



УНИВЕРЗИТЕТ У НОВОМ САДУ  
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У  
НОВОМ САДУ



Александар Лугоња

**Реализација клијентске апликације за  
управљање паметним ТВ  
пријемницима на iOS платформи**

**ДИПЛОМСКИ РАД**  
**- Основне академске студије -**

Нови Сад, јун 2014



## КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР:			
Идентификациони број, ИБР:			
Тип документације, ТД:	Монографска документација		
Тип записа, ТЗ:	Текстуални штампани материјал		
Врста рада, ВР:	Завршни (Bachelor) рад		
Аутор, АУ:	Александар Лугоња		
Ментор, МН:	др Иштван Пап		
Наслов рада, НР:	Реализација клијентске апликације за управљање паметним ТВ пријемницима на iOS платформи		
Језик публикације, ЈП:	Српски / латиница		
Језик извода, ЈИ:	Српски		
Земља публиковања, ЗП:	Република Србија		
Уже географско подручје, УГП:	Војводина		
Година, ГО:	2014		
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)	7/46/0/3/17/0/0		
Научна област, НО:	Електротехника и рачунарство		
Научна дисциплина, НД:	Рачунарска техника		
Предметна одредница/Кључне речи, ПО:	ДТВ, iOS, iPad, Графичка корисничка спрела		
УДК			
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад		
Важна напомена, ВН:			
Извод, ИЗ:	У овом раду приказана је реализација пројектовања графичке корисничке спреле апликације за iOS преносне уређаје која репродукује ТВ садржај послат од стране ДТВ пријемника. Описан је комплетан изглед и начин реализације компоненти графичке корисничке спреле.		
Датум прихватања теме, ДП:			
Датум одбране, ДО:			
Чланови комисије, КО:	Председник:	Др Невојша Пјевалица	
	Члан:	Др Илија Башичевић	Потпис ментора
	Члан, ментор:	Др Иштван Пап	



## KEY WORDS DOCUMENTATION

Accession number, <b>ANO:</b>		
Identification number, <b>INO:</b>		
Document type, <b>DT:</b>	Monographic publication	
Type of record, <b>TR:</b>	Textual printed material	
Contents code, <b>CC:</b>	Bachelor Thesis	
Author, <b>AU:</b>	Aleksandar Lugonja	
Mentor, <b>MN:</b>	Ištván Pap, PhD	
Title, <b>TI:</b>	Realisation of client application for controlling smart TV set on iOS platform	
Language of text, <b>LT:</b>	Serbian	
Language of abstract, <b>LA:</b>	Serbian	
Country of publication, <b>CP:</b>	Republic of Serbia	
Locality of publication, <b>LP:</b>	Vojvodina	
Publication year, <b>PY:</b>	2014	
Publisher, <b>PB:</b>	Author's reprint	
Publication place, <b>PP:</b>	Novi Sad, Dositeja Obradovica sq. 6	
Physical description, <b>PD:</b> (chapters/pages/ref./tables/pictures/graphs/appendixes)	7/46/0/3/17/0/0	
Scientific field, <b>SF:</b>	Electrical Engineering	
Scientific discipline, <b>SD:</b>	Computer Engineering, Engineering of Computer Based Systems	
Subject/Key words, <b>S/KW:</b>	DTV, iOS, iPad, Graphical User Interface	
<b>UC</b>		
Holding data, <b>HD:</b>	The Library of Faculty of Technical Sciences, Novi Sad, Serbia	
Note, <b>N:</b>		
Abstract, <b>AB:</b>	This paper presents solution graphical user interface design for portable devices based on iOS platform that displays reproduction of the TV content delivered from DTV device. The paper describes design and implementation of the components created and used in graphical user interface.	
Accepted by the Scientific Board on, <b>ASB:</b>		
Defended on, <b>DE:</b>		
Defended Board, <b>DB:</b>	President: Member: Member, Mentor:	Nebojša Pejvalica, PhD Ilija Bašićević, PhD Ištván Papp, PhD
		Menthor's sign

## **Zahvalnost**

Zahvaljujem se institutu RT-RK na pruženoj mogućnosti za realizaciju ovog rada.

Takođe se zahvaljujem mentoru dr Ištvanu Papu i stručnom saradniku Romanu Pavloviću na savetima i stručnoj pomoći prilikom izrade ovog rada.

Posebno se zahvaljujem članovima svoje porodice na pruženoj podršci u toku čitavog školovanja.

## SADRŽAJ

1.	Uvod.....	1
2.	Teorijske osnove .....	2
2.1	Digitalna televizija .....	2
2.2	iOS Platforma.....	3
2.3	Grafička korisnička sprege uređaja zasnovanih na iOS platformi (Cocoa Touch) ..	5
2.4	Xcode razvojno okruženje.....	6
2.5	Način realizacije grafičke korisničke sprege na iOS platformi.....	7
2.6	Princip rada dodatnog ekrana u prikazu televizije .....	8
3.	Koncept rešenja.....	10
3.1	Zahtevi aplikacije .....	10
3.2	Interakcija aplikacije sa sekundarnim ekranom sa ostalim uređajima .....	12
3.3	Arhitektura najvišeg nivoa aplikacije.....	13
4.	Programsko rešenje.....	15
4.1	Elementi navigacionog stabla.....	15
4.2	Moduli korisničke sprege .....	16
4.2.1	MainMenuViewController .....	16
4.2.2	ChannelListViewController.....	18
4.2.3	LiveTVViewController.....	18
4.2.4	EPGViewController.....	20
4.2.5	RemoteControllerViewController,	
	TouchpadViewController,	
	KeyboardViewController.....	23
4.2.6	MosaicViewController .....	25
4.2.7	MediaExplorerViewController .....	25

4.2.8    SettingsViewControlller .....	27
5.    Testiranje i verifikacija .....	28
5.1    Ručno ispitivanje programa .....	28
5.2    Automatsko testiranje.....	30
5.3    Profilisanje .....	31
5.4    Poređenje sa postojećim rešenjima .....	33
6.    Zaključak .....	35
7.    Literatura.....	36

## SPISAK SLIKA

Slika 1 GUI Arhitektura iOS platforme .....	5
Slika 2 Xcode razvojno okruženje .....	7
Slika 3 Komunikacioni protokoli korišteni za realizaciju aplikacije sa dodatnim ekranom .....	12
Slika 4 Blok dijagram najvišeg nivoa aplikacije aplikacije .....	13
Slika 5 Prikaz stabla aplikacije sa navigacionim putevima .....	16
Slika 6 Prikaz glavnog menija.....	17
Slika 7 Prikaz TV prenosa na uređaju sa sekundarnim ekranom.....	19
Slika 8 Prikaz gradivnih komponenti Elektronskog Programskog Vodiča.....	20
Slika 9 Prikaz elektronskog programskega vodiča .....	21
Slika 10 Prikaz asinhronne komunikacije prilikom dobijanja EPG sadržaja.....	21
Slika 11 Prikaz gradivnih komponenti grafičke korisničke sprege EPG-a.....	22
Slika 12 Daljinsko upravljanje sačinjeno iz modula: tastatura, touchpad i daljinski upravljač .....	24
Slika 13 Mozaični prikaz sadržaja na sekundarnom ekranu .....	25
Slika 14 Prikaz Media Explorer modula .....	27
Slika 15 Primer skripte korištene za automatsko testiranje aplikacije.....	30
Slika 16 Izgled grafika unutar okruženja zaprofilisanje aplikacije .....	32
Slika 17 Uporedni prikaz postojećih rešenja i mog rešenja .....	34

## SPISAK TABELA

Tabela 1 Testiranje grafičke korisničke sprege.....	29
Tabela 2 Prikaz zauzetosti memorije i procesora u LiveTV delu aplikacije.....	31
Tabela 3 Prikaz zauzetosti memorije i procesora u EPG delu aplikacije.....	32

## SKRAĆENICE

<b>DTV</b>	- <i>Digital Television</i> , Digitalna televizija
<b>GUI</b>	- <i>Graphical User Interface</i> , Grafička korisnička sprega
<b>TV</b>	- <i>Television</i> , Televizija
<b>DLNA</b>	- <i>Digital Living Network Alliance</i>
<b>UI</b>	- <i>User Interface</i> , Korisnička Sprega
<b>EPG</b>	- <i>Electronic Program Guide</i> , Elektronski programski vodič
<b>STB</b>	- <i>Set-top box</i> , prijemnik digitalnog televizijskog sadržaja
<b>UPNP</b>	- <i>Universal Plug and Play</i> , komunikacioni protokol

## 1. Uvod

U ovom radu predstavljena je realizacija klijentske aplikacije koja reprodukuje TV sadržaj i rukuje TV prijemnikom pomoću prenosnih uređaja zasnovanih na iOS platformi. Rešenje je realizovano za platforme sa iOS podrškom za verzije novije od 5.0. Realizovana aplikacija podržava sve generacije iPad uređaja, od iPad 1, pa sve do najnovijeg podržanog, iPad Air-a.

Glavni doprinos aplikacije jeste mogućnost rukovanja TV sadržajem koristeći prenosive uređaje. Sekundarni (dodatni) ekran je termin koji se odnosi na dodatni električni uređaj kao što je pametni telefon ili tablet (npr. iPad), koji omogućava TV gledaocima pristup sadržaju koji koriste poput TV emisija, video snimaka ili muzike.

Realizacija aplikacije treba da obuhvati i zadovolji sledeće zahteve:

- Interakcija sa korisnikom mora biti na visokom nivou, što podrazumeva brz odziv aplikacije, jednostavnost elemenata grafičke sprege kao i navigacije.
- Prenosni uređaj mora da oponaša sve funkcionalnosti naprednog daljinskog upravljača.
- Podrška za DLNA multimedijalne uređaje
- Prikaz EPG vodica
- Prikaz grafičkih elemenata mora biti optimizovano i zauzeće memorije i procesora potrebno za prikaz grafičke korisničke sprege mora biti minimalno.

Takve aplikacije dopunjuju funkcionalnosti TV uređaja i danas postaju sve zastupljenije. Naravno, mogu se koristiti u kombinaciji sa TV uređajima koji poseduju odgovarajuće multimedijalne mogućnosti i sprege.

## 2. Teorijske osnove

Glavni razlog sve češće upotrebe prenosivih uređaja poput pametnih telefona, tablet, kao i iOS pametnih uređaja kao sekundarnih ekrana u prikazu televizije jeste poboljšanje karakteristika tih uređaja, kao i sve veća popularizacija digitalne televizije.

### 2.1 Digitalna televizija

Televizijski uređaj je najrasprostranjeniji i najpopularniji kućni uređaj koji zabavlja, informiše i obrazuje ljude. Prva upotreba reči televizija zabeležena je 1900-te godine u Parizu, dok je prvi digitalni sistem visoke rezolucije prikazan 1990. godine. Prva demonstracija digitalnog televizijskog prenosa održana je 1995. godine.

Digitalna televizija predstavlja prenos slike i zvuka sa dodatnim informacijama u digitalnom formatu. Digitalni prenos obezbeđuje bolji kvalitet, a teoretski signal se može preneti na neograničenu udaljenost. Slika i zvuk se prilikom snimanja pretvaraju u digitalnu formu postupcima odmeravanja, kvantovanja, kodovanja i u takvom obliku se prenose kroz medijum prenosa, odnosno kanal. Takođe digitalni prenos štedi frekvencijski opseg potreban za prenos digitalnog sadržaja. Raznim tehnikama kompresije i multipleksiranja, digitalni prenos zauzima oko 10 puta manji frekvencijski opseg u odnosu na običnu analognu televiziju za istu količinu saobraćaja. [1]

STB (eng. *set-top box*) je uređaj koji služi za prijem i obradu digitalnih zemaljskih televizijskih i satelitskih signala u domaćinstvima. Osnovna uloga ovakvog uređaja je prijem digitalnog zemaljskog ili satelitskog televizijskog i radio signala iz odgovarajućeg mrežnog ili modulacionog signala i njegovo prosleđivanje na monitor ili TV (prikaz). Digitalan TV signal u prenosu je komprimovan odgovarajućim kodnim postupkom i to najčešće MPEG-2 ili MPEG-4.

STB zato mora u sebi da sadrži dekoder koji radi dekompresiju ovog signala i zatim na odgovarajući izlaz prosleđuje ka televizoru. Najčešći izlazni konektori su SCART, kompozitni analogni ulazi(eng. *chinch*) ili najnoviji HDMI(digitalni). Pored prijema i prikazivanja digitalnog TV signala STB poseduje mnoge druge mogućnosti kao što su:

- Programski vodič je mogućnost da pritiskom na posebno dugme daljinskog upravljača dobijete informacije o svim emisijama koje se emituju na TV kanalima kao i rasporedu emitovanja filmova, emisija, sporta i slično.
- Omiljeni kanal je opcija kojom možete da definišete nekoliko kanala koje najčešće gledate. Time se dobija da stiskom samo na jedno dugme daljinskog upravljača može da se prebaci na željeni kanal.
- Snimanje je opcija koju imaju skuplji STB. Najčešći način je snimanje na lokalni disk, ali neki provajderi ovu uslugu vrše preko centralizovanih server.
- Roditeljska kontrola je opcija koja služi roditeljima da kontrolišu sadržaj koji deca mogu da gledaju. Oni mogu da definisu šifru na pojedinim kanalima ili za emisije koje u svojoj oznaci imaju odgovarajuću starosnu grupu.

