

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Tihana Uležić

**Mobilna aplikacija za upravljanje
videokonferenčnega sistema**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: viš. pred. dr. Alenka Kavčič

Ljubljana 2014

Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavlanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .



Št. naloge: 00484 / 2013
Datum: 9.4.2013

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **TIHANA ULEŽIČ**

Naslov: **MOBILNA APLIKACIJA ZA UPRAVLJANJE VIDEOKONFERENČNEGA SISTEMA**
MOBILE APPLICATION FOR CONTROLLING A VIDEOCONFERENCING SYSTEM

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

V okviru diplomske naloge izdelajte aplikacijo za mobilne naprave, s katero lahko upravljamo delovanje videokonferenčnega sistema. Aplikacija naj bo zasnovana tako, da deluje na različnih mobilnih platformah, predvsem na platformi Android in iOS. Aplikacija naj omogoča vzpostavitev brezžične omrežne povezave z mobilnim multimedijem stojalom za videokonferenčne sisteme ter preko njega krmiljenje mikrofona in kamere videokonferenčnega sistema. Delovanje aplikacije testirajte na mobilnem multimedijem stojalu, ki je preko serijskega vmesnika nadgrajen z modulom za brezžično internetno omrežje, kar omogoča povezavo stojala in mobilne naprave ter s tem izmenjavo ukazov. Na koncu ocenite delovanje razvite aplikacije ter podajte možnosti za nadaljnje izboljšave.

Mentor:

Alenka Kavčič
viš. pred. dr. Alenka Kavčič

Dekan:

Nikolaj Zimic
prof. dr. Nikolaj Zimic



IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisana Tihana Uležić, z vpisno številko **63070414**, sem avtor diplomskega dela z naslovom:

Mobilna aplikacija za upravljanje videokonferenčnega sistema

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom viš. pred. dr. Alenke Kavčič,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne 11. marec 2014

Podpis avtorja:

Iskrena hvala dragima mami in očetu za omogočanje študija. Hvala tudi bratu za vzpodbudo in moralno podporo. Zahvaljujem se za neomajno podporo, ljubezen in zaupanje vame. Hvala za vso podporo in finančno pomoč pri študiju.

Posebej bi se rad zahvalila svoji mentorici viš. pred. dr. Alenki Kavčič za čas, pomoč in nasvete pri izdelavi diplomskega dela.

Najlepša hvala tudi prijateljem za vso spodbudo pri nastajanju diplomske naloge (veste kdo ste).

Hvala tudi mojim sošolcem, ki so me podpirali skozi celoten študij in tudi vsem ostalim, ki so karkoli pripomogli k nastajanju diplomske naloge.

Moji družini.

Kazalo

Seznam uporabljenih kratic

Povzetek

Abstract

| | | |
|----------|---|-----------|
| 1 | Uvod | 1 |
| 2 | Tehnologija videokonferenčnih sistemov | 3 |
| 2.1 | Zgodovina videokonferenc | 3 |
| 2.2 | Razdelitev videokonferenčnih sistemov | 4 |
| 2.3 | Mobilno multimedijско stojalo | 8 |
| 3 | Uporabljene tehnologije | 13 |
| 3.1 | Razvojno okolje | 13 |
| 3.2 | Povezava stojala iMMS in aplikacije | 16 |
| 4 | Razvoj aplikacije | 23 |
| 4.1 | Zajem zahtev | 24 |
| 4.2 | Analiza zahtev in načrtovanje | 26 |
| 4.3 | Uporabniški vmesnik in implementacija | 27 |
| 4.4 | Povezava aplikacije na modul in pošiljanje ukazov | 32 |
| 4.5 | Testiranje pravilnosti delovanja aplikacije | 35 |
| 5 | Sklepne ugotovitve | 37 |

KAZALO

Slike

| | | |
|-----|---|----|
| 2.1 | Primer izgleda videokonferenčne sobe za intenzivno teleprezenco. | 6 |
| 2.2 | Primer videokonferenčnega sistema visoke ločljivosti. | 7 |
| 2.3 | Primer izgleda Polycom opreme standardne ločljivosti. | 8 |
| 2.4 | Primer izgleda osebnih videokonferenčnih sistemov. | 8 |
| 2.5 | Mobilno multimedijsko stojalo – iMMS. | 10 |
| 3.1 | Prikaz razvojnega okolja Appcelerator Titanium Studio. | 14 |
| 3.2 | RN-174 modul. | 16 |
| 3.3 | Prikaz uspešnega testiranja komunikacije z RN-174 modulom. | 18 |
| 3.4 | Prikaz uspešne povezave RN-171 modula na omrežje. | 22 |
| 4.1 | Home meni aplikacije. | 28 |
| 4.2 | Izgled menija konfiguracije. | 29 |
| 4.3 | Meni konfiguracije po končanem skeniranju, če smo našli mo- dul za povezavo. | 29 |
| 4.4 | Meni mikrofona. | 30 |
| 4.5 | Meni "Camera". | 30 |
| 4.6 | Meni "Camera 2". | 33 |
| 4.7 | Prikaz pozicij za predstavitev kamere v modulu "Camera 2". . . | 33 |
| 4.8 | Uspešna povezava na modul. | 34 |
| 4.9 | Prikaz testiranja uspešne povezave modula in aplikacije ter pošiljanje ukazov. | 36 |

Seznam uporabljenih kratic

ASCII – American Standard Code for Information Interchange (ameriški standardni nabor za izmenjavo informacij)

ISDN - Integrated Services Digital Network (integrirane storitve preko digitalnega omrežja)

IP - Internet Protocol

USB - Universal Serial Bus

HD – High Definition

SD – Standard Definition

LCD - Liquid-crystal display

DLP - Digital Light Processing

PEC - Periodic Error Correction (periodično popravljanje napak)

PTZ – Pan-tilt-zoom (premikanje, nagib, zumiranje)

CIF - Common Intermediate Format

HDX - High Definition Experience

iMMS – Mobile Multimedia Stand (Mobilno multimedjsko stojalo)

HDMI - High-Definition Multimedia Interface

TCP – Transmission Control Protocol

UDP - User Datagram Protocol

Povzetek

V diplomskem delu bomo spoznali videokonferenčne sisteme in osnovno razdelitev sistemov. Hkrati bomo spoznali multimedijsko videokonferenčno stojalo, predstavljen bo postopek izdelave aplikacije, ki bo omogočala upravljanje s stojalom ter uporabljene tehnologije, okolje in proces testiranja aplikacije. Glavni namen izdelave aplikacije je bilo olajšanje upravljanja z mobilnim multimedijskim stojalom. Glede na to, da nam mobilno multimedijsko stojalo ponuja možnost poljubne konfiguracije, smo se odločili nadgraditi stojalo z modulom, ki omogoča brezžično povezavo videokonferenčnega stojala in mobilnih naprav oziroma tabličnih računalnikov. Aplikacija tako omogoča povezavo z videokonferenčnim stojalom ter preko nje upravljanje s kamero in mikrofonom stojala.

Ključne besede: videokonferenčni sistemi, mobilna aplikacija, Appcelerator Titanium Studio, Android

Abstract

In this thesis, we will get to know the videoconferencing systems and will introduce the basic system division. At the same time we will present the multimedia videoconferencing stand and the process of developing the application for controlling the stand. We will also present the technologies used and the testing process. The main purpose of developing the application was to facilitate the controlling of mobile multimedia stand. Given that the stand offers the possibility of random configurations we decided to upgrade it with a module that allows wireless connection between the stand and mobile devices/tablet computers. Application also allows you to connect with videoconferencing stand and through it management of the camera and microphone.

Keywords: videoconferencing systems, mobile application, Appcelerator Titanium Studio, Android

Poglavje 1

Uvod

Videokonferenca je prenos slik (videa), govora (avdia) in podatkov med ločenimi lokacijami. Videokonferenčni sistemi postajajo čedalje pogostejši pripomoček v poslovnem svetu in njihova uporaba postaja stvarnost tudi v vsakodnevni rabi. Vizualna komunikacija uporabnikom omogoča predvsem boljšo uporabniško izkušnjo, kot bi jo dobili pri običajni telefonski konferenci. Poslovnim uporabnikom omogoča tudi izmenjavo dokumentov in možnost sodelovanja na daljavo, kjer lahko kot udeleženci videokonference pregledujejo predstavitev ali se izobražujejo neposredno iz svoje pisarne. Modernejši sistemi omogočajo tudi storitve oddaljenega dostopa in video zveze v visoki ločljivosti na običajnih osebnih računalnikih s povprečno širokopasovno povezavo.

Prvi videokonferenčni sistem je bil testiran leta 1968 [1], od takrat je skladno z razvojem telekomunikacij potekal tudi razvoj teh sistemov. Z razvojem interneta in boljših kodiranih algoritmov se je v zadnjih desetih letih močno razširila ponudba videokonferenčnih sistemov, ki obsega veliko možnosti: od profesionalnih sistemov za konferenčne sobe pa do brezplačnih programov na osebnih računalnikih.

Delovanje videokonferenčnih sistemov temelji na kompresiji video signala, zajetega s kamere, in prilagoditvi podatkovnega toka v obliko, ki je primerna za prenos po komunikacijskih sistemih. Danes temelji delovanje različnih

videokonferenčnih sistemov na internetnem omrežju, kjer poteka prenos videokonferenčnih vsebin po različnih prenosnih medijih in tehnologijah. Delovanje osnovnih sistemov je omogočeno že na povprečnih širokopasovnih povezavah, medtem ko profesionalni sistemi zahtevajo prenosne poti z večjo pasovno širino ter predvsem z mehanizmi za zagotavljanje kvalitete (kakovosti) storitev.

