



УНИВЕРЗИТЕТ У НОВОМ САДУ

ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



УНИВЕРЗИТЕТ У НОВОМ САДУ

ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА

НОВИ САД

Департман за рачунарство и аутоматику

Одсек за рачунарску технику и рачунарске комуникације

ЗАВРШНИ (BACHELOR) РАД

Кандидат: Раде Максимовић

Број индекса: е12561

Тема рада: Интеграција DLNA подршке на пријемнику дигиталног телевизијског сигнала заснованог на Android оперативном систему

Ментор рада: др Милан Ђелица

Нови Сад, септембар 2016.



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:	Раде Максимовић		
Ментор, МН:	Др Милан Ђелица		
Наслов рада, НР:	Интеграција DLNA подршке на пријемнику дигиталног телевизијског сигнала заснованог на Android оперативном систему		
Језик публикације, ЈП:	Српски / латиница		
Језик извода, ЈИ:	Српски		
Земља публиковања, ЗП:	Република Србија		
Уже географско подручје, УГП:	Војводина		
Година, ГО:	2016.		
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)			
Научна област, НО:	Електротехника и рачунарство		
Научна дисциплина, НД:	Рачунарска техника		
Предметна одредница/Кључне речи, ПО:	DLNA, Google TV, Android, UPnP, мултимедија		
УДК			
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад		
Важна напомена, ВН:			
Извод, ИЗ:	У раду је описан приступ имплементације DLNA у Android оперативни систем и добављање расположивог мултимедијалног DLNA материјала коришћенjem QSB претраге.		
Датум прихватања теме, ДП:			
Датум одбране, ДО:			
Чланови комисије, КО:	Председник:	Др Јелена Ковачевић, Доцент	
	Члан:	Др Немања Лукић, Доцент	Потпис ментора
	Члан, ментор:	Др Милан Ђелица, Доцент	



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:	Rade Maksimović	
Mentor, MN:	Milan Bjelica, PhD	
Title, TI:	DLNA integration into DTV receiver based on Android operating system	
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:	2016.	
Publisher, PB:	Author's reprint	
Publication place, PP:	Novi Sad, Dositeja Obradovica sq. 6	
Physical description, PD: <small>(chapters/pages/ref./tables/pictures/graphs/appendices)</small>	Digital television, Teletext, Android, Linux	
Scientific field, SF:	Electrical Engineering	
Scientific discipline, SD:	Computer Engineering, Engineering of Computer Based Systems	
Subject/Key words, S/KW:	DLNA, Google TV, Android, UPnP, multimedia	
UC		
Holding data, HD:	The Library of Faculty of Technical Sciences, Novi Sad, Serbia	
Note, N:		
Abstract, AB:	<p>This paper describes an implementation of DLNA in Android operating system and using QSB for available DLNA material delivery.</p>	
Accepted by the Scientific Board on, ASB:		
Defended on, DE:		
Defended Board, DB:	President:	Jelena Kovačević, PhD
	Member:	Nemanja Lukić, PhD
	Member, Mentor:	Milan Bjelica, PhD
		Menthor's sign

Zahvalnost

Zahvaljujem se mentoru dr Milanu Bjelici na podršci, strpljenju i odvojenom vremenu te dr Nemanji Lukiću i Mladenu Iliću na smernicama i inženjerskim savetima u toku izrade rada.

Takođe se zahvaljujem porodici i kolegama na podršci koji su mi pružili tokom izrade ovog rada i tokom osnovnih akademskih studija.

Novi Sad, septembar 2016.

Rade Maksimović



УНИВЕРЗИТЕТ У НОВОМ САДУ

ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



SADRŽAJ

1.	Uvod	8
2.	Teorijske osnove.....	10
2.1	Digitalna televizija	10
2.2	UPnP protokol.....	11
2.3	DLNA protokol.....	12
2.3.1	Kućni mrežni uređaji	12
2.3.2	Ručni mobilni uređaji	12
2.3.3	Kućni infrastrukturni uređaji	13
2.3.4	Formati multimedijalnog sadržaja i transportni model	13
2.4	Android operativni sistem.....	14
2.4.1	GoogleTV	15
2.4.2	Google Quick Search Box	16
3.	Koncept rešenja	17
3.1	Opis ciljne platforme.....	17
3.2	Realizacija Android aplikacije za pretragu DLNA	18
3.2.1	Modul rukovaoca bazom podataka.....	18
3.2.2	Modul za dobavljanje sadržaja	18
3.2.3	Modul za pronalaženje DLNA servera i pretragu njihovog sadržaja	19
3.2.4	Moduli za prikaz DLNA sadržaja.....	19
4.	Programsko rešenje.....	20
4.1	Građenje izvornog koda	20
4.2	Programsko rešenje Android aplikacije za pretragu DLNA servera.....	21
4.2.1	Podešavanje AndroidManifest.xml	21

4.2.2	DLNADatabase.java.....	22
4.2.3	DLNAProvider.java.....	22
4.2.4	MainActivity.java.....	23
4.2.5	VideoView.java	23
4.2.6	PictureViewActivity.java	24
4.2.7	MultimediaContent.java	25
4.2.8	BrowseDialog.java	25
4.2.9	CustomAdapter.java	25
5.	Rezultati.....	26
6.	Zaključak	29
7.	Literatura	30

SPISAK SLIKA

Slika 1. Primer komunikacije korišćenjem UPnP.....	11
Slika 2. Primer DLNA mreže	14
Slika 3. Arhitektura Android operativnog sistema	15
Slika 4. Izgled Google TV QuickSearchBox aplikacije	16
Slika 5. RK-2010 platforma.....	18
Slika 6. Pristup bazi podataka preko Content Provider	19
Slika 7. Klase DLNADatabase i DLNAProvider	23
Slika 8. Klasa VideoViewActivity.....	23
Slika 9. Klasa PictureViewActivity	24
Slika 10. Klasa MultimediaContent.....	25
Slika 11. Klasa CustomAdapter.....	25
Slika 12. Koreni folder odabranog servera i lista DLNA servera (u pozadini)	26
Slika 13. Prikaz fotografija	27
Slika 14. Reprodukovanje audio i video fajlova.....	27
Slika 15. Primer pretrage korišćenjem QSB-a.....	28

SPISAK TABELA

Tabela 1. Formati multimedijalnog sadržaja kućnih DLNA uređaja.....	13
Tabela 2. Formati multimedijalnog sadržaja mobilnih DLNA uređaja	13

SKRAĆENICE

- AIDL - Android Interface Definition Language
- API - Application Programming Interface, Aplikacioni programski interfejs
- CRUD - Create Read Update Delete
- DLNA - Digital Living Network Alliance, Protokol za deljenje digitalnog sadržaja između multimedijalnih uređaja
- DMC - Digital Media Controller, Modul za kontrolu u DLNA mreži
- DMP - Digital Media Player, Modul za reprodukciju u DLNA mreži
- DMS - Digital Media Server, Poslužilac u DLNA mreži
- DVB - Digital Video Broadcasting, Emitovanje digitalnog videa
- EPG - Electronic Program Guide, Elektronski programski vodič
- HTTP - Hypertext Transfer Protocol, Aplikativni protokol za komunikaciju
- IP - Internet Protocol, Internet protokol
- JNI - Java Native Interface, Programska sprega u okviru Android operativnog sistema
- PVR - Personal Video Recorder
- QSB - Quick Search Box, Aplikacija za brzu pretragu
- TCP - Transmission Control Protocol, Protokol kontrole toka
- UPnP - Universal Plug and Play, Skup mrežnih protokola koji dozvoljavaju multimedijalnim uređajima da ostvare međusobne veze i uspostave razne mrežne usluge
- URI - Uniform Resource Identifier, Jedinstvena identifikacija resursa
- URL - Uniform Resource Locator, Jedinstvena adresa resursa

