



УНИВЕРЗИТЕТ У НОВОМ САДУ  
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У  
НОВОМ САДУ

---



Невена Јованов

**Реализација графичког корисничког  
интерфејса са 3Д подршком за  
дигитални сет-топ бокс уређај  
заснован на оперативном систему  
Андроид**

МАСТЕР РАД

Нови Сад, 2015



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

## КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

|                                                                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Редни број, <b>РБР</b> :                                                                 |                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Идентификациони број, <b>ИБР</b> :                                                       |                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Тип документације, <b>ТД</b> :                                                           | Монографска документација                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Тип записа, <b>ТЗ</b> :                                                                  | Текстуални штампани материјал                                                                                                                                                                                                                                                                                                                                                                                                                 |
| Врста рада, <b>ВР</b> :                                                                  | Дипломски – мастер рад                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Аутор, <b>АУ</b> :                                                                       | Невена Јованов                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Ментор, <b>МН</b> :                                                                      | доц. др Милан Бјелица                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Наслов рада, <b>НР</b> :                                                                 | Реализација графичког корисничког интерфејса са 3Д подршком за дигитални сет - топ бокс уређај заснован на оперативном систему Андроид                                                                                                                                                                                                                                                                                                        |
| Језик публикације, <b>ЈП</b> :                                                           | Српски / латиница                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Језик извода, <b>ЈИ</b> :                                                                | Српски                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Земља публикавања, <b>ЗП</b> :                                                           | Република Србија                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Уже географско подручје, <b>УГП</b> :                                                    | Војводина                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Година, <b>ГО</b> :                                                                      | 2015.                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Издавач, <b>ИЗ</b> :                                                                     | Ауторски репринт                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Место и адреса, <b>МА</b> :                                                              | Нови Сад; трг Доситеја Обрадовића 6                                                                                                                                                                                                                                                                                                                                                                                                           |
| Физички опис рада, <b>ФО</b> :<br>(поглавља/страна/ цитата/табела/слика/графика/прилога) | 7/57/0/5/29/0/1                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Научна област, <b>НО</b> :                                                               | Електротехника и рачунарство                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Научна дисциплина, <b>НД</b> :                                                           | Рачунарска техника                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Предметна одредница/Кључне речи, <b>ПО</b> :                                             | дигитална телевизија, графичка корисничка спрега, 3Д, Андроид платформа, пријемници дигиталног сигнала                                                                                                                                                                                                                                                                                                                                        |
| <b>УДК</b>                                                                               |                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Чува се, <b>ЧУ</b> :                                                                     | У библиотеци Факултета техничких наука, Нови Сад                                                                                                                                                                                                                                                                                                                                                                                              |
| Важна напомена, <b>ВН</b> :                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Извод, <b>ИЗ</b> :                                                                       | У овом раду описан је начин за реализацију ТВ апликације „следеће генерације“ на дигиталном сет-топ боксу, у прилагођеном графичком окружењу насталом сумирањем тржишних захтева. У циљу побољшања корисничког искуства коришћена је 3Д графика за имплементацију апликације кроз графичко окружење које не долази са Андроид платформом, већ је наменски развијено на софтверском стеку који садржи библиотеке за цртање нативне 3Д графике. |
| Датум прихватања теме, <b>ДП</b> :                                                       |                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Датум одбране, <b>ДО</b> :                                                               | 29.6.2015.                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Чланови комисије, <b>КО</b> :                                                            | Председник: проф. др Милан Видаковић                                                                                                                                                                                                                                                                                                                                                                                                          |
|                                                                                          | Члан: доц. др Данијела Лалић                                                                                                                                                                                                                                                                                                                                                                                                                  |
|                                                                                          | Члан, ментор: доц. др Милан Бјелица                                                                                                                                                                                                                                                                                                                                                                                                           |
|                                                                                          | Потпис ментора                                                                                                                                                                                                                                                                                                                                                                                                                                |



## KEY WORDS DOCUMENTATION

|                                                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                |
|----------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|----------------|
| Accession number, <b>ANO</b> :                                                               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                |
| Identification number, <b>INO</b> :                                                          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                |
| Document type, <b>DT</b> :                                                                   | Monographic publication                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                        |                |
| Type of record, <b>TR</b> :                                                                  | Textual printed material                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                        |                |
| Contents code, <b>CC</b> :                                                                   | Master Thesis                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                        |                |
| Author, <b>AU</b> :                                                                          | Nevena Jovanov                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                        |                |
| Mentor, <b>MN</b> :                                                                          | PhD Milan Bjelica                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                        |                |
| Title, <b>TI</b> :                                                                           | Implementation of graphical user interface with 3D support for Android based digital set-top box                                                                                                                                                                                                                                                                                                                                                                                                                    |                        |                |
| Language of text, <b>LT</b> :                                                                | Serbian                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                        |                |
| Language of abstract, <b>LA</b> :                                                            | Serbian                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                        |                |
| Country of publication, <b>CP</b> :                                                          | Republic of Serbia                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                        |                |
| Locality of publication, <b>LP</b> :                                                         | Vojvodina                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                        |                |
| Publication year, <b>PY</b> :                                                                | 2015.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                        |                |
| Publisher, <b>PB</b> :                                                                       | Author's reprint                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                        |                |
| Publication place, <b>PP</b> :                                                               | Novi Sad, Dositeja Obradovica sq. 6                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                        |                |
| Physical description, <b>PD</b> :<br>(chapters/pages/ref./tables/pictures/graphs/appendixes) | 7/57/0/5/29/0/1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                |
| Scientific field, <b>SF</b> :                                                                | Electrical Engineering                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                        |                |
| Scientific discipline, <b>SD</b> :                                                           | Computer Engineering, Engineering of Computer Based Systems                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                        |                |
| Subject/Key words, <b>S/KW</b> :                                                             | digital television, graphical user interface, Android platform, 3D, digital set-top box                                                                                                                                                                                                                                                                                                                                                                                                                             |                        |                |
| <b>UC</b>                                                                                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                |
| Holding data, <b>HD</b> :                                                                    | The Library of Faculty of Technical Sciences, Novi Sad, Serbia                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                        |                |
| Note, <b>N</b> :                                                                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                |
| Abstract, <b>AB</b> :                                                                        | This paper presents one implementation of „next generation“ TV application for digital set-top box, originated on summed up market requirements. To improve user experience and for higher responsiveness level, 3D graphics is rendered via custom framework that supports real 3D and is hardware accelerated. Implementation of the framework was conducted via libGDX library, that originally uses OpenGL ES library. Testing results show there are no memory leaks, user interface is fluent and responsive. |                        |                |
| Accepted by the Scientific Board on, <b>ASB</b> :                                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                |
| Defended on, <b>DE</b> :                                                                     | 29.6.2015.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                        |                |
| Defended Board, <b>DB</b> :                                                                  | President:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | Milan Vidaković, Ph.D. |                |
|                                                                                              | Member:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Danijela Lalić, Ph.D.  | Menthor's sign |
|                                                                                              | Member, Mentor:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Milan Bjelica, Ph.D.   |                |

## **Zahvalnost**

Zahvaljujem se mentoru doc. dr Milanu Bjelici na stručnoj pomoći i savetima za izradu ovog rada.

Posebno se zahvaljujem Nenadu Jovanoviću za utrošeno vreme i energiju u to da od nas za kratko vreme nastanu inženjeri i nekadašnjem C-More timu koji je prolazak kroz zajednička nova iskustva učinio nezaboravnim.

Zahvaljujem se svojim roditeljima, Miroslavu i Tatjani, na brizi i bezpogovornoj podršci tokom školovanja.

Na kraju, zahvaljujem se svim ljudima iz mog okruženja, jer su na direktan ili indirektan način doprineli izradi ovog završnog rada.

## SADRŽAJ

|                                                                                                     |    |
|-----------------------------------------------------------------------------------------------------|----|
| 1. Uvod.....                                                                                        | 8  |
| 2. Teorijske osnove .....                                                                           | 11 |
| 2.1 Digitalna televizija .....                                                                      | 11 |
| 2.1.1 Značaj grafičkog korisničkog interfejsa na prijemnicima digitalnog televizijskog signala..... | 11 |
| 2.2 Zahtevi moderne TV aplikacije.....                                                              | 12 |
| 2.2.1 Primeri rešenja grafičke korisničke sprege.....                                               | 12 |
| 2.3 Android platforma .....                                                                         | 14 |
| 2.3.1 Arhitektura Android sistema.....                                                              | 14 |
| 2.4 Realizacija TV aplikacije za Android OS .....                                                   | 15 |
| 3. Koncept rešenja.....                                                                             | 18 |
| 3.1 Uvod u koncept .....                                                                            | 18 |
| 3.2 Arhitektura sistema .....                                                                       | 18 |
| 3.2.1 Grafičko okruženje .....                                                                      | 19 |
| 3.2.1.1 Osnovni elementi grafičkog okruženja.....                                                   | 19 |
| 3.2.1.2 Pozicioniranje elemenata.....                                                               | 20 |
| 3.2.1.2.1 Pomeranje sa <i>Kamerom</i> .....                                                         | 20 |
| 3.2.1.3 Izvedeni elementi grafičkog okruženja .....                                                 | 20 |
| 3.2.1.3.1 Liste.....                                                                                | 22 |
| 3.2.1.4 Iscrtavanje elemenata pomocu grafičkog okruženja .....                                      | 22 |
| 3.2.1.4.1 Animacije .....                                                                           | 22 |
| 3.2.1.5 Interakcija korisnika sa elementima i međusobna interakcija elemenata....                   | 23 |
| 3.2.1.6 Resursi .....                                                                               | 24 |

---

|           |                                                                 |    |
|-----------|-----------------------------------------------------------------|----|
| 3.2.2     | TV aplikacija .....                                             | 25 |
| 4.        | Implementacija.....                                             | 27 |
| 4.1       | TV aplikacija .....                                             | 27 |
| 4.1.1     | <i>Info dijalog</i> .....                                       | 29 |
| 4.1.1.1   | <i>Info dijalog Vidžet-grupa</i> .....                          | 29 |
| 4.1.1.1.1 | Animacije u <i>Info dijalogu</i> .....                          | 30 |
| 4.1.1.2   | Stanja .....                                                    | 30 |
| 4.1.1.2.1 | „Info dijalog” stanje .....                                     | 30 |
| 4.1.1.2.2 | Stanje „Vremenske linije” .....                                 | 31 |
| 4.1.1.2.3 | Stanje „Liste kanala” .....                                     | 32 |
| 4.1.1.3   | Implementacija <i>Info dijaloga</i> u grafičkom okruženju ..... | 33 |
| 4.1.1.3.1 | Osluškivači <i>Info dijalog Vidžet-grupe</i> .....              | 35 |
| 4.1.1.4   | Upravljanje resursima.....                                      | 36 |
| 4.1.1.5   | Optimizacije .....                                              | 37 |
| 5.        | Testiranje i verifikacija .....                                 | 38 |
| 5.1       | Simulacija intenzivnog korišćenja aplikacije .....              | 38 |
| 5.2       | Broj slika u sekundi.....                                       | 43 |
| 5.3       | Iskorišćenost procesorske moći.....                             | 43 |
| 5.4       | Zauzeće radne memorije .....                                    | 44 |
| 5.5       | Zauzeće grafičke memorije .....                                 | 45 |
| 6.        | Zaključak .....                                                 | 46 |
| 7.        | Literatura.....                                                 | 47 |
|           | Prilog.....                                                     | 49 |

## SPISAK SLIKA

|                                                                                                                                                                                                                     |    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Slika 2.1 Prikaz snimka ekrana rešenja TV aplikacije NAGRA Open TV .....                                                                                                                                            | 13 |
| Slika 2.2 Prikaz snimka ekrana rešenja TV aplikacije na Android Lollipop platformi .....                                                                                                                            | 14 |
| Slika 2.3 Dijagram arhitekture Android sistema.....                                                                                                                                                                 | 15 |
| Slika 2.4 Prikaz Arhitekture sistema korišćenog za odabrano rešenje.....                                                                                                                                            | 16 |
| Slika 2.5 Prikaz arhitekture Android sistema i način na koji se odabrani sistem uklapa u Android sistem.....                                                                                                        | 17 |
| Slika 3.1 Detaljniji dijagram strukture TV aplikacije rešenja ovog rada.....                                                                                                                                        | 19 |
| Slika 3.2 Dijagram klasa osnovnih elemenata grafičkog okruženja .....                                                                                                                                               | 21 |
| Slika 3.3 Dijagram koji prikazuje hijerarhiju poziva funkcija za iscertavanje grafičkih elemenata.....                                                                                                              | 23 |
| Slika 3.4 Dijagram sekvence koji opisuje prosleđivanje ulaznih akcija .....                                                                                                                                         | 24 |
| Slika 3.5 Prikaz 3D Sveta, gde se vidi soba, par <i>Scena</i> i <i>Video Vidžet</i> .....                                                                                                                           | 25 |
| Slika 4.1 Struktura TV aplikacije opisana kroz <i>Scene</i> i <i>Vidžet-grupe</i> koje sadrži.....                                                                                                                  | 27 |
| Slika 4.2 Dijagram stanja TV aplikacije koji prikazuje tranzicije između <i>Scena</i> , predstavljen kao automat sa konačnim brojem stanja, gde su okidači tranzicija kodovi tastera na daljinskom upravljaču ..... | 28 |
| Slika 4.3 Dijagram stanja <i>Info dijaloga</i> .....                                                                                                                                                                | 30 |
| Slika 4.4 <i>Info dijalog</i> u inicijalnom stanju .....                                                                                                                                                            | 31 |
| Slika 4.5 <i>Info dijalog</i> u <i>Sceni vremenske linije</i> .....                                                                                                                                                 | 32 |
| Slika 4.6 <i>Info dijalog</i> u <i>Sceni liste kanala</i> .....                                                                                                                                                     | 32 |
| Slika 4.7 Dijagram klasa <i>Info dijaloga</i> .....                                                                                                                                                                 | 33 |
| Slika 5.1 Dijagram prolazaka kroz <i>Scene</i> prilikom simulacije korišćenja aplikacije .....                                                                                                                      | 39 |
| Slika 5.2 <i>Glavni meni scena</i> .....                                                                                                                                                                            | 39 |

---

|                                                                                                          |    |
|----------------------------------------------------------------------------------------------------------|----|
| Slika 5.3 <i>Scena programskog vodiča</i> .....                                                          | 40 |
| Slika 5.4 <i>Video na zahtev scena</i> .....                                                             | 40 |
| Slika 5.5 <i>Scena sa detaljima</i> .....                                                                | 41 |
| Slika 5.6 <i>Scena sa dodatnim sadržajem</i> .....                                                       | 41 |
| Slika 5.7 <i>Scena sa dodatnim aplikacijama</i> .....                                                    | 42 |
| Slika 5.8 <i>Scena sa podešavanjima</i> .....                                                            | 42 |
| Slika 5.9 Grafik broja slika u sekundi (FPS) u trajanju od sat vremena korišćenja aplikacije .....       | 43 |
| Slika 5.10 Grafik iskorišćenja centralnog procesora u trajanju od sat vremena korišćenja aplikacije..... | 44 |
| Slika 5.11 Grafik iskorišćenja memorije u trajanju od sat vremena korišćenja aplikacije ..               | 44 |
| Slika 5.12 Grafik iskorišćenja grafičke memorije u trajanju od sat vremena korišćenja aplikacije.....    | 45 |

## SPISAK TABELA

|                                                                                                                                                                                       |    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Tabela 2.1 Lista modernih TV funkcionalnosti sa opisom .....                                                                                                                          | 12 |
| Tabela 4.1 Prikaz metoda klase <i>ChannelListWidgetGroup.java</i> koje se pozivaju prilikom tranzicije iz <i>Scene</i> u <i>Scenu</i> .....                                           | 34 |
| Tabela 4.2 Prikaz metoda klase <i>TimelineWidgetGroup.java</i> koje se pozivaju prilikom tranzicije iz <i>Scene</i> u <i>Scenu</i> .....                                              | 35 |
| Tabela 4.3 Metode klase <i>InfoBannerWidgetGroup.java</i> korišćene prilikom prelaska iz <i>Scene</i> u <i>Scenu</i> , kao i za pojavljivanje i nestajanje <i>Info dijaloga</i> ..... | 36 |
| Tabela 5.1 Prikaz zauzeća memorije pre početka i nakon završetka rada aplikacije u bajtima .....                                                                                      | 45 |

## SKRAĆENICE

|            |                                                                         |
|------------|-------------------------------------------------------------------------|
| <b>TV</b>  | - <i>Television</i> , Televizija                                        |
| <b>STB</b> | - <i>Set-Top Box</i> , set-top boks                                     |
| <b>CPU</b> | - <i>Central Processor Unit</i> , Centralni procesor                    |
| <b>GPU</b> | - <i>Graphical Processor Unit</i> , Grafički procesor                   |
| <b>LRU</b> | - <i>Least Recently Used</i> , Najdalje korišćen                        |
| <b>RAM</b> | - <i>Random Access Memories</i> , memorija sa slučajnim pristupom       |
| <b>FPS</b> | - <i>Frames per Second</i> , broj slika u sekundi                       |
| <b>OS</b>  | - <i>Operating System</i> , operativni sistem                           |
| <b>OTT</b> | - <i>Over The Top</i> , sadržaji koji se prenose Internet vezom         |
| <b>TTM</b> | - <i>Time To Market</i> , vreme do pojavljivanja novih tržišnih zahteva |

## 1. Uvod

Televizijski uređaji gube svoju dominantnu ulogu u domaćinstvima, zahvaljujući rasprostranjenosti prenosivih uređaja (tablet, mobilni telefon) i raznih pogodnosti poput Interneta na njima. Vraćanje korisnika TV prijemniku kao centralnom uređaju u domu, izaziva pojavu određenih zahteva za „sledećom generacijom” grafičkih korisničkih interfejsa na TV aplikacijama kao što je hitar odziv na korisničku akciju, fluidnost aplikacije, određene moderne TV funkcionalnosti, OTT servisi, socijalne mreže i opštenamenske aplikacije.

Razvoj infrastrukture za prenos digitalnog televizijskog signala, omogućio je i razvoj tehnologija za prikaz informacija koje stižu krajnjem korisniku. Kako su set-top boks uređaji napredovali od prvobitnog pojavljivanja, širio im se opus funkcionalnosti. Osnovne aplikacije koje moderan set-top boks mora da obezbedi su, na primer, info dijalog sa listom kanala, elektronski vodič kroz program, video na zahtev, snimanje sadržaja, gledanje unazad, pauziranje i premotavanje, opštenamenske aplikacije i sl.

Grafički korisnički interfejsi su se takođe menjali. Od prvobitnih 2D okruženja, bez ili sa veoma malo animacija, relativno sporog odziva na korisničku akciju dolazimo do današnjih „pametnih” prijemnika digitalnog signala koji imaju svoje operativne sisteme i izvršavaju TV aplikacije, a podržavaju čak i instalaciju proizvoljnih aplikacija među kojima su, na primer, igre.

Iako se Android platforma [1] pojavila prvenstveno kao programska podrška za mobilne telefone sa ekranom osetljivim na dodir, današnja verzija ovog operativnog sistema [2] omogućava brz izlazak novog TV rešenja na tržište sa svim funkcionalnostima koje se zahtevaju od novih STB uređaja i TV prijemnika. Jedna od njegovih prednosti je što se aplikacije pišu u programskom jeziku Java [3][4], pa je rad sa memorijom olakšan po pitanju zauzimanja i oslobađanja iste. Sadrži i određen broj gotovih grafičkih komponenti. Ovakva rasprostranjenost proizilazi iz činjenice da je otvorenog koda (engl. *open-source*) [5]. To omogućava proizvođačima uređaja da ga menjaju i prilagođavaju svojim potrebama. Uticaj operativnog

sistema Android se širio, pa danas postoji značajan broj aplikacija pravljenih za ovu platformu, a ona se nalazi na raznim uređajima: mobilnim i fiksnim telefonima, tabletima, kućnim aparatima, televizijskim uređajima i sl.

Zbog pogodnosti koje nudi Android platforma, cilj ovog rada je formiranje grafičkog korisničkog interfejsa za prijemnik digitalnog televizijskog signala (digitalni set-top boks uređaj [6]) sa Android sistemom, i koji zadovoljava uslove tržišta i koristi hardverske mogućnosti (fizičke arhitekture) današnjih platformi.