V diplomskem delu bomo tudi predstavili razvoj aplikacije za upravljanje z videokonferenčnim sistemom. Spoznali bomo tudi glavno tematiko diplomske naloge, ki je bila povezovanje videokonferenčnih sistemov s tabličnimi računalniki oziroma mobilnimi napravami in izdelavo aplikacije, ki bo omogočila upravljanje z videokonferenčnim sistemom.

Videokonferenčno stojalo smo nadgradili z RS232 RN-174 modulom, s pomočjo katerega je realizirana komunikacija med aplikacijo in stojalom. RN-174 je modul, ki ga lahko programiramo in nadzorujemo s preprostimi ukazi, ki so zapisani z ASCII znaki (standardni nabor za izmenjavo informacij). Po uspešni konfiguraciji lahko modul začne samodejno iskanje dostopne točke, na katero se lahko poveže preko Wi-Fi omrežja.

Daljinski upravljalniki so precej nevhvaležni, saj ima nadzor nad potekom konference lahko samo ena oseba. Z izdelavo aplikacije smo omogočili, da lahko več oseb (oziroma več uporabnikov) stojalu sporoči določene ukaze (npr. da se kamera videokonferenčnega stojala obrne na osebo, ki trenutno govori).

V naslednjem poglavju se bomo spoznali s tehnologijo videokonferenčnih sistemov in mobilnim multimedijem stojalom, za katerega je bila izdelana aplikacija. V tretjem poglavju bomo predstavili razvojno okolje in način povezave mobilne aplikacije in stojala. V četrtem poglavju bomo predstavili postopek razvoja aplikacije.

Poglavje 2

Tehnologija videokonferenčnih sistemov

2.1 Zgodovina videokonferenc

Videokonferenčni sistemi za povezovanje ljudi na različnih lokacijah uporabljajo avdio in video komunikacije. To je lahko zelo enostavno, na primer: pogovor med osebami v pisarnah (point-to-point), sistem pa je lahko bolj zapleten, na primer: vključuje več mest v velikih prostorih, na več lokacijah (multipoint).

Preprosto analogno videofonsko komunikacijo je bilo mogoče vzpostaviti že od izuma televizije dalje. Takšen predhodnik je bil običajno sestavljen iz dveh zaprtih televizijskih sistemov, priključen preko koaksialnega kabla ali radia. Med letoma 1936 in 1940 je preko koaksialnega kabla potekalo video telefonsko omrežje, ki so ga uporabljali v Berlinu in drugih večjih nemških mestih [1].

Šele leta 1980 je postal digitalni prenos preko telefonskih omrežij mogoč in je zagotavljal minimalno bitno hitrost (128 kilobitov/sekundo) za stisnjen video in avdio prenos (npr. ISDN oziroma integrirane storitve preko digitalnega omrežja). V tem času so tudi začeli raziskovati digitalne video in avdio komunikacije, začeli so se pojavljati prvi namenski sistemi.

Leta 1984 je podjetje Concept Communication iz Združenih držav Amerike nadomestilo takratne računalnike z novimi računalniki za telekonference z vezjem, ki podvoji frekvenco, pri kateri naprava izdeluje zaporedje sličic do 30 sličic na sekundo in se prilagaja standardom osebnih računalnikov [1].

Videokonferenčni sistemi so zelo hitro napredovali – v 1990-ih letih od zelo drage lastniške opreme do standarda, ki temelji na tehnologiji, ki je na voljo širši javnosti po razumni ceni. Leta 1990 so postale možne videokonference, ki temeljijo na IP protokolu [1].

V naslednjih letih je projekt Diane (Diversified Information and Assistance Network) [1] povečal izkoristek različnih videokonferenčnih platform za ustvarjanje multidržavnega sodelovanja javnih služb in za izobraževalno omrežje na daljavo. Sestavljeno je bilo iz šol, lokalnih središč, knjižnic, muzejev, živalskih vrtov in parkov, centrov javne pomoči in drugih skupnosti in organizacij.

Po letu 2000 je bila videotelefonija popularizirana s pomočjo brezplačnih internetnih storitev, kot sta Skype in iChat, spletnih vtičnikov in on-line telekomunikacijskih programov, ki so cenovno ugodni, čeprav so slabše kakovosti.

Tehnološki razvoj na področju videokonferenc se je od leta 2010 razširil od zmogljivih videokonferenčnih sistemov do uporabe ročnih mobilnih naprav, ki združujejo uporabo videa, avdia in deljenja namizja v realnem času preko varnih omrežij neodvisno od lokacije. Mobilne komunikacije sedaj omogočajo večjemu številu ljudi, ki se nahajajo na težko dostopnih lokacijah (kot so na primer naftne platforme), sposobnost sodelovanja in razpravljanja o tekočih temah s kolegi/sogovorniki, ki so oddaljeni tisoče kilometrov.

2.2 Razdelitev videokonferenčnih sistemov

Komponente klasičnega videokonferenčnega sistema lahko razdelimo na:

- videokonferenčno kodek enoto,
- kamero,

- mikrofoni,
- video zaslon,
- omrežno povezavo.

Kodek označuje "možgane in srce" videokonferenčnega sistema. Ta komponenta zajema video in avdio iz kamere in mikrofona ali avdio podsistema in ga stiska ter pošilja prek telekomunikacijskega omrežja. Komponenta tudi dekompresira dohodni video in avdio signal, tako da si ga je mogoče ogledati na zaslonu naprave.

Vrste kamere segajo od majhnih USB kamer na vrhu računalniških monitorjev do kamer visokih ločljivosti, ki imajo daljinski upravljavec, nagibe in zoom (PTZ oziroma "pan tilt zoom").

Mikrofoni, povezani z osebnim računalnikom, so večinoma majhni USB ali analogni mikrofoni. Večina videokonferenčnih sistemov prihaja z analognim mikrofonom, ki deluje z majhnimi skupinami ljudi. Pri velikih videokonferenčnih sestankih ima lahko vsak posameznik svoj mikrofoni.

Video zaslon je naprava, na kateri se prikazujejo posnetki, zajeti s kamero na drugi strani, ali za prikazovanje deljenega namizja. Lahko uporabimo plazmo, LCD zaslon ali LCD/DLP projektor. Namizni sistemi prikazujejo video na računalniškem monitorju.

Omrežna povezava je fizična povezava, ki prenaša podatke med sistemi, ki komunicirajo drug z drugim. Pasovna širina povezave in sposobnost dostopanja do sistema dosledno določa tako video učinkovitost kot kakovost poteka videokonferenčne seje.

Obstaja več različic videokonferenčnih sistemov. V grobem jih delimo na t. i. skupinske ali sobne sisteme in sisteme za osebno rabo (PC rešitve) [1,2,3].

Glede na kvaliteto slike delimo videokonferenčne sisteme na:

- videokonferenca za intenzivno teleprezenco (immersive telepresence),

- videokonferenca visoke ločljivosti,
- videokonferenca standardne ločljivosti in
- videokonferenca za osebno rabo.

Videokonferenca za intenzivno teleprezenco je danes najšodnejša oblika in izvedba videokonferenčne aplikacije na trgu. Govorimo o namenskem videokonferenčnem prostoru, ki je prilagojen tako, da na udeležence naredi izreden multimedijški vtis. Gre za idealno dimenzioniran prostor glede sobne akustike, razsvetljave in barvnih kombinacij notranje opreme (pohištvo, omizja, fotelji itd.). Videokonferenčne sobe so oblikovno tipizirane in zasnovane na ideji multimedijskega "prehoda" iz lokalnega prostora v virtualnega. Veliki projekcijski zasloni omogočajo prikaz več oddaljenih zaslonov različnih dimenzij in kombinacij. Primer izgleda videokonferenčne sobe je prikazan na sliki 2.1.



Slika 2.1: Primer izgleda videokonferenčne sobe za intenzivno teleprezenco.

Videokonferenčno opremo visoke ločljivosti (slika 2.2) predstavlja družina HDX (High Definition Experience) sistemov. Glede na velikost prostorov in tipično število udeležencev videokonferenčnih sestankov se sistemi

delijo na manj in bolj kompleksne. Razlike se nanašajo na tipe kamer, na funkcionalnost strojne opreme glede na število priključkov, podporo standardom 720p in 1080p ipd. Standard PEC (periodično popravljanje napak) rešuje omrežne komunikacijske probleme, kadar prihaja do izgub avdio in video paketov, ki si jih videokonferenčni sistemi različnih nivojev izmenjujejo. Z družino videokonferenčnih sistemov visokih ločljivosti lahko omogočimo bolj ali manj zahtevne sisteme, ki so uporabni v telemedicini, v izobraževanju na daljavo, pri izvajanju sodnih postopkov, pri poslovnih srečanjih delovnih sestankov, pri delu od doma itd.



Slika 2.2: Primer videokonferenčnega sistema visoke ločljivosti.

Videokonferenca standardne ločljivosti (slika 2.3) je v manj zahtevnih okoljih in za manj kompleksne aplikacije zelo uporabna serija videokonferenčnih izdelkov. Sistem podpira standard SD video, s katerim je videokonferenčna industrija začela resneje osvajati svetovno tržišče. Mobilne PTZ kamere v razmerju 4 : 3 znajo tudi avtomatsko najti govornika v prostoru.

Osebni videokonferenčni sistemi (slika 2.4) so člani družin HD (visoka ločljivost) in SD (standardna ločljivost) rešitev. Namenjeni so predvsem



Slika 2.3: Primer izgleda Polycom opreme standardne ločljivosti.

individualnemu delu (izjemoma dvema osebama). Sistemi imajo integrirane vse elemente multimedijske naprave: zaslon, kamero, ozvočenje, tipkovnica itd. Obstajajo tudi sistemi, ki so integrirani s programsko opremo na oseb- nem računalniku. V kombinaciji s takimi rešitvami uporabljamo HD USB kamero in USB slušalke z mikrofonom.



