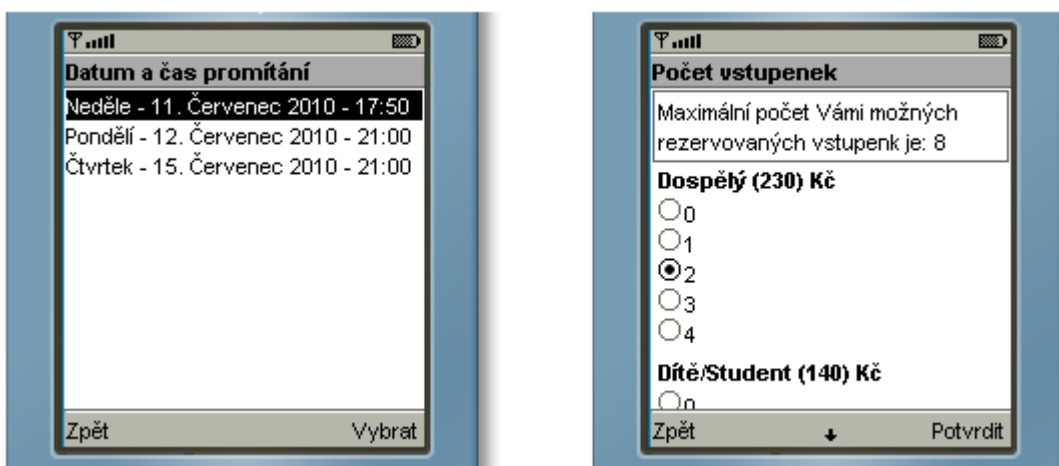
**Obr. B.5: Hlavní menu aplikace a menu pro výběr filmu**

K vybranému filmu si lze nechat vypsát na displej mobilního zařízení informace o promítaném filmu nebo nechat přehrát krátkou filmovou ukázkou (viz obr. B.6).



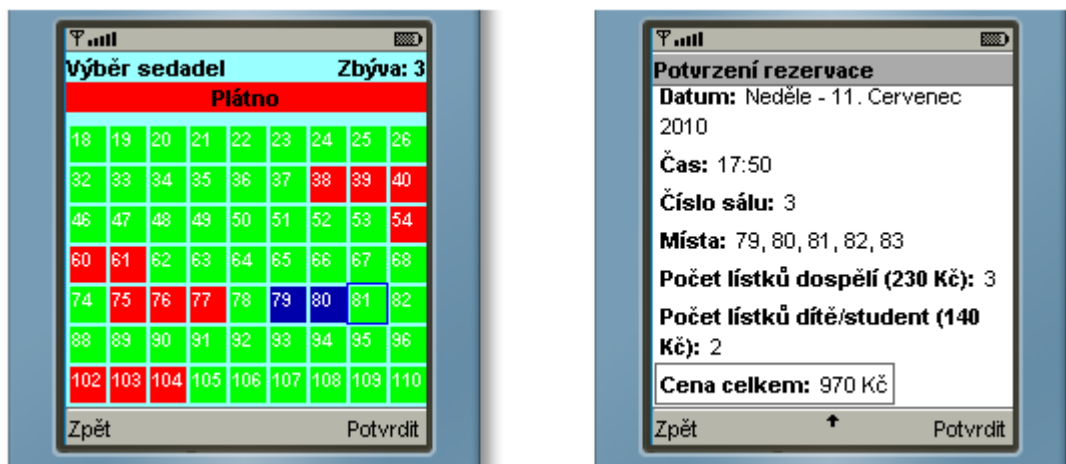
**Obr. B.6: Zobrazení informací o filmu a přehrávání ukázky k filmu**

Vybere-li si klient zvolením položky *Vybrat* z menu film, který by rád viděl, zobrazí se mu na displeji seznam, v jaký den a v jakou dobu je film promítán. Mobilnímu zařízení jsou posílány pouze takové doby promítání, které se uskuteční od následujícího dne, než je prováděna rezervace. Zvolí-li si klient vhodnou dobu promítání, může si v dalším kroku rezervace zvolit, kolik vstupenek by si rád rezervoval. Při jedné rezervaci může být maximální počet rezervovaných vstupenek osm (při menším počtu volných míst v promítacím sále i méně). Tyto kroky prováděné rezervace jsou ukázány na obrázku B.7.



