



OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN JA LIIKENTEEN ALA

VOIP-TOIMINTOJEN TO- TEUTUS

TEKIJÄ/T: Riku Kettunen

Koulutusala Tekniikan ja liikenteen ala	
Koulutusohjelma/Tutkinto-ohjelma Tietotekniikan koulutusohjelma	
Työn tekijä(t) Riku Kettunen	
Työn nimi VoIP-toimintojen toteutus	
Päiväys	28.5.2018
Sivumäärä/Liitteet	23
Ohjaaja(t) Lehtori Jussi Koistinen, Lehtori Keijo Kuosmanen	
Toimeksiantaja/Yhteistyökumppani(t) NettiTieto Oy	
<p>Tiivistelmä</p> <p>Työn toimeksiantajana toimiva NettiTieto Oy halusi laajentaa heidän soittojärjestelmäänsä lisäämällä VoIP-puhelutoiminnallisuudet heidän valmiina olevien GSM-puhelutoiminnallisuuksien rinnalle. Opinnäytetyön tarkoituksena oli löytää sopiva SIP-kirjasto, jonka avulla voisi soittaa ja vastaanottaa VoIP-puheluita. Tavoitteena oli myös saada yksi onnistunut VoIP-puhelu soitettua soittojärjestelmästä älypuhelimeen.</p> <p>Opinnäytetyö toteutettiin kolmessa vaiheessa. Ensimmäisessä vaiheessa kartoitettiin erilaiset kirjasto ja kehitystyökalu vaihtoehdot, joista yksi valittiin. Toisen vaiheen tarkoitus oli luoda C#-ohjelmointikielinen sovellus, joka käyttää valittua kirjastoa. Tähän sovellukseen tehtiin tilaajan haluamat VoIP-toiminnallisuudet, jotka kolmannessa vaiheessa integroitiin soittojärjestelmään ja muokattiin lopulliseen muotoonsa.</p> <p>Lopputuloksena saatiin hyvä kokonaisuus VoIP-toiminnallisuuksia soittojärjestelmään. Opinnäytetyön tavoitteisiin päästiin.</p>	
Avainsanat VoIP, SIP, Softphone	

Field of Study Technology, Communication and Transport			
Degree Programme Degree Programme in Information Technology			
Author(s) Riku Kettunen			
Title of Thesis Implementation of VoIP features			
Date	28 May 2018	Pages/Appendices	23
Supervisor(s) Mr. Jussi Koistinen, Senior Lecturer and Mr. Keijo Kuosmanen, Senior Lecturer			
Client Organisation /Partners NettiTieto Oy			
<p>Abstract</p> <p>NettiTieto Oy wanted to expand their call system by adding VoIP functionalities alongside with their existing GSM system. The purpose of this thesis was to find a suitable SIP library for making and receiving VoIP calls. The purpose was also to make one successful VoIP call from the call system to a smartphone.</p> <p>The thesis was done in three stages. The purpose of the first stage was to find different libraries and software development kits, to compare them to pick the most suitable one. In the second stage, the purpose was to develop an application that uses the selected library. The application was coded with the C# programming language. In the third stage, all desired VoIP functionalities were integrated from the application to the call system.</p> <p>The final result was a good set of VoIP functionalities for the call system. The goals of the thesis were achieved.</p>			
Keywords VoIP, SIP, Softphone			

ESIPUHE

Haluan kiittää työnantajaa Toni Riekkistä mielenkiintoisesta aiheesta. Lisäksi haluan kiittää työnhajajaa Matti Saarista työn aikana saadusta avusta ja ohjauksesta.

Kuopiossa 28.5.2018

Riku Kettunen

SISÄLTÖ

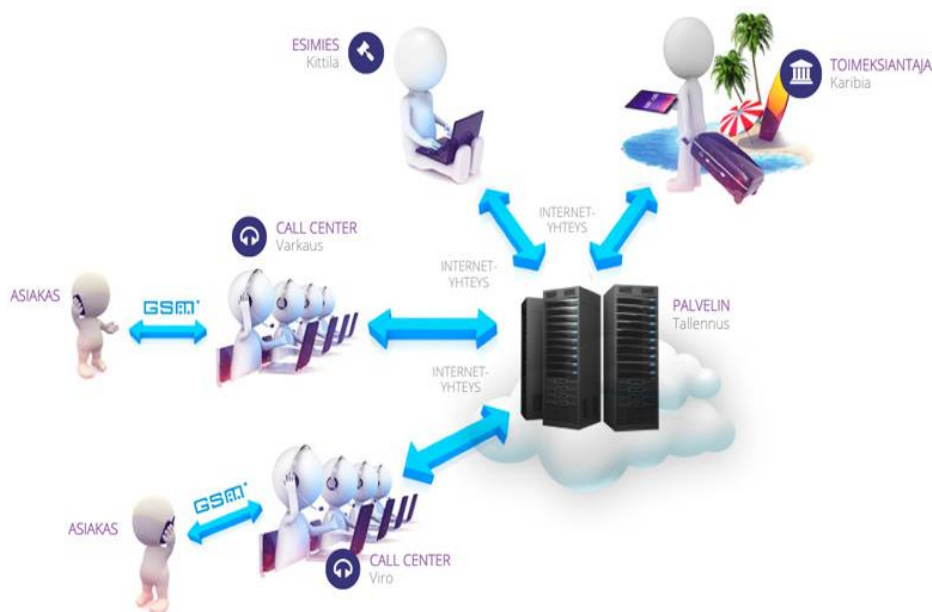
1	JOHDANTO	6
2	LYHENTEET JA MÄÄRITELMÄT	7
3	VOIP	8
3.1	Yleistä	8
3.2	Softphone	8
3.3	Hardphone	9
3.4	H.323 ja SIP	9
3.5	TCP ja UDP	9
3.6	RTP	9
3.7	Äänikoodekit	10
4	SELVITYSTYÖ	11
4.1	Vaatimukset	11
4.2	Kirjastovaihtoehdot	11
4.2.1	PortSIP VoIP SDK	12
4.2.2	ABTO VoIP SIP SDK	12
4.3	Kirjaston valinta	13
5	PROOF OF CONCEPT	14
5.1	Kirjaston rajapinta	14
5.2	Lähtevä puhelu	16
5.3	Saapuva puhelu	17
5.4	Puhelun lopetus	17
5.5	DTMF	17
5.6	Ääniasetukset	18
6	SOITTOJÄRJESTELMÄÄN INTEGROINTI	20
6.1	Toiminnot	20
7	YHTEENVETO	21
	LÄHTEET JA TUOTETUT AINEISTOT	22

1 JOHDANTO

Opinnäytetyöhön aihe saatiin NettiTieto Oy:ltä, koska he halusivat laajentaa heidän soittojärjestelmä sovellustaan. Soittojärjestelmässä on jo valmiina käyttöliittymä ja GSM-puhelutoiminnallisuudet. Tämän opinnäytetyön päätavoite on saada VoIP-toiminnallisuudet GSM-toiminnallisuuksien rinnalle. Nykyisen soittojärjestelmän toimintamallin näkee kuvasta 1.