XML - Extensive Markup Language, proširivi metajezik za označavanje tekstualnih dokumenata

1. Uvod

U proteklim godinama korisnici su prihvatili ubrzani razvoj digitalnih tehnologija. Vođeni internetom, žičanim i bežičnim kućnim mrežama, digitalnim uređajima kao što su muzički plejeri, digitalne kamere i kamkoderi, oni nabavljaju, uživaju i upravljaju raznim digitalnim sadržajima iz svoje kuće. Jedan od najboljih načina pristupa sadržaja na lokalnoj mreži je definisala organizacija DLNA forum. DLNA protokol stek je način deljenja, kontrole i reprodukcije multimedijalnog sadržaja u lokalnoj mreži. Nastao je iz UPnP protokola sa ciljem da jasno definiše rad sa multimedijalnim sadržajem unutar lokalne mreže.

Android zauzima vodeće mesto među operativnim sistemima za mobilne korisničke računarske sisteme kao što su tableti i telefoni. Dugi niz godina Android operativni sistem je bio razvijan za platforme zasnovane na ARM (Acorn(Advanced) RICS Machine) arhitekturi procesora. Razvojem i prilagođavanjem Androida otvorile su se nove mogućnosti među kojima je i da novi savremeni DTV prijemnici rade na Android operativnom sistemu. Ovo rešenje sistema za reprodukciju multimedijalnog sadržaja je nezavisno od fizičke platforme i može se koristiti na svim uređajima koji koriste Android operativni sistem.

Google TV je platforma bazirana na Android operativnom sistemu koji čini televizor "pametnim" uređajem što podrazumeva ne samo reprodukciju televizijskih kanala već i pružanje još većih mogućnosti korisniku kao što su pretragu interneta, snimanje emisija, igranje igara i slično.

U ovom radu, potrebno je iskoristiti QSB aplikaciju za pretraživanje raspoloživosti DLNA materijala, njihov prikaz krajnjem korisniku, kao i uključivanje DLNA rezultata pretrage u okviru QSB pretrage. Prilikom pretrage korisnik unosi ključne reči u polje za

pretragu, odabirom jednog od rezultata, korisnik dobija prikaz pronađenog multimedijalnog materijala na DLNA serverima lokalne mreže.

Rešenje je implementirano na operativnom sistemu Linuks, u programskim jezicima C i Java i prenešeno na Android operativni sistem.

2. Teorijske osnove

Ovo poglavlje sadrži pregled osnovnih principa digitalne televizije, osnove UPnP i DLNA protokola, kratak sadržaj o Android operativnom sistemu, Google TV arhitekturi i o aplikaciji za pretragu, Google Quick Search Box (QSB).

2.1 Digitalna televizija

Digitalna televizija predstavlja sasvim novu tehnologiju u emitovanju radio i televizijskog programa. Ono što je glavna ideja digitalne televizije jeste da se velika količina podataka u digitalnom obliku transportuje čime se povećava iskorišćenje frekvencijskog spektra i proširuju se mogućnosti koje pruža televizija. Digitalni prenos obezbeđuje bolji kvalitet slike i zvuka bez šumova i smetnji.

U 90-im godinama razvijen je standard za digitalnu televiziju od koncepta do gotovog proizvoda. Danas već zbog brzog napretka telekomunikacija i interneta, digitalna televizija je uspela da to sjedini i u velikoj meri iskoristi. Tako danas digitalna televizija ima interaktivne aplikacije, pristup internetu, opciju snimanja omiljenih emisija itd.

Connected TV uređaji (često danas nazivani Smart TV uređajima) predstavljaju DTV prijemnike i STB (set-top box) koji imaju mogućnost pristupa internetu korišćenjem WiFi i Ethernet sprega.

Omogućavaju korišćenje OTT (Over the top) sevisa poput:

- TV sadržaja koji se doprema posredstvom interneta (stream ili download) - Internet TV
- Socijalnih internet servisa
- Web servisa opšte namene

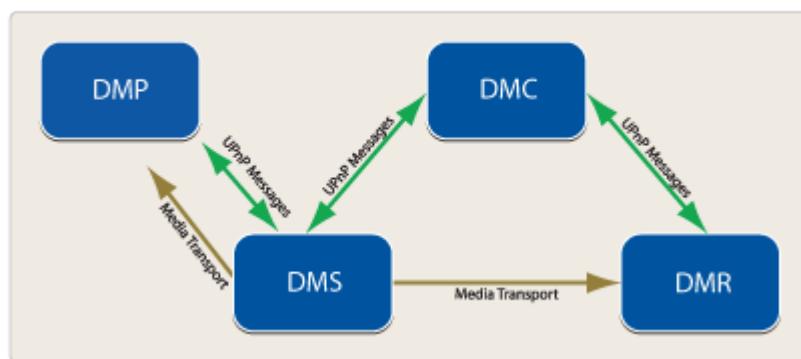
Connected TV platforme ili srednji sloj (eng. Middleware) imaju javni komplet za razvoj softvera (SDK) i/ili komplet za razvoj native softvera (NDK) tako da bilo koji programer može da razvija softver za tu platformu. Takođe, poseduje i prodavnici aplikacija(eng. app store) pomoću koje krajnji korisnik može da instalira aplikacije.

2.2 UPnP protokol

UPnP (Universal Plug and Play) je skup mrežnih protokola koji omogućavaju umreženim uređajima kao što su računari, štampači, bežične pristupne tačke, mobilni uređaji da bez problema otkriju prisustvo drugih uređaja u mreži i da uspostave mrežne usluge za deljenje podataka, komunikaciju i zabavu. Ovaj protokol je definisan od strane UPnP forum grupacije koju sačinjava preko 800 članova među kojima su i najveće svetske korporacije Microsoft, Nokia, Philips, Samsung, Intel, LG, iWedia. Namjenjen je pre svega lokalnim mrežama bez administratora. Nastao je iz PnP (Plug and Play) koji služi za povezivanje uređaja na računar.

UPnP podržava nula konfiguraciju što znači da korisnik ne mora ništa ručno da konfiguriše. Ovim se uređajima omogućava dinamički pristup mreži, dobijanje IP adrese i stavljanje na raspolaganje svoje sposobnosti drugima u mreži. Uređaj može napustiti mrežu u svakom trenutku ili automatski ako se više ne koristi. UPnP tehnologija je nezavisna od operativnog sistema i programskog jezika i izgrađena je na internet baziranim tehnologijama kao što su IP, TCP, UDP, HTTP i XML.

Mane UPnP protokola su uglavnom u smislu zaštite od zloupotrebe jer ovaj protokol nema implementiran sistem autentikacije.



Slika 1. Primer komunikacije korišćenjem UPnP

2.3 DLNA protokol

DLNA (Digital Living Network Alliance) je organizacija koja je definisala smernice za razmenu digitalnih multimedijalnih sadržaja između korisničkih uređaja kao što su računari, mobilni telefoni i kamere i ostalih multimedijalnih uređaja. DLNA je baziran na UPnP protokolu koga koristi za traženje, kontrolu i upravljanje multimedijom.

