



УНИВЕРЗИТЕТ У НОВОМ САДУ  
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У  
НОВОМ САДУ

---



Шијаков Душан

**Интеграција Skype на VCM3556 платформи**

ДИПЛОМСКИ РАД  
- Основне академске студије -

Нови Сад, 2012



УНИВЕРЗИТЕТ У НОВОМ САДУ • ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА  
21000 НОВИ САД, Трг Доситеја Обрадовића 6

## КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА


Редни број, РБР:	
Идентификациони број, ИБР:	
Тип документације, ТД:	Монографска документација
Тип записа, ТЗ:	Текстуални штампани материјал
Врста рада, ВР:	Завршни рад (Основне академске студије)
Аутор, АУ:	Душан Шијаков
Ментор, МН:	Др Иштван Пап
Наслов рада, НР:	Интеграција Skype на ВСМ3556 платформи
Језик публикације, ЈП:	Српски / латиница
Језик извода, ЈИ:	Српски
Земља публиковања, ЗП:	Република Србија
Уже географско подручје, УГП:	Војводина
Година, ГО:	2011
Издавач, ИЗ:	Ауторски репринт
Место и адреса, МА:	Нови Сад; трг Доситеја Обрадовића 6
Физички опис рада, ФО: (поглавља/страна/ цитата/табела/слика/графика/прилога)	7/34/0/1/18/0/0
Научна област, НО:	Електротехника и рачунарство
Научна дисциплина, НД:	Рачунарска техника
Предметна одредница/Кључне речи, ПО:	Skype, SkypeKit, Телевизија, Cabot Observatory, UI Framework
УДК	
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад
Важна напомена, ВН:	
Извод, ИЗ:	У раду је приказан развој Skype клијент апликације која нуди основне Skype функције као што су пријављивање, слање и примање текстуалних порука, блокирање и брисање контаката, претрага корисника и слање захтева за пријатељство и потврда или игнорисање на пристигле захтеве за пријатељство.
Датум прихватања теме, ДП:	
Датум одбране, ДО:	
Чланови комисије, КО:	Председник: Др Небојша Пјевалица
	Члан: Др Јелена Ковачевић
	Члан, ментор: Др Иштван Пап
	Потпис ментора



UNIVERSITY OF NOVI SAD • FACULTY OF TECHNICAL SCIENCES  
21000 NOVI SAD, Trg Dositeja Obradovića 6

## KEY WORDS DOCUMENTATION

Accession number, ANO:	
Identification number, INO:	
Document type, DT:	Monographic publication
Type of record, TR:	Textual printed material
Contents code, CC:	Bachelor Thesis
Author, AU:	Dušan Šijakov
Mentor, MN:	Ištvan Pap, PhD
Title, TI:	Skype integration on BCM3556 platform
Language of text, LT:	Serbian
Language of abstract, LA:	Serbian
Country of publication, CP:	Republic of Serbia
Locality of publication, LP:	Vojvodina
Publication year, PY:	2011
Publisher, PB:	Author's reprint
Publication place, PP:	Novi Sad, Dositeja Obradovica sq. 6
Physical description, PD: <small>(chapters/pages/ref./tables/pictures/graphs/appendixes)</small>	7/34/0/1/18/0/0
Scientific field, SF:	Electrical Engineering
Scientific discipline, SD:	Computer Engineering, Engineering of Computer Based Systems
Subject/Key words, S/KW:	Skype, SkypeKit, Television, Cabot Observatory, UI Framework
UC	
Holding data, HD:	The Library of Faculty of Technical Sciences, Novi Sad, Serbia
Note, N:	
Abstract, AB:	The paper describes the development of the Skype client application that offers basic Skype functions such as logging in, sending and receiving text messages, blocking and deletion of contacts, search users and send friend requests and confirmations or ignore the incoming requests for friendship.
Accepted by the Scientific Board on, ASB:	
Defended on, DE:	
Defended Board, DB:	President: Nebojša Pjevalica, PhD
	Member: Jelena Kovačević, PhD
	Member, Mentor: Ištvan Pap, PhD
	Mentor's sign

	УНИВЕРЗИТЕТ У НОВОМ САДУ • ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА 21000 НОВИ САД, Трг Доситеја Обрадовића 6	Број:
	<b>ЗАДАТАК ЗА ИЗРАДУ ЗАВРШНОГ (BACHELOR) РАДА</b>	

*(Податке уноси предметни наставник - ментор)*

Врста студија:	а) Основне академске студије б) Основне струковне студије
Студијски програм:	Рачунарство и аутоматика
Руководилац студијског програма:	Др Иван Луковић

Студент:	Душан Шијаков	Број индекса:	e11891
Област:	Рачунарска техника и рачунарске комуникације		
Ментор:	Др Иштван Пап		

НА ОСНОВУ ПОДНЕТЕ ПРИЈАВЕ, ПРИЛОЖЕНЕ ДОКУМЕНТАЦИЈЕ И ОДРЕДБИ СТАТУТА ФАКУЛТЕТА ИЗДАЈЕ СЕ ЗАДАТАК ЗА ЗАВРШНИ (Bachelor) РАД, СА СЛЕДЕЋИМ ЕЛЕМЕНТИМА:

- проблем – тема рада;
- начин решавања проблема и начин практичне провере резултата рада, ако је таква провера неопходна;
- литература

### НАСЛОВ ЗАВРШНОГ (BACHELOR) РАДА:

Интеграција Skype на ВСМ3556 платформи

### ТЕКСТ ЗАДАТКА:

У раду је приказан развој Skype клијент апликације која нуди основне Skype функције као што су пријављивање, слање и примање текстуалних порука, блокирање и брисање контаката, претрага корисника и слање захтева за пријатељство и потврда или игнорисање на пристигле захтеве за пријатељство.

Руководилац студијског програма:	Ментор рада:

Примерак за: О - Студента; О - Ментора

## SADRŽAJ

1. Uvod.....	1
2. Teorijske osnove.....	3
2.1 Cabot Observatory.....	3
2.2 SkypeKit.....	7
3. Koncept rešenja.....	9
3.1 Prevođenje biblioteka.....	9
3.2 Integracija SkypeKit biblioteka.....	10
3.3 Izrada dijaloga.....	10
4. Opis realizacije.....	19
4.1 Moduli u kojima su realizovani dijalozi.....	21
4.2 Moduli u kojima su realizovane Skype funkcije.....	22
5. Testiranje.....	23
6. Zaključak.....	33
7. Literatura.....	34

## SPISAK SLIKA

Slika 2.1 Izgled Cabot Observatory simulatora.....	3
Slika 2.2 DTV arhitektura. ....	5
Slika 2.3 Dijagram reakcije na asinhroni događaj.....	6
Slika 2.4 Primer mape poruka. ....	6
Slika 2.5 Objekti raspoređeni u strukturu oblika stabla.....	7
Slika 2.6 Sprega između aplikacije i SkypeKit-a.....	7
Slika 3.1 Put poruke od SkypeKit-a do aplikacije.....	11
Slika 3.2 Izgled <i>Login</i> dijaloga.....	12
Slika 3.3 Izgled <i>LogingIn</i> dijaloga.....	13
Slika 3.4 Neke od grešaka koje se mogu javiti pri prijavljivanju.....	13
Slika 3.5 Izgled glavnog Skype dijaloga.....	14
Slika 3.6 Izgled glavnog Skype dijaloga.....	15
Slika 3.7 Izgled <i>ConfirmContacts</i> dijaloga.....	16
Slika 3.8 Izgle <i>FindContact</i> dijaloga.....	17
Slika 3.9 Izgled <i>FindingContact</i> dijaloga.....	17
Slika 3.10 Izgled <i>AddContact</i> dijaloga.....	18
Slika 3.11 Izgled <i>LogingOut</i> dijaloga.....	18
Slika 4.1 Raspored modula i međusobna komunikacija.....	20

## **SPISAK TABELA**

Tabela 5.1 Tabela testnih slučajeva .....	32
---	----

**SKRAĆENICE**

TV	- <i>Televizija</i>
HDTV	- <i>High-definition television</i>
DTV	- <i>Digitalna televizija</i>
SoC	- <i>System-on-a-Chip</i>
DVB	- <i>Digital Video Broadcasting</i>
SVG	- <i>Scalable Vector Graphics</i>
XML	- <i>Extensible Markup Language</i>
PNG	- <i>Portable Network Graphics</i>
GIF	- <i>Graphics Interchange Format</i>
JPEG	- <i>Joint Photographic Experts Group Standard</i>
API	- <i>Application Programming interface</i>
TS	- <i>Transport Stream</i>
OS	- <i>Operativni sistem</i>



## 1. Uvod

Od pojave televizije u boji pa sve do sredine devedesetih godina prošlog veka nije bilo značajnijeg napretka u konceptu distribucije i prikaza TV (televizijske) usluge. Tek u poslednjoj deceniji počinje ponovni nagli razvoj televizije. Mogu se uočiti četiri glavna pravca razvoja televizije:

- Povećanje dimenzija i prelazak na tehnologije ravnih ekrana
- Povećanje rezolucije slike (HDTV)
- Primena multimedijalnih i internet tehnologija
- Pojava digitalne televizije

Pojava i uspostavljanje standarda koji se koriste u digitalnoj televiziji vezuje se za poslednju dekadu prošlog veka. Prva demonstracija digitalnog TV prenosa održana je 1995 godine. Od tada televizija se sve više i više razvija, digitalni prenos obezbeđuje bolji kvalitet slike/zvuka, koji više ne mogu biti u prenosu ometani interferencijom sa drugim signalima, bez obzira na rastojanje na koje se slika i zvuk prenose. Slika i zvuk koju digitalni signal nosi su isti kao i na izvoru emitovanja. Digitalna televizija omogućava nove aplikacije kao što je interaktivna televizija koja omogućava izvršavanje računarskih aplikacija razvijenih u raznim programskim jezicima. DTV (digitalna televizija) omogućava uvođenje novih usluga, izbor jezika za titlovanje i audio kanala, interaktivni i multimedijalni sadržaj, pristup internetu.

Aplikacije razvijene za ovakve televizijske prijemnike, čija su osnova pravi mali digitalni računari, mogu biti različitog stepena složenosti. Neke od njih su:

- Electronic Program Guide (EPG)
- Personal Video Recorder (PVR)
- Interaktivna Televizija (iTV)
- Igre

---

- Internet via TV

U ranim danima DTV, ljudi su veoma često pogrešno tumačili pojam interaktivne televizije i zamenjivali ga sa pristupom internetu putem televizije. TV kompanije su predviđale potiskivanje PC računara u korist TV uređaja za potrebe internet pretraživanja. Ipak, realnost nije potvrdila njihova predviđanja. Sa povećanjem rezolucije TV displeja, koncept Interneta preko televizije postaje ponovo aktuelan, pišu se razne aplikacije za pristup globalnim mrežama i aplikacije koje omogućavaju laku komunikaciju sa drugim ljudima širom sveta.

Cilj ovog zadatka je proširenje standardne funkcionalnosti već postojeće TV aplikacije razvijene u *Cabot Observatory* okruženju za kreiranje TV aplikacija korišćenjem *Microsoft Visual Studio 2008* razvojnog okruženja a namenjena je televizorima i set-top box-ovima sa ugrađenim *Broadcom BCM3556 SoC (System-on-a-Chip)* i *Linux* operativnim sistemom.

Postojeću TV aplikaciju potrebno je proširiti *Skype* klijentom integracijom *SkypeKit* skupa alata. *Skype* klijent treba da omogući funkcije prijavljivanja, slanja i primanja poruka. Takođe treba implementirati funkcije za rad sa kontaktima kao što su pretraga i dodavanje kontakata, potvrđivanje zahteva za dodavanje u listu prijatelja kao i funkcija za blokiranje i odblokiranje određenih kontakata.