Jotta tavoitteeseen päästään, työ on tehtävä vaiheissa. Opinnäytetyön ensimmäinen vaihe oli selvitystyö, jossa etsitään ja vertaillaan erilaisia SIP VoIP -kirjastoja ja kehitysokaluja. Toiseen vaiheeseen kuului Proof of Concept. Tässä vaiheessa tarkoitus oli tehdä demosovellus, jonka avulla voi esittää tilaajalle, että valittu kirjasto oli oikea ratkaisu. Kolmanteen vaiheeseen kuului VoIP-toiminnallisuuksien integroiminen demosovelluksesta soittojärjestelmään.

NettiTieto Oy on vuonna 2003 perustettu tietoyhteiskunnan arkkitehtuuriyritys. NettiTieto Oy on erikoistunut sähköisen liiketoiminnan palveluiden kehittämiseen ja ohjelmisto- ja viestintäratkaisujen tuottamiseen. (NettiTieto Oy)



KUVA 1. Soittojärjestelmän toimintamalli (NettiTieto Oy)

2 LYHENTEET JA MÄÄRITELMÄT

ASCII = American Standard Code for Information Interchange. Merkistön koodausstandardi, jossa jokaiselle merkillä on annettu numeerinen arvo.

DTMF = Dual-Tone Multi Frequency. Telekommunikaatiossa käytetty signaalin lähetystapa, joka muodostuu kahden eritaajuuden äänimerkin yhdistelmästä.

Hardphone = Fyysinen puhelin, johon on sisäänrakennettu VoIP-toiminnallisuus.

IP = Internet Protocol. Protokolla, joka mahdollistaa datapakettien toimitukset pakettikytkentäisessä verkossa.

Kirjasto = Paketti, joka sisältää aliohjelmia ja luokkia.

PBX = Private Branch Exchange. Yrityksen sisäinen puhelinverkkojärjestelmä.

RTP = Real-time Transport Protocol. Tietoliikenneprotokolla, joka mahdollistaa reaaliaikaisen äänen ja videon lähetyksen.

SDK = Software Development Kit. Joukko työkaluja, jotka helpottavat ohjelmiston kehitystä.

SIP = Session Initiation Protocol. Tietoliikenneprotokolla, jonka avulla muodostetaan ja hallitaan puheluyhteyttä.

Softphone = VoIP-sovellus älylaitteeseen.

SRTP = Secure Real-time Transport Protocol. Tietoliikenneprotokolla, joka mahdollistaa turvallisen reaaliaikaisen äänen ja videon lähetyksen.

TCP = Transmission Control Protocol. Tietoliikenneprotokolla, jonka avulla voidaan luotettavasti lähettää tietopaketteja verkon yli.

UDP = User Datagram Protocol. Tietoliikenneprotokolla, jonka avulla voidaan lähettää tietopaketteja ilman vahvistettua yhteyttä.

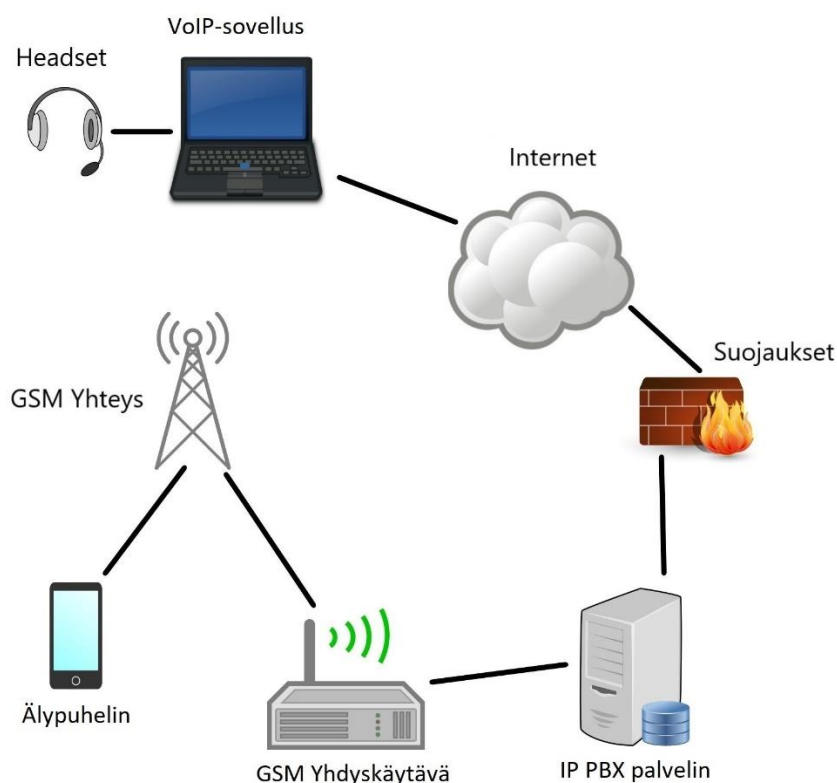
VoIP = Voice over Internet Protocol. Tekniikka, joka mahdollistaa reaaliaikaisen ääni- ja videokeskustelun internetin kautta.

Älylaite = Älypuhelin, tabletti tai tietokone.

3 VOIP

3.1 Yleistä

Voice over Internet Protocol eli IP-puhe on tekniikka, joka sisältää joukon protokollia, jotka mahdollistavat reaaliaikaisen äänen ja videon lähetyksen IP-verkon yli (Rouse). VoIP-puheluita voidaan soittaa älylaitteeseen tai IP-puhelimeen. VoIP-tekniikka mahdollistaa myös puhelut, joissa on enemmän kuin kaksi henkilöä. Kuvassa 2 näytetään VoIP-yhteys tietokoneelle asennetun VoIP-sovelluksen ja älypuhelimien välillä. Isoin ero GSM ja VoIP-puheluiden välillä on äänen siirtotapa. VoIP:ssa ääni muutetaan digitaalisiksi ja se lähetetään datapaketteina IP-verkon kautta vastaanottajalle. (Digium) Vastaanottajan päässä digitaaliset datapaketit muunnetaan takaisin analogiseksi ääneksi ja toistetaan älylaitteeseen.



KUVA 2. VoIP-yhteys softphonesta älypuhelimeen

3.2 Softphone

Softphonella tarkoitetaan VoIP-sovellusta, joka asennetaan haluamaansa älylaitteeseen. Näissä sovelluksissa on graafinen käyttöliittymä, joka helpottaa puhelujen soittamista. (NFON) Softphonea käyttäksesi riittää, että sinulta löytyy älylaite, internetyhteys ja headset. Olet saattanut käyttää tai kuulla VoIP-sovelluksista kuten Skype, Linnphone tai Discord. Nämä ovat sovelluksia, jotka käyttävät VoIP-viestintää. (Corpuz)

3.3 Hardphone

Hardphone eli IP-puhelin muistuttaa normaalia puhelinta, mutta siihen on sisäänrakennettu VoIP-toiminnallisuus. IP-puhelin on joko langattomasti tai Ethernet-verkkokaapelilla kytketty lähiverkkoon. IP-puhelin on softphonea kalliimpi vaihtoehto, koska IP-puhelin täytyy ostaa valmistajalta. Yritykset voivat käyttää IP-puhelinta kommunikaatioon työntekijöiden tai asiakkaiden välillä, vaikka he olisivat toisella puolella maapalloa. (Kelly)

3.4 H.323 ja SIP

VoIP:lla on kaksi yleisesti käytettyä merkinantoprotokollaa. Ensimmäinen niistä on H.323-signaalointiprotokolla. H.323-protokolla on yksi ITU-T:n (International Telecommunication Union - Telecommunication Standardization Sector) määrittämä standardi, joka antaa alustan äänen ja videon lähetykseen IP-verkkojen yli (3cx).

