



# УНИВЕРЗИТЕТ У НОВОМ САДУ ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



УНИВЕРЗИТЕТ У НОВОМ САДУ  
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА  
НОВИ САД  
Департман за рачунарство и аутоматику  
Одсек за рачунарску технику и рачунарске комуникације

## ЗАВРШНИ (BACHELOR) РАД

Кандидат: Немања Пајић  
Број индекса: РА184/2013

Тема рада: Реализација апликације мултимедијалног екрана са гестовном контролом засноване на оперативном систему Андроид

Ментор рада: Доц. др Милан Ђелица

Нови Сад, август, 2017



## КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

|   |  |                       |                |
|---|--|-----------------------|----------------|
| Редни број, РБР:  |  |                       |                |
| Идентификациони број, ИБР:  |  |                       |                |
| Тип документације, ТД:  | Монографска документација  |                       |                |
| Тип записа, ТЗ:   | Текстуални штампани материјал  |                       |                |
| Врста рада, ВР:   | Завршни (Bachelor) рад   |                       |                |
| Аутор, АУ:  | Немања Пајић   |                       |                |
| Ментор, МН:   | Доц. др Милан Ђелица   |                       |                |
| Наслов рада, НР:  | Реализација апликације мултимедијалног екрана са гестовном контролом засноване на оперативном систему Андроид  |                       |                |
| Језик публикације, ЈП:  | Српски / латиница  |                       |                |
| Језик извода, ЈИ:   | Српски   |                       |                |
| Земља публиковања, ЗП:  | Република Србија   |                       |                |
| Уже географско подручје, УГП:   | Војводина  |                       |                |
| Година, ГО:   | 2017   |                       |                |
| Издавач, ИЗ:  | Ауторски репринт   |                       |                |
| Место и адреса, МА:   | Нови Сад; трг Доситеја Обрадовића 6  |                       |                |
| Физички опис рада, ФО:<br>(поглавља/страница/цитата/табела/слика/графика/прилога) |  |                       |                |
| Научна област, НО:  | Електротехника и рачунарство   |                       |                |
| Научна дисциплина, НД:  | Рачунарска техника   |                       |                |
| Предметна одредница/Кључне речи, ПО:  |  |                       |                |
| УДК   |  |                       |                |
| Чува се, ЧУ:  | У библиотеци Факултета техничких наука, Нови Сад   |                       |                |
| Важна напомена, ВН:   |  |                       |                |
| Извод, ИЗ:  | Интегрисани склопови система у возилу, који пружају забавне и информационе садржаје, под називом In-Vehicle Infotainment (ИВИ), представљају нову генерацију мултимедијалних система у возилу. Заступљеност Андроид оперативног система у већини модерних телефона, подстиче аутомобилску индустрију да угради такав систем и у возила. Концепт лаког повезивања са мобилним уређајима, повезивање возила на интернет и коришћење већ направљених апликација за мобилне уређаје, представља будућност аутомобилске индустрије. У раду је представљено једно решење апликације мултимедијалног екрана за забавни и информативни садржај у возилу, са гестовном контролом, засноване на оперативном систему Андроид. |                       |                |
| Датум прихватања теме, ДП:  |  |                       |                |
| Датум одбране, ДО:  |  |                       |                |
| Чланови комисије, КО:   | Председник:  | Доц. др Иван Каштелан |                |
|   | Члан:  | Доц. др Миодраг Ђукић | Потпис ментора |
|   | Члан, ментор:  | Доц. др Милан Ђелица  |                |



## KEY WORDS DOCUMENTATION

|   |  |                    |
|---|--|--------------------|
| Accession number, <b>ANO:</b>   |  |                    |
| Identification number, <b>INO:</b>  |  |                    |
| Document type, <b>DT:</b>   | Monographic publication  |                    |
| Type of record, <b>TR:</b>  | Textual printed material   |                    |
| Contents code, <b>CC:</b>   | Bachelor Thesis  |                    |
| Author, <b>AU:</b>  | Nemanja Pajić  |                    |
| Mentor, <b>MN:</b>  | Milan Bjelica, PhD   |                    |
| Title, <b>TI:</b>   | Realization of a gesture controlled multimedia screen application based on Android OS  |                    |
| 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>  | 2017   |                    |
| 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/appendices) |  |                    |
| 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>   |  |                    |
| <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>  | Integrated set of vehicle systems which provide entertaining and informational content, under the name of In-Vehicle Infotainment (IVI), are representatives of a new generation of multimedia vehicle systems. Availability of Android OS on most of modern portable devices is driving the auto industry to build those systems into their products. The concept of simple vehicle to mobile device connection, vehicle to internet connection and usage of already made mobile applications is the future of auto industry. This work presents a solution for a gesture controlled In-Vehicle Infotainment multimedia screen application based on Android OS. |                    |
| Accepted by the Scientific Board on, <b>ASB:</b>  |  |                    |
| Defended on, <b>DE:</b>   |  |                    |
| Defended Board, <b>DB:</b>  | President:   | Ivan Kaštelan, PhD |
|   | Member:  | Miodrag Đukić, PhD |
|   | Member, Mentor:  | Milan Bjelica, PhD |
|   |  | Mentor's sign      |

## **Zahvalnost**

Zahvaljujem se doc. dr Miljanu Bjelici, Tihomiru Andeliću i Nikoli Vasiljevski na stručnoj pomoći i savetima prilikom izrade ovog rada.

Posebno se zahvaljujem članovima svoje porodice na pruženoj podršci tokom celog školovanja.

Na kraju se zahvaljujem institutu RT-RK, kao i svima ostalima koji su omogućili i doprineli realizaciji ovog rada.



# УНИВЕРЗИТЕТ У НОВОМ САДУ

## ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



### SADRŽAJ

|         |  |    |
|---------|--|----|
| 1.      | Uvod .....                               | 1  |
| 2.      | Teorijske osnove .....                   | 3  |
| 2.1     | Multimedijalni ekrani u vozilima .....   | 3  |
| 2.2     | HMI u vozilu .....                       | 5  |
| 2.2.1   | Gestovna kontrola.....                   | 6  |
| 2.2.2   | Sprežne tehnologije.....                 | 6  |
| 2.2.2.1 | Kotrolna mreža (CAN) .....               | 7  |
| 2.2.2.2 | Međuprocesna komunikacija (IPC) .....    | 7  |
| 2.2.3   | Sistemi za pomoć vozaču (ADAS) .....     | 7  |
| 2.3     | Android platforma.....                   | 7  |
| 2.3.1   | Android Auto.....                        | 9  |
| 2.3.2   | Prednosti i mane Androida u vozilu ..... | 9  |
| 2.3.3   | Razvojno okruženje .....                 | 9  |
| 2.3.4   | Dizajn Androida u vozilu .....           | 10 |
| 3.      | Koncept rešenja .....                    | 11 |
| 3.2     | Arhitektura programske podrške .....     | 11 |
| 3.3     | Grafička korisnička sprega .....         | 13 |
| 3.3     | Protokoli komunikacije .....             | 13 |
| 4.      | Programsko rešenje .....                 | 15 |
| 4.2     | Programsko okruženje.....                | 15 |
| 4.3     | Spisak fajlova i modula .....            | 16 |
| 4.3     | Opis rešenja i modula.....               | 17 |
| 4.3.1   | Moduli početnog menija .....             | 17 |



# УНИВЕРЗИТЕТ У НОВОМ САДУ

## ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА



|   |    |
|---|----|
| 4.3.2 Moduli fragmenta aplikacije ..... | 18 |
| 4.3.2.2 Donji meni.....                 | 19 |
| 4.3.2.3 Gornji meni .....               | 19 |
| 4.3.3 Dodatni moduli aplikacije .....   | 21 |
| 5. Rezultati.....                       | 23 |
| 5.2 Testiranje rada aplikacije.....     | 24 |
| 5.3 Ispitivanje performansi.....        | 27 |
| 6. Zaključak .....                      | 30 |
| 7. Literatura.....                      | 31 |

## SPISAK SLIKA

|   |    |
|---|----|
| Slika 2.1: Primer IVI sistema Android Auto .....                                | 4  |
| Slika 2.2: Primer prikaza višestrukih ekrana u vozilu .....                     | 5  |
| Slika 2.3: Android arhitektura.....   | 8  |
| Slika 3.1: Arhitektura sistema.....   | 12 |
| Slika 3.2 Hijerarhija grafičke sprege .....                                     | 13 |
| Slika 3.3: Primer funkcionisanja IPC stek-a.....                                | 14 |
| Slika 4.1: Spisak fajlova i modula.....   | 16 |
| Slika 4.2: Primer slanja poruke preko IPC-a .....                               | 18 |
| Slika 5.1: Ekran na dodir sa ADT-1 fizičkom arhitekturom .....                  | 24 |
| Slika 5.2: Izgled početnog ekrana aplikacije .....                              | 27 |
| Slika 5.3: Potrošnja memorije i procesora pri samom pokretanju aplikacije ..... | 28 |
| Slika 5.4: Potrošnja memorije i procesora pri pokretanju svih menija .....      | 28 |

## **SPISAK TABELA**

|   |    |
|---|----|
| Tabela 1: Testni slučajevi .....                            | 25 |
| Tabela 2: Testni slučajevi .....                            | 26 |
| Tabela 3: Prosečno vreme odziva na uobičajene komande ..... | 28 |

## SKRAĆENICE

|      |  |
|------|--|
| ADAS | - Advanced Driver Assistance System, Napredni sistem za pomoć vozaču |
| API  | - Application Programming Interface, Aplikativna programska sprega   |
| CAN  | - Controller Area Network  |
| HMI  | - Human-Machine Interface, Sprega čovek-računar                      |
| IP   | - Internet Protocol, Internet protocol                               |
| IPC  | - Inter-process communication, Međuprocesna komunikacija             |
| IVI  | - In-Vehicle Infotainment, Info-zabavni sistem u vozilu              |
| SDK  | - Software Development Kit, Skup softverskih alata                   |
| USB  | - Universal Serial Bus, Univerzalna serijska magistrala              |

## 1. Uvod

Potražnja za sadržajem zabavnog i informacionog karaktera u vozilu poslednjih godina je u porastu. Integrисani skupovi sistema u vozilu, koji pružaju zabavne i informacione sadržaje, pod nazivom In-Vehicle Infotainment (IVI), predstavljaju novu generaciju multimedijalnih sistema u vozilu. Tipične karakteristike koje obuhvataju IVI sistemi uključuju upravljanje i reprodukciju audio sadržaja, korišćenje navigacije, povezivanje sa mobilnim telefonom i upućivanje poziva, prikaz video sadržaja na ekranima zadnjih sedišta.