Činjenica da se u novije vreme STB uređaji zasnivaju na sve moćnijim procesorskim arhitekturama i Linux operativnom sistemu, dovodi do razvoja mnogih drugih mogućnosti. Čak se na pojedinim STB koristi i operativni sistem Android, što omogućava izvršavanje skoro svih aplikacija kao na mobilnim telefonima: igre, internet pretraživač, vremenska prognoza,...

Pojam pametnog televizora (eng. *SmartTV*) odnosi se na televizijske uređaje koji umesto tradicionalnog difuznog emitovanja TV sadržaja, imaju fokus ka interaktivnom sadržaju sa interneta, sadržaju prikazanom preko TV slike, kao i pristupu sadržaju na zahtev korisnika. Pametan TV omogućava pristup sadržaju generisanom od strane korisnika koji se može nalaziti na lokalnom disku ili disku na udaljenom serveru. Omogućen je pristup interaktivnim servisima i internet aplikacijama kao što je YouTube.

## 2.2 iOS Platforma

iOS (prethodno iPhone OS) jest operativni sistem razvijen od strane Apple Inc. kompanije 2007 godine. Na samom početku, kada je prvi put javno izložen, iOS je bio namenjen samo za iPhone, međutim Apple je tokom godina razvio podršku i za ostale svoje uređaje: iPod Touch (septembra 2007), iPad (januara 2010), iPad Mini (novembar 2012) i AppleTV druge generacije (septembar 2010). Za razliku od Windows Phone i Androida Apple ne izdaje licencu za instalaciju iOS operativnog sistema na uređajima proizvedenim van ove kompanije.

iOS kontroliše rad telefonske opreme i primenjuje neophodne tehnologije, potrebne za rad osnovnih aplikacija. Sistem takođe sadrži i nekoliko osnovnih aplikacija poput telefona,

elektronske pošte i internet pretraživača (Safari), muzičkog plejera, video plejera i centralnog kontrolera (Spring Board) šta zadovoljava osnovne korisničke potrebe.

Poput Androida čije jezgro predstavlja Linux operativni sistem, iOS nije kompletan operativni sistem, nego implementira modifikovano jezgro Darwin OS-a (verzija koja se pojavila na prvom iOS operativnom sistemu jeste Darwin 9.0.0d1).

Programski jezicikoj se koriste za razvijanje aplikacija na iOS operativnom sistemusuobjektni C(eng. *Objective C*), C++ i C. Aplikacije moraju biti razvijene i generisane isključivo za iOS i 32-bit ili 64-bit ARM arhitekturu.

Nakon verzije 4.0, iOS podržava rad više programa istovremeno, dok je pre verzije 4.0 ovo bilo omogućeno samo za sistemske aplikacije. Ovo je omogućilo bolje iskorišćenje procesorskih resursa koje iOS uređaji poseduju.

Arhitektura iOS operativnog sistema sastoji se iz pet osnovnih slojeva:

- **Jezgro i upravljački programi (eng. Kernel and Device Drivers)** – omogućava komunikaciju na nivou fizičke arhitekture, umrežavanje i upravljanje procesima i memorijom.
- **Jezgro OS (eng. Core OS)** – uključuje elemente potrebne za zaštitu aplikacija. Sadrži nekoliko elementarnih tehnologija: Ubrzanje, Disk posrednik, OpenCL, Otvoren direktorijum i Podešavanje sistema. *Ubrzanje* obuhvata sve tehnologije ubrzanja koda gde su prisutne kompleksne operacije i obrada slike. *Disk posrednik* rukuje svim operacijama potrebnim za povezivanje uređaja sa okolinom. *OpenCL*(Open Computing Language) vrši raspodelu zadataka u zavisnosti od broja procesorskih jezgara dostupnih na uređaju, kao i od trenutne opterećenosti procesora. *Otvoren direktorijum* je komponenta koja služi za dobavljanje baza podataka koje se nalaze lokalno ili u mreži. *Podešavanje sistemase* bavi pretragom dostupnih mreža, proverava da li su one dostupne pre povezivanja, i vrši konfigurisanje mreže.
- **Ključne usluge (Core Services)** – obezbeđuje suštinske usluge potrebne aplikaciji, ali nema dodir sa njenim korisničkim okruženjem. Ove usluge direktno su zavisne od tehnologija navedenih u prethodna dva sloja. Neke od usluga koje ovaj sloj uključuje jesu: usluge zaštite, informacije o imeniku, prepoznavanje govora, identiteta i lokacije, čuvanje podataka od važnosti...
- **Media (Media Services)** – realizuje 2D i 3D grafiku, animacije, efekte na slikama, kao i sve audio i video funkcionalnosti korištene od strane aplikacija. iOS podržava više do 100 multimedijalnih formata, koji pokrivaju razne audio i video formate, formate slika kao i formate za reprodukciju u realnom

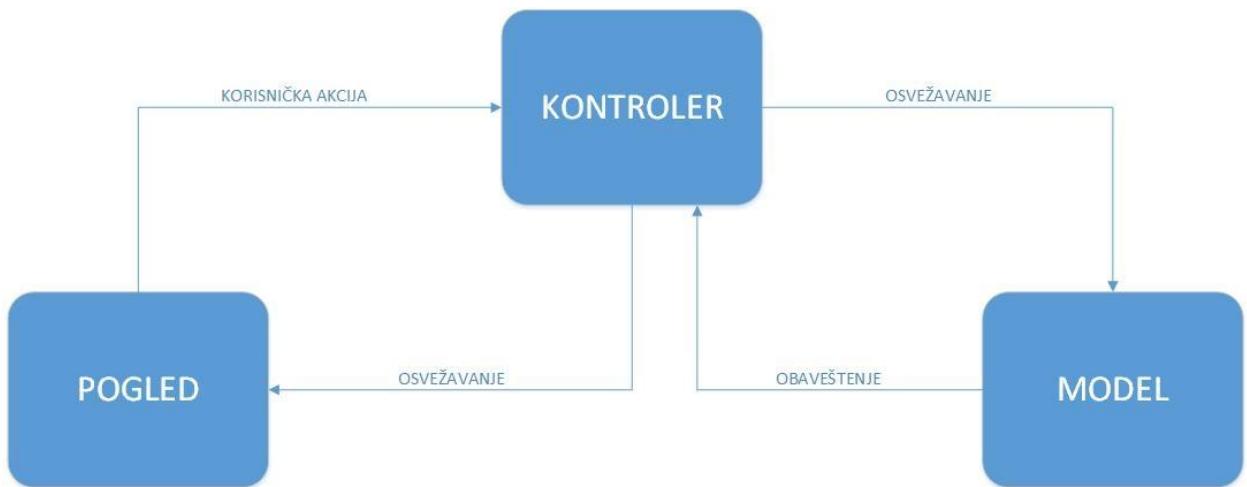
vremenu. Takođe ovaj sloj rukovodi podrškom za ispis teksta (font, veličina, upis i čuvanje) i bojama.

- **Cocoa Application** – sloj direktno odgovoran za prikaz svih komponenti aplikacije kao i za njen odziv na korisničku akciju. Neki od elemenata ovog sloja jesu: centar za upozorenja, centar za igre, automatsko i iznenadno gašenje aplikacija, pauziranje, automatsko čuvanje stanja, prikaz preko celog ekrana...

Prilikom razvoja aplikacije programer ima pristup samo poslednjem sloju operativnog sistema, a to je Cocoa Application. Ostali slojevi programeru nisu dostupni i pod zaštitom su operativnog sistema.

## 2.3 Grafička korisnička sprega uređaja zasnovanih na iOS platformi (Cocoa Touch)

Cocoa Touch je grafička korisnička sprega koja služi za kao osnova za implementaciju korisničkih aplikacija na iOS platformama (iPhone, iPod Touch i iPad). Dizajnirana je od strane kompanije Apple. Ona obezbeđuje komunikaciju između hardverskih i softverskih komponenti uređaja. Arhitektura programa bazirana je na Model-Pogled-Kontroler obrascu (eng. *Model-View-Controller – MVC*).



Slika 1 GUI Arhitektura iOS platforme

Cocoa Touch obezbeđuje strukturu i alate za realizaciju kontrolera koji obrađuje ulazne događaje poput pritiska na tastere i dodira ekrana, i pogled (eng. *View*) koji obrađuje i prikazuje grafičke informacije na ekranu.

**Model** predstavlja stanje modula. Stanje mogu menjati neke od operacija modela.

**Pogled** obezbeđuje korisniku interfejs (ekransku formu) pomoću koje će korisnik da unosi podatke i poziva odgovarajuće operacije koje treba da se izvrše nad modelom. Pogled prikazuje korisniku stanje modela.

**Kontroler** osluškuje i prihvata zahtev od korisnika za izvršenje operacije. Nakon toga poziva operaciju koja je definisana u modelu. Ukoliko model promeni stanje, kontroler obaveštava pogled da je promenjeno stanje.

MVC obrazac mora da zadovolji sledeća pravila:

- Kontroler prati događaje koji se izvršavaju nad pogledom i na odgovarajući način reaguje na njih.
- Kontroler je posrednik koji prima zahtev od pogleda i preusmerava ga do modela.
- Pogled mora da reflektuje stanje modela. Svaki put kada se promeni stanje modela, pogled treba da bude obavešten o tome.
- Model ne mora da zna ko je kontroler ili pogled.

*Connelly Barnes* je rekao: „Najlakši način da razumete MVC je: model je podatak, pogled je ekranska forma, a kontroler je lepak između modela i pogleda.“. [3]

Grafička sprega u iOS operativnom sistemu je definisana vizuelnim komponentama:

- 2D grafikom
- Vizuelnim kontrolama (dugme, slika, prozor za pomeranje)
- Privremenim dijalog prozorima
- Barovima za navigaciju
- Komponentama sa sadržajem (kolekcija, tabela)

Grafička korisnička sprega se sastoji od više formi prozora i ekrana, a svaki prozor se kontroliše pomoću Kontrolera Prozora – KP (eng. *ViewController*) koji ima kompleksni životni vek. Svim objektima u UI koji se nalaze u KP može se pristupiti tek kada je KP napravljen, a postoje sve dok je KP živ. Kontroleri prozora su međusobno povezani putem Kontrolera Navigacije – NK (eng *NavigationController*). Na ovaj način se sve odvija u jednoprocesnom režimu rada. Napredak iOS programske podrške predstavlja tehnologije za razvoj grafičke korisničke sprege sa kojima je moguće podeliti jednu veliku celinu u više zasebnih elemenata. To znači da svaki gradivni element se može izvršavati u odvojenoj programskoj niti koji je i dalje vezan za glavni proces (eng. *multithreading*).

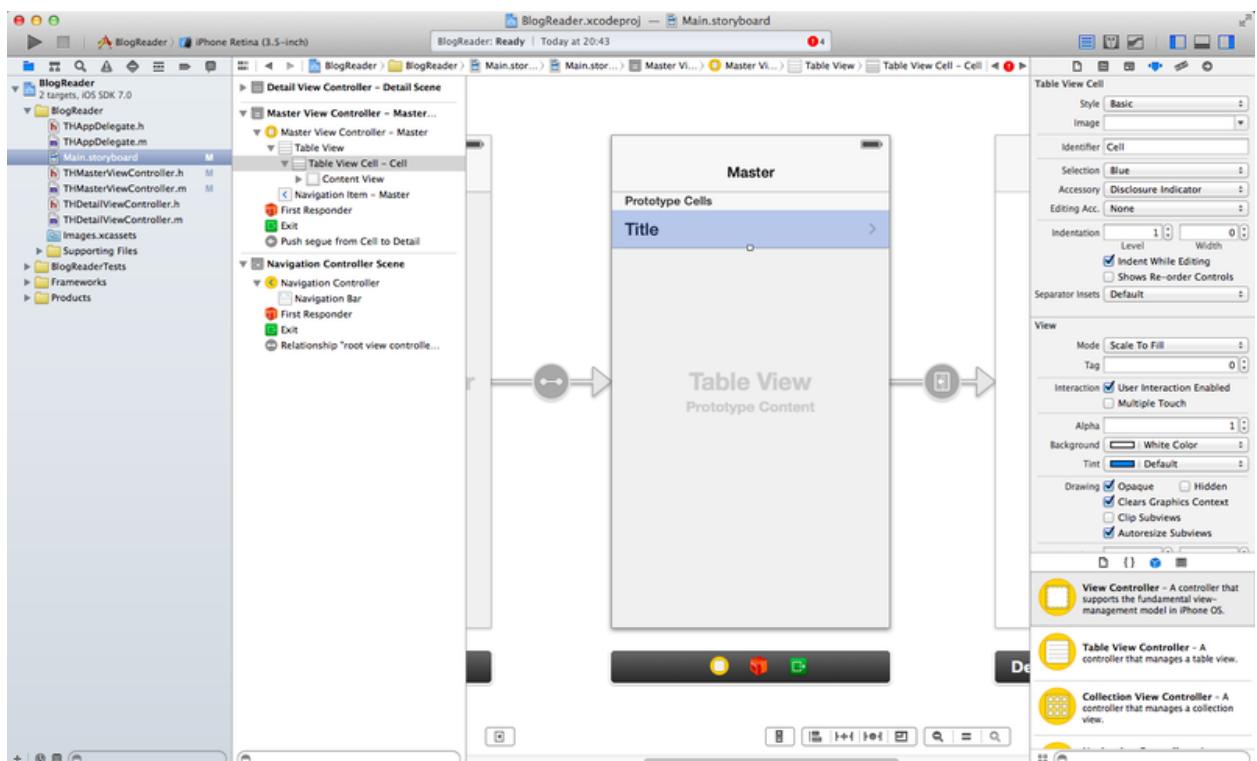
## 2.4 Xcode razvojno okruženje

Xcode je integrisano razvojno okruženje (eng. *Integrated Development Environment* - IDE) kompanije Apple, koje je namenjeno razvoju isključivo programa za OS X i iOS operativne sisteme. Program je besplatan i dostupan je preko Mac App Store aplikacije koja se nalazi na

svim Macintosh uređajima kompanije Apple. Poslednja stabilna verzija ove aplikacije jeste 5.1 i podržana je od strane sledećih verzija operativnih sistema: Mac OS X Lion I Mac OS X Mountain Lion i Mac OS X Maverick.