Session Initiation Protocol eli SIP on toinen laajasti VoIP:ssa käytetty protokolla, joka on alkanut syrjäyttämään H.323-protokollaa. SIP-protokollan standardoinnista vastaa IETF (The Internet Engineering Task Force). Tätä protokollaa käytetään puhelun session hallitsemiseen. SIP on tekstipohjainen protokolla, ja siksi se on helposti ymmärrettävä. (Porter, 2006 ss. 146-147) SIP ja H.323 protokolla käyttävät TCP- tai UDP-protokollia tiedon lähetykseen. Itse äänen ja videon toimitus vastaanottajalle toimii RTP- tai SRTP-protokollien avulla. (Porter, 2006 s. 9)

3.5 TCP ja UDP

TCP-protokolla muodostaa yhteyden laitteiden välillä ja varmistaa, että tietopaketit tulee perille toiseen päähän. Tämä protokolla mahdollistaa, että lähetetyt tietopaketit tulee täydellisenä toiselle ja oikeassa järjestyksessä. (OnSIP) TCP-protokolla ei sovellu reaaliaikaiseen datan lähetykseen, koska siihen voi tulla isojakin viiveitä datan varmistuksen takia.

UDP-protokolla lähettää tietopaketteja suoraan toiselle päätelaitteelle ilman yhteyden muodostamista. Tämä protokolla ei varmista, että tietopaketti on saapunut vastaanottajalle, vaan lähettää jatkuvalla syötöllä tietopaketteja. Koska tietopakettien lähetystä ei varmisteta, niin ei voi olla varmoja saapuuko tietopaketit vastaanottajalle oikeassa järjestyksessä vai saapuuko ne ollenkaan. VoIP-puheluihin suositellaan UDP-protokollaa juuri jatkuvan tietopakettien syötön takia. (OnSIP)

3.6 RTP

RTP-protokollan tarkoitus on mahdollistaa äänen ja videon siirron vastaanottajalle reaaliajassa. Tämä protokolla toimii parhaiten, kun sitä käytetään UDP-protokollan kanssa. (Rouse) SRTP-protokolla on vaihtoehtoinen tapa siirtää mediaa, joka antaa lisää suojausta RTP-protokollaan nähden (Rouse).

3.7 Äänikoodekit

Äänikoodekkien tarkoitus on pakata analoginen ääni pienemmäksi digitaaliseksi paketiksi ja purkaa se takaisin analogiseksi ääneksi (Martin, 2017). Äänikoodekin pakkaustapa voi olla häviötön tai häviöllinen. Häviöttömässä pakkaustavassa ääni puretaan täysin samanlaisena kuin se on pakattu. Häviöllisessä pakkaustavassa ääni pakataan tiukempaan ja tämän takia äänenlaatu kärsii, kun se puretaan takaisin. (Moore, 2017) VoIP:ssa yleisesti käytettyjä äänikoodekkeja ovat muun muassa G.711, G.729 ja GSM (TutorialsPoint).

4 SELVITYSTYÖ

4.1 Vaatimukset

Aloituspalaverissa kävi selväksi, että tilaaja halusi C# tai C++ kielisen VoIP-kirjaston, joka mahdollistaisi VoIP-puhelujen soittamisen ja vastaanottamisen. Ohjelmointikieli C# oli laitettu ensisijalle kirjastojen selvitystyössä. Kirjaston pitää käyttää SIP-protokollaa muodostamaan VoIP-puhelu. Kirjaston selvityksessä piti ottaa selvää muun muassa, onko kirjasto luotettava, onko kirjastosta olemassa Visual Studio NuGet-pakettia, mitä lisenssiä kirjasto käyttää, mitä toiminnallisuuksia kirjastolla on ja onko kirjaston lähdekoodi avoin.

4.2 Kirjastovaihtoehdot

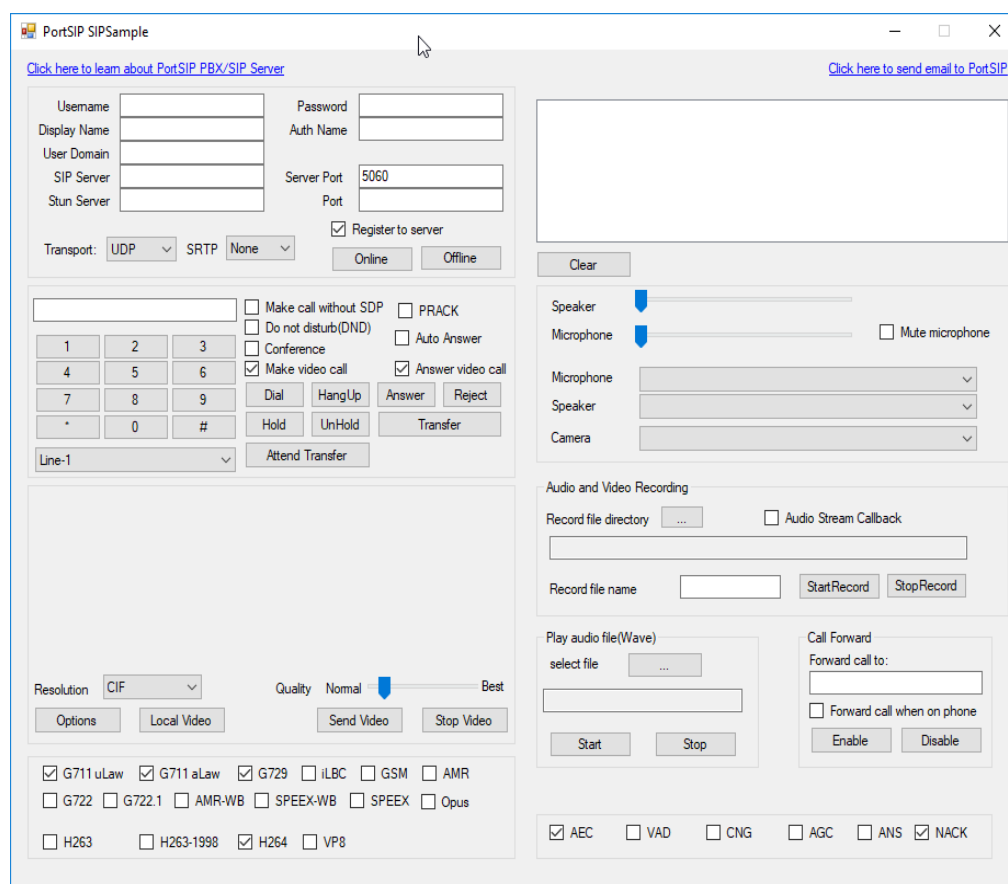
Selvitetyistä SIP VoIP -kirjastoista ja kehitystyökaluista osa ei ole enää kehityksessä, mutta niistä silti voisi tehdä toimivan VoIP-sovelluksen. Kirjastojen karsiutumiseen oli monia syitä. Syiksi voidaan luetella esimerkiksi seuraavat: kirjasto ei ole enää kehityksessä, huono maine, hinta, dokumentaation puute, vähäiset toiminnot tai lisenssi. Taulukossa 1 näytetään osa selvitetyistä kirjastoista ja kehitystyökaluista mukaan lukien myös niitä, jotka eivät vaikuttaneet lupaavilta.

