



УНИВЕРЗИТЕТ У НОВОМ САДУ  
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У  
НОВОМ САДУ



Лара Миловановић

**Проширење TV *Input Framework*  
компоненте са функционалношћу за  
предлагање ТВ садржаја у Андроид О  
окружењу**

**ДИПЛОМСКИ РАД  
- Основне академске студије -**

Нови Сад, 2018



## 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:	Лара Миловановић		
Ментор, МН:	Проф. др Илија Башичевић		
Наслов рада, НР:	Проширење TV Input Framework компоненте са функционалношћу за предлагање ТВ садржаја у Андроид О окружењу		
Језик публикације, ЈП:	Српски / латиница		
Језик извода, ЈИ:	Српски		
Земља публиковања, ЗП:	Република Србија		
У же географско подручје, УГП:	Војводина		
Година, ГО:	2018		
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)	7/46/0/1/19/0/0		
Научна област, НО:	Електротехника и рачунарство		
Научна дисциплина, НД:	Рачунарска техника		
Предметна одредница/Кјучне речи, ПО:	Дигитална телевизија, Андроид, ДТВ сервис, Дигитални ТВ пријемник		
УДК			
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад		
Важна напомена, ВН:			
Извод, ИЗ:	<p>Почевши од Андроида верзије Орео, апликација почетног екрана (<i>Leanback Launcher</i>) развијена од стране Google-а је претрпела велике измене са циљем бољег представљања различитог мултимедијалног садржаја крајњим корисницима. Главна измена у односу на претходну верзију је што свака апликација може да јавно објави свој предложен садржај који ће кориснику бити доступан и лако уочљив на почетном екрану уређаја. У овом раду је реализована апликација која врши предлагање ТВ садржаја у апликацији почетног екрана. ТВ садржај који се предлаже кориснику је <i>HLS</i> ТВ канал који се емитује уживо. Предложен садржај, поред основних карактеристика садржи и:</p> <ul style="list-style-type: none"><li>• Видео који се користи са брз увид у садржај – „Preview Video“</li><li>• Додавање садржаја након прегледања у „Watch Next“ канал</li></ul>		
Датум прихватања теме, ДП:			
Датум одбране, ДО:			
Чланови комисије, КО:	Председник:	Проф. др Небојша Ђевалица	
	Члан:	Проф. др Иштван Пап	Потпис ментора
	Члан, ментор:	Проф. др Илија Башичевић	



## 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:	Lara Miovanović	
Mentor, MN:	Ilija Bašičević, PhD	
Title, TI:	Extension of <i>TV Input Framework</i> component to support TV content recommendations in Android O environment	
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:	2018	
Publisher, PB:	Author's reprint	
Publication place, PP:	Novi Sad, Dositeja Obradovica sq. 6	
Physical description, PD: (chapters/pages/ref./tables/pictures/graphs/appendices)	7/46/0/1/19/0/0	
Scientific field, SF:	Electrical Engineering	
Scientific discipline, SD:	Computer Engineering, Engineering of Computer Based Systems	
Subject/Key words, S/KW:	Digital television, Android, DTV service, STB device	
UC		
Holding data, HD:	The Library of Faculty of Technical Sciences, Novi Sad, Serbia	
Note, N:		
Abstract, AB:	<p>Starting with Oreo version of Android, home screen application (<i>Leanback Launcher</i>) developed by Google has undergone major changes with the aim of better presenting of multimedia content to the user. The main change compared to the previous version is that each application can publish its recommended content that will be available to the user and easily visible on the home screen. This paper represents the extension of <i>TV Input Framework</i> component, in order to recommend live TV content in the home screen application. TV content that is presented to user is live broadcasted <i>HLS</i> TV channel. Recommended content, besides its basic content, includes:</p> <ul style="list-style-type: none"><li>• A video that is used for quick inside into content – „Preview Video”</li><li>• Adding last watched content in the „Watch Next” channel</li></ul>	
Accepted by the Scientific Board on, ASB:		
Defended on, DE:		
Defended Board, DB:	President:	Nebojša Pjevalica, PhD
	Member:	Ištván Pap, PhD
	Member, Mentor:	Ilija Bašičević, PhD
		Menthor's sign

## **Zahvalnost**

Zahvaljujem se Institutu RT-RK na pruženoj prilici za realizaciju ovog rada.  
Takođe, zahvaljujem se mentoru dr Ilijii Bašičeviću i stručnom saradniku Sretenu Tanackoviću na savetima i pomoći tokom izrade ovog rada.

## SADRŽAJ

1.	Uvod.....	1
2.	Teorijske osnove .....	3
2.1	Digitalna televizija .....	3
2.2	Digitalni TV prijemnik.....	5
2.3	Android.....	6
2.3.1	Android platforma .....	6
2.3.2	Android Open Source Project.....	8
2.3.3	Android TV platforma .....	8
2.3.4	Android Aplikacije .....	9
2.4	Android programska podrška TV prijemnika – TIF .....	10
2.4.1	Komponente Android programske podrške za TV uređaje .....	10
2.4.2	TV snabdevač .....	11
2.4.3	TV ulazni rukovaoc .....	12
2.4.4	TV ulaz .....	13
2.4.5	TV aplikacija .....	14
2.5	Pokretač aplikacija (eng. Launcher).....	15
2.5.1	Funkcionalnost „Program Preview” .....	17
2.5.2	Funkcionalnost „Watch Next" .....	17
3.	Koncept rešenja.....	18
3.1	Komponente sistema .....	18
3.2	Opis ciljne platforme I ograničenja.....	19
3.3	Programska podrška STB uređaja .....	20

3.4	Proširenje programske podrške za „Program Preview“ i „Watch Next“ .....	22
4.	Programsko rešenje .....	24
4.1	Kreiranje kanala .....	25
4.2	Kreiranje Preview programa .....	26
4.3	Kreiranje WatchNext programa .....	27
5.	Ispitivanje i verifikacija .....	29
6.	Zaključak .....	35
7.	Literatura .....	36

## SPISAK SLIKA

Slika 2.1 – Mapa sveta po odabranom DTV standardu .....	4
Slika 2.2 – Način isporuke TV sadržaja.....	5
Slika 2.3 – Arhitektura Android programskog steka .....	7
Slika 2.4 – TIF arhitektura .....	11
Slika 2.5 – TV snabdevač.....	12
Slika 2.6 – TV ulaz treće strane .....	14
Slika 2.7 – Izgled početnog ekrana za verziju Android Nougat .....	16
Slika 2.8 – Izgled početnog ekrana za verziju Android Oreo .....	16
Slika 3.1 – Razvojna platforma.....	19
Slika 3.2 – Struktura programske podrške TV prijemnika .....	20
Slika 3.3 – Ručno biranje kanala koje nudi aplikacija.....	22
Slika 5.1 – Rezultati JUnit testova .....	29
Slika 5.2 – Odabir kanala iz aplikacije.....	31
Slika 5.3 – Kanal na početnom ekranu, popunjen programima .....	31
Slika 5.4 – Emitovanje TV sadržaja uživo, iz kanala na početnom ekranu .....	32
Slika 5.5 – „Watch Next“ kanal popunjen poslednjim programom koji je korisnik gledao	32
Slika 5.6 – Ručno dodavanje programa u „Watch Next“ kanal .....	33
Slika 5.7 – „Watch Next“ kanal popunjen programski dodatim i ručno dodatim programima .....	33
Slika 5.8 – Nakon gore pomenute radnje, u kanalu „Watch Next“ ostaje samo jedan kanal...	34

## **SPISAK TABELA**

Tabela 5.1 – Ispitane funkcionalnosti i rezultati ..... 30

## SKRAĆENICE

<b>DTV</b>	- <i>Digital television</i> , digitalna televizija
<b>TIF</b>	- <i>TV Input Framework</i> , Android programska podrška TV prijemnika
<b>STB</b>	- <i>Set-top box</i> , digitalni TV prijemnik
<b>AOSP</b>	- <i>Android Open Source Project</i> , Android projekat otvorenog koda
<b>GTVS</b>	- <i>Google TV Services</i> , Android TV aplikacije razvijene od strane Google-a
<b>EPG</b>	- <i>Electronic Program Guide</i> , elektronski programski vodič
<b>API</b>	- <i>Application programming interface</i> , Aplikativna programska sprega
<b>CTS</b>	- <i>Compatibility Test Suite</i> , skup testova kompatibilnosti
<b>JNI</b>	- <i>Java Native Interface</i> , Java programska sprega ka C/C++ programskim slojevima
<b>HAL</b>	- <i>Hardware Abstraction Layer</i> , sloj za apstrakciju hardvera
<b>HTTP</b>	- <i>Hypertext Transfer Protocol</i> , protokol za prenos hiperteksta
<b>URL</b>	- <i>Uniform Resource Locator</i> , jedinstveni lokator resursa
<b>HLS</b>	- <i>HTTP Live Streaming</i> , protokol za prenos multimedijalnog sadržaja koji se oslanja na HTTP protokol
<b>DVR</b>	- <i>Digital Video Recorder</i> , digitalni snimač video sadržaja

## 1. Uvod

U ovom radu je realizovano proširenje TV Input Framework [1] komponente sa funkcionalnošću za predlaganje TV sadržaja na Android Oreo TV platformi [2].

Počevši od Android-a verzije Oreo, aplikacija početnog ekrana (eng. Launcher) razvijena od strane kompanije Google je pretrpela velike izmene sa ciljem boljeg predstavljanja različitog multimedijalnog sadržaja krajnjim korisnicima. Glavna izmena u odnosu na prethodnu verziju je što svaka aplikacija može da javno objavi svoj predložen sadržaj koji će korisniku biti dostupan i lako uočljiv na početnom ekranu uređaja. Ova aplikacija pruža niz različitih svojstava kako bi se postigla što veća moguća fleksibilnost prilikom objavljivanja sadržaja. Neke od tih svojstava su naziv, opis, slika, grupe, kategorije, i mnoga druga svojstva. Jedni od najbitnijih svojstava koja su bitne za ovaj rad su: kratak video prikaz (Video Preview) [3] i beleženje prethodno gledanog sadržaja (Watch next) [4].

Cilj realizacije ovog rada je napisati aplikaciju koja će da vrši predlaganje TV sadržaja u aplikaciji početnog ekrana. TV sadržaj koji se predlaže korisniku je HLS TV kanal koji se emituje uživo. Predložen sadržaj pored osnovnih karakteristika kao što su naziv, opis i slično, sadrži i:

- Video koji se koristi za brz uvid u sadržaj – „Preview Video“
- Mogućnost da se sadržaj nakon pregledanja doda na listu za nastavak gledanja – „Watch Next“

Za izradu zadatka korišćena je prethodno data TV Input aplikacija koja u sebi ima mogućnost dobavljanja i reprodukcije TV sadržaja i dalje je proširena sa prethodno navedenim funkcionalnostima.

Ovaj rad je sačinjen od 7 poglavlja.

Prvo poglavljje sadrži osnovne informacije o radu i kratak opis rada.

U drugom poglavju date su teorijeske osnove koje su neophodne za razumevanje načina funkcionisanja digitalne televizije, Android operativnog sistema koji je proširen podrškom za reprodukovanje televizijskog sadržaja, Android programska podrška za TV uređaje (eng. Tv Input Framework – TIF), kao i proširenje programske podrške za „Program Preview“ i „Watch Next“ funkcionalnosti.

Treće poglavje sadrži opis arhitekture celog sistema, kao i pregled slojeva sistema koji opisuju predlog rešenja, realizacije i prikaza implementiranih modula.

U četvrtom poglavljtu govori se o samoj implementaciji i povezivanju svih modula potrebnih za ispravan rad „Program Preview“ i „Watch Next“ mehanizama.

U petom poglavljtu su opisani načini ispitivanja i verifikacije realizovanih modula.

Šesto poglavje sadrži kratak pregled onoga što je urađeno u ovom radu, kao i predlog na čemu bi se mogao zasnovati dalji rad.

U sedmom poglavljtu je dat spisak literature koja je korišćena prilikom izrade ovog rada.

## 2. Teorijske osnove

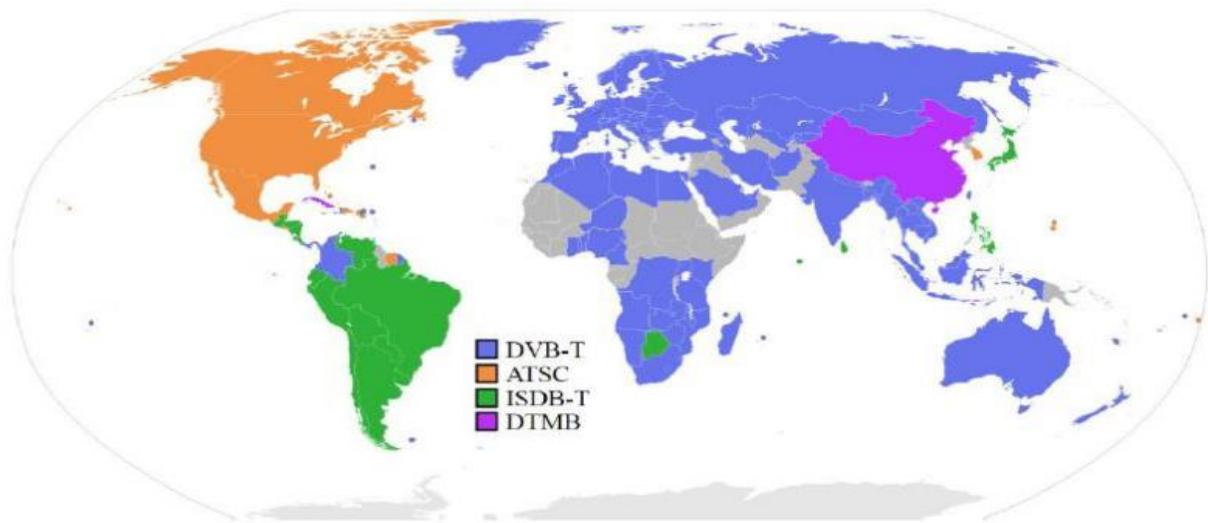
U ovom poglavlju dat je koncept digitalne televizije, opis i namena Android platforme, opis programske podrške za digitalnu televiziju koja koristi Android platformu i opis koncepta i funkcionalnosti ProgramPreview-a i WatchNext-a.