Izgled okruženja se može videti na Slika 2.

Sa leve strane okruženja nalazi se više prozora od kojih su najbitniji: pogled na projekat, koji na pregledan način prikazuje izvorne, projektne i resursne datoteke, statistika zauzetosti procesorskih resursa, spisak procesa unutar pokrenute aplikacije. U sredini Xcode-a nalazi se editor koda. Sa desne strane nalazi se Inspektor. Inspektor je alat koji u grafičkom režimu rada editora (Storyboard), korisniku prikazuje informacije o odabranom elementu i pruža mogućnost dodatnog podešavanja parametara vezanih za statičke objekte grafičke korisničke sprege. Takođe Inspektor pruža korisniku dodatne informacije o funkciji koja je trenutno obeležena u editoru koda. U donjem desnom uglu Xcode-a prilikom izmene grafičke korisničke sprege pojavljuje se prozor sa standardnim iOS komponentama, koje se mogu dodati na ekran uređaja jednostavnim prevlačenjem.



Slika 2 Xcode razvojno okruženje

## 2.5 Način realizacije grafičke korisničke sprege na iOS platformi

U početnim verzijama iOS platforme programiranje se zasnivalo jedino na programskom generisanju izgleda aplikacije, međutim sada je to olakšano dostupnim alatima za vizualizaciju uključenim u Xcode razvojno okruženje koji pružaju veliku podršku u stvaranju aplikacije. Kod

izrade iOS aplikacija postoje dva načina realizacije grafičke korisničke sprege: deklarativan i proceduralan.

Deklarativna realizacija grafičke korisničke sprege (statička korisnička sprega) koristi se za opis statičkih korisničkih elemenata koji se jednom postavljaju na ekran bez promene tokom rada aplikacije. Upotreba statičke grafičke korisničke sprege je najzastupljenija. Bazira se na grafičkom povezivanju elemenata ekrana korišćenjem iOS komponente zvane Storyboard ili programiranjem pomoću Objektnog-C jezikom. Storyboard komponenta automatski generiše XML kod na osnovu onoga što je grafički podešeno od strane programera.

Proceduralna realizacija opisuje ono što deklarativna ne može, a to je dinamički deo korisničke sprege, pokretljivi grafički elementi koji se dodaju i brišu po zahtevu korisnika aplikacije. Koristi se Objektni-C jezik za opisivanje korisničke sprege. Kod ovog načina implementacije svaki objekat korisničke sprege mora biti deklarisan i definisan u kodu kako bi mogao da se koristi, odnosno da se prikaze na ekranu uređaja.

Sama ideja spajanja ove dve sprege u jednu aplikaciju, odnosno korišćenje u prikazu jednog prozora na ekranu doprinelo je razvoju dinamičnih iOS aplikacija. Deklarativnu spregu i njene mogućnosti treba koristiti koliko je god moguće kako bi se dobilo na brzini prikazivanja, a dinamičku da bi se obradili zahtevi korisnika u realnom vremenu.

## 2.6 Princip rada dodatnog ekrana u prikazu televizije

Dnevna soba je prostor u kojem porodica provodi najveći period svog slobodnog vremena. Televizija često predstavlja izvor zabave, prikupljanja informacija, razlog započinjanja razgovora ili jednostavno razlog okupljanja čitave porodice. Nagli porast upotrebe računarskih uređaja (pametnih telefona, ručnih računara, laptopova) uticao je na promenu tradicionalnih porodičnih navika prilikom gledanja televizije. Ukućani sve češće koriste laptop kako bi pretraživali internet, koristili elektronsku poštu ili kako bi obavljali kupovinu preko interneta, za vreme gledanja televizije [6]. Intenzivno korišćenje dodatnih uređaja tokom gledanja TV-a otvorilo je mogućnosti za korišćenje sekundarnih ekrana. Sekundarni ekran je termin koji se odnosi na dodatni električni uređaj kao što je pametni telefon ili tablet, koji omogućava televizijskim gledaocima pristup sadržaju i funkcijama TV-a. Uporedo sa prikazom televizijskog programa, na sekundarnom ekranu postoji mogućnost prikaza dodatnih podataka, bez ometanja ostalih korisnika koji prate televizijski program preko TV ekrana. Aplikacija sa sekundarnim ekranom može da integriše interaktivne TV (*eng. iTV*) usluge, kao i da se orijentiše ka korišćenju televizijskog prikaza i interakciji sa korisnikom.

U istraživanjima [4,6,7] sugerisano je da daljinski upravljači sa većim mogućnostima pomažu korisnicima da postanu aktivniji u iTV okruženju u odnosu na raniji pristup samo

gledanja televizijskog programa. U [7] se može pronaći da su daljinski upravljači često smatrani kao slabo zastupljeni; šta više, rezultati ukazuju da bi grafičko prikazivanje pomoću daljinskog upravljača trebalo da bude znatno više korišćeno u budućnosti. U [4] je izведен zaključak da bi trebalo uvesti prefinjeniji oblik ulaza i kontrole iTV-a kako bi on dostigao svoje potpune mogućnosti. U [6] se navodi da je korisnije imati pristup različitim servisima na odvojenim ekranima, u odnosu na samo jedan pretrpan prikazni ekran. Oslanjajući se na ove rezultate, iako je malo verovatno da će tradicionalne navike gledanja nestati, trebalo bi težiti ka odgovarajućim alatima za aktivno povezivanje korisnika sa gledanim sadržajem.

U [8] je predstavljeno rukovanje DTV sadržajem. Programski sadržaj, podaci elektronskog programskog vodiča (eng. *Electronic Program Guide – EPG*), dodatni podaci o gledanom programu na glavnom ekranu, kontrola video snimanja sadržaja sa glavnog ekrana, neke su od stvari koje se mogu prikazati na sekundarnom ekranu. U [5] je objašnjen mozaik sačinjen od paralelnog prikaza nekoliko kanala, uz mogućnost promene trenutnih kanala na željeni, ali na TV prijemniku.

Korisnik aplikacije ne zavisi od sadržaja prikazanog na glavnom ekranu, već može da menja izgled sadržaja u svom mozaiku iz aplikacije sa sekundarnim ekranom, gleda uživo televizijski program, pa shodno tome sekundarni ekran za korisnika postaje glavni. Takođe korisnik ima pristup EPG-u ili bilo kojoj drugoj kontroli, bez povezanosti sa glavnim ekranom i njegovim pozadinskim procesima. Korisnik može upravljati glavnim ekranom pomoću sekundarnog ekrana, koji u ovom slučaju implementira i proširuje funkcije daljinskog upravljača. Dodatna funkcionalnost aplikacije sa sekundarnim ekranom jeste mogućnost prikaza multimedijalnog sadržaja sa servera za prikazivanje video sadržaja u realnom vremenu.

## 3. Koncept rešenja

Poboljšanje performansi lokalnog TV uređajakao i sve češća prisutnost digitalne televizije, glavni je razlog sve češće upotrebe ručnih uređaja poput pametnih telefona i tableta za potrebe prikaza TV sadržaja na sekundarnom ekranu. Poglavlje prikazuje idejno rešenje arhitekture jednog rešenja sistema.

### 3.1 Zahtevi aplikacije

Analizom zahteva identifikovano je 7 celina kako bi aplikacija zadovoljila željene uslove prikaza celokupnog dostupnog televizijskog sadržaja:

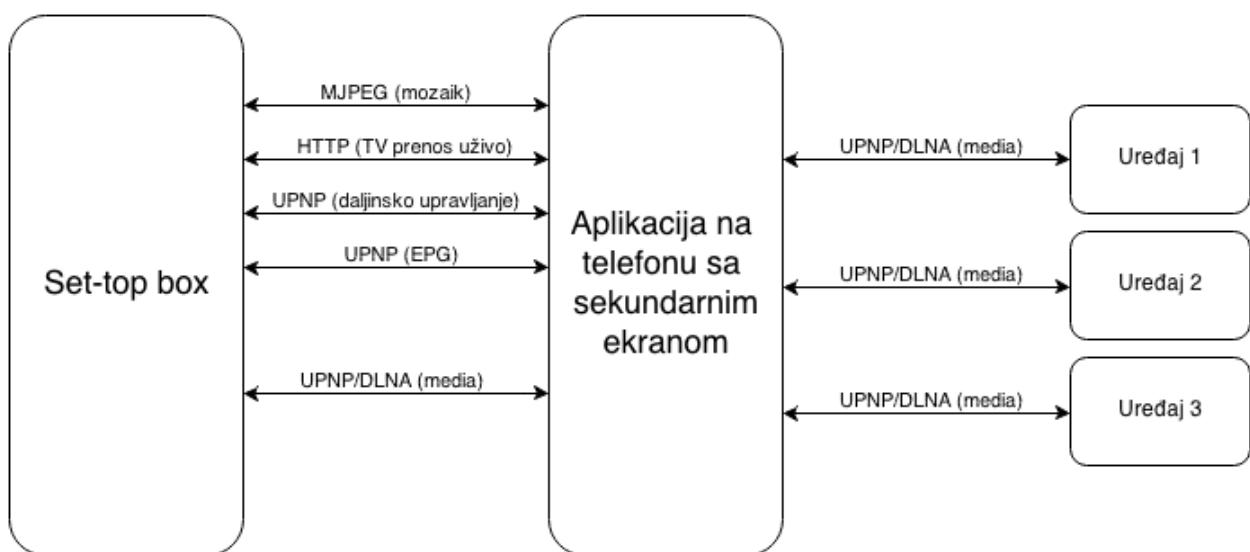
1. Potrebno je da se pretraga poslužitelja sa mogućnošću prenosa TV sadržaja preko mreže uređajima sa dodatnim ekranom vrši putem SSDP protokola. Komunikacija aplikacije sa odabranim STB mora se vršiti putem UPNP protokola.
2. Potrebno je da aplikacija prikaže trenutno gledan TV kanal na uređaju sa sekundarnim ekranom, da ima mogućnost promene kanala, kao i na TV ekranu. Promenu kanala potrebno je vršiti lakim pristupom funkcionalnostima daljinske kontrole implementirane unutar aplikacije, ali i putem liste za odabir kanala i klizanjem prsta po ekranu sekundarnog uređaja. Sistem pruža video u MPEG2 video formatu, što nije podržano osnovnom komponentom. Potrebno je integrisati postojeći video plejer koji ovaj format podržava. U istraživanju [9] pokazano je da je za MPEG2 video format VLC komponenta prikazala najbolje performanse i da se sa njom troši najmanje procesorskih resursa. Potrebno je takođe prikazati osnovne informacije o trenutno gledanom kanalu, uključujući podatke o svim dostupnim emisijama gledanog kanala (EPG). Gledanje kanala treba omogućiti i u modu preko celog ekrana, gde podaci o kanalu moraju biti sakriveni. Uživo prenos TV sadržaja od

STB ka aplikaciji potrebno je realizovati putem HTTP protokola slanjem zahteva za dobijanje video sadržaja željenog kanala.