TAULUKKO 1. Kirjastot ja kehitystyökalut

Kirjasto / SDK	Avoin lähdekoodi	Visual Studio NuGet	Jatkuvassa kehityksessä	Lisenssi	Syy(t) karsintaan
Lumisoft	Kyllä	Kyllä	Ei	CPOL	Ei kehityksessä, huono dokumentaatio
Ozeki	Ei	Kyllä	Kyllä	Kaupallinen	Lisenssi
PJSIP	Kyllä	Ei	Kyllä	Dual-lisenssi (Kaupallinen ja GPL versio 2)	Ohjelmointikieli C
SipekSDK	Kyllä, mutta ei enää saatavilla alkuperäiseltä sivustolta	Ei	Kehitys lopetettu vuonna 2010	GPL versio 2 tai myöhempi versio	Ei kehityksessä, lisenssi
Sipek PJSIP C# wrapper	Kyllä, mutta ei enää saatavilla alkuperäiseltä sivustolta	Ei	Kehitys lopetettu vuonna 2010	PJSIP:n mukainen lisenssi	Ei kehityksessä
PortSIP	Ei	Ei	Kyllä	Kaupallinen	Lisenssin hinta
ABTO VoIP	Ei	Ei	Kyllä	Kaupallinen	

4.2.1 PortSIP VoIP SDK

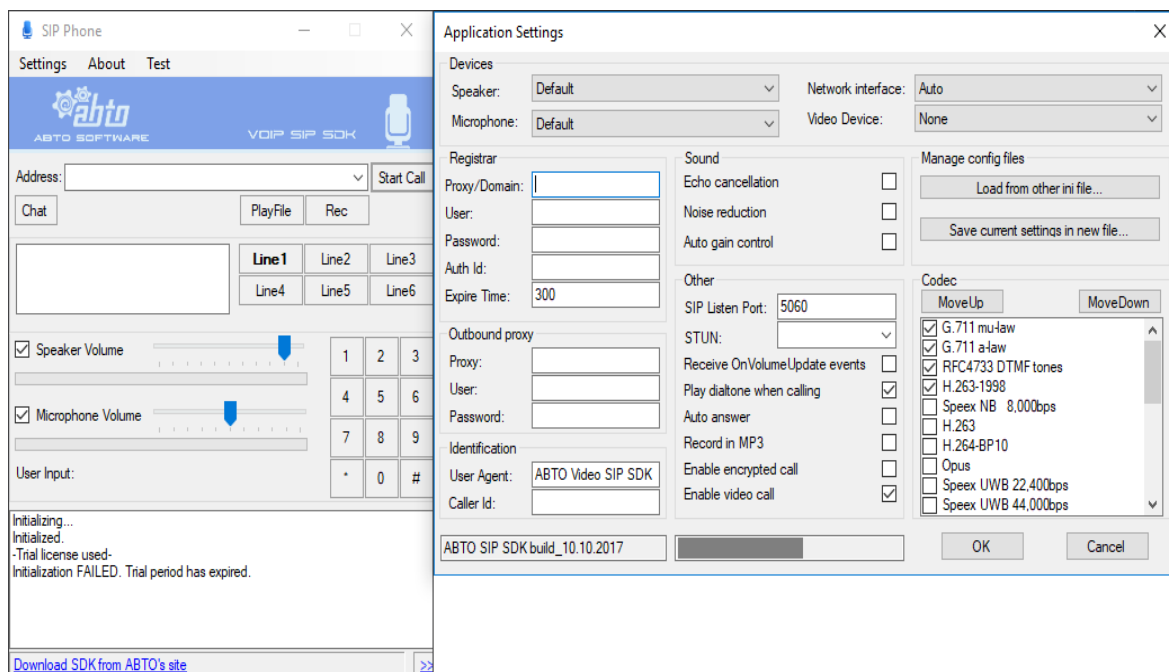
PortSIP:n tarjoama kehitystyökalu on todella kattava ja helposti käytettävä. Tämä kehitystyökalu on tarkoitettu niille, jotka haluavat saada paljon erilaisia toiminnallisuksia omaan sovellukseensa. PortSIP:n kehitystyökalulla voi tehdä VoIP-ohjelmistoja monille eri alustoille, kuten esimerkiksi Androidille, Windowsille ja iOS:lle (PortSIP). PortSIP antaa kehitystyökalun ja demosovelluksen koekäyttöön ennen lisenssin ostamista. Kehitystyökalun mukana tulee myös kattavat dokumentoinnit. Kuvassa 3 näkee kehitystyökalun demosovelluksen käyttöliittymän. Vaikka tämä kirjasto antaa paljon erilaisia toiminnallisuksia, niistä ei isoa osaa tarvitse käyttää.



KUVA 3. PortSIP:n demosovellus

4.2.2 ABTO VoIP SIP SDK

ABTO VoIP SIP SDK on Abto Softwaren tekemä kehitystyökalu, joka antaa ohjelmoijalle hyvän alustan rakentaa oma VoIP-sovellus. Myös ABTO antaa demosovelluksen ja kehitystyökalun koekäyttöön. Koeaika tällä kehitystyökalulla on kymmenen päivää, jonka kuitenkin pystyy uusimaan ottamalla yhteyttä heidän tekniseen tukeen. Tämän kehitystyökalun mukana tulee myös puheluun tarvittavat äänikoodekit. ABTO:n kirjastosta löytyy kaikki toiminnallisuudet, mitä yksinkertaiseen ja toimivaan VoIP-sovellukseen voisi tarvita. Kuvassa 4 näkyy ABTO:n demosovelluksen käyttöliittymä. ABTO:n dokumentaatioissa olisi parantamisen varaa, mutta se on tarpeeksi selkeä, että sitä voi ymmärtää.