*Skype* je najzastupljeniji program za telefoniranje i dopisivanje preko interneta sa bilo kim u svetu. Pozivi i slanje poruka između računara koji imaju *Skype* i pristup internetu su besplatni, dok se pozivi i sms poruke ka telefonskim mrežama naplaćuju ali je cena ovakvih poziva znatno niža. *Skype* takođe omogućava video pozive, slanje fotografija i drugih multimedijalnih sadržaja, grupe pozive i dopisivanja, i iz dana u dan se sve više razvija.

*BCM3556* je sledeća generacija *Broadcom SoC* proizvoda namenjenih digitalnoj televiziji za evropsko tržište. Karakteriše ga veliki stepen integracije i veoma dobar kvalitet slike. Ovo integrisano kolo kombinuje DVB-T COFDM zemaljski i PAL/SECAM demodulator, dva DVI/HDMI prijemnika, transportni procesor, digitalni audio procesor, 3D/2D grafički procesor, Ethernet adapter, digitalni procesor za analodni audio i video, analodni video digitalizator i DAC funkcije, stereo audio DAC-ove, MIPS peocesor na 400MHz i perifernu kontrolnu jedinicu koja pruža različite funkcije televizijske kontrole. Procesor je napravljen u 65nm tehnologiji i na njemu se istovremeno mogu izvršavati dve niti. *BCM3556* poseduje video decoder sposoban da podrži AVC, VC-1, i DVB-T MPEG-2 tokove podataka visoke rezolucije [1].

## 2. Teorijske osnove

### 2.1 Cabot Observatory

*Cabot Observatory* je napredno okruženje za *Windows* platformu koje omogućava proizvođačima da razviju, testiraju i simuliraju aplikacije za DVB (*Digital Video Broadcasting*) programsku podršku. Kompletna implementacija *Cabot Observatory* okruženja za *Windows* je u okviru razvojnog okruženja *Microsoft Visual Studio 2008*. Takođe postoji i okruženje za *Linux* koje je implementirano u okviru razvojnog okruženja *Eclipse*. Testiranje se može vršiti nad stvarnim video tokovima sa lokalnog diska sa simulacijom daljinskog upravljača. Okruženje kroz svoju arhitekturu i biblioteku za SVG (*Scalable Vector Graphics*) datoteke predstavlja moćan i fleksibilan alat za razvijanje DVB aplikacija uz mogućnost lake izmene izgleda izmenom SVG datoteka.



Slika 2.1 Izgled Cabot Observatory simulatora.

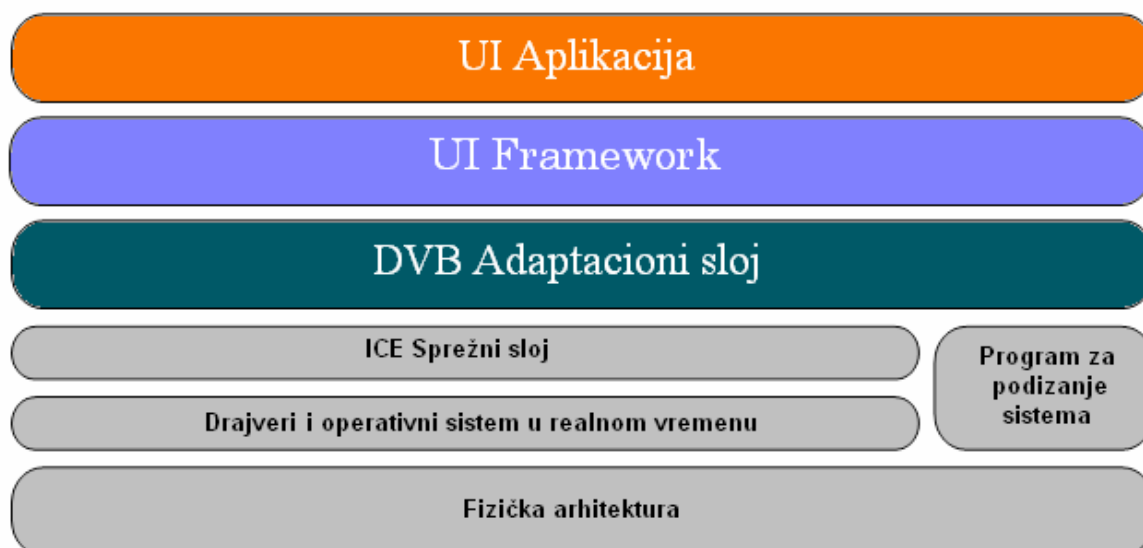
SVG je familija specifikacija formata datoteka baziranih na XML (*Extensible Markup Language*) konceptima za opis dvodimenzionalne vektorske grafike, kako statičke tako i dinamičke (interaktivne i animirane). Pošto su SVG slike opisane XML kodom to znači da se one mogu pretraživati, indeksirati, skriptovati, itd. Takođe moguće ih je kreirati i editovati bilo kojim tekstualnim editorom [2]. Postoje i programi sa razvijenom grafičkom podrškom za SVG datoteke poput *Adobe Illustrator*-a čime je proces stvaranja i editovanja datoteka mnogo olakšan. Ovo sve omogućava da se izgled aplikacije menja samo izmenom njenih SVG datoteka nekim editorim poput *Adobe Illustrator* i prevede već postojeći kod.

Osim podrške za SVG datoteke *Cabot Observatory* pruža i podršku za iscrtavanje PNG (*Portable Network Graphics*), datoteka. PNG je prvobitno namenjen prenosu slika preko interneta, koder koristi algoritam za kodovanje bez gubitaka i podržava RGB format slike sa 24 i 32-obitnom paletom boja [3]. Pored ova dva tipa datoteka koje se najviše koriste za iscrtavanje grafičkog interfejsa postoji još i podrška za GIF (*Graphics Interchange Format*) i JPEG (*Joint Photographic Experts Group Standard*) datoteka.

U *Cabot Observatory* okruženju unutar *UI Framework*-a su već napravljene neke osnovne komponente za interakciju sa korisnikom. To su:

- SVGButton
- SVGSelectionBox
- SVGTextEntryBox
- SVGTextBox
- SVGProgressBar
- SVGScrollBar
- SVGBasicListBox
- SVGBasicListBoxRow
- SVGOverlay
- SVGImage
- SVGMenu
- SVGDialog

Na slici 2.2 prikazana je arhitektura aurora DTV programske podrške.

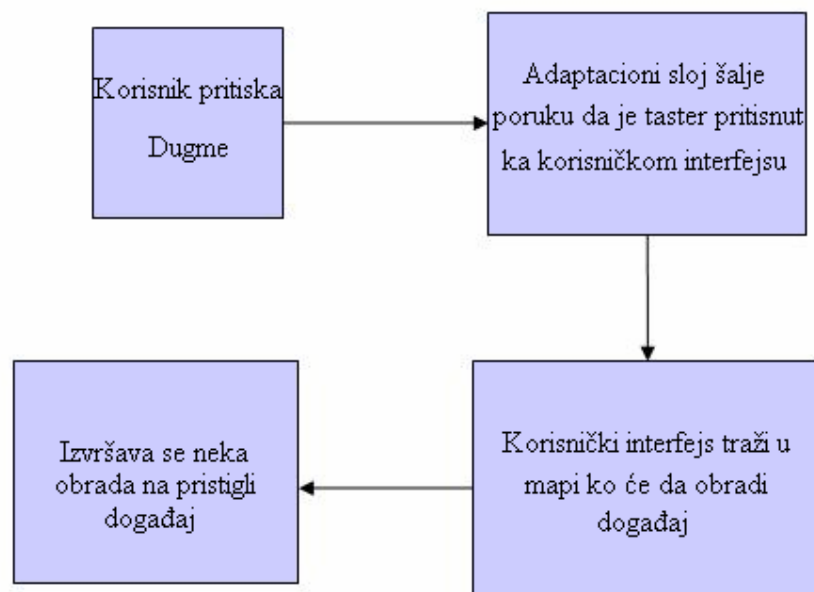


Slika 2.2 DTV arhitektura.

Operativni sistem je prvi programski sloj iznad fizičke arhitekture. Osnovna uloga operativnog sistema u digitalnom TV prijemniku je da obezbedi okruženje za podršku višeprocenih zahteva DTV softvera. Ograničeni resursi digitalnih TV prijemnika zahtevaju da memorija koju koristi operativni sistem bude što manja.

Iznad operativnog sistema nalazi se takozvani adaptacioni sloj, koji obezbeđuje uloge višeg nivoa koji su sastavni deo DTV funkcionalnosti i omogućava izvršavanje naprednih DTV aplikacija. On obezbeđuje odgovarajuću programsku spregu koja apstrahuje funkcionalnost DTV uređaja, fizičke arhitekture, kao i funkcije operativnog sistema čime je omogućeno da proizvođači aplikativnog DTV softvera ne moraju da poznaju specifičnosti fizičke arhitekture digitalnog televizijskog prijemnika. Ovim je omogućeno da se aplikacije koje koriste isti adaptacioni sloj mogu izvršavati na različitim fizičkim arhitekturama, to jest adaptacioni sloj formira virtualnu mašinu.

*UI Framework* predstavlja vezu između krajnje aplikacije i adaptacionog sloja. Svaka aplikacija mora najmanje da ima *MyApp.cpp* izveden iz *UIApp*. i *CustomUIFactory.cpp* izveden iz *UIFactory*. Pokretanje aplikacije obavlja se pozivanjem njene *init()* funkcije koja treba da uradi svu inicijalizaciju i podešavanja aplikacije. Nakon ovoga aplikacija je upravljana asinhronim događajima. Asinhroni događaji stižu u *UI Framework* a zatim se prosleđuju aplikaciji ili nekom prozoru unutar aplikacije. Primer jednog takvog događaja je pritisak nekog dugmeta daljinskog upravljača. Kada korisnik stisne neko dugme adaptacioni sloj o tome obaveštava *UI Framework*. *UI Framework* traži u mapi poruka objekat koji trenutno očekuje poruku. Objekat započinje proces obrade poruke i na osnovu rezultata obrade izvršava određenu akciju



Slika 2.3 Dijagram reakcije na asinhroni događaj.

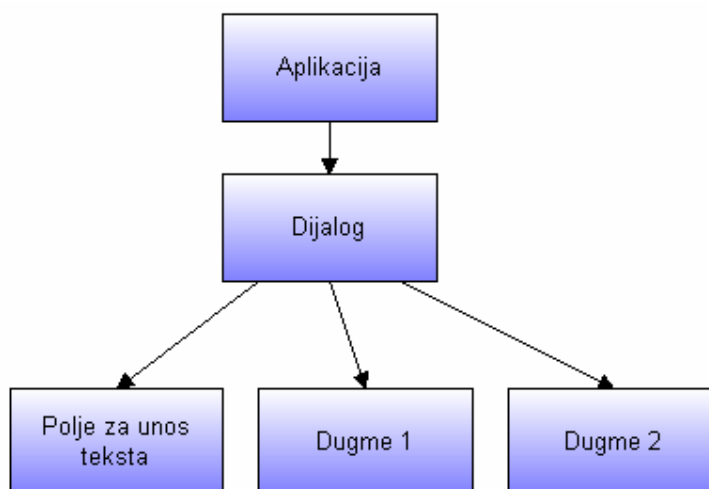
Ove *UI* poruke se u *UI Framework*-u smestaju u odgovarajući red i prosleđuju po redu po kom su i pristigle. Takođe moguće je poruku poslati na prvo mesto u redu. Sa ovim ipak treba biti oprezan jer se samo jedna poruka može smestiti na početak reda i ukoliko prethodna poruka nije obrađena može doći do njenog brisanja. Objekat koji može da obradi poruku je bilo koji objekat koji nasleđuje klasu *UIMessageTarget*. Svako određište *UI* poruka ima svoju mapu poruka. Primer jedne takve mape dat je na slici.

```

MESSAGE_MAP_BEGIN(Banner, UIDialog)
  ON_MESSAGE(key_press, keyMenu, &Banner::onMenu)
  ON_MESSAGE(key_press, keyProgramUp, &Banner::onProgramChange)
  ON_MESSAGE(key_press, keyProgramDown, &Banner::onProgramChange)
  ON_MESSAGE(key_press, ID_ANY_ID, &Banner::onOtherKey)
  ON_MESSAGE(timer, TIME_UPDATE_TIMER_EVENT_ID, &Banner::onUpdateTime)
  ON_MESSAGE(timer, DISAPPEAR_TIMER_EVENT_ID, &Banner::onDisappearTimer)
MESSAGE_MAP_END
  
```

Slika 2.4 Primer mape poruka.