Osnovna prepreka realizaciji modernih 3D efekata [7][8] i animacija, a samim tim i potpunog skupa savremenih funkcionalnosti i brzine odziva na naredbe korisnika, bila je nepostojanje adekvatne grafičke podrške u Android OS. Kako Android OS ima svoje 2D grafičko okruženje, glavno ograničenje je simulirati 3D efekte, što bi zahtevalo više matematičkih operacija i samim tim i više procesorskog vremena.

Postoji način da se pri implementaciji grafičkih komponenata ne koristi grafičko okruženje, već da se korisnički interfejs realizuje u biblioteci OpenGL ES [9]. Ova biblioteka neposredno upravlja fizičkim komponentama računara i implementacija grafike pomoću nje iziskuje nezanemarljivu količinu vremena za upoznavanje sa njenim programskim interfejsom.

Kako vreme do dolaska inovacija (eng. *time to market*) predstavlja kratak period, odlučeno je da se koristi apstrakcija biblioteke Open GL ES, a to je biblioteka libGDX [10]. Biblioteka libGDX predstavlja osnov grafičkog okruženja koje u potpunosti podržava 3D grafiku i ima zadovoljavajuću brzinu odziva [11], i realizovano je u okviru istog projekta kao i TV aplikacija ovog rada [12].

Rad sadrži sedam poglavlja.

Prvo poglavlje sadrži uvod u kom su izneseni problemi i okolnosti poput stanja u oblasti i na tržištu koji su doveli do izrade ovog rada.

Drugo poglavlje iznosi osnovne informacije o digitalnoj televiziji i korisničkim interfejsima na digitalnim prijemnicima, zahtevima moderne TV aplikacije uz izložene primere, Android platformi i njenoj arhitekturi i daje odgovarajuću diskusiju.

U okviru trećeg poglavlja je postavljeno grafičko okruženje na osnovu koga je TV aplikacija realizovana. Izložena je arhitektura kompletnog sistema i objašnjeno je konceptualno rešenje TV aplikacije.

Četvrto poglavlje govori o implementaciji TV aplikacije kroz primer info dijaloga, koji je detaljno opisan.

Peto poglavlje se odnosi na testiranje performansi sistema radi utvrđivanja validnosti rešenja ovog rada.

Zaključak sadrži kratak pregled implementiranog rešenja sa mogućnostima daljeg napretka.

## **2. Teorijske osnove**

### **2.1 Digitalna televizija**

Osnovna karakteristika digitalne televizije [13] predstavlja prenos audio i video zapisa sa dodatnim informacijama u digitalnom formatu. Digitalni prenos obezbeđuje bolji kvalitet slike i zvuka.

Kako su računari i savremene telekomunikacione mreže zasnovani takođe na digitalnoj tehnologiji, ove dve oblasti se sa digitalnom televizijom mogu kombinovati. Kao rezultat kombinovanja digitalna televizija podržava interaktivne funkcionalnosti kao što su video na zahtev, odloženo gledanje, premotavanje, pauziranje i razne druge.

#### **2.1.1 Značaj grafičkog korisničkog interfejsa na prijemnicima digitalnog televizijskog signala**

Korisnički interfejs [14] postoji da olakša komunikaciju čoveka sa uređajem i učini je efikasnom uz što bolje korisničko iskustvo [15]. Minimalan trud korisnika treba da rezultuje željenim izlazom na ekranu.

Ugođaj korisnika je cilj dizajna korisničkog interfejsa. Industrija ulaže u dizajn zbog njegovog uticaja na prodaju proizvoda. Počelo je razdoblje 3D dizajna [8] u razvoju korisničkih interfejsa za TV prijemnike. Kako do sada to nije bio slučaj, na tržište su tek počela da izlaze rešenja u 3D-u. Fizičke komponente današnjih platformi omogućavaju ispunjenje modernih zahteva tržišta, ali iskorišćenje tih mogućnosti je izazov za moderna softverska rešenja.

Ono što osim dizajna utiče na korisničko iskustvo su funkcionalnosti televizijske aplikacije, pa tako opus usluga koje ona pruža korisniku postaje sve veći. Ideja pri implementiranju ovakvih aplikacija, je da prijemnik digitalnog signala preuzme mesto određenog

broja korisničkih uređaja kao što su mobilni telefon, tablet računar ili računar ili da omogući njihovo uparivanje i zajedničko korišćenje.

## 2.2 Zahtevi moderne TV aplikacije

Da bi se realizovala TV aplikacija sledeće generacije neophodno je ispratiti sledeće uslove:

1. brz odziv na korisničku akciju [11];
2. dizajn koji ima tok, svi delovi aplikacije su logično povezani;
3. savremene funkcionalnosti (Tabela 2.1);
4. moderan izgled sa efektima i animacijama (3D).

| Naziv funkcionalnosti                                    | Kratak opis                                                                                                    |
|----------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| <i>Info banner</i> (info dijalog)                        | nudi informacije o trenutnoj TV emisiji uz moguće dodatne informacije                                          |
| Lista kanala                                             | sadrži spisak TV kanala na TV uređaju                                                                          |
| <i>EPG</i> (elektronski vodič kroz program)              | prikazuje vremenski raspored TV emisija uz kratak opis u narednom periodu                                      |
| <i>VoD</i> (video na zahtev)                             | na zahtev, korisniku omogućava gledanje video sadržaja koji ne moraju biti povezani sa televizijskim programom |
| <i>Catch up TV</i> (vremenski ograničen video na zahtev) | omogućava gledanje TV programa nekoliko dana unazad                                                            |
| Snimanje                                                 | ostavlja mogućnost snimanja TV programa radi kasnijeg gledanja                                                 |
| Pauziranje i premotavanje                                | dozvoljava korisniku zaustavljanje TV emisije u toku digitalnog prenosa, kao i premotavanje                    |
| Preporuka sadržaja                                       | preporučivanje TV emisija ili drugih sadržaja na osnovu korisničkih preferencija                               |
| Korišćenje aplikacija                                    | dozvoljava korišćenje opštenamenskih aplikacija                                                                |
| Pristup Internet sadržaju                                | dozvoljava direktan pristup Internet stranicama i servisima                                                    |

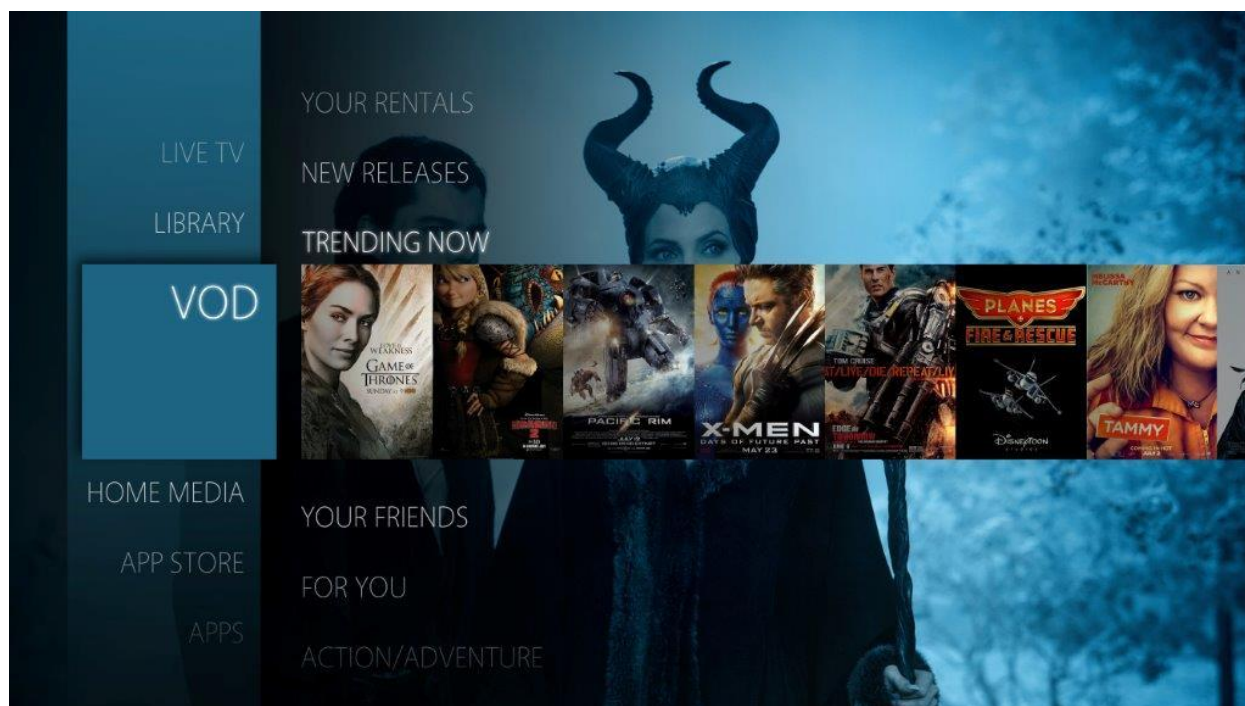
Tabela 2.1 Lista modernih TV funkcionalnosti sa opisom

### 2.2.1 Primeri rešenja grafičke korisničke sprege

U tekstu koji sledi biće izložena dva rešenja TV aplikacije sledeće generacije.

Korisnički interfejs zasnovan na internet pregledaču je najkorišćeniji metod u razvoju set-top bokseva i TV uređaja. Primer TV aplikacije sa ovakvim pristupom rada je NAGRA OpenTV

[16] (Slika 2.1). Ovakva vrsta aplikacije predstavlja HTML5 stranicu koja je napisana u skript jeziku Javascript i ona se korisniku prikazuje kroz internet pregledač, što znači da je neophodna



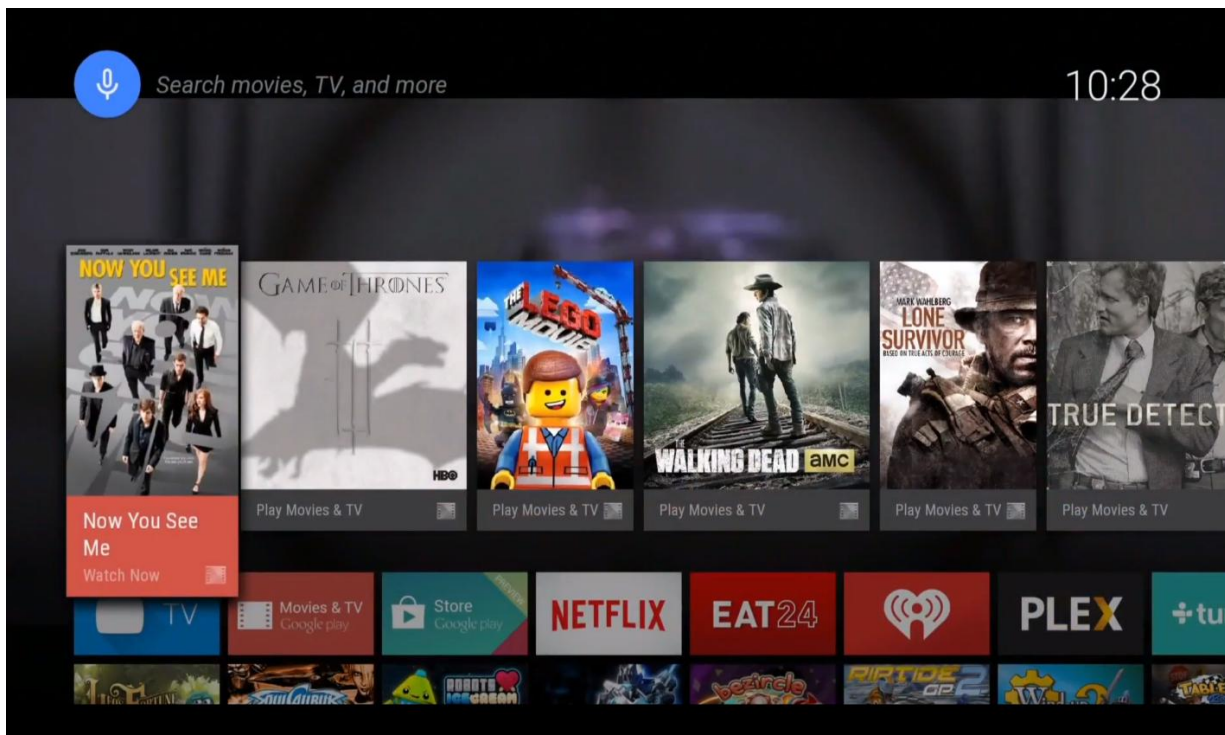
Slika 2.1 Prikaz snimka ekrana rešenja TV aplikacije NAGRA Open TV

programska podrška i platforma da bi ovakav sistem funkcionisao. Iako je velika prednost ovog pristupa brza izmena i unapređivanje korisničkog dizajna jer su sve informacije na serveru, činjenica je da se brzina odziva na ovakvom tipu korisničkog interfejsa može poboljšati samo do određene granice, upravo zbog problema koji se odnose na asinhronost učitavanja i nepredvidivost izgleda stranica.

Kompanija Google koja vodi razvoj Android sistema je u najnovijoj verziji ove platforme dodala podršku za TV funkcionalnosti. Najnovija verzija, nazvana Android Lollipop [2], predstavlja drugi primer moderne TV aplikacije (Slika 2.2). Kako je ova korisnička sprega realizovana u programskom jeziku Java, programeru su omogućene razne biblioteke i sijaset gotovih grafičkih komponenata za pravljenje aplikacija. Uzimajući u obzir koliko je Android danas rasprostranjen postoji i veliki broj već napisanih aplikacija za ovaj sistem. Međutim, sve grafičke komponente Androida imaju određen dizajn koji je u dve dimenzije, pa tako ukoliko bi bilo potrebe za 3D izgledom ili animacijama kao što tržište započinje, jedini način bi bio da se simuliraju. Simulacija 3D prostora u 2D prostoru zahteva više računanja, a samim tim i više procesorskog vremena nego da se radi o pravom 3D-u. Ovo može da uspori odziv i time naruši korisnički doživljaj.

## 2.3 Android platforma

Iako je prvenstveno nastao kao programska podrška za personalne uređaje poput mobilnih telefona i tablet računara i za ARM arhitekturu (engl. *Advanced RISC Machine*), sa porastom popularnosti prilagođen je i za velike ekrane koji nisu upravljani dodirom, kao na primer TV uređaj i za x86 i MIPS arhitekturu. Danas postoji nekoliko rešenja TV aplikacija napisanih na Android operativnom sistemu [17][18].



Slika 2.2 Prikaz snimka ekrana rešenja TV aplikacije na Android Lollipop platformi

Android je otvorena programska platforma za razvoj aplikacija. Distribucija Androida počinje 2007. godine, sa osnivanjem „Open Handset” alijanse, konzorcijuma sastavljenog od 86 proizvođača fizičke arhitekture, programske podrške kao i kompanija koje se bave telekomunikacijama posvećenih unapređenju otvorenih standarda za mobilne uređaje.

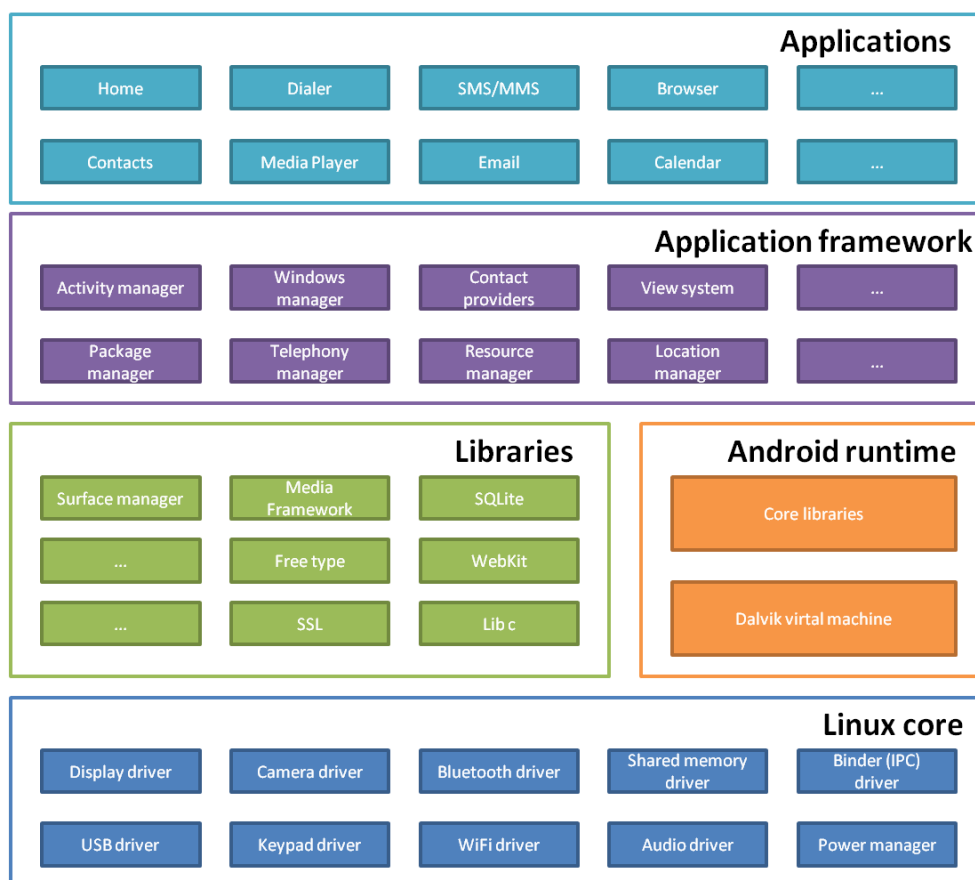
Iako ni današnji Android sistem ne poseduje DTV podršku, proširen je mehanizmima za integraciju iste, za razliku od starijih verzija. U okviru Android Java radnog okruženja postoji odgovarajući podsistem za integraciju DTV podrške, tako da je moguće ostvariti primarnu televizijsku funkcionalnost.

### 2.3.1 Arhitektura Android sistema

Arhitektura Android platforme je organizovana u više slojeva softvera (Slika 2.3) na sledeći način:

- Linuks jezgro (kernel) predstavlja osnovu softvera,
- izvršno okruženje u kom se nalaze Dalvik virtuelna mašina i glavne biblioteke,

- okruženje za aplikacije i
- same aplikacije.



Slika 2.3 Dijagram arhitekture Android sistema

## 2.4 Realizacija TV aplikacije za Android OS

Budući da je Android OS platforma za TV aplikaciju opisano u ovom radu, implementacija 3D grafičkog interfejsa nudi više načina za izradu.

Mogućnosti za realizaciju grafičkog interfejsa [19] su sledeće:

1. grafičko okruženje Android OS;
2. preko Internet pregledača u obliku HTML/HTML5 stranica;
3. nativna realizacija preko OpenGL ES biblioteke.

Pri implementaciji korisničkog interfejsa preko grafičkog okruženja Android operativnog sistema, određenu količinu vremena bi oduzele izmene gotovih grafičkih komponenata, kao i računanje pozicija elemenata tako da 2D izgleda kao 3D, gde nije uračunat rad sa resursima i preraspoređivanje procesorskog vremena utrošenog u sva navedena računanja. Sa obzirom na zahtev da grafika „teče” – bez zaustavljanja i na mane Android grafičkog okruženja – 2D prostor i komponente sa već gotovim dizajnom i na potreban trud uložen u simuliranje 3D efekata, prvi način implementacije korisničkog interfejsa je neadekvatan za rešenje opisano u ovom radu.

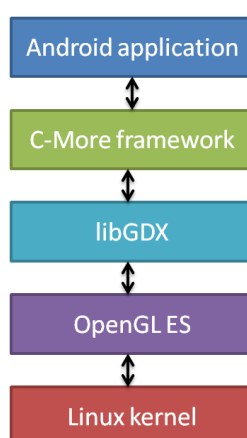
Kako se nedostaci implementacije preko Internet pregledača najviše odnose na robusnost i korisničko iskustvo [19], a onemogućen je veliki broj specifičnih grafičkih rešenja i animacija (uz ograničenja na 2D i skromne mogućnosti tranzicija), ni način implementacije preko pregledača nije adekvatan za rešenje opisano u ovom radu.

Nativna realizacija podrazumeva rad sa fizičkim komponentama, jer može se reći da OpenGL ES predstavlja programski interfejs fizičke arhitekture platforme. Samim tim rad sa Open GL ES bibliotekom bi oduzeo dosta vremena programerima koji treba da prvo da se upoznaju sa njom. Takođe, nativan način realizacije rezultuje korisničkom spregom koja je zavisna od platforme, odnosno nije prenosiva. Ipak, prednost ovog pristupa je hardversko ubrzanje pri crtanju grafike jer pomenuta biblioteka ima podršku za 3D.