DLNA uređaji su podeljeni u tri klase kućne mrežne uređaje, ručne mobilne uređaje i kućne infrastrukturne uređaje.

2.3.1 Kućni mrežni uređaji

- Digital Media Server (DMS): čuvaju podatke i čine ih dostupnima na mreži digital media player-ima (DMP) i digital media renderer-ima (DMR). Primer DMS-a su personalni računari i network-attached storage (NAS) uređaji.
- Digital Media Player (DMP): pretražuju sadržaj digital media servers-a (DMS) i pružaju reprodukciju i prikaz datog sadržaja. Primer DMP-a su televizori, kućni bioskopi, igračke konzole i slično.
- Digital Media Renderer (DMR): vrši reprodukciju matrijala primljenog od digital media controller-a (DMC), čiji će sadržaj naći na digital media server-u (DMS). Primeri uključuju televizore, audio / video prijemnike, zvučnike za muziku i slično. Važno je napomenuti mogućnost da jedan uređaj funkcioniše kao DMP i DMR.
- Digital Media Controller (DMC): pronalazi sadržaj DMS-a i daje instrukcije DMR da prikaže taj sadržaj. Sadržaj se ne strimuje od niti kroz DMC. Primeri DMC-a su Internet tableti, Wi-Fi digitalne kamere i "pametni" telefoni.
- Digital Media Printer (DMPr): DLNA kućnoj mreži pruža usluge štampanja.

2.3.2 Ručni mobilni uređaji

- Mobile Digital Media Server (M-DMS): čuvaju podatke i čine ih dostupnima na žičanoj/bežičnoj mreži mobilnim media plejerima (M-DMP), media rendererima (DMR) i štampačima (DMPr). Primer je mobilni telefon ili prenosni muzički sistem.
- Mobile Digital Media Player (M-DMP): pretražuju sadržaj media servera (DMS) ili mobilnih media servera (M-DMS). Npr. mobilni telefon prilagođen za reprodukovanje multimedijalnog materijala.
- Mobile Digital Media Uploader (M-DMU): šalje (elg. upload) materijala na media server (DMS) ili mobilni media server (M-DMS). Npr. digitalne kamere ili mobilni telefoni.

- Mobile Digital Media Downloader (M-DMD): pronalazi i prihvata (eng. download) materijal sa media servera (DMS) ili mobilnog media servera (M-DMS). Npr. prenosni muzički sistemi ili mobilni telefoni.
- Mobile Digital Media Controller (M-DMC): pronalazi sadržaj media servera (DMS) ili mobilnog media server (M-DMS) i šalje ga na media renderer (DMR). Primer su mobilni telefoni.

2.3.3 Kućni infrastrukturni uređaji

- Mobile Network Connectivity Function (M-NCF): pruža usluge premošćavanja između kućnih mrežnih uređaja i ručnih mobilnih uređaja.
- Media Interoperability Unit (MIU): pružaju usluge transformacije multimedijalnog sadržaja u format koji podržavaju DLNA uređaji.

2.3.4 Formati multimedijalnog sadržaja i transportni model

U sladećim tabelama dati su formati multimedijalnog sadržaja koji DLNA uređaji moraju da podržavaju i oni koji su opcioni:

Vrsta mm sadržaja	Podrazumevani formati	Opcioni formati
Slike	JPEG	GIF, TIFF, PNG
Audio	LPCM (2 channel)	MP3, WMA9, AC-3, AAC, ATRAC3plus
Video	MPEG2	MPEG1, MPEG4, WMV9

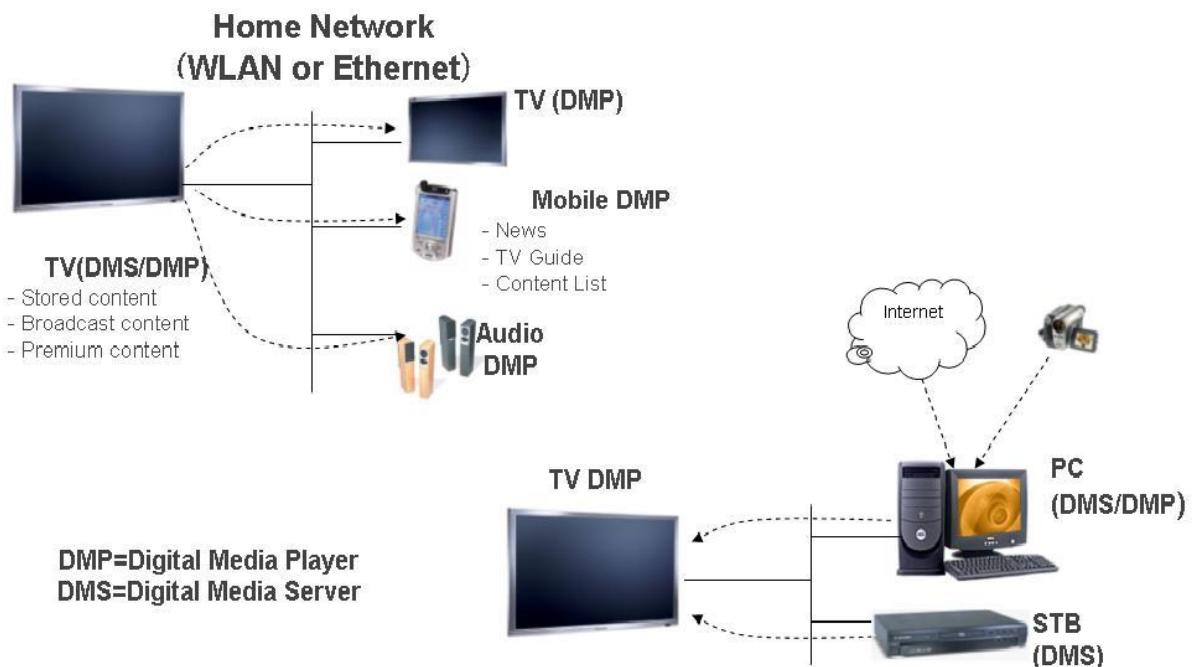
Tabela 1. Formati multimedijalnog sadržaja kućnih DLNA uređaja

Vrsta mm sadržaja	Podrazumevani formati	Opcioni formati
Slike	JPEG	GIF, TIFF, PNG
Audio	MP3 and MPEG4 AAC LC	MPEG4 (HE AAC, AAC LTP, BSAC), AMR, ATRAC3plus, G.726, WMA, LPCM
Video	MPEG4 AVC (AAC LC Assoc Audio)	VC1, H.263, MPEG4 part 2, MPEG2, MPEG4 AVC (BSAC ili drugi za Assoc. Audio)

Tabela 2. Formati multimedijalnog sadržaja mobilnih DLNA uređaja

Da bi se izbegao problem zagruženja mreže, komunikacija između DMP-a i DMS-a je takva da DMP prosleđuje parametre koje on podržava (veličina, format, MIME tip multimedijalnog materijala i sl.), a DMS mu prosleđuje URI koji najbolje odgovara njegovim zahtevima. To znači da server transkoduje multimedijalni materijal u format koji je optimalan za reprodukciju na DMP-u.

Najčešće korišćeni uređaji i tipičan izgled DLNA mreže prikazan je na slici 2.



Slika 2. Primer DLNA mreže

2.4 Android operativni sistem

Android je prva otvorena, kompletna i besplatna softverska platforma kreirana za prenosive uređaje razvijena od strane The Open Handset Alliance (grupa od preko 30 kompanija predvođenih Google kompanijom).