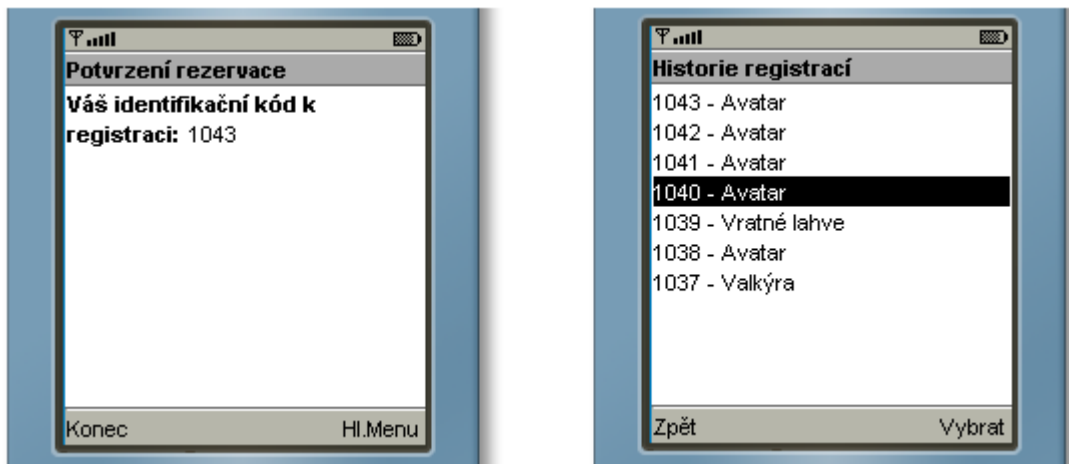
**Obr. B.7: Výběr data a času promítání a volba počtu vstupenek k rezervaci**

Dále si uživatel může vybrat místa k sezení v promítacím sále (viz obr. B.8). Na displeji mobilního zařízení se zobrazí schéma promítacího sálu, kde si uživatel může pomocí kurzoru vybrat místa, kde by rád seděl. Zelenou barvou jsou značena políčka s volnými místy, červenou s již obsazenými místy a modrou jsou vyznačena místa vybraná uživatelem. Po výběru míst k sezení se na displeji objeví stručné shrnutí prováděné rezervace a následuje její potvrzení.



**Obr. B.8: Výběr míst v promítacím sále a potvrzení rezervace vstupenek**

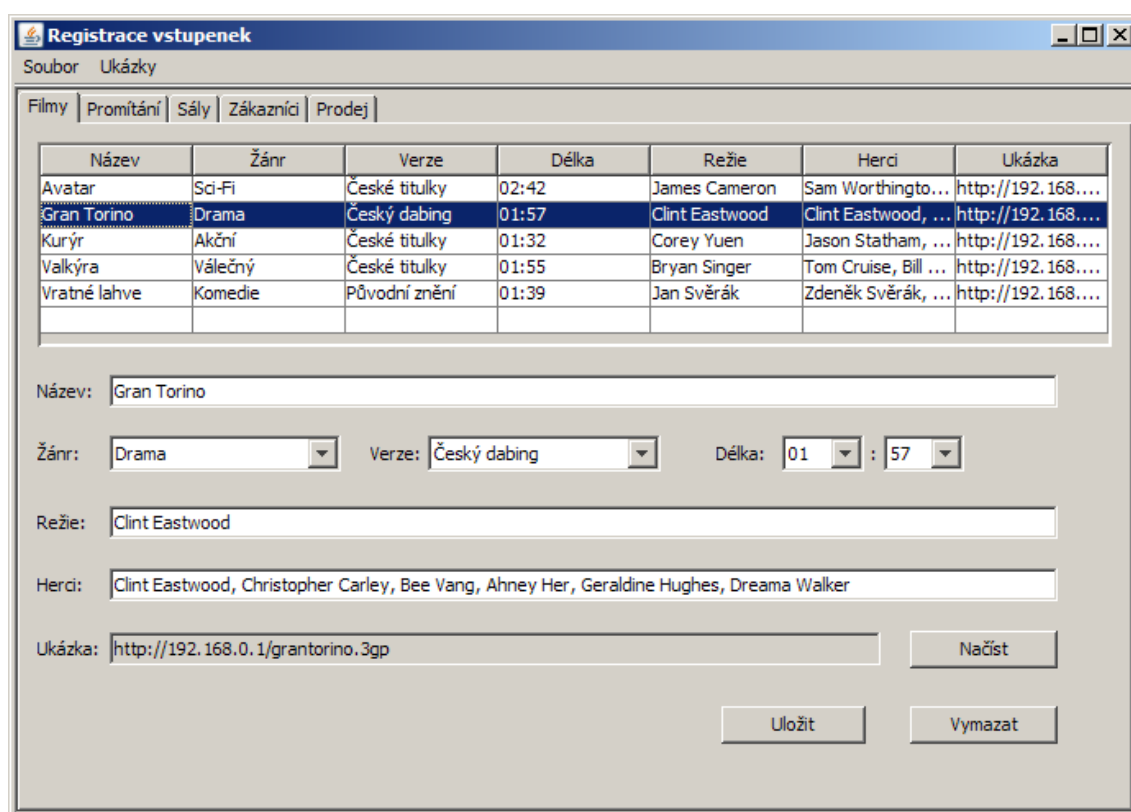
Po potvrzení rezervace vstupenek bude k provedené rezervaci přiřazen identifikační kód, jehož znalostí se klient může prokázat při výdeji vstupenek. Uživatel si může kdykoliv později z nabídky menu, po volbě položky *Rezervované lístky*, nechat podrobně vypsát záznamy o všech jim provedených rezervacích (viz obr B.9).