Vreme koje nam tržište nudi za izradu rešenja sledeće generacije dok se ne pojave novi trendovi i zahtevi, odgovara poslednjem rešenju uz kompromis: biće korišćena grafička biblioteka koja apstrahuje programski interfejs OpenGL ES biblioteke, a opet dozvoljava slobodu da na njoj bude napravljeno prilagođeno grafičko okruženje zahtevima dizajna sledeće generacije, u kom će biti realizovana TV aplikacija ovog rada.

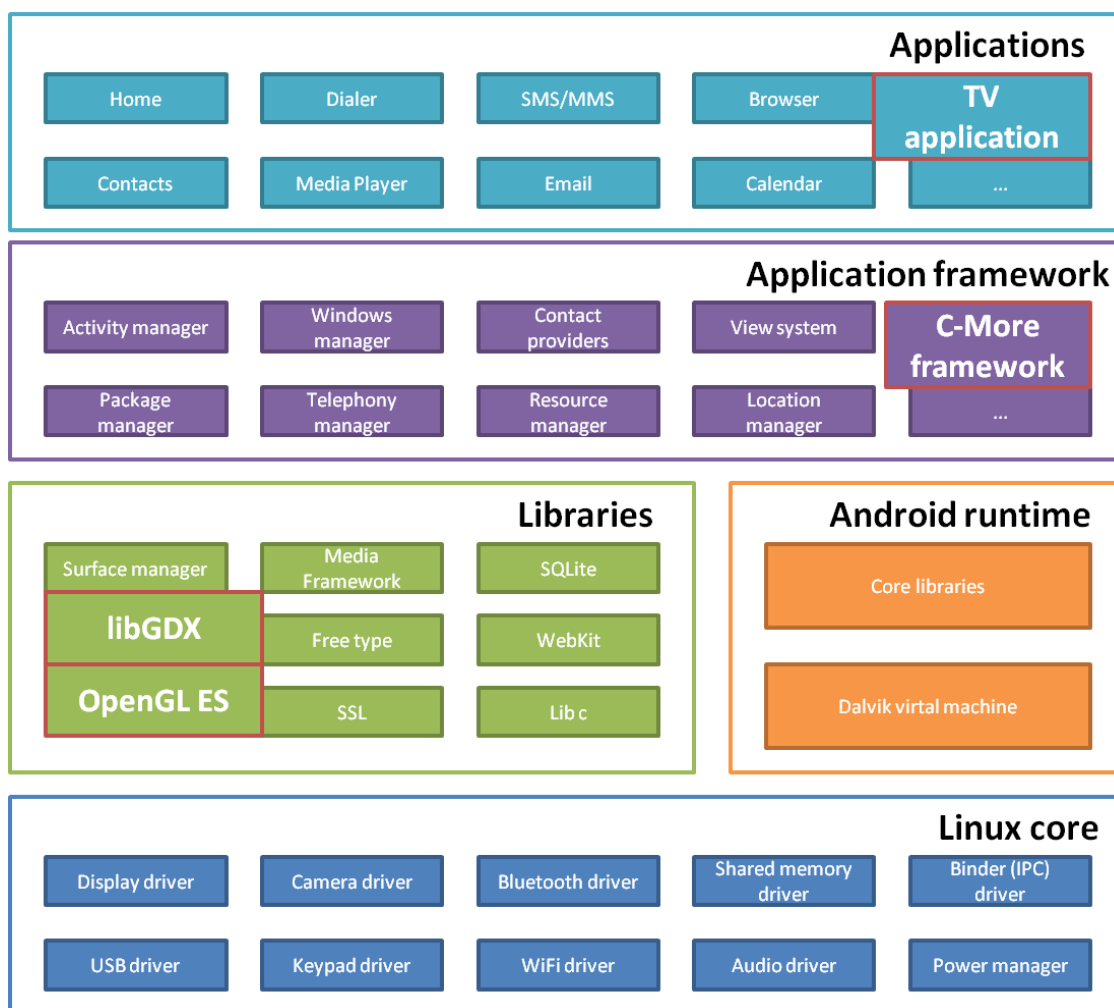
Biblioteka za apstrakciju OpenGL ES programske sprege je birana tako da podržava 3D prostor, da je otvorenog koda i da se i dalje razvija, da postoji određen broj prethodno realizovanih aplikacija pomoću nje, poput igara u kojima se vide efekti koje ona omogućava i da je njen programski interfejs dovoljno jasan za korišćenje i za potrebe realizovanog grafičkog okruženja. Odabrana je biblioteka libGDX, koja je prenosiva na sledeće platforme: Android, BlackBerry, iOS i HTML5 i ispunjava navedene zahteve.

Arhitektura odabranog sistema se nalazi na Slici 2.4.



Slika 2.4 Prikaz Arhitekture sistema korišćenog za odabrano rešenje

Način na koji je odabrano rešenje za implementaciju TV aplikacije enkapsulirano u Android sistem se nalazi na Slici 2.5.



Slika 2.5 Prikaz arhitekture Android sistema i način na koji se odabrani sistem uklapa u Android sistem

## 3. Koncept rešenja

### 3.1 Uvod u koncept

Grafički korisnički interfejs za TV aplikaciju realizovanu u ovom radu zasniva se na grafičkom okruženju koje se zove *C-More Framework* [20], nastalog u okviru istog projekta kao i aplikacija. Da bi koncept i implementacija aplikacije bili jasni, potrebno je proći kroz osnovne delove grafičkog okruženja u kom je realizovana.

U ovom poglavlju biće predstavljeno okruženje u kom je realizovana TV aplikacija i biće izložen osnovni koncept TV aplikacije opisane u ovom radu.

### 3.2 Arhitektura sistema

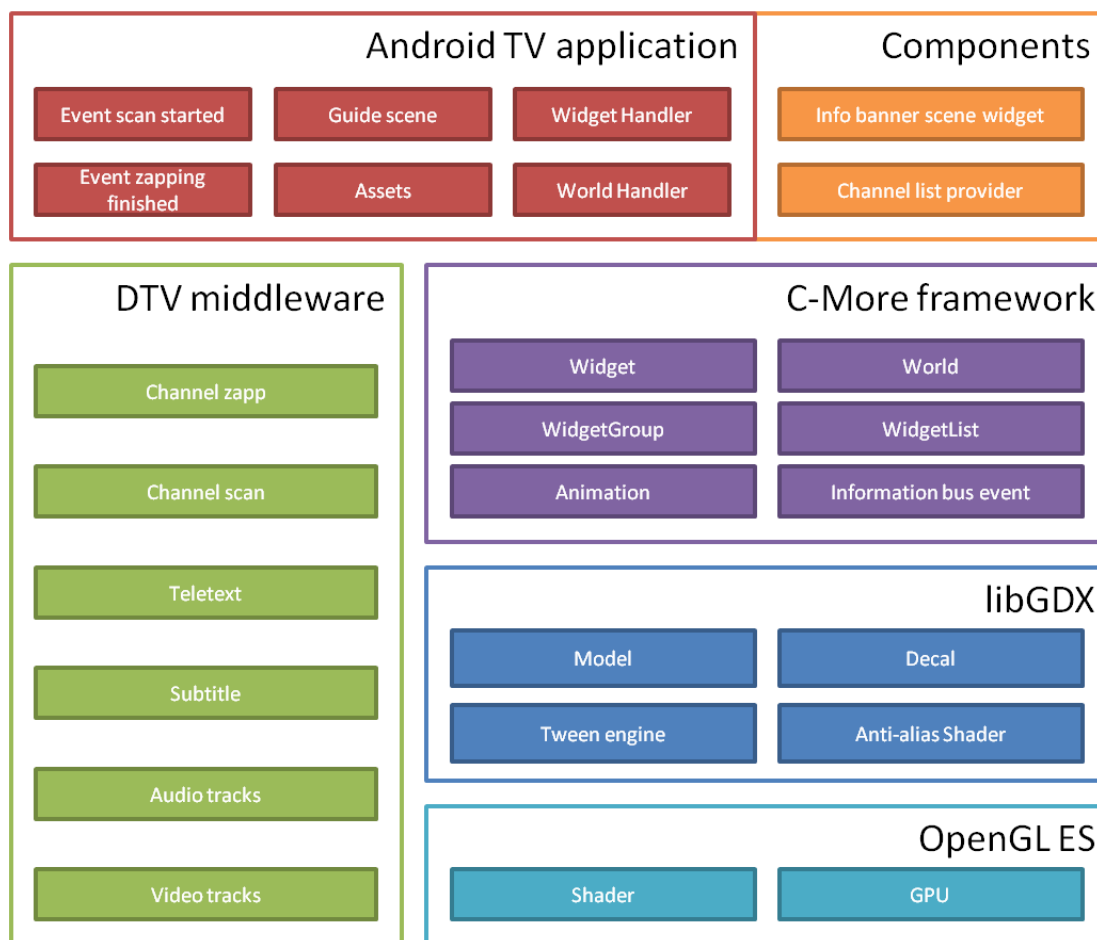
Kao što je objašnjeno u poglavlju II, TV aplikacija je realizovana korišćenjem Android platforme. Nije korišćeno grafičko okruženje Androida zbog nedostataka poput ograničenosti na gotove grafičke komponente i njihov dizajn, neprilagođenosti pomenutih gotovih grafičkih komponentata za TV funkcionalnosti, manjka podrške za 3D efekte, premalo prostora za inovacije u prostoru animacija, izgleda i ponašanja grafičke sprege. U tom smislu je omogućen tok podataka kroz Android platformu, kroz koji se iscrtava grafika.

Grafičko okruženje koje je korišćeno nudi neke osnovne grafičke komponente, kao na primer za prikaz teksta, slike i video toka, indikator napredovanja, dugme, više verzija liste i slično. Dizajn komponentata nije zadat okruženjem, već ga je potrebno definisati. Ulazni podaci poput pritiska tastera na daljinskom upravljaču se obrađuju i njima se rukuje kroz grafičko okruženje.

C-More Framework okruženje je realizovano pomoću libGDX biblioteke i poziva njene funkcije za iscrtavanje objekata. Takođe, koristi usluge DTV srednjeg sloja, gde su realizovane

TV funkcionalnosti, što nije deo koji je obuhvaćen u ovom radu. U okviru biblioteke libGDX koristi se ekstenzija Tween engine, modul koji rukuje animacijama.

Biblioteka libGDX poziva funkcije OpenGL ES biblioteke ili joj prosleđuje programe optimizovane za GPU (eng. *Graphics Processing Unit*).



Slika 3.1 Detaljniji dijagram strukture TV aplikacije rešenja ovog rada

Ovakvom hijerarhijom i strukturom softverskog steka se dobija hardversko ubrzanje zbog toga što OpenGL ES biblioteka ima podršku za pravi 3D i za grafiku koristi GPU, koje bi se kod iscrtavanja korišćenjem Android operativnog sistema svelo na CPU (eng. *Central Processing Unit*). Kako bi CPU morao da radi računice radi simulacije 3D efekata, ovo bi moglo da uspori odziv na korisničku akciju.

Pregled arhitekture sistema se nalazi na Slici 3.1.

### 3.2.1 Grafičko okruženje

#### 3.2.1.1 Osnovni elementi grafičkog okruženja

Osnovni element grafičkog okruženja je *Vidžet* (eng. *widget*). Sve što se iscrtava i zahteva interakciju sa korisnikom nasleđuje *Vidžet*, koji je apstraktna klasa.

*Vidžet-grupa* (eng. *widget group*) je takođe apstraktna klasa koja nasleđuje *Vidžet*, a ona može da sadrži proizvoljan broj potomaka istog tipa. Jedno dete u jednom momentu može da bude aktivno, što će kroz dalji tekst biti razjašnjeno.

Ovi *Vidžeti* i *Vidžet-grupe*, koji takođe nasleđuju *Vidžete*, se registruju u *Scene* (eng. *scene*). Klasa *Scena* nasleđuje *Vidžet-grupu* i predstavlja jednu poziciju kamere u 3D *Svetu*.

Sve *Scene* pripadaju *Svetu*. *Svet* (eng. *world*) je klasa koja rukuje *Scenama*. U aplikaciji postoji jedna instanca klase *Svet* i tačno jedna *Scena* je aktivna u datom trenutku. Mehanizam pomeranja kamere sa scene na scenu se nalazi u klasi *Svet*.

*Kamera* je poseban element grafičkog okruženja, koji predstavlja 2D projekciju 3D prostora na ekranu.

Opisana hijerarhija je prikazana na dijagramu klasa na Slici 3.2.

### 3.2.1.2 Pozicioniranje elemenata

Svaki *Vidžet* ima dve vrednosti pozicije; *pivot poziciju* i *relativnu poziciju*.

Zbir ove dve vrednosti je *apsolutna pozicija* u 3D *Svetu*. To je njegova stvarna pozicija i na toj lokaciji se element iscrtava.

*Pivot pozicija* *Vidžeta* je zapravo *apsolutna pozicija* njegovog pretka.

*Relativna pozicija* predstavlja pomeraj u odnosu na apsolutnu poziciju pretka.

*Vidžeti* imaju i vrednost *rotacije*. *Rotacija* pretka izaziva promenu *pivot pozicije* potomaka.

#### 3.2.1.2.1 Pomeranje sa *Kamerom*

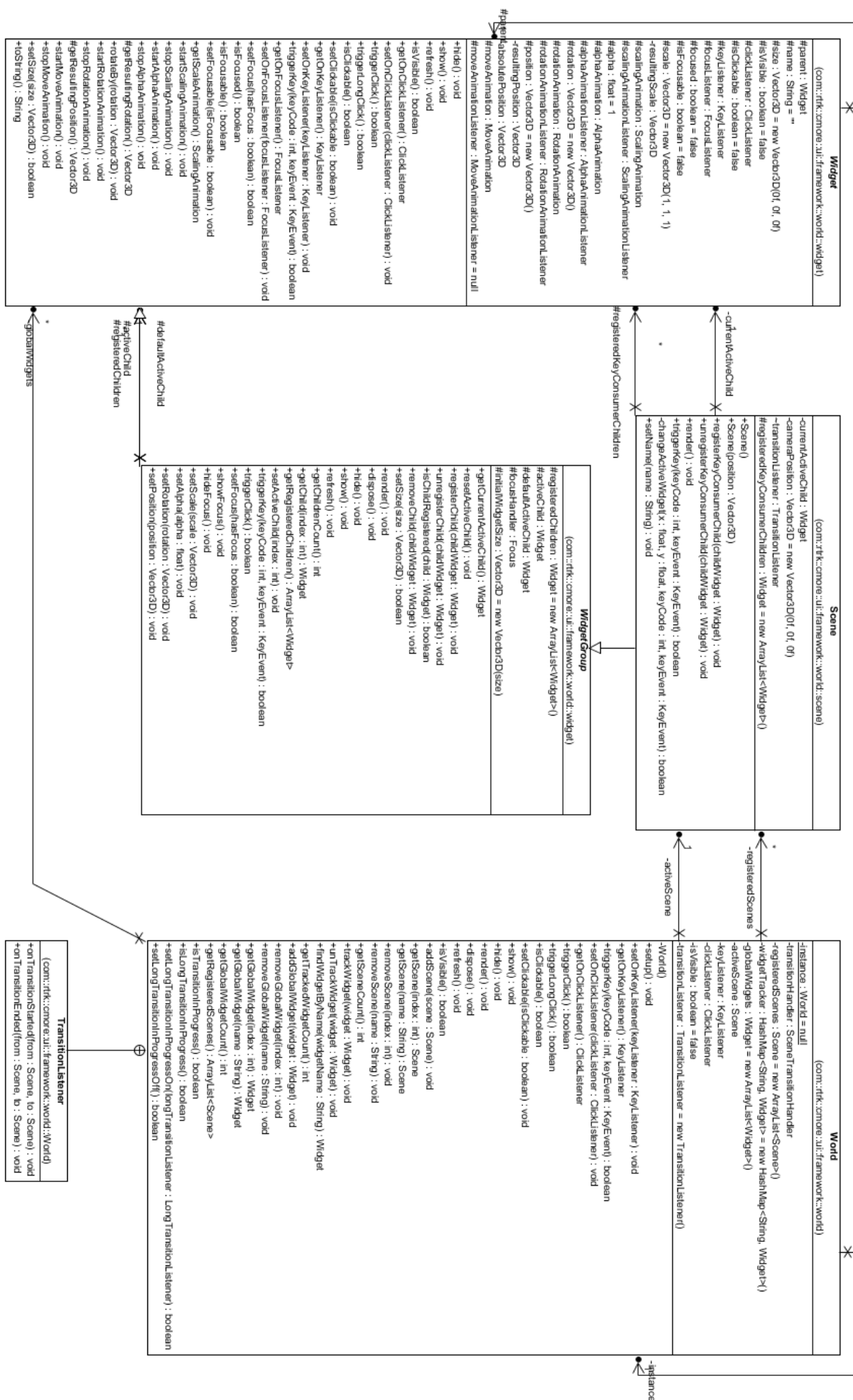
Prilikom implementacije TV aplikacije opisane u ovom radu javlja se slučaj gde se *Vidžet* se kreće istovremeno sa *Kamerom* po *Svetu*. Razvijen je mehanizam gde se kretanje odabranog *Vidžeta* pozivom određene metode vezuje za kretanje *Kamere*, u smislu da pomeraj *Kamere* rezultuje pomeranjem svih *Vidžeta* „vezanih” za nju.

#### 3.2.1.3 Izvedeni elementi grafičkog okruženja

*Vidžet-grupe* sadrže nijedno, jedno ili više potomaka koji su *Vidžeti* ili *Vidžet-grupe*. *Vidžeti* se dodaju u listu potomaka kod pretka (registruju) i time postaju njegovi potomci, a mogu se iz te liste ukloniti. Aktivni potomak predstavlja potomak koje je trenutno u fokusu.

Postoji posebna vrsta dece *Vidžeta*, a to su ona čija *pivot pozicija* ne odgovara apsolutnoj poziciji njihovih roditelja.

Globalni *Vidžeti* predstavljaju *Vidžeti* koji ne moraju da budu registrovani u *Sceni*, već se registruju direktno u *Svet*. *Vidžeti* koji se pojavljuju i interaguju sa korisnikom kroz više scena su obično *globalni Vidžeti*.



Slika 3.2 Dijagram klasa osnovnih elemenata grafičkog okruženja

### 3.2.1.3.1 Liste

Cilj *Liste* je da *Vidžete* potomke grafički prikaže poredane po nekom kriterijumu i da dozvoli kretanje kroz *Listu*.

*Lista* nasleđuje *Vidžet-grupu* i svi njeni potomci koji su vidljivi u datom trenutku na ekranu su registrovani. Kad se desi događaj koji uzrokuje promenu vidljivog dela *Liste* (eng. *scroll*), *Lista* ponovo računa pozicije elemenata i stari skup potomaka zamenjuje novim.

U sebi, *Lista* sadrži instancu klase iz koje dobavlja *Vidžete*, koja je implementirana kao LRU (eng. *least recently used*) keš memorija. Dakle, elementi *Liste* se ponovo koriste (bez novog instanciranja koliko je moguće) i smanjena je količina vremena utrošena na pravljenje novih objekata i količina podataka za uništavanje (podaci koji se više ne koriste).

Postoje tri vrste *Lista*: *regularna lista*, *lista sa fiksiranim fokusom* i *lista sa definisanim pozicijama*.

### 3.2.1.4 Iscrtavanje elemenata pomocu grafičkog okruženja

Svaki *Vidžet* ima vrednost veličine i faktor uvećanja kojim se vrednost veličine množi kada se element iscrtava.

Iscrtavanje elemenata se obavlja niz prethodno pomenutu hijerarhiju, *Svet* poziva metode za iscrtavanje registrovanih *Scena* i *globalnih Vidžeta*. *Scene* pozivaju metode svojih potomaka i tako niz hijerarhiju do krajnjih *Vidžeta* koji ne pozivaju dalje već iscrtaju objekte (Slika 3.3).

#### 3.2.1.4.1 Animacije

Postoji nekoliko tipova animacija.

*Pomeranje* je animiranje grafičkog elementa sa jedne pozicije na drugu postepeno. Ovakva animacija može da menja vrednost pivot ili relativne pozicije.