### 2.1 Digitalna televizija

Digitalna televizija obuhvata prenos televizijskog signala od emitera do gledaoca digitalnim putem, korišćenjem digitalnih modulacionih tehnika, u obliku digitalnog prenosnog toka u okviru kojeg se vremenski multipleksirano prenose paketi vezani za audio i video sadržaje, meta-podatke i dodatne sadržaje koji se odnosne na više servisa (programa) [5].

S obzirom na fleksibilnost podržanih formata, mogućnost vremenskog multipleksiranja i organizacije paketa prenosnog toka, sadržaj digitalnog signala koji se prenosi u jednom telekomunikacionom kanalu može imati fleksibilnu strukturu koja prenosi širok spektar komponenti. Sve ovo dovodi do povećanja kvaliteta i raznovrsnosti usluga. Usluge koje su tipične za digitalnu televiziju uključuju:

- izbor audio jezika i jezika za prevod
- elektronski vodič kroz program sa podsetnicima
- roditeljsku kontrolu
- digitalni teletekst i multimedijalne aplikacione portale
- interaktivnu televiziju
- pristup internetu
- prenos i izvršavanje aplikacija najšireg skupa funkcionalnosti



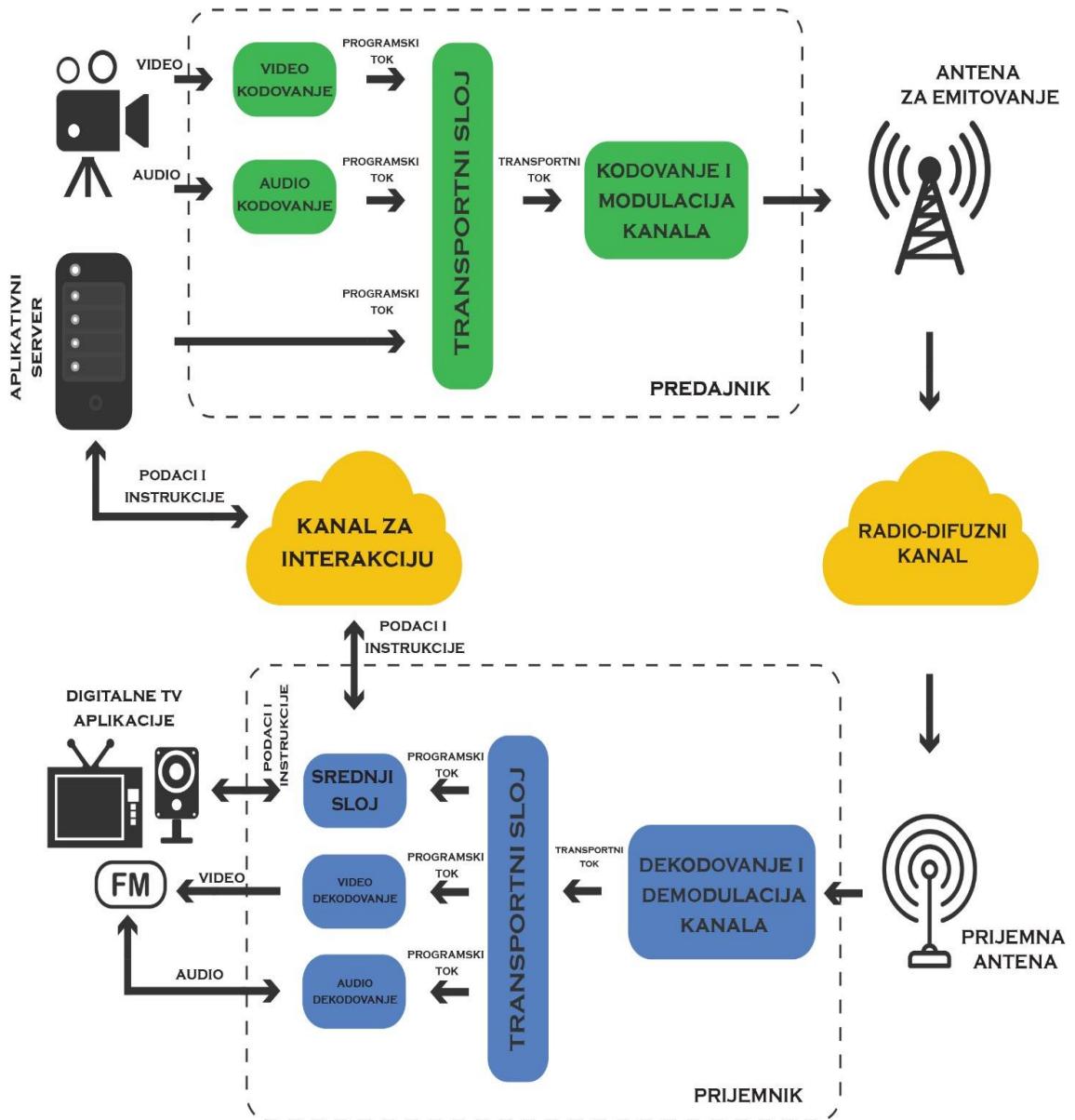
Slika 2.1 – Mapa sveta po odabranom DTV standardu

Najznačajniji standardi digitalne televizije su:

- DVB – grupa standarda koja se koristi u Evropi ali i u vecem delu sveta
- ATSC – standardi koji se primenjuju u digitalnom TV prenosu zemaljskim vezama u SAD, Kanadi, Meksiku, Juznoj Koreji
- OCAP – standard koji se koristi u SAD u kablovskoj digitalnoj televiziji
- ISDB – grupa DTV standarda nastala u Japanu
- DTMB – grupa DTV standarda nastala u Kini

Pored navedenih standarda koji se koriste u različitim delovima sveta, postoje i različiti načini isporuke TV sadržaja:

- Zemaljski (eng. Terrestrial)
- Satelitski (eng. Satellite)
- Kablovski (eng. Cable)
- Preko mreže – putem Internet-a (eng. IP):
  - Datagram:
    - Multicast
    - Unicast
  - Adaptivni protokoli:
    - HLS
    - DASH
    - Smooth Streaming



Slika 2.2 – Način isporuke TV sadržaja

## 2.2 Digitalni TV prijemnik

Digitalni televizijski prijemnik [6] je uređaj koji vrši konverziju televizijskog signala sa ciljem prikaza sadržaja na standardnim LCD (eng. Crystal Display), LED (eng. Light emitting diode) ili analognim televizorima. Prijemnik može biti ugrađen u TV uređaj ili odvojen uređaj tj. STB (eng. Set-Top Box – STB) [7].

Koncept STB se pojavljuje 60-ih godina 20. veka kao uređaj za prevodenje UHF (eng. Ultra High Frequency) frekvencija na VHF (eng. Very High Frequency) frekvencije. Istorijски, prvi digitalni STB uređaji su primali signal sa kablovske mreže. Uvedeni su da nadoknade nedostatke

izazvane razlikama između emitovanog i kablovskog signala [8]. Međutim, u digitalnoj eri uloga digitalnog STB uređaja nije samo prijem signala, već uređaj koji ima većinu osobina personalnog računara i multimedijalnog uređaja [9].

Osnovne funkcionalnosti STB uređaja su:

- Obrada emitovanog signala – STB uređaj mora da odabere opseg radio frekvencije (eng. Radio Frequencies – RF) koja je vezana za odabrani kanal.
- Sigurnost – obezbeđivanje zaštite signala od neovlašćenog pristupa. Ako korisnik nije pretplaćen na emitovani kanal, signal se ne dekriptuje, pri čemu je zadovoljena sigurnost na STB uređaju.
- Obrada audio i video podataka – audio i video signali se dekoduju i pretvaraju u format koji se može reprodukovati na povezanom TV uređaju.
- Interakcija sa korisnikom - korisnik može da odabere sadržaj koji želi da gleda pomoću daljinskog upravljača.. Da bi se poboljšala usluga, opraterima može da se šalje povratna informacija o kanalima koje korisnik gleda.

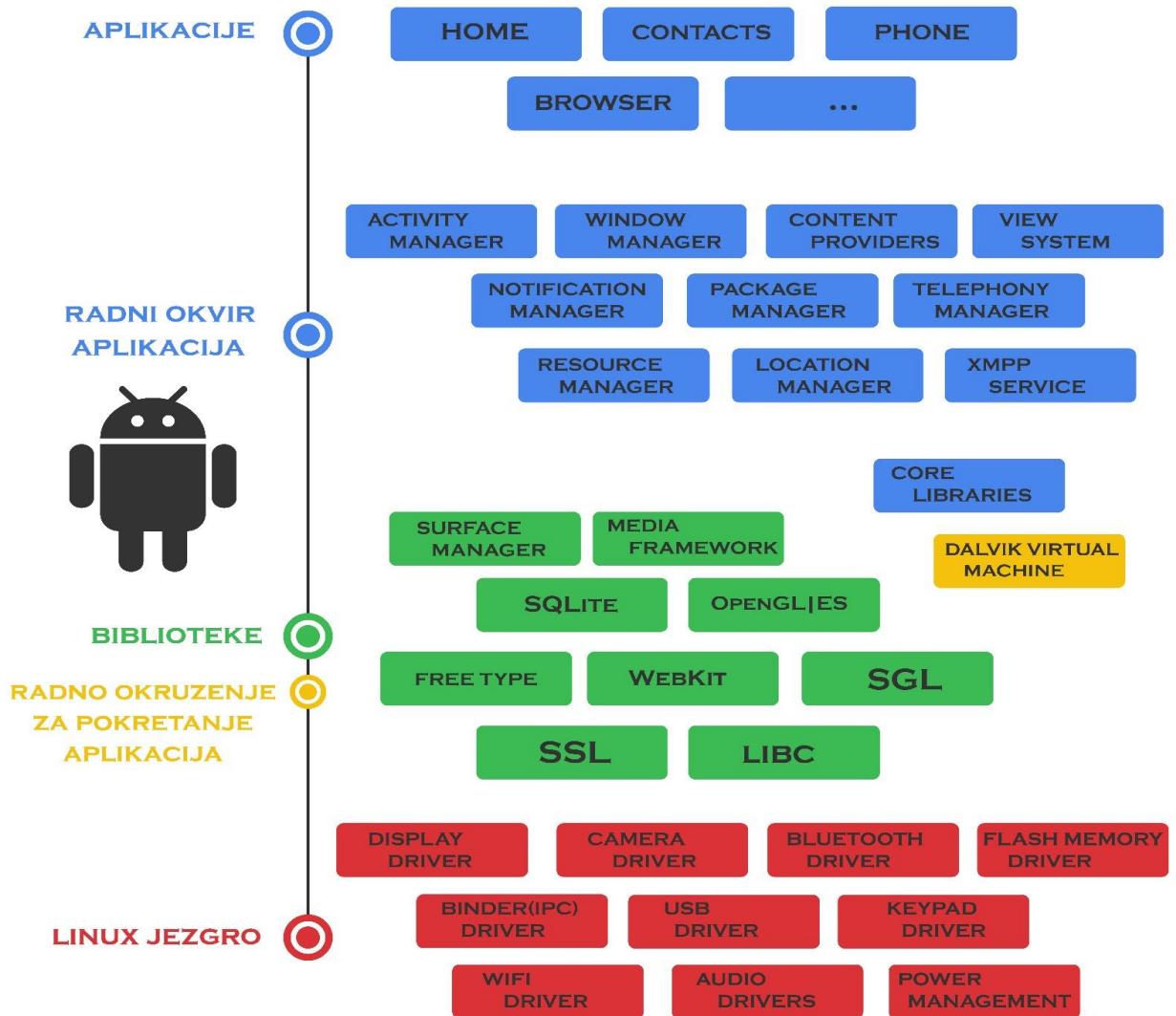
## 2.3 Android

### 2.3.1 Android platforma

Android platforma [10] se zasniva na Linux jezgru. Prednost Android platforme je njena mogućnost lakog prilagođavanja prirodi krajnjih uređaja zbog slojevite arhitekture koja je karakteriše.

Prvobitno je Android platforma bila namenjena za ARM procesorsku arhitekturu. Danas je našla primenu na velikom broju različitih uređaja koje karakterišu različite arhitekture, bilo ARM, MIPS ili x86.

Softverska arhitektura sistema predstavlja standardnu arhitekturu Linux-a gde su segmenti sistema razdvojeni po nivoima na kojima rade.