Opšti trendovi koji su zastupljeni u automobilskoj industriji se većim delom odnose na povećanje nivoa bezbednosti i uvođenje novih multimedijalnih sadržaja, i usmereni su uglavnom na:

- autonomnu vožnju;
- kontrolu parkiranja;
- sistem za detekciju pogleda (ako se skrene pogled sa puta);
- sistem za detekciju pešaka na putu;
- kontrolu pokreta ruke;
- više ekrana u vozilu i njihovu integraciju.

Da bi podaci bili prikupljeni i prikazani vozaču, neophodno je sve komponente automobila povezati i omogućiti njihovu komunikaciju u realnom vremenu.

Zastupljenost Android operativnog sistema na većini modernih telefona kao i sve veća potreba za umrežavanjem različitih uređaja, podstiče automobilsku industriju da pristupi ovom trendu koji je samo pre nekoliko godina bio nezamisлив zbog striktnih bezbednosnih

zahteva u automobilskoj industriji. Koncept lakog povezivanja vozila sa mobilnim uređajima, povezivanje vozila na internet i korišćenje već napravljenih aplikacija za mobilne uređaje po svoj prilici predstavlja budućnost automobilske industrije.

U skladu sa novim trendovima i uvođenjem novih tehnologija u oblast automobilske industrije kreiran je zadatak ovog rada koji ima za cilj implementaciju aplikacije multimedijalnog ekrana za zabavu i informativni sadržaj zasnovane na operativnom sistemu Android. Ključne funkcionalnosti koje aplikacija treba da sadrži obuhvataju:

- mapu sa navigacijom;
- reprodukciju audio sadržaja;
- povezivanja telefona sa multimedijalnim sistemom vozila;
- pomoć pri vožnji;
- napredne kontrole za upravljanje multimedijalnim sistemom vozila;
- moderan grafički interfejs.

Uvođenje naprednih multimedijalnih sistema u vozilo posredno ima uticaja na bezbednost iz aspekta upravljanja novim multimedijalnim funkcijama. Jedan od načina da se poboljša nivo bezbednosti jeste kontrola pokretima ruku. Na ovaj način se obezbeđuje da pažnja vozača bude usmerena na aktivnosti vezane za vožnju tako što se prilikom kontrolisanja aplikacije na multimedijalnom ekranu koriste gestovne kontrole pokretima ruke.

Rad je organizovan u 7 poglavlja.

U drugom poglavlju predstavljene su tehnologije koje su korišćene u izradi rada.

Treće poglavlje opisuje arhitekturu sistema aplikacije, i predstavljen je koncept rešenja zadatka.

Četvrto poglavlje opisuje module programske realizacije.

U petom poglavlju je dat opis okruženja za ispitivanje rešenja, navedeni su ključni slučajevi upotrebe i izvršena je verifikacija i ispitivanje performansi rešenja što u ovom slučaju podrazumeva računanje potrošnje procesora i memorije.

Šesto poglavlje sadrži kratak opis šta je urađeno u ovom radu, kao i dalje pravce razvoja.

U sedmom poglavlju je naveden spisak literature korišćene prilikom pisanja rada.

## 2. Teorijske osnove

U ovom poglavlju opisane su tehnologije koje su korišćene pri izradi ovog rada. Prvo poglavlje opisuje IVI sisteme i njihovu primenu u automobilskoj industriji. U sledećem poglavlju opisane su komunikacione tehnologije. Pomoću njih omogućava se komunikacija između vozača i vozila. Na kraju opisana je platforma za razvoj aplikacije. Multimedijalni ekran realizuje se na Android platformi, tako da su izložene osnovne informacije o toj platformi.

### 2.1 Multimedijalni ekrani u vozilima

IVI predstavlja skup fizičke arhitekture i programske podrške u vozilima koji korisnicima pružaju zabavu ili sadržaj informacionog karaktera [1]. Pre početka razvoja IVI sistema, jedini izvor zabavnog sadržaja u vozilima bili su audio sistemi koji su se sastojali od FM radija, kaseta i CD plejera [2]. Međutim, razvojem tehnologije i digitalizacijom sve većeg broja uređaja IVI sistemi su postali jedna od ključnih i najbrže rastućih tehnologija u automobilskoj industriji.

Uz masovno usvajanje pametnih telefona širom sveta pojavilo se novo pitanje: korišćenje povezanih uređaja u automobilu. Prema istraživanju iz 2015. godine koje je AT&T sprovedla sa uzorkom od preko 2.000 američkih ispitanika, "7 od 10 ljudi koristi pametni telefon dok vozi", uključujući društvene medije (40%), web pregledanje (30%), pa čak i video čakanje (10%). Ovo podiže sigurnosne probleme koji se odnose na ometanje pažnje vozača i podstiče automobilsku industriju da integriše te uređaje na siguran način. Automobilske industrije sve više se oslanjaju na programsku podršku i elektronsku ekspertizu auto-dobavljača i tehnoloških kompanija kako bi im pomogli u dizajniranju takvih sistema. Mobilni operativni sistemi kompanija Google i Apple čine najveći deo tržišta za mobilne uređaje. Dve tehnološke kompanije razvile su režime projekcije (Android Auto, Apple

CarPlay) kako bi omogućili mobilnim uređajima da se koriste u vozilima preko glavne jedinice kontrolne table, tako da korisnici vozila ne manipulisu svojim uređajima direktno, koristeći interfejs sa kojim su upoznati, i da budu usresređeni na put [2].

Na slici 2.1 prikazan je izgled jednog IVI sistema Android Auto.



Slika 2.1: Primer IVI sistema Android Auto

IVI sistemi koriste razne audio/video (A/V) interfejse, ekrane osetljive na dodir, tastature i mnoge druge vrste uređaja, zahvaljujući kojima mogu da ponude sledeće:

- A/V funkcije i alate za ostvarivanje dvosmerne komunikacije kao što su standardni radio i CD plejeri, glasovne komande vozila, obavljanje telefonskog razgovora bez korišćenja ruku (eng. Hands-free) (korisnik obavlja razgovor pomoću mikrofona i spikerfona) i mnoge druge vrste interaktivnog audia ili videa;
- Ekrane na zadnjim sedištima koji omogućavaju putnicima gledanje filmova ili drugih vizuelnih medija (Slika 2.2);
- Mogućnost povezivanja mobilnog uređaja - novija vozila imaju čitav niz sistema koji omogućavaju uređajima poput pametnih telefona ili laptop računara povezivanje sa vozilom;
- Automobilske navigacione sisteme;
- Bluetooth i USB vezu;
- Video plejere;

- Internet, WiFi;
- Mnogi IVI sistemi takođe imaju bezbednosne funkcije koje sprečavaju vozače da koriste bilo koju video uslugu ili druge elemente sistema koji bi mogli uticati na njihovu pažnju;
- Head-up displej – HUD (eng. head-up display) – transparentni displej koji prikazuje podatke vozaču bez potrebe skretanja pogleda sa puta;
- Sistem za pomoć vozačima (eng. Advanced Driver Assistance Systems, ADAS).



Slika 2.2: Primer prikaza višestrukih ekrana u vozilu

## 2.2 HMI u vozilu

Interakcija čovek-mašina (eng. Human-Machine Interface, HMI) tradicionalno se sastoji od više sistema koji omogućavaju vozačima da komuniciraju sa svojim vozilom. U današnjem automobilskom dizajnu, HMI takođe prikazuje povratne informacije od vozila prema vozaču [3].

Ova interakcija počinje u trenutku kada vozač otključava vrata automobila, nastavlja se tokom vožnje i završava se trenutkom kada vozač izlazi i zaključava automobil [3].

Neki od najčešće prepoznatljivih HMI sistemskih modula za poboljšanje vozačevog iskustva su ulazak bez ključa, kontrola električnih sedišta, kontrola bočnog ogledala, detekcija pešaka na putu, interakcija višestrukih ekrana u vozilu.

Jedan izazov sa kojim se suočava automobilsko tržište je koliko se brzo može usvojiti i prilagoditi ovim novim tehnologijama. Danas poluprovodničke kompanije nude širok spektar

kvalifikovanih proizvoda za automobilsku industriju sa integrisanim razvojnim alatima kako bi poboljšali dizajne automatskih sistema za dizajniranje, testiranje, optimizaciju i lansiranje dizajna koji bi se inače mogli videti samo na sajmu potrošačke elektronike [3].