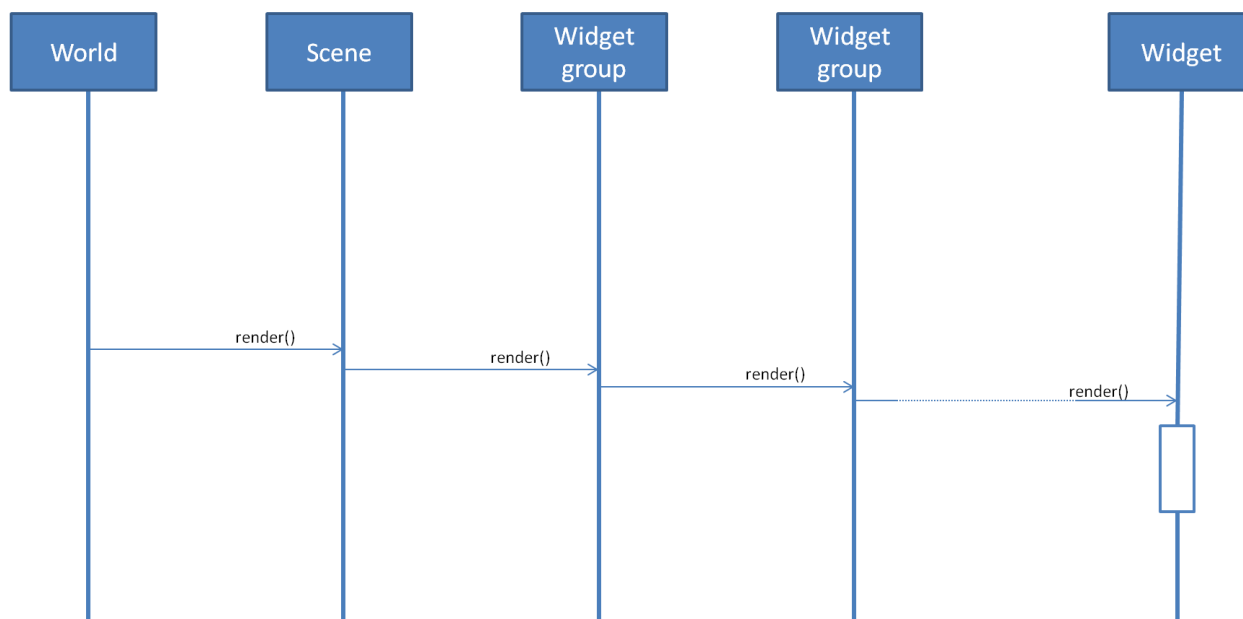
*Animacija transparentije* menja vidljivost *Vidžeta*, od potpuno transparentnog do potpuno vidljivog.

*Rotacija* menja vrednost rotacije elementa u prostoru.

Faktor uvećanja se menja *animacijom skaliranja*, samim tim se menja i rezultujuća veličina elementa.

Moguće je da jedan *Vidžet* u jednom trenutku bude animiran po jednom instancom animacije svake navedene vrste.

*Animacija približavanja i udaljavanja je specifična za Kameru.*



Slika 3.3 Dijagram koji prikazuje hijerarhiju poziva funkcija za iscrtavanje grafičkih elemenata

### 3.2.1.5 Interakcija korisnika sa elementima i međusobna interakcija elemenata

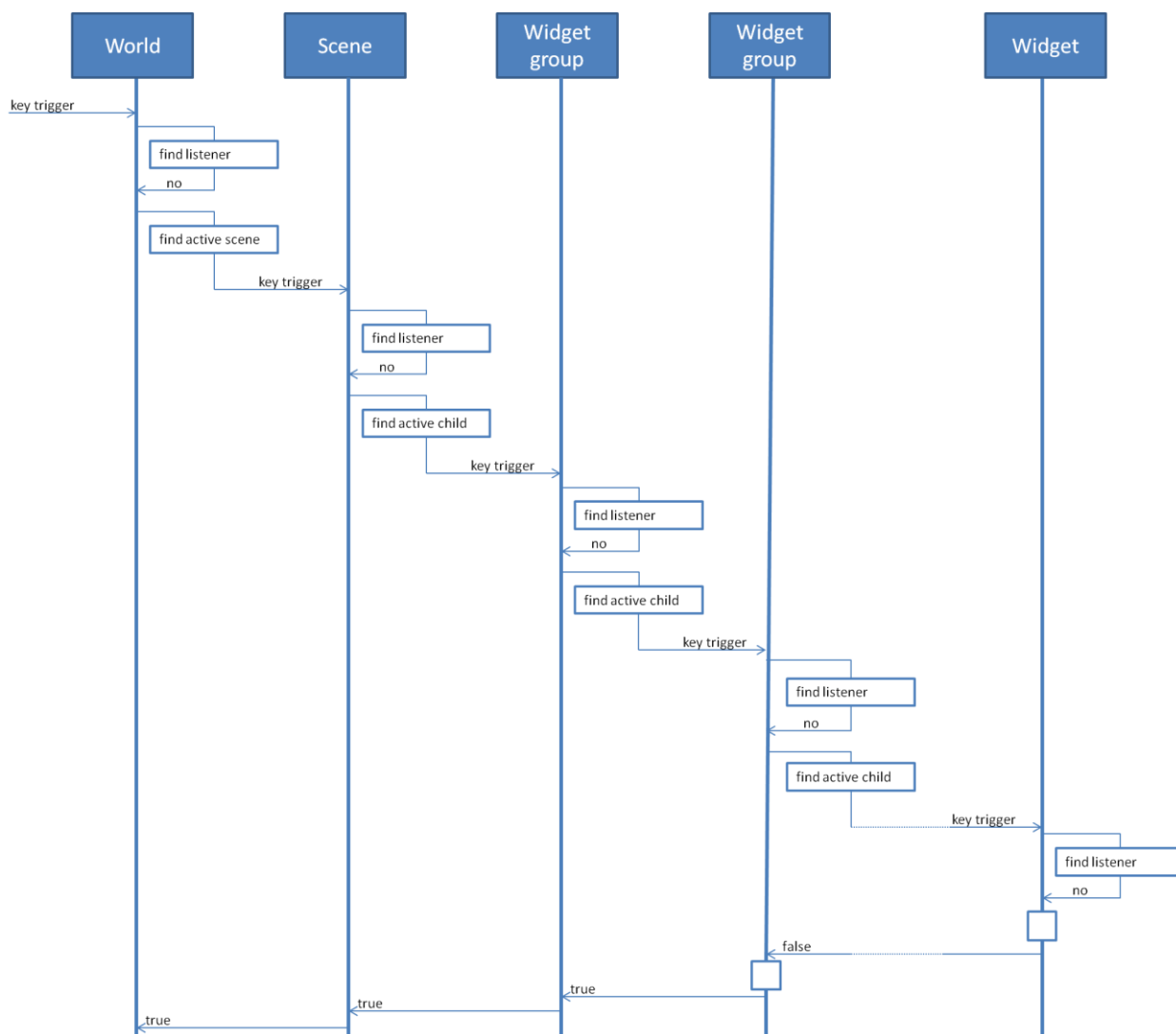
Ulazna akcija kao što je pritisak tastera na daljinskom upravljaču se prosleđuje niz hijerarhiju predak – potomak, što će biti detaljno objašnjeno u daljem tekstu.

Pre nego što se objasni algoritam prosleđivanja akcija, potrebno je naglasiti da svaki grafički element: *Vidžet*, *Vidžet-grupa*, *Scena* i *Svet*, koji može da prihvati korisničku akciju, može da se ima podešen osluškivač na prijem akcija (eng. *listener*) i može proizvoljno da reaguje na ulazne akcije korisnika.

Ukoliko se desi ulazna akcija, pozove se metoda klase *Svet*. Prvo se u toj metodi proveriti da li ima osluškivač postavljen na *Svet* i ukoliko ima, prosledi mu se ulazna akcija. On vraća vrednost *TACNO* ukoliko ne želi da se ova korisnička akcija prosleđuje dalje niz hijerarhiju ili obratno. Ukoliko vrati vrednost *NETAČNO*, trenutno aktivnoj *Sceni* (ukoliko takva postoji) se prosledi akcija. Trenutno aktivna *Scena* analogno ovome, prosleđuje akciju svom osluškivaču ukoliko on postoji, a ako on ne postoji ili vrati vrednost *NETAČNO*, prosleđuje akciju dalje svom aktivnom potomku. Analogno ovome se akcija prosleđuje dalje kroz hijerarhiju dok neko ne vrati vrednost *TACNO*, jer nakon toga se smatra da je na nju reagovano.

Prednost osluškivača akcija i razlog njegovog postojanja je što programski kod koji on izvršava ne mora da bude realizovan u klasi pozivaocu.

Navedena interakcija između elemenata i prosleđivanje ulaznih akcija je prikazana na Slici 3.4.



Slika 3.4 Dijagram sekvence koji opisuje prosleđivanje ulaznih akcija

### 3.2.1.6 Resursi

Svi statički resursi koje koristi aplikacija se čuvaju u memoriji. Ostali resursi se preuzimaju sa Interneta. Kada su preuzeti, stavljaju se u keš memoriju (eng. *cache*) i čuvaju se određeno vreme.

Kada aplikacija zatraži resurs, prvo se proverava da li se taj resurs nalazi među statičkim resursima. Ukoliko resurs nije pronađen, proverava u keš memoriji koja se nalazi u fajl sistemu. Ako resurs nije ranije preuzet sa Interneta, onda se preuzima i stavlja u keš memoriju, a referenca se šalje aplikaciji.

Keš memorija se prazni jedino kada nema više mesta u njoj i to po principu LRU.

### 3.2.2 TV aplikacija

Aplikacija koja je realizovana u objašnjenom grafičkom okruženju sadrži 12 *Scena*: *Video scena*, *Lista kanala*, *Vremenska linija*, *Glavni meni*, *Vodič kroz program*, *Video na zahtev*, *Scena sa detaljima*, *Dodatni sadržaj*, *Media scena*, *Dodatne aplikacije*, *Pretraživanje* i *Podešavanja*.

Sve *Scene* se nalaze unutar statičkog trodimenzionalnog modela koji ima zidove i popularno je nazvan „soba”. *Soba* predstavlja granice 3D prostora (Slika 3.5) u TV aplikaciji i izvan nje ne postoji ništa.

*Kamera* se pomera unutar *sobe*, prelazeći sa *Scene* na *Scenu*.



Slika 3.5 Prikaz 3D Sveta, gde se vidi soba, par *Scena* i *Video Vidžet*

*Video Vidžet* predstavlja pravougaoni deo grafičkog sloja koji je transparentan i kroz koji se vidi reprodukovani video. Njegova pozicija je fiksirana u prostoru i on je ortogonalan u odnosu na *Kameru*.

*Scene* su različito udaljene od *Video Vidžeta*, a za korisnika ovo rezultuje različitom veličinom reprodukovanog video toka na ekranu u različitim *Scenama*. Udaljenost videa od *Scene* zavisi od količine povezanosti televizijskog toka i konkretne *Scene*. Tako na primer, u *Video na zahtev sceni* je *Video Vidžet* manji nego u *Programskom vodiču*.

*Video scena* je *Scena* u kojoj se *Video Vidžet* preko celog ekrana. Iz nje se može otvoriti Dijalog sa opcijama za audio trake, prevod i teletext ili Info dijalog sa informacijama o trenutnom kanalu. Kroz Info dijalog se može preći u *Scenu sa vremenskom linijom* ili u *Listu kanala*. *Vremenska linija* prikazuje emisije na odabranom kanalu u narednom periodu i odabirom određene emisije se može preći u *Scenu sa detaljima*.

---

U *Glavni meni* se ulazi iz *Video scene*. Odavde je omogućena opcija prelaska u *Programski vodič*, *Video na zahtev*, *Media scenu*, *Dodatne aplikacije*, *Pretraživanje* i *Podešavanja*.

*Programski vodič* nudi informacije o TV emisijama u narednom periodu, raspoređene po kategorijama. Odabirom neke emisije se otvara *Scena sa detaljima*.

Iz *Video na zahtev scene* se takođe prelazi u *Scenu sa detaljima*.

*Scena sa detaljima* nudi korisniku prelazak u *Scenu sa dodatnim informacijama*, puštanje odgovarajućeg video sadržaja i slično.

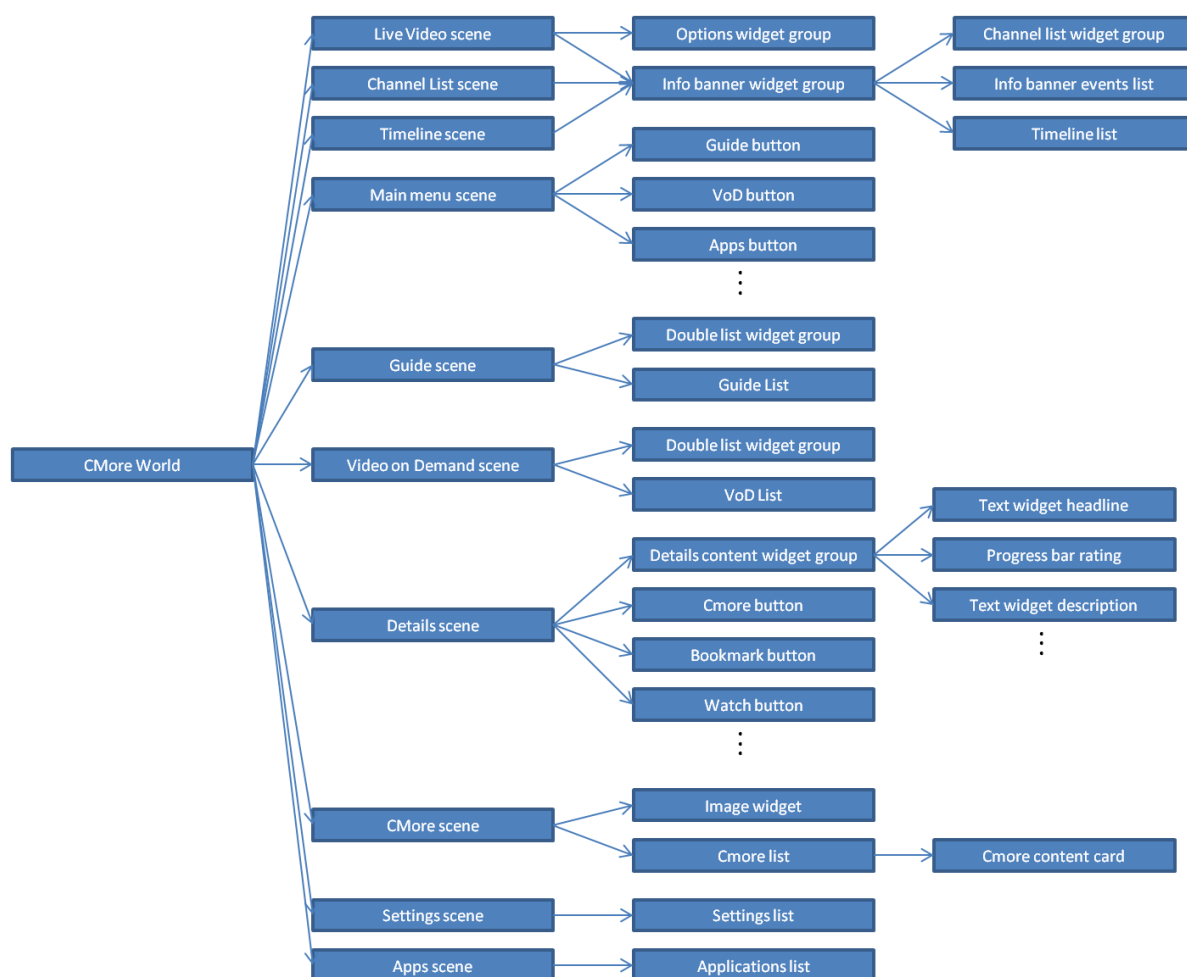
Po prelasku u *Scenu sa aplikacijama*, korisniku se prikazuju Android aplikacije koje može da koristi.

*Podešavanja* se odnose na skeniranje kanala i postavke.

## 4. Implementacija

### 4.1 TV aplikacija

U poglavlju III dat je grub opis TV aplikacije, njenih *Scena* i tranzicija. Struktura TV aplikacije se vidi na Slici 4.1, gde je dato je samo nekoliko primera *Vidžet-grupa*.

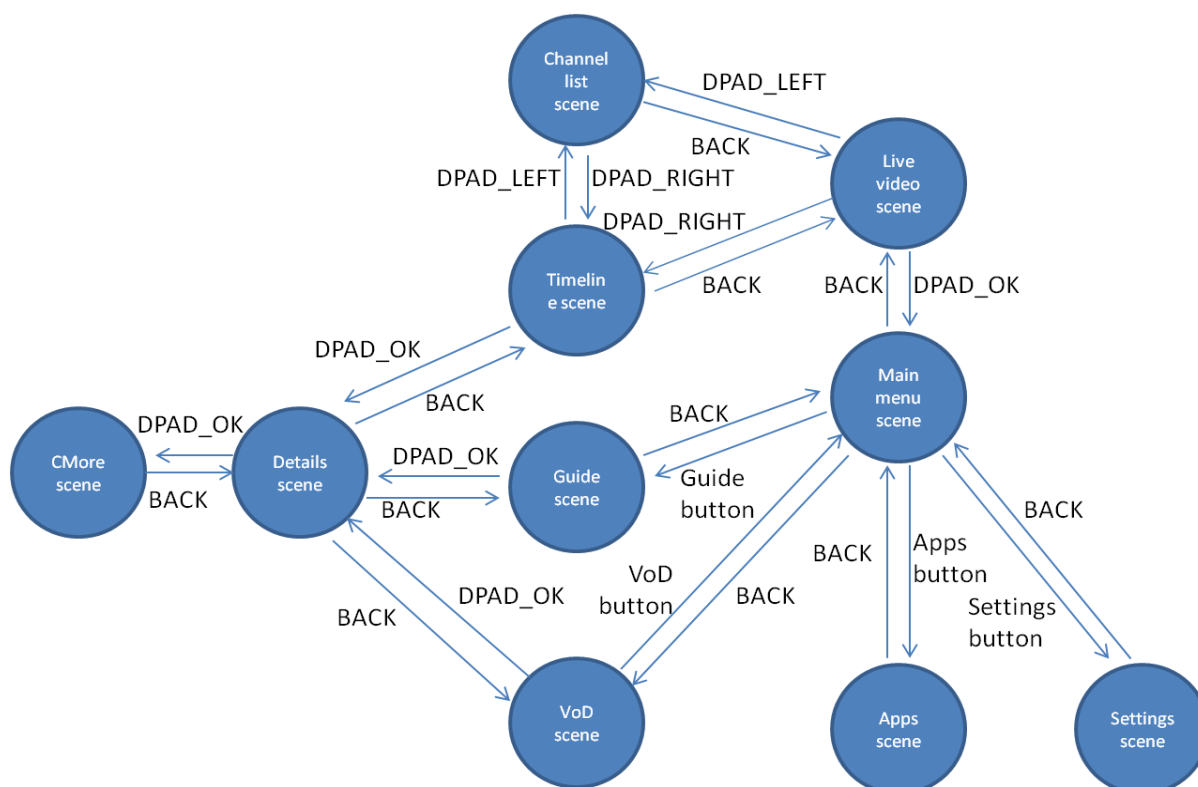


Slika 4.1 Struktura TV aplikacije opisana kroz *Scene* i *Vidžet-grupe* koje sadrži

*Scene* su nalaze na različitim pozicijama u 3D prostoru. U zavisnosti od njihove povezanosti sa televizijskim tokom, nalaze se na različitim udaljenostima od *Video Vidžeta*. Tako je u *Video sceni* *Video Vidžet* raširen preko celog ekrana. U *Scenama Liste kanala* i *Vremenske linije* *Video Vidžet* zauzima veći deo ekrana. *Programski vodič* se nalazi na većoj Z-udaljenosti od *Video Vidžeta* u odnosu na navedene *Scene*, a na manjoj u odnosu na *Video na zahtev scenu*. U *Sceni sa detaljima* i *Sceni sa dodatnim sadržajem*, *Kamera* je rotirana, a *Video Vidžet* nije vidljiv korisniku jer ga zaklanja *Vidžet za prikaz slike*. *Scena sa aplikacijama* sadrži samo jednu listu, pa ima dovoljno prostora da deo ekrana pokriva video.

Dijagram stanja TV aplikacije koji prikazuje tranzicije između *Scena* je prikazan na Slici 4.2.

Uzimajući u obzir da je aplikacija obimna, implementacija će biti objašnjena na primeru *Info dijaloga*, *Liste kanala* i *Vremenske linije*, koji su potpuno implementirani u okviru ovog master rada.