3. Potrebno je realizovati Elektronski programski vodič po uzoru na onaj sa glavnog TV ekrana.Potrebno ga je prikazati na sekundarnom ekranu kao odvojen prikaz u odnosu na sadržaj koji se trenutno prikazuje na glavnom TV ekranu. Treba omogućiti detaljan prikaz informacija vezanih za jednu emisiju unutar odabranog kanala. Informacije elektronskog programskog vodiča potencijalno mogu da sadrže veliku količinu podataka (npr. 50 kanala, 80 emisija za svaki kanal, dodatne informacije za svaku emisiju). Iz tog razlogaobijanje ovih podataka mora biti potpuno odvojeno od korisničke sprege.Dobavljanje informacija o elektronskom programskom vodiču mora se odvijati nezavisno od glavne niti aplikacije, kako korisnik ne bi morao čekati na završetak prikupljanja informacija. Potrebno je periodično osvežavati podatke koji su predstavljeni korisniku, kako bi se obavestio o promenama koje se mogu desiti za vreme korišćenja aplikacije.
4. Prikazati prozor za paralelan prikaz 4 kanala. Svaki od 4 prikazanih kanala se može podešavati. Potrebno je podržati prelazak na uživo praćenjeodabranog kanala iz ovog prikaza.
5. Pored simulacije daljinskog upravljača, potrebno je realizovati pokazivačku pločicu i simulaciju tastature sa svim funkcionalnostima koje TV prijemnik podržava. Slanje komandi sa ovih komponenti treba slati preko UPNP protokola.
6. Većina STBima podršku za prikaz multimedijalnogsadržaja. Iz tog razloga potrebno je realizovati modul koji će komunicirati sa STB i koji će moći da pušta sadržaj sa njega na sekundarnom ekranu posredstvom DLNA protokola. Takođe iz razloga što su oba uređaja priključena na istu mrežu, potrebno je ovu funkcionalnost napraviti za sve uređaje koji se nalaze na istoj lokalnoj mreži kao i uređaj sa sekundarnim ekranom. Pretraga svih kompatibilnih multimedijalnihuređaja unutar mreže mora se vršiti preko DLNA protokola.
7. Aplikacija treba imati prozor sa podešavanjima parametara kao što je osetljivost pokazivačke pločice na dodir (brzina kretanja po glavnom ekranu). Potrebno je implementirati mogućnost pretrage mreže u potrazi za listom poslužitelja koji su u stanju da šalju video sadržaj uređaju sa sekundarnim ekranom.

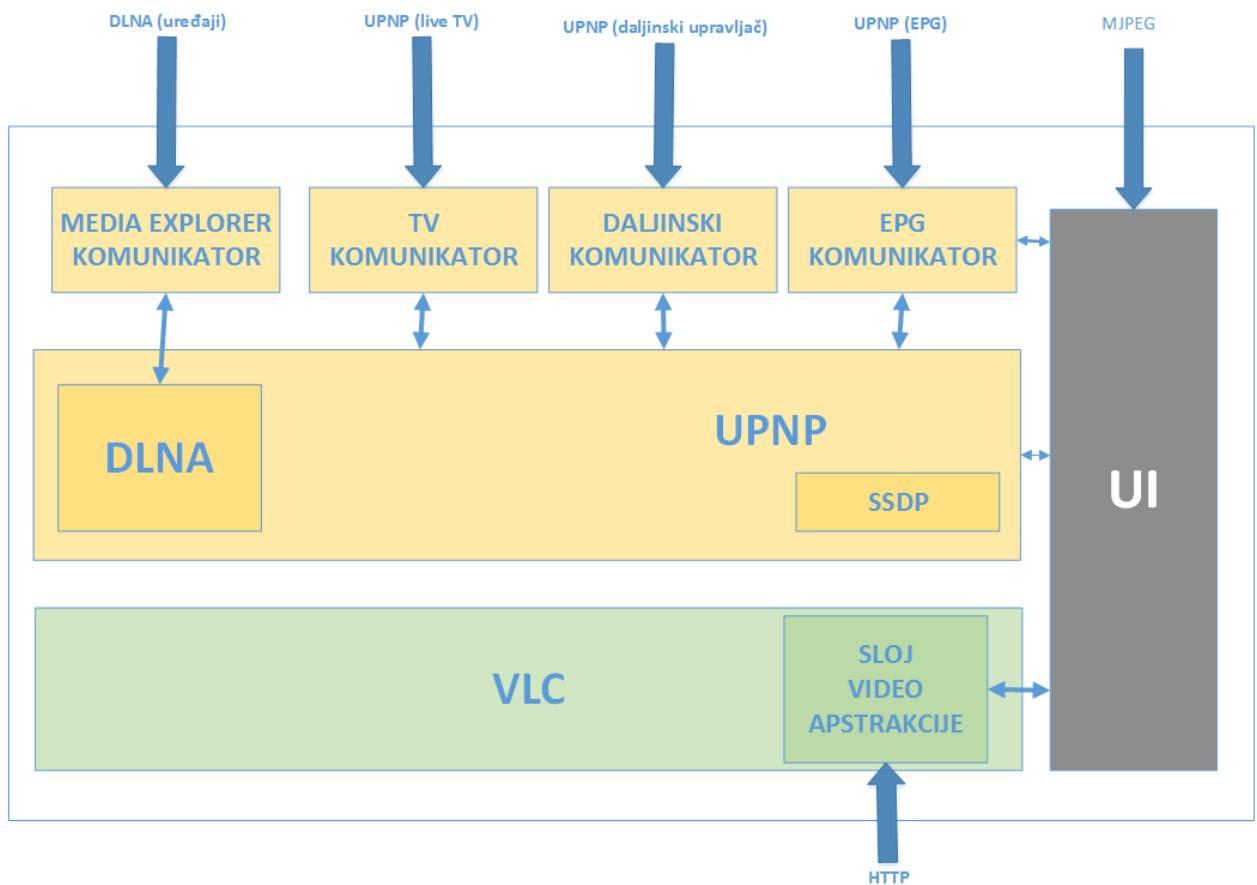
## 3.2 Interakcija aplikacije sa sekundarnim ekranom sa ostalim uređajima

Klijentska aplikacija dobavlja TV usluge različitim kanalima (Slika 3). Prenos programskog sadržaja koji se prikazuje na sekundarnom TV ekranu iOS uređaja, obavlja se putem HTTP protokola gde poslužitelj aplikaciji kontinualno šalje podatke za željeni kanal putem MPEG2 transportnog toka, MPEG2 kodovanjem sadržaja. Mozaični prikaz digitalnog sadržaja se realizuje upotrebom MJPEG toka podataka gde se korisniku pokazuje niz slika, a ne aktivna slika kao u sekundarnom TV ekranu. Funkcionalnost mozaičnog prikaza podržava do 4 video toka istovremeno. Servisi daljinskog upravljanja i EPG sadržaj šalju se putem UPNP zahteva poslužitelju. Za razliku od ovih zahteva DLNA je korišćen za prikupljanje informacija o multimedijalnom sadržaju posluživača, kao i listi poslužitelja. Funkcionalnosti pretrage medijalnih poslužioca se odnosi na sve dostupne poslužitelje u mreži.



Slika 3 Komunikacioni protokoli korišteni za realizaciju aplikacije sa dodatnim ekranom

### 3.3 Arhitektura najvišeg nivoa aplikacije



Slika 4 Blok dijagram najvišeg nivoa aplikacije aplikacije

Na Slika 4 mogu se videti 3 osnovne celine unutar najvišeg gradivnog sloja aplikacije:

- Blokovi označeni narandžastom bojom koji posredstvom UPNP biblioteke komuniciraju sa STB uređajem, kao i ostalim uređajima unutar mreže koji imaju mogućnost deljenja multimedijalnog sadržaja. SSDP blok služi za pretragu uređaja unutar mreže. Pretraga se može vršiti u dva pravca: dobavljanje uređaja koji mogu deliti televizijski sadržaj i dobavljanje uređaja koji mogu deliti multimedijalni sadržaj. DLNA Blok služi za komunikaciju sa uređajima unutar mreže koji imaju sposobnost deljenja multimedijalnog sadržaja. Komunikatorski blokovi predstavljaju posebne niti koje nezavisno od grafičke korisničke sprege aplikacije šalju i dobijaju poruke od strane STB-a. Nakon dobijene komande od strane STB-a, komunikatori obrađuju primljenu poruku i prosleđuju je UI bloku.
- UI blok predstavlja grafičku korisničku spregu aplikacije i odgovoran je za prikaz sadržaja na dodatnom ekranu. Ovaj blok prikazuje statičke elemente grafičke korisničke sprege, ali takođe služi i za prikaz elemenata dobijenih od strane

komunikatora. IsCRTavanje elemenata unutar ovog bloka vrši se isključivo u glavnoj niti aplikacije.

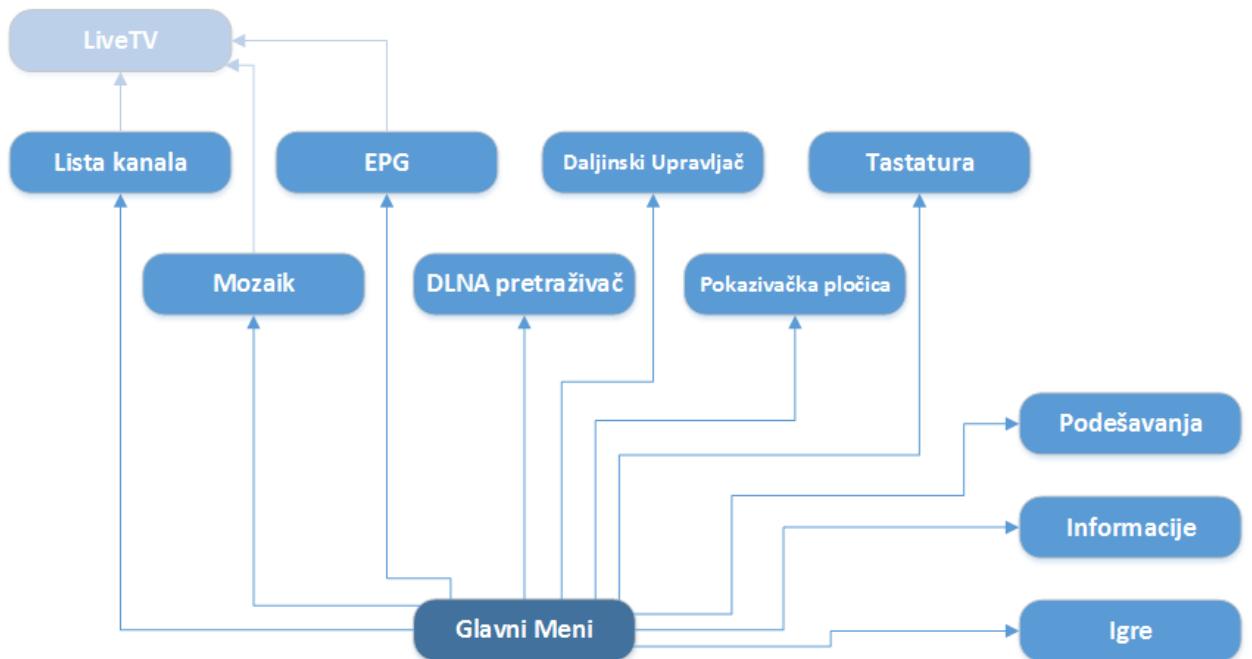
- Blokovi označeni zelenom bojom predstavljaju video okruženje neophodno za realizaciju prikaza video sadržaja u MPEG-2 video formatu. Ovaj video format nije podržan od strane ugrađene komponente za reprodukciju video sadržaja na dodatnom ekranu, pa se iz tog razloga koristi VLC biblioteka. Na zahtev UI bloka, blok video apstrakcije pokreće HTTP konekciju ka STB-u, nakon čega od STB dolazi tok podataka koji ovaj blok posredstvom VLC biblioteke obrađuje. Obrađen sadržaj šalje se nazad ka UI bloku, nakon čega se video sadržaj prikazuje na ekranu.

## 4. Programsко rešenje

U ovom poglavlju su prikazani detalji realizacije rešenja. Programsko rešenje sastoji se od navigacionog stabla koje ispunjava ceo ekran i realizovano je u programskom jeziku Objective-C uz iOS programsku podršku. Poglavlje se usredsređuje na televizijski prikaz digitalnog sadržaja u realnom vremenu i karakteristične usluge digitalne televizije. Sistem se zasniva na pametnom TV prijemniku koji u istom trenutku može da opsluži do 8 klijenata sa različitim sadržajem. Svaki klijent može da prati željeni televizijski sadržaj nezavisno od samog TV prijemnika.

### 4.1 Elementi navigacionog stabla

Zbog karakteristične funkcionalnosti iOS uređaja koji imaju samo jedno dugme za izlaz iz aplikacije, mora se realizovati kretanje kroz kontrolere prozora primenom navigacionog kontrolera, koji u svaki KP dodaje bar sa navigacijom na vrhu ekrana. Bar sa navigacijom ima povratni mehanizam u vidu dugmeta za povratak na prethodni KP. Prikaz kompletног stabla aplikacije, kao i navigacioni putevi prikazani su na Slika 5.



Slika 5 Prikaz stabla aplikacije sa navigacionim putevima

Zaobljen pravougaonik tamno plave boje predstavlja početni KP. Plavim zaobljenim pravougaonicima predstavljene su sve opcije glavnog menija kojima se može pristupiti, dok se svetlo plavom (ekranu koji uživo reproducuje sadržaj želenog TV kanala) može pristupiti isključivo iz liste kanala, mozaika ili EPG-a.

## 4.2 Moduli korisničke spregе

U ovom poglavlju su opisani programski moduli, kao i njihove funkcije potrebne za implementaciju ovog rešenja. Svaki moduo predstavlja jedan od elemenata sa slike iz prethodnog poglavlja.

### 4.2.1 MainMenuViewController

Ovaj modul realizuje grafičku korisničku spregu glavnog menija (Slika 6).