**Obr. B.9: Přiřazení identifikačního kódu rezervace a historie provedených rezervací**

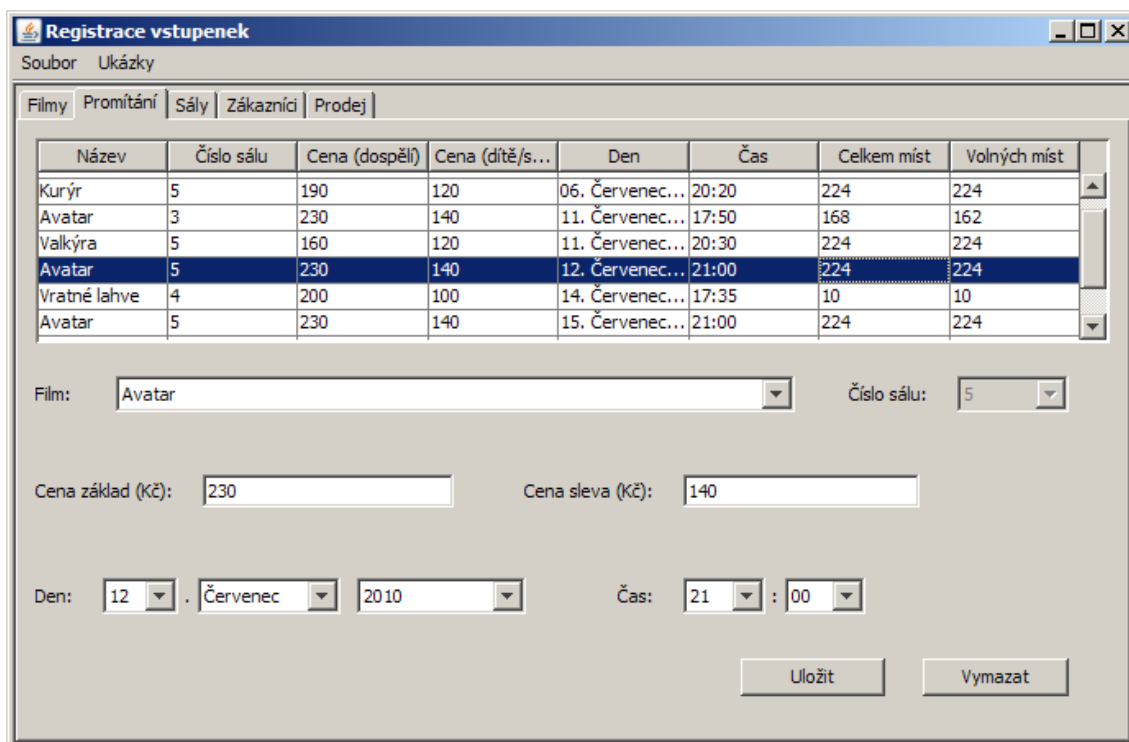
## C. Ukázky aplikace pro správu databáze