Slika 4.2 Dijagram stanja TV aplikacije koji prikazuje tranzicije između *Scena*, predstavljen kao automat sa konačnim brojem stanja, gde su okidači tranzicija kodovi tastera na daljinskom upravljaču

### 4.1.1 *Info dijalog*

*Info dijalog* najčešće predstavlja informacije o trenutno gledanom programu. Pojavljuje se nakon prebacivanja kanala ili na korisničku komandu i posle određenog vremena korisnikove neaktivnosti se zatvara.

U rešenju iznesenom u ovom radu, *Info dijalog* prikazuje informacije o aktivnom kanalu, TV emisiji koja je trenutno u toku, TV emisiji koja je sledeća na programu i o dve emisije koje će kasnije biti prikazivane, a koje su odabrane u skladu sa afinitetima korisnika. Naziv ovih TV emisija je *info dijalog* lista i predstavlja listu u kojoj su sva četiri elementa uvek vidljiva.

Emisije u *Info dijalogu* su prikazane u vidu informacija o naslovu, vremenu početka i završetka trajanja i slike emisije.

U levom uglu ekrana nalazi se lista kanala koja se sastoji od tri elementa, kroz koju korisnik može da se kreće. U zavisnosti od toga koji je kanal u fokusu u listi, opisane TV emisije u *Info dijalogu* se menjaju.

Kako su u *Info dijalogu* prikazana samo tri kanala, prelaskom u *Scenu liste kanala* korisniku je vidljiva kompletna lista.

Iz *Scene liste kanala* kao i iz *Video scene*, ukoliko korisnika interesuje šta ima u toku 24 časa na odabranom kanalu, on može preći u *Scenu vremenske linije*. Ova *Scena* sadrži listu TV emisija u toku jednog dana na odabranom kanalu.

Odabirom emisije na vremenskoj liniji, korisniku se otvara *Scena sa dodatnim informacijama* o izabranoj emisiji.

#### 4.1.1.1 *Info dijalog Vidžet-grupa*

*Vidžet-grupa Info dijalog* ima tri glavna potomka *Vidžeta*, a to su: lista TV emisija, lista kanala i lista vremenske linije.

To je *globalni Vidžet* koji se kreće sa *Kamerom*. On pripada trima *Scenama* – *Video sceni* u kojoj se inicijalno pojavljuje, *Sceni vremenske linije* gde su u njemu prikazan TV program u naredna 24 časa i *Sceni liste kanala* gde je prikazana proširena lista.

Kako je neophodno da rukuje ulaznim korisničkim akcijama, registrovan je kao potomak u sve tri navedene *Scene*, ali kao prethodno spomenuta posebna vrsta potomka koja od pretka prima samo ulazne akcije.

*Info dijalog* uvek treba da je na istoj poziciji – na donjoj polovini ekrana, a kako *Kamera* prelazi iz *Scene* u *Scenu*, on se po pojavljivanju kači za nju i kreće se sa njom sve do zatvaranja.

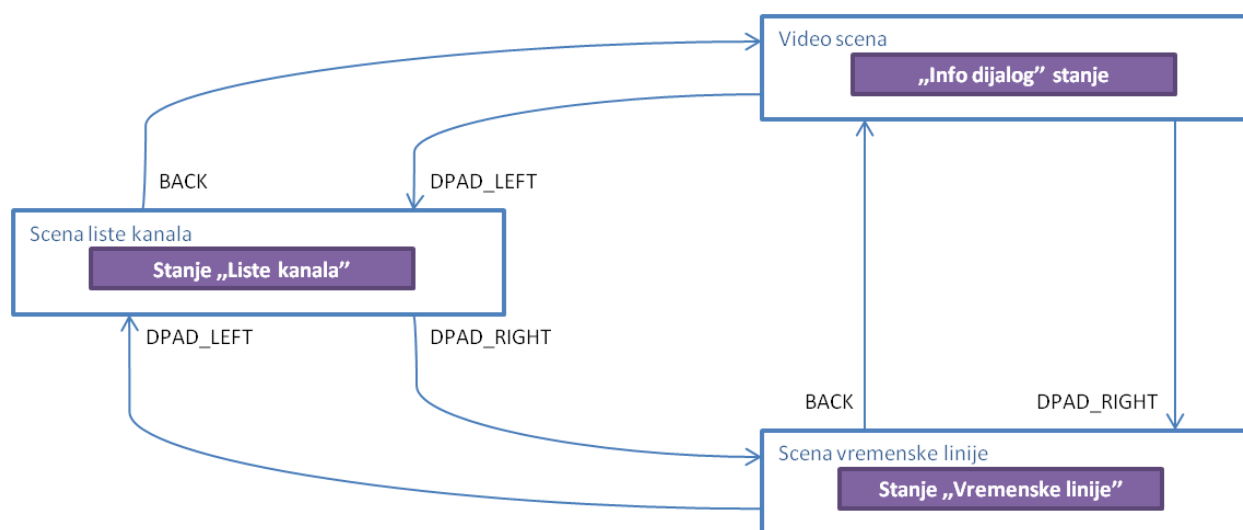
#### 4.1.1.1.1 Animacije u *Info dijalogu*

Animacije koje se koriste u *Info dijalogu* *Vidžet-grupe* su animacija transparentije i animacija pomeranja i to pri otvaranju i zatvaranju *Info dijaloga* i tranzicijama u *Scenu liste kanala* i *Scenu vremenske linije*.

#### 4.1.1.2 Stanja

*Info dijalog* predstavlja automat sa tri stanja (Slika 4.3). Prelasci iz stanja u stanje su određeni korisničkim akcijama i uzrokuju prelazak sa *Scene* na *Scenu*.

U tekstu koji sledi će biti opisano svako stanje ponaosob radi razumevanja implementacije *Info dijaloga*.



Slika 4.3 Dijagram stanja *Info dijaloga*

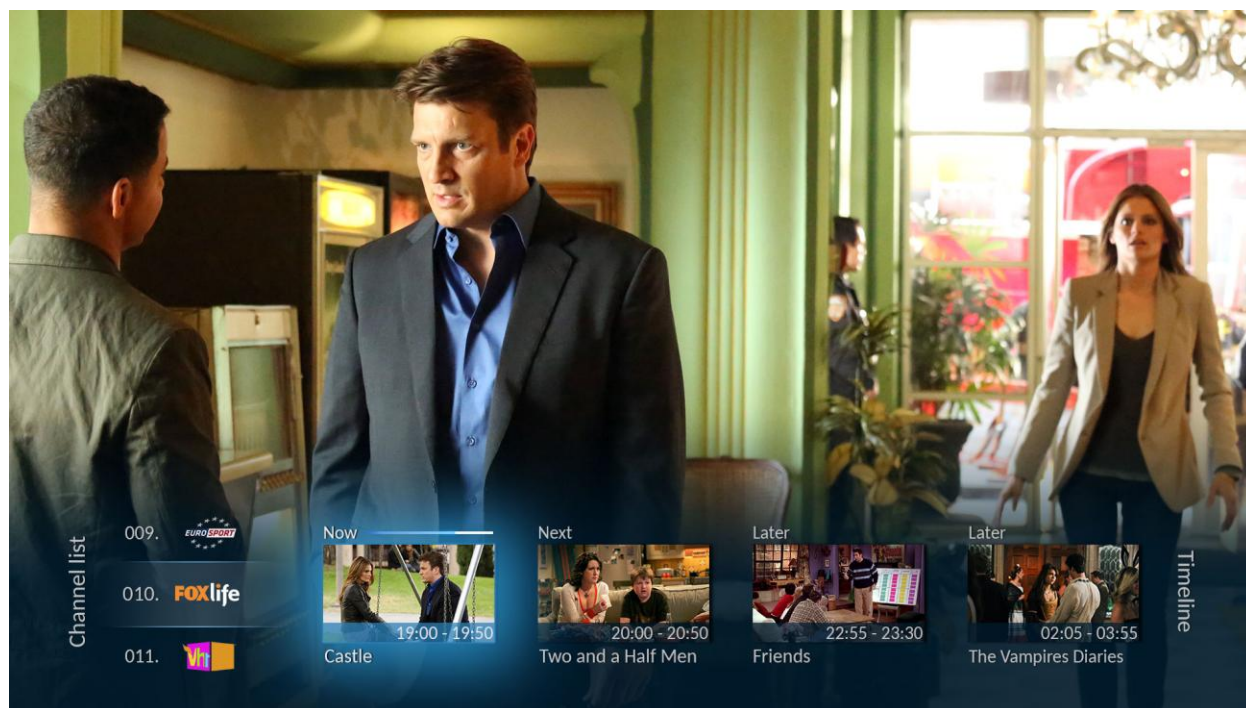
#### 4.1.1.2.1 „Info dijalog“ stanje

„Info dijalog“ stanje je inicijalno stanje (Slika 4.4). Dakle, uvek kada dođe do pojavljivanja ove *Vidžet-grupe*, ona se nalazi u inicijalnom „Info dijalog“ stanju.

Ovo stanje je aktivno u *Video sceni*, gde se *Info dijalog* pojavljuje animacijama transparentije i pomeranja. Animacije izgledaju na sledeći način: *Info dijalog* *Vidžet-grupa* se pojavljuje od ispod vidnog polja kamere dok mu se vidljivost povećava.

U ovom stanju, korisnik je u mogućnosti da obavlja navigaciju kroz četiri prikazana elementa *Info dijaloga* i da prolazi kroz listu u kojoj su vidljiva tri kanala.

Ukoliko je lista kanala u fokusu, pritiskom na taster za navigaciju levo dešava se tranzicija u *Scenu liste kanala* što implicira pomeranje *Kamere*, pri čemu *Info dijalog* prelazi u stanje „Lista kanala“ i uz određene animacije se prikazuje proširena lista kanala.



Slika 4.4 *Info dijalog* u inicijalnom stanju

Drugi slučaj prelaska u drugo stanje i drugu scenu, jeste odabir određenog elementa info dijalog liste pritiskom na taster na daljinskom upravljaču ili pritisak tastera za navigaciju desno kada je krajnji desni element info dijalog liste u fokusu. Ove korisničke akcije vode u Scenu vremenske linije i *Info dijalog* prelazi u istoimeno stanje.

Posle određenog vremena neaktivnosti nad *Info dijalogom*, on se automatski zatvara. Drugi način da se on sakrije je pritiskom na taster za povratak, nakon čega se na njemu primenjuje animacija transparentije nakon čega je *Video Vidžet* ponovo ceo vidljiv.

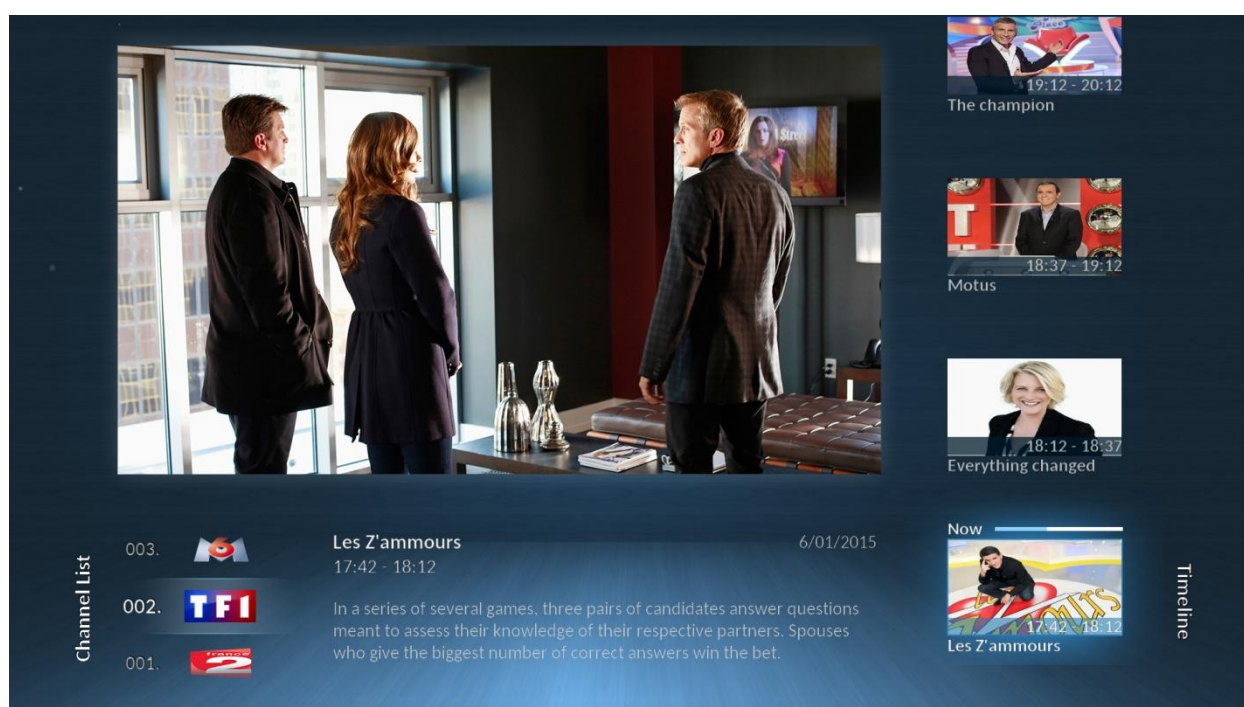
#### 4.1.1.2.2 Stanje „Vremenske linije”

Kada je *Info dijalog* u stanju „Vremenske linije” (Slika 4.5), korisnik je u mogućnosti da se kreće kroz TV emisije na odabranom kanalu i za emisiju u fokusu da pogleda dodatne informacije poput kratkog sadržaja.

Odabirom određene emisije korisniku se prikazuje *Scena sa detaljima*, ali ona nije tema ovog poglavlja.

Prelazak u stanje „Lista kanala” i istoimenu *Scenu* indukuje potpuno ista korisnička akcija koja to radi u „Info dijalog” stanju.

Pritiskom tastera za povratak, korisnik se vraća u *Video scenu* kom je *Info dijalog* u inicijalnom stanju.

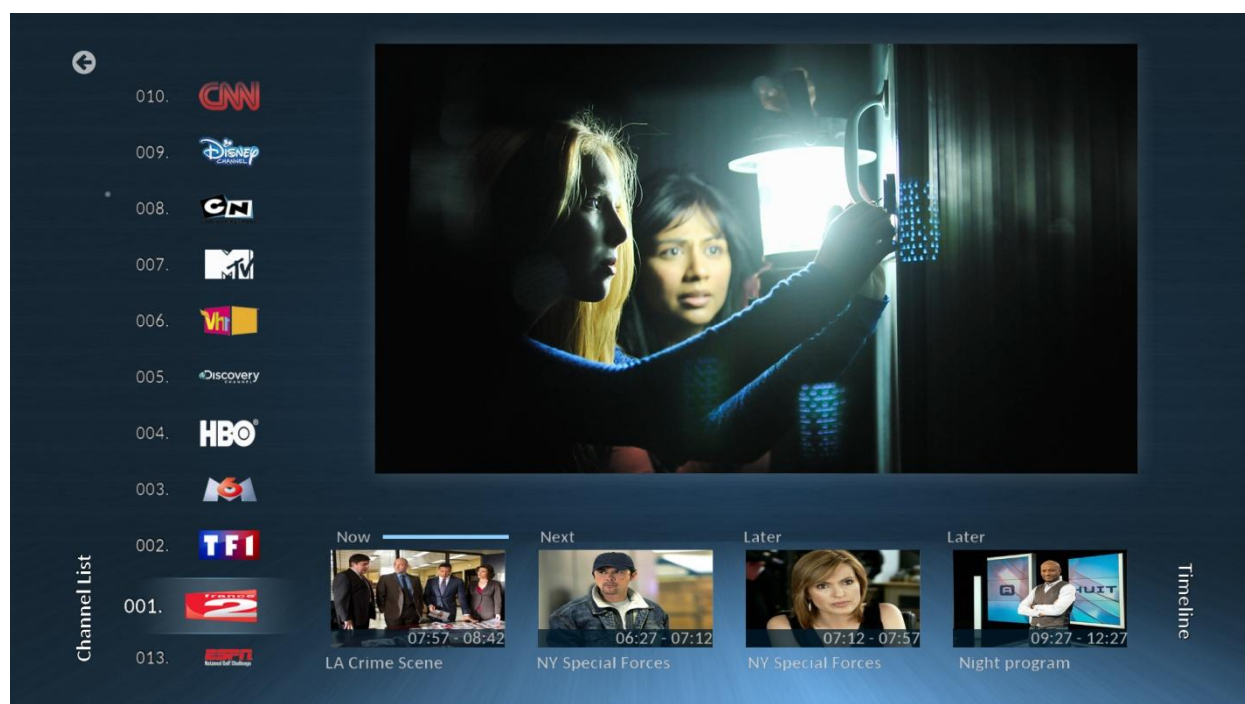


Slika 4.5 Info dijalog u Sceni vremenske linije

#### 4.1.1.2.3 Stanje „Liste kanala”

U stanju „Liste kanala” korisniku je prikazana *Scena liste kanala* (Slika 4.6). Odatle je navigacija do *Scene vremenske linije* potpuno analogna navigaciji iz „Info dijalog” stanja, povratak u inicijalno stanje i *Video scenu* se postiže tasterom za povratak.

Korisniku je prikazano višestruko više kanala jer se lista kanala proširila i zauzima kompletnu levu stranu ekrana.



Slika 4.6 Info dijalog u Sceni liste kanala

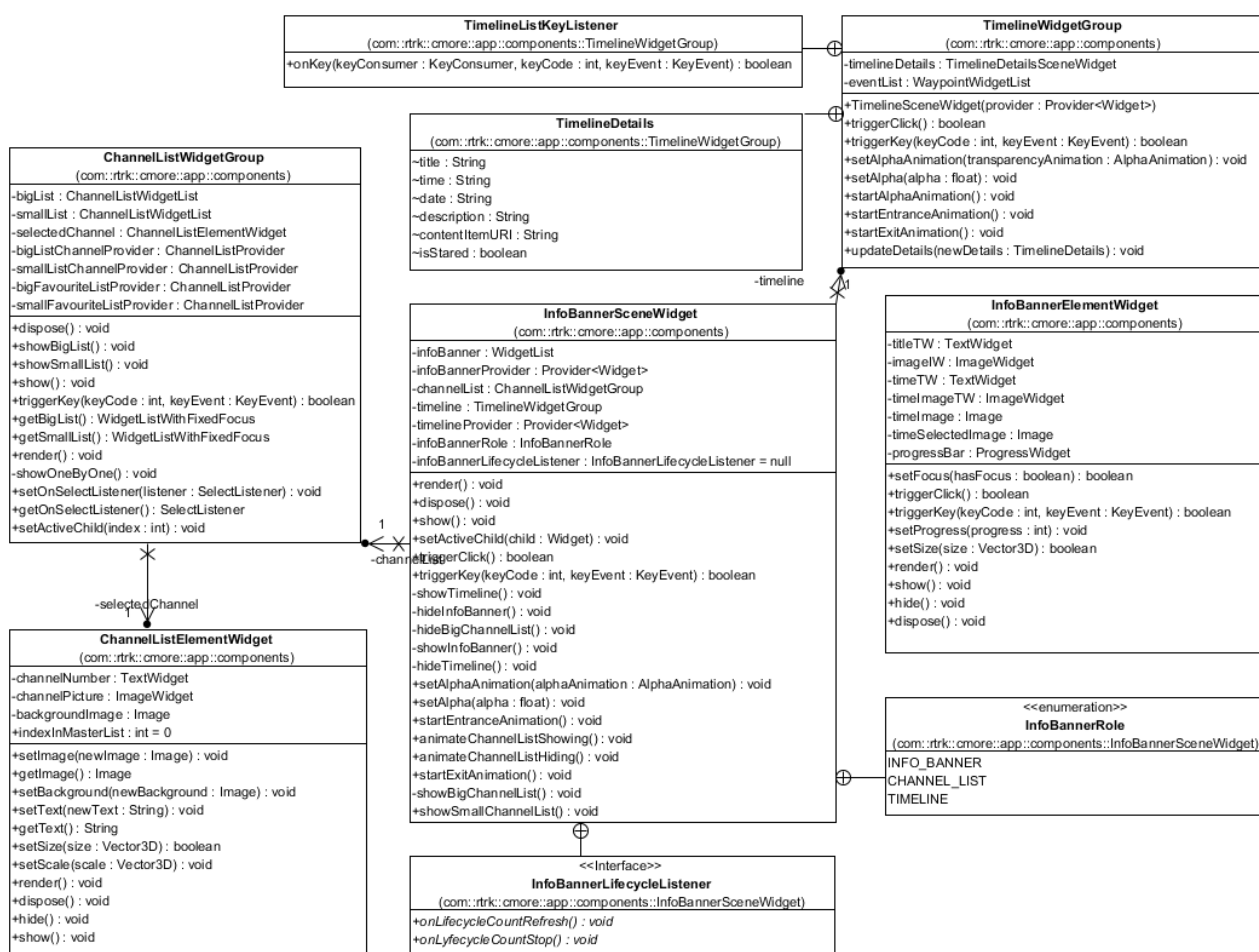
### 4.1.1.3 Implementacija *Info dijaloga* u grafičkom okruženju

*Info dijalog* je implementiran kao klasa *InfoBannerWidgetGroup.java* i predstavlja *Vidžet-grupu* sa tri glavna potomka koji su takođe *Vidžet-grupe*:

1. *WidgetList.java* (info dijalog lista TV emisija),
2. *TimelineWidgetGroup.java* (lista vremenske linije sa opisom, nalazi se u Prilogu rada),
3. *ChannelListWidgetGroup.java* (lista kanala).