Slika 2.4: Primer izgleda osebnih videokonferenčnih sistemov.

2.3 Mobilno multimedijско stojalo

Mobilno multimedijško stojalo (znano tudi kot iMMS) je izdelek podjetja DEK Electronic d. o. o.. iMMS je videokonferenčno stojalo, ki vsebuje vse komponente klasičnega konferenčnega sistema z možnostjo nadgraditve s poljubnimi elementi.

Mobilno multimedijško stojalo ima tri nivoje opremljenosti in se lahko uporabi v več kombinacijah:

- osnovna verzija,
- razširjena verzija in
- prestižna verzija.

Stojalo je uporabno pri videokonferenčnih sejah in glede na to da ponuja možnost nadgradnje s poljubnimi elementi, smo to izkoristili pri opremljenosti stojala.

Osnovna verzija stojala je opremljena z ozvočenjem. Vsebuje stikalo za vklop/izklop in dva priključka za priklop dodatnih zunanjih avdio/video naprav.

Razširjena verzija stojala je opremljena s kvalitetnim ozvočenjem. Vgrajena je tudi elektronika za ozvočenje in mešanje avdio signalov. Z zadnje desne strani je dostopno stikalo za vklop/izklop in dva napetostna priključka za priklop dodatnih zunanjih avdio/video naprav; na levi strani se nahaja priklopna vtičnica za avdio/video naprave.

Prestižna verzija stojala (slika 2.5) je opremljena z vsemi lastnostmi, ki jih imata osnovna in razširjena verzija, in z možnostjo upravljanja preko tabličnih računalnikov (npr. Android platforma). Dodatna vgrajena elektronika omogoča distribucijo HDMI signalov na način, ki ponuja več konfiguracij zunanjih avdio/video naprav.

2.3.1 Strojna in programska oprema stojala

Strojna oprema prestižne verzije stojala vsebuje dodatni modul za CPE. Modul ima 3 serijske kanale, digitalne vhodne ter izhodne signale. Nameščen je na nosilnem tiskanem vezju in omogoča še določene razširitve in prilagoditve funkcionalnim potrebam stojala.

V grobem lahko navedemo naslednje karakteristike strojne opreme:



Slika 2.5: Mobilno multimedijско stojalo – iMMS.

- **Prvi serijski kanal RS232-1** je namenjen izključno komunikaciji z RN-174 modulom. Ta kanal bo imel časovno prioriteto za dvosmerno komuniciranje med strojno opremo stojala in tabličnim računalnikom.
- **Drugi serijski kanal RS232-2** je rezerviran za multipliciranje¹ serijskih kanalov na ustrezno število. Te serijske kanale se uporabljajo za nadzor avdio/video opreme stojala ali zunanjih avdio/video naprav (Codec, zasloni, projektorji itd.). Izvedba strojne opreme razširitvenega vezja podpira tako standardne dvosmerne serijske kanale kot enosmerne za potrebe oddajnih diod.

¹Postopek razdeljevanja prenosnega kanala na več kanalov, po katerih potujejo podatki iz različnih virov, tako da ima vsak izvor svoj lastni kanal. Drugače povedano, lahko uporabimo eno samo komunikacijsko zvezo za prenos večjega števila signalov istočasno ali v hitrem zaporedju.

- **Tretji serijski kanal RS232-3** se uporabi za komunikacijska sporočila med tabličnim računalnikom in stojalom. To je neke vrste testni serijski kanal, ki se uporablja za sledenje izvajanja programa (tracer) ter za pomoč pri odkrivanju napak v zgodnji fazi razvoja programske opreme. Na ta kanal priključimo npr. notesnik s primerno programsko opremo (hyper terminal, RS232 debugger ...).
- Digitalne izhode (16 x) se lahko uporabi:
 - 8 x za relejske izhode za opsijsko priključevanje zunanjih naprav preko sponk na panelni plošči zadaj (npr dvig in spust projekcijskega platna itd.),
 - 8 x rezerva.
- Digitalne vhode (24 x).

Izdelali smo dve ločeni programske opremi: ena se izvaja kot aplikacija na tablici z operacijskim sistemom Android in druga, ki se izvaja na RN-174 modulu. Aplikacija na tablici omogoča uporabniku enostavno upravljanje z iMMS-jem. Aplikacija na modulu se ukvarja s konkretnimi nalogami svoje strojne opreme in z napravami, ki se priključijo na stojalo iMMS.

Osnovne naloge, ki jih obe programske opremi izvajata, so naslednje :

- En program se izvaja na WiFi RN-174 modulu, in sicer preko RS232 vrat vpiše ustrezne IP naslove in definira množico potrebnih parametrov. Ko zaključi začetno nastavljanje, preklopi modul RN-174 v uporabniški način delovanja, v katerem ostane ves čas praktične uporabe aplikacije. WiFi kanal je odprt in pripravljen za dvosmerno komunikacijo. Večina ukazov, ki jih pošilja aplikacija stojalu, je enosmernih in imajo lahko dodatne parametre. Nekateri ukazi zahtevajo odgovore oz. podatke od stojala.
- Drug program se izvaja na mobilni napravi (tabličnem računalniku ali pametnem telefonu) in omogoča upravljanje z modulom, ki se nahaja na videokonferenčnem stojalu.

Poglavje 3

Uporabljene tehnologije

Za razvoj aplikacije in omogočanje povezave z iMMS so uporabljene različne tehnologije. V nadaljevanju so na kratko predstavljena orodja in tehnologije, ki smo jih uporabili pri razvoju mobilne aplikacije.

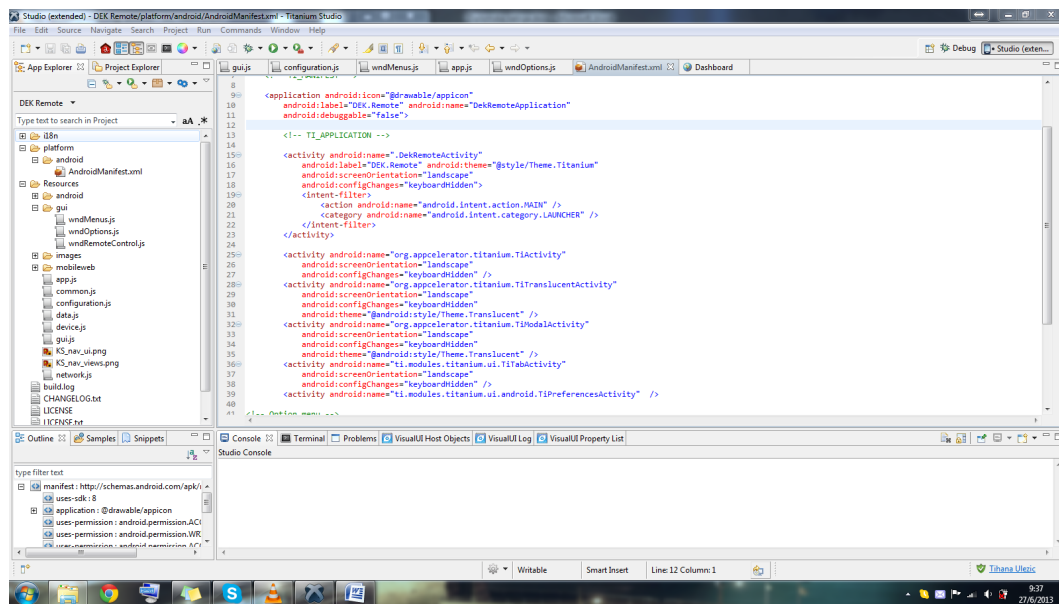
3.1 Razvojno okolje

Na začetku je bilo načrtovano, da bi aplikacijo za Android naprave izdelali v orodju Eclipse z uporabo programskega jezika Java in aplikacijo za iPhone/iPad v orodju DragonFireSDK z uporabo programskega jezika C++. Glede na to, da bi morali pisati dve različni programski kodi za izdelavo iste aplikacije, smo se odločili olajšati delo z uporabo enega razvojnega okolja, ki nam bo omogočil uporabo iste kode na različnih platformah.

Odločili smo se za uporabo programskega okolja Appcelerator Titanium Studio, ki nam omogoča uporabo iste kode na več različnih mobilnih platformah, kod so Android, iOS, BlackBerry ... [4]

Aplikacija je bila izdelana v programskem okolju Appcelerator Titanium Studio (slika 6). Appcelerator Titanium je platforma za razvoj mobilnih, tabličnih in namiznih aplikacij z uporabo spletnih tehnologij. Appcelerator Titanium je razvilo podjetje Appcelerator Inc. in je bil uveden decembra 2008. Podpora za razvoj Android in iPhone aplikacij je bila dodana v juniju

leta 2009, za iPad v aprilu 2010. Program uporablja sintakso JavaScripta in Titanium API, ki je precej drugačen od znanih spletnih okvirjev, kot je npr. jQuery. Lahko jo uporabljamo na operacijskem sistemu Windows 7 (in novejših) ter na operacijskih sistemih Linux in Mac [5].



Slika 3.1: Prikaz razvojnega okolja Appcelerator Titanium Studio.

Appcelerator Titanium Mobile je ena od številnih mobilnih spletno usmerjenih aplikativnih rešitev, ki omogočajo ustvarjanje aplikacij za Android in iPhone ter spletnih in hibridnih aplikacij.

V nadaljevanju lahko preberemo definicije [6] posameznih vrst aplikacije, ki jih lahko razvijemo z uporabo orodja Titanium:

- **Izvirne aplikacije** so razvite za določen operacijski sistem z uporabo ustreznega SDK, z uporabo orodja in jezika, ki jih navadno zagotovi prodajalec naprav (npr. Xcode / Objective-C za iOS, Eclipse / Java za Android, Visual Studio / C# za Windows Phone).
- **Mobilne spletne aplikacije** so strežniške aplikacije, zgrajene z uporabo katere koli strežniške tehnologije (PHP, Node.js, ASP.NET).