Po spuštění aplikace pro správu databáze se na obrazovce osobního počítače objeví okno programu tak, jak je ukázáno na obrázku C.1. Pomocí pěti záložek umístěných v horní části okna se dá přepínat mezi jednotlivými sekcemi programu. V první záložce *Filmy* lze přidávat, mazat a upravovat uložené filmy v databázi. Nový film se přidá do databáze takovým způsobem, že musí být nejdříve vybrán prázdný řádek (poslední řádek) tabulky. Uložení záznamu nebo jeho změny se provede stiskem tlačítka *Uložit*. Má-li být záznam z databáze vymazán, stane se tak po stisku tlačítka *Vymazat*. Ke každému filmu lze přiřadit krátkou filmovou ukázkou umístěnou na webovém serveru. Přiřazení této ukázky k filmu lze provést po stisku tlačítka *Načíst*.



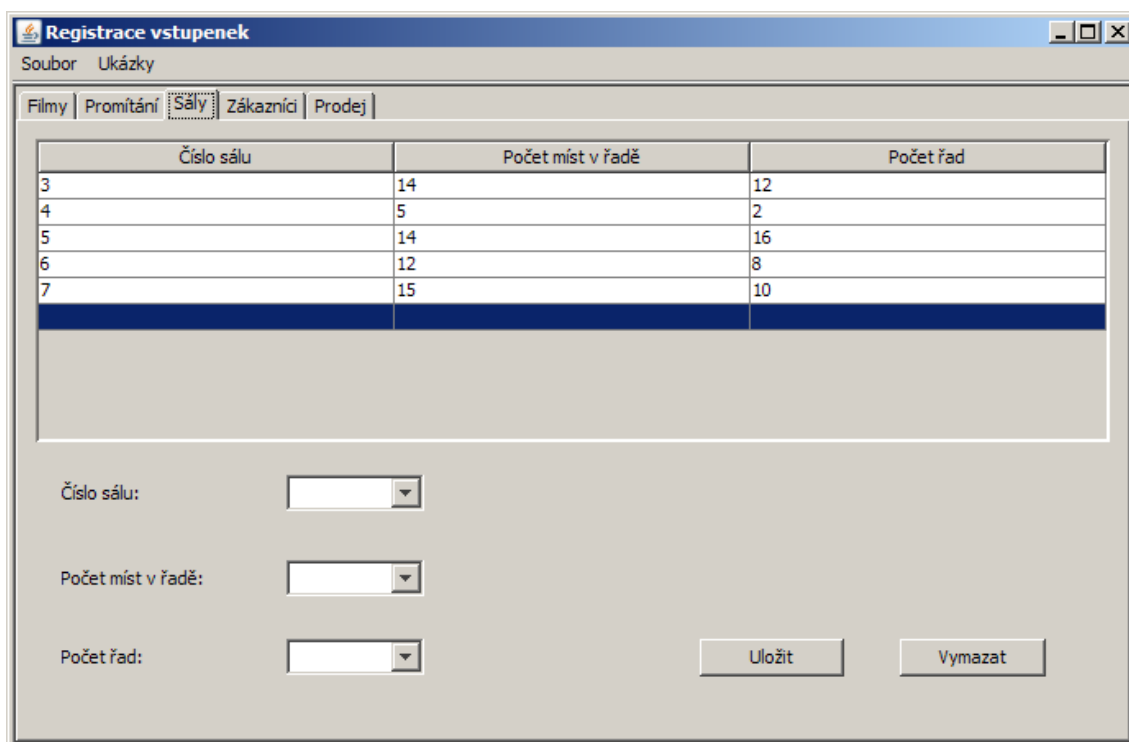
Obr. C.1: Sekce programu pro správu filmů

V další sekci programu, dostupnou pod záložkou pojmenovanou jako *Promítání*, lze jednotlivým filmům přiřadit jejich promítací dobu (datum a čas), cenu vstupenek (dospělí/dítě) a promítací sál (viz obr. C.2). Nový záznam do tabulky může být opět přidán po výběru prázdného řádku tabulky (poslední řádek).