KUVA 4. ABTO:n demosovellus

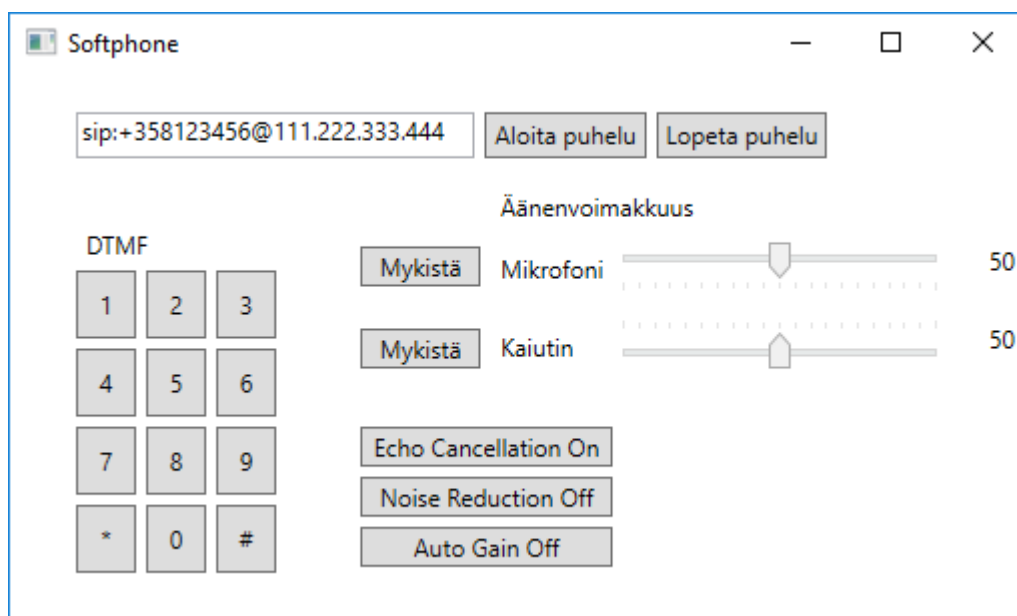
4.3 Kirjaston valinta

Selvitetyistä kirjastoista ja kehitystyökaluista kaksi oli ylitse muiden. Nämä olivat PortSIP VoIP SDK ja ABTO VoIP SIP SDK. Kummassakin kehitystyökalussa on paljon enemmän toiminnallisuuksia kuin työ vaatii, mutta se on hyvä asia. Ylimääräiset toiminnallisuudet kuitenkin mahdollistavat ohjelman jatkekehityksen, jos yritys jossain vaiheessa haluaa lisätä uusia toimintoja sovellukseensa. Kehitystyökalun valinta PortSIP VoIP SDK:n ja ABTO VoIP SIP SDK:n välillä oli tiukka. Valintaan vaikutti vahvasti lisenssin hinta, joka oli ABTO:lla huomattavasti halvempi. Loppujen lopuksi päädyttiin valitsemaan ABTO VoIP SIP SDK.

5 PROOF OF CONCEPT

Proof of Concept on karkeasti suomennettuna demovaihe. Demovaiheessa oli tarkoituksena tehdä oma testisovellus käyttäen valittua kirjastoa. Testisovellus oli tehty Visual Studiolla käyttäen C#-ohjelmointikieltä. Kuvan 5 mukaisesti sovelluksen käyttöliittymästä tuli todella yksinkertainen, jossa on muutamia ääniasetuksien säätöpainikkeita, DTMF-näppäimet ja puhelimen aloitus sekä lopetus näppäimet. Osa demovaiheesta näytetyistä kuvista ja koodeista ovat yksinkertaistetussa ja sensuroidussa muodossa.

Tämän sovelluksen tarkoitus oli testata ABTO VoIP SDK:n kirjaston ominaisuuksia ja osoittaa, että valittu kirjasto on toimiva kokonaisuus. Kirjaston käyttö oli aluksi hieman hankalaa, koska kaikkia toimintoja ei löydy dokumentaatiosta. Demovaiheen tavoitteena oli saada yksi onnistunut VoIP-puhelu soitettua sovelluksesta älypuhelimeen. NettiTieto Oy:lla oli demovaihetta varten valmiina IP PBX-palvelin ja GSM-yhdyskäytävä, jonka avulla puhelut saatiin ohjattua GSM-verkkoon.



KUVA 5. Demovaiheen sovellus

5.1 Kirjaston rajapinta

ABTO:n kirjaston rajapinta on sovelluksessa alustettu AbtoPhone nimiseen muuttujaan. Ohjelman käynnistymisen alkuvaiheessa täytyy alkaa kuuntelemaan rajapinnan antamia tapahtumia. Kuvasta 6 näkee tarvittavat tapahtumankäsittelijät, jotka mahdollistavat puhelun hallinnan.

OnInitialized ja OnRegistered tapahtumat laukaistaan kirjaston puolella ja ne palauttavat virheviestin, jos jokin on mennyt pieleen kirjaston alustamisessa tai rekisteröinnissä. OnEstablishedConnection ja OnClearedConnection mahdollistavat puhelulinjojen hallinnan. OnEstablishedCall tapahtuma laukaistaan aina kun puheluun vastataan ja OnClearedCall laukaistaan, kun puheluun ei vastata tai puhelu sammutetaan. OnIncomingCall tapahtuma laukaistaan, jos joku yrittää soittaa sovellukseen. (Abto Software)

```
// Tapahtumakäsittelijät
// Kirjaston alustus ja rekisteröinti
this.AbtoPhone.OnInitialized += new _IAbtoPhoneEvents_OnInitializedEventHandler(this.AbtoPhone_OnInitialized);
this.AbtoPhone.OnRegistered += new _IAbtoPhoneEvents_OnRegisteredEventHandler(this.AbtoPhone_OnRegistered);

// Puhelun aloitus
this.AbtoPhone.OnEstablishedConnection += new _IAbtoPhoneEvents_OnEstablishedConnectionEventHandler(this.AbtoPhone_OnEstablishedConnection);
this.AbtoPhone.OnEstablishedCall += new _IAbtoPhoneEvents_OnEstablishedCallEventHandler(this.AbtoPhone_OnEstablishedCall);

// Saapuva puhelu
this.AbtoPhone.OnIncomingCall2 += new _IAbtoPhoneEvents_OnIncomingCall2EventHandler(this.AbtoPhone_OnIncomingCall2);

// Puhelun lopetus
this.AbtoPhone.OnClearedConnection += new _IAbtoPhoneEvents_OnClearedConnectionEventHandler(this.AbtoPhone_OnClearedConnection);
this.AbtoPhone.OnClearedCall += new _IAbtoPhoneEvents_OnClearedCallEventHandler(this.AbtoPhone_OnClearedCall);
```

KUVA 6. Tapahtumakäsittelijät

Jotta sovellusta voi käyttää softphonenä, niin on alustettava muutama rajapinnan asetus SIP-palvelimen tiedoilla. Halutessaan asetuksiin voi lisätä muun muassa käytettävien porttien numerot, käyttäjagentin ja lokikirjoitustason (Abto Software). Kuvasta 7 näkyy yksinkertaistettu rajapinnan asetusten alustus. Näihin asetuksiin täytyy laittaa

- Toimialue
- Käyttäjänimi
- Salasana
- Tiedon lähetystapa
- Kirjaston lisenssi ID
- Kirjaston lisenssiavain.

```
CConfig softPhoneConfiguration = AbtoPhone.Config;
//softPhoneConfiguration.Load("config.ini");

softPhoneConfiguration.RegDomain = "123.123.123.123";
softPhoneConfiguration.RegUser = "1234";
softPhoneConfiguration.RegPass = "1234567890";
softPhoneConfiguration.RegExpire = 300;

softPhoneConfiguration.SignallingTransport = (int)TransportType.eTransportUDP;
softPhoneConfiguration.LogLevel = LogLevelType.eLogDebug;

softPhoneConfiguration.LicenseUserId = ██████████
softPhoneConfiguration.LicenseKey = ██████████

//softPhoneConfiguration.Store("config.ini");

AbtoPhone.ApplyConfig();
AbtoPhone.Initialize();
```