- **Hibridne aplikacije** so avtohtone aplikacije, ki se lahko izvajajo kot izvorne aplikacije na določeni napravi in so zapisane z uporabo spletnih tehnologij (HTML5, CSS, JavaScript).

Na začetku razvoja aplikacije smo se odločili, da bomo izdelali hibridno aplikacijo. Hibridne aplikacije so ena od boljših izbir za razvoj aplikacije, ki lahko deluje na različnih operacijskih sistemih, glede na to, da lahko uporabimo prednosti tako izvornih kot spletnih aplikacij. Prednost hibridnih aplikacij je v tem, da lahko napišemo in vzdržujemo eno kodo in jo uporabljamo na več različnih napravah. Kodo aplikacije lahko napišemo z uporabo spletnih tehnologij in jo objavimo v trgovini z aplikacijami.

Temeljne značilnosti Appcelerator Titanium:

- Podpora za spletne tehnologije, ki temeljijo na standardih HTML, CSS in JavaScript, na vseh platformah skupaj s PHP, Python in Ruby za namizne platforme.
- Vgrajena podpora za JavaScript in Ajax, vključno z jQuery, YUI, Mootools, Scriptaculous itd.
- Od operacijskega sistema neodvisen API za dostop do izvornih komponent uporabniškega vmesnika, vključno z navigacijskimi tipkami, meniji, pogovornimi okni in opozoril ter funkcionalnosti naprave, vključno z datotečnim sistemom, zvokom, omrežjem in lokalno bazo podatkov.
- API dostop do izvornih mobilnih funkcionalnosti, kot so geografska lokacija, merilnik pospeškov in zemljevidi.

Ena od glavnih prednosti izbranega razvojnega okolja je možnost izdelave aplikacij, ki lahko delujejo na več različnih platformah z izdelavo ene kode. Razvojno okolje je brezplačno. Slabost izbranega okolja je bila v tem, da bi morali uporabljati operacijski sistem iOS, če bi želeli izdelati aplikacijo, ki bo delovala na iOS napravah. To so omejitve s strani podjetja Apple. Potrebno je imeti licenco podjetja Apple, da bi lahko namestili aplikacijo na njihovo mobilno napravo.

3.2 Povezava stojala iMMS in aplikacije

Da bi omogočili komunikacijo med videokonferenčnim sistemom in mobilnimi napravami, je bilo potrebno nadgraditi konfiguracijo stojala iMMS z modulom RS232, ki bo omogočal komunikacijo med njimi. Za povezavo smo uporabili RN-174 WiFly Super Modul (slika 3.2). RN-174 je modul, ki omogoča dvosmerno komunikacijo in ga lahko programiramo in nadzorujemo s preprostimi ASCII ukazi.

Modul lahko konfiguriramo s poljubnimi ukazi, ki nam omogočajo upravljanje. Ko je nastavljanje modula zaključeno, lahko začne modul samodejno iskati dostopne točke, na katere se lahko poveže. Lahko se avtenticira in poveže preko katerega koli Wi-Fi omrežja. Poleg tega lahko modul samodejno pošlje podatke oddaljenemu gostitelju in ko je prenos podatkov dokončan, se vrne nazaj v status mirovanja.



Slika 3.2: RN-174 modul.

Program, ki smo ga napisali za krmiljenje modula, je shranjen v pomnilniku ROM modula. Modul RN-174 ima naslednje značilnosti:

- 128 KB RAM in 2 MB ROM spomina, 2 KB baterijsko podprtega pomnilnika, 8 megabitov Flash pomnilnika.

- Inteligentno vgrajeno upravljanje napajanja s programabilnim bujenjem.
- Ura realnega časa za časovno označevanje, samodejno mirovanje in samodejno zbujanje.
- Varna WiFi avtentikacija WEP-128, WPA-PSK (TKIP), WPA2-PSK (AES).
- Vgrajene mrežne aplikacije: DHCP, DNS odjemalec, ARP, ICMP ping, FTP, TELNET, HTTP, UDP, TCP.
- IEEE 802.11 varčevanje z energijo in funkcije gostovanja.

3.2.1 Konfiguracija

WiFly modul ima dva načina, in sicer podatkovni način in način upravljanja. V podatkovnem načinu modul je pripravljen tako, da sprejme dohodne povezave ali začne odhodne povezave. Če želimo nastaviti parametre in/ali si ogledati trenutno konfiguracijo modul mora biti v načinu upravljanja (ime-novan tudi način konfiguracije). V danem trenutku se modul lahko nahaja bodisi v načinu upravljanja ali v podatkovnem načinu. Po vklopu je vedno v podatkovnem načinu [7].

Vstop v način upravljanja (command mode)

Za vstop v način upravljanja, je napravi potrebno poslati ukaz \$\$\$ z uporabo terminala ali preko brezžične povezave pri uporabi aplikacije. Naprava se bo odzvala z odgovorom CMD, kar pomeni, da je naprava v načinu upravljanja (oz. v ukaznem načinu). Veljavni ukazi vrnejo odgovor AOK, neveljavni vrnejo ERR. Za izhod iz načina upravljanja je potrebno vnesti ukaz **exit**.

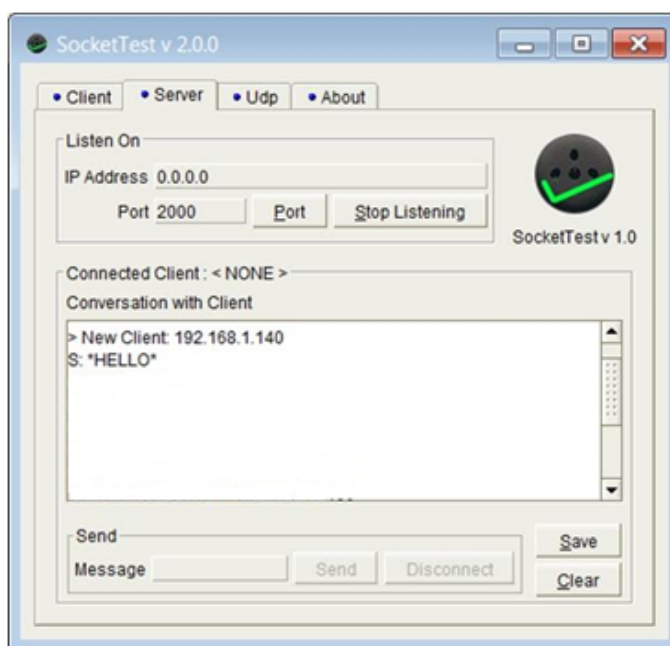
Oddaljena konfiguracija modula

Za konfiguracijo modula WiFly smo uporabili Ad Hoc način konfiguracije. V tem načinu modul ustvari svoje brezžično omrežje na zahtevo, na katerega

se lahko povežemo z računalnikom ali mobilno napravo. Omrežje ni vezano na že obstoječo infrastrukturo, kot so usmerjevalniki v ožičenih omrežjih ali dostopne točke brezžičnih omrežij.

Uporaba Ad Hoc načina za konfiguracijo naprave odpravlja potrebo po povezavi modula z dostopno točko omrežja. Z računalnika se lahko povežemo na omrežje WiFly-GSX-XX. To je odprto omrežje, ki ne zahteva gesla. Trenutno WiFly podpira le odprt način za ustvarjanje Ad Hoc omrežij. Ko je modul povezan in dobi IP naslov, je WiFly povezan na vratih 2000.

Če je povezava med modulom in aplikacijo bila uspešna, bi morali videti odziv *HELLO*. Po uspešni povezavi, lahko vstopimo v ukazni način z vnosom zaporedja \$\$\$\$. Prikaz uspešne povezave si lahko ogledamo na sliki 3.3.



Slika 3.3: Prikaz uspešnega testiranja komunikacije z RN-174 modulom.

3.2.2 Organizacija ukazov

Ob vklopu je modul v podatkovnem načinu. Vsi nastavitveni podatki se po vklopu naložijo v RAM iz datoteke "config". Za preklon modula v način konfiguracije je potrebno poslati ukaz \$\$\$.

Ukaze lahko razdelimo v pet kategorij:

- **SET COMMANDS** (ukazi za nastavitev parametrov),
- **GET COMMANDS** (ukazi za pridobitev nastavljenih vrednosti parametrov),
- **STATUS COMMANDS** (statusni ukazi),
- **ACTION COMMANDS** (ukazi za izvedbo določene akcije),
- **FILE IO COMMANDS** (ukazi za upravljanje s datotekami).

V nadaljevanju bomo pojasnili kategorije ukazov in prikazali nekatere od uporabljenih ukazov za konfiguracijo modula.

SET ukazi lahko spremenijo samo podatke, zapisane v RAM-u in jih uporabljamo za konfiguracijo modula. To so stalni ukazi in začnejo takoj veljati. Po shranitvi v konfiguracijsko datoteko so ukazi na voljo.

set option deviceid <string>

Ukaz smo uporabili za spremembo imena modula.

set wlan phrase <string>

Ukaz, ki ga uporabljamo za nastavitev gesla za povezavo na WPA omrežja. Če se povežemo na zaščiteno omrežje, je ta ukaz potrebno izvesti, preden se povežemo – to je eden od pogojev za uspešno povezavo.

GET ukazi prikazujejo trajno shranjene informacije na zaslonu.

get everything

Prikazuje vse nastavitve modula.

get option

Prikaže nastavitve opcij, kot je na primer ID naprave.

STATUS ukazi prikažejo, kaj se trenutno dogaja z vmesnikom, IP statusom itd.

show net <n>

Prikazuje trenutno stanje omrežja, avtentikacijo itd.

ACTION ukazi izvedejo ukrepe, kot so skeniranje, preklopitve stikala itd.