**Obr. C.2: Sekce programu pro správu promítání**

Každému promítacímu sálu může být přiřazeno číslo a nastavena kapacita sálu, které je definována počtem řad a počtem míst v řadě v sále (viz obr. C.3). Z kapacity sálu je pak odvozen počet vstupenek, které budou dispozici k rezervaci.



**Obr. C.3: Sekce programu pro správu sálů**



Další sekce programu, nacházející se pod záložkou *Zákazníci*, slouží k prohlížení a úpravě uživatelských účtů klientů (viz obr. C.4). Po stisku tlačítka *Registrace*, se na obrazovce osobního počítače zobrazí tabulka s doposud provedenými rezervacemi vstupenek vybraného uživatele (viz obr. C.5).

Jméno	Login	Heslo	Email	Telefon	Sip URI
František Vomáčka	franta	f61c4d1a1fc8f761749...	franta@ericsson.com	22222222	sip:franta@ericsson.com
Jan Novák	jan	fa27ef3ef6570e32a79...	jan.novak@ericsson....	11111111	sip:jan@ericsson.com
Pavel Otoupalík	pavel	ef1652b79c940145b6...	pavel.otoupalik@sez...	728161390	sip:pavel@ericsson.com

Jméno: Pavel Otoupalík

Login: pavel

Heslo: ef1652b79c940145b600de7a2fe0288e

Email: pavel.otoupalik@seznam.cz

Telefon: 728161390

SIP URI: sip:pavel@ericsson.com

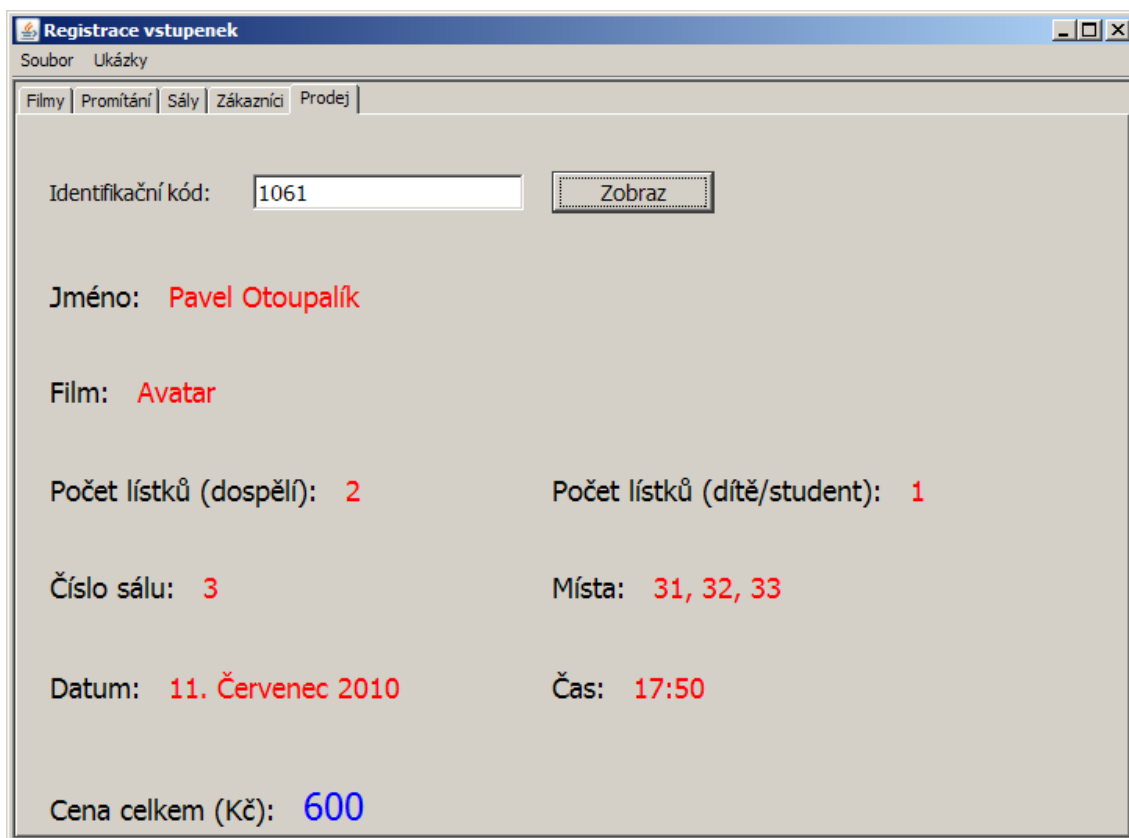
Registrace Uložit Vymazat

**Obr. C.4: Sekce programu pro správu uživatelských účtů**

Identifik...	Název fil...	Den	Čas	Cena cel...	Počet lís...	Počet lís...	Číslo sálu	Místa