HMI tehnologije se mogu podeliti na 4 grupe:

- Mehanički (pedale, menjač, volan, kontroler, ekran na dodir itd);
- Akustički (sirene, signali upozorenja);
- Vizuelni (multimedijalni i klaster ekrani);
- Gestovna kontrola (kontrola ekrana pokretima ruku).

### **2.2.1 Gestovna kontrola**

Kapacitivna tehnologija na dodir nudi fleksibilnost i visok nivo prilagođavanja, omogućavajući automobilskim dizajnerima da spoje nove karakteristike sa već postojećim mehaničkim dizajnom za poboljšanje funkcionalnosti, koristeći kapacitivne ekrane osetljive na dodir, senzor osetljiv na blizinu ili njihovu kombinaciju [3].

Korišćenje uređaja sa ekransom na dodir tokom vožnje može postati teško pri većim brzinama ili lošim uslovima na putu. Sistemi za prepoznavanje gesta služe i za poboljšanje sigurnosti, omogućavajući vozaču da bez skretanja pažnje sa puta može na lak i brz način da kontroliše sistem u automobilu. Jedno od korišćenja gestovne kontrole u automobilu je senzor osetljiv na blizinu koji pruža mogućnost da npr. promenimo pesmu u media plejeru tako što mahnemo levo ili desno ispred senzora, ili da se približavanjem ruke ekranu izazove prelazak sistema u aktivno stanje.

### **2.2.2 Sprežne tehnologije**

U nastavku su predstavljene specijalizovane interne komunikacione mreže koje povezuju komponente unutar vozila. Preko njih IVI sistemi mogu dobiti sve potrebne podatke o automobilu [13].

- CAN – prenosi podatke kao što su informacije o vožnji i pritisku tastera na kontroleru;
- IPC – prenosi informacije sa senzora neposredne blizine (eng. proximity), vrši slanje video poruka, poruka promene stanja;
- LIN – prenosi podatke kao što je podešavanje sedišta, svetlo, zaključavanje vrata;
- Ethernet – prenosi informacije sa radara, kamere, Head Unit aplikacije;
- FlexRay – prenosi kontrole gasa i kočnice.

### 2.2.2.1 Kotrolna mreža (CAN)

Kotrolna mreža (eng. Controller Area Network bus, CAN) ili CAN magistrala [4] je standardna asinhrona serijska magistrala u vozilima dizajnirana sa ciljem da omogući elektronskim kontrolnim jedinicama (eng. Electronic Control Units, ECUs: kočnica, motor, automatski menjac, elektronsko ubrizgavanje goriva, itd.) poznatim kao čvorovi, da međusobno komuniciraju u vozilu bez centralnog računara. Uredaji koji su povezani na CAN mreže su obično senzori, pogoni i drugi kontrolni uređaji. Ovi uređaji su povezani na CAN magistralu preko centralnog računara, CAN kontrolera i CAN primopredajnika.

### 2.2.2.2 Međuprocesna komunikacija (IPC)

Međuprocesna komunikacija (eng. Inter-process communication, IPC) je skup programskih interfejsa koji omogućavaju programeru da koordinira aktivnostima između različitih programskih procesa koji mogu istovremeno da se pokreću u operativnom sistemu [5]. Ovo omogućava programu da istovremeno rukuje mnogim korisničkim zahtevima. Budući da čak i jedan korisnički zahtev može dovesti do više procesa koji se pokreću u operativnom sistemu, procesi moraju da komuniciraju jedni sa drugima. IPC interfejsi to omogućavaju. Aplikacije mogu koristiti IPC, kategorizovani kao klijenti i serveri, gde klijent traži podatke i server odgovara zahtevima klijenta. Mnoge aplikacije su i klijenti i serveri, što se obično vidi u distribuiranom računarstvu.

### 2.2.3 Sistemi za pomoć vozaču (ADAS)

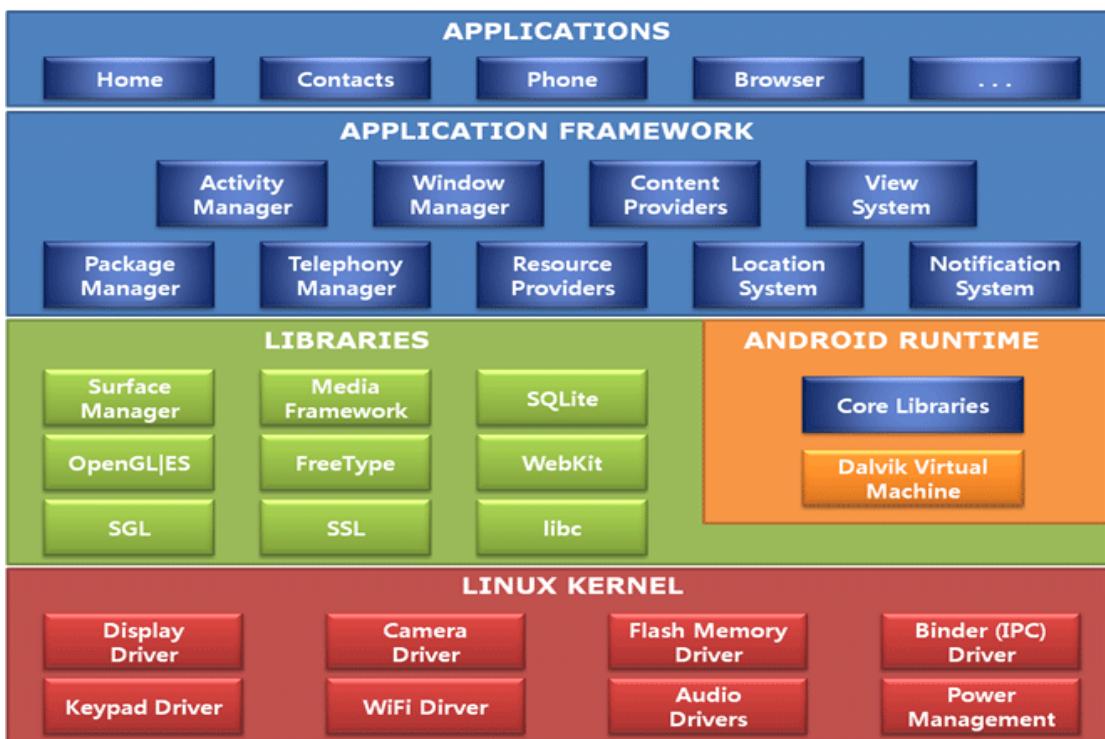
Napredni sistemi za pomoć vozaču (eng. Advanced driver assistance systems, ADAS) su sistemi razvijeni da poboljšaju sigurnost vozača. Sigurnosni dodaci su osmišljeni tako da izbegnu saobraćajne nesreće, nudeći tehnologije koje upozoravaju vozača na potencijalne probleme ili preuzimaju kontrolu nad vozilom [6]. Neki od primera sistema za pomoć vozaču su:

- sistem za detekciju pešaka (eng. Pedestrian Safety Systems);
- prilagodljivi tempomat – ACC (eng. Adaptive Cruise Control);
- pomoć pri parkiranju (eng. Park Assist);
- sistem za prikaz potpunog okruženja vozila (eng. Surround View).

## 2.3 Android platforma

Android operativni sistem je trenutno najrasprostranjeniji operativni sistem za mobilne telefone, zasnovan je na Linux jezgru i prilagođen je prvenstveno za mobilne uređaje sa ekransom osetljivim na dodir.

Android je paket programske podrške za mobilne uređaje koji uključuje operativni sistem, srednji sloj (eng. middleware) i ključne aplikacije. Android SDK obezbeđuje alatke i API potreban za razvoj aplikacija na Android platformi korišćenjem Java programskog jezika. Android obuhvata skup C i C++ biblioteka koje koriste razne komponente Android sistema. Ove mogućnosti su dostupne programeru kroz Android aplikacione okvire.



Slika 2.3: Android arhitektura

Korisnički interfejs Androida je zasnovan na direktnoj manipulaciji objektima na ekranu, korišćenjem ulaza u vidu dodira koji odgovaraju pokretima u realnom svetu. Dodatna fizička arhitektura, kao što su akcelerometar, žiroskop, i senzor blizine, se koristi za dodatne zahteve korisnika, kao na primer podešavanje orijentacije ekrana u zavisnosti položaja uređaja [7].

Android je operativni sistem otvorenog koda dostupan pod Apache licencom. Ona dopušta slobodnu izmenu i distribuciju programske podrške od strane proizvođača uređaja, telekomunikacionih operatera i programera [7]. Android takođe ima veliku zajednicu programera aplikacija, koje proširuju funkcionalnost uređaja i najčešće se razvijaju u programskom jeziku Java. Baš zbog lake dostupnosti koda i velike rasprostranjenosti u uređajima Android sistem je jedan od najpopularnijih operativnih sistema.

### 2.3.1 Android Auto

Android platforma se prvo bitno koristila u mobilnoj industriji, ali se odnedavno ona integriše i u automobile nove generacije. Proizvođači automobila razvijaju aplikacije za multimedijalni ekran jer na taj način održavaju specifičnost i korisnicima omogućavaju upotrebu već poznatog okruženja u automobilu.

Android Auto ima sloj abstrakcije fizičke arhitekture (eng. Hardware abstraction layer, HAL) koji obezbeđuje dosledan programski okvir za Android bez obzira na fizički transportni sloj [8].