Slika 6 Prikaz glavnog menija

Sadrži 10 opcija: LiveTV, Mozaic, EPG, Media Explorer, Info, Settings, Remote controller, Keyboard, Touchpad, Games. Izborom opcija prelazi se na naredni KP unutar stabla aplikacije. Ako internet konekcija nije obezbeđena pre ulaska u aplikaciju, prilikom pritiska na bilo koje dugme, osim Settings i Info dugmeta, korisniku se prezentuje poruka o grešci sa internet konekcijom. Dugme Games u ovoj verziji aplikacije nije aktivno, a njegova realizacija planirana je u kasnijoj fazi realizacije aplikacije.

Pritiskom Info dugmeta dobija se dijalog sa pojašnjenjem ostalih opcija glavnog menija. Umesto standardnog kontrolera prozora kao što je slučaj za ostalu dugmad, za prikaz poruke koristi se iskačući prozor.

Otkrivanje STB uređaja sa podrškom za prenos televizijskog sadržaja vrši se pomoću SSDP protokola. Otkriveni uređaji se sastavljaju u listu svih dostupnih uređaja. Ova lista se kasnije koristi u SettingsViewController modulu za prikaz liste poslužitelja. Postoji proverava se da li poslednji korišten poslužitelj postoji u ovoj listi i ako postoji omogućava se pristup opcijama glavnog menija. Ako poslužitelj ne postoji, iskačući prozor sa informacijom da se poslužitelj mora prvo odabrati u Settings delu se pojavljuje. SSDP komunikacija se pokreće na samom pokretanju programa i globalno je dostupna tokom rada aplikacije.

#### 4.2.2 ChannelListViewController

Pritiskom dugmeta LiveTV iz glavnog menija korisnik se usmerava na modul koji prikazuje listu kanala dobijenih UPNP komunikacijom sa poslužiocem. Odabirom željenog kanala iz liste, otvara se dijalog sa sledećim opcijama:

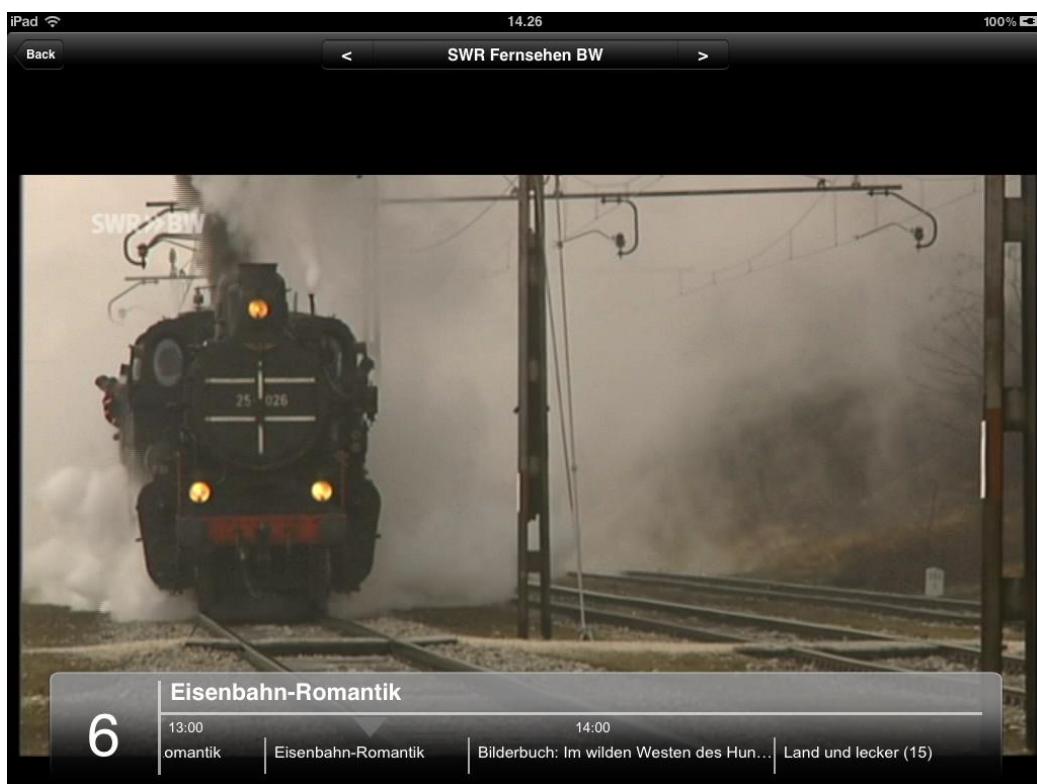
- postavljanje željenog kanala u mozaični pregled kanala
- promena kanala na glavnom TV ekranu na odabrani kanal
- mogućnost uživo pregleda odabranog kanala na sekundarnom ekranu.

Gradivne komponente ovog ViewController-a predstavljaju listu čiji su elementi slika i ime kanala, kao i dijalog koji predstavlja opcije za prikaz odabranog kanala iz liste.

#### 4.2.3 LiveTVViewController

Omogućava gledanje TV prenosa uživo na uređaju sa sekundarnim ekranom nezavisno od glavnog ekrana TV prijemnika. Kanal može biti različit ili kopija onog koji se trenutno prikazuje na glavnom TV ekranu. Za vreme gledanja prenosa na sekundarnom ekranu, korisnik može menjati kanale i pristupati dodatnim uslugama omogućenim od strane trenutno posmatranog kanalanezavisno od glavnog TV ekrana. Najčešće korišćene kontrole dostupne su korisniku u vidu ručnih gestova, kao što je klizanje prstom na levu ili desnu stranu, koje menja prethodni ili naredni kanal. Klizanjem prstapo ekranu na gore, aktivira se prepoznavanje gestova koji menja trenutno posmatrani kanal na glavnom TV ekranu u kanal koji je trenutno aktivan na sekundarnom ekranu. Preko trenutno puštanog video sadržaja korisnik u donjem delu ekrana ima prikaz informacija o trenutno gledanom kanalu kao što su broj kanala, ime trenutno gledane emisije, ali i pojednostavljen elektronski programski vodič sa programskim sadržajem samo trenutno gledanog kanala (Slika 7). Na navigacionom baru postoji izbor kanala pritiskom na dugme sa imenom trenutno puštanog kanala, gde se korisniku prikazuje lista svih dostupnih kanala, dok je pritiskom na dugmad pored omogućeno menjanje kanala na prethodni ili naredni.

Standardna video komponenta iOS operativnog sistema ne podržava puštanje video sadržaja kodovanog MPEG2 formatom, pa je iz tog razloga korišten VLC plejer koji podržava ovaj standard kodovanja. Integriran je korišćenjem VLCMediaPlayer strukture koja u sebi sadrži nekoliko biblioteka potrebnih za realizaciju ove komponente. Ova struktura u sebi sadrži i poseban video plejer koji sadržaj prikazuje na ekranu.

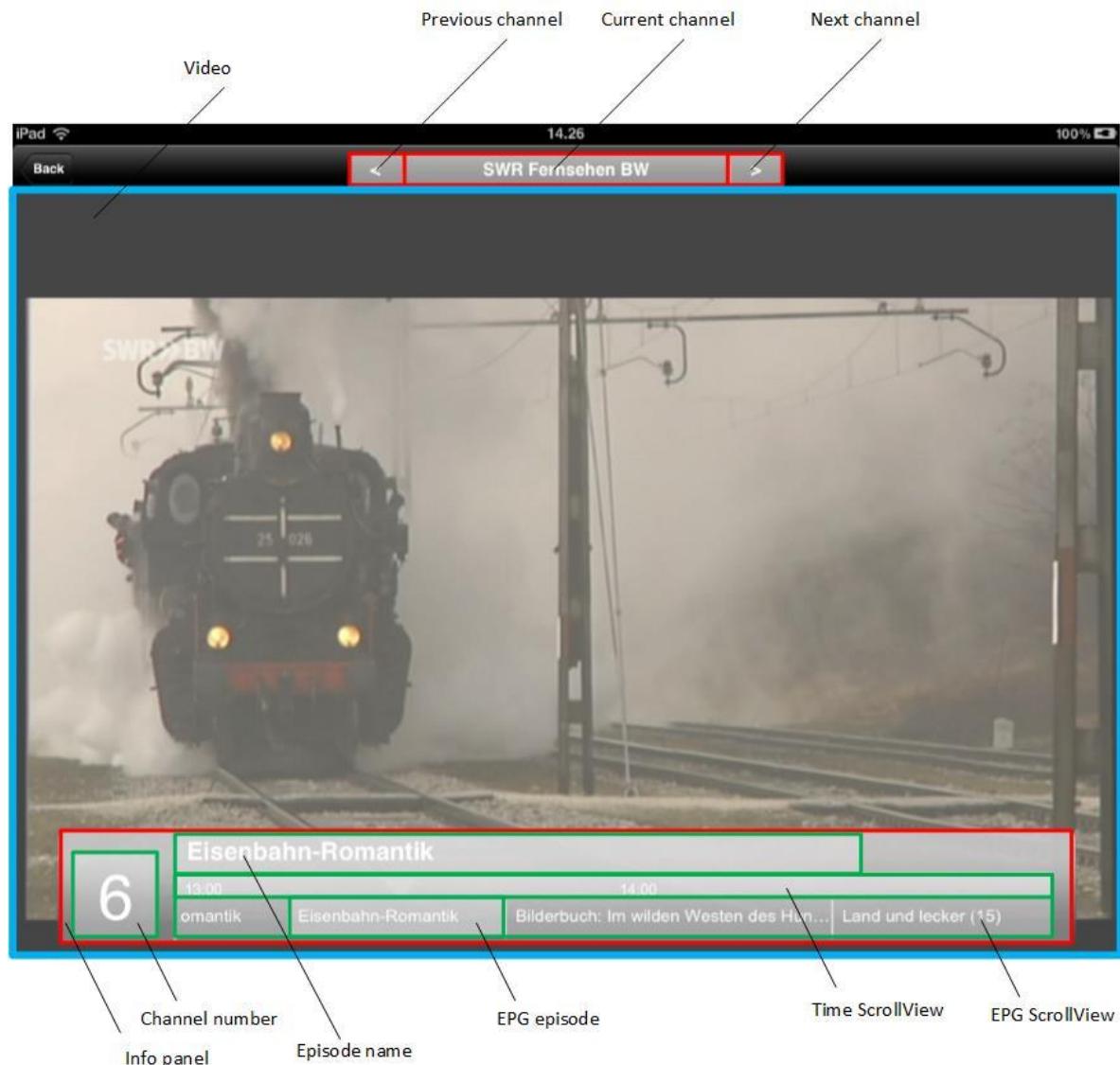


Slika 7 Prikaz TV prenosa na uređaju sa sekundarnim ekranom

Gradivne komponente prozora za prikaz video sadržaja uživo na uređaju sa dodatnim ekranom prikazane su na Slika 8.

- Komponenta **Video** jeste primerak VLC video plejera, korišćenog za prikaz video sadržaja dobijenog od strane server.
- **Previous Channel** dugme omogućava korisniku promenu kanala na sekundarnom ekranu sa trenutnog na predhodni kanal iz liste.
- **Next channel** dugme menja trenutni kanal na sledeći iz liste kanala, dok dugme **current channel** korisniku prikazuje kompletну listu kanala dostupnih na server u vidu iskačućeg prozora, a odabirom kanala iz liste menja se video koji se prikazuje u VLC komponenti za prikaz video sadržaja na ekranu.
- **Info panel** u donjem delu ekrana predstavlja komponentu koja se prikazuje van režima puštanja video sadržaja preko celog ekrana (eng. *full screen*), koja daje dodatne informacije o trenutno gledanom kanalu u vidu EPG sadržaja, rednog broja kanala iz liste (**Channel number**) i imena trenutno gledane emisije (**Episode name**).
- Sadržaj elektronskog programskog vodiča sačinjen je od 2 glavne komponente: **Time ScrollView** koji sadrži informacije o trenutnom vremenu u toku dana i **EPG ScrollView** koji sadrži informacije o epizodama koje se prikazuju tokom celog jednog dana. EPG ScrollView realizuje se kao namenska kontrola sa mogućnošću

pomeranja u desno za prikaz budućih i u levo za prikaz prošlih emisija. **EPG episode** komponenta predstavlja ime epizode i od ovih komponenti sačinjena je komponenta **EPG ScrollView**.