1062	Avatar	11. Červ...	17:50	690	3	0	3	77, 76, 6
1061	Avatar	11. Červ...	17:50	600	2	1	3	31, 32, 33

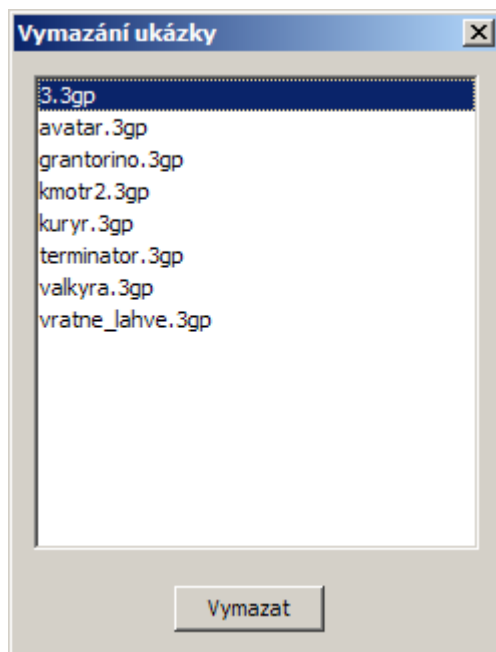
**Obr. C.5: Rezervované vstupenky vybraného uživatele**

Na poslední kartě, pojmenované jako *Prodej*, lze po zadání identifikačního kódu rezervace provést výdej vstupenek (viz obr. C.6).



Obr. C.6: Sekce programu pro prodej lístků

Výběrem položky *Ukázky* z hlavního menu programu, mohou být nahrávány a mazány krátké filmové ukázky na webový server. Tyto ukázky musí být ve video formátu 3gp. Formulář pro mazání filmových ukázek z webového serveru je ukázán na obrázku C.7.



Obr. C.7: Vymazání filmové ukázky z webového serveru

Z hlavního menu, po výběru položky *Soubor -> Nastavení*, lze provést konfiguraci programu (viz obr. C.8). Pomocí tohoto formuláře mohou být nastaveny parametry určené pro přístup aplikace do databáze MySQL, k webovému serveru a poštovnímu serveru. Nastavené parametry jsou uloženy do textového souboru *setting.txt*, který se nachází ve stejném adresáři jako samotná aplikace

**Nastavení**

Databáze:

Adresa: 192.168.0.1

Port: 3306

Název: tickets

Uživatel: ticketscontrol

Heslo: ticketscontrol

Webový server:

Adresa: 192.168.0.1

Port: 9080

Email:

Adresa: tickets.ericsson@gmail.com

Uživatel: tickets.ericsson

Heslo: ticketsericsson

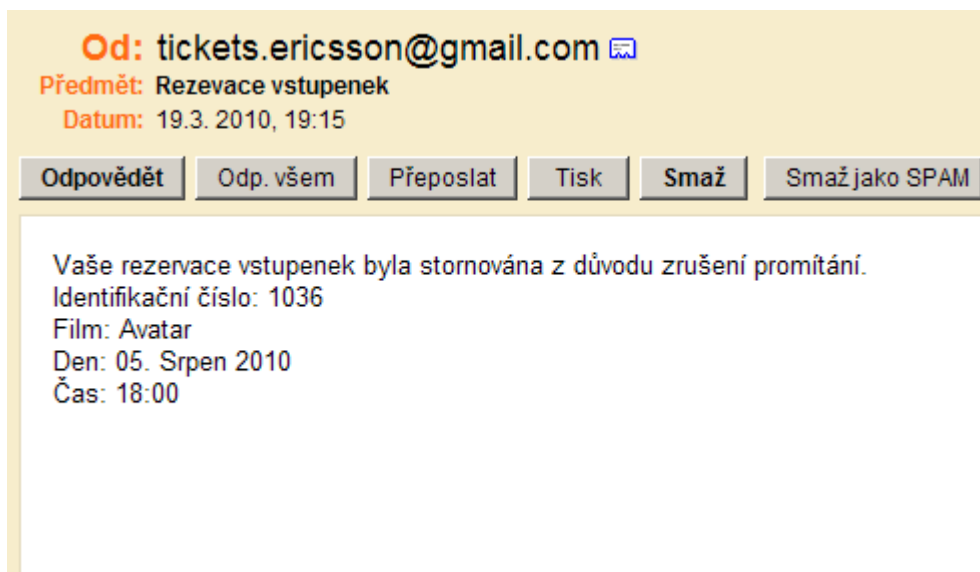
SMTP: smtp.gmail.com

Port: 465

Zpět OK

**Obr. C.8: Nastavení aplikace pro správu databáze**

Pokud obsluha programu provede zrušení promítání filmu, objeví se na obrazovce monitoru dialog umožňující zaslat email o zrušeném promítání těm klientům, kteří měli pro toto promítání uskutečněnou rezervaci vstupenek (viz obr. C.9).



**Obr. C.9: Emailová zpráva informující o zrušeném promítání**

## D. Obsah příloženého DVD

Složka / soubor	Popis
TicketsClient	Klientská aplikace určena mobilním zařízením s podporou platformy J2ME umožňující se klientovi přihlásit do IMS sítě a využívat služeb systému pro rezervaci vstupenek. Importovat v SDS.
TicketsServlet	SIP Servlet aplikace, která komunikuje s klientskou aplikací prostřednictvím IMS sítě. Aplikace má přístup do MySQL databáze. Aplikaci je nutné umístit do SIP kontejneru aplikačního serveru GlassFish rozšířený o projekt SailFin. Importovat v SDS.
TicketsControl	Aplikace pro správu MySQL databáze. Umožňuje přidávat, upravovat nebo odebírat záznamy ve vytvořené databázi <i>tickets</i> . Importovat v NetBeans.
SamplesServlet	HTTP Servlet aplikace, která umožňuje nahrávat a mazat filmové ukázky na webový server. Aplikaci je nutné umístit do HTTP kontejneru aplikačního serveru GlassFish. Importovat v SDS.
Databáze	Exportovaná MySQL databáze <i>tickets</i> s vytvořenými uživatelskými účty. V databázi jsou pro ukázkou již vloženy záznamy. Importovat pomocí nástroje phpMyAdmin.
SDS Settings	Vyexportované nastavení simulované IMS sítě tak, jak bylo popsáno v příloze A. Složku <i>database</i> zkopírovat do C:\Ericsson\SDS4.1FD1\Simulators\CSCF a soubor <i>dns.xml</i> do C:\Ericsson\SDS4.1FD1\Simulators\DNS.
Filmové ukázky	Filmové ukázky ve formátu 3gp, které lze nahrát na webový server pomocí aplikace <i>TicketsControl</i> .
Programy a knihovny	
SDS 4.1 FD1	Vývojové prostředí SDS 4.1 FD1 od firmy Ericsson.
NetBeans	Vývojové prostředí NetBeans IDE 6.8.
Wireless ToolKit	Wireless ToolKit 2.5.2_01 for CLDC.
SE SDK	Sony Ericsson SDK for the Java ME platform 2.5.0.6.
XAMPP	Balíček XAMPP 1.7.3 obsahující instalaci MySQL databáze 5.1.41 a webového serveru Apache 2.2.14.
JDBC	JDBC ovladač MySQL Connector/J ve verzi 5.1.11. Soubor <i>mysql-connector-java-5.1.11-bin.jar</i> zkopírovat do C:\Ericsson\SDS4.1FD1\Simulators\sailfin\lib.
KXML	Knihovna kXML2 2.3.0.
JavaMAIL	JavaMail 1.4.3 API.
LCRYPTO	Bouncy Castle Crypto API 1.45.
Diplomová práce.pdf	Diplomová práce v elektronické podobě.