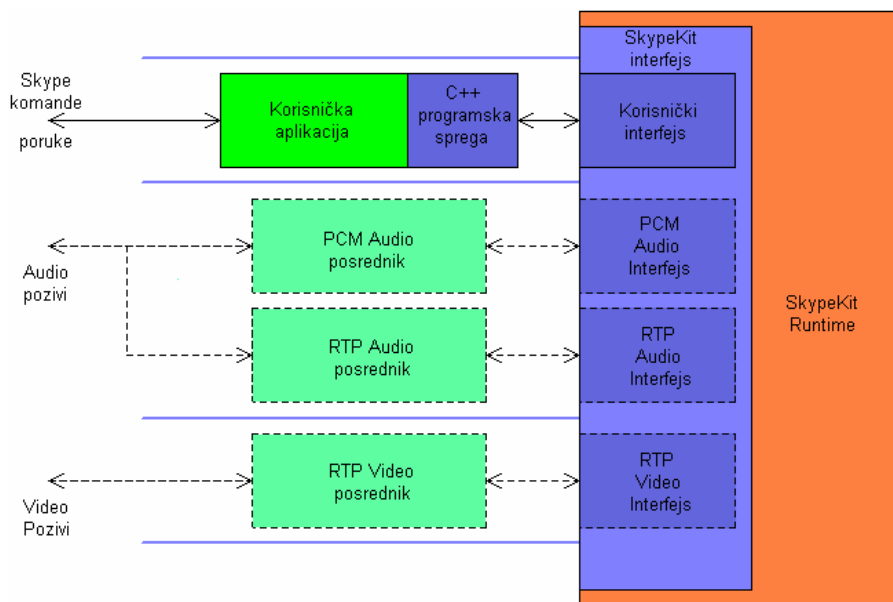
Sa slike se vidi da svaka poruka ima svoj tip i svoj identifikacioni broj i pri njenom pristizanju se poziva odgovarajuća funkcija koja treba da je obradi. Svi objekti su povezani jedan za drugim u strukturu oblika stabla. Za svaku poruku se proverava da li može da je obradi najniži objekat stabla pre nego što se prosledi višem nivou [4, 5].



Slika 2.5 Objekti raspoređeni u strukturu oblika stabla.

## 2.2 SkypeKit

*SkypeKit* je kolekcija alata i sprega koje omogućavaju razvijateljima programske podrške za potrošačku elektroniku i aplikacije koje imaju pristup internet da ponude funkcije *Skype* tekstualnih poruka, audio i video poziva, transfera podataka, itd. *SkypeKit* je dizajniran da radi sa velikim brojem čipova, operativnih sistema i audio i video uređaja. *SkypeKit* radi u pozadini bez *Skype* grafičke korisničke sprege. Korisnička *Skype* aplikacija komunicira sa ostalim *Skype* korisnicima preko *SkypeKit Runtime* aplikacije. Pored toga moguće je i povezivanje audio i video tokova kroz opcione komponente. Postoji više verzija *SkypeKit Runtime* aplikacije, to jest verzija sa i bez podrške za video pozive.



Slika 2.6 Sprega između aplikacije i SkypeKit-a

Pre pokretanja korisničke aplikacije treba da se pokrene *SkypeKit Runtime* aplikacija preko koje se komunicira sa ostalim Skype klijentima. Komunikacija sa *SkypeKit Runtime* aplikacijom ostvarena je preko *SkypeKit* biblioteka. Za sad postoji podrška za 3 programska jezika [6]:

- *C++*
- *Java*
- *Python*

Pošto je postojeća DVB aplikacija razvijena u C++ programskom jeziku koristiće se i *SkypeKit* C++ programska podrška.



## 3. Koncept rešenja

Proces razvoja aplikacije može se podeliti u 3 koraka:

- Prevođenje biblioteka – potrebno je dobiti biblioteke u kojima su implementirane funkcije koje *SkypeKit* nudi.
- Integracija *SkypeKit* biblioteka – ove biblioteke treba integrisati sa već postojećom DTV aplikacijom.
- Izrada dijaloga – treba izraditi odgovarajuće dijaloge koji će predstavljati *Skype* klijent aplikaciju.

### 3.1 Prevođenje biblioteka

Da bi uopšte mogle da se dodaju neke od funkcija koje nudi *SkypeKit* prvo moraju da se prevedu 2 projekta koji se dobijaju uz njega. To su:

- *skypekit-cppwrapper\_2\_lib*
- *skypekit-cyassl\_lib*

U *Windows*-u ovi projekti se prevode pomoću *Microsoft Visual Studio 2008* razvojnog okruženja a njihovim prevođenjem dobijaju se dve istoimene biblioteke sa *.lib* ekstenzijom. U ovim bibliotekama su realizovane funkcije koje nudi *SkypeKit* programska podrška i koriste se u *Cabot Observatory* simulatoru.

Da bi se kasnije ova aplikacija mogla izvršiti i na ciljnoj platformi potrebno je ove projekte prevesti i pod *Linux* operativnim sistemom kako bi se dobile odgovarajuće biblioteke za *Linux* koji se izvršava na platformi. Ovo prevođenje se obavlja pomoću *GNU Make* alata. Unutar *Makefile* datoteke potrebno je navesti da se u procesu prevođenja koristi *gcc* i *g++* prevodioc za

*MIPSel* procesor koji poseduje *BCM3556* integrisano kolo. Nakon uspešnog prevođenja treba da se dobiju biblioteke sa *.a* ekstenzijom.

## 3.2 Integracija SkypeKit biblioteka

Nakon prevođenja SkypeKit biblioteka sledi njihova integracija u adaptacioni sloj aplikacije i njihovo zajedničko prevođenje. Pri ovome se javljaju neki problemi kao što su ista imena nekih enumeracija i makroa unutar *SkypeKit API*-ja i *Cabot Observatory* aplikacije. Isto tako mogu da se jave problemi prilikom uvezivanja zbog različitih podešavanja *C/C++* prevodioca pri prevođenju prethodne dve biblioteke i prevodioca sa kojim je prevedena *Cabot Observatory* aplikacija. Prvi problem se rešava izmenom određenih imena unutar aplikacije a drugi problem menjanjem parametara *Runtime Library* sa *Multi-threaded Debug DLL (/MDd)* na *Multi-threaded Debug (/MTd)* u prevođenju prethodne dve biblioteke.

Prevođenje aplikacije za ciljnu platformu vrši se pokretanjem odgovarajuće *shell* skripte u kojoj se nalaze sve potrebne informacije i komande za uspesno prevođenje aplikacije. Ovaj proces prevođenja započinje se sledećom naredbom:

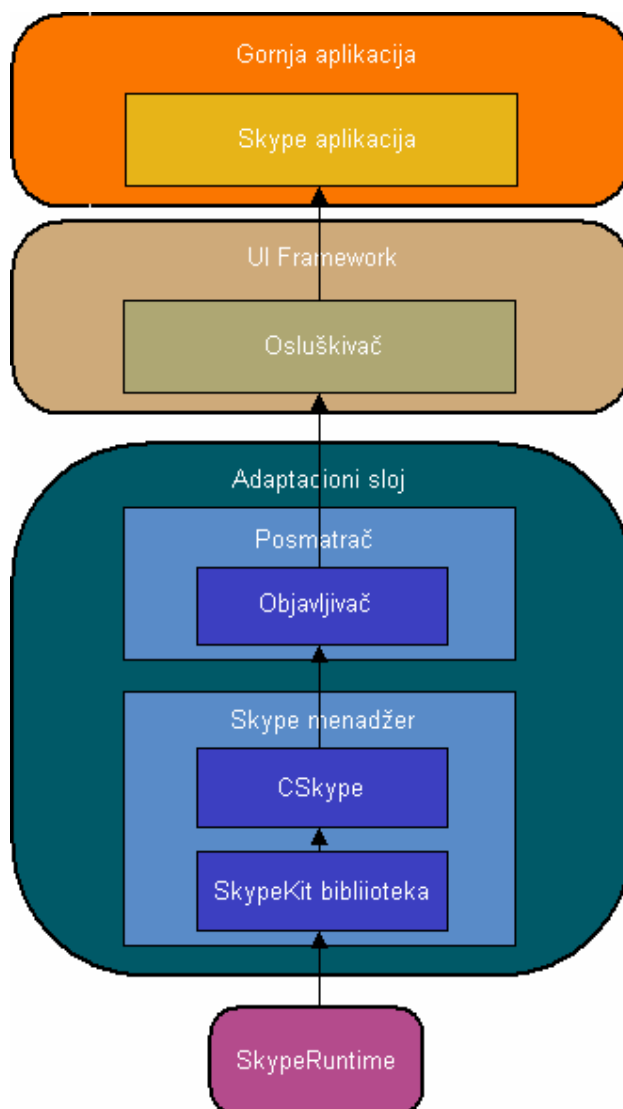
```
./build.sh --options=rtrk+mb71 elf
```

Pri zajedničkom prevođenju biblioteka i aplikacije za ciljnu platformu javio bi se samo prvi problem ali je on rešen još u procesu simulacije tako da će prevođenje proći bez problema.