Slika 2.3 – Arhitektura Android programskog steka

Najniži sloj ove arhitekture se sastoji od modifikovanog Linux jezgra zaduženog za podršku hardvera i funkcija niskog nivoa.

Dalje se satoji iz skupa biblioteka zaduženih za dodatne podrške kao što su iscrtavanje grafika, podrška za dekodovanje video snimka, podrška za SSL enkripciju, itd. U sklopu biblioteka se nalazi i odvojeni sloj "Android Runtime" koji sadrži osnovne, bazne, biblioteke i Dalvik virtualnu mašinu. Dalvik virtualna mašina je zadužena za pokretanje aplikacija višeg nivoa napisanih u Java programskom jeziku.

Na višem nivou od biblioteka su sistemske aplikacije koje su neophodne za upotrebu sistema od strane korisnika, i tu se nalaze rukovaoci prozora, resursa, instalacionih paketa, kao i aplikacije zadužene za obavljanje osnovnih funkcija vezanih za uređaj na kom je instaliran Android.

Na najvišem nivou se nalaze krajnje korisničke aplikacije, odnosno aplikacije koje direktno koristi korisnik.

### 2.3.2 Android Open Source Project

Android ima aktivnu zajednicu programera koji koriste Android projekat otvorenog koda (eng. Android Open Source Project – AOSP) [11] da bi razvili i proizveli njihove lične verzije ovog operativnog sistema. Izdanja koja napravi ova zajednica obično donose nove dodatke i nadogradnje uređaja brže nego što bi to uradio zvanični proizvodjač, sa uporedivim nivoom kvaliteta. Takođe oni nastavljaju sa pružanjem podrške za zastarele uređaje koji više ne dobijaju zvanične nadogradnje ili pokreću Android na uređajima koji su prvobitno bili namenjeni za neke druge operativne sisteme. Ova izdanja često pružaju korisniku sva prava i dozvole i sadrže izmene koje nisu obezbeđene od strane prodavca, kao što je mogućnost promene takta i napona procesora uređaja.

Google je prepoznao ogroman potencijal Android platforme, zbog čega je rešio da otkupi sva prava od njegovog tvorca. Sledeći korak Google kompanije je možda i najznačajniji pri razvoju ove platforme, jer je Android ostao potpuno dostupan za programere i proizvođače uređaja, što je dovelo do današnjeg ogromnog Android tržišta. Google piše i održava sve Android izvorne kodove, što je zapravo AOSP. AOSP je besplatan za krajnje korisnike, ali i za proizvođače uređaja baziranih na Androidu.

### 2.3.3 Android TV platforma

Android TV je proširenje osnovne Android platforme razvijeno od strane Google-a, posebno za televizijske i samostalne multimedijalne plejere, poput STB-a i sličnih uređaja za reprodukciju video sadržaja.

U okviru Android TV platforme, Google nudi skup GTVS (eng. Google TV Services) [12] aplikacije. GTVS aplikacije su skup aplikacija razvijenih od strane Google-a sa ciljem da se poboljša iskustvo korišćenja Android TV platforme. Neki od pripadnika Google TV services aplikacija i servisa su unapred instalirane na sistemsku particiju Android TV-a, dok korisnik može da instalira i ostale koje se nalaze u Google-ovoj prodavnici aplikacija (eng. Google Play Services). Neke od aplikacija razvijenih od strane GTVS-a, koje su unapred instalirane na sistemsku particiju Android TV-a su:

- „Leanback Launcher“ – pokretač aplikacija za Android TV
- „Google Assistant“ – pomoć i pretraga
- „Play Store“ – zvanična Google prodavnica Android aplikacija
- „Play Games“ – aplikacija za igrice za Android TV
- „Play Movies“ – aplikacija za gledanje filmova i drugih TV emisija
- „You Tube“ – aplikacija za promovisanje video sadržaja

Da bi proizvođači Android uređaja dobili dozvolu da preintegrišu GTVS aplikacije, Android TV platforma mora da prođe Google-ov sertifikacioni proces.

### 2.3.4 Android Aplikacije

Android aplikacije [13] se pišu u Java i Kotlin programskom jeziku. Android SDK (eng. Software Development Kit) alati prevode kod, zajedno sa svim podacima i potrebnim datotekama u Android paket. Android paket predstavlja arhivsku datoteku sa ekstenzijom *apk*. Ova datoteka se šalje na Android uređaj, gde se postavlja (engl. Install) i omogućava pristup od strane krajnjih korisnika.

Aplikacione komponente su osnovni gradivni blokovi Android aplikacije. Postoje četiri različite vrste aplikacionih komponenti, gde svaka ima jasnu svrhu i poseban životni ciklus koji definiše kako se komponenta kreira i uništava:

- Aktivnosti (eng. Activity) predstavljaju jedan prikaz na ekranu sa grafičkom korisničkom spregom. Aplikacija može da se sastoji iz više aktivnosti. Kombinacijom više aktivnosti formira se jedinstven utisak o aplikaciji kod korisnika. Ipak svaka od aktivnosti je nezavisna i može se samostalno pokrenuti u zavisnosti od situacije i potrebe.
- Servis (eng. Service) je komponenta koja radi u pozadini i obavlja dugotrajne radnje bez direktnе interakcije sa korisnikom. Servis ne sadrži grafičku korisnicku spregu. Druge komponente mogu pokrenuti servis ili se povezati na njega kada im je potrebno da izvrše uslugu koju on obezbeđuje. Jedan primer funkcionalnosti servisa je to da servis može da pušta muziku u pozadini, dok se korisnik nalazi u nekoj drugoj aplikaciji.
- Dobavljač sadržaja (eng. Content Provider) omogućava deljenje podataka između aplikacija. Moguće je skladištiti podatke u sistemu datoteka, SQLite bazi podatka, na internetu ili bilo kojoj drugoj lokaciji za skladištenje kojoj aplikacija može da pristupi.
- Primalac emitovanih poruka (eng. Broadcast Receiver) je komponenta Android sistema koja je vezana za slanje i prijem sistemskih poruka. Ove sistemske poruke najčešće mogu da nastaju u trenutku kada je ekran isključen, baterija prazna, i drugo.

Android aplikacije napisane u Java programskom jeziku imaju pristup samo aplikativnom sloju Android platforme. Kako bi se nadomestio ovaj nedostatak, Android podržava povezivanje koda aplikacije sa kodom koji nije pisan u Java programskom jeziku pomocu JNI sprege. JNI (eng. Java Native Interface) je programski okvir koji omogućava spregu izmedju koda napisanog u Java programskom jeziku i koda napisanog u drugim jezicima kao što su C, C++ i Assembler. Na ovaj način aplikacija može imati pristup sistemskom prostoru.

Aplikacija može da zatraži dozvolu za pristup podacima kao što su kontakti korisnika, SMS poruke, kamera i drugo. Sve dozvole programer nabraja u *AndroidManifest.xml* datoteci, a moraju

biti odobrene od strane korisnika u trenutku instalacije aplikacije, inače aplikacija neće biti instalirana. Pre nego što Android platforma može da pokrene neku komponentu aplikacije, mora da sazna da ta komponenta postoji, čitajući AndroidManifest.xml datoteku aplikacije. Sve komponente aplikacije se moraju definisati u ovoj datoteci.

## 2.4 Android programska podrška TV prijemnika – TIF

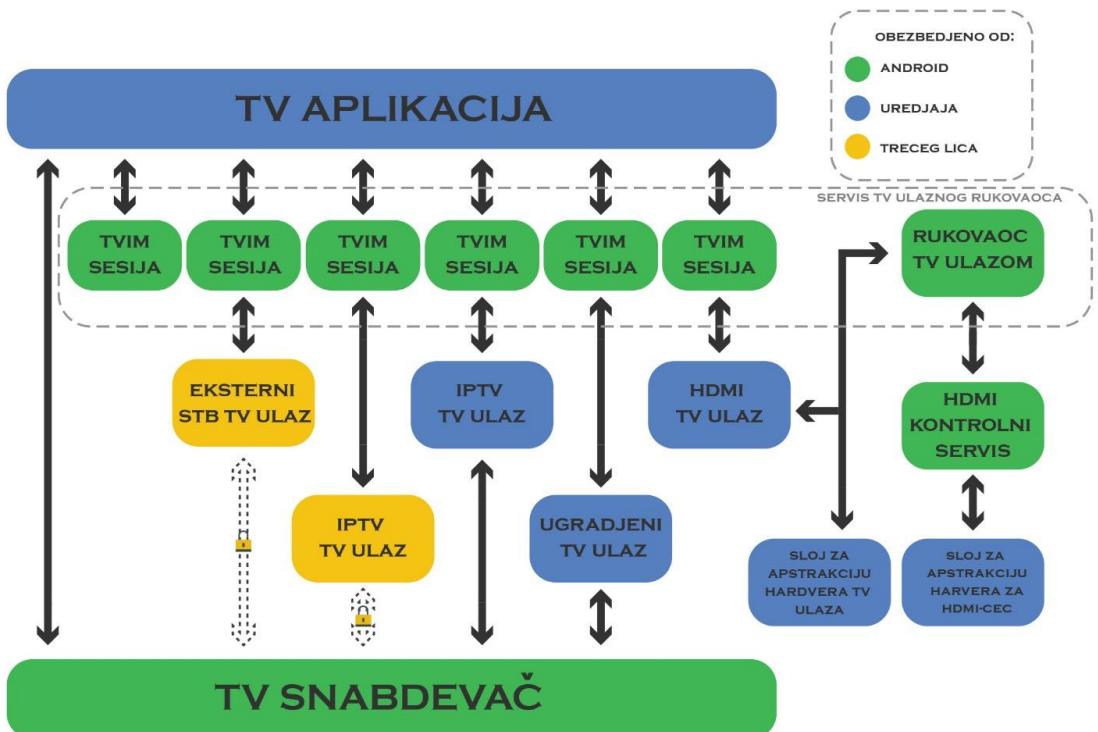
Android programska podrška za TV uređaje (eng. TV Input Framework – TIF) definiše aplikativnu programsku spregu i standardizuje način implementacije dopremanja emitovanog sadržaja do TV aplikacije. TIF omogućuje implementaciju TV ulaza preko kog je moguće reprodukovati sadržaj sa mrežnog poslužioca i/ili iz DVB transportnog toka podataka kao i pretragu televizije uživo. Programska podrška ne teži ka tome da implementira TV standarde ili regionalne zahteve, ali proizvođačima uređaja omogućava da lakše ispune regionalne digitalne TV standarde bez ponovne implementacije.

### 2.4.1 Komponente Android programske podrške za TV uređaje

TV aplikacija putem TIF-a pristupa ugrađenim modulima koji su isporučeni od strane proizvođača ili neke druge strane putem TV ulaznog rukovaoca. TIF se sastoji od:

- TV aplikacija (eng. TV Application): aplikacija pomoću koje korisnik rukuje sa hibridnom programskom podrškom za TV uređaje
- TV ulazni rukovaoc (eng. TV Input Manager): omogućuje TV ulazima da komuniciraju sa TV aplikacijom
- TV snabdevač (eng TV Provider): baza podataka sa kanalima, programima i pratećim dozvolama
- TV ulaz (TV Input): aplikacija koja predstavlja jedan izvor TV sadržaja
- Ulagna TV fizička arhitektura (eng. TV Input Hardware Abstraction Layer): fizička definicija arhitekture koja omogućuje sistemskim TV ulazima pristup fizičkoj arhitekturi specifičnoj za televiziju
- Roditeljska kontrola (eng Parental Control): tehnologija koja omogućava blokiranje kanala i programa
- HDMI-CEC: tehnologija koja omogućuje daljinsku kontrolu raznih uređaja putem HDMI-CEC poruka

Ove komponente će detaljnije biti objašnjene u tekstu koji sledi. Dijagram ispod vizuelno oslikava arhitekturu TIF-a.



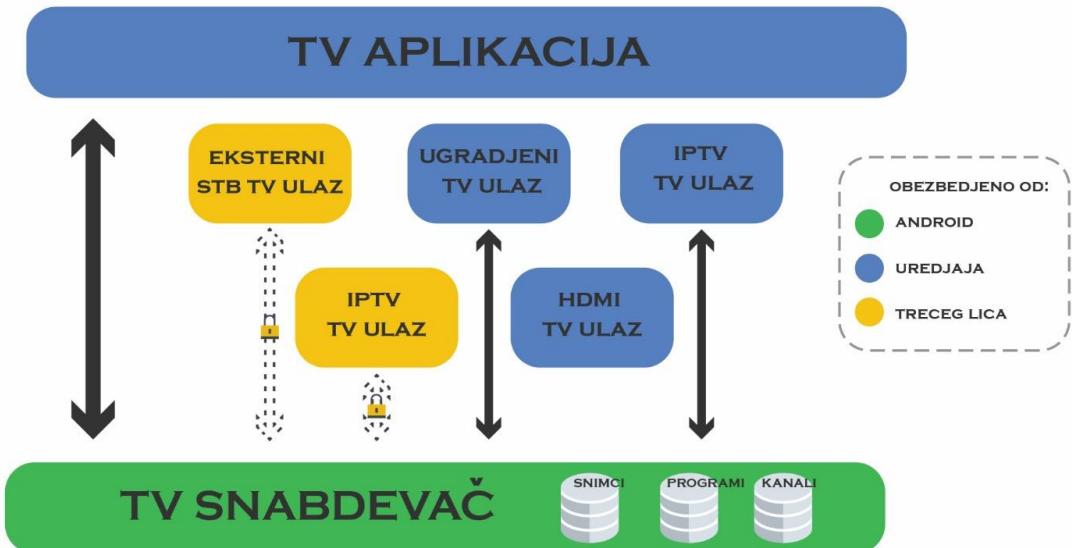
Slika 2.4 – TIF arhitektura

Na osnovu arhitekture sa slike se može zaključiti:

- Da korisnik može da vidi i bude u direktnom kontaktu sa TV aplikacijom
- TV aplikacija prikazuje sadržaj sa TV ulaza
- TV aplikacija nije u mogućnosti da direktno komunicira sa TV ulazima, već je komunikacija ostvarena pomoću rukovaoca TV ulazom koji identificiše stanje TV ulaza za TV aplikaciju

#### 2.4.2 TV snabdevač

TV snabdevač (eng. TV Provider) je komponenta koja u sebi sadrži bazu u kojoj se čuvaju lista kanala, programa (EPG) i snimljenog sadržaja. TV snabdevač podatke dobavi od strane TV ulaza (eng. TV Input). TV snabdevač takođe obezbeđuje i rukuje pratećim dozvolama kako bi TV ulazi bili u mogućnosti da vide samo svoje zapise, odnosno kako ne bi mogli da vide servise i programe drugih TV ulaza. Na taj način obezbeđena je konzistentnost podataka koji su popunjeni od strane drugih TV ulaza.