Cilj Android Auta je proširenje mogućnosti Android uređaja na kontrolnu tablu automobila. Kako bi se sistem koristio, korisnik mora posedovati uređaj temeljen na Andorid Lollipop ili novijem operativnom sistemu i automobil koji podržava Android Auto nakon čega se Android uređaj spaja na automobil pomoću USB kabla [9]. Umesto pokretanja vlastitog operativnog sistema, kontrolna tabla automobila služi kao spoljni ekran Android uređaja, koji pokreće svu programsku podršku prikazujući korisnički programski okvir specifičan za automobil ugrađen u operativni sistem. Izdat je i Android SDK koji omogućuje programerima da prilagode svoje aplikacije Android Autu.

### 2.3.2 Prednosti i mane Androida u vozilu

Android ima prednost zbog već postojećih funkcija koje su implementirane u mobilnoj industriji, a mogu se preneti i na automobilsku industriju. Prednost u odnosu na druge platforme u automobilu, kao što su AGL i GENIVI, je što donosi i neke pogodnosti iz mobilne industrije kao što je nekoliko proverenih verzija koje su već na tržištu. Iz tog razloga proizvođačima štedi vreme i smanjuje troškove [10].

Manu koju je imala Android platforma jeste što samo jedan prozor može biti vidljiv u određenom vremenskom trenutku, ali pomoću dodataka taj problem se može rešiti i korisnik nema potrebu da izlazi iz aplikacije ukoliko želi da koristi neku od popularnih aplikacija [12].

### 2.3.3 Razvojno okruženje

Android Studio predstavlja razvojno okruženje namenjeno za pisanje Android aplikacija. Google je IntelliJ platformu razvio Android Studio i objavio 2013. godine. On je zamenio Eclipse koji se do njegove pojave koristio za programiranje Android aplikacija. Dostupan je besplatno. Može se koristiti na svim operativnim sistemima. Nalazi se pod Apache 2.0 licencom.

### 2.3.4 Dizajn Androida u vozilu

Android Auto pruža standardizovani korisnički programski okvir i model interakcije korisnika koji funkcioniše na svim vozilima [11]. Dizajneri, ne moraju da brinu o razlikama fizičke arhitekture.

Postoje specifični zahtevi za dizajnom koje aplikacija mora ispuniti da bi se kvalifikovala kao Android Auto aplikacija na Google Play-u. Pridržavajući se ovih zahteva, može se smanjiti napor za izgradnju i testiranje aplikacije.

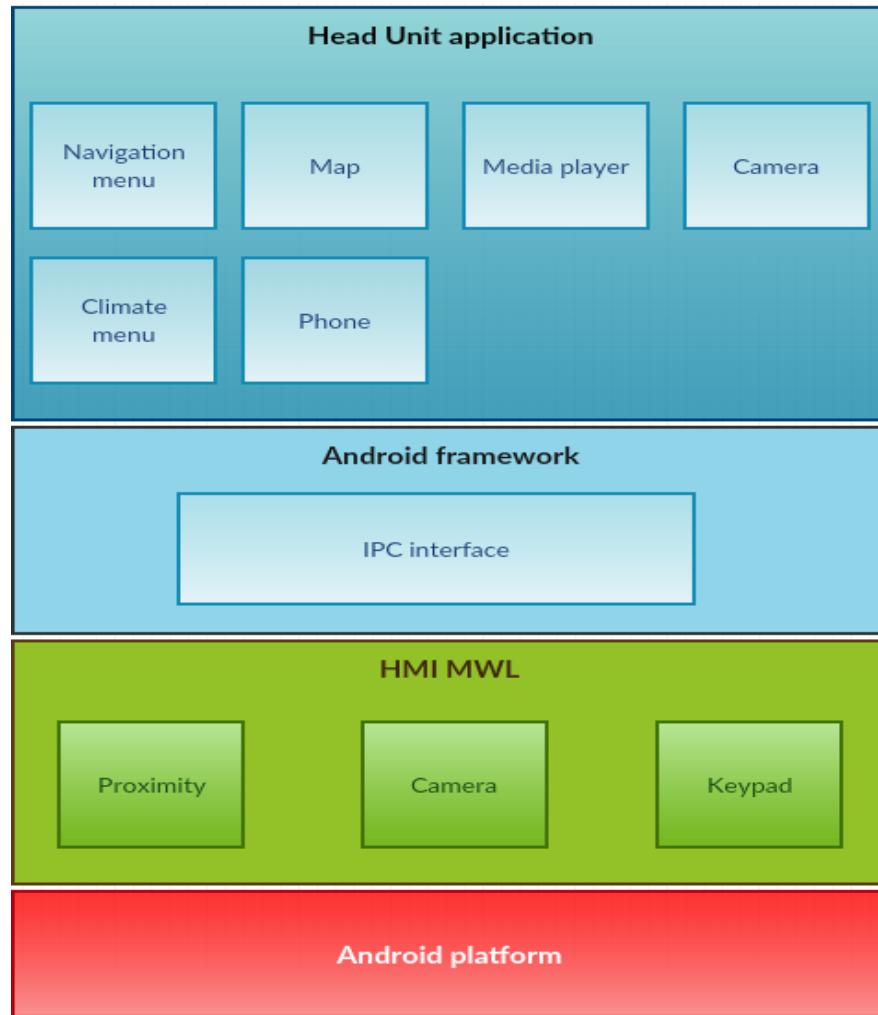
## 3. Koncept rešenja

U ovom poglavlju je cilj da se prikaže arhitektura sistema, kao i protokoli i tehnologije koje su korišćene.

### 3.2 Arhitektura programske podrške

Na slici 3.1 predstavljena je arhitektura sistema hijerarhijski, koja obuhvata četiri međusobno povezana sloja:

- Andorid platforma;
- HMI sprežni sloj (eng. Middleware Layer, MWL);
- Android aplikativni okvir;
- Head Unit aplikacija.



Slika 3.1: Arhitektura sistema

Aplikacija se pokreće na platformi koja podržava Android operativni sistem. Android platforma bazirana je na Linux jezgru i sadrži sve upravljačke programe nižeg nivoa za različite komponente fizičke arhitekture Android uređaja.

Iznad Android platforme nalazi se HMI MWL, koji je implementacija funkcionalnosti i kontrole HMI interfejsa. HMI omogućava komunikaciju korisnika i vozila preko IPC komunikacijskog interfejsa. HMI MWL dobija korisničke ulaze senzora blizine, kamere, tastature i prosleđuje njihove poruke aplikaciji.

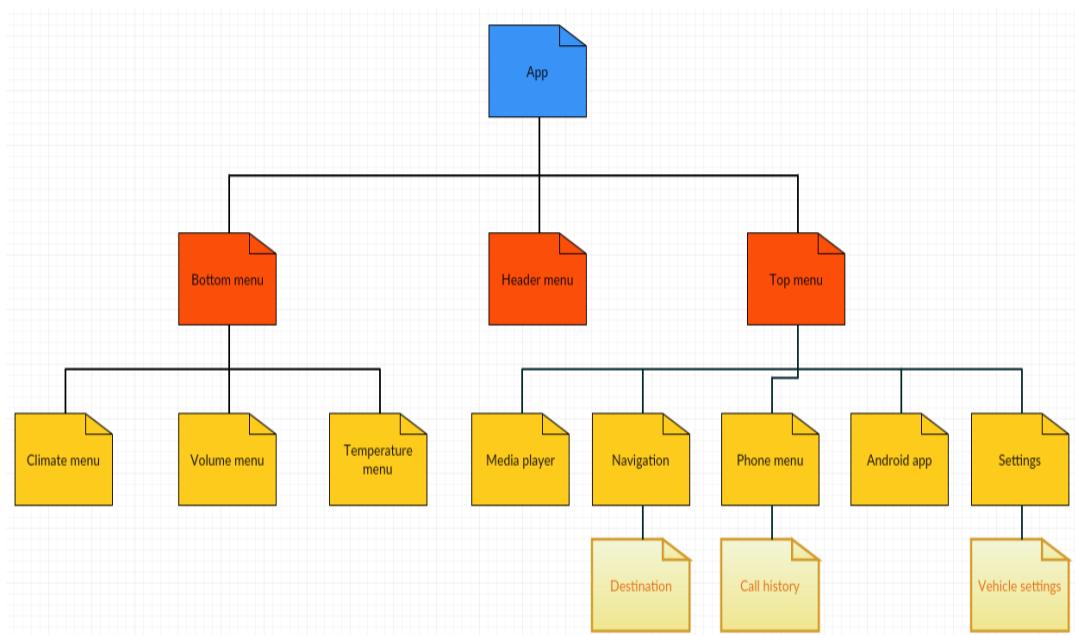
Aplikativni radni okvir (eng. Application Framework) napisan je u programskom jeziku Java i sadrži proširiv skup programskih komponenti kojeg koriste sve aplikacije uređaja. Međuprocesna komunikacija IPC omogućava komunikaciju između HMI sloja i sloja aplikacije, slanjem poruka senzora blizine ili kamere.

Aplikativni sloj je poslednji sloj u arhitekturi sistema i čine ga korisničke aplikacije uređaja. Predstavlja sloj vidljiv krajnjem korisniku. Ovaj sloj obuhvata sve aplikacije koje se isporučuju sa Android uređajima, kao i aplikacije koje se instaliraju sa Google Play.

Aplikativni sloj ove aplikacije sastoji se iz dela za navigaciju, mape, kamere, menija za klimu, telefona.