Slika 8 Prikaz gradivnih komponenti Elektronskog Programskog Vodiča

#### 4.2.4 EPGViewController

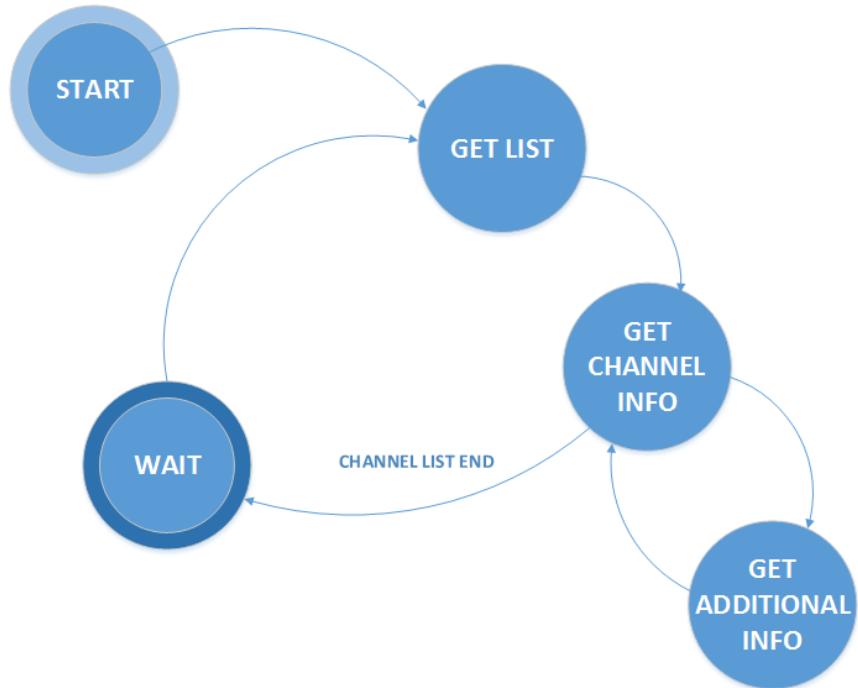
Modul je povezan sa prethodnim modulom. Grafička korisnička sprega modula prikazana je na Slika 9.



Slika 9 Prikaz elektronskog programskog vodiča

Kreirase svaki put kada se odabere opcija EPG u glavnom meniju ako je upaljena internet konekcija. Uloga modula jeste prikaz EPG sadržaja dobijenog od strane STB-a.

Dobavljanje EPG informacija se izvršava u zasebnoj niti nezavisno od glavne niti aplikacije. Osvežavanje grafičke korisničke sprege se izvršava tek kada se pribave informacije za jedan kompletan kanal. Osvežavanje se može izvršiti u dva slučaja: kada se dobavi novi kanal ili kada se promeni sadržaj kanala koji je već prikazan korisniku. Detaljan prikaz stanja u toku asinhronne komunikacije dat je na Slika 10.



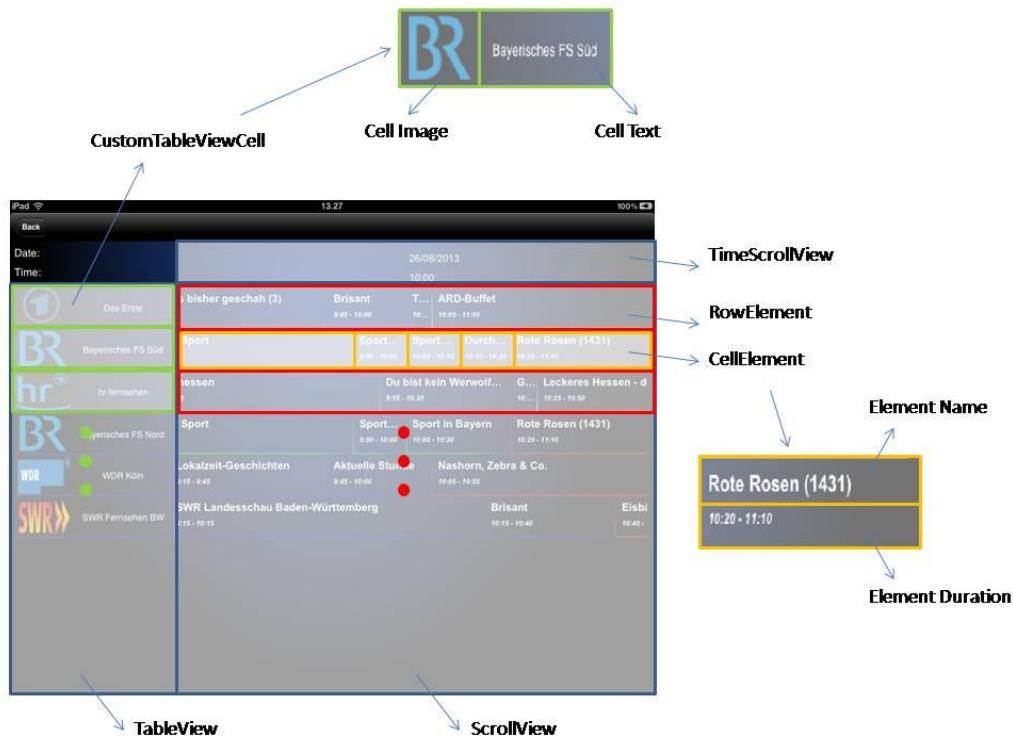
Slika 10 Prikaz asinhronne komunikacije prilikom dobijanja EPG sadržaja

Komunikacija se izvršava u 3 faze:

- slanje UPNP komande za dobijanje liste kanala (**get list**)
- slanje UPNP komandi za dobijanje EPG sadržaja za konkretni kanal (**get channel info**)
- slanje UPNP zahteva za proširenim informacijama o jednoj emisiji unutar EPG sadržaja jednog kanala (**get additional info**).

Slika kanala dobija se posebno, takođe slanjem prilagođene UPNP komande serveru koji se nalazi na STB. Nakon prikupljenog kompletног EPG sadržaja jednog kanala (slika kanala i sadržaj elektronskog programskog vodiča), nit obaveštava glavni viewController koji prikazuje novo dobijeni sadržaj korisniku (Slika 9, levi prozor). Obaveštavanje kontrolera vrši se preko ugradjenih iOS *delegate* funkcija. Ove funkcije se aktiviraju nakon nekog preciziranog događaja. U ovom slučaju događaj predstavlja završetak dobavljanja kompletних informacija jednog kanala što uključuje EPG sadržaj i sliku kanala.

Grafička korisnička sprege kreira se na osnovu dodatnih podataka o kanalima dobijenih od STB. Elementi grafičke korisničke sprege elektronskog programskog vodiča mogu se videti na Slika 11.



Slika 11 Prikaz gradivnih komponenti grafičke korisničke sprege EPG-a

**TableView** komponenta prikazuje tabelu svih dostupnih kanala dobijenih od STB. Svaki element tabele (kanal) predstavljen je komponentom **CustomTableViewCell** koja u sebi sadrži sliku kanala (**Cell Image**) i ime kanala (**Cell Text**).

Elektronski programski vodič prikazuje kompletan spisak kanala i emisija u period od 24 sata. Pregledavanje kanala i emisija dostupno je klizanjem prsta u levu ili desnu stranu, pri čemu se prikazuje ostatak sadržaja koji pre klizanja nije bio dostupan korisniku. Za tu svrhu je iskorišćena nativna **ScrollView** komponenta. Zbog zahteva aplikacije da se omogući rad i a iOS platformi za iPad 1 (iOS 5.1) morala se koristiti komponenta ScrollView. Postoji lakša realizacija pomoću komponente CollectionView, međutim ona nije podržana na iPad 1 uređaju zbog svog ograničenja da radi na platformama počevši od iOS 6.0. Unutar ove komponente nalazi se komponenta **RowElement**, koja predstavlja kompletan programski sadržaj jednog kanala. ScrollView komponenta u sebi sadrži onoliko RowElement komponenti, koliko ima dostupnih kanala na STB. Unutar svake RowElement komponente, nalazi se N **CellElement** blokova, gde je N broj emisija koje se nalaze unutar jednog kanala. Svaki blok prikazuje ime emisije (**Element Name**), kao i vreme kada emisija počinje i kada se završava (**Element Duration**). Svakom CellElementu se dodaje gest koji obrađuje kratak pritisak, a pritiskom na blok otvara se dijalog sa detaljnim informacijama vezanim za dati kanal: datum, vreme prikaza, ime i slika kanala, ime emisije, žanr, kratak opis emisije. Dijalog sadrži i 4 dugmeta: promenu kanala na glavnom TV ekranu, dodavanje izabranog kanala u mozaik i gledanje uživo izabranog kanala u LiveTV delu aplikacije. Svaki CellElement uokviren je bojom koja predstavlja jedan od 12 predefinisanih žanrova. Izgled panela, kao i neke od boje programa u zavisnosti od žanra, mogu se videti na Slika 9.

Pored svih ovih komponenti EPG sadržaj sadrži i vremensku dimenziju, za šta se koristi **TimeScrollView** komponenta. To je namenski napravljena komponenta za prikaz 24 sata na osi kompletног prikaza emisija. Gestovima je direktno povezana sa ScrollView komponentom i kreću se istovremeno. Unutar TimeScrollView komponente klizajući gestovi nisu dostupni. Komponenta korisniku daje informaciju o trenutnom vremenskom položaju unutar EPG sadržaja.

#### **4.2.5 RemoteControllerViewController, TouchpadViewController, KeyboardViewController**

Pored prikaza sadržaja sa glavnog ekrana, aplikacija sa sekundarnim ekranom treba uključiti i mehanizme upravljanja TV uređajem. Ona korisniku pruža mogućnost da pomoću replika daljinskog upravljača oponaša funkcionalnosti primarnog daljinskog upravljača za kontrolu STB-a. Daljinska kontrola je proširena tastaturom i mišem u obliku pokazivačke pločice (eng. *touchpad*). Izgled pomenutih komponenti može se videti na Slika 12. Tastatura i pokazivačka pločica koriste se za rukovanje sadržajem unutar Android dela STB-a, dok se daljinski upravljač koristi kao klasični daljinski upravljač u interakciji sa TV-om.



Slika 12 Daljinsko upravljanje sačinjeno iz modula: tastatura, touchpad i daljinski upravljač

RemoteControllerViewController modul sadrži ScrollView komponentu. Ova komponenta u sebi sadrži 3 ekrana za prikaz daljinskog upravljača kojima se može pristupiti preko ručnih gestova klizanjem prsta u levu ili desnu stranu, ili odabirom opcije na navigacionom baru. Prvi ekran predstavlja kompletну repliku standardnog daljinskog upravljača za TV uređaje, drugi predstavlja samo numerički deo, a treći samo komande za rukovanje meni opcijama.

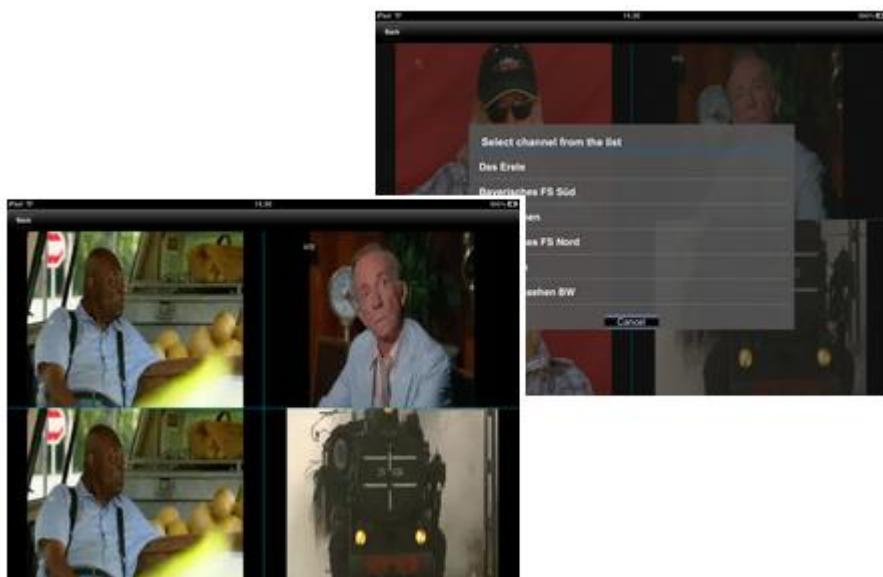
KeyboardVlewController modul realizuje standardnu iOS tastaturu, proširenu navigacionim strelicama i specijalnim karakterima. Tastatura je realizovana kao niz dugmadi prilagođenih tako da izgled podseća na izgled dugmadi na daljinskom upravljaču. Svako dugme sadrži informaciju o specijalnom karakteru koji se nakon pritiska dugmeta šalje ka STB u vidu komande preko UPNP protokola. Specijalni karakteri na tastaturi dobijaju se kao i kod standardne tastature pritiskom specijalnog dugmeta koje umesto slova, korisniku prikazuje listu specijalnih karaktera.

TouchpadVlewController modul pomoću namenske kontrole TouchpadControll realizuje funkcionalnosti pokazivačke pločice, proširene za komande vertikalnog i horizontalnog klizača, kao i klika na površinu namenske kontrole koji simulira klik na glavnom TV ekranu.

#### 4.2.6 MosaicViewController

Modul omogućava korisniku pristup paralelnom video prikazu do četiri kanala na jednom ekranu sa mogućnošću odabira kanala koje želi da gleda preko celog ekrana (Slika 13). Ovaj prikaz sadržaja može se iskoristiti za pregled omiljenih kanala.