KUVA 7. Kirjaston rajapinnan asetukset

Asetukset voidaan tallentaa config.ini tiedostoon, jota on myös mahdollista lukea ohjelman käynnistyessä softPhoneConfiguration.Load() funktiolla. Tämä ei kuitenkaan ole suotavaa, koska asetukset sisältävä tiedosto ei ole salattu ja sen takia siitä voi lukea tärkeitä tietoja. Kuvasta 8 näkyy esimerkki config.ini tiedoston sisällöstä. Myös lokiin kirjoitetuista viesteistä pitää olla varuillaan, sillä testaustason ja infotason loki kirjoittaa IP-osoitteita ja palvelimen osoitteita tekstitiedostoon.

```

AutoAnswerEnabled=0
CallInviteTimeoutSeconds=60
DialToneEnabled=1
EncryptedCallEnabled=0
IntenalVolumImplEnabled=-1
LicenseKey=[REDACTED]
LicenseUserId=[REDACTED]
ListenPort=[REDACTED]
LocalAudioEnabled=1
LocalTonesEnabled=1
LogLevel=7
MP3RecordingEnabled=0
MixerFilePlayerEnabled=1
NetworkInterface=Auto
OutgoingTcpPort=0
PlaybackDevice=Default
RecordDevice=Default
RegDisplName=Riku
RegDomain=[REDACTED]
RegExpire=300
RegPass=[REDACTED]
RegUser=[REDACTED]
RingToneEnabled=0
RingToneFile=telephone-ring.wav
RingerDevice=
RtpStartPort=[REDACTED]
SamplesPerSecond=48000
SignallingTransport=3
StunAddress=
UserAgent=RikunUserAgent
VideoCallEnabled=1
VideoDevice=None
VolUpdSubscribed=0

```

KUVA 8. Config.ini tiedoston sisältö

5.2 Lähtevä puhelu

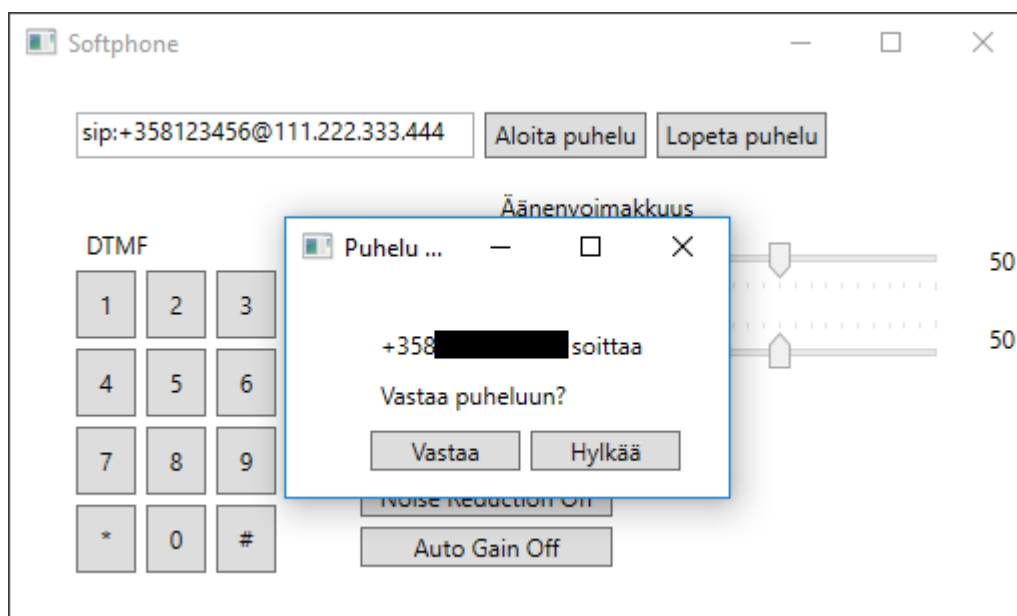
Puhelu aloitetaan siten, että kirjoitetaan kuvassa 5 näkyvään tekstikenttään vastaanottajan SIP-osoite. SIP-osoite on muodossa: sip:extension@server:port (Abto Software). Jos SIP-osoite on oikein kirjoitettu niin AbtoPhone.startCall() funktion avulla voi lähettää ABTO:n kirjastolle kutsun, joka aloittaa SIP-kättelyn IP PBX-palvelimen kanssa. Joissakin tapahtumissa kirjasto palauttaa SIP-vastauskoodin. Vastauskoodista saa selville esimerkiksi, jos puheluun ei vastattu, puhelu hylättiin tai puhelu sammutettiin onnistuneesti.

Kun kirjasto on saanut yhteyden vastaanottajaan, se laukaisee OnEstablishedConnection-tapahtuman. Tässä tapahtumassa saadaan tieto, että vastaanottajan puhelimeen on saatu yhteys, mutta

puheluun ei ole vielä vastattu. Puheluun vastattaessa kirjasto laukaisee OnEstablishedCall-tapahtuman, jossa voi aloittaa puhelun nauhoittamisen tai aktivoida ajastimen, joka mittaa puhelun keston.

5.3 Saapuva puhelu

Silloin kun älypuhelimesta soitetaan softphoneen, niin kirjasto laukaisee OnEstablishedConnection-tapahtuman, joka ilmoittaa, että yhteys on luotu yhdelle vapaista puhelulinjoista. Tämän jälkeen kirjasto laukaisee OnIncomingCall-tapahtuman, joka näyttää soittajan ja puhelulinjan. Tässä vaiheessa sovelluksen käyttäjälle alkaa soimaan soittoääni ja eteen ilmestyy kuvan 9 mukainen ponnahdusikkuna, josta voi vastata saapuvaan puheluun tai hylätä puhelun. Puheluun vastatessa OnEstablishedCall-tapahtuma laukaistaan. Käyttäjä saa myös tiedon, jos puhelu on meneillään ja uusi puhelu on saapumassa. Uuteen saapuvaan puheluun on myös mahdollista vastata toisella vapaana olevalla linjalla. Tätä toimintoa ei kuitenkaan toteutettu, sillä sitä ei tarvittu.



KUVA 9. Saapuva puhelu

5.4 Puhelun lopetus

Puhelua lopetettaessa on tarkastettava, että puhelu katkaistaan oikealta puhelulinjalta. Puhelun voi lopettaa `AbtoPhone.HangUpCallLine()` kutsulla. Kun jompikumpi osapuoli on lopettanut puhelun, niin kirjasto pudottaa yhteyden ja laukaisee `OnClearedConnection`-tapahtuman ja sen jälkeen `OnClearedCall`-tapahtuman. Näiden tapahtumien jälkeen on mahdollista soittaa uusi puhelu.

5.5 DTMF

Softphonessa DTMF-näppäimet vastaa normaalin puhelimen numeronäppäimiä, joita painaessa kuuluu dual-tone ääni. Tämä ääni sisältää korkean ja matalan taajuuden äänimerkin. (DialABC) Taulukosta 2 näkee näppäintä vastaavat taajuudet. DTMF-näppäimet ovat tarpeellisia, jos haluaa soittaa palvelunumeroon, jossa on robottivastaaja ja pitää painaa tiettyä näppäintä päästäkseen eteenpäin. Sovelluksessa DTMF-näppäimiä voi käyttää vain, jos puhelu on käynnissä.

TAULUKKO 2. Dual-tone multi-frequency taajuudet (DialABC)

Taajuus	1209 Hz	1336 Hz	1477 Hz
697 Hz	1	2	3
770 Hz	4	5	6
852 Hz	7	8	9
941 Hz	*	0	#

Kuvan 5 DTMF-näppäintä painamalla saadaan laukaistua painallus tapahtuma. Tässä tapahtumassa katsotaan switch case valintarakenteella, että mikä nappi on kyseessä. Kuvasta 10 näkee, kuinka DTMF äänimerkki lähetetään kirjaston kautta vastaanottajalle.