## 3.3 Izrada dijaloga

Tek kada je završeno uspešno prevođenje *SkypeKit*-a sa aplikacijom mogu da se implementiraju klase i dijalozi koji će se koristiti unutar *Skype* klijent aplikacije. Odmah nakon pokretanja aplikacije pokreće se *SkypeKit RunTime* aplikacija. Kao što je već spomenuto objekat koji poseduje sve funkcije za komunikaciju sa *SkypeKit RunTime* aplikacijom smešta se u adaptacioni sloj odakle će sve te funkcije preko *EASI* modula biti dostupne unutar cele aplikacije. Takođe treba napomenuti da objekat koji se nalazi unutar adaptacionog sloja ne zna za *Skype* klijent aplikaciju već aplikaciji sve informacije stižu putem *UI* poruka. Kada se desi neki događaj poziva se neka od povratnih funkcija *SkypeKit* programske sprege. Ove funkcije su virtualne što znači da se mogu redefinisati i može da im se doda mehanizam za slanje poruka ka *UI Framework*-u. Kada se desi neki događaj obaveštavaju se svi posmatrači koji su se registrovali na njega određenom porukom. Oni ovu poruku prosleđuju svojim objavljivačima. U

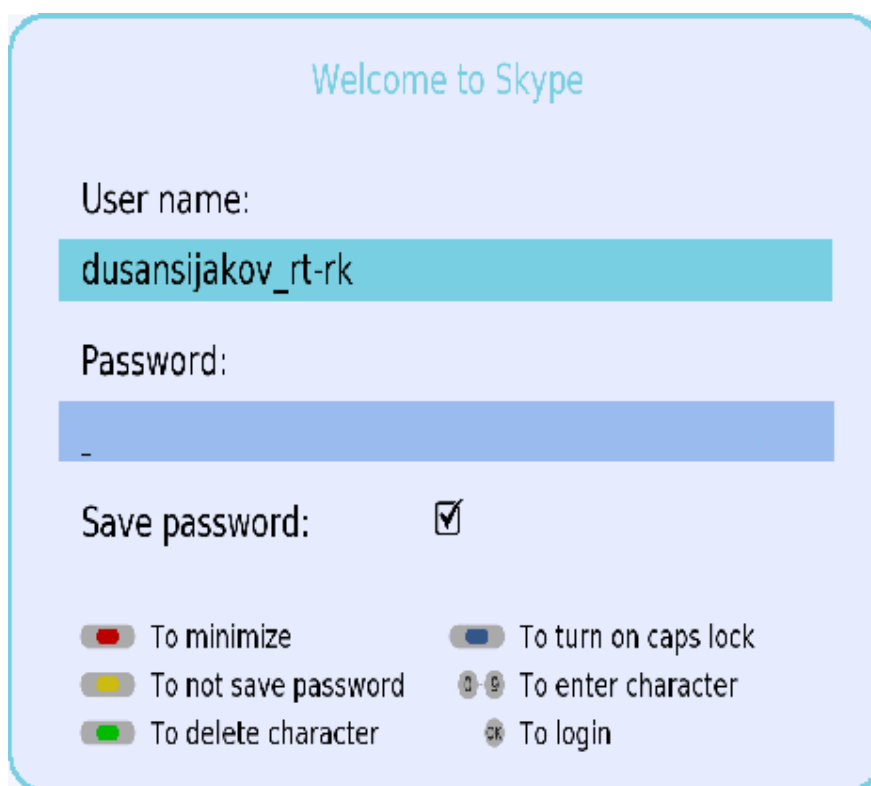
ovom rešenju ima samo jedan posmatrač i objavljiivač jer se po prijemu neke poruke o njoj samo obaveštava krajnja aplikacija, dok bi kod implementacije audio i video poziva morao da bude bar još jedan posmatrač i objavljiivač koji bi izvršili prebacivanje zvuka sa trenutnog kanala na dolazeće *Skype* pozive. Sad taj objavljiivač obaveštava sve svoje osluškivače da je dobio poruku. Jedan od osluškivača je i sam *UI Framework* koji šalje *UI* poruku. Ovu poruku će dobiti ili neki od dijaloga ili *DemoApp* ukoliko nema nijedan dijalog koji želi da primi datu poruku. Ovde još treba napomenuti da se poruka koja se prosleđuje do osluškivača razlikuje od poruke koju on šalje. Na slici 3.1 je prikazan put poruke koju šalje *SkypeKit RunTime* aplikacija.



Slika 3.1 Put poruke od SkypeKit-a do aplikacije

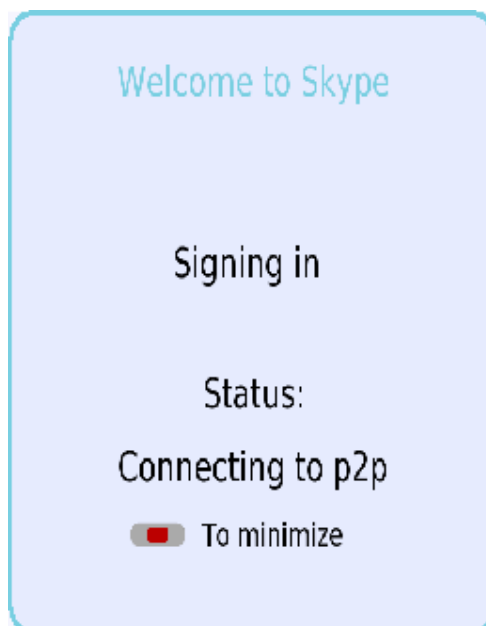
Posle pokretanja aplikacije i pritiska na plavo dugme na daljinskom upravljaču pokreće se prvi *Login* dijalog. U njemu se nalaze 2 polja za unos teksta. Na dugmiće strelica gore i dole

odabira se jedno od ova dva polja a pritiskom na numeričke dugmiće unosi se korisničko ime i lozinka slično kucanju poruke na mobilnom telefonu. Plavo dugme ima ulogu *caps lock*-a čijim pritiskom se uključuju i isključuju velika slova dok zeleno dugme ima ulogu *back space* dugmeta i njegovim pritiskom se briše prethodno uneti karakter. Proces prijavljivanja započinje se pritiskom na *OK* čime se pokreće *LogingIn* dijalog. Pritiskom žutog dugmeta se odabira da li da *Skype* pamti korisničko ime i lozinku. Ukoliko su korisničko ime i lozinka zapamćeni, automatski se pri sledećem pokretanju TV prijemnika i pokretanju *Skype* aplikacije započinje proces prijavljivanja čime se preskače ovaj dijalog. Još jedna od mogućih opcija u ovom dijalogu je pritisak crvenog dugmeta daljinskog upravljača čime se gasi dijalog (efekat minimiziranja). Ponovno pokretanje dijaloga (efekat maksimiziranja) postiže se ponovnim pritiskom na plavo dugme upravljača.

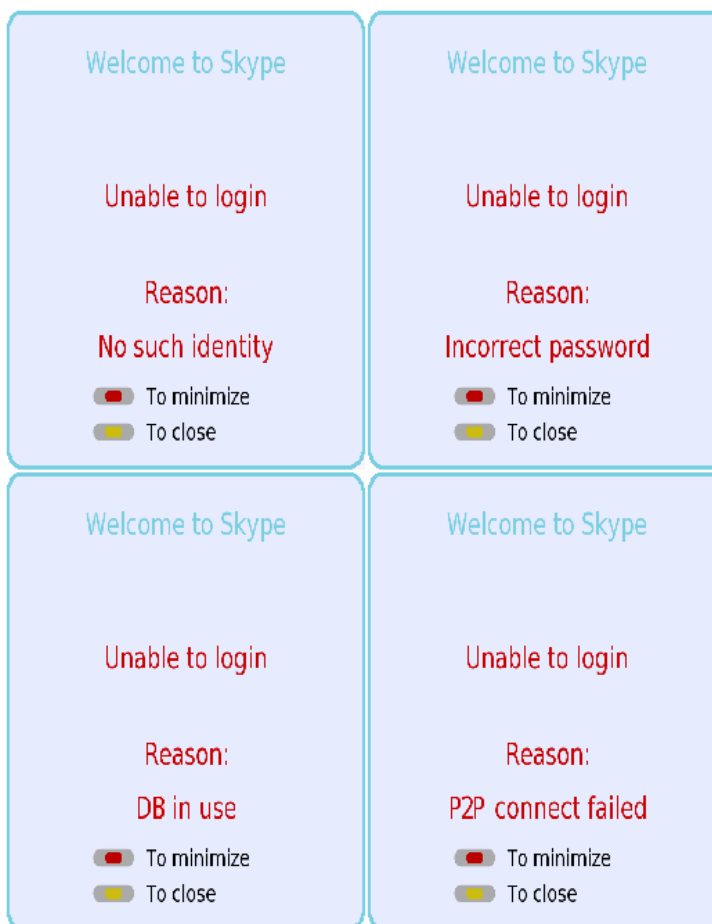


Slika 3.2 Izgled *Login* dijaloga

Nakon pritiska *OK* dugmeta iz *Login* dijaloga ili odmah pri pritisku plavog dugmeta (ukoliko je prethodno bilo zapamćeno korisničko ime i lozinka) pokreće se *LogingIn* dijalog. U ovom dijalogu se ispisuju sve informacije o prijavljivanju. U procesu prijavljivanja jedina opcija koju korisnik ima je pritisak na crveno dugme čime se postiže efekat minimiziranja dijaloga isto kao u prethodnom dijalogu.

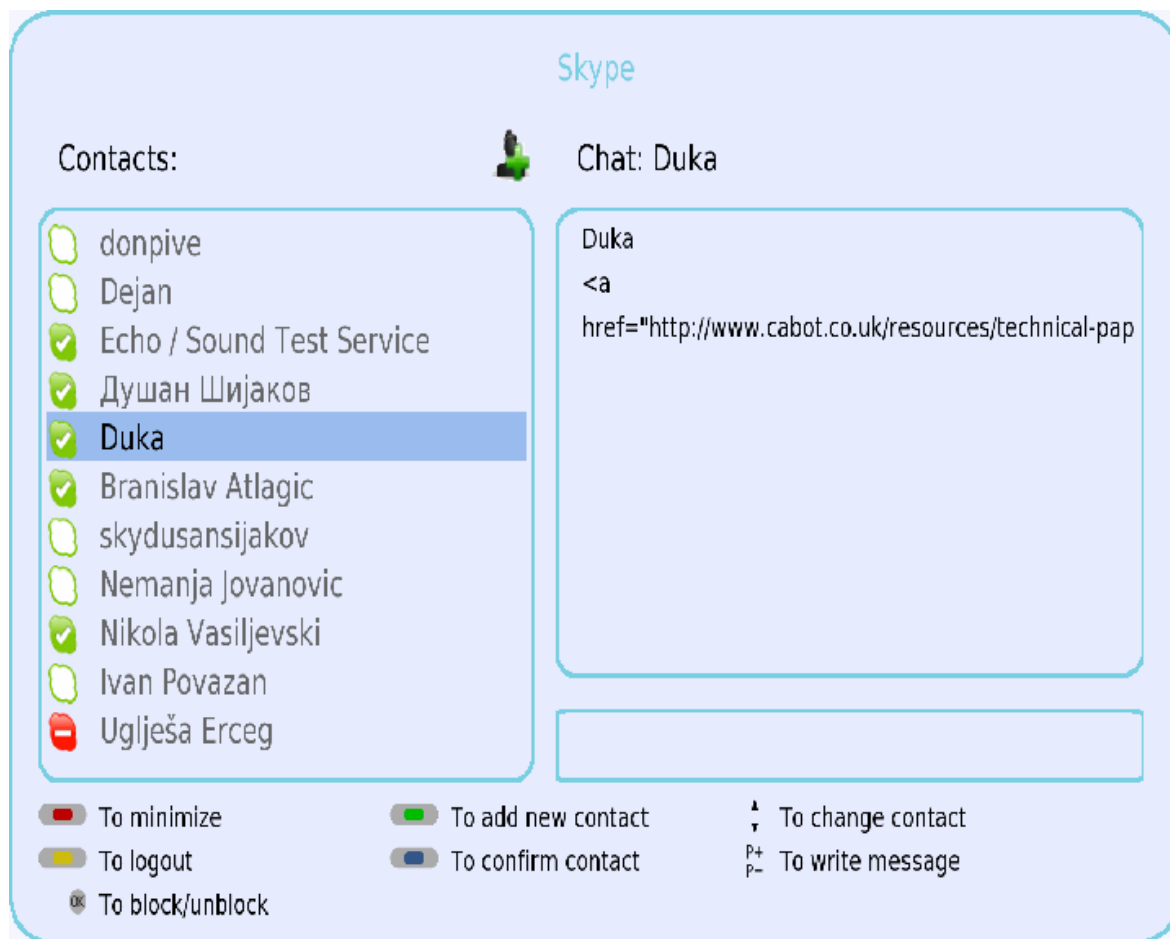
Slika 3.3 Izgled *LogingIn* dijaloga

Ukoliko je u procesu prijavljivanja došlo do neke greške ispisuje se odgovarajuća greška i vrši se povratak na povratak na početni *Login* dijalog (automatski ili pritiskom na žuto dugme).