Iz mozaičnog prikaza korisnik ima mogućnost da gleda uživo na uređaju sa sekundarnim ekranom odabrani kanal jednim klikom na željeni video. Dodatne mogućnosti za svaki TV kanal dostupne su u vidu priručnog menija, dugačkim pritiskom na željeni video. Na odabir željenog kanala iz mozaika, korisnik pristupa LiveTVViewController modulu za prikaz uživo TV prenosa. Svaki od kanala prikazuje se u posebnoj **ImageView** komponenti iz razloga što on ima podršku za prikaz MJPG video formata. Lista kanala koja se prikazuje u priručnom meniju, kao i provera da li kanali koji su prethodno sačuvani u memoriju kao poslednji gledani, obavlja se pri ulasku u prozor MosaicViewController-a u istoj niti u kojoj se aplikacija izvršava.



Slika 13 Mozaični prikaz sadržaja na sekundarnom ekranu

#### 4.2.7 MediaExplorerViewController

Osnovna funkcionalnost ovog modula jeste pretraživanje sadržaja na media poslužiocima. DLNA Media Explorer pretraživač nakon prikaza svih dostupnih uređaja korisniku omogućava pristup svim direktorijumima unutar posluživača, kao i mogućnost prikaza slika, video i audio zapisa preko lokalnih iOS komponenti za reprodukciju zvuka, videa ili pregleda slika. Navigacija kroz stablo pretraživača implementirana je kao namenska kontrola koja omogućava povratak na bilo koji nivo iz stabla pretraživača. Pored multimedijalnog sadržaja, aplikacija sa dodatnim ekranom pruža korisniku uvid u ostale dokumente dostupne na poslužitelju, međutim nije realizovana podrška za otvaranje takvog sadržaja (Slika 14).

Ograničenje sistema koji koristi DLNA i UPNP na iOS platformi jeste taj da se klijenti koji opslužuju zahteve korisnika ne mogu kreirati istovremeno. To znači da se svaki put pri ulasku u MediaExplorer mora startovati nova instance DLNA, a pri povratku na glavni meni mora startovati nova instanca UPNP objekta. Ovo ograničenje javlja se iz razloga što svaki klijent mora biti kreiran u odvojenom procesu, a operativni sistem aplikacijama dozvoljava da svaka sadrži u jednom trenutku vremena samo jedan ovakav proces. Primjeno na kontekst grafičke korisničke sprege, to znači da pri ulasku u Media Explorer, korisnik svaki put mora sačekati nekoliko sekundi, kako bi se inicijalizacija jednog ovakvog klijenta izvršila.

Komunikacija se sastoji iz ispitivanje mreže i pretraživanje uređaja sa sposobnošću prikaza multimedijalnog sadržaja. Nakon odabira želenog uređaja, datom uređaju šalju se poruke za dobijanje sadržaja na zahtev. Sadržaj može biti lista elemenata unutar direktorijuma ili dobavljanje celokupnog multimedijalnog sadržaja u vidu slike, zvuka ili video fajla. Nakon dobijenog odgovora, grafička korisnička sprega obaveštava se o prispelim informacijama.

Grafička korisnička sprega sastoji se iz ScrollView komponente na vrhu i liste za prikaz sadržaja ispod nje. ScrollView komponenta u sebi sadrži dugmad koja predstavljaju navigaciju kroz sadržajno stablo. Krajnje desno dugme predstavlja trenutno izlistan direktorijum. Pritisom na dugme unutar ove komponente, prikazuje se sadržaj direktorijuma koji je pritisnut. Smisao ScrollView komponente jeste da prikaže veći broj dugmadi kada njihovo iscrtanje premaši granice ekrana, a da pritom korisniku omogući povratak u bilo koji deo stabla klizanjem prsta po ovoj komponenti u levu ili desnu stranu. Lista sadržaja trenutnog direktorijuma inicijalno pri ulasku u ovaj modul prikazuje listu uređaja koji se preko SSDP protokola iz DLNA biblioteke javi aplikaciji (DLNA biblioteka korištena u ovoj aplikaciji jeste već postojeća DLNA biblioteka razvijena od strane RTRK instituta). Ova lista se u tu svrhu u korenskom direktorijumu periodično osvežava. Odabirom elementa u listi pristupa se narednom nivou stabla. Tipovi elemenata koji mogu biti odabrani su sledeći:

- Direktorijum – u ovom slučaju poslužitelju koji se pretražuje šalje se zahtev za dobavljanje sadržaja trenutnog direktorijuma
- Slika – otvara se novi kontroler za prikaz slike (standardni iOS kontroler za pregled sadržaja ovog tipa), a podržani formati slike su sledeći: jpeg, gif, bmp.
- Video – otvara se novi kontroler za prikaz video sadržaja (standardni iOS video plejer), a podržani formati video sadržaja su sledeći: mp4, mov, 3gp, mpv.
- Zvuk – otvara se novi kontroler za prikaz audio sadržaja (standardni iOS audio plejer), a podržani formati audio sadržaja su sledeći: mp3, mid, wav.

- Nepoznat fajl – u ovu grupu elemenata spadaju svi dokumenti čiji formati nisu podržani od strane aplikacije. Ovakvi elementi neće biti otvoreni, a trenutni direktorijum, posle njihovog klika, neće se promeniti.



Slika 14 Prikaz Media Explorer modula

#### 4.2.8 SettingsViewControlller

Modul realizuje tabelu koja prikazuje listu svih dostupnih poslužitelja unutar mreže sa mogućnošću odabira jednog od njih. Indikator za odbran poslužitelja nalazi se sa desne strane liste. Osetljivost pokazivačke pločice takođe se može podesiti unutar ovog modula.

## 5. Testiranje i verifikacija

Testiranje i verifikacija su izvršeni primenom dva različita pristupa:

- Prvi pristup predstavlja testiranje programa ručno primenom sistemskog testiranja, testiranje osnovnih funkcionalnosti i integracionih testova. Ovaj pristup koristi testne scenarije kojima se ispituje funkcionalnost korisničke sprege, kao i subjektivni kvalitet odziva aplikacije na korisničke komande.

Drugi pristup je automatsko testiranje programa korišćenjem standardnog alata Instruments sadržanog u xCode okruženju gde je vršeno samo sistemsko testiranje. U daljem tekstu opisće se pojedini pristupi.

### 5.1 Ručno ispitivanje programa

Ispitivanje i verifikacija funkcionalnosti, navigacije i interakcije sa korisnikom u iOS aplikaciji svodi se na oponašanje upotrebe aplikacije od strane korisnika. Treba ispitati da li se svaka komponenta, vidljiva na ekranu, može obeležiti i izabrati, da li je pritisak dugmeta propraćen odgovarajućom reakcijom, odnosno kakve su posledice nasumičnog pritiskanja dugmadi.

Glavna ispitivanja bila su funkcionalnost korisničke sprege, odziv grafičke korisničke sprege na spoljašnje događaje, kao i otpornost aplikacije na greške prilikom korišćenja u dužem vremenskom periodu. U daljem tekstu biće navedeno nekoliko primera ovakvih testova.

Rb	Opis testa	Očekivani rezultat	Uspešnost
1	<b>Opis:</b> Puštanje TV prenosa uživo i dobijanje dodatnih informacija za selektovani kanal. <b>Početni uslovi:</b> STB i iPad moraju biti povezani na istu lokalnu mrežu. Aplikacija se mora nalaziti na ekranu sa listom dostupnih kanala	Na ekranu treba da se pojavi LiveTVViewController prozor, panel sa dodatnim podacima u vidu EPG-a za odabrani kanal, brojem kanala u listi, kao i trenutno ime epizode koja se trenutno	✓

	(ChannelListViewController).	emituje. Video komponenta treba krenuti sa puštanjem video sadržaja dobijenog od servera nakon nekog kraćeg vremenskog perioda.	
2	<p><b>Opis:</b> Prikaz EPG sadržaja za 1 kanal</p> <p><b>Početni uslovi:</b> STB i iPad moraju biti povezani na istu lokalnu mrežu. Aplikacija se mora nalaziti na početnom ekranu(MainMenuViewController).</p> <p><b>Postupak:</b></p> <ul style="list-style-type: none"> <li>• Odabratи jedan od ponuđenih ekrana.</li> <li>• Sačekati da program izvrši pokretanje LiveTVViewController moodula i da se tok video podataka pokrene.</li> </ul>	Dijalog se nakon dobijanja kompletног EPG sadržaja jednog kanala uklanja sa ekrana, a na ekranu se prikazuju slika kanala, ime kanala i emisije povezane sa kanalom.	✓
3	<p><b>Opis:</b> Pristup jednom od DLNA poslužitelja iz liste</p> <p><b>Početni uslovi:</b> iPad mora biti povezan na neku lokalnu mrežu. Aplikacija se mora nalaziti na početnom ekranu(MainMenuViewController).</p> <p><b>Postupak:</b></p> <ul style="list-style-type: none"> <li>• Pritisnuti Media Explorer dugme.</li> <li>• Sačekati nekoliko sekundi da se dijalog ukloni sa ekrana i da se prikaže barem jedan media poslužitelj.</li> <li>• Odabratи jedan od prikazanih poslužitelja.</li> </ul>	Dijalog se nakon pronašlaska prvog poslužitelja uklanja i korisniku je dozvoljen pristup tom poslužitelju. Nakon klika na jedan od poslužitelja iz liste, otvara se root direktorijum na serveru i prikazuje se njegov sadržaj u obliku tabele koja zamjenjuje listu poslužitelja na ekranu.	✓
4	<p><b>Opis:</b> Odabir drugog STB uređaja unutar podešavanja aplikacije</p> <p><b>Početni uslovi:</b> iPad mora biti povezan na lokalnu mrežu. Aplikacija se mora nalaziti na početnom ekranu(MainMenuViewController).</p> <p><b>Postupak:</b></p> <ul style="list-style-type: none"> <li>• Pritisnuti Settings dugme.</li> <li>• Iz liste svih dostupnih servera, odabratи željeni uređaj.</li> </ul>	Ulaskom u podešavanja aplikacije prikazuje se lista dostupnih STB uređaja, a trenutno odabran STB je označen. Pritiskom nekog drugog STB-a stara oznaka se uklanja, a novi postaje označen.	✓

Tabela 1 Testiranje grafičke korisničke sprege

## 5.2 Automatsko testiranje

Automatski testovi su veoma korisni pri testiranju aplikacije “za vreme noći”. Izvršavaju se pomoću ugrađenog alata unutar xCode okruženja zvanog Instruments. Ovaj alat u sebi sadrži komponentu **UIAutomation** koja pravljenjem skripte u JavaScript programskom jeziku može da oponaša korisnika bez ikakvog dodira uređaja. Ovaj alat podržan je od iOS 4.0 verzije i moguće je testirati aplikacije na samim uređajima, kao i na simulatorima dostupnim u xCode okruženju. Glavna prednost ovakvih testiranja aplikacije jeste što jednom napisan test može da se ponovi na različitim uređajima kako bi se proverila podrškarazličitim konfiguracijama i verzijama iOS operativnog sistema. Takođe jednom napisan test može se ponoviti nekoliko puta u toku samog pravljenja aplikacije čime se može uštedeti na vremenu. Jedan primer izvršenog teksta biće objašnjen u daljem tekstu zajedno sa primerom skripte koja je korištena za njegovo izvršenje.

```

var target = UIATarget.localTarget();
var app = target.frontMostApp();
var mainWindow = app.mainWindow();

function initialScreenTest() {
    var testname = "Initial screen test";
    UIALogger.logStart(testname);
    mainWindow.logElementTree();
    UIATarget.localTarget().tap({x:100, y:200});
    UIALogger.logPass(testname);
}

function tableScreenTest() {
    var testname = "Table screen test";
    UIALogger.logStart(testname);
    var rootTable = mainWindow.tableViews()[0];
    rootTable.cells()[2].tap();
    target.delay(1);
    UIATarget.localTarget().tap({x:320, y:350});
    UIALogger.logPass(testname);
}

function liveTVScreenTest() {
    var testname = "LiveTV screen test";
    UIALogger.logStart(testname);
    target.delay(4);
    UIATarget.localTarget().dragFromToForDuration({x:500, y:300},{x:20,y:300},0.5);
    UIALogger.logPass(testname);
}

initialScreenTest();
tableScreenTest();
liveTVScreenTest();

```

Slika 15 Primer skripte korištene za automatsko testiranje aplikacije

Skripta data na Slika 15 Primer skripte korištene za automatsko testiranje aplikacije predstavlja automatski test promene kanala sa trenutno gledanog na naredni kanal u listi. Sadrži 3 funkcije koje predstavljaju 3 glavne celine ovog testa:

- Prvi deo predstavlja simulaciju pritiska dugmeta unutar glavnog menija.
- Drugi deo predstavlja odabir elementa iz liste i prilikom pojavljivanja dijaloga sa opcijama biranje opcije da se željeni kanal gleda na uređaju sa sekundarnim ekranom. Nakon odabira kanala prelazi se u LiveTV deo aplikacije, gde se na ekranu prikazuje video sadržaj odabranog kanala.
- Treći deo testa predstavlja simulaciju ručnog gesta pomoću kojeg se video prekida, a video sadržaj narednog kanala iz liste prikazuje korisniku.

Izvršeni automatski testni slučajevi poklapaju se sa slučajevima korištenim prilikom ručnog testiranja. Dobijeni rezultati ne odstupaju od rezultata testiranja iz predhodnog poglavlja o ručnom testiranju.