### 3.3 Grafička korisnička sprega

Head Unit aplikacija predstavlja grafičku korisničku spregu. Izgled aplikacije realizovan je u Android razvojnog okruženju, i aplikacija se kontroliše pomoću ekrana osetljivog na dodir. Ekran osetljiv na dodir jeste posebna vrsta uređaja koja reaguje na pritisak ili dodir, i koji pored prikazivanja informacija na ekranu može detektovati poziciju tačke dodira. Grafički deo u Android okruženju realizuje se u jeziku za označavanje podataka (eng. Extensible Markup Language, XML). Princip realizacije je vrlo jednostavan: odgovarajući sadržaj treba se uokviriti odgovarajućim oznakama koje ga opisuju i imaju poznato, ili lako shvatljivo značenje. Aplikacija se sastoji iz više prozorskih menija, kao što su media plejer, mapa sa navigacijom, prikaz kontrole klime, deo za aplikacije i druge grafičke funkcionalnosti (Slika 3.2).



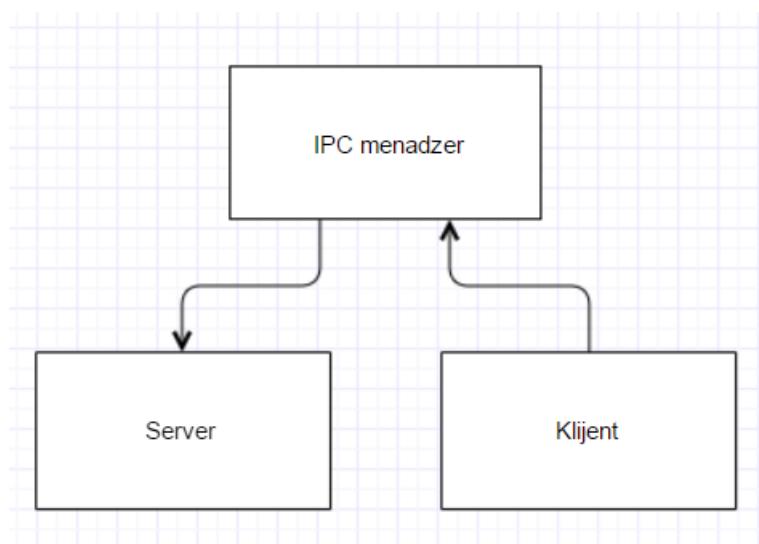
Slika 3.2 Hjерархија граfičке спреge

### 3.3 Protokoli komunikacije

Pokretanjem aplikacije kreiraju se komunikacione veze, IPC menadžer kreira server i klijent na strani aplikacije. Kada je server pokrenut, server čeka da klijent pošalje zahtev za uspostavljanje veze. Ako je uspela veza između klijenta i servera, server će čekati da pristignu poruke od korisnika. Kada primi poruku server je prosleđuje svim fragmentima aplikacije koji u svojoj implementaciji imaju funkciju koja čeka da pristigne poruka od servera. Senzor za

blizinu (eng. Proximity) prilikom približavanja ruke ekranu, ili mahanjem levo i desno šalje poruku preko HMI-a. On se ponaša kao klijent, šalje poruku serveru. Što se tiče klijenta na strani aplikacije on služi za povratne informacije od vozila ka čoveku. Ako je pokrenut video, vozilo šalje poruku ka vozaču sa naznakom da je video pokrenut i da će sad biti aktivne video poruke, umesto poruka koje su zadužene za senzor blizine.

N slici 3.3 opisana je arhitekura IPC stek-a.



Slika 3.3: Primer funkcionisanja IPC stek-a

## 4. Programsко rešenje

U ovom poglavlju dat je prikaz programskog rešenja Android aplikacije za sisteme u vozilu nove generacije. Programsko rešenje sastoji se od grafičkog dela i dela koji kontroliše grafičke sprege.

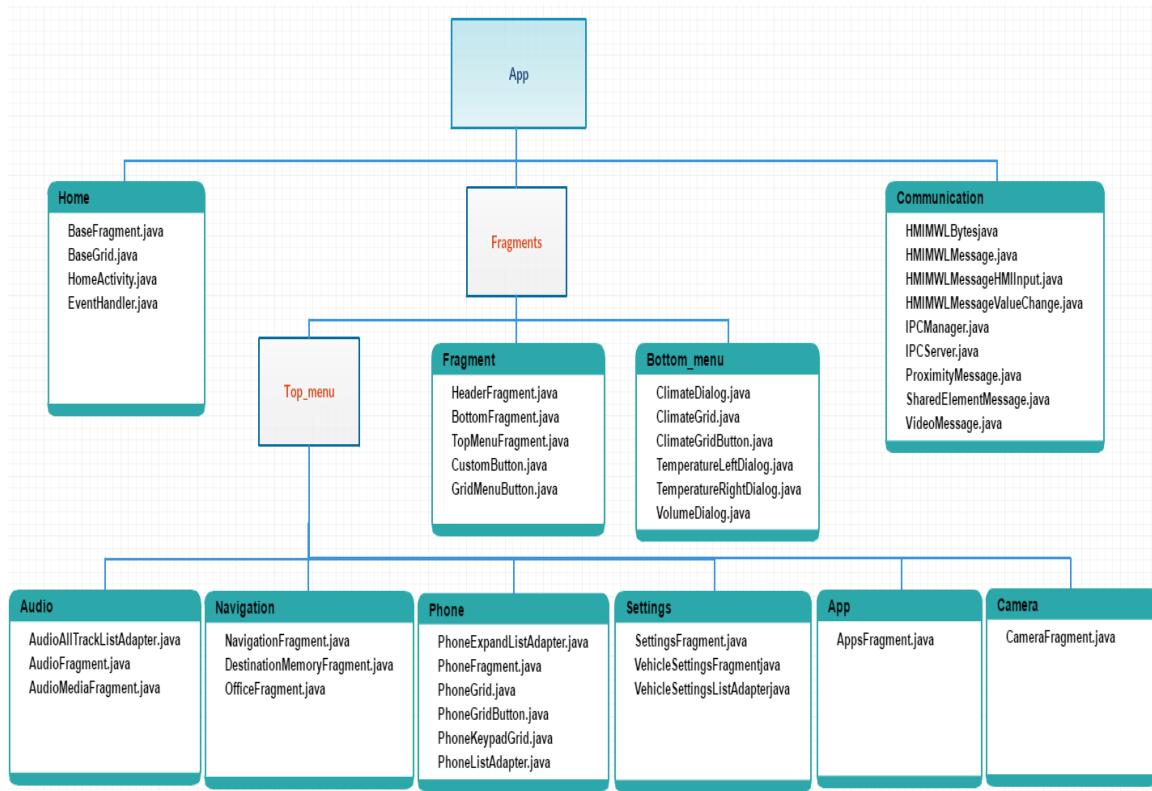
### 4.2 Programsko okruženje

Okruženje za izradu je Android Studio u programskim jezicima Java i XML. Programski jezik Java prilagođen je potrebama i standardima Android operativnog sistema.

Android studio razvojno okruženje omogućava laku ugradnju potrebnih alata i pruža širok spektar mogućnosti u realizaciji Android aplikacija, SDK je kolekcija alata koji pomaže u kreiranju Android aplikacija.

## 4.3 Spisak fajlova i modula

Na slici 4.1 prikazan je spisak fajlova i modula, a detaljno objašnjenje dano je u nastavku.



Slika 4.1: Spisak fajlova i modula

Na slici je prikazana struktura koda. Kod se prvobitno deli na tri modula:

- Home;
- Communication;
- Fragments.

Home i Communication moduli sadrže Java fajlove koji su prikazani na slici. Dalje Fragments modul se deli na tri pod modula:

- Fragment;
- Bottom\_menu;
- Top\_menu.

Fragment i Bottom\_menu pod moduli sadrže Java fajlove sa slike, a Top\_menu se dalje deli na još šest podmodula:

- Audio;
- Navigation;

- Phone;
- Settings;
- App;
- Camera.

Ovi moduli takođe sadrže Java fajlove specifične za njih.

## 4.3 Opis rešenja i modula

U nastavku je opisana funkcionalnost svakog modula aplikacije, dato je objašnjene korišćenih funkcija i klasa.

### 4.3.1 Moduli početnog menija

Pokretanjem aplikacije pokreće se početni video, nakon toga, pre kreiranja mape i početnog menija kreira se klasa **EventHandler** koja je zadužena za funkcionisanje komunikacije između aplikacije i HMI-a preko IPC steka-a. Kreiranjem klase pokreće se server koji će čekati da se na njega poveže klijent od strane korisnika. Nakon toga kreira se početni meni koji u sebi ima mapu, glavni meni, gornji i donji meni.

Kada se prikaže početni meni aplikacija je pokrenuta i spremna je za korišćenje. U nastavku će biti navedeno koji su sve dijalozi, klase i funkcije korišćene u aplikaciji.

**class EventHandler** – omogućava komunikaciju aplikacije sa korisnikom. Pri kreiranju ove klase pokreću se server i klijent i čeka se da se korisnik poveže sa aplikacijom. Da bi se korisnik povezao, IPC menadžeru se prosleđuje određeni ulaz i IP adresa. Nakon toga pozivom funkcija:

- *mIpcManager.startServer();*
- *mIpcManager.startClient();*

pokreću se server i klijent.