Slika 3.4 Neke od grešaka koje se mogu javiti pri prijavljivanju

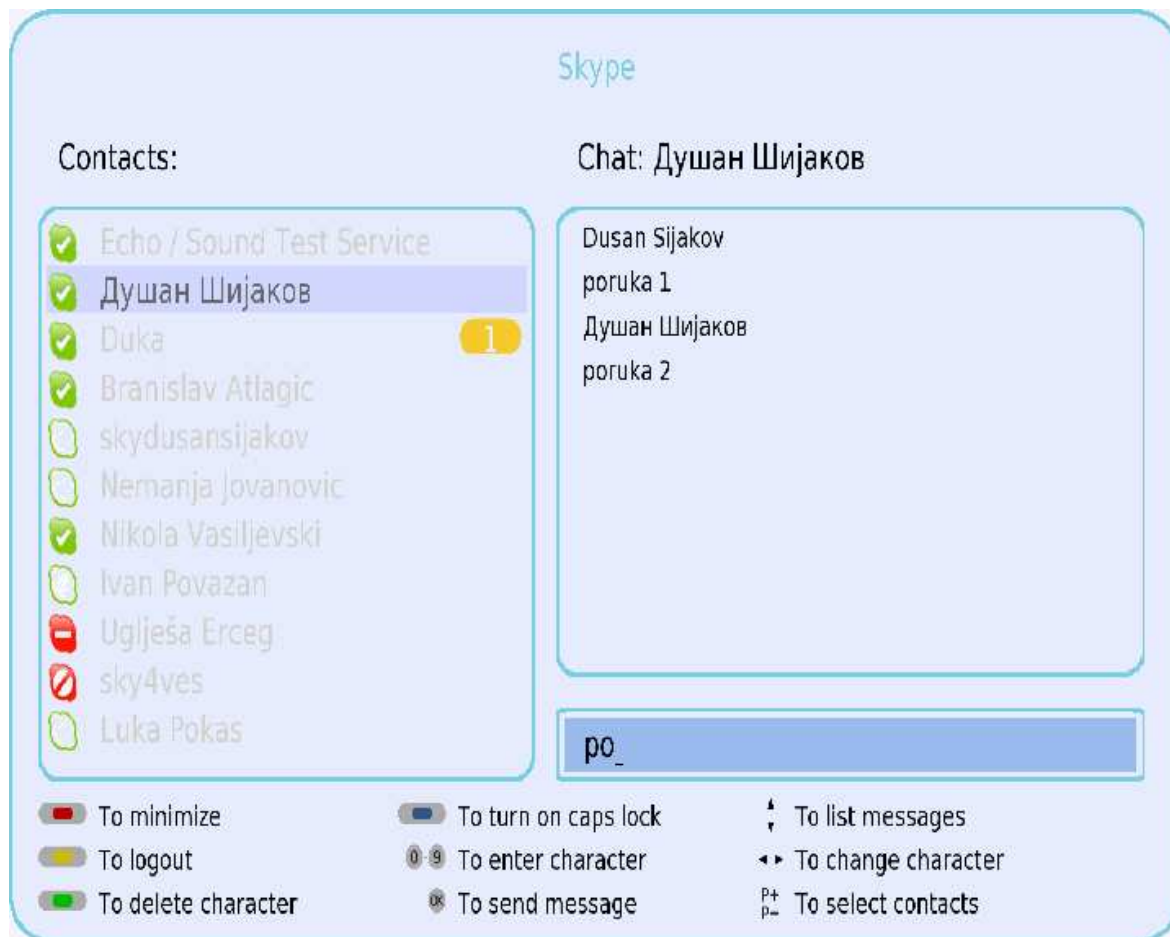
Posle zatvaranja *LoginIn* dijaloga u slučaju uspešnog prijavljivanja pokreće se glavni Skype dijalog. On sadrži listu kontakata, prozor u kojem su ispisane poruke za trenutno izabrani kontakt kao i polje u koje se unosi poruka za slanje.



Slika 3.5 Izgled glavnog Skype dijaloga

Kada je odabrana lista sa kontaktima pritiscima na dugmiće strelica gore i dole vrši se promena selektovanog kontakta čime se istovremeno osvežava i lista sa pristiglim i poslatim porukama za dati kontakt. Pritiskom na crveno dugme kao u predhodna dva slučaja vrši se minimizacija dijaloga. Žuto dugme sada pokreće *logout* funkciju čime se pokreće *LoggingOut* dijalog a nakon završene odjave i *Login* dijalog. Ukoliko ima pristiglih zahteva za prijateljstvo biće iscrtana odgovarajuća ikonica. Takođe će se u pomoćnom meniju pojaviti plavo dugme čijim pritiskom se pokreće *ConfirmContacts* dijalog za prihvatanje ili odbijanje pristiglih zahteva. Pritisak zelenog dugmeta pokreće dijalog za pretragu i slanje zahteva za prijateljstvo a pritiskom na *OK* dugme se blokira ili deblokira trenutno selektovani kontakt. Pritiskom na

dugmiće *PageUp* i *PageDown* selektuje se polje za unos poruke za slanje i polje u kojem se ispisuju primljene i poslate poruke za trenutno selektovani kontakt.

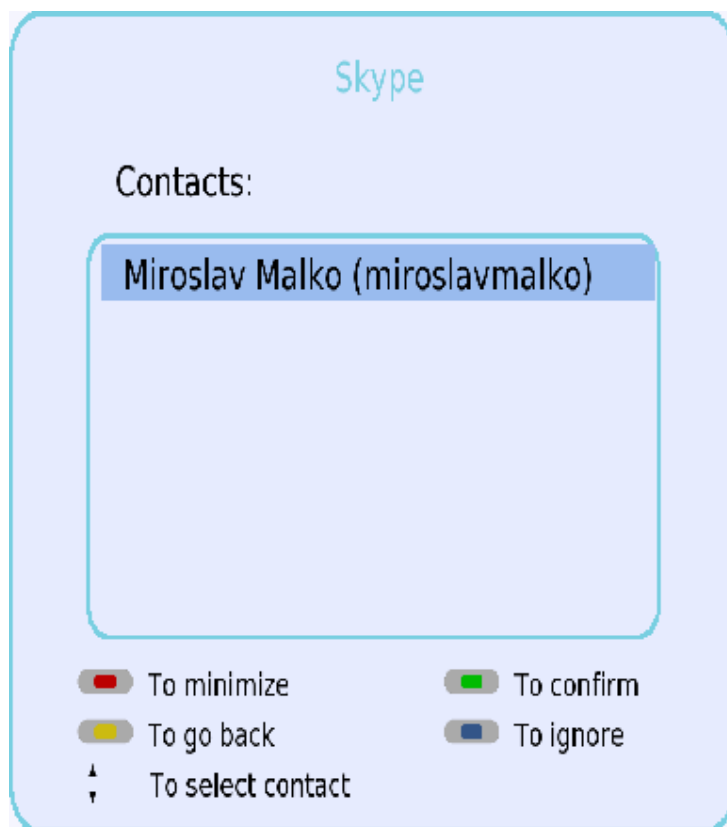


Slika 3.6 Izgled glavnog Skype dijaloga

Kada je selektovano polje za unos poruke za slanje i polje u kojem se ispisuju primljene i poslate poruke na dugmiće *Up* i *Down* vrši se listanje primljenih i poslatih tekstualnih poruka a sa dugmićima *Left* i *Right* se vrši selekcija karaktera koji se trenutno unosi. Na digitalne dugmiće se unosi trenutno selektovani karakter slično kucanju poruke na mobilnom telefonu s tim što se specijalni karakteri unose na nulu. Plavo dugme ima ulogu *caps lock*-a a zeleno *back space*-a. Poruka se šalje pritiskom na *OK* dugme. Takođe se pritiskom na crveno dugme minimizuje *Skype* dijalog a pritiskom na žuto dugme poziva se *logout* funkcija. Ako pristigne nova tekstualna poruka a upaljen je glavni *Skype* dijalog poruka se smešta u bafer i vrši se provera da li je pristigla poruka za trenutno selektovani kontakt i ako jeste osvežava se ispis. U slučaju da je dijalog isključen ulogu baferovanja poruka preuzima *DemoApp* i obaveštava korisnika o prispeću nove tekstualne poruke.

Treba još napomenuti da je broj baferovanih poruka za jednog korisnika ograničen zbog ograničene količine memorije koja je na raspolaganju. Takođe pri gašenju televizora ne sačuvavaju se poruke koje su bile u baferu zbog ograničene veličine NVRAM-a kao i ograničenog broja čitanja i upisa u njega.

Ukoliko je pored teksta “*Chat:*” sa desne strane iscrtana sličica koja znači da neko želi da bude na listi prijatelja, pritiskom na plavo dugme otvara se *ConfirmContacts* dijalog sa spiskom svih kontakata koji su poslali zahtev za prijateljstvo.

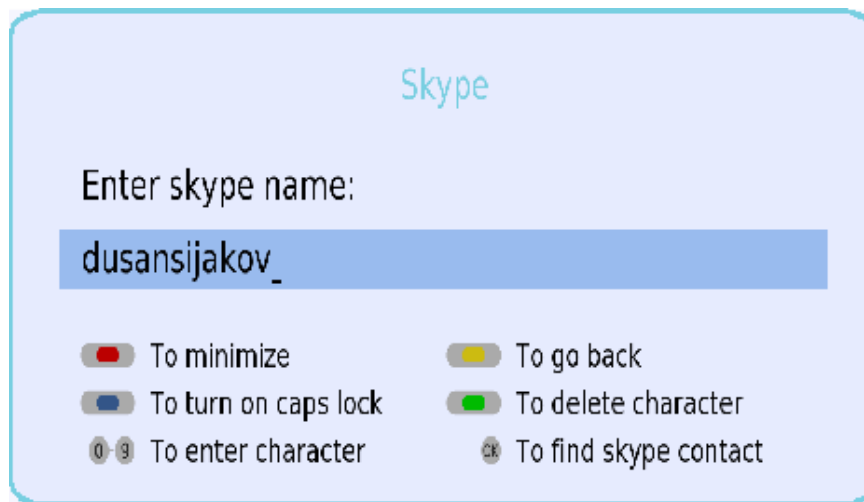


Slika 3.7 Izgled *ConfirmContacts* dijaloga

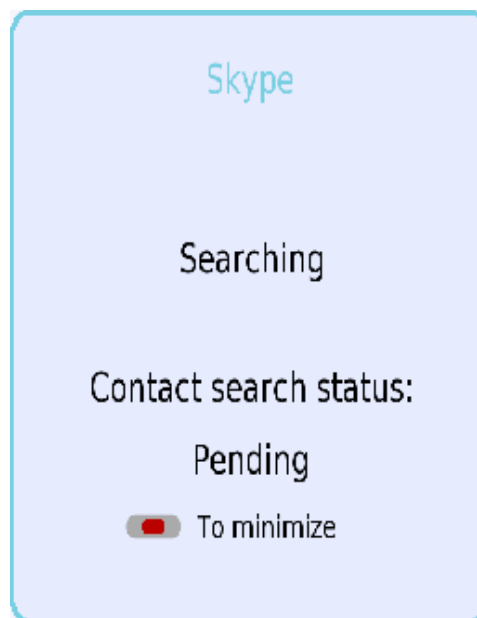
Na dugmiće *Up* i *Down* vrši se selekcija kontakta. Pritiskom na zeleno dugme kontakt se dodaje u listu prijatelja dok se pritiskom plavog dugmeta ignoriše. Takođe je moguće minimizirati dijalog pritiskom crvenog dugmeta i vratiti na glavni dijalog pritiskom žutog.