Dijagram klasa koje nasleđuju *Vidžet-grupu*, kao i njihova povezanost su prikazani na Slici 4.7.

Klasa koja je navedena prva predstavlja neizmenjenu listu grafičkog okruženja. Ona uvek ima 4 elementa koji su klase *InfoBannerElementWidgetGroup.java* u kojoj se na određenim pozicijama nalaze *Vidžet sa slikom* (*ImageWidget.java*), *Vidžeti sa tekstom*, vremenom početka i završetka trajanja i naslovom kojim je naznačeno da li je to trenutna, sledeća ili neka TV emisija koja će ići kasnije (*TextWidget.java*) i ukoliko je trenutna *Vidžet sa indikatorom napredovanja* (*ProgressBar.java*). Ova klasa je se pojavljuje sa Info dijalogom i vidljiva je u *Video sceni* i *Sceni liste kanala*.



Slika 4.7 Dijagram klasa *Info dijaloga*

Ona nestaje prelaskom u *Scenu vremenske linije*, gde se animacijom pojavljuje *Vidžet-grupa* predstavljena klasom pod brojem 2. *Vidžet-grupa* vremenske linije ima dva glavna potomka: lista vremenske linije i dodatni opis. Lista vremenske linije je realizovana kao neizmenjena lista sa definisanim pozicijama (*WayPointWidgetGroup.java*) čiji su elementi klase *InfoBannerElementWidgetGroup.java*. Lista sa definisanim pozicijama je odabrana zbog toga što su elementi liste na različitim Z-odstojanjima. U polju za dodatni opis koji se sastoji od klasa *TextWidget.java* i *ImageWidget.java* su prikazane informacije vezane za aktivni element liste vremenske linije.

Jedina klasa koja je vidljiva u svim stanjima je klasa *ChannelListWidgetGroup.java*. U *Sceni liste kanala* je prikazana njena proširena verzija, a u ostalim *Scenama* je verzija u kojoj su vidljiva samo tri elementa. Ova *Vidžet-grupa* se sastoji od tri glavna potomka: velika i mala lista koje su implementirane kao liste sa fiksiranim fokusom (*WidgetListWithFixedFocus.java*) i selektovani kanal koji je klase *ChannelListElementWidgetGroup.java* kao i elementi pomenutih lista. Liste su složene jedna preko druge i fokusirani kanal stoji najbliže kameri po Z-osi, tako da korisniku na prvi pogleda deluje kao da je reč o jednoj listi čiji je aktivni element veći od ostalih. Velika lista sa više od tri elementa je prikazana samo u *Sceni liste kanala*. Ona se pojavljuje i nestaje animacijama transparentije, element po element. Manja lista je prikazana u svim *Scenama*, kao i selektovani element. Selektovani element je izdvojen kao poseban *Vidžet* u odnosu na liste zbog animacija kretanja liste gde neki element ukoliko treba da bude aktivan nestane i pojavi se na mestu aktivnog elementa. On je klase *ChannelListElementWidgetGroup.java*, koja sadrži dve slike – pozadinu kompletnog elementa i sliku kanala (*ImageWidget.java*) i broj kanala (*TextWidget.java*).

U toku objašnjavanja implementacije je izostavljeno nekoliko irelevantnih *Vidžeta* koji su takođe registrovani kao potomci, ali imaju samo dekorativnu ulogu.

| Metoda                      | Opis metode                                                                     | Mera kontrole  |
|-----------------------------|---------------------------------------------------------------------------------|----------------|
| <i>void showBigList()</i>   | Poziva metode svoje dece tako da veća lista uz animiranje bude vidljiva.        | <i>public</i>  |
| <i>void showSmallList()</i> | Poziva metode svoje dece za izlaznu animaciju kod zatvaranja info dijaloga.     | <i>public</i>  |
| <i>void showOneByOne()</i>  | Metoda koja animacijom prikazuje jedan po jedan element proširene liste kanala. | <i>private</i> |

Tabela 4.1 Prikaz metoda klase *ChannelListWidgetGroup.java* koje se pozivaju prilikom tranzicije iz *Scene* u *Scenu*

U Tabelama 4.1, 4.2 i 4.3 su prikazane funkcije navedene tri *Vidžet-grupe*, koje se pozivaju prilikom tranzicija iz *Scene* u *Scenu*.

Kod realizacije programskog interfejsa navedenih klasa, metode su osmišljene tako da se odnose na funkcionalnosti koje obavljaju, a enkapsuliraju animacije i efekte koje uključuju.

Prilikom tranzicije između *Scene liste kanala*, *Scene vremenske linije* i *Video scene*, *Info dijalog Vidžet-grupa* u potpunosti rukuje promenom stanja info dijaloga. Stanje je interno polje u ovoj klasi i zaštićeno je tako što pravo pristupa ovom polju ima samo klasa u kojoj se nalazi.

Klasa *InfoBannerWidgetGroup.java* poziva svoje metode i metode svojih potomaka i na taj način se vizuelno menja (prikazan je *Info dijalog* u inicijalnom stanju, sa proširenom listom kanala ili sa listom vremenske linije). Zbog toga su mere kontrole svih funkcija ove klase takve da se samo iz unutrašnjosti klase mogu pozivati, osim funkcija za prikazivanje i zatvaranje *Info dijaloga*. Pojavljivanjem i nestajanjem *Info dijaloga* se rukuje van klase *InfoBannerWidgetGroup.java*.

Kao i promenama stanja, *Info dijalog Vidžet-grupa* interno rukuje i prelascima između *Scena* u kojima je vidljiva.

#### 4.1.1.3.1 Oslušivači *Info dijalog Vidžet-grupe*

Da bi funkcionalnost *Info dijaloga* bila ostvarena, korišćeni su oslušivači.

Na listu kanala podešen je oslušivač u kom je realizovano zapovanje (prebacivanje) na odabrani kanal u listi. Time je postignuto da kada je lista kanala u fokusu i korisnik pritisne taster za odabir, poziva se oslušivač koji rukuje ovom korisničkom akcijom i prebacuje kanal. Oslušivač u ovom slučaju vraća povratnu vrednost *TAČNO*, jer korisnička akcija ne treba da se prosleđuje dalje kroz hijerarhiju.

| Metoda                               | Opis metode                                                               | Mera kontrole |
|--------------------------------------|---------------------------------------------------------------------------|---------------|
| <i>void startEntranceAnimation()</i> | Poziva metode svoje dece tako da se pojavi Vidžet-grupa vremenske linije. | <i>public</i> |
| <i>void startExitAnimation()</i>     | Poziva metode svoje dece tako da nestane Vidžet-grupa vremenske linije.   | <i>public</i> |

Tabela 4.2 Prikaz metoda klase *TimelineWidgetGroup.java* koje se pozivaju prilikom tranzicije iz *Scene* u *Scenu*

Na elemente liste info dijaloga podešen je oslušivač tako da kada se objavi događaj pritiska na taster za odabir, *Info dijalog* prelazi u stanje „Vremenska linija” gde je fokusirani element liste vremenske linije baš taj nad kojim se desila korisnička akcija u info dijalog listi.

Kako ova akcija zahteva i pomeranje *Kamere* tj. prelazak u drugu *Scenu*, osluškivač vraća povratnu vrednost *NETAČNO*, gde se dalje u hijerarhiji rukuje prelaskom sa *Scene* na *Scenu*.

Osluškivač koji je podešen na listu vremenske linije obezbeđuje da kada korisnik pritisne taster za odabir počinje tranzicija na *Scenu sa detaljima*.

Tranzicije između *Scena* koje nisu izazvane pritiskom tastera za odabir, su direktno realizovane unutar klase *Info dijalog Vidžet-grupe*, u metodi koja se poziva za rukovanje korisničkim akcijama (ne preko osluškivača).

| Metoda                                  | Opis metode                                                                               | Mera kontrole  |
|-----------------------------------------|-------------------------------------------------------------------------------------------|----------------|
| <i>void startEntranceAnimation()</i>    | Poziva metode svoje dece za ulaznu animaciju kod pojavljivanja info dijalog Vidžet-grupe. | <i>public</i>  |
| <i>void startExitAnimation()</i>        | Poziva metode svoje dece za izlaznu animaciju kod zatvaranja info dijaloga.               | <i>public</i>  |
| <i>void showInfoBanner()</i>            | Metoda koja animacijom prikazuje info dijalog listu elemenata.                            | <i>private</i> |
| <i>void hideInfoBanner()</i>            | Metoda koja animacijom sakriva info dijalog listu elemenata.                              | <i>private</i> |
| <i>void showTimeline()</i>              | Metoda koja animacijom prikazuje listu vremenske linije.                                  | <i>private</i> |
| <i>void hideTimeline()</i>              | Metoda koja animacijom sakriva listu vremenske linije.                                    | <i>private</i> |
| <i>void animateChannelListShowing()</i> | Metoda koja animacijom pomera elemente info dijalog liste po Z-osi.                       | <i>private</i> |
| <i>void animateChannelListHidding()</i> | Metoda koja animacijom vraća elemente info dijalog liste na istu Z-udaljenost.            | <i>private</i> |
| <i>void showBigChannelList()</i>        | Metoda koja animira i prikazuje proširenu listu kanala.                                   | <i>private</i> |
| <i>void hideBigChannelList()</i>        | Metoda koja animira i sakriva proširenu listu kanala.                                     | <i>private</i> |

Tabela 4.3 Metode klase *InfoBannerWidgetGroup.java* korišćene prilikom prelaska iz *Scene* u *Scenu*, kao i za pojavljivanje i nestajanje *Info dijaloga*

#### 4.1.1.4 Upravljanje resursima

Radi uštede memorije, pri implementacijama lista je već u grafičkom okruženju uvedeno ponovo korišćenje istih instanca *Vidžeta* elemenata liste.

Ono što je specifično za *info dijalog Vidžet-grupu*, jeste što se po promeni selektovanog kanala u listi kanala sa Interneta preuzimaju informacije o novim TV emisijama. Ovo se dešava

onoliko često koliko korisnik to koristi i može biti dovoljno često korišćeno da iako se memorija puni novim *Vidžetima* i blagovremeno prazni može doći do zagušenja ili usporenja grafičke sprege, što narušava brzinu odziva.

Da bismo izbegli navedene probleme, svi elementi se ponovo koriste. Kada se preuzmu sa Interneta, novi podaci se skladište u memoriju, a već postojeći *Vidžeti* se osvežavaju novim informacijama. To su četiri elementa info dijaloga liste i onoliko elemenata koliko vidljivo na ekranu liste vremenske linije.

#### **4.1.1.5 Optimizacije**

Prilikom prolaska kroz listu kanala, dobavljaju se novi sadržaji koji će biti prikazani u info dijalogu liste TV emisija. Svaki put kada je novi kanal u fokusu pokrene se vremenska kontrola i nakon 350 ms se pokreće osvežavanje sadržaja. Ukoliko se za manje od 350 ms fokusirani kanal promeni, onda se odbrojava novih 350 ms. Optimizacija je uvedena u smislu da je smanjen broj zahteva koji se šalju i samim tim izbegnuto je pristizanje irelevantnih podataka.

Nakon zadržavanja na određenom kanalu više od navedenog perioda, dobavljaju se informacije o trenutnom kanalu sa Interneta. Kada stignu, informacije se čuvaju u neizmenjenom obliku i odatle se osvežava info dijalog lista emisija i lista vremenske linije. Ovo je optimizacija u smislu korisničkog iskustva, jer ukoliko korisnik otvori scenu vremenske linije, informacije već postoje u memoriji.

Da bi što manje objekata bilo čuvano u memoriji i manje vremena utrošeno na pravljenje novih instanci, liste ponovo koriste već postojeće elemente. Tako lista vremenske linije, kao i liste kanala, u memoriji drže onoliko elemenata koliko je prikazano na ekranu.

## 5. Testiranje i verifikacija

U ispitnom okruženju izmerene su performanse sistema. Mereno je zauzeće radne i grafičke memorije, iskorišćenost centralnog procesora i broj slika u sekundi.

Vrednosti su merene svake sekunde u ukupnom trajanju od jednog sata. TV aplikacija je korišćena regularno u toku ovih sat vremena, simulacijom intenzivnog korišćenja aplikacije.

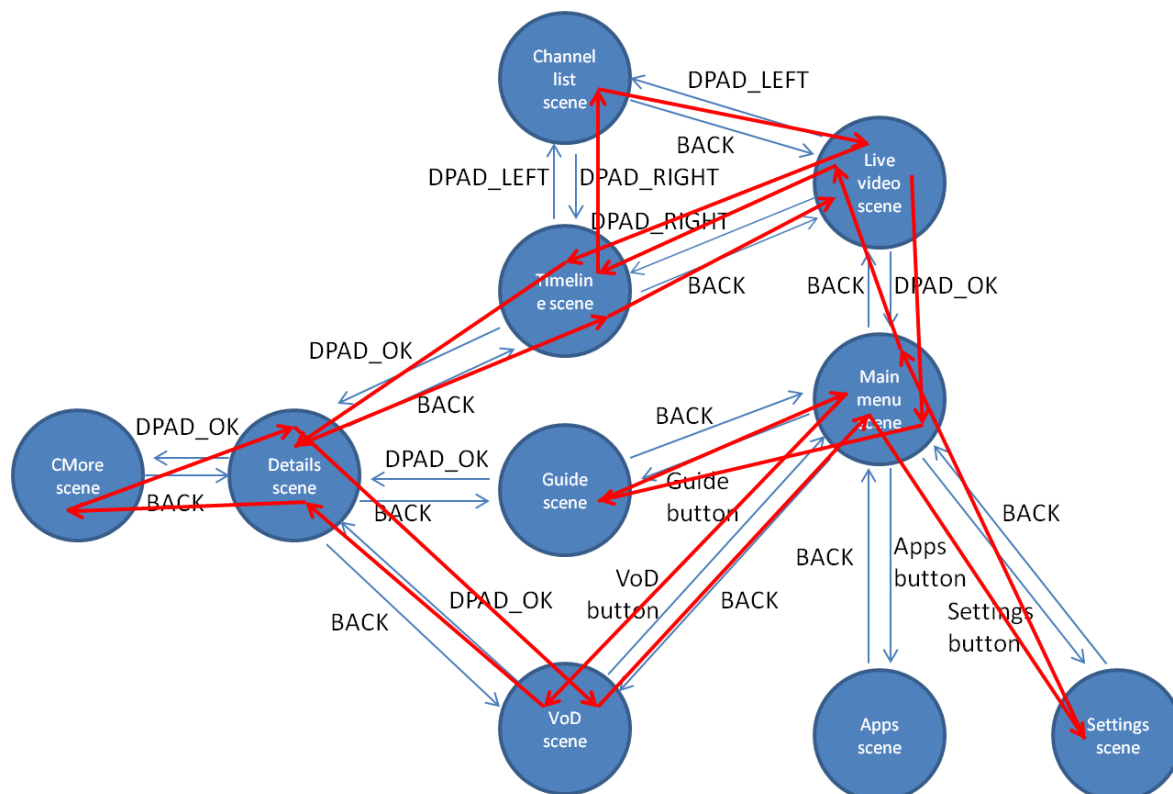
Ispitivane vrednosti su analizirane u tekstu koji sledi.

### 5.1 Simulacija intenzivnog korišćenja aplikacije

Simulacija korišćenja aplikacije bazirana je na stanju aplikacije, kada se korisničke akcije automatski šalju aplikaciji i pri tom se prolazi kroz sve *Scene* i testiraju se funkcionalnosti.

Dijagram prolazaka kroz *Scene*, objašnjen u daljem tekstu, se vidi na Slici 5.1.

Prvo se ulazi u *Glavni meni scenu* (Slika 5.2), odakle se prelazi u *Scenu programskog vodiča* (Slika 5.3). Zatim se vraća u *Glavni meni*, odakle se prelazi u *Video na zahtev scenu* (Slika 5.4). Iz *Video na zahtev scene* se prelazi u *Scenu sa detaljima* (Slika 5.5) i u *Scenu sa dodatnim sadržajem* (Slika 5.6). Nakon povratka u *Glavni meni*, prelazi se u *Scenu sa podešavanjima* (Slika 5.7) i *Scenu sa dodatnim aplikacijama* (Slika 5.8). Odatle se prelazi nazad u *Video scenu*, odakle se otvara *Info dijalog*, kroz koji se navigira do *Scene vremenske linije* i *Scene liste kanala*. Iz *Scene vremenske linije* se prelazi u *Scenu sa detaljima*, i vraća se nazad u *Video scenu*.