Kada je korisnik povezan on je u mogućnosti da šalje poruke aplikaciji, ali isto tako i da prima poruke od aplikacije ili u konkretnom slučaju od vozila. U ovom radu implementirane su dve vrste poruka: poruke od senzora blizine (eng. Proximity) i Video poruke zadužene za kameru. Prvenstveno je potrebno izdvojiti koja je poruka pristigla i nakon toga je proslediti aplikaciji. Slanje se vrši pozivanjem funkcija:

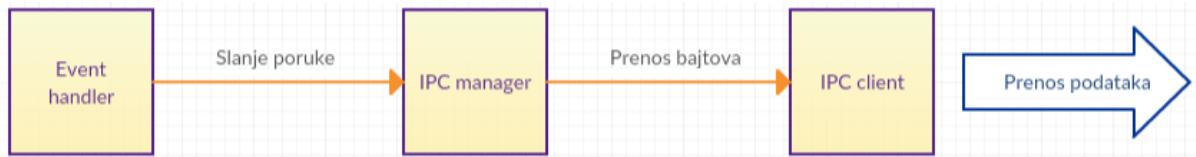
- *handleProximityMessage* – za poruke pristigle od senzora blizine;
- *handleShareMessage* – za poruke pristigle od kamere;

koje su implementirane da šalju poruke preko **intent broadcast-a**.

Usled nedostatka fizičkih komponenti i zbog potreba testiranja, implementirana je simulacija slanja poruka senzora blizine funkcijama:

- `public void sendProximity(ProximityMessage.eProximityAction action);`
- `public void sendVideoMessage(VideoMessage.eVideoActionType action, String filepath, int filePathSize, int seekPosition).`

Funkcije se pozivaju iz početnog menija pritiskom na grafički taster.



Slika 4.2: Primer slanja poruke preko IPC-a

Aplikacija je implementirana korišćenjem većeg broja dijaloga za određenu funkcionalnost.

**HomeActivity** – početni meni. Glavni meni zadužen za inicijalizaciju komunikacije, mape i opšte funkcionisanje aplikacije. Sledeće funkcije su ključne u ovom dijalogu:

- `showIntroVideo(mVideoView)` - funkcija je zadužena za pokretanje početnog videa, kada se video završi kreira se mapa i aplikacija je spremna za korišćenje. Funkcija poziva:
  - `videoView.start()` – funkcija koja pokreće video;
  - `videoView.stopPlayback()` – funkcija koja zaustavlja video;
- `mMapView.onCreate(savedInstanceState)` - funkcija kreira mapu.

### 4.3.2 Moduli fragmenta aplikacije

Pored mape pri pokretanju glavnog menija kreiraju se i tri glavna fragmenta:

- **HeaderFragment** – nalazi se na samom vrhu ekrana i prikazuje, trenutno vreme, procenat baterije, da li je uljučen wi-fi, bluetooth;
- **BotMenuFragment** – meni na samom dnu aplikacije, u njemu se nalaze meni za klimu, temperaturu i jačinu zvuka medija;
- **TopMenuFragment** – nalazi se tačno iznad donjeg menija, opširan meni sa navigacijom, media plejerom itd.

#### 4.3.2.2 Donji meni

**BotMenuFragment** – kreira se pri pokretanju aplikacije, predstavlja donji meni i omogućava biranje nekih od sledećih menija:

- **ClimateDialog** – meni za klimu;
- **VolumeDialog** – meni za zvuk;
- **TemperatureLeftDialog** – meni za temperaturu leve klime;
- **TemperatureRightDialog** - meni za temperaturu desne klime.

Ova 4 dijaloga su prošireni sa Android ugrađenom **DialogFragment** klasom. Ovo omogućava da se pri pokretanju nekog od menija on prikaže preko gornjeg menija i mape.

**class ClimateDialog** – pokreće meni za klimu. Omogućava podešavanje jačine protoka vazduha klime, da li je klima uključena ili isključena. Postoji opcija za automatsko podešavanje, uključivanje i isključivanje grejanja sedišta, sušenje prednjeg i zadnjeg stakla.

**class VolumeDialog** – otvara se meni za podešavanje jačine zvuka.

**class TemperatureLeftDialog** – podešavanje temperature za klimu koja se nalazi na levoj strani auta.

**class TemperatureLeftDialog** – podešavanje temperature za klimu koja se nalazi na desnoj strani auta.

#### 4.3.2.3 Gornji meni

**TopMenuFragment** – meni koji se nalazi iznad donjeg menija, omogućava biranje između informacionog i zabavnog sadržaja kojeg čine:

- **NavigationFragment** – meni za navigaciju;
- **AudioFragment** – meni za audio plejer;
- **PhoneFragment** – meni za telefon;
- **AppFragment** – meni za aplikacije;
- **SettingsFragment** – meni za podešavanja;
- **CameraFragment** – meni za kameru.

**class NavigationFragment** – kreira meni za navigaciju. Iz ovog menija imamo mogućnost da otvorimo meni sa zapamćenim destinacijama.

---

**class DestinationMemoryFragment** – u ovom meniju je implementirano da se pritiskom na **Office** destinaciju otvori meni sa podacima o toj destinaciji, koja je kreirana klasom **OfficeFragment**.

**AudioFragment** – pri pokretanju ovog menija otvara se **AudioMediaFragment** koji predstavlja media plejer.

**class AudioMediaFragment** – pri kreiranju inicijalizuje se media plejer i programska nit (eng. thread). Nakon toga pozivaju se funkcije:

- *mediaPlayer.start()* - pokreće se pesma;
- *public void startTimer(final TextView text, final ProgressBar progressBar)* – pokreće se programska nit pozivom funkcije *thread.start()*, argumenti funkcije su:
  - **TextView** text – tekst koji prikazuje trenutno vreme pesme;
  - **ProgressBar** progressbar – statusna traka koja na osnovu trenutnog vremena pesme prikazuje to vreme vizuelno.

Pritiskom na grafički taster za pauzu privremeno se zaustavlja pesma, ponovnim pritiskom na taj grafički taster, pesma nastavlja gde je predhodno stala. Pritiskom na grafički taster **allTrack**, otvara se lista pesama koje su trenutno dostupne. Još jedna funkcionalnost, da se pritiskom na grafički taster **Radio** prelazi u meni radija.

**class AudioFragment** – otvara meni radio stanice. Tu se automatski pušta pesma, a pritiskom na grafički taster **Media** vraća se u media plejer.

**class PhoneFragment** – početni deo ovog menija je opcija da se izaberu omiljeni brojevi, pored toga postoje grafički tasteri za istoriju poziva, prikaz kontakata, i mogućnost unosa broja i pozivanja istog. To je u ovom momentu dostupno samo kao vizuelni deo, i pritiskom na poziv ispisuje se na ekranu broj koji se poziva.

**class AppFragment** – prikazuje meni sa sadržajem dostupnih aplikacija.

**class SettingsFragment** – otvara se meni sa opcijama podešavanja. Za sad je samo funkcionalan **Vehicle** grafički taster, koji kad se pritisne otvara meni za podešavanje auta.

**class VehicleSettingsFragment** – meni za podešavanje stanja u vozilu.

**class CameraFragment** – pri otvaranju ovog dijaloga pokreće se video.

### 4.3.3 Dodatni moduli aplikacije

Do sad je opisano kako sistem funkcioniše po delovima. U nastavku su predstavljene i objašnjene funkcije i klase koje se koriste u većini gore navedenih klasa i koje omogućavaju da ceo sistem funkcioniše u celini.

Dodatne funkcije koje se koriste su:

***onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState)*** - pozivanjem funkcije, po prvi put se iscrtava korisnički programski okvir fragmenta aplikacije.

***onDestroyView()*** - pozivanjem funkcije, uništava se korisnički programski okvir trenutnog dela.

***setOnLongClickListener()*** - dužim pritiskom na neki grafički taster, sliku ili slično poziva se ova funkcija. Funkcija će uraditi neku akciju, u zavisnosti kako smo je implementirali.

***private BroadcastReceiver()*** – poziva se u većini menija. Uloga ove funkcije je da osluškuje da li se nešto poslalo preko intent-a, što je slučaj sa porukama senzora blizine i video porukama.

Funkcija je u svakoj klasi implementirana da odreaguje u skladu sa potrebama te klase. Tako na primer u većini klase ona prima poruke pristigle od senzora blizine i na osnovu toga prebacuje grafičke tastere u aktivno ili neaktivno stanje. Poseban slučaj je kod media plejera, gde pored poruka za približavanje ruke ekranu postoje i poruke od senzora za levo i desno, pri čemu se tako menjaju pesme.

Funkcija se poziva:

```
IntentFilter filter = new IntentFilter(ProximityFlag);  
context.registerReceiver(customButtonBroadcastReceiver, filter).
```

Klase koje se najviše koriste su:

**class BaseFragment** – klasa je proširena klasom **BaseFragment** i deo je svakog fragmenta koji se koristi.

**class BaseGrid** – proširena je **BaseAdapter** klasom i omogućava da se podese adapteri za sve dvodimenzionalne grupe za prikaz u aplikaciji izuzev dela za klimu. On ima svoj poseban adapter ClimateGrid.

**class CustomButton** – je ručno pravljena komponenta koja se sastoji od pozadine koja izgleda kao grafički taster u stanju aktivnom, a nevidljiva je ako je stanje neaktivno, pozadine preko nje koja će na običan pritisak grafičkog tastera postati siva, a dužim pritiskom će postati bela i izvršiće se određena akcija vezana za taj grafički taster. Na taj način oponaša se funkcionalnost pravog grafičkog tastera. Pored toga u sredini se nalaze tekst i sličica, i u zavisnosti od potreba može se podesiti da jedno od ta dva bude nevidljivo.