\$\$\$

Preklopi modul v način konfiguracije.

exit

Izhod iz načina konfiguracije.

close

Prekinitev TCP povezave.

join <ssid>

Povezava na ssid omrežje. Če je omrežje zaščiteno, moramo nastaviti geslo z ukazom "set wlan", preden se lahko povežemo.

reboot

Prisilni ponovni zagon modula.

FILE IO ukazi se uporabljajo za nadgradnjo, naložitev in shranjevanje konfiguracijskih nastavitev. Lahko tudi brišemo ali shranjujemo nove datoteke.

ls

Prikaže vse datoteke, shranjene v sistemu

load <name>

Prebere nastavitveno datoteko s podanim imenom.

save

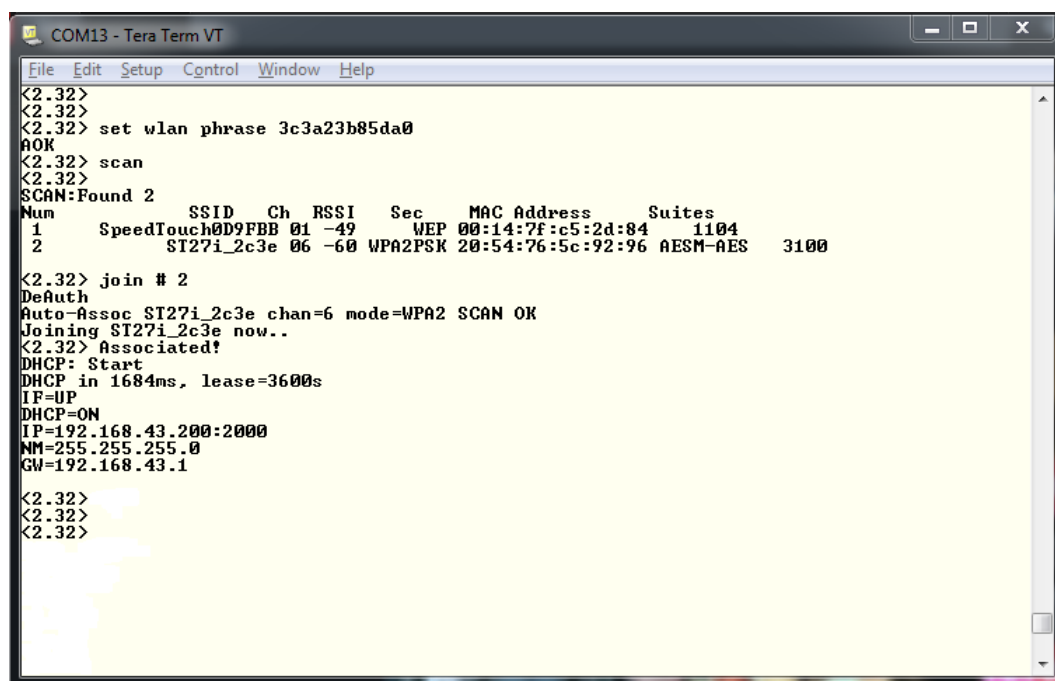
Shrani konfiguracijske nastavitve v datoteko "config" (privzeta datoteka).

save <name>

Shrani konfiguracijske nastavitve v datoteko s podanim imenom.

Konfiguracijske ukaze lahko pošiljamo modulu preko terminala. V ta namen smo uporabili odprtokodno programsko opremo Tera Term, ki deluje kot emulator terminala. Virtualni terminal podpira telnet, SSH 1 & 2 in serijske povezave. Izgled okolja Tera Term in prikaz pošiljanja ukazov za povezavo na omrežje si lahko ogledamo na sliki 3.4.

Vse spremembe je potrebno shraniti, sicer bo modul uporabil nastavitve, shranjene pred ponastavitvijo oziroma vklopom. Nastavitve lahko shranimo z uporabo ukaza **save**.



```
COM13 - Tera Term VT
File Edit Setup Control Window Help
<2.32>
<2.32>
<2.32> set wlan phrase 3c3a23b85da0
AOK
<2.32> scan
<2.32>
SCAN:Found 2
Num      SSID      Ch  RSSI   Sec   MAC Address      Suites
1        SpeedTouch0D9FBB 01 -49   WEP 00:14:7f:c5:2d:84 1104
2        ST27i_2c3e 06 -60   WPA2PSK 20:54:76:5c:92:96 AESM-AES 3100
<2.32> join # 2
DeAuth
Auto-Assoc ST27i_2c3e chan=6 mode=WPA2 SCAN OK
Joining ST27i_2c3e now..
<2.32> Associated!
DHCP: Start
DHCP in 1684ms, lease=3600s
IF=IP
DHCP=ON
IP=192.168.43.200:2000
NM=255.255.255.0
GW=192.168.43.1
<2.32>
<2.32>
<2.32>
```

Slika 3.4: Prikaz uspešne povezave RN-171 modula na omrežje.

Za povezavo smo uporabili ukaz "join" in glede na to, da smo se povezovali na zaščiteno omrežje, smo morali z uporabo ukaza set wlan najprej zapisati geslo omrežja. Po uspešni povezavi modula na omrežje dobili smo odgovor "Associated!" in IP naslov modula.

Poglavje 4

Razvoj aplikacije

”DEK.Remote” je aplikacija za mobilne naprave z operacijskim sistemom Android in iOS. Glavni namen izdelave aplikacije je bil poenostavitev upravljanja z iMMS. Pri zdajšnjem načinu uporabe se za upravljanje z videokonferenčnim sistemom uporabljalo veliko daljinskih upravljalcev in je samo ena oseba lahko upravljala s sistemom. Z izdelavo aplikacije smo želeli poenostavi upravljanje.

Torej: glavni namen aplikacije je upravljanje z videokonferenčnim stojalom in deluje kot daljinski upravljalnik iMMS-ja. Z razvojem mobilne aplikacije smo poskušali nadomestiti in predvsem čim bolj poenostaviti dosednji postopek upravljanja s stojalom. Razvoj aplikacije je zahteval izvedbo v dveh delih. Prvi del je način, s katerim se aplikacija povezuje s stojalom. Drugi del je pošiljanje ukazov stojalu. Za popolno delovanje aplikacije potrebujemo povezavo v internet, ki nam omogoča pošiljanje podatkov iz aplikacije proti stojalu. Ko je komunikacija med aplikacijo in stojalom vzpostavljena, lahko upravljamo s komponentami stojala.

Razvoj aplikacije smo razdelili na naslednje faze:

- zajem zahtev,
- analiza zahtev in načrtovanje,
- implementacija aplikacije in

- testiranje delovanja aplikacije.

4.1 Zajem zahtev

V začetni fazi planiranja izdelave aplikacije smo imeli veliko zahtev, ki niso bile od začetka podrobno tehnično izražene in ni bilo znano, kako naj bi najenostavneje implementirali vse potrebne funkcionalnosti in kaj naj bi bilo prikazano v posameznih menijih. Na začetku so bile zahtevane naslednje možnosti upravljanja s stojalom:

- Vklp/izklop sistema.
- Pregled imenika iMMS-ija in možnost vzpostavitve klica.
- Nadzor videokonferenčne sobe:
 - Kamera:
 - * Pregled cele sobe.
 - * Premik kamere na določeno pozicijo.
 - * Premik kamere levo, desno, gor ali dol od trenutne pozicije.
 - Mikrofon:
 - * Mute/unmute – izklop/vklop zvoka.
 - * Nastavitev glasnosti.
 - Projektor/zaslon stojala:
 - * Power on/off – vklop/izklop.
- Možnost izbora, kateri meniji naj se prikažejo v home meniju.

Glede na to, da je bila zahtevnost aplikacije kompleksna, smo se odločili razdeliti izdelavo aplikacije na več inkrementov in v prvi verziji aplikacije implementirati samo module, za katere so bile vse funkcionalnosti točno določene.

Pri modulu za pregled imenika smo se srečali s problemom dvosmerne komunikacije med stojalom in aplikacijo in potencialno slabo odzivnostjo glede na veliko količino podatkov, shranjenih v imeniku; zaradi tega smo se odločili modul pregleda imenika implementirati v eni od naslednjih verziji aplikacije.

Odločili smo se, da bomo v prvo verzijo aplikacije vključili možnost upravljanja s kamero in mikrofonom videokonferenčnega stojala. Hkrati smo se tudi odločili, da bomo za povezovanje aplikacije in iMMS-ja uporabili RS232 modul (RN-174). V modul za upravljanje z mikrofonom smo dodali možnost vklopa in izklopa mikrofona. Izdelana sta dva modula, ki omogočata postavljanje kamere videokonferenčnega sistema.

Hkrati so bile določene tudi naslednje zahteve, ki smo jih vključili v aplikacijo:

- Jezik:
 - Gumbi/meniji in ostali deli aplikacije so napisani v angleščini.
- Možnost nadgradnje:
 - Aplikacijo moramo zastaviti modularno, tako da bo dodajanje novih funkcionalnosti možno brez večjih popravil izvorne kode.
- Uporabniški vmesnik:
 - Uporabniški vmesnik moramo zastaviti čim bolj enostavno in pregledno. Tukaj se moramo držati načela več je manj.
- Aplikacije se lahko izvaja na Android in iOS napravah.
- Komunikacija med aplikacijo in mobilnim multimedijem stojalom je dvosmerna.
- Aplikacija bo preko RS232 sprejemala ukaze od Android oziroma iOS naprave in jih lokalno procesirala. V osnovi gre za API ukaze, ki jih znajo razumeti avdio/video naprave (ploščati zasloni, DVD, videokonferenčni sistemi itd.).