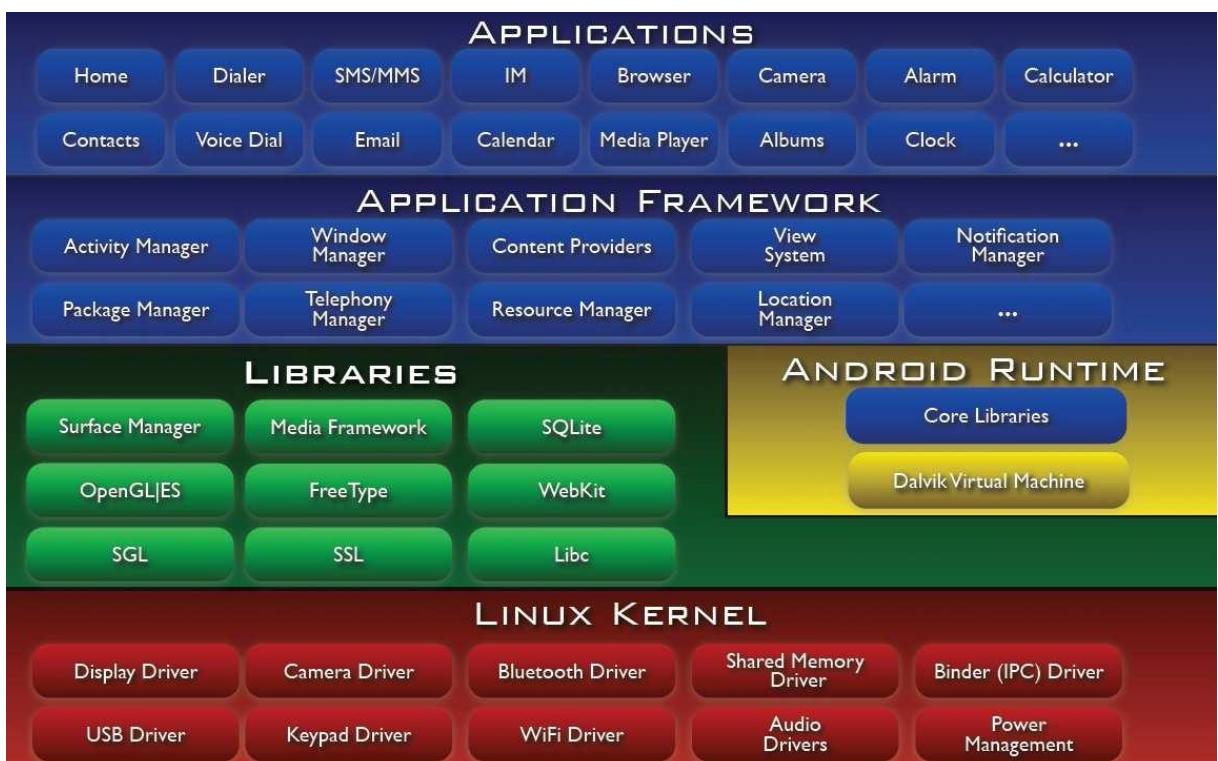
Trenutno je Android najzastupljeniji na prenosivim uređajima poput mobilnih telefona i tablet računara, ali ima potencijal da se proširi na uređaje gde god su veličina memorije i brzina procesora ograničeni.

Android je platforma koja se zasniva na Linuks kernelu verzije 3.x (sve do verzije 4.0) Android-a, koristio se kernel verzije 2.6) sa posrednim softverom (eng. middleware, MW), bibliotekama i API-jima, zatim aplikacionog softvera koji se oslanja na aplikacioni radni okvir (eng. framework). Iako su C i C++ programski jezici korišteni za radni okvir, većina aplikacija pisana je u Java programskom jeziku koristeći Android Software Development Kit (eng .SDK). On u sebi sadrži biblioteke za interakciju izmedju komponenata (eng. API) što često u praksi specificira rutine, strukture podataka, objektne klase i promenljive, zatim razvojne alate za izgradnju aplikacija (eng. Build) i otklanjanje grešaka (eng. Debug). Uz njega dobijate Eclipse razvojno okruženje, Android razvojne alate, podršku za različite Android platforme i emulator za testiranje razvijenih aplikacija.

Postoji mogućnost pisanja aplikacija i u C/C++ programskom jeziku, no tada se koristi Android Native Code Development Kit (NDK) koji je androidov alat za razvoj aplikacija u

nativnom delu operatining sistema. Ovakvim postupkom omogućuje se bolje raspolaganje resursima i korišćenje biblioteka programa iz kernela i radnog okruženja.

Razlog zašto je baš Linuks uzet za osnovu Androida je to što je već dokazan na tržištu kao stabilan operativan sistem koji se stalno unapređuje. Odlično kontroliše memoriju i upravlja procesima. Povrh svega koristi se zato što je Linuks izvorni kod besplatan i dostupan javnosti (eng. Open source). Na slici 3 je prikazan arhitektura Android operativnog sistema.



Slika 3. Arhitektura Android operativnog sistema

Osnovni način komunikacije između procesa je pomoću Binder-a koji je drajver jezgra operativnog sistema. Binder je bezbedan način komunikacije i prenosa primitivnih podataka između različitih nivoa. Za ove potrebe Android ima rešenje koje se zove AIDL. AIDL je jezik koji je zasnovan logici uprošćavanja kompleksnih struktura na primitivne strukture koje su potrebne Binder-u. Drugi način komunikacije između JAVA i nativnog dela je JAVA nativni interfejs (eng. JNI). Za razmenu podataka na JAVA nivou koristi se Intent koji je najviša abstrakcija Binder-a.

2.4.1 GoogleTV

GoogleTV je proširenje za Android platformu koje podržava DVB funkcionalnosti kao što su skeniranje servisa, kontrolisanje servisa, EPG, PVR, teletekst itd.

Softverska arhitektura GoogleTV-a se bazira na servisu koji pruža osnovne DVB funkcionalnosti (MediaDeviceService) i na mnogobrojnim televizijskim kontrolerima.

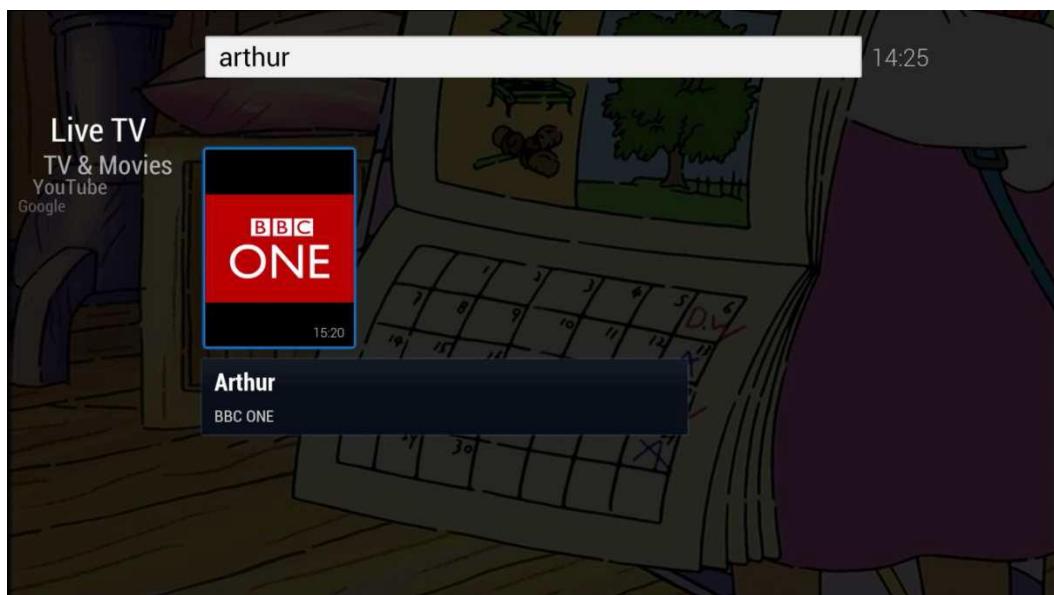
Sa stanovišta korisničkog interfejsa, razlike Google TV-a u odnodu na Android su sledeće:

- Google TV ne poseduje dodirni ekran(eng. touchscreen), nego se koriste razne verzije daljinskih upravljača koje porazumevaju Dpad, touchpad pa i „qwerty” tastaturu. Dodatna opcija je korišćenje Google TV Remote second-screen aplikacije koja mobilini ili tablet uređaj pretvara u daljinski uređaj. Ova komunikacija se ostvaruje korišćenjem Anymote protokola.
- Ne poseduju svi uređaji svoj displej, nego koriste panele različitih rezulucija kome trebaju da se prilagode. S obzirom da se koriste sa veće daljine (za razliku od mobilnih i tablet uređaja), elementi korisničkog interfejsa treba da budu dovoljno veliki da bi bili vidljivi.

Za razvoj aplikacija koristi se Android SDK koji sadrži alate i API korišćenjem programskog jezika JAVA. Sve do Google TV verzije 4 (koji je baziran na Android JB 4.2.x) nije postojala podrška za NDK. Predhodne verzije su koristile glibc biblioteku, a od verzije 4 se koristi bionic. To dovodi do toga da Google TV sada podržava veliku većinu aplikacija sa Google Play-a.

2.4.2 Google Quick Search Box

QSB je aplikacija dostupna od Android verzije 1.6. Ona predstavlja radni okvir koji služi za pretragu podataka aplikacija i interneta. Da bi aplikacija bila pretraživa mora da ima implementiran Content Provider (više o njegovom funkcionisanju biće u nastavku teksta) i sva razmena podataka se vrši preko njega. Podaci se razmenjuju preko Intent poruka.



Slika 4. Izgled Google TV QuickSearchBox aplikacije

3. Koncept rešenja

Integracija DMP-a u Android operativni sistem zahteva postojanje više slojeva programske podrške. Svaki od slojeva predstavlja prilagođen servis za druge slojeve.

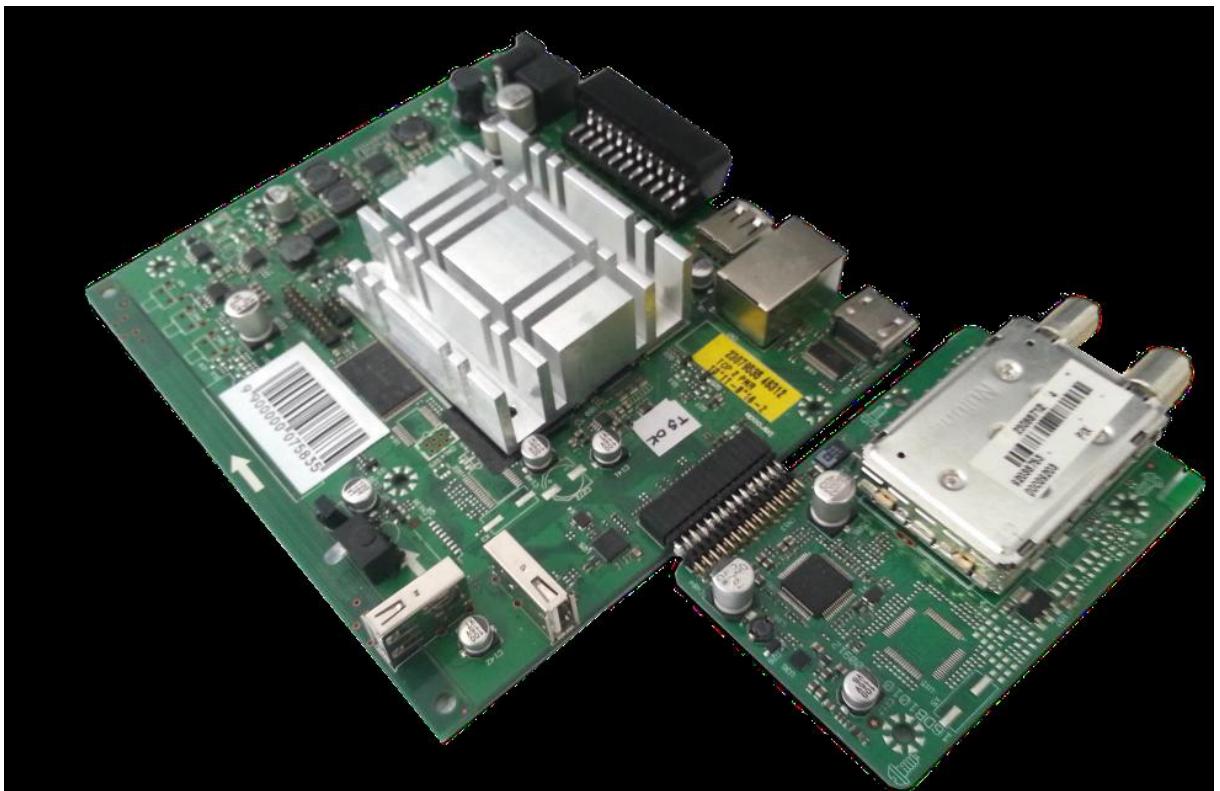
Na najvišem aplikativnom nivou se nalazi Android aplikacija koja omogućuje pretragu i reprodukciju slika, audio i video sadržaja koji se nalaze na mrežnom poslužiocu.

Sledeći srednji sloj je Java Native Interface (JNI) sloj koji omogućuje komunikaciju između aplikativnog dela pisanog u Java programskom jeziku i izvornog koda pisanog u programskom jeziku C.

Poslednji nivo je izvorni C kod u kome su implementirane biblioteke libdlna, libupnp koje omogućavaju puno korišćenje DLNA i UPnP resursa.

3.1 Opis ciljne platforme

Rešenje koje je prikazano u ovom radu realizovano je na RK-2010 DVB-T2 prijemniku digitalnog TV signala. RK-2010 platforma raspolaže sledećim ključnim komponentama: zemaljski birač kanala tipa NuTune FP3114, procesor ARM familije Marvell BG2 88DE3100, 1GB DDR3 operativne memorije, 8GB NAND memorije, 1 HDMI izlaz, 1 SCART izlaz, 1S/PDIF izlaz, 1 Ethernet port i 3 USB priključka. Na platformi je instaliran 4.2.x Ice Cream Sandwich Android operativni sistem. Upravo zbog ovih karakteristika ova platforma je izabrana kao ciljna platforma jednog prijemnika digitalnog televizijskog signala koji je zasnovan na Android operativnom sistemu.



Slika 5. RK-2010 platforma

3.2 Realizacija Android aplikacije za pretragu DLNA

Sledeća poglavља opisuju module iz kojih se aplikacija sastoji.