Slika 2.5 – TV snabdevač

Samo aplikacije koje se nalaze u privilegovanoj sistemskoj particiji mogu da čitaju podatke iz baze podataka TV snabdevača, dok samostalne aplikacije mogu da pristupe samo onim poljima u bazi podataka koje su oni popunili. Kao dodatak standardnim poljima za kanale i programe, baza podataka TV snabdevača takođe nudi i polja za binarno velike objekte (eng. Binary Large Object – BLOB), u svakoj tabeli koju TV ulazi koriste za skladištenje proizvoljnih podataka. Binarni podaci mogu da sadrže prilagođene informacije, kao što je na primer frekvencija povezanog frekventnog odabirača. Dostupno je i polje „column\_searchable“, koje može da izuzme neke kanale iz pretrage (kako bi podržali specifične zahteve pojedinih država za zaštitu sadržaja). Pretraga može biti tekstualna ili glasovna. Sva polja su vidljiva svakome sa pravom pristupa odgovarajućem redu. Nijedno polje nije direktno pristupačno korisniku. Korisnik samo vidi ono što TV aplikacija, sistemske aplikacije ili TV ulazi prikažu.

#### 2.4.3 TV ulazni rukovaoc

TV ulazni rukovaoc (eng. TV Input Manager) obezbeđuje centralnu sistemsku aplikativnu spregu (eng. Application Programming Interface – API) za ceo TIF.

TV ulazni rukovaoc nadgleda interakciju između aplikacija i TV ulaza i obezbeđuje funkcionalnost roditeljske kontrole. Sesija TV ulaznog rukovaoca mora biti kreirana jedan na jedan sa TV ulazom. Ovaj deo TIF-a može da dozvoli pristup aplikacijama TV ulazu tako da aplikacije mogu da:

- Listaju TV ulaze i proveravaju njihove statuse
- Naprave sesiju i rukuju slušaocima

Što se tiče sesija, TV aplikacije mogu da se povežu na TV ulaze samo preko URI-ja koji su dodale u bazu podataka TV snabdevača, osim prolaznih TV ulaza na koje se mogu povezati koristeći `TvContact.buildChannelUriForPassthroughInput()`. TV ulazi koji su obezbeđeni i imaju potpisani sertifikat od strane proizvođača uređaja ili druge aplikacije instalirane u sistemsku particiju imaju potpun pristup bazi podataka TV snabdevača. Ovaj pristup može da se iskoristi da se realizuju aplikacije koje će pretraživati sve dostupne TV kanale i programe.

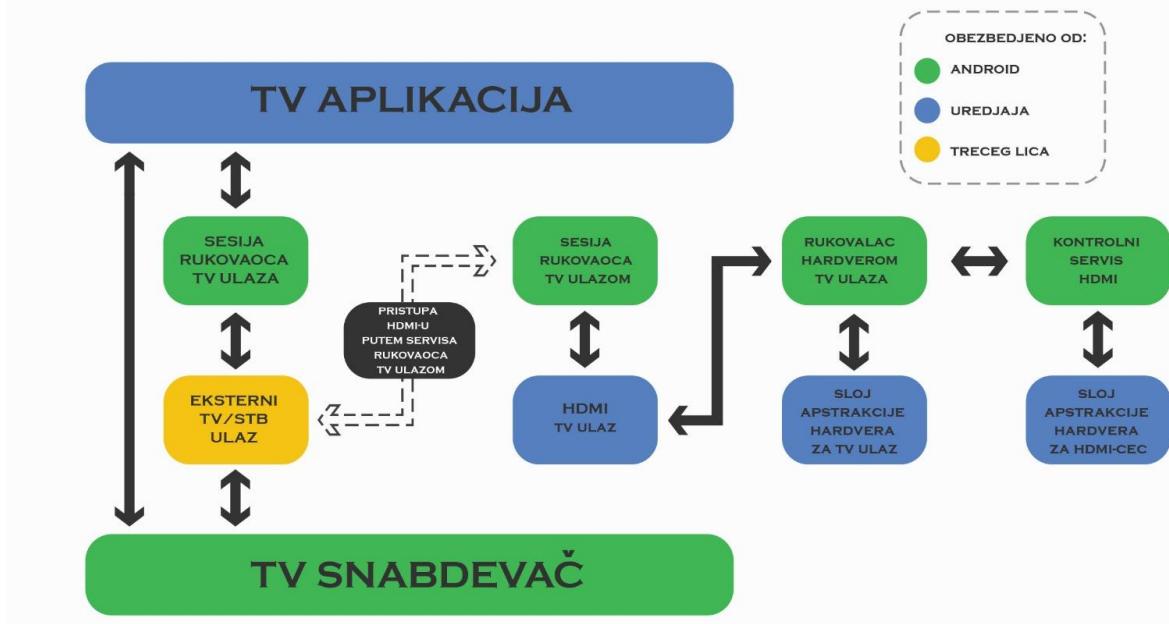
Aplikacije mogu da naprave i prijave `TvInputCallback` koji će da pozove `android.media.tv.TvInputManager` ukoliko se promeni stanje TV ulaza ili ukoliko se doda ili ukloni neki od TV ulaza. Na primer, TV aplikacija može da reaguje kada se neki TV ulaz isključi tako što će ga prikazati kao isključenog i zabraniće mogućnost biranja istog.

Rukovaoc TV ulazima apstrahuje komunikaciju između TV aplikacije i TV ulaza. Standardna sprega između rukovaoca TV ulazima i TV ulaza dozvoljava da više proizvođača naprave svoje TV aplikacije dok pomažu svim samostalnim TV ulazima da rade sa svim TV aplikacijama.

#### 2.4.4 TV ulaz

TV ulazi (eng. TV Input) su Android aplikacije u smislu da imaju `AndroidManifest.xml` i instaliraju se. Android TV podržava unapred instalirane sistemske aplikacije, aplikacije sa potpisanim sertifikatom od strane proizvođača uređaja I samostalne (eng. Third Party) TV ulaze.

Neki ulazi, kao što je na primer HDMI (eng. High-Definition Multimedia Interface) ulaz ili ugrađeni ulaz frekventnog odabirača, mogu biti obezbeđeni samo od strane proizvođača zato što oni komuniciraju direktno sa fizičkom arhitekturom. Drugi, kao na primer IPTV ili spoljni STB, mogu biti u vidu APK (eng. Android Package Kit) datoteka. Jednom kada se preuzmu i instaliraju takve aplikacije, novi ulazi mogu biti izabrani u samoj TV aplikaciji. U nastavku teksta će biti opisan primer spoljašnjeg samostalnog STB TV ulaza.



Slika 2.6 – TV ulaz treće strane

S obzirom da TV ulaz ne može direktno da pristupi HDMI video izvoru koji pristiže, mora da pristupi preko TV rukovaoca ulazom, i da koristi TV ulaz dobavljen od strane proizvođača uređaja.

Pomoću rukovaoca TV ulaza, spoljni STB TV ulaz može da komunicira sa HDMI TV ulazom i da ga "pita" da prikaže video putem HDMI-a. Na taj način STB TV ulaz može da kontroliše TV, dok HDMI TV ulaz, koji je obezbeđen od strane proizvođača, obrađuje video.

#### 2.4.5 TV aplikacija

Sistemska TV aplikacija (eng. TV application) [14] prezentuje TV sadržaj korisniku. Referentna TV aplikacija (Live TV) je obezbeđena zajedno sa Android platformom, koja može biti korišćena takva kakva jeste, može se promeniti, proširiti ili zameniti od strane proizvođača uređaja.

Najmanje što sistemska TV aplikacija mora da podrži na Android Oreo verziji je:

1. Podesavanje i konfiguracije:

- Automatska detekcija TV ulaza
- Da omogući TV ulazima da inicijalizuju postavke kanala
- Kontrolu roditeljskih podešavanja
- Promene (eng. Editing) kanala

2. Gledanje:

- Pristup i navigacija svim TV kanalima
- Pristup informacionim podacima TV programa
- Prikaz elektronskog programskog vodiča
- Podrška za više zvučnih traka, i prevod traka
- Prikaz PIN zahteva roditeljske kontrole (eng. PIN challenge)
- Popunjavanje rezultata pretrage za TV kanale i programe
- Rukovanje snimanjem digitalnog sadržaja i odloženo gledanje sadržaja
- Slika u slici (eng. Picture-in-picture – PiP)

Ovaj skup funkcija će se povećati sa novijim verzijama kako se i TIF API-ji budu proširivali. Da bi se proverilo da li aplikacija ima sve potrebne komponente postoji CTS verifikator (eng. Compatibility test suite) [15] koji obezbeđuje ispitivanje usklađenosti. Proizvođači moraju da implementiraju TV aplikaciju koja uključuje rezultate pretrage koji će biti uključeni u globalnu pretragu kako bi osigurali najbolje korisničko iskustvo. Referentna (Live TV) aplikacija pruža implementaciju koja obezbeđuje rezultate samostalnih ulaza kao i rezultate ugrađenih izlaza.

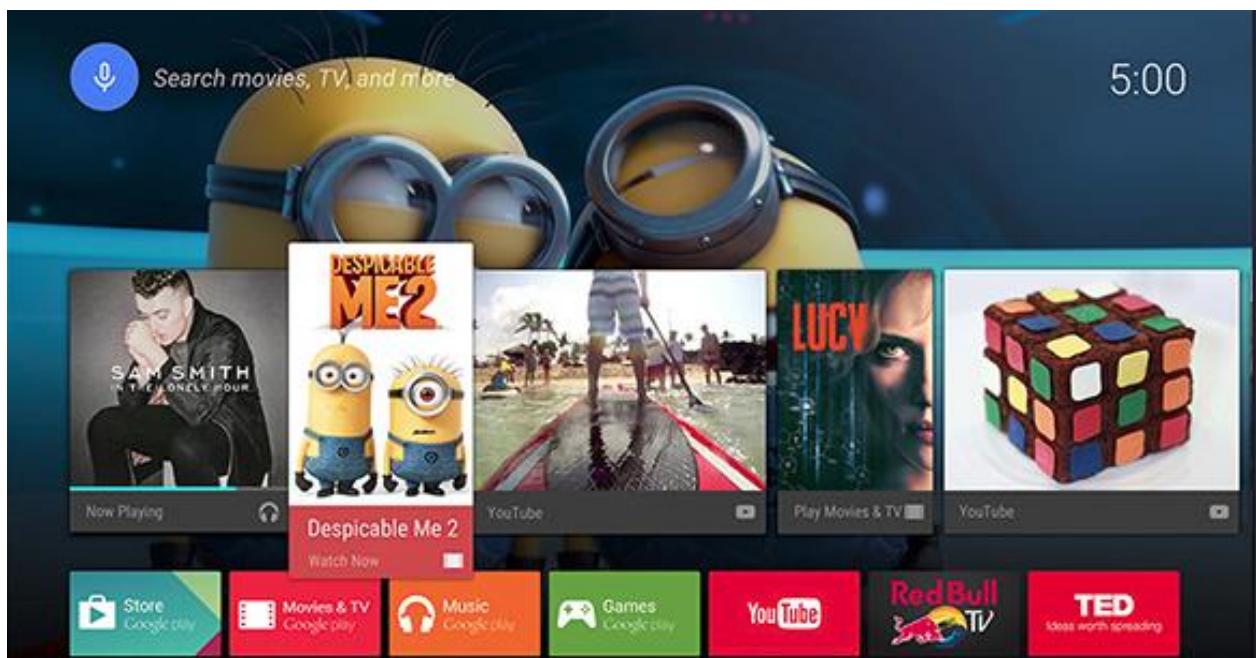
## 2.5 Pokretač aplikacija (eng. Launcher)

Sa pojavom Android TV platforme na Oreo verziji, u GTVS paketu aplikacija postoji novi pokretač aplikacija – „Leanback Launcher“, koji omogućuje da svaka aplikacija ima svoj kanal. Svaka aplikacija u svom kanalu može da predstavlja/promoviše željeni sadržaj korisnicima.