4.2 Analiza zahtev in načrtovanje

V fazi zajema in specifikaciji zahtev smo določili osnovne funkcionalne ter tehnološke omejitve, potrebne za pravilno delovanje aplikacije. Hkrati smo se odločili, katere funkcionalnosti bomo implementirali v prvi verziji aplikacije. Na podlago definiranih zahtev smo si zastavili tudi glavne cilje pri razvoju aplikacije in naredili okvirni načrt izdelave.

V prvi fazi smo se odločili, da bomo implementirali modul za povezavo aplikacije in iMMS-ja, modul za upravljanje z mikrofonom in kamero. V naslednjih fazah razvoja je načrtovana implementacija funkcije imenika, predstavitev ter ostalih funkcij, ki naj bi olajšale uporabo videokonferenčnega sistema.

V tej fazi smo se tudi odločili, da bomo za razvoj aplikacije uporabili Appcelerator Titanium Studio razvojno okolje, ki omogoča razvoj aplikacij z uporabo JavaScripta in Titanium API sintakse. Izdelali smo opis poslovnega okolja, na katerega se nanaša razvoj, izdelali smo načrt programskih modulov in načrt testiranja.

V opisu poslovnega okolja smo določili, da bo prvo verzijo aplikacije možno uporabljati samo v videokonferenčni sobi podjetja, za katero je aplikacija izdelana. Vse položaje kamere bomo vnaprej definirali v kodi aplikacije. Hkrati je tudi določeno, na kateri poziciji se mora nahajati stojalo, če želimo, da aplikacija optimalno deluje. V naslednjih verzijah aplikacije bo implementirana možnost poljubnega pozicioniranja in bomo lahko uporabljali aplikacijo v kateri koli videokonferenčni sobi, kjer imamo na voljo iMMS.

V načrtu testiranja smo določili potek testiranja. Odločili smo se razdeliti testiranje na dve glavni fazi, testiranje komunikacije oziroma povezave med aplikacijo in stojalom ter testiranje delovanja posameznih modulov aplikacije.

Načrt programskih modulov opisuje module aplikacije v podrobnosti in osnovne funkcionalnosti uporabniškega vmesnika. Uporabniški vmesnik smo morali zastaviti čim bolj enostavno in pregledno, da bi povečali uporabnost aplikacije med videokonferenčno sejo. Tukaj smo se držali načela več je manj.

V ozadje aplikacije smo postavili sliko stojala in konferenčne sobe, v kateri bomo uporabljali aplikacijo. Gumb za vstop v modul konfiguracije bo dostopen iz vseh modulov aplikacije in ga bomo postavili v zgornji desni kot aplikacije. Gumb za dostop do ostalih modulov bomo postavili v eno vrsto in jih bomo prikazali na sredini zaslona.

4.3 Uporabniški vmesnik in implementacija

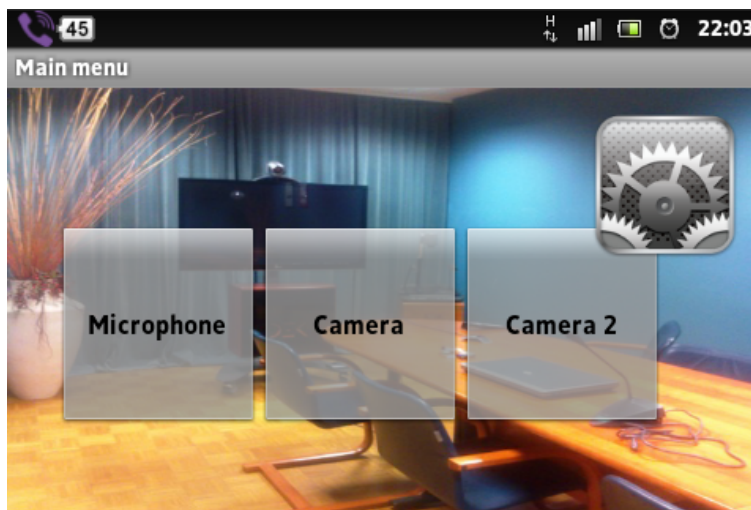
Razvoj mobilne aplikacije smo razdelili na manjše podsklope. Vsak sklop je neodvisen, vendar je pa še vedno združljiv z drugimi sklopi. V tej fazi smo se ukvarjali s kreiranjem izgleda uporabniškega vmesnika, kodiranjem aplikacije in vključevanjem vse načrtovane funkcionalnosti.

Uporabniški vmesnik smo razdelili na štiri menije: glavni oz. home meni, dva menija za upravljanje s kamero, meni za upravljanje z mikrofonom in meni konfiguracije. Za dva menija za upravljanje s kamero smo se odločili zato, da so pozicije točno določene; v prvem meniju imamo možnost pozicioniranja kamere na levo oziroma desno stran konferenčne sobe, v drugem pa lahko na točno določen stol v konferenčni sobi. Izgled home menija si lahko ogledamo na sliki 4.1.

Glavna datoteka naše aplikacije se imenuje *app.js*. V tisti datoteki je definirana inicializacija aplikacije in osnovni parametri aplikacije (platforma, na kateri se aplikacija izvaja, velikost zaslona itd.). Povezana je tudi z ostalimi datotekami aplikacije.

Konfiguracijo menijev in podmenijev aplikacije smo se odločili definirati v posebni datoteki. V isti datoteki smo določili tudi izgled glavnega menija, velikost gumbov in dodane komponente, ki naj se prikažejo ob zagonu aplikacije (npr. gumb za upravljanje z mikrofonom, kamero itd.). V datoteki so določeni podmeniji aplikacije, z izjemo menija nastavitvev, ki smo ga definirali v posebni datoteki, ker smo želeli imeti dostop do menija iz vseh modulov aplikacije.

Vstopna stran je oblikovana tako, da že ob prvem pogledu lahko vidimo,



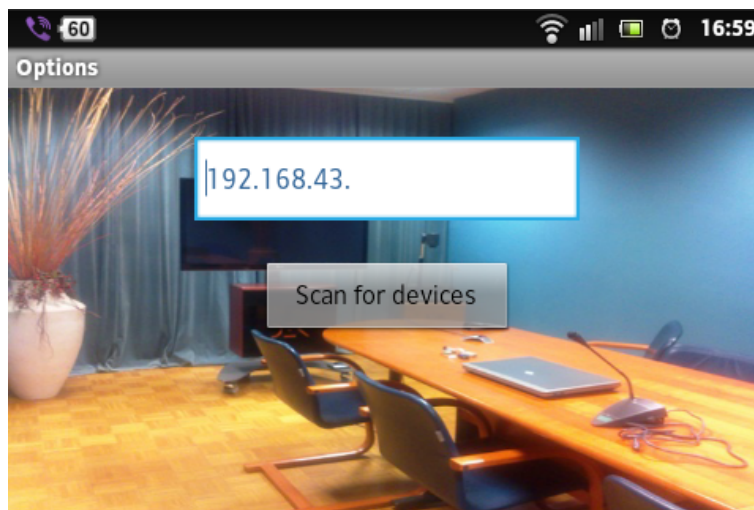
Slika 4.1: Home meni aplikacije.

katere možnosti upravljanja nam ponuja aplikacija. Hkrati je vstopna stran tudi prvi meni aplikacije. Na meniju so prikazani štiri gumbi. V zgornjem desnem kotu aplikacije je gumb za vstop v meni konfiguracije. Na sredini zaslona so gumbi za prehod v meni za upravljanje z mikrofonom in kamero. Iz vstopne strani aplikacije lahko vstopimo v kateri koli drugi modul.

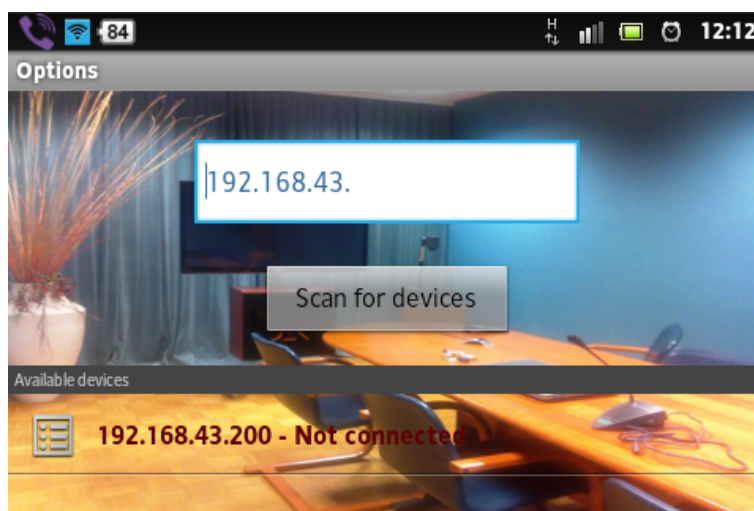
V meniju konfiguracije se nahaja vnosno polje, kjer lahko vpišemo poljubni IP naslov. Lahko tudi vpišemo točen IP naslov našega modula in ga poiščemo s pritiskom na gumb. Izgled menija konfiguracije je prikazan na sliki 4.2; na sliki 4.3 je prikazan izgled menija po končani fazi skeniranja, če smo uspešno našli modul za povezavo. Meni uporabljamo za iskanje in povezovanje z modulom. Možni so trije scenariji:

1. S skeniranjem nismo našli modula za povezavo.
2. Našli smo iskani modul, povezava ni bila uspešna.
3. Našli smo iskani modul in smo se uspešno povezali.