Funkcija koja sadrži glavnu funkcionalnost ove klase je:

**private void init(final Context context, AttributeSet attrs)** – u ovoj funkciji implementirane su sve gore navedene funkcionalnosti, takođe i ova funkcija poziva funkciju *BroadcastReceiver()* i na osnovu nje će znati da li je prva pozadina aktivna ili ne.

## 5. Rezultati

Okruženje u kom je testirana aplikacija je ekran na dodir (eng. Touchpad) rezolucije 1980x1080 (slika 5.1). Na ekranu je povezan Android ADT-1 fizička arhitektura. Android razvojni alati ADT dizajnirani su da daju moćno integrisano okruženje u kome se grade Android aplikacije. ADT-1 fizička arhitektura sa velikim ekranom na dodir omogućava pokretanje i preglednost aplikacije. Karakteristike ADT-1 su:

- Sistem na čipu – Nvidia Tegra 4;
- Sistemska memorija – 2GB RAM;
- Memorija za skladištenje – 16GB flash;
- Video izlaz – HDMI;
- Mogućnost povezivanja - Ethernet, 2×2 MIMO dual-channel WiFi, Bluetooth 4.0.



Slika 5.1: Ekran na dodir sa ADT-1 fizičkom arhitekturom

## 5.2 Testiranje rada aplikacije

Cilj testiranja je bio verifikacija svih osnovnih funkcija aplikacije. Pokretanjem aplikacije otvara se početni meni sa mapom. U početnom meniju postoji mogućnost biranja između menija navigacije, klime, media plejera, telefona, aplikacija, podešavanja zvuka i temperature u vozilu. Na vrhu ekrana nalaze se četiri grafička tastera, "left", "right", "approach", "unApproach", koji omogućavaju simuliranje senzora za blizinu,

Ispitani su ključni slučajevi upotrebe. Prvenstveno je provereno da li je izgled programskog okvira u skladu sa zahtevima. Nakon toga otvarani su svi dijalozi i proveravana je funkcionalnost svih grafičkih tastera. Vršena je provera puštanja pesme u medija plejeru, kao i privremeno zaustavljanje i ponovno puštanje od pozicije gde je predhodno zaustavljeno. Više puta su simulirani senzori za blizinu u svakom meniju. Test funkcionalnosti je prikazan u tabeli 1 i tabeli 2.

| Testni slučaj                            | Opis koraka  | Očekivani rezultat   | Uspešnost |
|--|--|--|-----------|
| Puštanje početnog videa                  | 1. Pokrenuti aplikaciju  | Prikazana je video sinhronizacija, aplikacija je spremna za korišćenje   | Uspešno   |
| Uvećavanje mape                          | 1. Pokretima dva prsta prema dole na ekranu, mapa se uvećava.  | Mapa je uvećana  | Uspešno   |
| Umanjivanje mape                         | 1. Pokretima dva prsta prema gore na ekranu, mapa se umanjuje.   | Mapa je umanjena   | Uspešno   |
| Senzor za blizinu                        | 1. Pritisnuti grafički taster "unApproach" u nekom od menija<br>2. Pritisnuti grafički taster "approach" u nekom od menija                       | Simulira se da je ruka udaljena od ekrana.<br>Pritiskom na "unApproach", grafički tasteri su neaktivni. Pritiskom na "approach", grafički tasteri su ponovo aktivni. | Uspešno   |
| Senzor za mahanje ruke ka levo ili desno | 1. Pritisnuti grafički taster "Audio" u početnom meniju<br>2. Kada je otvoren meni media plejera, pritisnuti grafičke tastere "left" ili "right" | Simulira se mahanje ruke levo ili desno, pesma i odgovarajuća slika uz pesmu se menjaju  | Uspešno   |
| Otvaranje menija za navigaciju           | 1. Pritisnuti grafički taster "Navi" u početnom meniju<br>2. Ponovo pritisnuti da bi se meni zatvorio  | Otvara se meni za navigaciju   | Uspešno   |
| Otvaranje menija za medija plejer        | 1. Pritisnuti grafički taster "Audio" u početnom meniju<br>2. Ponovo pritisnuti da bi se meni zatvorio   | Otvara se meni medija plejera  | Uspešno   |
| Otvaranje menija za telefon              | 1. Pritisnuti grafički taster "Phone" u početnom meniju<br>2. Ponovo pritisnuti da bi se meni zatvorio   | Otvara se meni za telefon  | Uspešno   |
| Otvaranje menija za aplikacije           | 1. Pritisnuti grafički taster "App" u početnom meniju<br>2. Ponovo pritisnuti da bi se meni zatvorio   | Otvara se meni za aplikacije   | Uspešno   |
| Otvaranje menija za podešavanja          | 1. Pritisnuti grafički taster "Settings" u početnom meniju   | Otvara se meni za podešavanja  | Uspešno   |
| Otvaranje menija za kameru               | 1. Pritisnuti grafički taster za desno u početnom meniju<br>2. Kada je početni meni pomeren udesno, pritisnuti grafički taster "Camera"          | Otvara se meni za kameru u kom se automatski pušta video   | Uspešno   |
| Otvaranje dijaloga za jačinu zvuka       | 1. Pritisnuti grafički taster "Vol" u početnom meniju  | Otvara se dijalog za podešavanje jačine zvuka  | Uspešno   |
| Otvaranje dijaloga za klimu              | 1. Pritisnuti grafički taster "Climate" u početnom meniju<br>2. Pritiskom na strelicu dole zatvara se meni klime                                 | Otvara se dijalog za klimu   | Uspešno   |
| Puštanje i zaustavljanje pesme           | 1. Pritisnuti grafički taster "Audio" u početnom meniju<br>2. Nakon otvaranja media plejera, pritisnuti na grafički taster "   " ili " >"        | U media plejeru pritiskom na grafički taster " >" pesma se pušta, a pritiskom na "   " pesma se zaustavlja.  | Uspešno   |