Cilj novog početnog ekrana je prikazivanje multimedijalnog sadržaja kanala u centralnom delu ekrana, stavljajući akcenat na TV emisije, filmove i video klipove koje će korisnici gledati. Takođe je dizajniran tako da smanji broj koraka koji je potreban da bi se došlo do određenog sadržaja.

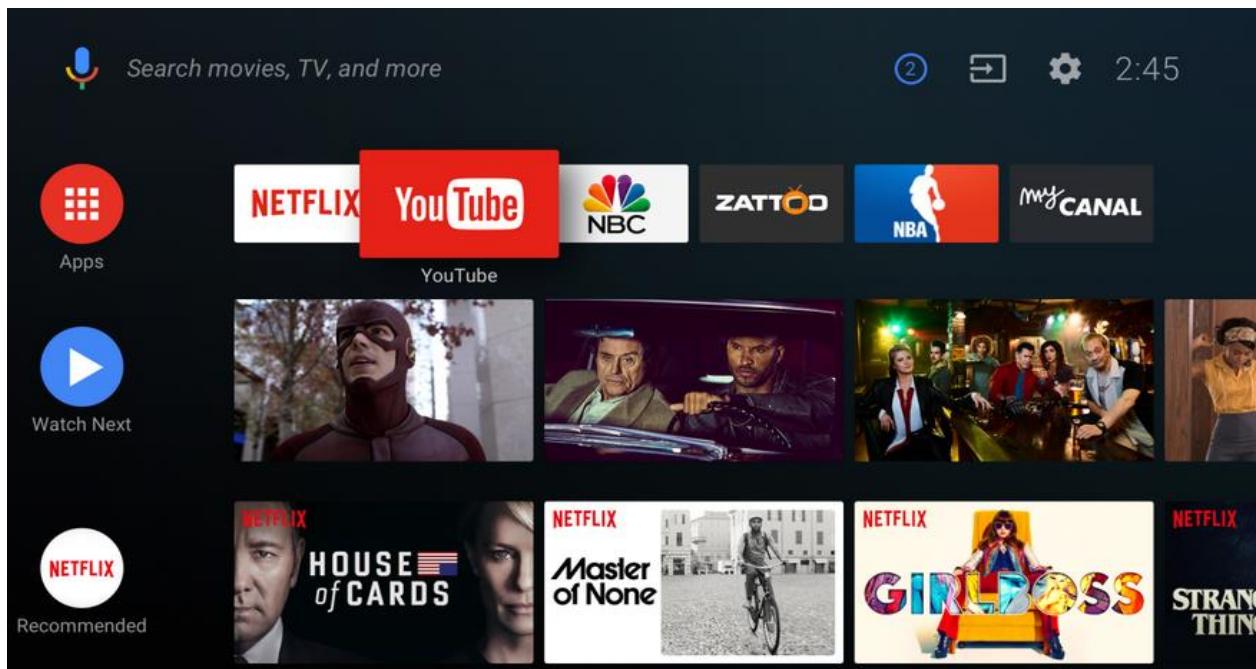
Prethodna verzija početnog ekrana za Android TV, Android Nougat, sastojala se od tri reda: pretraživač na vrhu, praćen preporučenim sadržajem iz svakog programa, sa aplikacijama na dnu ekrana. Da bi se došlo do željenog sadržaja, potrebno je izlistati preporučeni sadržaj, ili pokrenuti odgovarajuću aplikaciju.

Na slikama 2.8 i 2.9 mogu se videti razlike između početnih ekrana za Android Nougat i Android Oreo verziju.



Slika 2.7 – Izgled početnog ekrana za verziju Android Nougat

Uz Android Oreo, Google je predstavio novi način organizovanja sadržaja na TV-u. Organizovani u redove, kanali omogućavaju direktni pristup emisijama, filmovima i klipovima iz određenog izvora. Ne postoji ograničenje za broj kanala koji se može naći na početnom ekranu, korisnici imaju kontrolu nad količinom kanala i redosledom u kom se oni nalaze.



Slika 2.8 – Izgled početnog ekrana za verziju Android Oreo

### 2.5.1 Funkcionalnost „Program Preview”

Osim početnog ekrana, Android Oreo uvodi brojne nove mogućnosti u Android TV. Jedna od njih je funkcionalnost „Program Preview”, odnosno pregled programa, koja korisnicima omogućava da, zadržavajući se na nekoj od slika koje se nalaze u sklopu kanala, gledaju reprodukovani video zapis bez pokretanja aplikacije. Jos jedan nov detalj je to što, prolaskom kroz različite slike u sklopu kanala, pozadina menja boju.

Pregled programa je prevashodno namenjen za prikaz multimedijalnog sadržaja iz aplikacija koje nude sadržaj na zahtev („YouTube”, „Netflix”, „Play Movies”...). Predstavlja potpuno inovativan način dostavljanja TV sadržaja, jer omogućava korisnicima da zarone duboko u TV aplikaciju već na početnom ekranu. Ovo svakako doprinosi dinamičnosti iskustva gledanja TV-a, ali i umnogome olakšava korisnicima pronalaženje željenog sadržaja.

### 2.5.2 Funkcionalnost „Watch Next”

Još jedna nova funkcionalnost predstavlja mehanizam nazvan „Watch Next”. Ovaj mehanizam služi kao predlog korisniku da završi prethodno prekinutu akciju (da nastavi sa gledanjem filma, serije, emisije...). Predstavljen je grafičkim komponentama u drugom redu u okviru početnog ekrana, njega kreira i održava sistem. Ovaj kanal predlaže korisnicima sadržaj na osnovu prethodno gledanih programa, ali isto tako omogućava korisnicima da sami dodaju sadržaj od interesa, tako što će se pozicionirati na željeni program, i dugo pritisnuti dugme „select”. Dakle, u „Watch next” kanalu može se naći sledeći sadržaj:

- Programi koje je korisnik označio kao zanimljive
- Programi koje je korisnik počeo da gleda ali nije završio
- Programi koji su povezani sa sadržajem koji korisnik gleda (npr. sledeća epizoda u seriji, ili naredna sezona emisije)

## 3. Koncept rešenja

Ovo poglavlje sadrži analizu zadatog problema i koncept rešenja. Integracija digitalne televizije na Android platformu zahteva postojanje više slojeva programske podrške. Svaki od slojeva predstavlja prilagođen servis za druge slojeve. Potrebni elementi sistema i ciljna platforma su opisani u nastavku.

### 3.1 Komponente sistema

Na najvišem nivou apstrakcije sistema ka krajnjim korisnicima nalaze se Android aplikacije koje obavljaju grafičko prikazivanje videa, prikazivanje informacija elektronskog programske vodiča, liste dostupnih kanala i drugih informacija značajnih za televizijski program. Nakon integracije digitalne televizije na Android platformi i izlaganja Java programske sprege ka programerima, različite televizijske aplikacije se mogu pisati na identičan način kao i ostale televizijske aplikacije. Sledeći sloj čini skup funkcija u okviru programske sprege koji omogućava razmenu podataka između Android aplikacija i programske podrške televizijskog prijemnika. Postojanjem takve programske sprege, programer više ne mora da bude upoznat sa realizacijom programske podrške samog televizijskog prijemnika. Vezu izmedju skupa funkcija datog u obliku programske sprege i sprežnog sloja programske podrške televizijskog prijemnika predstavlja JNI sloj. Nakon ovog sloja sledi prilagodni sloj programske podrške televizijskog prijemnika. Ovaj sloj ujedinjuje programske sprege ka višim slojevima programske podrške (u ovom slučaju Java sloj). Ukoliko dođe do zamene sprežnog sloja programske podrške televizijskog prijemnika jednog proizvođača sa programskom podrškom drugog proizvodjača, ovaj sloj i svi iznad ostaju nepromjenjeni. Srž sistema predstavlja sledeći sloj, to jest sprežni sloj programske podrške televizijskog prijemnika, koji mora biti definisan i realizovan u skladu sa DVB standardima. Sledeći sloj je sloj apstrakcije fizičke arhitekture,

predstavljen kao sloj između operativnog sistema i programske podrške televizijskog prijemnika. Da bi se izvršilo prilagođenje aplikacije i programske podrške televizijskog prijemnika nekoj drugoj platformi, dovoljno je izvršiti promenu samo ovog sloja. Ukoliko se desi neki asinhron događaj karakterističan za televizijski prijemnik, a za čije promene se korisnik prijavio, programska podrška obaveštava JNI sloj koji poziva određenu funkciju u Android okruženju. Funkcija, po potrebi, javlja prijavljenim aplikacijama da se promena određenog tipa upravo desila.

### 3.2 Opis ciljne platforme I ograničenja

DTV prijemnik koršćen pri izradi ovog rada je Android platforma bazirana na Marvell BG5CT čipsetu. Komponente pomenute platforme su:

- Procesor – 1.6 GHz 15K DMIPs Quad Core ARM
- GPU – 2.8 Gpixel per sec.
- RAM memorija: 2 GB
- FLASH memorija: 8GB



Slika 3.1 – Razvojna platforma

### 3.3 Programska podrška STB uređaja

Funkcionalnost Android TV platforme je orijentisana više ka načinu kako prezentovati korisnicima TV sadržaj i ne sadrži direktnu podršku za upravljanje HLS transportnim tokom podataka kao ni upravljanje fizičkim komponentama DTV prijemnika. Da bi se omogućilo upravljanje fizičkim komponentama DTV prijemnika i upravljanje HLS transportnim tokom podataka, na postojeće slojeve Android platforme dodat je deo DTV programske podrške. Na slici je prikazana struktura programske podrške TV prijemnika zasnovanom na Android platformi.



Slika 3.2 – Struktura programske podrške TV prijemnika

TV aplikacija je komponenta koja je definisana od strane TV ulaznog radnog okvira i ona predstavlja grafičko rešenje Android TV aplikacije. Zadužena je za prikaz svih grafičkih elemenata, odnosno informacija dobijenih u komunikaciji sa programskom podrškom za TV uređaje i opsluživanje upravljačkih komandi pristiglih od strane krajnjeg korisnika. TV aplikacija se oslanja na programsku podršku za TV uređaje. Programska podrška za TV uređaje, kao i TV aplikacija, opisani su u poglavlju 2.6 Android programska podrška TV prijemnika – TIF.

Servis programske podrške za TV uređaje predstavlja standardni Android servis koji objedinjuje DTV funkcionalnost i obezbeđuje skup DTV funkcija izložen gornjim slojevima (eng. Application programming interface – API). Pošto android servis može da pribavlja i isporučuje podatke u pozadini, dok je u isto vreme prikazana grafička sprega kojoj servis obezbeđuje te podatke, TIF propisuje ovaj mehanizam koji predstavlja sponu izmedju sprege za rukovanje DTV srednjim slojem i TV ulaznim okvirom. Servis programske podrške za TV uređaje je opisan u okviru poglavlja 2.6.4 TV ulaz.

DTV servis srednjeg sloja je proširenje DTV srednjeg sloja i on pruža direktnu spregu sa Linux servisom srednjeg sloja pružajući dodatne funkcionalnosti visokog nivoa realizovane u programskom jeziku Java.

Sloj za spregu sa DTV srednjim slojem (eng. Middleware Abstraction Layer) je sprega koja omogućava programskoj podršci srednjeg sloja da komunicira sa višim slojevima. Ona omogućava višim slojevima da koriste funkcionalnosti srednjeg sloja, bez poznавања unutrašnjih detalja реализације.

Programska podrška srednjeg sloja (eng. Middleware) obezbeđuje DTV funkcionalnost. Zadužena je za rukovanje i obradu digitalnog toka podataka, kao i za podešavanje DTV prijemnika, upravljanje listom programa, audio i video prikazom i pristup informacijama o emisijama. Takođe, obezbeđuje podršku za prikaz prevoda I teleteksta, zakazivanje podsetnika, funkcionalnost snimanja video sadržaja, izvršavanje MHEG aplikacije, prikaz HbbTV sadržaja itd. Komunikaciju sa fizičkom arhitekturom ostvaruje preko najnižeg sloja programske podrške, HAL sloja.