### 5.3 Profilisanje

U ovom poglavlju prikazano je merenje procesorskih resursa prilikom rada aplikacije. Merenja su izvršena samo na izolovanim delovima aplikacije za koje je procenjeno da mogu trošiti najviše resursa, a to su LiveTV i EPG. Prvi deo profiliše se u trenutku puštanja video snimka i prilikom promene kanala na sekundarnom ekranu gde se troši najviše memorije i zauzima najviše procesorskog vremena. Drugi deo aplikacije koji je veoma zahtevan za memoriju zbog količine podataka jeste dobavljanje i prikaz elektronskog programske vodiča iz razloga što se veoma velika količina podataka dobijena od STB-a skladišti u memoriji, a potom se od tih podataka crtaju grafički elementi na ekranu.

Profilisanje je obavljeno korišćenjem Instruments alata unutar xCode okruženja gde je u svrhu testiranja, korišten iPad Air sa 1.4 GHz procesorom sa 2 jezgra i 1GB RAM memorije. Aplikaciji je na raspolaganju 768MB memorije preko koje granice aplikacija prestaje sa svojim radom iz sigurnosnih razloga. Za razliku od memorije, procesorsko zauzeće nema ograničenje i može dostići 100% u kojem trenutku aplikacija prestaje da reaguje na komande koje korisnik zadaje, dok se procesorsko zauzeće ne smanji.

U tabeli 1 može se videti zauzeće procesora i memorije za vreme prikaza video sadržaja korisniku i za vreme promene kanala na sekundarnom ekranu. Vrednost u tabeli predstavlja prosečnu vrednost 20 odbiraka sakupljenih u periodu od 4 sekunde.

	Procesor	Memorija
Prikaz sadržaja	25,5%	62MB
Promena kanala	43,2%	50MB

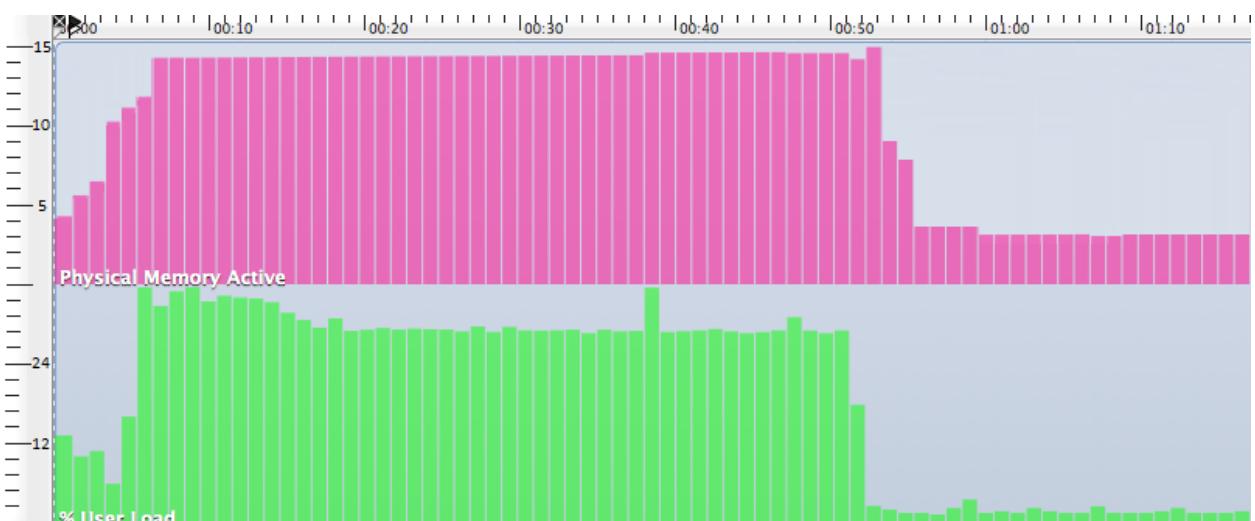
Tabela 2 Prikaz zauzetosti memorije i procesora u LiveTV delu aplikacije

Tabela 2 prikazuje zauzeće procesora i memorije za vreme prikupljanja podataka potrebnih za prikaz elektronskog programskega vodiča, kao i nakon prikupljanja kompletne sadržaje i prikaza grafičkih komponenti na ekran. Vrednost u tabeli predstavlja prosečnu vrednost 20 odbiraka sakupljenih u periodu od 4 sekunde.

	Procesor	Memorija
<b>Za vreme prikupljanja podataka</b>	38,2%	62MB
<b>Nakon prikupljanja podataka</b>	4,8%	58MB

Tabela 3 Prikaz zauzetosti memorije i procesora u EPG delu aplikacije

Primer grafika na osnovu kojeg su izmereni rezultati iz tabele 2 i 3, dobijen unutar alata korištenog za profilisanje aplikacije dat je na Slika 16. Gornja polovina grafika predstavlja izmerenu potrošnju memorije u intervalu vremena, dok donja polovina predstavlja izmereno procesorsko zauzeće u istom tom intervalu vremena.



Slika 16 Izgled grafika unutar okruženja zaprofilisanje aplikacije

Iz prethodno izvršenih testova može se primetiti da u najzahtevnijim delovima, aplikacija i dalje radi u sigurnoj zoni, daleko od sistemske granice. Ovako veća potrošnja memorije i zauzeće procesora jeste očekivana zbog prenosa video formata preko mreže, kao i zbog konstantnog isrtavanja nove slike u prikazu video sadržaja. Takođe zbog velike količine pristiglih podataka unutar EPG-a i zbog velikog broja grafičkih komponenti korisničke sprege dolazi do većeg zauzeća memorije, dok se procesorsko zauzeće smanjuje čim se komunikacija sa serverom izvrši. Aplikacija za vreme svog rada u glavnom meniju nakon pokretanja zauzima 12MB i 2% procesorskog vremena.

## 5.4 Poređenje sa postojećim rešenjima

Realizovana aplikacija nije jedinstvena i samim tim kompanije koje se bave uvođenjem sekundarnog ekrana u televizijsku interakciju sa korisnikom već korisnicima svojih TV uređaja nude svoje aplikacije za ove potrebe. U daljem tekstu biće navedene konkurentne aplikacije i biće navedeni delovi tih aplikacija relevantni za uporedni test koji će kasnije biti sproveden.

### 1. Samsung TV Media Player i Samsung SmartView

- Kompanija Samsung za svoje televizijske prijemnike objavila je TV Media Player aplikaciju [12] sa sledećim funkcionalnostima: pretraživanje media uređaja priključenih na lokalnu mrežu, kao i prikaz video i audio sadržaja sa media uređaja unutar aplikacije ili na TV uređaju.
- Samsung SmartView aplikacija [14] uvodi daljinsko upravljanje TV uređaja, kao i prikaz televizijskog sadržaja sa glavnog ekrana na sekundarnom ekranu.

### 2. LG TV Remote i LG TV Media Player

- Kompanija LG je za potrebe sekundarnog ekrana za prikaz televizijskog sadržaja objavila aplikaciju LG TV Remote [10]. Glavne podržane funkcionalnosti ove aplikacije jesu kontinualni pregled TV prenosa uživo kao i daljinsko upravljanje putem daljinskog upravljača, tastature i pokazivačke pločice.
- Druga aplikacija kompanije LG za prikaz televizijskog sadržaja objavila je aplikaciju LG TV Media Player [11] koja od funkcionalnosti sadrži TV prenos uživo i pretraživanje media uređaja priključenih na lokalnu mrežu.

### 3. Roomie Remote

- Aplikacija je razvijena od strane kompanije Cyphersoft [13] i podržava prikaz dodatnog sadržaja u vidu elektronskog programske vodiča putem DLNA zahteva, daljinsko upravljanje TV prijemnika i pretraživanje media uređaja priključenih na lokalnu mrežu. Aplikacija se koristi za kontrolu TV uređaja marke Samsung, Elite, Panasonic, Sony, Sharp i Elite.

### 4. AirPlay

- Aplikacija je stvorena od strane kompanije Apple kako bi Apple TV povezali sa ostalim proizvodima sa sekundarnim ekranom ove kompanije [15]. Aplikacija preko lokalne mreže omogućava deljenje sadržaja kao i prikaz tog sadržaja na televizijskom prijemniku uporedno sa njegovim prikazom na sekundarnom ekranu.

	Live TV	EPG	Mosaic	DLNA	Remote	Touch Pad	Keyboard
Samsung TV MP	✓	✗	✗	✓	✓	✗	✗
Samsung SmartView	✓	✗	✗	✗	✓	✗	✓
LG TV Remote	✓	✗	✗	✓	✓	✓	✓
LG TV MP	✓	✗	✗	✓	✗	✗	✗
Roomie Remote	✓	✓	✗	✗	✓	✗	✗
AirPlay	✗	✗	✗	✓	✗	✗	✗
Moje Rešenje	✓	✓	✓	✓	✓	✓	✓

Slika 17 Uporedni prikaz postojećih rešenja i mog rešenja

Za razliku od postojećih rešenja, dato rešenje implementira kompletan set funkcionalnosti u okviru jedne aplikacije što se može videti na Slika 17. Funkcionalnosti kao što su elektronski programski vodič, sekundarni TV ekran...povezane su tako da korisniku omoguće lakše rukovanje. Već navedeni ručni gestovi za promenu kanala u sekundarnom TV ekranu, prelazak na sekundarni TV ekran iz mozaičnog pregleda kanala klikom na kanal...

Za razliku od Samsungovog i LG rešenja aplikacije sa sekundarnim ekranom, podržan je prikaz kanala koji nije prikazan na TV prijemniku. Takođe dostupan je mozaični prikaz do 4 kanala istovremeno, što navedene aplikacije ne podržavaju.

## 6. Zaključak

Ovaj rad predstavlja kompletну realizaciju klijentske aplikacije sa dodatnim ekranom kao kompletne zamene televizijskog prikaza na glavnom ekranu, koja potpuno odvojeno od glavnog ekrana prikazuje i upravlja TV sadržajem. Aplikacija obuhvata prenos televizijskog sadržaja na sekundarnom ekranu, EPG, mozaični prikaz video sadržaja, pretragu sadržaja na multimedijalnih poslužiteljima, kao i neke od komponenata za rukovanje i prikaz digitalnog sadržaja.

Sama mogućnost sistema za prikaz više televizijskih sadržaja u paraleli omogućila je aplikaciji da implementira kompletan set funkcionalnosti ne vezano od sadržaja prikazanog na glavnom ekranu.

Ovo rešenje za prikaz televizijskog sadržaja podržava nekoliko formata za slanje TV sadržaja: H264, MPEG2, MPEG4, što proširuje set uređaja na koje ovaj sistem može da se poveže.

## 7. Literatura

- [1] Bratislav Milovanović, Miodrag Temerinac, Nikola Teslić, Jugoslav Joković: *Digitalizacija zemaljskih TV sistema*, XXVI Simpozijum o novim tehnologijama u poštanskom i telekomunikacionom saobraćaju - PosTel, Beograd, 2008
- [2] Oficijalna Apple dokumentacija o iOS MVC  
obrazcu <https://developer.apple.com/library/ios/documentation/general/conceptual/devpedia-cocoacore/MVC.html>
- [3] Dr. Siniša Vlajić doc., *Softverski paterni*, radni materijal sa predavanja, Beograd, 2011
- [4] L. Cruickshank, E. Tsekleves, R. Whitham, A. Hill, K. Kondo “Making Interactive TV Easier to Use: Interface Design for a Second Screen Approach”, The Design Journal, vol. 10, Number 3, Novembar 2007, pp.41-53
- [5] I. Vidal, J. Garcia-Reinosa, F. Valera, A. Bikfalvi, “Enabling layered video coding for ISM-based IPTV home services”, IEEE Journal & Magazines, vol.23, issue 6, pp.30-35
- [6] W. Seager, H. Knoche, M. Sasse, “TV-Centricity-requirements gathering for triple play services”, Adjunct Proceedings of the European Conference on Interacitve Television, pp.274-278, 2007
- [7] R. Bernhaupt, M. Obrist, A. Weiss, E. Beck, M. Tscheiligi, “Trends in the living room and beyond”, Proceedings of the European Conference on Interactive Television, pp.146-155, 2007
- [8] P. Cesar, D. C. A. Bulterman, J. Jansen, “Leveraging user impact: an architecture for secondary screens usage in interactive television”, Multimedia System Journal, vol. 15, Issue 3, pp.127-142, 2009

- [9] I.Bogdanović, I. Papp, R. Pavlović, B. Brazovac, "Analiza dekodovanja i prikaz performansi različitih okruženja za obradu video sadržaja na iOS operativnom sistemu", Etran, 2014
- [10] <https://itunes.apple.com/us/app/lg-tv-remote/id509979485?mt=8>
- [11] <https://itunes.apple.com/gb/app/lg-tv-media-player/id514993821?mt=8>
- [12] <https://itunes.apple.com/us/app/samsung-tv-media-player/id514991150?mt=8>
- [13] <http://www.roomieremote.com/>
- [14] <https://itunes.apple.com/us/app/samsung-smartview/id359580639?mt=8>
- [15] <https://www.apple.com/airplay/>