Ob vstopu v meni mikrofona se nam prikažejo trije gumbi. Prva dva gumba nam omogočata upravljanje z videokonferenčnim sistemom, tretji



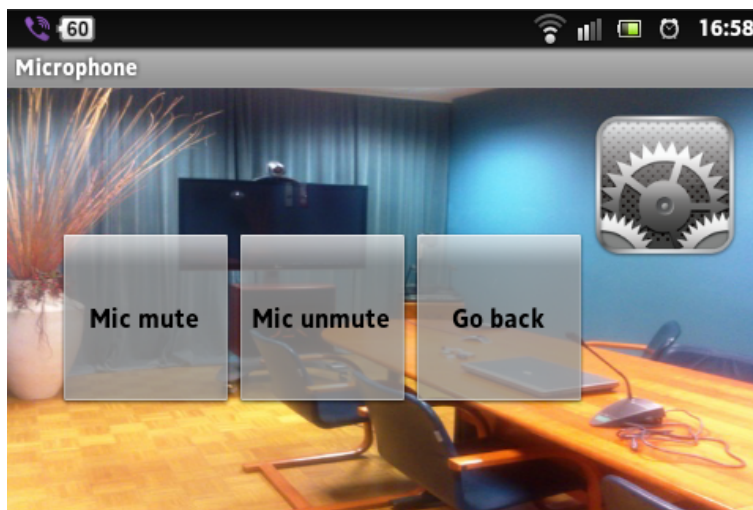
Slika 4.2: Izgled menija konfiguracije.



Slika 4.3: Meni konfiguracije po končanem skeniranju, če smo našli modul za povezavo.

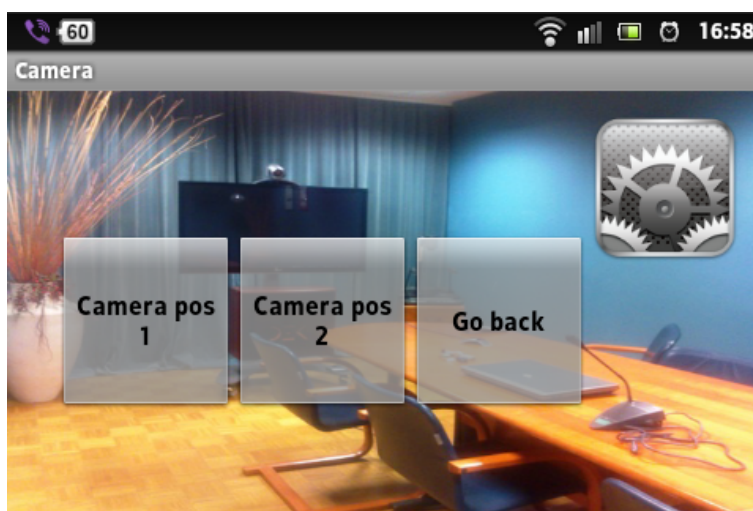
gumb je uporabljen za vrnitev v glavni meni aplikacije. Z uporabo tega menija lahko izklopimo ali vklopimo mikrofonski videokonferenčni stojal. Izgled menija si lahko ogledamo na sliki 4.4

Aplikacija ponuja dva menija za upravljanje s kamero videokonferenčnega sistema. V obeh menijih je pozicioniranje kamere fiksno določeno. V prvem



Slika 4.4: Meni mikrofona.

meniju kamere (prikazan na sliki 4.5) lahko kamero zamaknemo od trenutne pozicije levo (pozicija 1) ali desno (pozicija 2).



Slika 4.5: Meni "Camera".

Pozicije, na katere naj bi se postavila kamera stojala ob pritisku na gumb, so točno določene in že zakodirane v kodi aplikacije.

```
title: "Camera",
  commands: [
    {
      title: "Camera pos 1",
      command: "camera near setposition 20 40 60",
      width: 7
    },
    {
      title: "Camera pos 2",
      command: "camera near setposition 120 140 160",
      width: 7
    },
    {
      title: "Go back",
      link: "Main menu",
      width: 7
    },
  ],
]
```

Ob pritisku na določeni gumb se pokliče funkcija "OnButtonClicked". Če konfiguracija gumba vsebuje spremenljivko "command", pokličemo funkcijo "SendTextCommandToAllClients", ki pošlje definirani ukaz stojalu; če vsebuje spremenljivko "link", se ob kliku prikaže modul, ki je definiran v konfiguraciji spremenljivke.

```
OnButtonClicked : function(button) {
  if(button.command) {
    app.data.SendTextCommandToAllClients(button.command);
  }

  if(button.link) {
    app.gui.wndMenus.SetActiveMenu(button.link);
  }
}
```

```
}
```

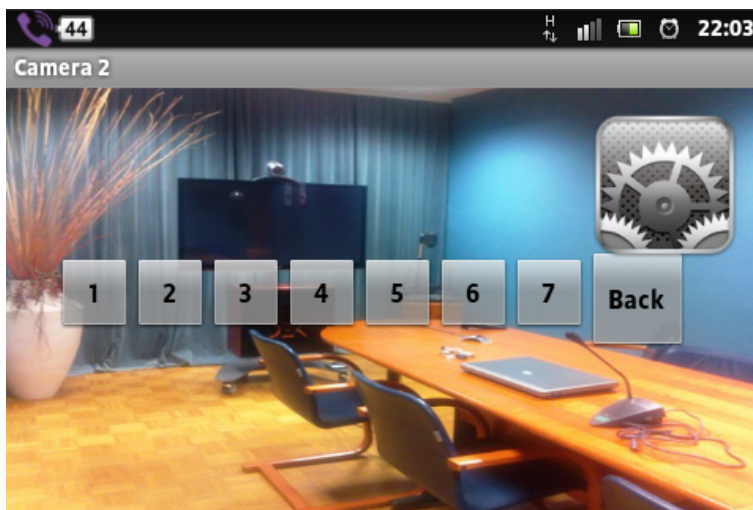
Funkcija "SendTextCommandToAllClients" poišče vse module, na katere smo trenutno povezani, in pošlje vrednost, ki smo jo definirali v spremenljivki "command". Da se v spremenljivki `command` zapisan ukaz ob sprejemu na modul tudi dejansko izvede, moramo podani ukaz zaključiti še s simulacijo pritiska na tipko Enter: `\r\n`.

```
SendTextCommandToAllClients : function( command ) {  
    for( idx in this.connectedDevices ) {  
        this.connectedDevices[idx].WriteToSocket( command );  
    }  
}  
  
WriteToSocket : function(str) {  
    this.socket.write(Ti.createBuffer({ value: str + "\r\n"}));  
}
```

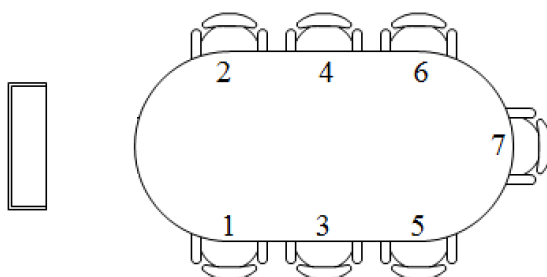
V drugem meniju kamere imamo možnost pozicioniranja kamere na 7 različnih pozicij. Pozicije so določene tako, da sledijo izgledu mize videokonferenčne sobe, prikazane v ozadju aplikacije. Odločili smo se implementirati 7 različnih pozicij za kamero, saj imamo v konferenčni sobi 7 stolov. Vsak gumb predstavlja en stol konferenčne sobe; prvi gumb postavi kamero na prvi levi stol konferenčne sobe, drugi gumb predstavi kamero na prvi stol na desni strani mize ... Izgled menija 'Camera 2' si lahko ogledamo na sliki 4.6. Na sliki 4.7 so prikazane definirane pozicije za pozicioniranje kamere stojala.

4.4 Povezava aplikacije na modul in pošiljanje ukazov

Osnova za vzpostavitev komunikacije med aplikacijo in videokonferenčnim stojalom je brezžična povezava. Aplikacija in modul se povezujeta preko TCP/IP protokola in uporabljata brezžično povezavo za pošiljanje podatkov.



Slika 4.6: Meni "Camera 2".

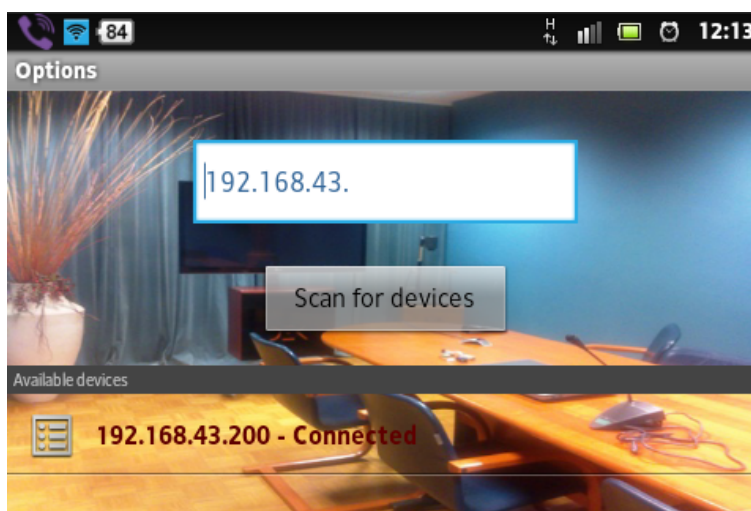


Slika 4.7: Prikaz pozicij za predstavitev kamere v modulu "Camera 2".

Modul se lahko poveže na omrežje z uporabo osnovnih ACTION ukazov. Ukaz, ki smo ga uporabili za povezavo modula na omrežje, je "join". Glede na to, da smo se povezovali na SSID, ki je bil zaščiten z geslom, smo morali v konfiguracijo modula najprej zapisati geslo, saj je to bil pogoj za vzpostavitev povezave. Geslo smo vpisali v konfiguracijo modula z uporabo ukaza "set wlan phrase <geslo>", saj smo se povezovali na WPA zaščiteno omrežje. Predpogoj za uspešno povezavo aplikacije na stojalo je uspešna povezava modula na brezžično omrežje. Šele ko je povezava uspešna, bo aplikacija zaznala modul in se nanj povezala z uporabo Ad Hoc načina.