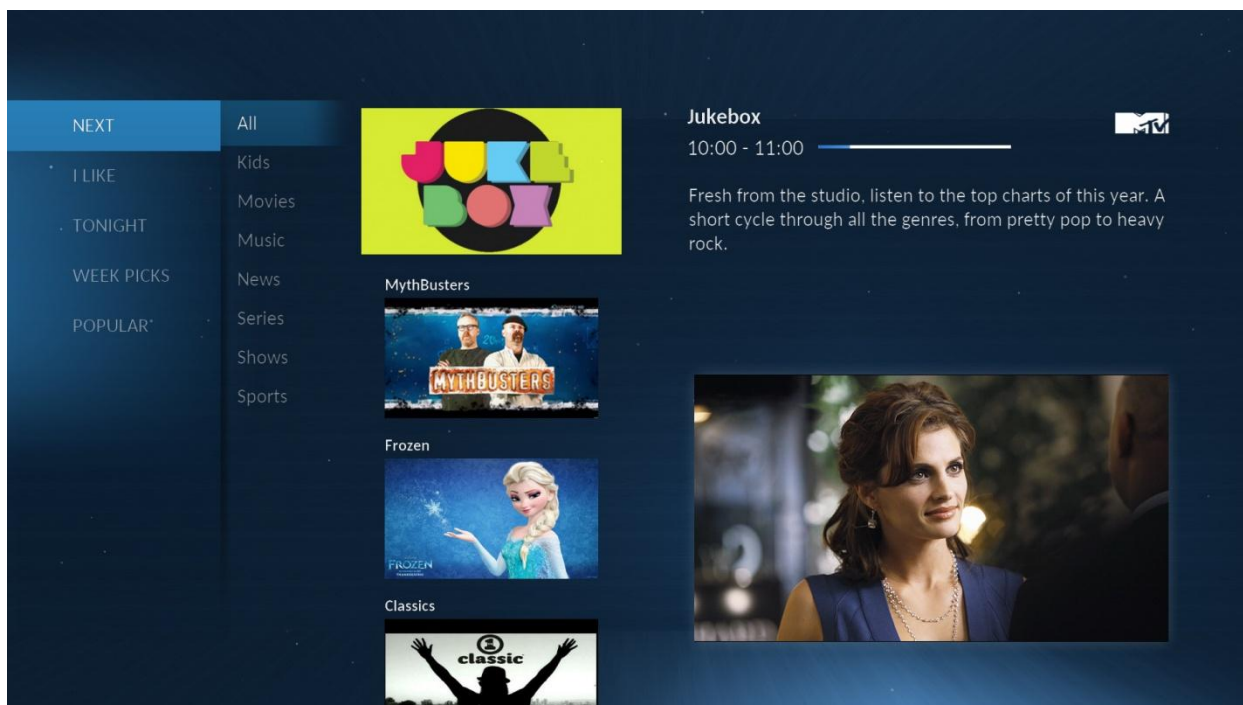


Slika 5.1 Dijagram prolazaka kroz *Scene* prilikom simulacije korišćenja aplikacije

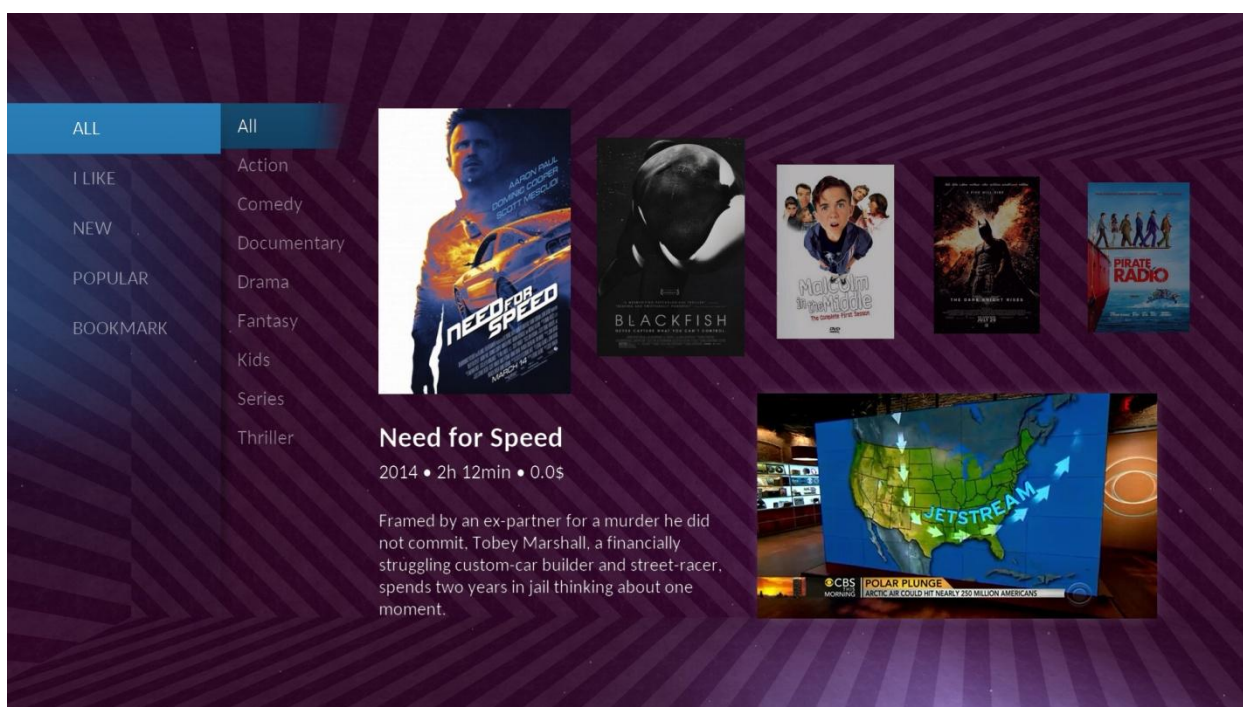


Slika 5.2 Glavni meni scena

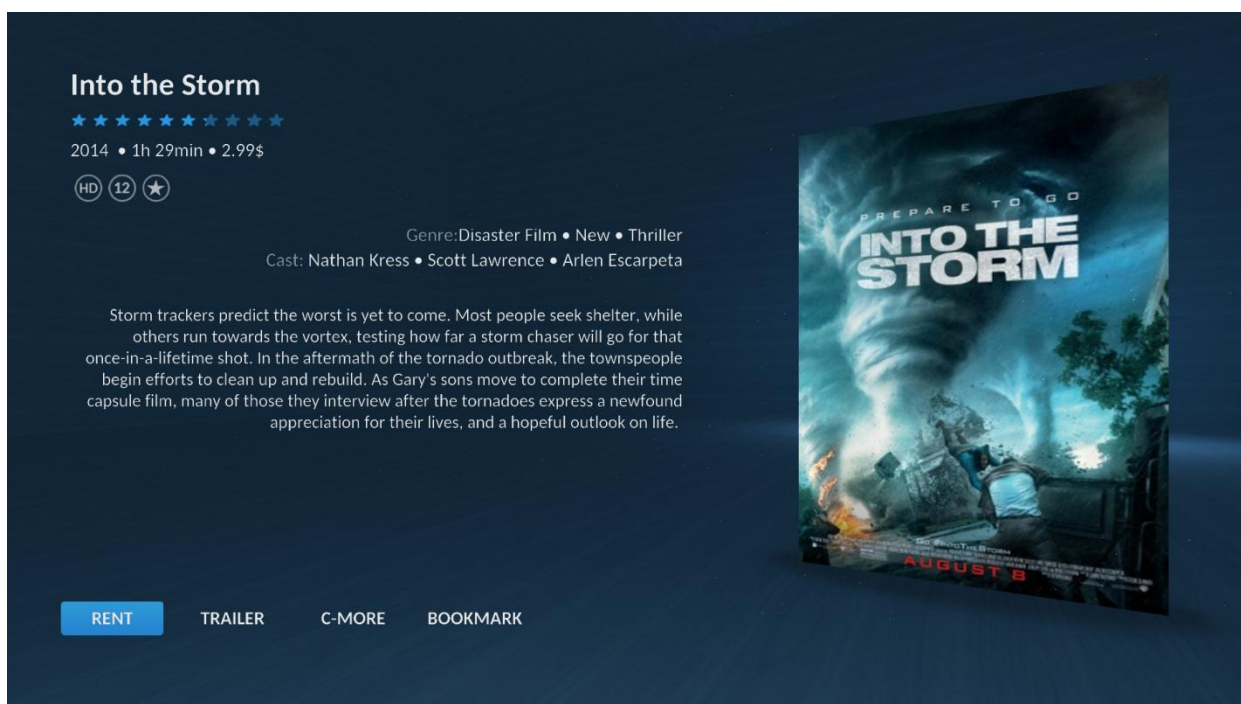
U svakoj *Sceni* se provede određena količina vremena, u toku koje se testiraju funkcionalnosti koje ona nudi.



Slika 5.3 Scena programskog vodiča



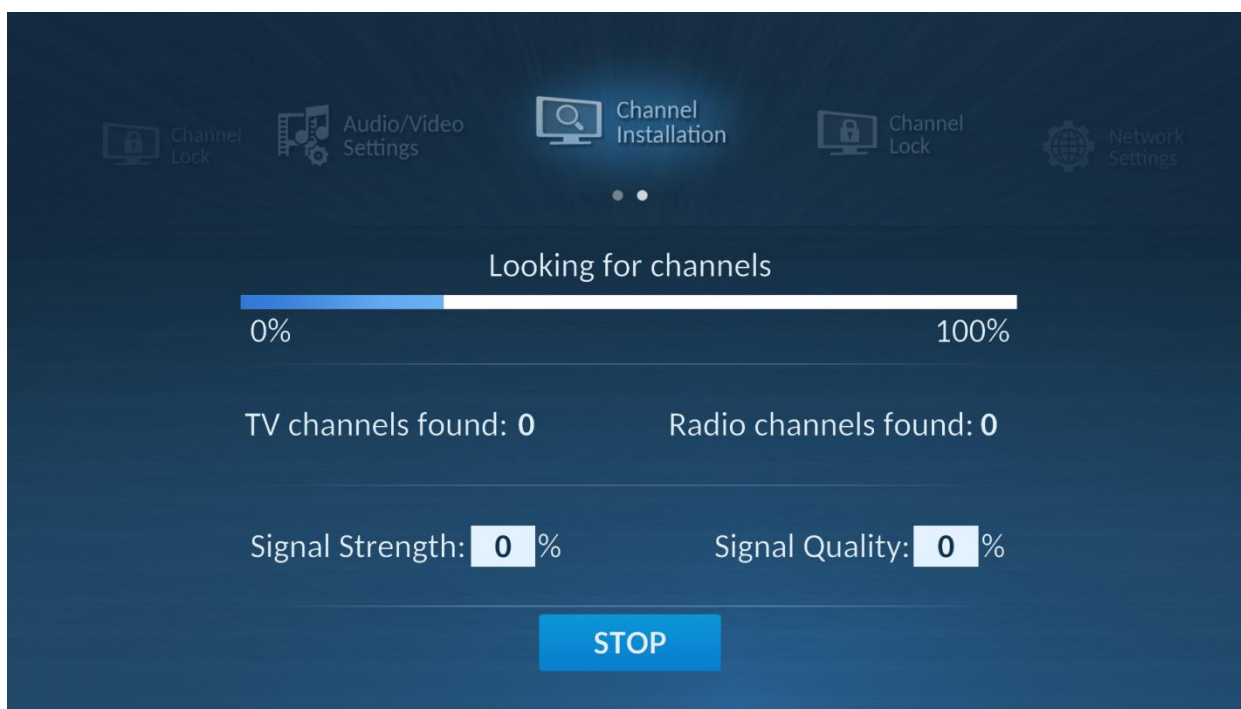
Slika 5.4 Video na zahtev scena



Slika 5.5 Scena sa detaljima



Slika 5.6 Scena sa dodatnim sadržajem



Slika 5.8 Scena sa podešavanjima



Slika 5.7 Scena sa dodatnim aplikacijama

## 5.2 Broj slika u sekundi

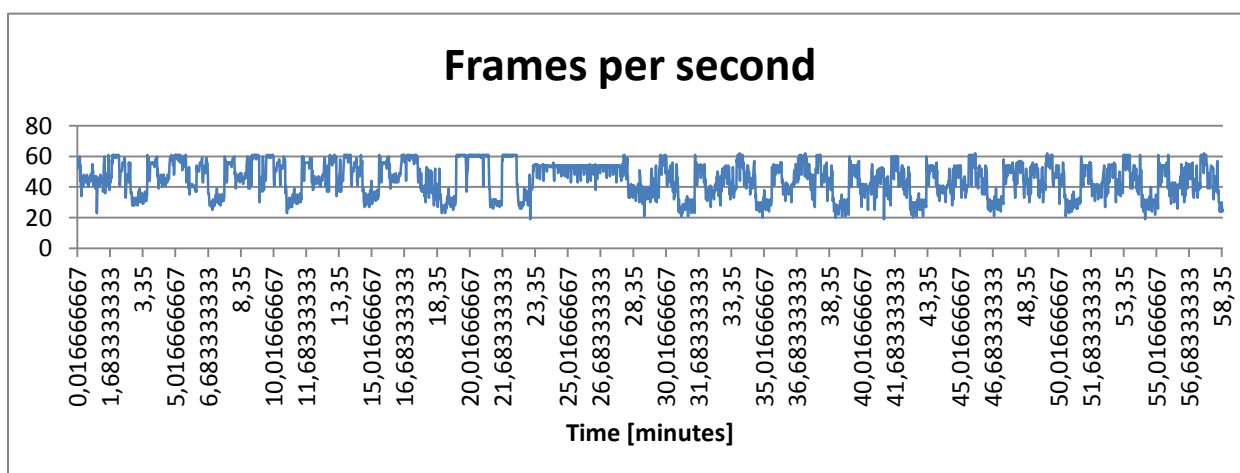
Broj slika u sekundi (eng. *frames per second*, FPS) predstavlja relevantnu vrednost koja utiče na fluentnost cele aplikacije i na korisničko iskustvo. Što je ova vrednost manja, aplikacija ima tendenciju da radi nestabilno uz povremena usporenja, performanse animacija nisu zadovoljavajuće i grafički interfejs ne odgovara toliko hitro na korisničku akciju.

Maksimalna vrednost koju je moguće dostići je 60 FPS. Minimalna vrednost, mereno po korisničkom iskustvu je 24 FPS [21][22], sve ispod ove vrednosti će biti funkcionalno, ali će korisnički doživljaj znatno da se pogorša.

Na grafiku prikazanom na Slici 5.9 vide se izmerene vrednosti broja slika u sekundi u toku jednog sata prilikom korišćenja aplikacije.

Vrednost na grafiku nikada nije manja od 20 FPS. Nalazi se između 20 FPS i 24 FPS samo tokom preuzimanja veće količine sadržaja sa Interneta, što se desi kod ulazaka u *Scene* poput *Programskog vodiča*, *Videa na zahtev*, prolaska kroz *Info dijalog*. Prilikom implementacije je uloženo vreme u to da se ova neizbežna uska grla prerasporede i da se dešavaju što više kada hitar odziv nije važan i nema animacija koje bi mogle da se uspore. Na primer, kod velike količine slika učitavaju samo one koje su korisniku vidljive, a ostale kada je korisnički interfejs statičan.

Vrednost od 60 FPS se dostiže svakih nekoliko sekundi, kao što se vidi na Grafiku 5.1.

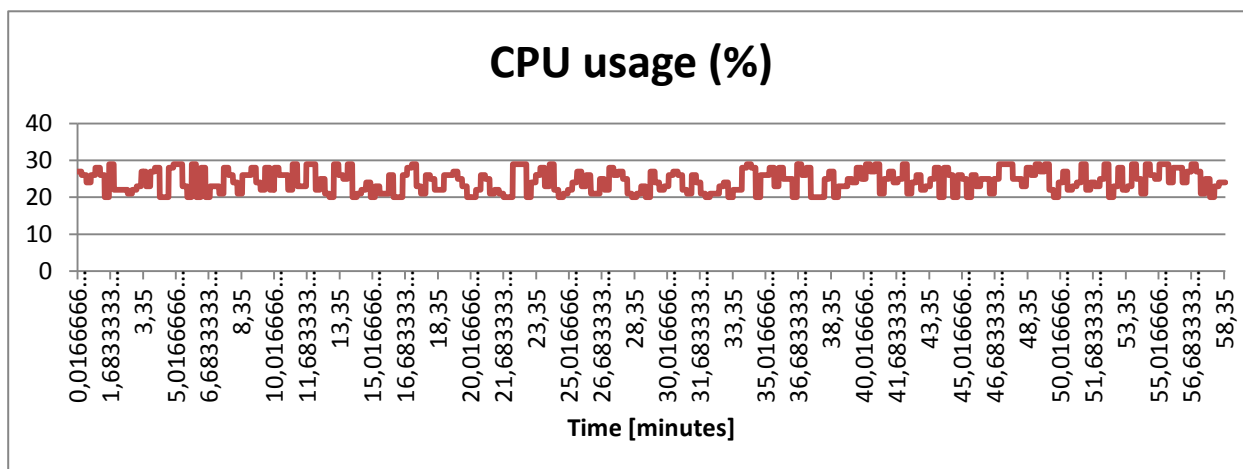


Slika 5.9 Grafik broja slika u sekundi (FPS) u trajanju od sat vremena korišćenja aplikacije

## 5.3 Iskorišćenost procesorske moći

Da bi utvrdili da je korišćenjem OpenGL ES biblioteke zaista dobijeno hardversko ubrzanje, odnosno iscrtavanje 3D okruženja je prebačeno na GPU jedinicu, merena je iskorišćenost CPU jedinice.

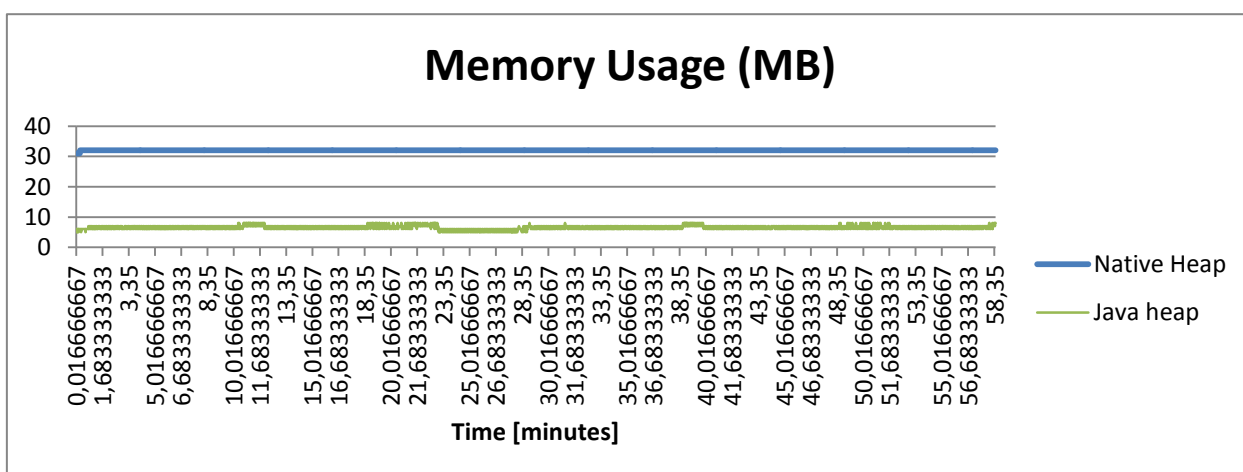
Kako je vrednost zauzetosti centralnog procesora na grafiku između 20% i 30% (videti Sliku 5.10), pokazano je da nema dodatnih računanja radi simulacije 3D efekata. Ovo je dovoljno dobro jer je centralni procesor između 80% i 70% slobodan za druge zadatke.



Slika 5.10 Grafik iskorišćenja centralnog procesora u trajanju od sat vremena korišćenja aplikacije

## 5.4 Zauzeće radne memorije

Kada se govori o RAM memoriji ovog rešenja, Java programski jezik zauzima memoriju za svoje objekte. Kako je Android deo implementacije realizovan u Java programskom jeziku, a ostatak je u nativnom kodu (u programskim jezicima C i C++), zauzetost Java i nativne memorije su merene odvojeno (Slika 5.11).



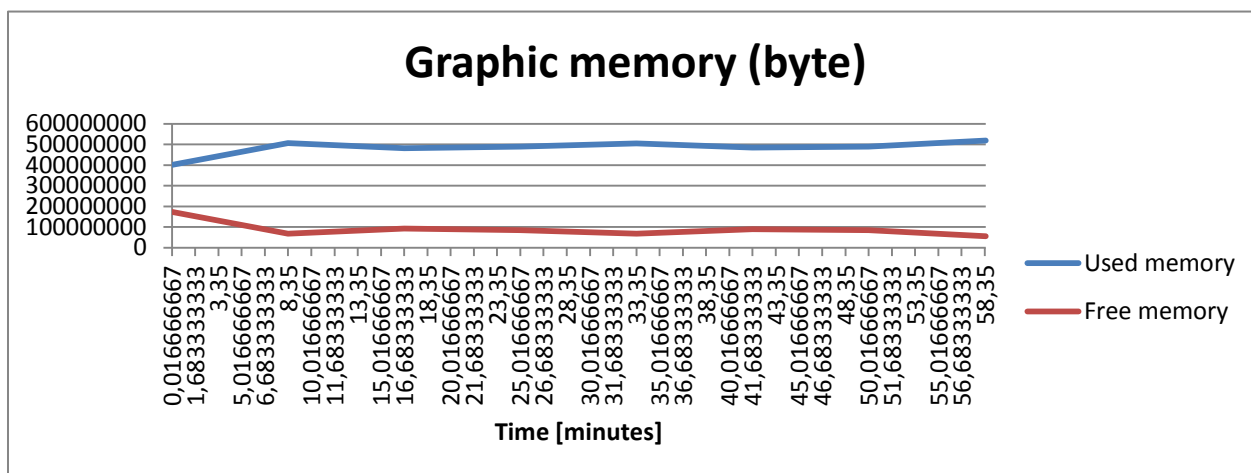
Slika 5.11 Grafik iskorišćenja memorije u trajanju od sat vremena korišćenja aplikacije

Potrošnja nativne memorije tokom gotovo celih sat vremena ostaje na istom nivou, odnosno zauzetost iznosi 32MB. Zauzetost Java memorije se kreće između vrednosti 5 MB i 8 MB, što znači da nema većih oscilacija. U toku sat vremena intenzivnog korišćenja aplikacije,

zauzetost memorije je bila relativno nepromenljiva, što govori o odsustvu problema „curenja memorije”.

## 5.5 Zauzeće grafičke memorije

Kod korišćenja aplikacije merena je potrošnja grafičke memorije, da bi se verifikovalo da mehanizmi automatskog oslobađanja nekorišćenih grafičkih objekata održavaju grafičku memoriju bez većih oscilacija. Merenje je izvršeno korišćenjem komponente *ion* [23] i rezultati su prikazani na Slici 5.12.



Slika 5.12 Grafik iskorišćenja grafičke memorije u trajanju od sat vremena korišćenja aplikacije

Vrednosti su uzimane na svakih 10 minuta u toku rada aplikacije u trajanju od jednog časa.

Zauzeće memorije nema ekstremnih vrednosti u toku rada aplikacije, što potvrđuje da je rad sa grafičkom memorijom regularan.