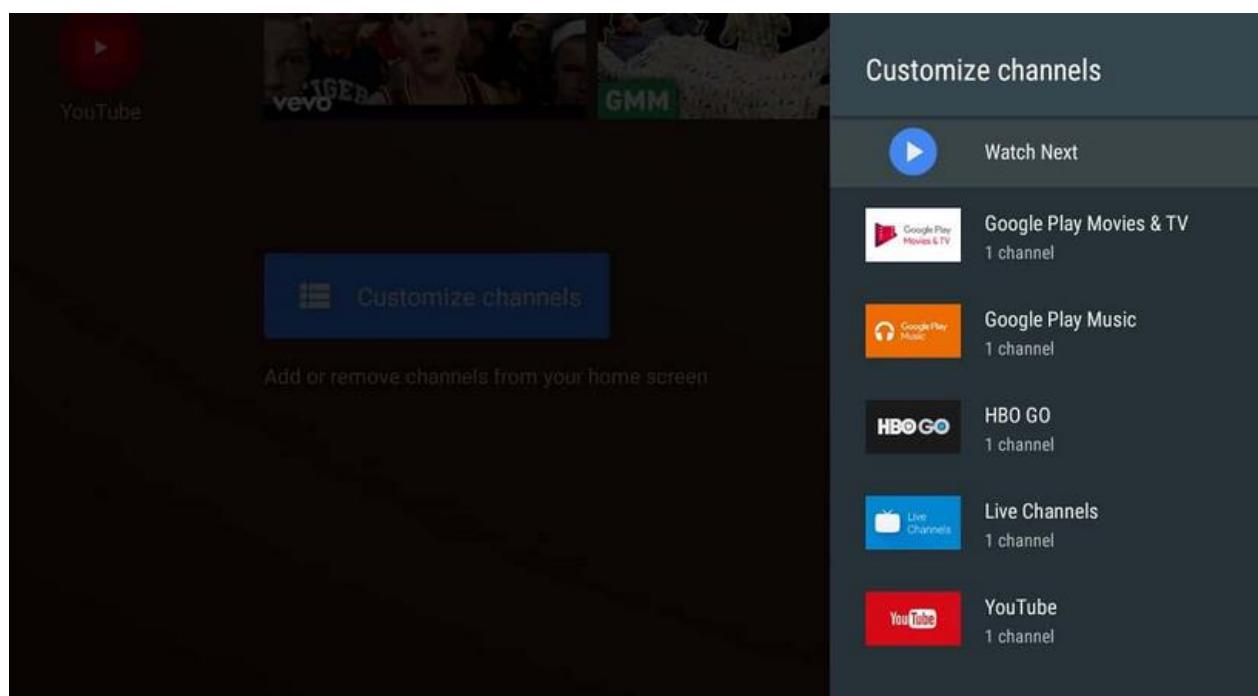
Sloj za apstrakciju fizičke arhitekture (eng. Hardware Abstraction Layer – HAL) je najniži nivo programske podrške I predstavlja apstrakciju fizičke arhitekture TV prijemnika.

Zadužen je za rukovanje složenim podacima koje dobija od fizičke arhitekture i obezbeđuje brojne usluge višim slojevima, kao što su podešavanje frekvencije, jačine zvuka, upravljanje formatima slike itd. Takođe, raspolaze skupom funkcija koje apstrahuju funkcionalnosti vezane za korišćeni operativni sistem (synchronizacija, kritične sekcije, semafori itd) i obaveštava više slojeve o nastalim događajima. Uvođenjem ovog sloja, proizvođaci DTV programske podrške ne moraju

poznavati specifičnosti fizičke arhitekture, odnosno omogućeno je izvršavanje programske podrške DTV srednjeg sloja na Android platformi.

### **3.4 Proširenje programske podrške za „Program Preview“ i „Watch Next“**

Android TV pokretač – Leanback Launcher, omogućava aplikacijama da kreiraju svoj kanal, sa ciljem prezentacije sopstvenog sadržaja krajnjim korisnicima. Jedna aplikacija može da ponudi proizvoljan broj korisničkih kanala, koji se mogu dodati na početni ekran tako što sto će korisnik iz željenih aplikacija ručno izabrati kanale koje želi da prikaže. Svaka aplikacija ima mogućnost da programski odabere i prijavi podrazumevani kanal. Podrazumevani kanal se automatski pojavljuje na početnom ekranu, bez eksplicitnog zahteva korisnika. Sistem koristi TV snabdevač (eng. TV Provider) za upravljanje kanalima i programima na početnom ekranu. Aplikacija takođe komunicira sa TV snabdevačom prilikom instaliranja i nadogradnje TV kanala i programa.



Slika 3.3 – Ručno biranje kanala koje nudi aplikacija

Mehanizam „Program Preview“ je prevashodno namenjen za prikaz sadržaja iz multimedijalnih servisa koji nude sadržaj videa na zahtev („YouTube“, „Netflix“, „Play Movies“...). Ideja je da se isti mehanizam upotrebi za promovisanje TV sadržaja. Kako bi se omogućilo ovakvo ponašanje, potrebno je proširiti postojeći TV Input.

Prvi korak ka realizaciji ove ideje jeste da se prilikom pokretanja aplikacije pribavi lista dostupnih TV kanala. Korisnička aplikacija treba da sadrži jedan kanal, koji će se na zahtev korisnika naći na početnom ekranu. Kada je lista kanala pribavljena, korisnički kanal biće popunjen programima – jedan program predstavljaće jedan TV kanal. Korisnik će moci da pregleda sadržaj TV kanala tako što će se zadržati na slici koja predstavlja program. Ovo će biti omogućeno mehanizmom „Program Preview”, koji je postao dostupan sa pojavom Android Oreo verzije. Korisniku će biti omogućeno da direktno iz aplikacije, odnosno kanala, pristupi aplikaciji „Live Channels”, tako što će se pozicionirati na neki od programa i pritinuti dugme „select”.

Takođe, zamisao je i da se poslednji kanal iz aplikacije „Live channels” koji je korisnik gledao, doda u kanal „Watch Next”, kako bi se smanjio broj koraka koji je potreban da nakon ponovnog pokretanja Android TV aplikacije, korisnik nastavi sa gledanjem istog kanala.

## 4. Programsко rešenje

U ovom poglavlju će detaljno biti opisano programsko rešenje u kom je implementirano pomenuto proširenje TV Input Framework komponente. Android aplikacija koja koristi funkcionalnost programske podrške televizijskog prijemnika se razvija korišćenjem dostupnih funkcija i elemenata u Android okruženju.

Za realizaciju zadatka iskorišćen je već postojeći TV ulaz sa sledećim funkcionalnostima:

- Pribavljanje TV kanala
- TV i radio usluge
- Elektronski programski vodič
- Roditeljska kontrola
- Promena audio zapisa i titlova

Proširen je mehanizmom za „Program Preview“ i „Watch Next“, na način koji će biti opisan u nastavku..

Kanali koji se prezentuju na početnom ekranu vrše reprodukciju videa u okviru samog Launcher-a, pomoću komponente za reprodukciju video sadržaja iz Android Framework-a. Iz tog razloga, video sadržaj koji se može reprodukovati je samo onaj skup video kodeka koji može da reprodukuje Android platforma (mp4, MPEG...). Tu ne spada DVB transportni tok podataka, pa se zbog ovog ograničenja za realizaciju zadatka koristi TV ulaz za HLS (eng. HTTP Live Streaming) kanale, koji su javno dostupni i besplatni .

Početni ekran Android TV-a koristi programsku spregu TV snabdevača za upravljanje kanalima i programima koje stvara aplikacija. Da bi pristupili podacima dobavljača, potrebno je dodati sledeći kod u XML datoteku:

```
<uses-permission
    android:name="android.permission.READ_EXTERNAL_STORAGE" />

<uses-permission
    android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

## 4.1 Kreiranje kanala

Da bi se omogućila funkcionalnost pregleda programa, prevashodno je potrebno napraviti kanal, u okviru pokretača aplikacija, u kome će ti programi biti prikazani. Prilikom kreiranja kanala, vrši se upis tog kanala u bazu podataka TV snabdevača. Kanal je kreiran na sledeći nacin:

- Kreiranje objekta kanala pomoću graditelja kanala, i postavljanje njegovih atributa:

```
Channel.Builder builder = new Channel.Builder();

builder.setType(TvContractCompat.Channels.TYPE_PREVIEW)

.setDisplayName("HLS TV Input")

.setAppLinkIntentUri(Uri.parse(TvContractCompat.Channels.

COLUMN_APP_LINK_INTENT_URI));
```

- Dodavanje kanala u TV snabdevač:

```
Uri channelUri = context.getContentResolver().insert(
    TvContractCompat.Channels.CONTENT_URI,
    builder.build().toContentValues());
```

- Parsiranje Uri-ja (eng. Unique Resource Identifier) kanala kako bi se došlo do njegovog ID-a, koji će biti potreban zbog dodavanja programa u kanal:

```
long channelId = ContentUris.parseId(channelUri);

• Postavljanje ikonice za kanal:
```

```
int channelLogo = R.drawable.ic_video_library_blue_80dp;

Bitmap bitmap = convertToBitmap(context, channelLogo);

storeChannelLogo(context, channelId, bitmap);
```

Funkcija storeChannelLogo smešta ikonicu u bazu snabdevača sistema i povezuje je sa datim ID-om kanala. Ova funkcija takođe omogućava prikazivanje ikonice na početnom ekranu.

Kreiranje kanala poziva se prilikom inicijalizacije objekta klase TifManager, koja sledi odmah nakon pokretanja Android servisa.

## 4.2 Kreiranje Preview programa

Nakon što je kanal uspešno kreiran, moguće je u njega dodati programe. Informacije koje su potrebne za kreiranje jednog programa uzete su iz liste kanala iz TV inputa.

Prilikom kreiranja programa, korišćen je PreviewProgram graditelj, iz razloga što želimo da se prilikom zadržavanja na slici koja predstavlja program, reprodukuje video sadržaj TV kanala. Kreiranje programa ralizovano je na sledeći nacin:

- Kreiranje objekta programa i postavljenje atributa, pridruživanje programa određenom kanalu (preko channelId-a):

```
PreviewProgram.Builder builder = new PreviewProgram.Builder();
builder.setChannelId(channelId)
.setType(TvContractCompat.PreviewPrograms.TYPE_CLIP)
.setTitle(title)
.setPosterArtUri(cardImageUri)
.setPreviewVideoUri(videoUri)
.setIntentUri(intentUri);
```

- Dodavanje programa u TV snabdevač:

```
Uri programUri = context.getContentResolver().insert(
    TvContractCompat.PreviewPrograms.CONTENT_URI,
    builder.build().toContentValues());
```

- Parsiranje Uri-ja programa kako bi se došlo njegovog ID-a, koji će biti potreban za nadogradnju programa:

```
long programId = ContentUris.parseId(programUri);
```

- Nadogradnja programa:

```
context.getContentResolver().update(
    TvContractCompat.buildPreviewProgramUri(programId),
    builder.build().toContentValues(), null, null);
```

Prilikom kreiranja programa, potrebno je proveriti da li on već postoji u bazi. Ukoliko postoji, nadogradujemo ga, u suprotnom, kreiramo novi program. Funkcija za kreiranje programa poziva se takođe prilikom inicijalizacije objekta klase TifManager, koja sledi odmah nakon pokretanja Android servisa. Prilikom kreiranja videoUri-ja, korišćen je javno dostupan HTTPS URL. CardImageUri je slika koju program prikazuje u kanalu. Sistem pokreće IntentUri kada korisnik izabere program iz kanala u lanseru. Ovaj Uri uključuje Id programa kako bi aplikacija mogla da nađe i reprodukuje TV kanal iz baze kada korisnik odabere program, odnosno kako bi se pokrenula aplikacija „Live Channels” na tačno onom TV kanalu koji je odabrao korisnik iz „HLS TV Input” kanala.

### 4.3 Kreiranje WatchNext programa

Watch Next kanal je mehanizam razvijen od strane Googl-a i pruža mogućnost aplikacijama da u njemu predlažu sadržaj iz svoje aplikacije. U ovoj realizaciji, program koji će se naci u „Watch Next” kanalu je poslednji TV kanal koji je korisnik gledao u aplikaciji „Live Channels”.

Prilikom kreiranja Watch Next programa, koriscen je WatchNextProgram graditelj, kako bi program koji želimo bio dodat u sistemsku aplikaciju „Watch Next”. Kreiranje programa ralizovano je na sledeći nacin:

- Kreiranje objekta WatchNext programa I postavljenje atributa:

```
WatchNextProgram.Builder builder = new WatchNextProgram.Builder();
builder.setType(TvContractCompat.WatchNextPrograms.TYPE_CLIP)
.setWatchNextType(TvContractCompat.WatchNextPrograms.WATCH_NEXT_TYPE_WATCHLIST)
.setTitle(title)
.setPosterArtUri(cardImageUri)
.setPreviewVideoUri(videoUri)
.setIntentUri(intentUri);
```

- Dodavanje WatchNext programa u TV snabdevač:

```
Uri watchnextProgramUri = context.getContentResolver().insert(
TvContractCompat.WatchNextPrograms.CONTENT_URI,
builder.build().toContentValues());
```

- Parsiranje Uri-ja WatchNext programa kako bi se došlo do njegovog ID-a, koji će biti potreban za nadogradnju WatchNext programa:

```
long programId = ContentUris.parseLong(watchNextProgramUri);
```

- Nadogradnja WatchNext programa:

```
context.getContentResolver().update(  
    TvContractCompat.buildWatchNextProgramUri(programId),  
    builder.build().toContentValues(), null, null);
```

- Brisanje WatchNext programa:

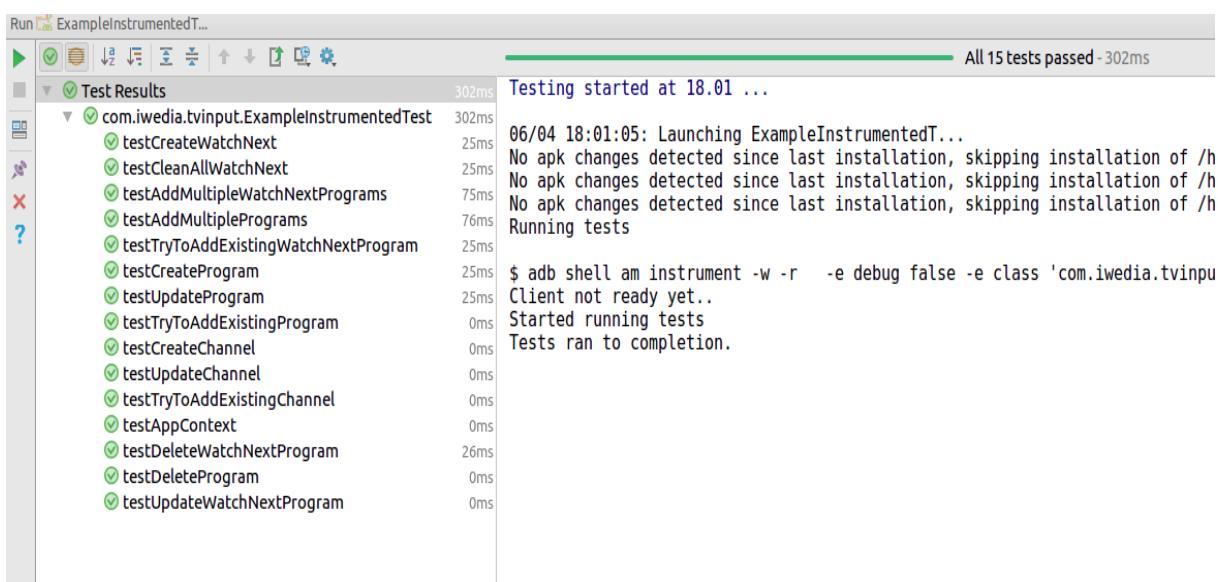
```
context.getContextResolver().delete(TvContractCompat.  
    buildWatchNextProgramUri(watchNextProgramId), null, null);
```

Funkcija u kojoj se kreira WatchNext program, poziva se prilikom svake promene TV kanala daljinskim upravljačem. Logika dodavanja programa u kanal „WatchNext” realizovana je u okviru funkcije PlayNext(). U ovoj funkciji dobavljamo Uri kanala koji se trenutno reproducuje, iz klase TvSession, i na osnovu njega dolazimo do ID-a programa u bazi. Pomoću ID-a programa, dolazimo do svih atributa programa koji želimo da dodamo u „Watch Next” kanal, iz liste programa u bazi, i pozivamo kreiranje WatchNext programa postavljajući dobavljene atrubute kao parametre funkcije. Jedan od parametara funkcije za kreiranje WatchNext programa je ID prethodno gledanog programa, koji prosleđujemo funkciji za brisanje WatchNext programa. Ova funkcija poziva se pre pozivanja PlayNext() funkcije, jer je zamisao da se u kanalu „Watch Next” uvek nalazi jedan, poslednji gledani TV program, ne računajuci one koje je korisnik ručno dodaо. Funkcija za brisanje kanala poziva se i na samom početku funkcije za kreiranje WatchNext programa, da bi se izbeglo dodavanje svakog kanala koji je korisnik gledao. Da bi se i nakon gašenja i ponovnog startovanja Android TV-a poslednji gledani TV program nalazio u „Watch Next” kanalu, bilo je potrebno da ID poslednjeg kanala dodatog u „Watch Next” sačuvamo u objekat tipa SharedPreferences. Funkcija loadLastPlayNextProgram() učitava podatke iz objekta tipa SharedPreferences, i kao povratnu vrednost vraca ID programa. Prilikom pozivanja ove funkcije i pri startovanju Androida, ID poslednjeg gledanog TV programa dobija vrednost iz Shared Preferences-a.

## 5. Ispitivanje i verifikacija

U okviru ovog rada obavljena su ispitivanja realizovanih modula. Ručno ispitivanje vršeno je u Launcher-u. Druga vrsta ispitivanja definiše nezavisne JUnit ispitne slučajeve, kojima su verifikovani svi moduli koji su predmet ovog rada. Za potrebe ispitivanja i verifikacije u Android aplikaciji napravljen je poseban modul koji je sačinjen od niza funkcija grupisanih po vrsti testova. Modul je napisan kao JUnit testna klasa. JUnit ispitni slučajevi ne mogu da daju kompletan dokaz o ispravnosti realizovane funkcionalnosti iz razloga što su to funkcionalni ispitni slučajevi kojima se može proveriti da li je očekivana funkcionalnost ispravna na programskom nivou. Iz tog razloga je rađeno i ručno ispitivanje slučajeva koji su od značaja za proveru ispravnog funkcionisanja aplikacije. Izvršeni testovi prikazani su u tabeli i potvrdili su ispravnost realizovanih modula.

Performance testovi nisu rađeni, jer se iz realizovanih JUnit testova može videti da svaka operacija zahteva vrlo malo vremena za izvršavanje i time ne opterećuje i ne narušava performanse uređaja.



Slika 5.1 – Rezultati JUnit testova

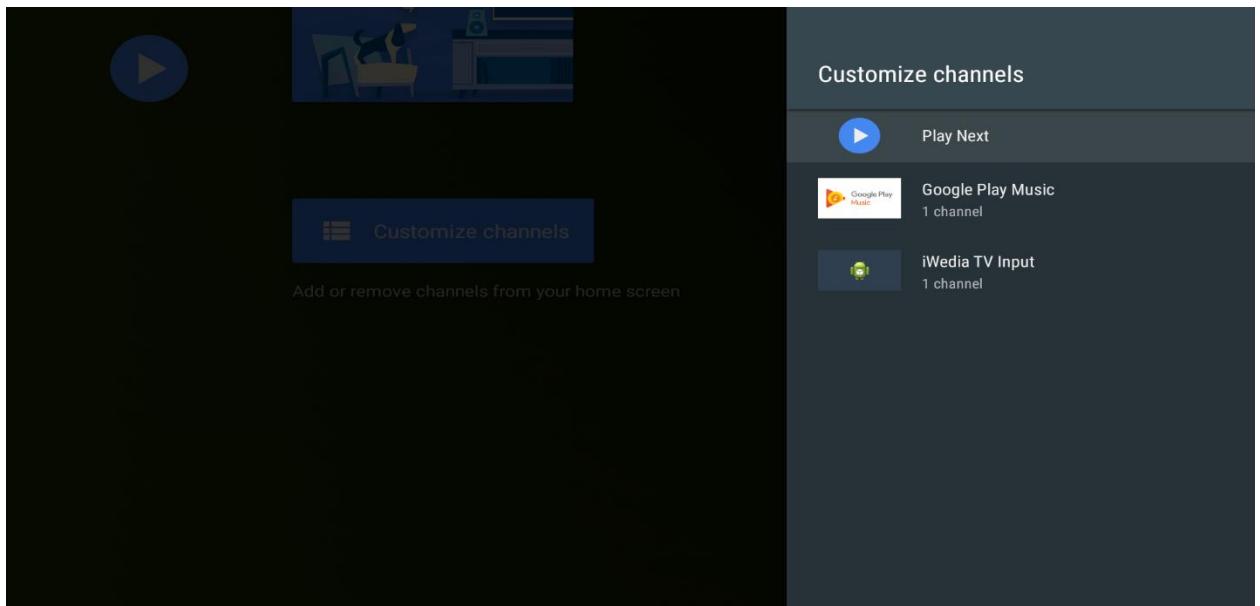
TEST	REZULTATI
Kreiranje kanala - testCreateChannel()	PROŠAO
Nadogradnja kanala - testUpdateChannelTitle()	PROŠAO
Pokušaj dodavanja postojećeg kanala - testTryToAddExistingChannel()	PROŠAO
Kreiranje programa - testCreateProgram()	PROŠAO
Nadogradnja programa - testUpdateProgram()	PROŠAO
Brisanje programa - testDeleteProgram()	PROŠAO
Pokušaj dodavanja postojećeg programa u kanal - testTryToAddExistingProgram()	PROŠAO
Dodavanje više programa u kanal - testAddMultiplePrograms()	PROŠAO
Dodavanje programa u WatchNext - testCreateWatchNext()	PROŠAO
Nadogradnja WatchNext programa - testUpdateWatchNextProgram()	PROŠAO
Brisanje programa iz WatchNext kanala - testDeleteWatchNextProgram()	PROŠAO
Pokušaj dodavanja postojećeg programa u WatchNext kanal - testTryToAddExistingWatchNextProgram()	PROŠAO
Dodavanje više kanala u WatchNext kanal - testAddMultipleWatchNextPrograms()	PROŠAO
Brisanje svih programski dodatih programa iz WatchNext kanala - testCleanAllWatchNext()	PROŠAO

Tabela 5.1 – Ispitane funkcionalnosti i rezultati

Ručno ispitivanje realizovanih modula urađeno je za sve moguće slučajeve zbog kojih je moglo doći do nekonzistentnog stanja aplikacije.

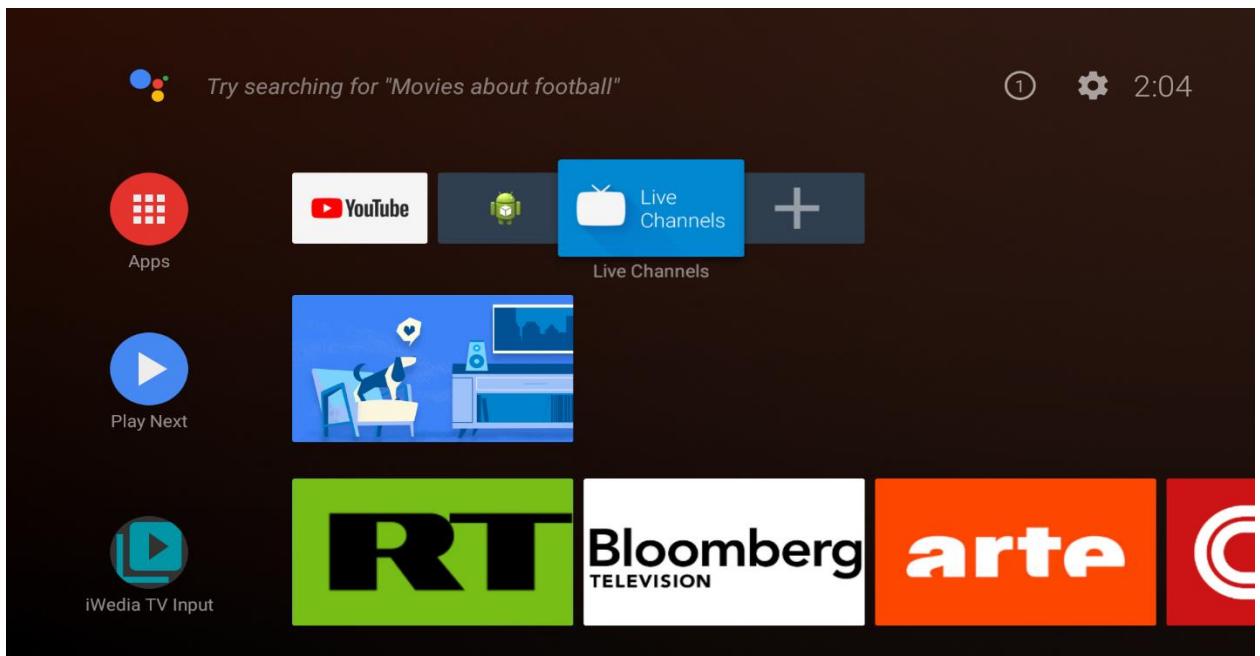
Na sledećim slikama može se videti na koji način se reprodukuju realizovane funkcionalnosti:

- Kada je aplikacija instalirana, korisnik treba da odabere kanal iz aplikacije kako bi bio prikazan na početnom ekranu:



Slika 5.2 – Odabir kanala iz aplikacije

- Pokretanjem aplikacije, nakon učitavanja dostupnih TV kanala, na početnom ekranu se pojavljuje kanal popunjen programima, koji predstavljaju sve dostupne TV kanale iz aplikacije „Live Channels“:



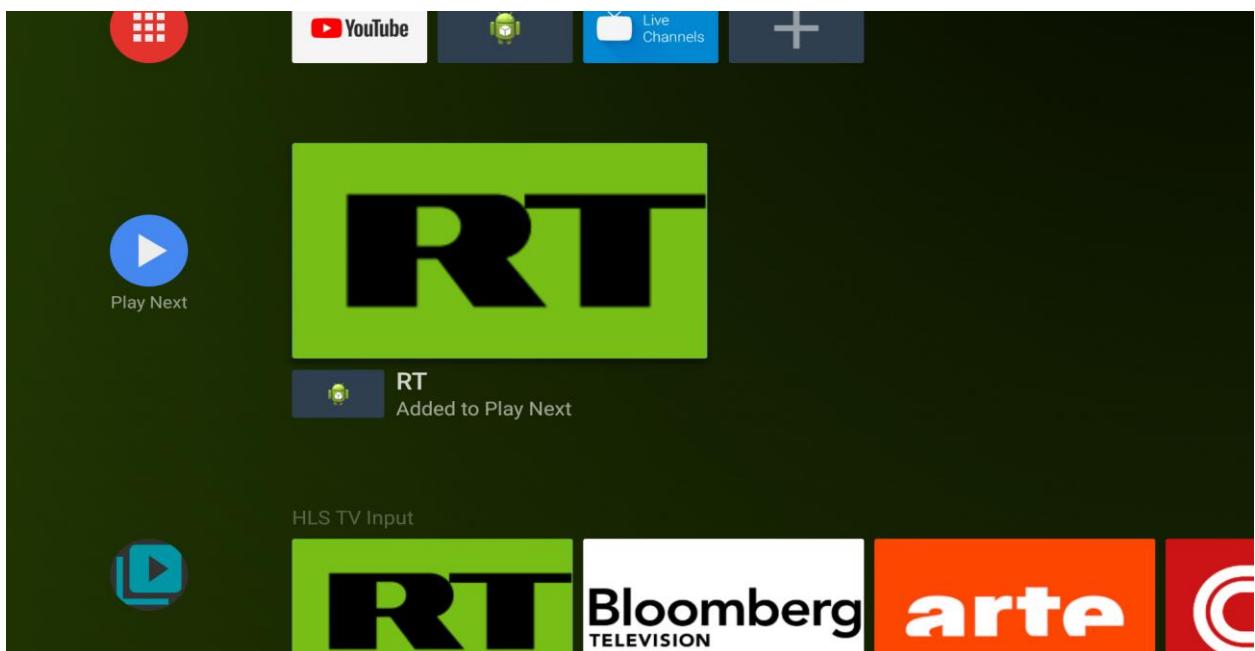
Slika 5.3 – Kanal na početnom ekranu, popunjena programima

- Zadržavanjem na nekom od programa, počinje da se emituje sadržaj TV kanala uživo:



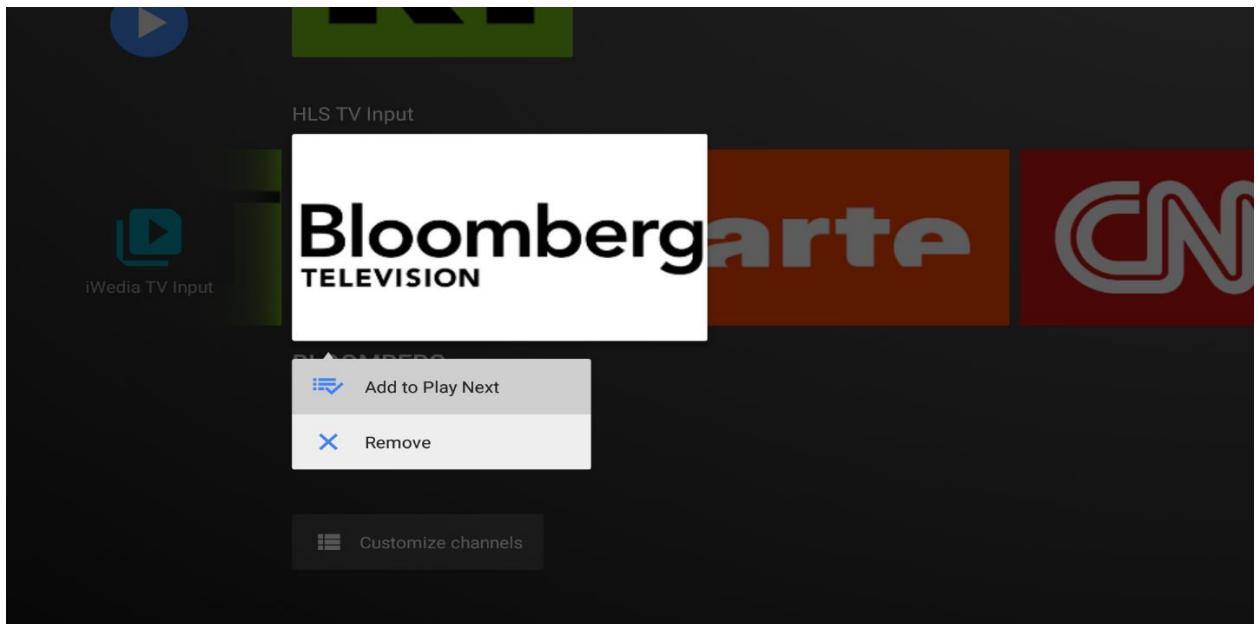
Slika 5.4 – Emitovanje TV sadržaja uživo, iz kanala na početnom ekranu

- Odabirom nekog od programa, pokreće se odgovarajući TV kanal u aplikaciji „Live Channels“. Izlaskom iz aplikacije Live Channels, u kanal Play Next dodat je poslednji kanal koji je korisnik gledao, kao predlog da nastavi sa gledanjem:



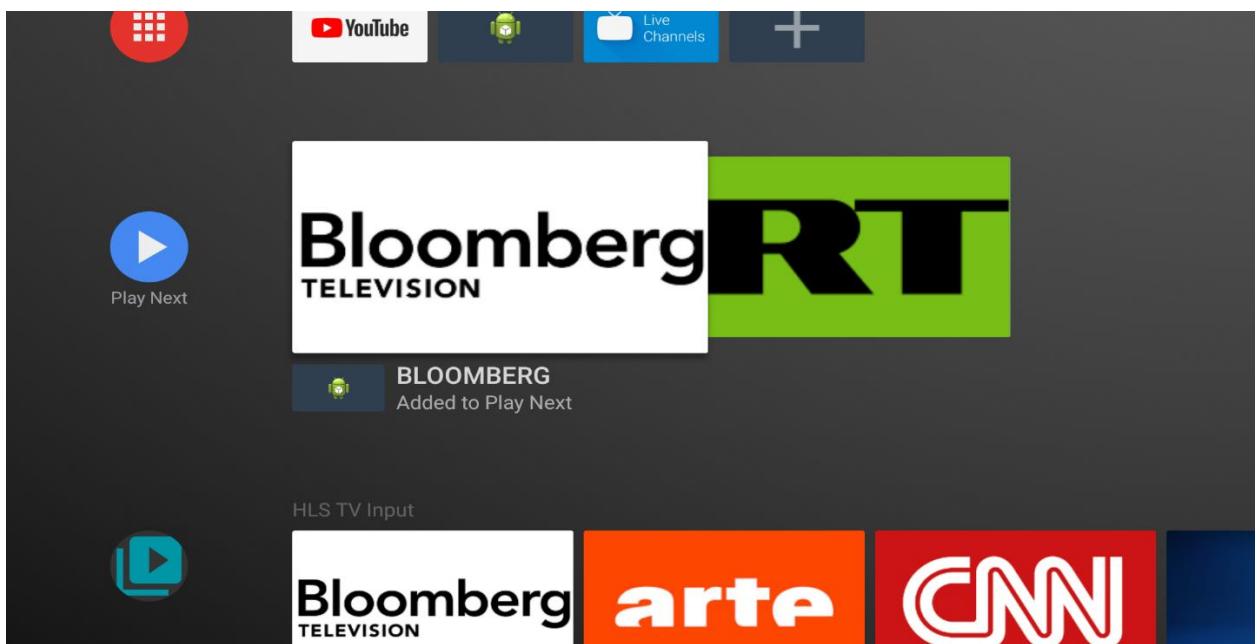
Slika 5.5 – „Watch Next“ kanal popunjjen poslednjim programom koji je korisnik gledao

- Moguće je dodati program u kanal „Watch Next“ i ručno, dugim pritiskom na program i odabirom odgovarajuće radnje:



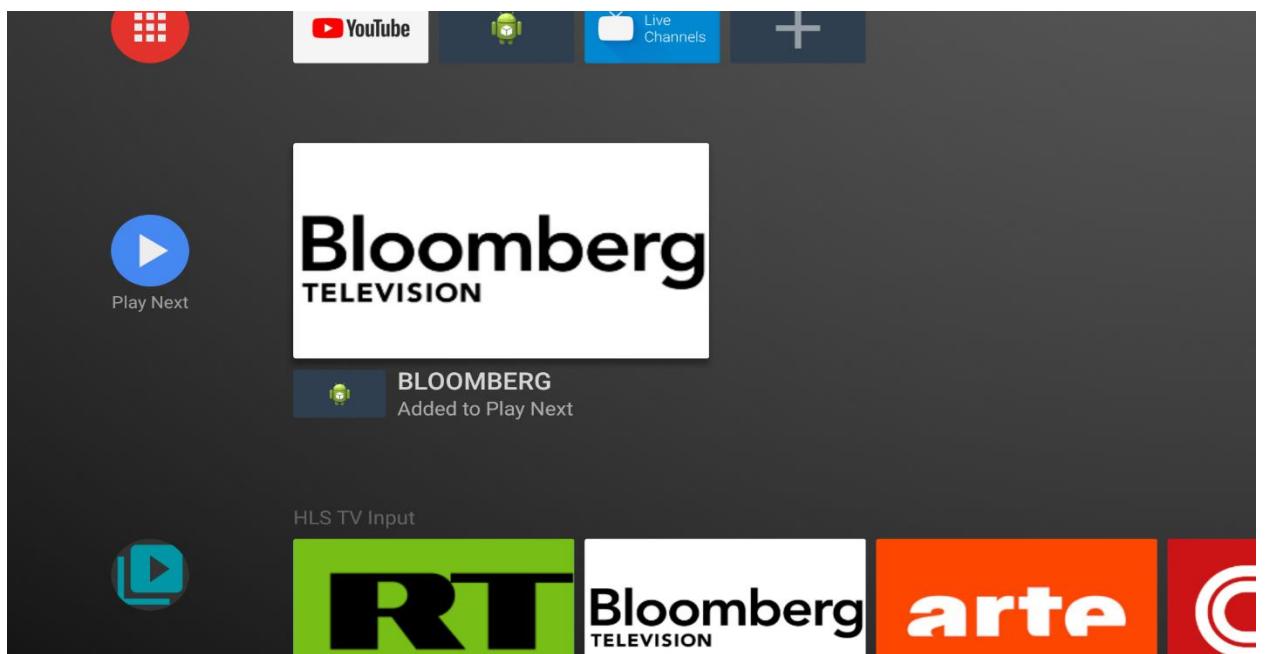
Slika 5.6 – Ručno dodavanje programa u „Watch Next“ kanal

- Nakon ponovnog pokretanja aplikacije „Live Channels“, prethodni program koji je poslednje gledan, zamenjen je novim, a program koji je korisnik dodaо ručno ostaje i dalje u „Watch Next“ kanalu:



Slika 5.7 – „Watch Next“ kanal popunjeno programskim dodatim i ručno dodatim programima

- Kanal koji je dodat ručno može biti obrisan iz „Watch Next“ kanala ručno, a ukoliko se kanal koji je korisnik poslednji gledao već nalazio u pomenutom kanalu iz nekog razloga, neće biti dodat ponovo, samo će se obrisati prethodni kanal koji je korisnik poslednji gledao:



Slika 5.8 – Nakon gore pomenute radnje, u kanalu „Watch Next“ ostaje samo jedan kanal

## 6. Zaključak

U ovom radu je realizovana programska podrška za „Program Preview“ funkcionalnost, odnosno proširen je postojeći TV ulaz tako da se na početnom ekranu Android TV-a vrši predlaganje TV sadržaja u okviru Launcher kanala. Kanal je popunjen programima, koji predstavljaju HLS TV kanale iz TV ulaza. Takođe, implementiran je „Watch Next“ mehanizam na način da se poslednji TV kanal koji je korisnik gledao iz „Live Channels“ aplikacije, doda u „Watch Next“ kanal koji je predstavljen grafičkim komponentama u drugom redu početnog ekrana. Na ovaj način je proširena podrška za predlaganje sadržaja krajnjim korisnicima, koja je inicijalno namenjena za multimedijalni sadržaj na zahtev, sa funkcionalnošću predlaganja TV sadržaja.

Dalji rad bi se zasnivao na proširenju ovog rešenja sa predlaganjem TV kanala dobavljenim na neki drugi način prenosa televizijskog signala uživo (DVB-C, DVB-T, DVB-S, IP, DASH... ), ili sa predlaganjem video sadržaja prethodno snimljenog pomoću digitalnog video snimača (eng. DVR).

## 7. Literatura

- [1] Tv Input Framework, <https://source.android.com/devices/tv>, jun 2018
- [2] Android TV, <https://www.android.com/tv/>, jun 2018
- [3] Previewing videos | Android Developers,  
<https://developer.android.com/training/tv/discovery/preview-videos>, jun 2018
- [4] Recommend TV content | Android Developers,  
<https://developer.android.com/training/tv/.../recommendations>, jun 2018
- [5] Dr Milan Z. Bjelica, dr Nikola Teslić, mr Velibor Mihić, Softver u televiziji i obradi slike  
1
- [6] W. Fischer, Digital Video and Audio Broadcasting Technology, A Practical Engineering Guide, Springer; 3rd ed. 2010
- [7] S. Pekowsky, R. Jaeger, „The se-top box as ulti-medi-terminal“, Consumer Electronics, IEEE Transactions on , vol.44, no.3, pp833,840, Aug 1998
- [8] G. O'Driscoll, „The Essential Guideto Digital Set-top Boxes and Interactive TV “, Printice Hall PTR, Ireland, 2000
- [9] T. Datta, „Digitall Covergence: the Set-top Box and its Digital Interface“, White paper, Enthink, July 1998
- [10] Android, <https://www.android..com/>, jun 2018
- [11] Android Open Source Project, <https://source.android.com/>, jun 2018
- [12] Android – Google TV Services, <http://www.androd.com/gtvs/>, jun 2018
- [13] Application | Android Developers,  
<https://developer.android.com/reference/android/app/Application>, jun 2018

- [14] M. Vidaković, N. Teslić, T. Maruna, V. Mihić, Android4TV: a proposition for integration of DTTV in Android Devices, IEEE 30th International Conference on Consumer Electronics (ICCE), Las Vegas, January 2012
- [15] CTS, <https://source.android.com/compatibility/cts/>, jun 2018
- [16] <https://github.com/iWedia/iWediaSimpleTvInputService>