Pritiskom na zeleno dugme iz glavnog *Skype* dijaloga otvara se *FindContact* dijalog u koji se unose imena korisnika koje želimo da pronađemo. Može se uneti ili korisničko ime ili puno ime i prezime korisnika.

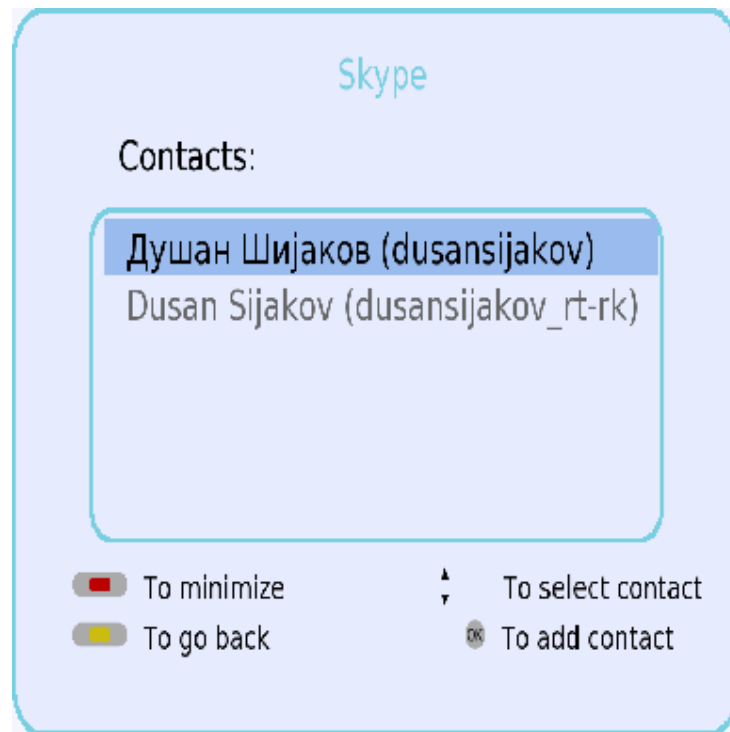


Slika 3.8 Izgle *FindContact* dijaloga

Kao i u predhodnim slučajevima karakter se selektuje na dugmiće *Left* i *Right* a na numeričke dugmiće se unosi trenutno selektovani karakter. Isto tako crveno dugme ima *minimize* funkciju a plavo i zeleno dugme *caps lock* i *back space*. Pritiskom na *OK* dugme započinje se pretraga i otvara se *FindingContact* dijalog. U ovom dijalogu se ispisuju informacije o procesu pronalaženja kontakta. I ovaj dijalog je moguće minimizirati pritiskom na crveno dugme.

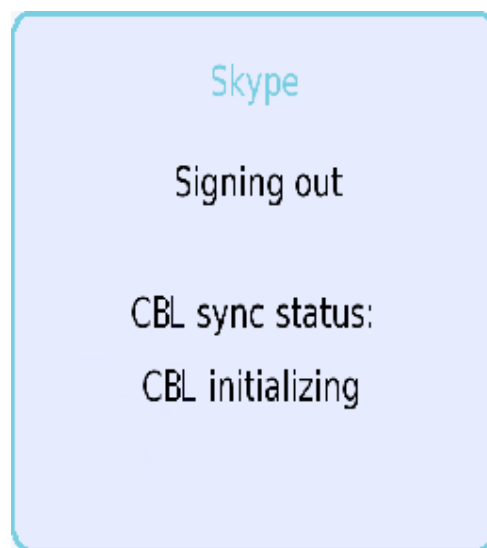
Slika 3.9 Izgled *FindingContact* dijaloga

Nakon uspešne pretrage otvara se *AddContact* dijalog u kome su ispisani kontakti koji su dobijeni kao rezultat pretrage.

Slika 3.10 Izgled *AddContact* dijaloga

Na dugmiće *Up* i *Down* vrši se selekcija kontakta a na dugme *OK* se kontakt dodaje u listu prijatelja. Takođe je moguće vratiti se na glavni Skype dijalog pritiskom na žuto dugme i minimizirati *Skype* pritiskom na crveno.

Iz glavnog *Skype* dijaloga pritiskom na žuto dugme započinje se proces odjavljivanja i otvara se *LoggingOut* dijalog. U ovom dijalogu se ispisuju sve informacije o procesu odjavljivanja.

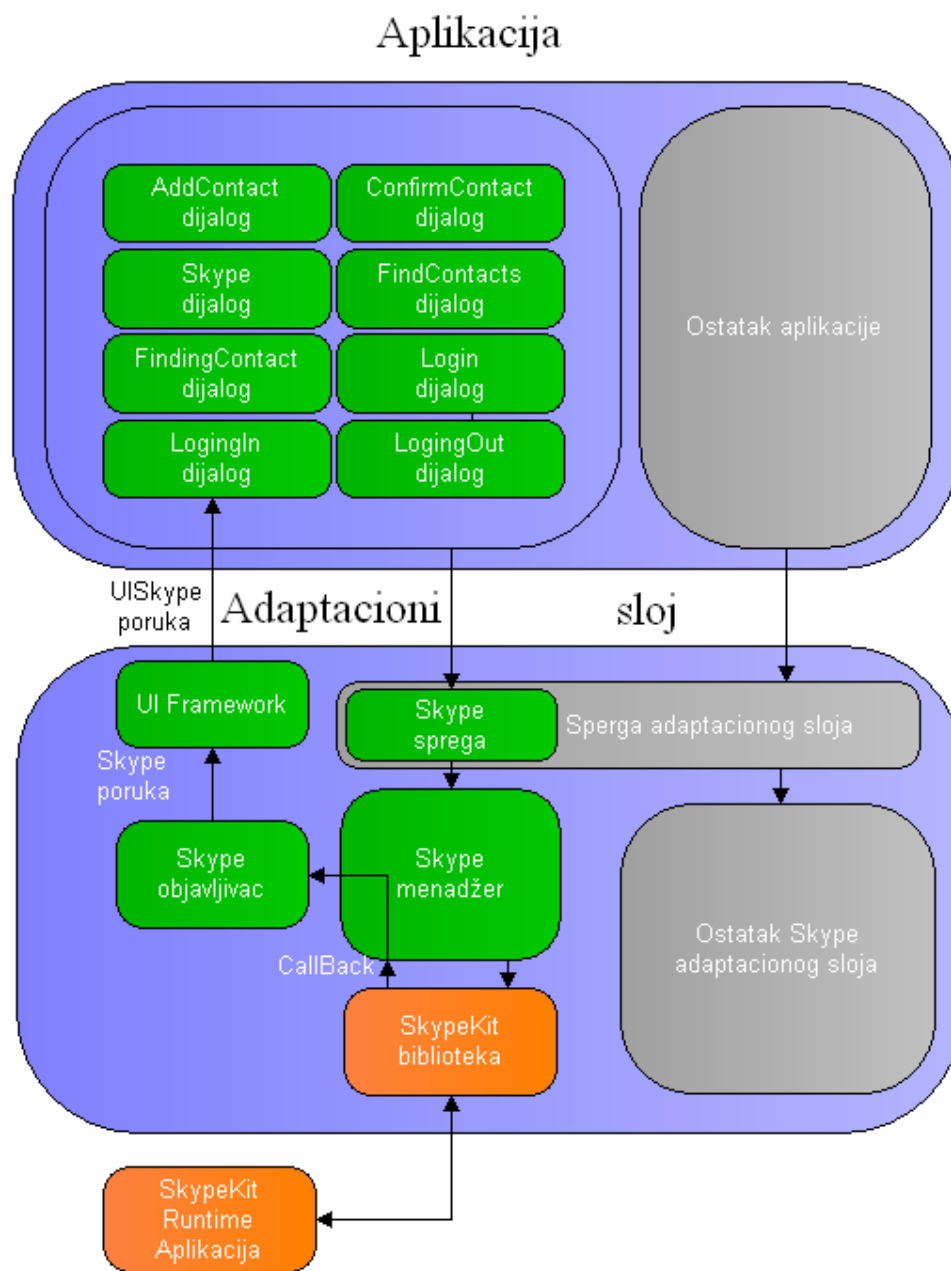
Slika 3.11 Izgled *LoggingOut* dijaloga

## 4. Opis realizacije

Moduli programskog rešenja se mogu podeliti u dve grupe:

- Moduli u kojima su realizovani dijalozi - Ovi moduli se nalaze unutar aplikacije. Preko njih se vrši komunikacija sa korisnikom. Oni delovanjem korisnika preko *UI Framework*-a pozivaju određene *Skype* funkcije koje se nalaze unutar adaptacionog sloja. Takođe primaju *UISkypeMessages* poruke kojima obaveštavaju korisnika kada se desi neki događaj.
- Moduli u kojima su realizovane sve *Skype* funkcije koje poziva aplikacija - Ovi moduli smešteni su unutar adaptacionog sloja. Moduli vrše komunikaciju sa *SkypeKit RinTime* aplikacijom i šalju određene *UI Skype* poruke (*UISkypeMessages*) preko *UI Framework*-a ukoliko se desio neki događaj. Ovi moduli ne znaju za gornju aplikaciju.

Na slici 4.1 prikazani su osnovni moduli, njihov raspored unutar slojeva i međusobna komunikacija. Zelenom bojom su označeni moduli koji su realizovani u procesu izrade ovog zadatka. Narandžastom bojom je označeno sve što se dobilo uz *SkypeKit* skup alata a sivom ostatak bojom aplikacije koji je ranije implementiran i u koju je integrisana *Skype* klijent aplikacija.



Slika 4.1 Raspored modula i međusobna komunikacija

## 4.1 Moduli u kojima su realizovani dijalozi

- *SkypeAddContactDialog* - Modul predstavlja dijalog koji se otvara nakon pretrage korisnika. Ima listu sa rezultatima pretrage sa mogućnošću dodavanja selektovanog kontakta.
- *SkypeConfirmContactDialog* - Modul predstavlja dijalog koji se pokreće kada korisnik stisne plavo dugme iz *SkypeDialog* a u tom trenutku postoje kontakti koji žele da se nađu u listi prijatelja. Ovaj dijalog poseduje listu u kojoj se ispisuju svi kontakti koji žele da se nađu u listi prijatelja. Dijalog poseduje opcije za ignorisanje i potvrdu ovih kontakata.
- *SkypeContactsBoxRow* – Modul predstavlja jednu kolonu *SkypeContactsListBox* liste. Jedna kolona ima tri polja. U prvom, skroz levo se iscrtava status ikonica, desno od nje ime kontakta i na kraju skroz desno se iscrtava kružić sa brojem primljenih poruka ukoliko ih ima.
- *SkypeContactsListBox* – Modul predstavlja listu u kojoj se ispisuju kontakti koji se nalaze u listi prijatelja.
- *SkypeDialog* - Modul predstavlja dijalog koji se otvara nakon uspešnog prijavljivanja. Sadrži listu kontakata i kretanjem kroz nju ispisuje primljene i poslate poruke za trenutno selektovani kontakt. Takođe poseduje opcije za slanje nove poruke, otvaranje dijaloga za pretragu korisnika i dijaloga za potvrdu korisnika koji žele da se nađu u listi prijatelja.
- *SkypeFindContactsDialog* - Modul predstavlja dijalog koji se otvara nakon pritiska na zeleno dugme iz *SkypeDialog*-a. U njemu se unose podaci potrebni za pretragu korisnika.
- *SkypeFindingContactsDialog* - Modul predstavlja dijalog koji se otvara nakon započete pretrage korisnika. U njemu se ispisuju određene informacije o pretrazi.
- *SkypeLoginDialog* - Modul koji predstavlja prvi dijalog koji se otvara pri pokretanju Skype-a. Ima polja za unos korisničkog imena, lozinke i polje za potvrdu u kojem se bira da li da se zapamti korisničko ime i lozinka.
- *SkypeLoggingInDialog* - Modul koji predstavlja dijalog koji se pokreće nakon pritiska na *OK* dugme predhodnog dijaloga. Ispisuje informacije o prijavljivanju i razlog neuspešnog prijavljivanja ukoliko je do njega došlo.