ABTO:n kirjastossa näppäimen lähetys tapahtuu käyttämällä ASCII-numeroita, esimerkiksi numero 0 on ASCII-numerona 48 ja merkki * on ASCII-numerona 42 (Abto Software). Kuvan 10 mukaan AbtoPhone.SendToneEx() kutsu muodostuu siten, että ensimmäinen numero on ASCII-arvo näppäimestä, toinen numero on äänimerkin pituus millisekunteina ja loput numerot määrittävät miten äänimerkki lähetetään. Äänimerkin lähetykseen on kolme tapaa: in-band, out-of-band ja SIP info (Abto Software). Kuvan 10 esimerkissä äänimerkki lähetetään 200 millisekunnin pituisena ja lähetystapa on in-band.

```

private void DtmfClick(object sender, RoutedEventArgs e)
{
    Button tone = (Button)sender;
    try
    {
        switch (tone.Tag.ToString())
        {
            case "0":
                AbtoPhone.SendToneEx(48, 200, 1, 0, 0);
                break;
            case "1":
                AbtoPhone.SendToneEx(49, 200, 1, 0, 0);
                break;
            //
            //
            //
            case "#":
                AbtoPhone.SendToneEx(35, 200, 1, 0, 0);
                break;
            case "*":
                AbtoPhone.SendToneEx(42, 200, 1, 0, 0);
                break;
        }
    }
    catch (Exception ex)
    {
        Thread.Sleep(200);
    }
}

```

KUVA 10. DTMF-merkkiäänen lähetys

5.6 Ääniasetukset

ABTO:n kirjasto käyttää älylaitteen oletusarvoilla olevaa toistolaitetta ja mikrofonia. Käytettävän toistolaitteen ja mikrofonia voi halutessaan vaihtaa rajapinnan asetuksista. Sovelluksen ääniasetukset

alustetaan (kuva 11) sovelluksen käynnistysvaiheessa, mutta niitä voi kuitenkin vaihtaa milloin haluaa. Kirjasto tarjoaa äänen prosessointiin kaiun poiston, kohinan vaimennuksen ja automaattisen äänenvoimakkuuden säädön. Kuvan 11 koodissa 0 tarkoittaa, että asetukset on pois päältä ja 1 tarkoittaa, että se on päällä. Käytettävät äänikoodekit laitetaan tärkeysjärjestykseen kuvan 11 mukaan. Järjestys tarkoittaa sitä, että missä järjestyksessä kirjasto tarjoaa äänikoodekkeja IP PBX-palvelimelle, kun puhelua luodaan.

```
// Audiokoodekkien priority järjestys
softPhoneConfiguration.SetCodecOrder("G.711 mu-law|G.711 a-law|GSM|Opus|RFC4733 DTMF tones|", 0);

// Mahdollistaa äänenvoimakkuuden säädön
softPhoneConfiguration.IntenalVolumeImplEnabled = true;

softPhoneConfiguration.EchoCancelationEnabled = 1;
softPhoneConfiguration.NoiseReductionEnabled = 1;
softPhoneConfiguration.AutoGainControlEnabled = 0;
```

KUVA 11. Ääniasetukset

Kun ääniasetukset on alustettu, niin kuvan 5 mukaisella liikusäätimellä voi vaihtaa äänenvoimakkuutta. Toistolaitteen äänenvoimakkuuden säätö onnistuu kuvan 12 mukaisella koodilla. Mikrofonin äänenvoimakkuuden voi vaihtaa samalla tavalla kuin toistolaitteen.

```
AbtoPhone.PlaybackVolume = (int)sliderSpeaker.Value;
```

KUVA 12. Äänenvoimakkuuden säätö

Äänen prosessoinnin asetuksia voi vaihtaa kuvan 13 mukaisella koodilla missä vaiheessa tahansa, myös kesken puhelun. Kohinan vaimennuksen ja automaattisen äänenvoimakkuuden säätö tapahtuu samalla tavalla.

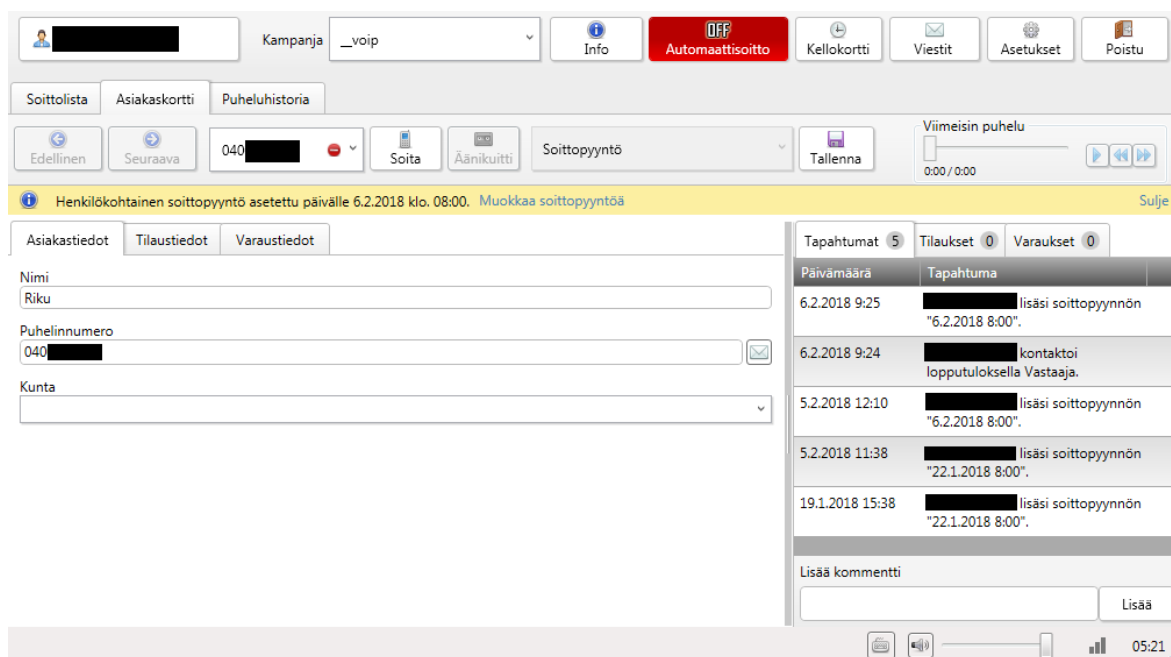
```
private void buttonEcho_Click(object sender, RoutedEventArgs e)
{
    CConfig audioCfg = AbtoPhone.Config;

    if(audioCfg.EchoCancelationEnabled == 1)
    {
        audioCfg.EchoCancelationEnabled = 0;
        buttonEcho.Content = "Echo Cancellation Off";
    }
    else
    {
        audioCfg.EchoCancelationEnabled = 1;
        buttonEcho.Content = "Echo Cancellation On";
    }
    AbtoPhone.ApplyConfig();
}
```

KUVA 13. Kaiun poisto

6 SOITTOJÄRJESTELMÄÄN INTEGROINTI

Kun demovaiheen tavoite, eli VoIP-puhelun soittaminen testisovelluksesta puhelimeen onnistui, niin alkoi opinnäytetyön kolmas vaihe. Sovelluksen integroinnilla tarkoitetaan demovaiheen sovelluksen toimintojen tekemistä soittojärjestelmään (kuva 14). Osa soittojärjestelmän luokista piti muokata, jotta VoIP-toiminnallisuudet voitaisiin lisätä. Sovellukseen integrointi oli jatkuvana osana projektia. Suuri osa tilaajan haluamista toiminnoista testattiin ensin testisovelluksella, jonka jälkeen ne siirrettiin soittojärjestelmään ja muokattiin lopulliseen muotoonsa. Tämän vaiheen tavoitteeksi oli asetettu yhden onnistuneen VoIP-puhelun soittaminen soittojärjestelmästä puhelimeen.