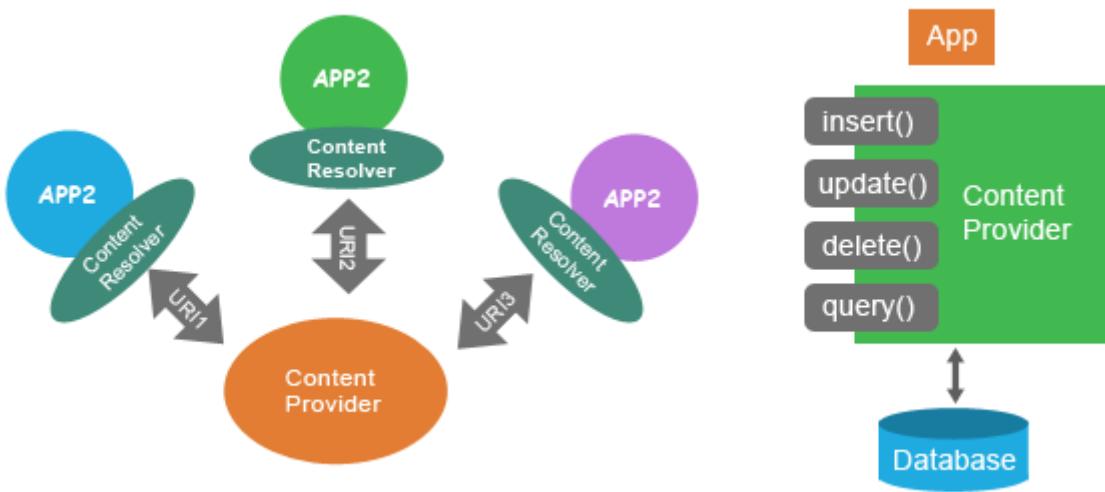
3.2.1 Modul rukovaoca bazom podataka

Modul rukovaoca bazom podataka ima osnovne tzv. CRUD (kreiraj, čitaj, ažuriraj, obriši) funkcionalnosti za rad sa bazom podataka.

Pri pokretanju aplikacije, inicijalizuje se baza podataka i priprema za pisanje i čitanje podataka.

3.2.2 Modul za dobavljanje sadržaja

Baza podataka u androidu je privatna i dostupna je samo aplikaciji koja ju je kreirala. Da bi različite aplikacije mogle pristupati jednoj privatnoj bazi podataka, u Androidu se koriste dobavljači sadržaja (eng. Content Provider). Konkretna primena ovog modula je kada QSB aplikacija prosledi upit modulu, na koji će modul da pretraži bazu podataka i vrati rezultate QSB aplikaciji.



Slika 6. Pristup bazi podataka preko Content Provider

3.2.3 Modul za pronalaženje DLNA servera i pretragu njihovog sadržaja

Modul poziva funkcije iz libupnp i libdlna biblioteka preko JNI sloja. Biblioteka libupnp omogućava povezivanje multimedijalnih uređaja unutar kućne mreže dok biblioteka libdlna omogućava deljenje multimedijalnog sadržaja između uređaja unutar kućne mreže.

3.2.4 Moduli za prikaz DLNA sadržaja

Ovaj modul zadužen je za prikaz slika ili reprodukciju audio/video materijala. To se postiže tako što DMS prosledi DMP-u URI na osnovu kojega DMP strimuje materijal.

4. Programsко rešenje

U ovom poglavlju opisano je programsko rešenje implementacije DLNA podrške u okviru GoogleTV platforme. Sastoji se od građenja dobijenog izvornog C koda za Android operativni sistem, realizacije aplikacije za pretragu raspoloživosti DLNA materijala, te njihov prikaz krajnjem korisniku. Aplikacija je realizovana u JAVA programskom jeziku.

4.1 Građenje izvornog koda

Za građenje izvornog C koda za rad na Android operativnom sistemu potrebno je po konvenciji napraviti datoteku sa ekstenzijom “.mk”. Glavna datoteka se uvek naziva Android.mk, i ona se nalazi na vrhu izvornog stabla. Sledi primer Android.mk, na vrhu izvornog stabla, sa kojim gradimo deljenu biblioteku u kojoj se nalaze funkcije koje preko JNI sloja koristimo u aplikaciji:

```
LOCAL_PATH := $(call my-dir)
#####
#      d l n a      #
#####
include $(CLEAR_VARS)
include $(LOCAL_PATH)/src/common/common.mk

# DMC and DMP
include $(LOCAL_PATH)/src/dmc/dmc.mk

# DLNA JNI
include $(LOCAL_PATH)/src/java/java.mk

LOCAL_SRC_FILES := $(LIB_SRCS)
```

```

LOCAL_UPNP := $(LOCAL_PATH)/../libupnp-1.6.12
LOCAL_UTIL := $(LOCAL_PATH)/../libiwediautil

LOCAL_CFLAGS += $(C_FLAGS)
LOCAL_CFLAGS += -I$(LOCAL_PATH)/src/common -I$(LOCAL_PATH)/src/dmc
LOCAL_CFLAGS += -DANDROID_UPNP_BUILD -DANDROID_LIBDLNA_BUILD

LOCAL_C_INCLUDES := $(LOCAL_UPNP) \
                    $(LOCAL_UPNP)/ixml/inc \
                    $(LOCAL_UPNP)/threadutil/inc \
                    $(LOCAL_UPNP)/upnp/inc \
                    $(LOCAL_UTIL)/src

LOCAL_MODULE_TAGS := optional
LOCAL_MODULE := libdlna_jni
LOCAL_PRELINK_MODULE := false

LOCAL_SHARED_LIBRARIES:= libiwediautil
LOCAL_WHOLE_STATIC_LIBRARIES += libupnp libixml libthreadutil

include $(BUILD_SHARED_LIBRARY)

#build JAR file
include $(LOCAL_PATH)/src/java/src/Android.mk

```

4.2 Programsko rešenje Android aplikacije za pretragu DLNA servera

U sledećim poglavljima biće prikazano programsko rešenje za module ove Java aplikacije.

4.2.1 Podešavanje AndroidManifest.xml

Da bi korisnik mogao da pristupi podacima van aplikacije, potrebno je da uradi sledeća podešavanja u AndroidManifest.xml

Pod tagom <activity> potrebno je dodati intent filter i referencu na searchable.xml:

```

<intent-filter>
    <action android:name="android.intent.action.SEARCH" />
    <category android:name="android.intent.category.DEFAULT" />
</intent-filter>

<meta-data android:name="android.app.searchable"
          android:resource="@xml/searchable" />

```

Zatim, potrebno je dodati dobavljač sadržaja (engl. content provider) koji će da pruža usluge pretrage baze podataka:

```
<provider android:name="DictionaryProvider"
          android:authorities="dictionary"
          android:syncable="false" />
```

Primer fajla Searchable.xml dat je u nastavku teksta:

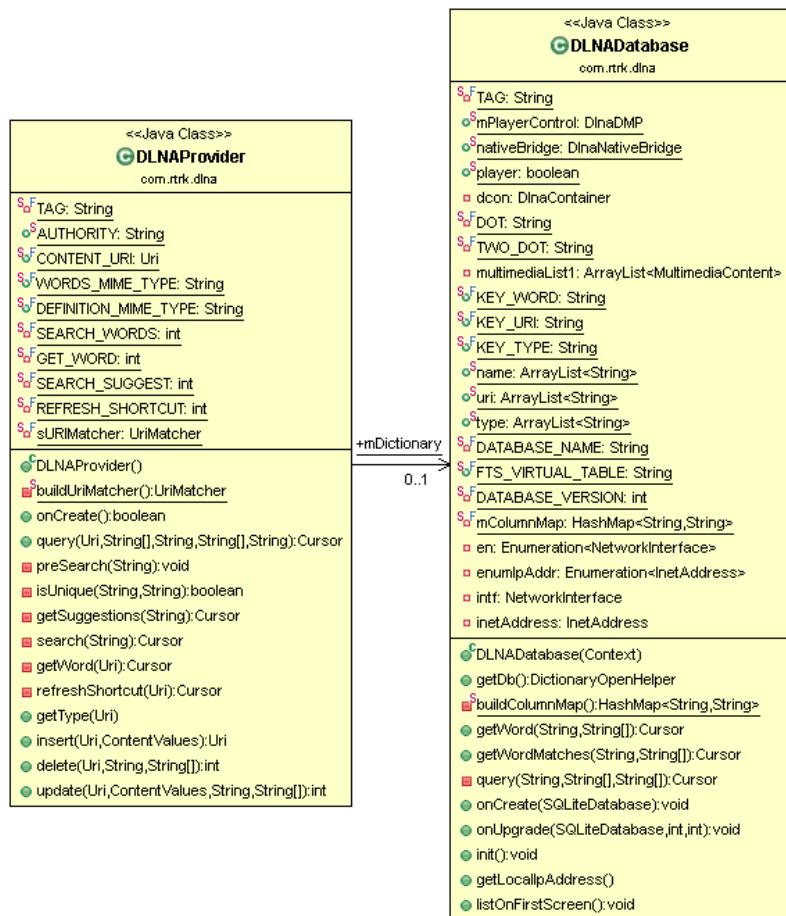
```
<searchable xmlns:android="http://schemas.android.com/apk/res/android"
            android:label="@string/search_label"
            android:searchSettingsDescription="@string/settings_description"
            android:includeInGlobalSearch="true"
            android:searchSuggestAuthority="dictionary"
            android:searchSuggestIntentAction="android.intent.action.VIEW">
</searchable>
```

4.2.2 DLNADatabase.java

Nasleđuje klasu SQLiteOpenHelper i koristi se za rad sa bazom podataka.

4.2.3 DLNAProvider.java

Nasleđuje klasu ContentProvider i ima ulogu posrednika između aplikacije i baze podataka.



Slika 7. Klase DLNADatabase i DLNAProvider

4.2.4 MainActivity.java

Programski modul zadužen za inicijalizaciju DLNA plejera, pretragu i prikaz dostupnih DLNA servera krajnjem korisniku.

4.2.5 VideoViewActivity.java

Prvo je potrebno setovati plejer, zatim dati lokaciju u formi URL-a gde se nalazi multimedijalni sadržaj, pripremiti i startovati reprodukciju. Za prekid reprodukcije potrebno je prvo stopirati, a zatim i oslobođiti plejer.



Slika 8. Klasa VideoViewActivity

4.2.6 PictureViewActivity.java

Koristi se za prikaz slika strimovanih sa DMS-a. Mala količina virtualne memorije predstavlja problem kod prikaza slika visokog kvaliteta. Zato se serveru šalju dva zahteva.

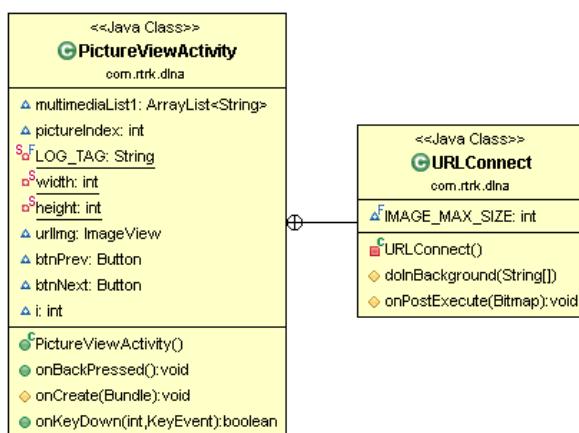
Prvi zahtev se šalje sa poljem inJustDecodeBounds klase BitmapFactory setovanim na true što će da setuje polja outHeight i outWidth na veličinom slike koju treba preuzeti, ali memorija neće biti alocirana:

```
BitmapFactory.Options o = new BitmapFactory.Options();
o.inJustDecodeBounds = true;
o.inPurgeable = true;
BitmapFactory.decodeStream(input, null, o);
```

Na osnovu dobijenih parametara slike računa se optimalan nivo skaliranja i prosledjuje se polju inSampleSize, nakon čega se slika strimuje i dekoduje:

```
int scale = 1;
if (o.outHeight > IMAGE_MAX_SIZE || o.outWidth > IMAGE_MAX_SIZE) {
    scale=(int) Math.pow(2,(int) Math.round(Math.log(IMAGE_MAX_SIZE
        / (double) Math.max(o.outHeight,
        o.outWidth))/Math.log(0.5)));}

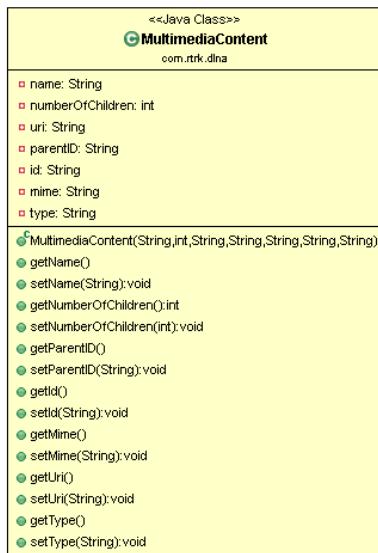
BitmapFactory.Options o2 = new BitmapFactory.Options();
o2.inSampleSize = scale;
o2.inPurgeable = true;
Bitmap b = BitmapFactory.decodeStream(input, null, o2);
```



Slika 9. Klasa PictureViewActivity

4.2.7 MultimediaContent.java

Klasa koja služi za čuvanje podataka vezanih za DLNA objekat



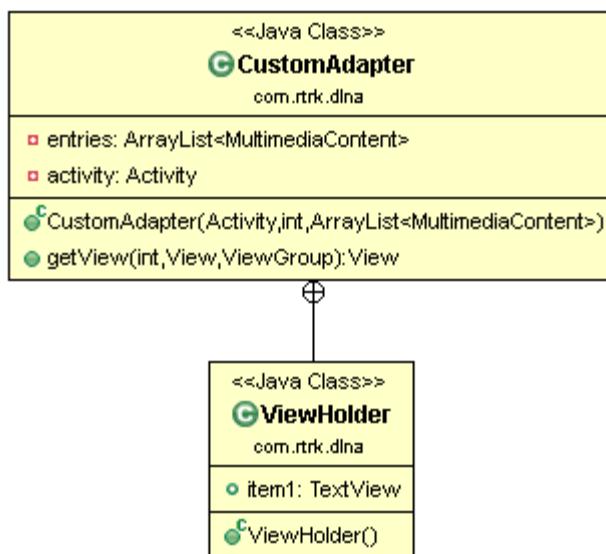
Slika 10. Klasa MultimediaContent

4.2.8 BrowseDialog.java

Nasleđen od klase Dialog i prilagođen da prikazuje listu imena direktorijuma i fajlova tekućeg direktorijuma.