- *SkypeLoggingOutDialog* - Modul koji predstavlja dijalog koji se pokreće nakon pritiska na žuto dugme iz *SkypeDialog*-a. Ispisuje informacije o procesu odjavljivanja

## 4.2 Moduli u kojima su realizovane Skype funkcije

- *AnySkypeEventListener* - Modul koje treba da nasledi svaki oslušivač koji hoće da se prijavi kod nekog *SkypeEventPublisher*-a da bi primao *SkypeMessage* poruke
- *AnySkypeEventPublisher* - Modul koji treba da nasledi svaki objavljiivač koji želi da prima *SkypeMessage* poruke i da ih prosleđuje registrovanim *AnySkypeEventListener*-ima
- *AnySkypeObserver* - Modul koji treba da nasledi bilo koji posmatrač koji želi da prima *SkypeMessage* poruke i šalje ih gore preko svojih objavljiivača
- *SkypeCodes* - Modul u kojem su definisani tipovi svih *Skype* poruka
- *SkypeEventPublisher* - Nasleđuje *AnySkypeEventPublisher* i redefiniše njene virtualne metode, kada *SkypeObserver* primi poruku on je preko svog objavljiivača šalje gore prema registrovanim oslušivaćima.
- *SkypeManager* - Najbitniji modul. U njemu se nalaze sve funkcije, promenljive i potrebne informacije koje su potrebne kako bi se obezbedila *Skype* funkcionalnost. Dijalozi pozivaju funkcije ovog modula a on ih putem *SkypeMessages* poruka obaveštava o nekim događajima.
- *SkypeMessage* - Nasleđuje klasu *UIAnyMessage* i predstavlja tip poruke koju *SkypeManager* šalje ka *SkypeEventListener*-ima
- *UISkypeMessage* - Nasleđuje klasu *UIAnyMessage* i predstavlja tip poruke koju *UIFramework* šalje ka *DemoApp*-u i dijalogima.
- *UserConversation* – Ovaj modul nosi informacije o tekstualnim porukama za jednog korisnika.
- *UserConversationMessage* – Ovaj modul nosi informacije o jednoj tekstualnoj poruci.

## 5. Testiranje

Aplikacija je testirana u *Cabot Observatory* okruženju na Windows XP operativnom sistemu. Urađen je 21 testni slučaj i aplikacija je uspešno prošla sve slučajeve. U tabeli 5.1 dat je spisak i opis testnih slučajeva kao i očekivanih rezultata.

Testni slučajevi			
Rb	Opis testa	Očekivani rezultat	Uspešnost
1	<p>Opis: Cilj testa je provera uspešnosti prijavljivanja</p> <p>Početni uslov: Ne sme biti sačuvana korisnička lozinka u prethodnom prijavljivanju</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se aplikacija</li> <li>2. Pritisak na plavo dugme</li> <li>3. Unos ispravnog korisničkog imena</li> <li>4. Unos ispravne lozinke</li> <li>5. Pritisak na <i>OK</i> dugme</li> </ol>	Nakon pritiska na <i>OK</i> dugme treba da se pojavi dijalog u kojem će se ispisivati informacije o prijavljivanju i nakon uspešnog prijavljivanja da se pojavi glavni <i>Skype</i> dijalog	PROŠAO
2	Opis: Cilj testa je provera automatskog prijavljivanja sa prethodno sačuvanim	Nakon ponovnog pokretanja aplikacije treba automatski da	PROŠAO

	<p>korisničkim imenom i lozinkom</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se aplikacija</li> <li>2. Pritisak na plavo dugme</li> <li>3. Unos ispravnog korisničkog imena</li> <li>4. Unos ispravne lozinke</li> <li>5. Gašenje <i>Skype</i> aplikacije</li> <li>6. Ponovno pokretanje <i>Skype</i> aplikacije</li> </ol>	<p>se prijavi korisnik koji je prethodno bio prijavljen i pri tom zapamtio korisničko ime i lozinku</p>	
3	<p>Opis: Cilj testa je provera ispisa greške pri unosu nepostojećeg korisničkog imena</p> <p>Početni uslov: Ne sme biti sačuvana korisnička lozinka u prethodnom prijavljivanju</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se aplikacija</li> <li>2. Pritisak na plavo dugme</li> <li>3. Unos nepostojećeg korisničkog imena</li> <li>4. Unos lozinke</li> <li>5. Pritisak na <i>OK</i> dugme</li> </ol>	<p>Nakon pritiska na <i>OK</i> dugme treba da se pojavi dijalog u kojem će se ispisivati informacije o prijavljivanju. Nakon neuspešnog prijavljivanja treba da se pojavi crveni tekst sa porukom: <i>No such identity</i></p>	PROŠAO
4	<p>Opis: Cilj testa je provera ispisa greške pri nepravilno unetoj lozinci</p> <p>Početni uslov: Ne sme biti sačuvana korisnička lozinka u prethodnom prijavljivanju</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se aplikacija</li> <li>2. Pritisak na plavo dugme</li> </ol>	<p>Nakon pritiska na <i>OK</i> dugme treba da se pojavi dijalog u kojem će se ispisivati informacije o prijavljivanju. Nakon neuspešnog prijavljivanja treba da se pojavi crveni tekst sa porukom: <i>Incorrect password.</i></p>	PROŠAO



	<ol style="list-style-type: none"> <li>3. Unos korisničkog imena</li> <li>4. Unos nepravilne lozinke</li> <li>5. Pritisak na <i>OK</i> dugme</li> </ol>		
5	<p>Opis: Cilj testa je provera ispisa greške tokom prijavljivanja pri nestanku interneta</p> <p>Početni uslov: Ne sme biti sačuvana korisnička lozinka u prethodnom prijavljivanju</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se aplikacija</li> <li>2. Pritisak na plavo dugme</li> <li>3. Unos korisničkog imena</li> <li>4. Unos lozinke</li> <li>5. Pritisak na <i>OK</i> dugme</li> <li>6. Isključenje mrežnog kabla kako bi prekinuli internet vezu</li> </ol>	<p>Nakon pritiska na <i>OK</i> dugme treba da se pojavi dijalog u kojem će se ispisivati informacije o prijavljivanju. Nakon neuspešnog prijavljivanja treba da se pojavi crveni tekst sa porukom: <i>p2p connect failed</i></p>	PROŠAO
6	<p>Opis: Cilj testa je provera ispisa status ikonice korisnika u listi prijatelja.</p> <p>Početni uslov: Mora postojati i drugi korisnik koji će biti prijavljen na drugoj <i>Skype</i> klijent aplikaciji i menjati svoj status</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Na drugu <i>Skype</i> klijent aplikaciju se prijaviti sa korisnikom koji se nalazi u spisku prijatelja aplikacije koja se testira.</li> <li>3. U drugoj <i>Skype</i> klijent aplikaciji menjati status</li> </ol>	<p>Nakon prijavljivanja i nakon menjanja statusa korisnika unutar neke druge <i>skype</i> aplikacije u listi prijatelja aplikacije koja se testira treba da se isctavaju odgovarajuće ikonice. (<i>online, busy, away, offline, blocked</i>)</p>	PROŠAO

7	<p>Opis: Cilj testa je provera prijema i ispisa poruke kada je kontakt selektovan.</p> <p>Početni uslov: Mora postojati i drugi korisnik koji će biti prijavljen i slati poruke</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Selektovati korisnika koji će da šalje poruke</li> <li>3. Na drugou <i>Skype</i> klijent aplikaciju se prijaviti sa korisnikom koji se nalazi u spisku prijatelja u aplikaciji koju testiramo</li> <li>4. Poslati nekoliko poruka iz druge <i>Skype</i> klijent aplikacije.</li> </ol>	<p>Nakon prijema poruke ukoliko je selektovan baš kontakt koji ju je poslao u polju za ispis treba da se pojavi tekst poruke.</p>	PROŠAO
8	<p>Opis: Cilj testa je provera prijema poruke i ispisa žutog kružića sa brojem primljenih poruka kada korisnik koji šalje poruku nije selektovan.</p> <p>Početni uslov: Mora postojati i drugi korisnik koji će biti prijavljen na drugu <i>Skype</i> klijent aplikaciju i slati poruke</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. U drugoj <i>Skype</i> klijent aplikaciji se prijaviti sa korisnikom koji se nalazi u spisku prijatelja u aplikaciji koju testiramo</li> <li>3. Poslati nekoliko poruka iz te druge <i>Skype</i> aplikacije.</li> </ol>	<p>Nakon uspešnog prijavljivanja i prijema poruke, ukoliko kontakt koji šalje poruku nije selektovan u listi prijatelja, desno od njegovog imena treba da se iscrta kružić u kojem će se ispisati broj primljenih poruka dok polje za ispis selektovanog korisnika treba da ostane nepromenjeno.</p>	PROŠAO
9	<p>Opis: Cilj testa je provera mogućnosti minimizacije prozora i njegovog ponovnog</p>	<p>Nakon pokretanja <i>Skype</i> klijent aplikacije pritiskom na crveno</p>	PROŠAO

	<p>pokretanja kada korisnik nije prijavljen.</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se Skype klijent aplikacija</li> <li>2. Nakon pokretanja <i>Skype</i> klijent aplikacije pritisnuti crveno dugme</li> <li>3. Pritisak na plavo dugme</li> </ol>	<p>dugme treba da se ugasi prozor. Ponovnim pritiskom na plavo dugme treba da se pojavi isti prozor.</p>	
10	<p>Opis: Cilj testa je provera mogućnosti minimizacije prozora i njegovog ponovnog pokretanja kada je korisnik prijavljen.</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se Skype klijent aplikacija</li> <li>2. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>3. Nakon prijavljivanja pritisnuti crveno dugme</li> <li>4. Pritisak na plavo dugme</li> </ol>	<p>Nakon pokretanja <i>Skype</i> klijent aplikacije i uspešnog prijavljivanja pritiskom na crveno dugme treba da se ugasi prozor. Ponovnim pritiskom na plavo dugme treba momentalno da se pojavi taj isti prozor (korisnik mora ostati i dalje prijavljen)</p>	PROŠAO
11	<p>Opis: Cilj testa je provera mogućnosti minimizacije prozora i njegovog ponovnog pokretanja u procesu prijavljivanja.</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Pokreće se aplikacija</li> <li>2. Pritisak na plavo dugme</li> <li>3. Unos korisničkog imena</li> <li>4. Unos lozinke</li> <li>5. Pritisak na <i>OK</i> dugme</li> <li>6. Pre uspešnog prijavljivanja (u procesu prijavljivanja) pritisnuti crveno dugme</li> <li>7. Pritisak na plavo dugme</li> </ol>	<p>Nakon pokretanja <i>Skype</i> klijent aplikacije i započetog postupka prijavljivanja pritiskom na crveno dugme treba da se ugasi prozor. Ponovnim pritiskom na plavo dugme treba momentalno da se pojavi jedan od 3 moguća prozora.</p> <ol style="list-style-type: none"> <li>1. <i>Login</i> prozor ukoliko prijavljivanje nije uspelo</li> <li>2. <i>LogingIn</i> prozor ukoliko je prijavljivanje još u toku</li> <li>3. Glavni <i>Skype</i> prozor ako je prijavljivanje</li> </ol>	PROŠAO