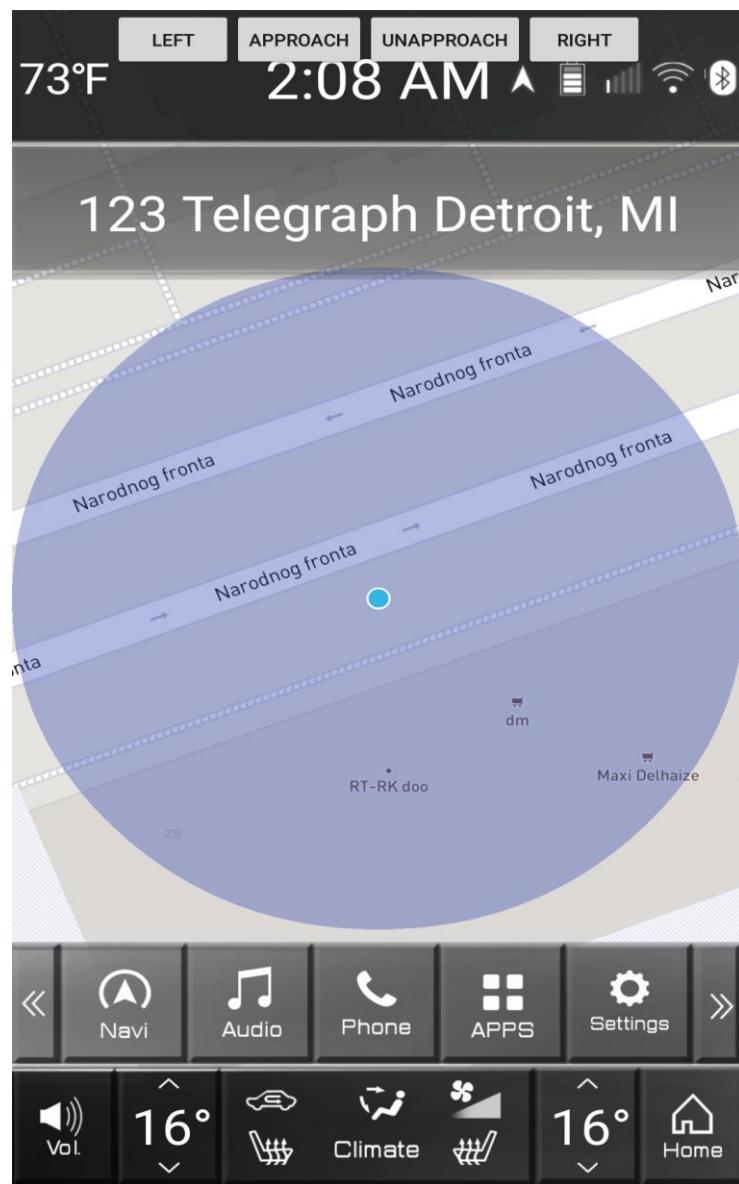
Tabela 1: Testni slučajevi

| Testni slučaj                                   | Opis koraka   | Očekivani rezultat  | Uspešnost |
|---|---|---|-----------|
| Smanjivanje ili povećanje protoka vazduha klime | 1. Pritisnuti grafički taster "Climate" u početnom meniju<br>2. Kada je otvoren dijalog klime, pritisnuti grafički taster "-" ili "+"   | Ako je pritisnut grafički taster "-" smanjuje se protok vazduha klime, žuta traka se smanjuje. Ako je pritisnut grafički taster "+" povećava se protok vazduha klime, žuta traka se povećava.                           | Uspešno   |
| Isključivanje rada klime                        | 1. Pritisnuti grafički taster Climate u početnom meniju<br>2. Kada je otvoren dijalog klime, pritisnuti grafički taster "Off"   | Grafički taster "Off" je obeležen crnom bojom. Protok vazduha klime je ugašen, a traka za obeležavanje je prazna.   | Uspešno   |
| Funkcionalnost menija telefona                  | 1. Pritisnuti grafički taster "Phone" u početnom meniju<br>2. Kada se otvori meni za telefon, pritisnuti na jedan od tri tastera "HISTORY", "PHONEBOOK" ili "KEYPAD"<br>3. Ako je pritisnut taster "KEYPAD", pritisnuti neke od brojeveva i pritisnuti grafički taster "call" | Pritiskom na "KEYPAD" otvara se lista poziva.<br>Pritiskom na "PHONEBOOK" otvara se lista kontakata.<br>Pritiskom na "KEYPAD" otvara se tastatura sa brojevima za pozivanje, pritiskom na "call" poziva se željeni broj | Uspešno   |
| Otvaranje menija zapamćenih destinacija         | 1. Pritisnuti grafički taster "Navi" u početnom meniju<br>2. Kada je otvoren meni navigacije, pritisnuti grafički taster "Destination Memory" u meniju navigacije<br>3. Pritisnuti grafički taster "back" (vraća se u prethodni meni)   | Otvoren je meni zapamćenih destinacija.   | Uspešno   |
| Pokretanje menija za podešavanje vozila         | 1. Pritisnuti grafički taster ">>" u početnom meniju<br>2. Kada je početni ekran pomeran udesno, pritisnuti grafički taster "Vehicle Settings"<br>3. Pritisnuti grafički taster "back" (vraća se u prethodni meni)  | Otvara se meni za podešavanje vozila  | Uspešno   |

Tabela 2: Testni slučajevi

Sve funkcionalnosti proveravane su više puta i uvek su bile uspešne. Aplikacija ni u kom slučaju nije neočekivano prestala sa radom. Pored ADT-1 aplikacija je testirana i na tabletu koji u sebi ima operativni sistem Android. I na tabletu su dobijeni isti rezultati.

Na slici 5.2 predstavljen je početni meni aplikacije.



Slika 5.2: Izgled početnog ekran aplikacije

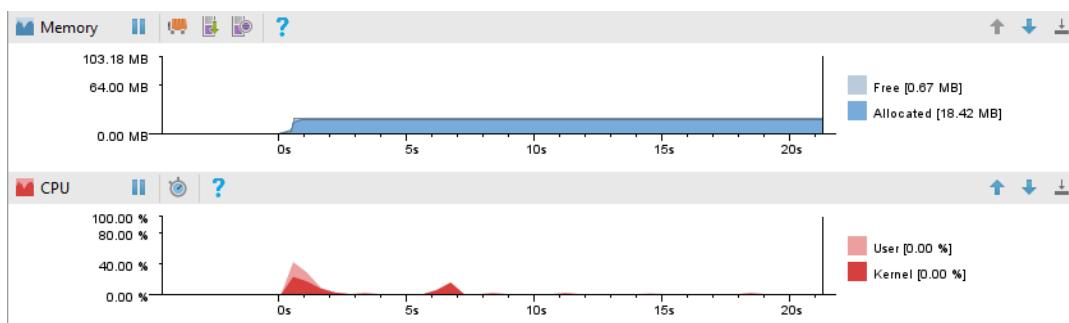
### 5.3 Ispitivanje performansi

Vreme odziva aplikacije testirano je na ADT-1fizičkoj arhitekturi, snimanjem testiranja aplikacije zabeleđeni su podaci brzine odziva. Zabeleženi podaci vremena odziva aplikacije na neke od komandi predstavljeni su u tabeli 3.

| Testirane komande                            | Prosečno vreme odziva [s] |
|--|---------------------------|
| Pokretanje aplikacije                        | 0.20                      |
| Senzor blizine "unApproach"                  | 0.50                      |
| Senzor blizine "approach"                    | 0.50                      |
| Senzor blizine "left"                        | 0.25                      |
| Senzor blizine "right"                       | 0.25                      |
| Otvaranje dijaloga u gornjem meniju          | 0.20                      |
| Otvaranje dijaloga u donjem meniju           | 0.35                      |
| Dugi pritisak na funkcionalne dugmiće        | 0.85                      |
| Stopiranje i ponovno puštanje audio sadržaja | 0.85                      |

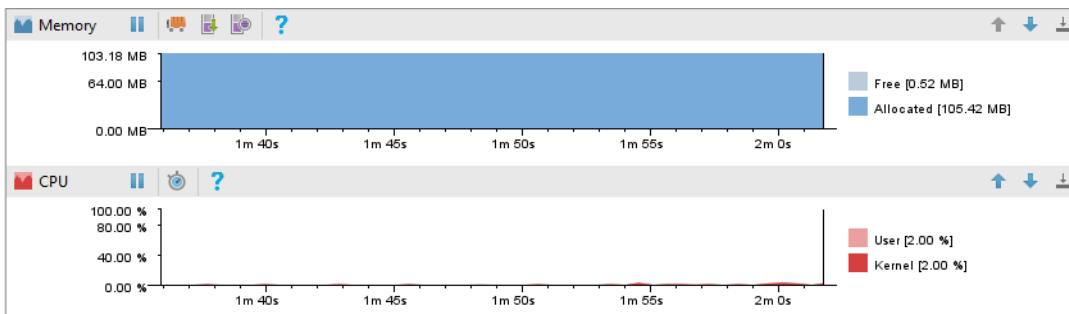
Tabela 3: Prosečno vreme odziva na uobičajjene komande

Pri pokretanju aplikacije zauzetost memorije je oko 20 mb, a potrošnja procesorskog vremena je zanemarljiva.



Slika 5.3: Potrošnja memorije i procesora pri samom pokretanju aplikacije

Kada se aplikacija optereti pokretanjem svih dijaloga, videoa i medija plejera dobijaju se sledeći rezultati:



Slika 5.4: Potrošnja memorije i procesora pri pokretanju svih menija

Maksimalnim opterećenjem aplikacije dobija se veća zauzetost memorije do 105 mb, a procesorko vreme do 2%. Što se tiče procesorkog vremena to je samo privremena zauzetost 2%.

U odnosu na ukupnu količinu memorije koja je na ciljanoj platform 2GB, memoriju koju aplikacija zauzima je procentualno 5%, na osnovu čega možemo zaključiti da je rešenje prikazano u ovom radu primenljivo.

## 6. Zaključak

U ovom radu je predstavljeno jedno rešenje aplikacije multimedijalnog ekrana za zabavni i informativni sadržaj u vozilu, sa gestovnom kontrolom, zasnovane na operativnom sistemu Android.

Ispitane su performanse rešenja, a rezultati su pokazali da je ovo rešenje primenljivo, i da ne daje vidljivo prekoračenje u smislu dodatnog vremena odziva na neke od tipičnih komandi. Dokazana je ispravnost i robustnost aplikacije, a jedini nedostatak testiranja je nemogućnost provere reagovanja aplikacije na senzor blizine, usled nedostatka odgovarajuće elektronike i senzora.

Prostor za dalje unapređenje obuhvata proširivanje funkcionalnosti aplikacije. Neke od budućih funkcionalnosti su podrška za povezivanje mobilnog uređaja i vozila, navigacija sa putokazom, implementacija nekih Android aplikacija koje bi koristile korisnicima u vozilu, ugradnja senzora i kamera, glasovna kontrola.

## 7. Literatura

- [1] <https://www.techopedia.com/definition/27778/in-vehicle-infotainment-ivi> jul 2017.
- [2] [https://en.wikipedia.org/wiki/In-car\\_entertainment](https://en.wikipedia.org/wiki/In-car_entertainment) jul 2017.
- [3] [http://www.eetimes.com/document.asp?doc\\_id=1272865](http://www.eetimes.com/document.asp?doc_id=1272865) jul 2017.
- [4] [https://en.wikipedia.org/wiki/CAN\\_bus](https://en.wikipedia.org/wiki/CAN_bus) jul 2017.
- [5] <http://whatis.techtarget.com/definition/interprocess-communication-IPC>
- [6] A. Hakanović, "Sistemi za pomoć vozaču", Sarajevo, 2011.
- [7] [https://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)](https://en.wikipedia.org/wiki/Android_(operating_system)) jul 2017.
- [8] <https://source.android.com/devices/automotive/> jul 2017.
- [9] [https://en.wikipedia.org/wiki/Android\\_Auto](https://en.wikipedia.org/wiki/Android_Auto) jul 2017.
- [10] Diego Buffa, "Android & IVI systems",  
[http://www.windriver.com/seminars/7839-android/4\\_IVI\\_ANDROID.pdf](http://www.windriver.com/seminars/7839-android/4_IVI_ANDROID.pdf) jul 2017
- [11] <https://developer.android.com/design/auto/index.html> jul 2017
- [12] B. Kovacevic, M. Kovacevic, T. Maruna, D. Rasic, "Android4Auto: A proposal for integration of Android in vehicle infotainment systems", 2016 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 7-11 Januar, 2016.
- [13] A. Dakic, T. Srejic, M. Z. Bjelica, M. Krbanjevic, "Pilot In-Vehicle Infotainment Learning Platform", 5 – 7 Septembar 2016.