4.2.9 CustomAdapter.java

Koristi se za prikaz polja NAME klase MultimediaContent u listi servera MainActivity-ja i listi foldera koji se prikazuju u BrowseDialog-u

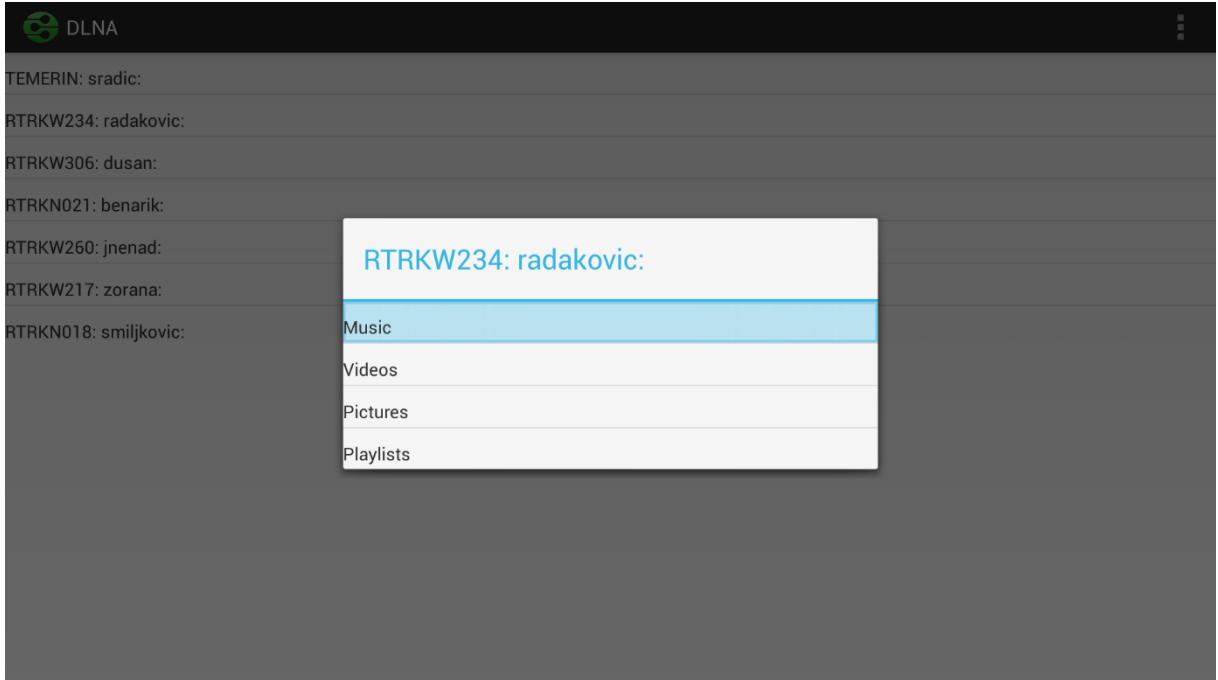


Slika 11. Klasa CustomAdapter

5. Rezultati

Za potrebe testiranja i verifikacije ovog programskog rešenja korišćen je uređaj na kojem se nalazi Android OS verzije 4.2.2.

Prilikom pokretanja aplikacije prikazuje se lista dostupnih DLNA servera prikazanih na slici 6. , dok se za to vreme pretražuju svi dostupni DLNA serveri lokalne mreže i njihov multimedijalni sadržaj dodaje u bazu podataka aplikacije.



Slika 12. Koreni folder odabranog servera i lista DLNA servera (u pozadini)

Odabirom željenog DLNA servera otvara se njegov koreni direktorijum. Odavde je moguće odabrati koji će se multimedijalni materijal reprodukovati.

Za odabir slika trenutnog foldera mogu se koristiti DPad tasteri tastature ili daljinskog upravljača.



Slika 13. Prikaz fotografija

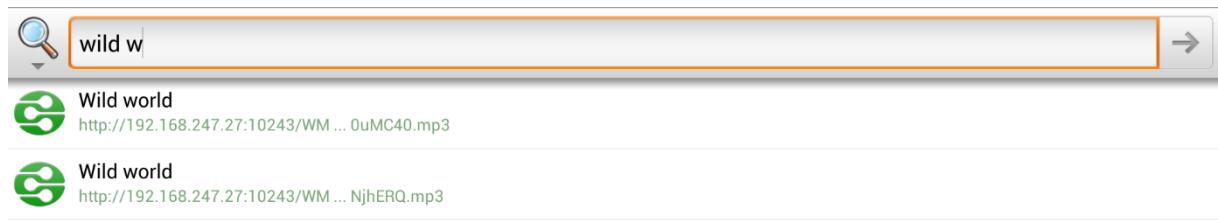
Media plejer pruža standardne komande reprodukcije videa koji su prikazani na slici 8.



Slika 14. Reprodukovanje audio i video fajlova

Da bi se izbegla ručna pretraga traženog fajla na serveru moguće je koristiti QSB. Test je vršen tako što su u polje za pretragu QSB aplikacije unošena imena fajlova koji postoje na serverima što je vraćalo listu svih pronađenih fajlova. Ukoliko na serveru ne postoji fajl sa traženim imenom, QSB aplikacija ne vraća nikakve rezultate korisniku kako je i predviđeno.

Na sledećim slikama možemo da vidimo primere testiranja pretrage DLNA sadržaja pomoću QSB aplikacije.



Slika 15. Primer pretrage korišćenjem QSB-a

6. Zaključak

DLNA protokol napravljen je sa ciljem da omogući razmenu multimedijalnog sadržaja između korisničkih uređaja koji pristupaju lokalnoj DLNA mreži. Ovaj protokol je zaživeo na tržištu i mnoge velike svetske kompanije u oblasti mobilnih telefona i DTV prijemnika, kao što su Nokia, Samsung Electronics, Sharp Corporation, na svojim uređajima imaju implementirane aplikacije za reprodukciju multimedijalnog sadržaja.

U ovom radu je opisano jedno rešenje realizacije sistema za reprodukciju multimedijalnog sadržaja u DLNA mreži na uređajima baziranim na Android operativnom sistemu.

Sistem se sastoji od Java aplikacije, JNI sloja i biblioteka u kojima se nalazi realizacija DLNA i UPnP protokola.

Rešenje je univerzalno i može se koristiti na mobilnim telefonima, tabletima i DTV prijemnicima.

Dalji razvoj bi mogao biti usmeren u smeru optimizacije pretrage u smislu da se pretraga vrši kada je operativni sistem najmanje zauzet čime bi se štedeli sistemski resursi odnosno da se pretraga filtrira prema tipu multimedijalnog sadržaja, što bi značajno smanjilo vreme pretrage.

7. Literatura

- [1] DLNA, Digital Living Network Alliance, www.dlna.org
- [2] UPnP Forum, Universal Plug and Play Forum, www.upnp.org
- [3] Android Developer, <http://developer.android.com/develop/index.html>
- [4] Ilić Mladen, Janković Miloš, Leporis Milko: ONE SOFTWARE SOLUTION FOR PLAYBACK MULTIMEDIA CONTENT USING DLNA PROTOCOL ON ANDROID OS; 20th Telecommunications Forum (TELFOR) Belgrade, Serbia, November, 20-22, 2012
- [5] Android TV Apps Development, Paul Trebilcox-Ruiz, ISBN: 978-1-4842-1784-9, 2016