Ko je modul povezan na omrežje, lahko z uporabo menija konfiguracije

DEK.Remote aplikacije poiščemo modul in se nanj povežemo. Če je bila povezava na modul uspešna, bomo prejeli ”*HELLO*”, na zaslonu aplikacije se bo prikazalo obvestilo o uspešni povezavi. Prikaz izgleda zaslona ob uspešni povezavi si lahko ogledamo na sliki 4.8.



Slika 4.8: Uspešna povezava na modul.

Komunikacija med aplikacijo in modulom ni zakriptirana, ukazi se pošiljajo kot preprosti besedni ukazi. Ko smo se uspešno povezali na modul, za preklop v način upravljanja pošljemo ukaz \$\$\$\$. Od modula dobimo odgovor CMD, kar pomeni, da smo uspešno preklopili v način upravljanja. V primeru, da stojalu pošljemo neveljaven ukaz, dobimo odgovor ERR.

Ukazi za krmiljenje kamere in mikrofona so zakodirani v kodi aplikacije. Programska oprema, ki se izvaja na Wi-Fi RN-174 modulu, je uporabljena kot dostopna točka do konfiguracije stojala in je uporabljena za posredovanje podatkov, sprejetih od aplikacije, proti stojalu. Ob sprejemu ukaza modul stojalu posreduje določene parametre za izvedbo določene akcije in vrne aplikaciji potrditev, da so ukazi bili uspešno sprejeti.

4.5 Testiranje pravilnosti delovanja aplikacije

V praksi smo proces testiranja aplikacije razdelili na dva glavna dela:

1. Izdelavo testne aplikacije za testiranje komunikacije med Android napravo in videokonferenčnim sistemom preko RN174 modula.
2. Testiranje funkcionalnosti aplikacije.

Preden smo začeli s postopkom izdelave aplikacije za upravljanje z videokonferenčnim sistemom, smo izdelali testno aplikacijo, ki je vsebovala dva gumba, ki sta omogočala vzpostavitev povezave in testiranje pravilne komunikacije med aplikacijo in videokonferenčnim sistemom (oz. testiranje RS232 RN174 modula in aplikacije).

Testna aplikacija je imela dva gumba; en gumb vzpostavi povezavo med aplikacijo in modulom, drugega smo uporabili za vklop LED indikatorjev na modulu – z uporabo gumba smo jih lahko vklopili/izklopili. Z uporabo testne aplikacije smo odpravili vse odkrite napake, ki so vplivale na način povezovanja in pošiljanja podatkov med aplikacijo in iMMS-jem. Hkrati je bila osnova za izdelavo DEK.Remote aplikacije.

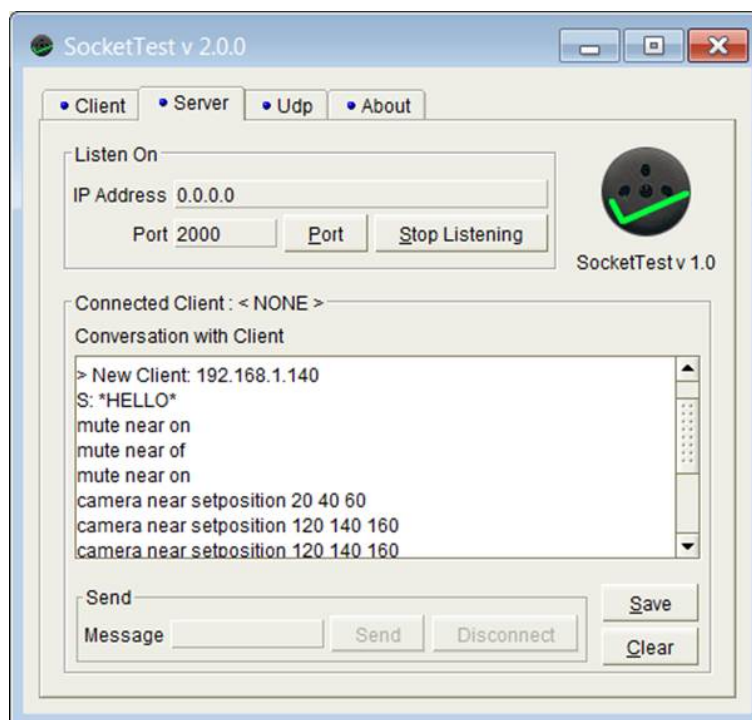
Testiranje posameznih delov aplikacije oz. testiranje programskih enot je potekalo že med samim razvojem aplikacije, tako na emulatorju kot tudi na pametnem telefonu. Pri testiranju posameznih modulov se je pojavilo kar nekaj napak, ki smo jih sproti odpravili. Najbolj pogoste napake so bile povezane s povezavo aplikacije na modul in pošiljanjem ukazov. Povezavo smo tudi testirali z uporabo "SocketTest" aplikacije (slika 18). SocketTest [8] je Java orodje za testiranje vtičnic. Z uporabo orodja lahko ustvarimo TCP/UDP vtičnico na strežniku ali na odjemalcu. Lahko se uporablja za testiranje strežnikov ali odjemalcev, ki za komunikacijo uporabljajo TCP ali UDP protokol. Z uporabo orodja smo lahko preverili uspešno povezavo aplikacije in stojala in način pošiljanja ukazov stojalu.

Glede na to, da obstaja velik nabor naprav, ki jih lahko razdelimo po velikosti in gostoti zaslonov, smo morali testirati tudi grafični vmesnik pro-

gramske opreme. Testiranje je vsebovalo tudi nastavljanje velikosti pisave za menije in gumbе in pozicioniranje le-teh na zaslonu.

Pred testiranjem smo si najprej pripravili tesni scenarij (upoštevali smo vse možne scenarije pri vsaki izmed funkcionalnosti aplikacije), nato pa smo se lotili testiranja najprej na emulatorju, potem pa še na pametnem mobilnem telefonu. Pri testiranju delovanja aplikacije nismo uporabili orodja, ki bi nam omogočilo avtomatsko testiranje modulov. Odločili smo se napisati natančen postopek testiranja modulov in jih ročno izvesti v drugi fazi testiranja.

V končni fazi testiranja nismo našli nobenih napak, kar pomeni, da je bilo testiranje posameznih modulov aplikacije v začetku dobro opravljeno. Pri praktični uporabi aplikacije do sedaj tudi nismo našli novih napak.



Slika 4.9: Prikaz testiranja uspešne povezave modula in aplikacije ter pošiljanje ukazov.

Poglavje 5

Sklepne ugotovitve

V diplomskem delu smo predstavili tehnologijo videokonferenčnih sistemov ter tehnologijo in razvojna orodja, ki so bila povezana z razvojem aplikacije na Android platformi. V diplomskem delu smo predstavili vse faze razvoja mobilne aplikacije, katerih končni rezultat je delujoča aplikacija "DEK.Remote". Zajema tudi opis tehnologij, ki smo jih uporabili za izdelavo mobilne aplikacije.

Implementirane so bile vse funkcionalnosti aplikacije, ki so bile zahtevane in načrtovane, z izjemo delovanja aplikacije na operacijskem sistemu iOS, glede na to, da moramo uporabljati MacOS operacijski sistem in imeti licenco za objavo aplikacije na Apple store.

DEK.Remote je mobilna aplikacija, ki jo lahko uporabljamo na kateri koli mobilni napravi, ki uporablja Android operacijski sistem. Aplikacija omogoča veliko lažje in hitrejše upravljanje z videokonferenčnim sistemom. Prav tako bo aplikacije razbremenila uporabnika, ki uporabljajo videokonferenčno stojalo iMMS.

Uporabniški vmesnik je intuitiven in enostaven za uporabo. Aplikacija je zastavljena tako, da omogoča enostavno nadgradnjo z novo funkcionalnostjo. V prihodnosti bomo lahko implementirali še dodatne module, ki so potrebni za popolno upravljanje z videokonferenčnim sistemom.

Aplikacijo lahko nadgradimo s poljubnimi komponentami. V naslednji

verziji aplikacije je načrtovana razdelitev na vrste prestižne verzije stojala in možnost izbire stojala, ki ga uporabljamo (single monitor, double monitor ...). Glede na to, da stojalo lahko uporabljamo v kombinaciji zaslona in projektorja, smo se odločili dodati tudi možnost upravljanja z njima v samo aplikacijo.

Ena od glavnih omejitev aplikacije je, da so trenutno položaji kamere dodani že v kodo aplikacije in se lahko uporabljajo samo v konferenčni sobi podjetja. Glede na to omejitev je v naslednji verziji aplikacije načrtovano omogočanje vnosa poljubnih koordinat, na katere naj bi se postavila kamera stojala in tudi izboljšava grafičnega izgleda aplikacije.

Literatura

- [1] (2014) Videokonferenčni sistemi. Dostopno na:
<http://en.wikipedia.org/wiki/Videoconferencing>.
- [2] (2014) Seznam videokonferenčnih sistemov. Dostopno na:
http://en.wikipedia.org/wiki/List_of_video_telecommunication_services_and_product_brands
- [3] (2014) Polycom. Dostopno na:
<http://www.polycom.com/>.
- [4] (2014) Appcelerator Titanium Studio. Dostopno na:
<http://www.appcelerator.com/>.
- [5] (2014) Appcelerator Titanium Studio. Dostopno na:
http://en.wikipedia.org/wiki/Appcelerator_Titanium.
- [6] (2014) Definicije vrst mobilnih aplikacij. Dostopno na:
<http://blogs.telerik.com/appbuilder/posts/12-06-14/what-is-a-hybrid-mobile-app->
- [7] (2014) RN-171 modul. Dostopno na:
<http://www.rovingnetworks.com/products/RN171>.
- [8] (2014) SocketTest. Dostopno na:
<http://sockettest.sourceforge.net/>