		uspelo	
12	<p>Opis: Cilj testa je provera mogućnosti minimizacije prozora kada je otvoren neki od <i>FindContact</i> ili <i>AddContact</i> dijaloga .</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Odlazak u neki od ovih dijaloga</li> <li>3. Pritisak na crveno dugme</li> <li>4. Pritisak na plavo dugme</li> </ol>	<p>Pritiskom na crveno dugme treba da se ugasi prozor. Ponovnim pritiskom na plavo dugme treba momentalno da se pojavi glavni <i>Skype</i></p>	PROŠAO
13	<p>Opis: Cilj testa je provera ispisa obavještajuće poruke po prijemu <i>Skype</i> poruke kada je <i>Skype</i> klijent aplikacija minimizirana.</p> <p>Početni uslov: Mora postojati i drugi korisnik koji će biti prijavljen na drugoj <i>Skype</i> klijent aplikaciji i slati poruke</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Nakon uspešnog prijavljivanja pritisnuti crveno dugme kako bi se aplikacija minimizirala.</li> <li>3. Prijaviti se na drugu <i>Skype</i> klijent aplikaciju sa korisnikom koji se nalazi u spisku prijatelja aplikacije koja se testira i poslati nekoliko poruka.</li> </ol>	<p>Po prijemu poruke dok je <i>Skype</i> klijent aplikacija minimizirana na ekranu treba da se ispiše poruka sa imenom korisnika koji ju je poslao i opcija u kojoj je moguće upaliti <i>Skype</i> klijent ili ugaziti poruku. Ukoliko se ne odabere ni jedna od ponuđenih opcija poruka se gasi sama u roku od 6 sekundi.</p>	PROŠAO
14	<p>Opis: Cilj testa je provera uspešnog prijema poruke kada ona stigne dok je <i>Skype</i> klijent minimiziran a nakon prijema se ponovo pokrene.</p> <p>Početni uslov: Mora postojati i drugi</p>	<p>Po prijemu poruke dok je <i>Skype</i> klijent aplikacija minimizirana ponovo pokrenuti <i>Skype</i> klijent aplikaciju. Ukoliko nakon pokretanja</p>	PROŠAO

	<p>korisnik koji će biti prijavljen na drugoj <i>Skype</i> klijent aplikaciji i slati poruke</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Nakon uspešnog prijavljivanja pritisnuti crveno dugme kako bi se aplikacija minimizirala.</li> <li>3. Prijaviti se sa korisnikom koji se nalazi u spisku prijatelja u aplikacije koja se testira i poslati nekoliko poruka.</li> <li>4. Ponovo pokrenuti <i>Skype</i> klijent</li> </ol>	<p>aplikacije nije selektovan korisnik koji je poslao poruku pored njegovog imena u listi kontakata treba da se se iscrta žuti kružić sa brojem primljenih poruka. U suprotnom u polju za ispis treba da se ispiše tekst primljenih poruka</p>	
15	<p>Opis: Cilj testa je provera <i>block</i> funkcije</p> <p>Početni uslov: Korisnik kojeg želimo da blokiramo ne sme prethodno biti blokiran jer ćemo ga ovim postupkom odblokirati</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Selektovati nekog korisnika i pritisnuti <i>OK</i> dugme</li> </ol>	<p>Nakon pokretanja <i>Skype</i> klijent aplikacije i blokiranja nekog korisnika par sekundi posle treba da se promeni i njegova status ikonica u crvenu <i>block</i> ikonu.</p>	PROŠAO
16	<p>Opis: Cilj testa je provera <i>unblock</i> funkcije</p> <p>Početni uslov: Korisnik kojeg želimo da deblokiramo mora prethodno biti blokiran ili ćemo ga u suprotnom ovim postupkom blokirati</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Selektovati nekog korisnika i pritisnuti <i>OK</i> dugme</li> </ol>	<p>Nakon pokretanja <i>Skype</i> klijent aplikacije i deblokiranja nekog korisnika par sekundi posle treba da se promeni i njegova status ikonica iz crvene <i>block</i> ikone u neku od preostale četiri status ikone u zavisnosti od toga koji je status korisnika (<i>online, busy, away, offline</i>).</p>	PROŠAO
17	<p>Opis: Cilj testa je provera ispisa ikonice</p>	<p>Nakon ovoga u testnoj</p>	PROŠAO

	<p>koja nas obaveštava da nas je neki kontakt dodao za prijatelja.</p> <p>Početni uslov: Mora postojati i drugi korisnik prijavljen na drugoj <i>Skype</i> klijent aplikaciji kojeg trenutno nemamo u listi prijatelja</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. U drugoj <i>Skype</i> aplikaciji se prijaviti sa korisnikom koji se ne nalazi u spisku prijatelja u aplikaciji koju testiramo</li> <li>3. Dodati iz te druge aplikacije za prijatelja kontakt koji je prijavljen na testnoj aplikaciji</li> </ol>	<p>aplikaciji treba da se iscrta ikonica koja signalizira da je neko poslao zadatak za prijateljstvo. Ovim takođe treba da se u <i>HelpBar</i>-u pojavi opcija za odlazak u listu zahteva.</p>	
18	<p>Opis: Cilj testa je provera funkcije za potvrdu korisnika koji nas je dodao za prijatelja.</p> <p>Početni uslov: Mora postojati i drugi korisnik prijavljen na drugoj <i>Skype</i> klijent aplikaciji kojeg trenutno nemamo u listi prijatelja</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se na <i>Skype</i> klijent aplikaciju</li> <li>2. U drugoj <i>Skype</i> aplikaciji se prijaviti sa korisnikom koji se ne nalazi u spisku prijatelja u testnoj aplikaciji</li> <li>3. Dodati iz te druge aplikacije za prijatelja kontakt koji je ulogovan</li> </ol>	<p>Nakon svega ovoga pritiskom na zeleno dugme potvrđuje se zahtev za prijateljstvo. Posle ovoga u listi prijatelja treba da se pojavi korisnik koji je poslao zahtev za prijateljstvo.</p>	PROŠAO

	<p>na testnoj aplikaciji</p> <p>4. Iz aplikacije pritiskom na plavo dugme pokrenuti dijalog za dodavanje korisnika.</p> <p>5. Pritisnuti zeleno dugme</p>		
19	<p>Opis: Cilj testa je provera funkcije za ignorisanje korisnika koji nas je dodao za prijatelja.</p> <p>Početni uslov: Mora postojati i drugi korisnik prijavljen na drugoj <i>Skype</i> klijent aplikaciji kojeg trenutno nemamo u listi prijatelja</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Na drugu <i>Skype</i> klijent aplikaciju se prijaviti sa korisnikom koji se ne nalazi u spisku prijatelja aplikacije koja se testira</li> <li>3. Dodati iz te druge aplikacije za prijatelja kontakt koji je prijavljen u aplikaciju koja se testira</li> <li>4. Iz aplikacije koja se testira pritiskom na plavo dugme pokrenuti dijalog za dodavanje korisnika.</li> <li>5. Pritisnuti zeleno dugme</li> </ol>	<p>Nakon svega ovoga pritiskom na plavo dugme ignoriše se zahtev za prijateljstvo. Posle ovoga u listi prijatelja ne treba da bude nikakvih promena dok kod korisnika koji je poslao zahtev za prijateljstvo status korisnika kojeg je dodao treba da bude <i>offline</i></p>	PROŠAO
20	<p>Opis: Cilj testa je provera funkcije za pretragu korisnika</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent</li> </ol>	<p>Nakon unetog teksta za pretragu i završetka procesa pretrage treba da se pojavi dijalog sa listom rezultata</p>	PROŠAO

	<p>aplikaciju</p> <ol style="list-style-type: none"> <li>2. Pritiskom na plavo dugme pokrenuti dijalog za unos teksta za pretragu</li> <li>3. Pritiskom na <i>OK</i> dugme pokrenuti proces pretrage</li> </ol>	pretrage	
21	<p>Opis: Cilj testa je provera funkcije za dodavanje korisnika</p> <p>Početni uslov: Mora postojati i drugi korisnik prijavljen na drugoj <i>Skype</i> klijent aplikaciji kojeg trenutno nemamo u listi prijatelja</p> <p>Postupak:</p> <ol style="list-style-type: none"> <li>1. Prijaviti se u <i>Skype</i> klijent aplikaciju</li> <li>2. Funkcijom za pretragu korisnika dobiti listu rezultat i selektovati željenog korisnika</li> <li>3. Pritiskom na zeleno dugme poslati zahtev za prijateljstvo</li> </ol>	<p>Nakon selektovanja korisnika iz liste sa rezultatima pretrage pritiskom na zeleno dugme se šalje zahtev za prijateljstvo. Nakon ovoga kontakt će se pojaviti u listi korisnika a njegov status zavisi od toga da li je prihvatio zahtev ili ne. Ako ne prihvati zahtev status će mu stalno biti <i>offline</i></p>	PROŠAO

Tabela 5.1 Tabela testnih slučajeva



## 6. Zaključak

U ovom radu je implementirano jedno rešenje Skype klijenta sa osnovim funkcijama koje Skype nudi. Velika je potreba za razvojem ovakvih aplikacija namenjenih set-top box-ovima i televiziji jer skoro i da nema ovakvih aplikacija.

U aplikaciji su implementirane funkcije za prijavljivanje na Skype, slanje i prijem tekstualnih poruka, pretragu korisnika i slanje zahteva za prijateljstvo, potvrdu i ignorisanje primljenih zahteva, blokiranje i brisanje korisnika. U radu je prikazan razvoj aplikacije od jezgra koje vrši komunikaciju sa *Skype RunTime*-om do grafičke sprege sa korisnikom. U jezgro aplikacije je ugrađen *SkypeKit* skup alata i funkcija koji je i napravljen kako bi se razvoj ovakvih aplikacija do neke mere olakšao. Grafička sprega je napravljena pomoću komponenti koje nudi *UI Framework*.

Može se zaključiti da je aplikacija ispunila sve definisane zahteve i uspešno prošla sve testove. Ovo je potvrđeno u okviru *Cabot Observatory* simulatora, a na samom kraju aplikacija je uspešno testirana i na ciljnoj platformi.

Daljim razvojem treba poboljšati izgled aplikacije i proširiti skup podržanih funkcija koje Skype nudi, pre svega audio i video pozive. Da bi ovo bilo moguće televizori bi morali biti opremljeni odgovarajućim mikrofonom i kamerom.

## 7. Literatura

- [1] BCM3556 SoC, <http://www.broadcom.com/products/Digital-TV/Digital-TV-Solutions/BCM3556>
- [2] Aurora arhitektura, [http://cabot.com.tr/download\\_data/pdf/EclipseDataSheet.pdf](http://cabot.com.tr/download_data/pdf/EclipseDataSheet.pdf)
- [3] CABOT Communications, *DTV UI Differentiation Using SVG*
- [4] SVG, [http://en.wikipedia.org/wiki/Scalable\\_Vector\\_Graphics](http://en.wikipedia.org/wiki/Scalable_Vector_Graphics)
- [5] PNG, [http://en.wikipedia.org/wiki/Portable\\_Network\\_Graphics](http://en.wikipedia.org/wiki/Portable_Network_Graphics)
- [6] Skype Kit, <https://developer.skype.com/public/skypekit>