Kuva 14. Soittojärjestelmä

6.1 Toiminnot

Kaikki soittojärjestelmään tehty koodi ja tarkemmat kuvaukset toiminnoista on salassa pidettävää aineistoa, jota tässä opinnäytetyössä ei käydä läpi. Soittojärjestelmään tehdyt toiminnot ovat seuraavat:

- Softphonen alustus säikeessä kirjautumisen yhteydessä
- Äänilaitteiden hallinta, käyttäen apuna NAudio-kirjastoa
- Ääniasetukset
- Puhelun ja linjojen hallinta
- Lähtevät ja saapuvat VoIP-puhelut
- DTMF
- Virhetilanteiden käsittely
- Apufunktioita, kuten puhelinnumeron jäsentäminen SIP-osoitteeksi.

7 YHTEENVETO

Opinnäytetyön ensimmäisen vaiheen eli selvitystyön tavoitteena oli löytää tarkoitukseen sopiva kirjasto. Kirjaston valintaan vaikutti erilaiset syyt, kuten lisenssi ja ohjelmointikieli. Valinta osui loppuksen lopuksi Abto Softwaren tekemään ABTO SIP VoIP -kehitystyökaluun, joka osoittautui sopivaksi tähän projektiin.

Toisen vaiheen tekeminen aloitettiin heti, kun kirjaston valinta oli tehty. Demovaiheen tavoitteena oli saada yksi puhelu soitettua sovelluksesta puhelimeen. Sovelluksen käyttöliittymän tekeminen oli nopeaa, mutta kirjaston ja sen rajapinnan käyttäminen oli odotettua hankalampaa, koska kaikkia kirjaston ominaisuuksia ei ollut mainittu dokumentoinnissa. Ongelmia ilmestyi myös DTMF-äänissä. Numerojen 1-6 äänet toimivat täydellisesti, mutta 7, 8, 9, 0, merkit * ja # kuulostivat pätkiviltä. Lopulta demosovelluksesta saatiin soitettua puhelu, sovellukseen pystyi soittamaan puhelun ja osa DTMF-näppäimistä saatiin toimimaan.

Opinnäytetyön päätavoite oli saada tarvittavat VoIP-toiminnallisuudet soittojärjestelmä sovellukseen GSM-toiminnallisuuksien rinnalle. Lopputuloksena saatiin toimiva kokonaisuus erilaisia VoIP-toimintoja soittojärjestelmään. Soittojärjestelmän käyttäjä ei itse sovellusta käyttäessään huomaa, että soittaako hän GSM- vai VoIP-puheluita. Jatkokehityksessä soittojärjestelmään voi lisätä puhelun nauhoittamisen ja mahdollisesti myös konferenssipuhelun luomisen ja hallinnan.

Kaiken kaikkiaan opinnäytetyössä onnistuttiin ilman suurempia ongelmia. Työn asetettuihin tavoitteisiin päästiin. SIP-pohjaisen VoIP-sovelluksen ohjelmoiminen alusta lähtien olisi todella aikaa vievää ja hankalaa, jos ei olisi valmiita kirjastoja ja kehitystyökaluja saatavilla. ABTO:n kehitystyökalu ja kirjasto sopivat VoIP-sovelluksen tekoon vajaan mainiosti.

LÄHTEET JA TUOTETUT AINEISTOT

3CX. What is H.323. [Verkkoaineisto]. [Viitattu 2018-04-18] Saatavissa:

<https://www.3cx.com/pbx/h323/>

ABTO SOFTWARE. Developer Manual. [Verkkoaineisto]. [Viitattu 2018-05-13] Saatavissa:

http://www.voipsip SDK.com/Download/docs/ABTO_Software_VoIP_SIP_SDK_Developer_Manual.pdf

CORPUZ, John 2017-03-09. Best VoIP Apps for Your Desktop. [Verkkoaineisto]. [Viitattu 2018-05-

17] Saatavissa: <https://www.tomsguide.com/us/pictures-story/519-best-voip-apps.html#s1>

DIALABC. DTMF Tones. [Verkkoaineisto]. [Viitattu 2018-05-16] Saatavissa:

<http://www.dialabc.com/sound/dtmf.html>

DIGIUM. VoIP Basics for Business. [Verkkoaineisto]. [Viitattu 2018-05-15] Saatavissa:

<https://www.digium.com/solutions/ip-phone-systems/voip-basics>

KELLY, Timothy V. Understanding VoIP Hardphones. [Verkkoaineisto]. [Viitattu 2018-04-18]

Saatavissa: <http://www.dummies.com/education/internet-basics/understanding-voip-hard-phones/>

MARTIN, Matt. 2017-06-23. Guide to VoIP audio codecs. [Verkkoaineisto]. [Viitattu 2018-05-15]

Saatavissa: <https://www.nurango.ca/blog/voip-codecs-guide>

MOORE, Cortney. 2017-06-26. Audio Codecs: What They are and Why They Matter. Saatavissa:

[Verkkoaineisto]. [Viitattu 2018-05-15] <https://www.tomsguide.com/us/what-are-audio-codecs,review-4469.html>

NETTITIETO OY. NettiTieto. [Verkkoaineisto]. [Viitattu 2018-05-13] Saatavissa: [https://www.netti-](https://www.netti-tieto.fi/)

[tieto.fi/](https://www.netti-tieto.fi/)

NFON. Softphones. [Verkkoaineisto]. [Viitattu 2018-04-18] Saatavissa:

https://www.nfon.com/en_de/cloud-telephone-system/resources/glossary/softphones/

ON SIP. UDP versus TCP for VoIP. [Verkkoaineisto]. [Viitattu 2018-05-16] Saatavissa:

<https://www.onsip.com/blog/udp-versus-tcp-for-voip>

PORTER, Thomas. 2006. Practical VoIP Security. Rockland, MA : Syngress

PORTSIP. VoIP SDK Documentation. [Verkkoaineisto]. [Viitattu 2018-05-13] Saatavissa:

https://www.portsip.com/docs/html_man/win/html/index.html

ROUSE, Margaret. Real-Time Transport Protocol (RTP). [Verkkoaineisto]. [Viitattu 2018-05-16]
Saatavissa: <https://searchnetworking.techtarget.com/definition/Real-Time-Transport-Protocol>

ROUSE, Margaret. SRTP (Secure Real-Time Transport Protocol or Secure RTP). [Verkkoaineisto].
[Viitattu 2018-05-16] Saatavissa: <https://whatis.techtarget.com/definition/SRTP-Secure-Real-Time-Transport-Protocol-or-Secure-RTP>

ROUSE, Margaret. VoIP (voice over IP). [Verkkoaineisto]. [Viitattu 2018-05-15] Saatavissa:
<https://searchunifiedcommunications.techtarget.com/definition/VoIP>

TUTORIALSPPOINT. SIP-Codecs. [Verkkoaineisto]. [Viitattu 2018-05-15] Saatavissa:
https://www.tutorialspoint.com/session_initiation_protocol/session_initiation_protocol_codecs.htm