Sa obzirom da se za iscrtavanje kompletne grafike aplikacije na ekranu koristi grafička memorija, ona je većinom zauzeta, ali u svakom momentu postoji jedan slobodan deo koji se održava nezauzetim.

Izmerena je i vrednost zauzetosti memorije pre početka rada aplikacije, kao i nakon završetka, radi verifikacije da je aplikacijom zauzeta memorija i oslobođena. Rezultati se nalaze u Tabeli 5.1 i potvrđuju potpuno oslobodjenje zauzete memorije nakon zatvaranja aplikacije.

| Objašnjenje                     | Zauzet prostor (bajt) | Slobodan prostor (bajt) |
|---------------------------------|-----------------------|-------------------------|
| Pre početka rada aplikacije     | 103837696             | 469733376               |
| Nakon završetka rada aplikacije | 103837696             | 469733376               |

Tabela 5.1 Prikaz zauzeća memorije pre početka i nakon završetka rada aplikacije u bajtima

## 6. Zaključak

Sa razvojem mobilnih tehnologija, porasla je popularnost Android OS. Danas postoji nekoliko rešenja TV aplikacije na Android platformi, realizovanih na digitalnom prijemniku. Uz već veliki broj opštenamenskih aplikacija realizovanih za Android OS, on predstavlja adekvatnu platformu za TV aplikaciju „sledeće generacije”. Korisničko iskustvo je glavni parametar validnosti aplikacije, a uz to se nalazi i problem 3D grafike koja danas postaje popularna. Tržišni zahtev je brz odziv na korisničku akciju, a zahteva se bogat korisnički interfejs. Da bi rešenje ovog rada bilo ostvareno, odlučeno je da se ne koristi ugrađeno grafičko okruženje Android platforme, već je implementirano grafičko okruženje adekvatno za rešavanje postavljenih zahteva.

Problemi TV aplikacije „sledeće generacije” su rešeni implementacijom opisanom u ovom radu. Zahvaljujući biblioteci libGDX, okruženje se može preneti na Android, BlackBerry, iOS i HTML5.

Na projekat kom pripada TV aplikacija i grafičko okruženje opisani u ovom radu je potrošeno toliko vremena da je po pojavljivanju aplikacija ispratila tržišne trendove. Iako ispunjava zahteve moderne TV aplikacije, ona predstavlja i osnov za dalji rad.

Plan za dalji razvoj aplikacije je u skladu sa industrijom podložan izmenama, a u ovom trenutku se odnosi na procenu uticaja na korisnike kroz korisničke eksperimente, poboljšanja parametara koji utiču na korisnički doživljaj, optimizaciju broja koraka kretanja, dalju optimizaciju korišćenja resursa.

U zavisnosti od industrijskih zahteva, izvesno je ispitivanje rada aplikacije na platformama koje nisu Android.

## 7. Literatura

- [1] M. Murphy: *Beggining Android*, Apress, 2009.
- [2] <https://www.android.com/versions/lollipop-5-0/>, jun 2015.
- [3] J. Bloch: *EffectiveJava (2nd Edition)*, Addison-Wesley, 2008.
- [4] N. Bencomo, F. Losavio, A. Matteo: *Java implementations of user-interface frameworks*, Technology of Object-Oriented Languages and Systems, 1997.
- [5] <https://source.android.com/source/licenses.html>, jun 2015.
- [6] [https://en.wikipedia.org/?title=Set-top\\_box](https://en.wikipedia.org/?title=Set-top_box), jun 2015.
- [7] H. Fangliang, C. Yue, Y. Yong, Z. Hao, T. Hao: *ZiTime studio: A rapid generation framework of 3D user interface based on mobile platform*, Audio, Language and Image Processing (ICALIP), 2014.
- [8] D. Lin, Q. Peng, S. Jianping, S. Wenjuan, X. Yan, Z. Wei: *Immersive 3D user interface for 3D TVS*, 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2011.
- [9] Munshi: *OpenGL ES 2.0 Programming Guide*, Addison-Wesley Professional, 2008.
- [10] <http://libgdx.badlogicgames.com/>, jun 2015.
- [11] J. Nielsen: *Usability Engineering*, Morgan Kaufmann, 1993.
- [12] N. Jovanov, Đ. Kovačević, S. Pejić, D. Nađ, G. Velikić: *One solution of visualising Internet based EPG data combined with additional Internet content on an Android based set-top box*, The 4th IEEE International Conference on Consumer Electronics – Berlin, 2014.
- [13] M. Robin: *Digital Television Fundamentals*, McGraw-Hill Education, 2000.
- [14] J. Johnson: *GUI Bloopers 2.0, Second Edition: Common User Interface Design Don'ts and Dos (Interactive Technologies)*, Morgan Kaufmann, 2007.

- 
- [15] E. Reiss: *Usable Usability: Simple Steps for Making Stuff Better*, Wiley, 2012.
  - [16] <http://dtv.nagra.com/products-and-solutions/connected-consumer-technologies/opentv-middleware/>, jun 2015.
  - [17] M. Milanović, B. Pavlović, I. Petrović, T. Maruna: *Jedno rešenje grafičke korisničke sprege TV aplikacije na prijemniku digitalnog televizijskog signala zasnovanog na Android platformi*, TELFOR, 2013.
  - [18] <https://www.bouyguestelecom.fr/offres-internet/bbox-miami>, jun 2015.
  - [19] M. Bjelica, I. Papp, N. Teslić: *Moderni pravci razvoja grafičkih korisničkih sprega za STB i TV uređaje*, ETRAN, 2015.
  - [20] N. Jovanov, V. Ilkić, M. Knežević, N. Smiljković: Design solution of 3D graphical user interface for television receivers, IEEE CE Workshop Novi Sad, 2015.
  - [21] S. Schaefermeyer: *Digital Video BASICS*, Thomson/Course Technology, 2008.
  - [22] [http://web.archive.org/web/20110708155615/http://www.cinemaweb.com/silentfilm/bookshelf/18\\_kb\\_2.htm](http://web.archive.org/web/20110708155615/http://www.cinemaweb.com/silentfilm/bookshelf/18_kb_2.htm), jun 2015.
  - [23] <http://android-revolution-hd.blogspot.com/2013/08/the-android-ion-memory-manager.html>, jun 2015.

## Prilog

```

package com.rtrk.cmore.app.components;

import com.rtrk.cmore.ui.framework.assets.Assets;
import com.rtrk.cmore.ui.framework.entities.AlphaAnimation;
import com.rtrk.cmore.ui.framework.entities.Image;
import
com.rtrk.cmore.ui.framework.entities.Interpolation.InterpolationFunc
tion;
import com.rtrk.cmore.ui.framework.entities.Vector2D;
import com.rtrk.cmore.ui.framework.entities.Vector3D;
import com.rtrk.cmore.ui.framework.focus.MovingFocus;
import com.rtrk.cmore.ui.framework.listeners.ImageListener;
import com.rtrk.cmore.ui.framework.listeners.KeyConsumer;
import com.rtrk.cmore.ui.framework.listeners.KeyListener;
import com.rtrk.cmore.ui.framework.utils.KeyCode;
import com.rtrk.cmore.ui.framework.utils.UnitConverter;
import com.rtrk.cmore.ui.framework.world.widget.Widget;
import com.rtrk.cmore.ui.framework.world.widget.WidgetGroup;
import
com.rtrk.cmore.ui.framework.world.widget.custom.PulsatingImageWidget;
import
com.rtrk.cmore.ui.framework.world.widget.custom.WaypointWidgetList;
import com.rtrk.cmore.ui.framework.world.widget.provider.Provider;

import java.util.ArrayList;

public class TimelineWidgetGroup extends WidgetGroup {

    private static final Vector3D timelineDetailsPosition = new
Vector3D(UnitConverter.getInstance().convertToUnits(0),
UnitConverter.getInstance().convertToUnits(-352), 0f);
    private static final Vector3D eventListPosition = new
Vector3D(UnitConverter.getInstance().convertToUnits(570),
UnitConverter.getInstance().convertToUnits(90), 0f);

    private static final Vector2D timelineDetailsSize = new
Vector2D(UnitConverter.getInstance().convertToUnits(862),
UnitConverter.getInstance().convertToUnits(237));
    private static final Vector2D eventListSize = new

```

---

```

Vector2D(UnitConverter.getInstance().convertToUnits(400),
UnitConverter.getInstance().convertToUnits(1000));

    private static final String focusImageUri =
    "Vod_details_selector.png";
    private static final int timelineShownMaximum = 4;

    private TimelineDetailsWidgetGroup timelineDetails;

    private WaypointWidgetList eventList;

    private PulsatingImageWidget focusWidget;

    public TimelineWidgetGroup(Provider<Widget> provider) {

        timelineDetails = new
TimelineDetailsWidgetGroup(timelineDetailsSize, "", "", "", "", "");
        registerChild(timelineDetails);
        timelineDetails.setPosition(timelineDetailsPosition);

        ArrayList<Vector3D> positions = new ArrayList<Vector3D>();

        Vector3D pos0 = new
Vector3D(UnitConverter.getInstance().convertToUnits(144.0f),
UnitConverter.getInstance().convertToUnits(777.9f),
UnitConverter.getInstance().convertToUnits(-138) * 0.625f);
        Vector3D pos1 = new
Vector3D(UnitConverter.getInstance().convertToUnits(144.0f),
UnitConverter.getInstance().convertToUnits(467.9f),
UnitConverter.getInstance().convertToUnits(-138) * 0.625f);
        Vector3D pos2 = new
Vector3D(UnitConverter.getInstance().convertToUnits(144.0f),
UnitConverter.getInstance().convertToUnits(157.9f),
UnitConverter.getInstance().convertToUnits(-138) * 0.625f);
        Vector3D pos3 = new
Vector3D(UnitConverter.getInstance().convertToUnits(144.0f),
UnitConverter.getInstance().convertToUnits(-152.1f),
UnitConverter.getInstance().convertToUnits(-138) * 0.625f);
        Vector3D pos4 = new
Vector3D(UnitConverter.getInstance().convertToUnits(55.0f),
UnitConverter.getInstance().convertToUnits(-457.1f),
UnitConverter.getInstance().convertToUnits(0) * 0.625f);
        Vector3D pos5 = new
Vector3D(UnitConverter.getInstance().convertToUnits(55.0f),
UnitConverter.getInstance().convertToUnits(-762.1f),
UnitConverter.getInstance().convertToUnits(0) * 0.625f);

        positions.add(pos5);
        positions.add(pos4);
        positions.add(pos3);
        positions.add(pos2);
        positions.add(pos1);
        positions.add(pos0);

        AlphaAnimation listElementsFadeOut = new AlphaAnimation();
        listElementsFadeOut.setDuration(1f);

        eventList = new WaypointWidgetList(provider, positions, new

```

```

Vector2D(eventListSize.x, eventListSize.y), 0);
eventList.setCroppingOff();
eventList.setOnKeyListener(new TimelineListKeyListener());
eventList.setFadeOutBeginAnimation(listElementsFadeOut);
eventList.setFadeOutEndAnimation(listElementsFadeOut);
eventList.setAnimationDuration(0.35f);

registerChild(eventList);

eventList.setPosition(eventListPosition);

this.setActiveChild(eventList);

focusWidget = new PulsatingImageWidget();
focusWidget.setEndingTransparency(0.75f);
focusWidget.setPulsatingDuration(1.5f);
focusWidget.setPosition(new Vector3D(0, 0, -0.00001f));
focusWidget.setParent(eventList);

Assets.getInstance().loadImage(focusImageUri, new
ImageListener() {

    @Override
    public void onLoadingFailed(ImageListenerError errorCode) {
    }

    @Override
    public void onImageLoaded(Image image) {
        focusWidget.setSize(new
Vector3D(UnitConverter.getInstance().convertToUnits(image.getWidth()),
UnitConverter.getInstance().convertToUnits(image.getHeight()), 0f));
        focusWidget.setImage(image, null);
    }
});

MovingFocus movingFocus = new MovingFocus();
movingFocus.setFocusWidget(focusWidget);
movingFocus.setScalingFactor(new Vector2D(1.8f, 2.3f));
movingFocus.setDuration(0.5f);

eventList.setFocusHandler(movingFocus, false);
}

@Override
public boolean triggerClick() {

    if (clickListener != null) {
        if (clickListener.onClick(this)) {
            return true;
        }
    }

    if (activeChild != null) {
        if (activeChild.triggerClick()) {
            return true;
        }
    }
else if (defaultActiveChild != null) {
        if (defaultActiveChild.triggerClick()) {

```

---

```

        return true;
    }
}

return false;
}

@Override
public boolean triggerKey(int keyCode, KeyEvent keyEvent) {

    if (keyListener != null) {
        if (keyListener.onKey(this, keyCode, keyEvent)) {
            return true;
        }
    }

    if (activeChild != null) {
        if (activeChild.triggerKey(keyCode, keyEvent)) {
            return true;
        }
    } else if (defaultActiveChild != null) {
        if (defaultActiveChild.triggerKey(keyCode, keyEvent)) {
            return true;
        }
    }

    if (keyCode == KeyCode.ARROW_UP) {
        return eventList.triggerKey(KeyCode.ARROW_LEFT, keyEvent);
    } else if (keyCode == KeyCode.ARROW_DOWN) {
        return eventList.triggerKey(KeyCode.ARROW_RIGHT, keyEvent);
    }

    return false;
}

@Override
public void setAlphaAnimation(AlphaAnimation transparencyAnimation)
{
    super.setAlphaAnimation(transparencyAnimation);
    focusWidget.setAlphaAnimation(transparencyAnimation);
}

@Override
public void setAlpha(float alpha) {
    super.setAlpha(alpha);
    focusWidget.setAlpha(alpha);
}

@Override
public void startAlphaAnimation() {
    super.startAlphaAnimation();
    focusWidget.startAlphaAnimation();
}

public void startEntranceAnimation() {
    AlphaAnimation transparencyAnimation = new AlphaAnimation();

```

```

transparencyAnimation.setInterpolationFunction(InterpolationFunction.Q
UAD_OUT);
    transparencyAnimation.setDuration(1f);
    transparencyAnimation.setAlpha(1f);

    int elementCount = Math.min(eventList.getChildrenCount(),
timelineShownMaximum);

    this.setAlpha(1f);

    for (int i = 0; i < elementCount + 1; i++) {
        transparencyAnimation.setDelay(0.1f * i);

        if (i < elementCount) {
            ((InfoBannerElementWidget)
eventList.getChild(i)).setAlpha(0f);
            ((InfoBannerElementWidget)
eventList.getChild(i)).setAlphaAnimation(transparencyAnimation);
            ((InfoBannerElementWidget)
eventList.getChild(i)).startAlphaAnimation();
        } else {
            timelineDetails.setAlpha(0f);
            timelineDetails.setAlphaAnimation(transparencyAnimation);
            timelineDetails.startAlphaAnimation();
        }
    }
}

public void startExitAnimation() {

    setAlpha(1f);

    AlphaAnimation transparencyAnimation = new AlphaAnimation();

transparencyAnimation.setInterpolationFunction(InterpolationFunction.Q
UAD_OUT);
    transparencyAnimation.setDuration(1f);
    transparencyAnimation.setAlpha(0f);

    setAlphaAnimation(transparencyAnimation);
    startAlphaAnimation();

}

public WaypointWidgetList getEventList() {
    return eventList;
}

public TimelineDetailsWidgetGroup getTimelineDetails() {
    return timelineDetails;
}

public void setTimelineDetails(TimelineDetailsWidgetGroup
timelineDetails) {
    this.timelineDetails = timelineDetails;
}

```

```
public void updateDetails(TimelineDetails newDetails) {
    timelineDetails.setTitle(newDetails.getTitle());
    timelineDetails.setTime(newDetails.getTime());
    timelineDetails.setDescription(newDetails.getDescription());
    timelineDetails.setDate(newDetails.getDate());
    timelineDetails.setStared(newDetails.getStared());
    if (newDetails.getStared())
        timelineDetails.getStaredImage().show();
    else
        timelineDetails.getStaredImage().hide();
}

public static class TimelineDetails {
    String title;
    String time;
    String date;
    String description;
    String contentItemURI;
    boolean isStared;

    public TimelineDetails(String title, String time, String date,
String description, String contentItemURI, boolean isStared) {
        super();
        this.title = title;
        this.time = time;
        this.date = date;
        this.description = description;
        this.contentItemURI = contentItemURI;
        this.isStared = isStared;
    }

    public String getTitle() {
        return title;
    }

    public String getTime() {
        return time;
    }

    public String getDate() {
        return date;
    }

    public String getDescription() {
        return description;
    }

    public boolean getStared() {
        return isStared;
    }

    public void setStared(boolean isStared) {
        this.isStared = isStared;
    }

    public String getContentItemURI() {
        return this.contentItemURI;
    }
}
```

```

    }
}

private class TimelineListKeyListener implements KeyListener {

    @Override
    public boolean onKey(KeyConsumer keyConsumer, int keyCode,
        KeyEvent keyEvent) {

        if (keyEvent == KeyEvent.KEY_DOWN) {
            switch (keyCode) {
                case KeyCode.ARROW_DOWN: {

                    AlphaAnimation fadeAnimation = new AlphaAnimation();

                    fadeAnimation.setAlpha(1);
                    fadeAnimation.setDuration(0.5f);

                    InfoBannerElementWidget fadeOutChild =
                        (InfoBannerElementWidget) eventList.getCurrentActiveChild();

                    fadeOutChild.getTimeTW().setAlphaAnimation(fadeAnimation);
                    fadeOutChild.getTimeTW().startAlphaAnimation();

                    fadeOutChild.getTimeImageWT().setAlphaAnimation(fadeAnimation);
                    fadeOutChild.getTimeImageWT().startAlphaAnimation();

                    fadeOutChild.getTitleTW().setAlphaAnimation(fadeAnimation);
                    fadeOutChild.getTitleTW().startAlphaAnimation();

                    fadeAnimation.setAlpha(0);

                    int nextActiveChildIndex;
                    if (eventList.getActiveChildIndex() !=
                        eventList.getProviderChildrenCount() - 1) {
                        nextActiveChildIndex =
                            eventList.getActiveChildIndex() + 1;
                    } else {
                        nextActiveChildIndex = 0;
                    }

                    InfoBannerElementWidget fadeInChild =
                        (InfoBannerElementWidget)
                            eventList.getChildFromProvider(nextActiveChildIndex);

                    fadeInChild.getTimeTW().setAlphaAnimation(fadeAnimation);
                    fadeInChild.getTimeTW().startAlphaAnimation();

                    fadeInChild.getTimeImageWT().setAlphaAnimation(fadeAnimation);
                    fadeInChild.getTimeImageWT().startAlphaAnimation();

                    fadeInChild.getTitleTW().setAlphaAnimation(fadeAnimation);

```

---

```

        fadeInChild.getTitleTW().startAlphaAnimation();

        break;
    }
    case KeyCode.ARROW_UP: {

        AlphaAnimation fadeAnimation = new AlphaAnimation();

        fadeAnimation.setAlpha(1);
        fadeAnimation.setDuration(0.5f);

        InfoBannerElementWidget fadeOutChild =
        (InfoBannerElementWidget) eventList.getCurrentActiveChild();

        fadeInChild.getTimeTW().setAlphaAnimation(fadeAnimation);
        fadeInChild.getTimeTW().startAlphaAnimation();

        fadeInChild.getTimeImageWT().setAlphaAnimation(fadeAnimation);
        fadeInChild.getTimeImageWT().startAlphaAnimation();

        fadeInChild.getTitleTW().setAlphaAnimation(fadeAnimation);
        fadeInChild.getTitleTW().startAlphaAnimation();

        fadeAnimation.setAlpha(0);

        int nextActiveChildIndex;
        if (eventList.getActiveChildIndex() != 0) {
            nextActiveChildIndex =
eventList.getActiveChildIndex() - 1;
        } else {
            nextActiveChildIndex =
eventList.getProviderChildrenCount() - 1;
        }

        InfoBannerElementWidget fadeInChild =
        (InfoBannerElementWidget)
eventList.getChildFromProvider(nextActiveChildIndex);

        fadeInChild.getTimeTW().setAlphaAnimation(fadeAnimation);
        fadeInChild.getTimeTW().startAlphaAnimation();

        fadeInChild.getTimeImageWT().setAlphaAnimation(fadeAnimation);
        fadeInChild.getTimeImageWT().startAlphaAnimation();

        fadeInChild.getTitleTW().setAlphaAnimation(fadeAnimation);
        fadeInChild.getTitleTW().startAlphaAnimation();

        break;
    }
}
}

```

---

```
        return false;
    }

}
```