

CHALMERS



Generaliserad fjärrstyrning med Android applikation Generalized remote control with Android application

Martin Jonasson
Daniel Hultman

Examensarbete
Högskoleingenjörsprogrammet för datateknik
CHALMERS TEKNISKA HÖGSKOLA
Institutionen för data- och informationsteknik
Göteborg 2012

Innehållet i detta häfte är skyddat enligt Lagen om upphovsrätt, 1960:729, och får inte reproduceras eller spridas i någon form utan medgivande av författaren. Förbudet gäller hela verket såväl som delar av verket och inkluderar lagring i elektroniska och magnetiska media, visning på bildskärm samt bandupptagning.

©Martin Jonasson, Daniel Hultman, Göteborg 2012

Sammanfattning

Fjärrstyrning används allt flitigare idag än vad det gjordes för bara några fåtal år sedan. Nu kan man styra i princip allt man har inkopplat i sitt hem. Detta har lett till att fler företag har skapats för ta sig in i den växande marknaden. För att kontrollera dessa fjärrstyrningssystem så använder man en form av styrdon. Många har idag skapat en smartphone applikation för att samla alla styrmöjligheter i en enhet. Men det finns fortfarande de som använder en fast styrpanel som är skräddarsydda för att passa till just det systemet. Detta examensarbete är grundat på granskning av relevant litteratur för att designa en Android applikation. Med olika typer av bedömningar skall den göras generell för att kunna samla all fjärrstyrning i en och samma applikation. För att lyckas med detta har prototypen evaluerats med den internationella ISO standarden ISO/IEC 9126. När prototypen väl blev klar utfördes ett experiment för att testa dess användarkvalité som är en av ISO/IEC 9126 testfall. Med hjälp av resultaten från kvalitetsbedömningen kan applikationen utvecklas på så sätt att det är möjligt att implementera flera typer av fjärrstyrningssystem. Slutprodukten för denna första version av prototypen blev en Android applikation, anpassad till företaget Kramers fjärrstyrningssystem, som kan styra enheter som kan finnas i ett konferensrum.

Abstract

Remote control are today used more frequently than just a few years ago. Now you can control pretty much everything that is plugged in in your home. This has led to that more and more companies have been created to get into the growing market. To manage these remote control systems a kind of steering gear is used. Many companies has developed smartphone applications to steer the remote control system. However, there are still those who uses stationary control panels with customized interfaces to suit their specific system. This thesis is based on a review of relevant literature to design an Android application. With different measurements it will be developed in a general way to gather all remote control possibilities in one application. To succeed with these goals the prototype will be evaluated with the international ISO standard ISO/IEC 9126. When the prototype were done an experiment was carried out to test its usability which is one of ISO/IEC 9126 test cases. From the results of the quality assessment can the application be developed in a way that is possible to implement several different remote control systems. The final product of this prototype became an Android application, adapted to the company Kramers remote control system, which is able to steer units usually used in a conference room.

Keywords: Fjärrstyrning, Android, Android applikation, ISO/IEC 9126, Usability

Innehållsförteckning

1.0 Inledning.....	1
1.1 Bakgrund.....	1
1.2 Problemställning.....	1
1.3 Syfte.....	1
1.4 Mål.....	1
1.5 Avgränsningar.....	2
• Applikationen kommer endast fungera till Android telefoner.....	2
1.6 Rapportdisposition.....	2
2.0 Metod.....	3
2.1 Versioner.....	3
3.0 Teknisk Bakgrund.....	5
3.1 Fjärrstyrning.....	5
3.1.1 Historia.....	5
3.1.2 Fjärrstyrningssystem.....	8
3.1.2.2 Aktuellt arbetssätt.....	9
3.2 Kommunikation.....	9
3.2.1 RS-232.....	9
3.2.1.1 Historia.....	9
Figur 8. Koppling mellan DTE och DCE.....	10
3.2.1.2 Kablar.....	10
3.2.1.3 Anslutning.....	10
3.2.2 Infraröd strålning(IR).....	12
3.2.2.1 Historia.....	12
3.2.2.2 Kommunikation.....	12
3.2.3 User Datagram Protocol (UDP).....	13
3.3 Android.....	14
3.3.1 Komponenter.....	14
3.3.1.1 Activities.....	14
3.3.1.2 Intents.....	15
3.3.1.3 Services.....	15
3.3.1.3 Content Provider.....	15
3.4 ISO/IEC 9126.....	16
3.4.1 Quality model.....	16
3.4.1.1 Functionality.....	16
3.4.1.2 Reliability.....	16
3.4.1.3 Usability.....	17
3.4.1.4 Efficiency.....	17
3.4.1.5 Maintainability.....	17
3.4.1.6 Portability.....	17

3.4.2 Intern kvalitet, extern kvalitet och användarberoende kvalitet	18
4.0 Design och Implementering	19
4.1 Android.....	19
4.1.1 Design	19
4.1.2 Implementering	21
4.1.1 Implementering av "Device setup"	21
4.1.2 Implementering av ikonplacering	21
4.1.2 Implementering av "Device managing"	23
4.1.3 Implementering av ljudkontroll	23
4.2 Kommunikation	24
4.2.1 Design	24
4.2.2 Implementering	25
4.2.2.1 Implementering av "Communication"	25
4.2.2.2 Implementering av "Command send helper"	25
5.0 Evaluering.....	26
5.1 Evalueringsattribut.....	26
5.1.1 Functionality.....	26
5.1.2 Usability	27
5.1.3 Reliability	27
5.1.4 Portability	27
5.2 Experiment	28
5.2.1 Testgrupp	28
5.2.2 Testmoment.....	28
5.0 Testresultat	32
5.1 Utvecklaraspekt.....	32
5.2 Användaraspekt	33
5.3 Diskussion.....	35
6.0 Slutsats.....	37
6.1 Kritisk diskussion.....	37
6.2 Fortsatt forskning.....	37
Referenser.....	38

Förkortningar och begrepp

Aktivitet, Aktiviteter - I Android applikationer använder man klasser som kallas Activity. Det finns en Activity för varje skärmbild man får upp på telefonen.

AVD - Android Virtual Device: en Android emulator för att testköra de program man skriver på en dator.

DCE - Data Circuit-termination Equipment: en enhet som är inkopplad mellan en DTE och en överföringsenhet. Vanligt är DCE ett modem.

DTE - Data Terminal Equipment: en slutenhet som konverterar och återomvandlar användar information till signaler. Kommunikerar med en DCE.

GUI - Graphical User Interface: ett grafiskt användargränssnitt för att underlätta interaktion mellan telefon och användare.

SDK - Software Development Kit: ett paket med utvecklingsverktyg för att skapa applikationer till en specifik plattform.

1.0 Inledning

1.1 Bakgrund

Vid fjärrstyrning idag används för det mesta en panel eller en fjärrkontroll för att styra de enheter som är inkopplade till ett styrsystemet. Dessa kontroller är skräddarsydda för det företag man köpt systemet ifrån. I detta projekt skall en lösning till detta problem framställas i form av en applikation till Android telefoner. Det skall vara en generell applikation som är tänkt till att fungera för flera fjärrstyrningssystem. Den här produkten kommer att till en början endast fungera till Kramers styrsystem men det kommer vara möjligt att vidareutveckla den till att även fungera för andra fjärrstyrningssystem. Man kan via fjärrstyrning styra i princip alla enheter som har en anslutning som matchar styrsystemets ingångar. Till exempel, en enhet som har IR anslutning kan anslutas till ett system med IR-ingång. Detta gör att det finns nästintill oändligt många tekniska produkter man kan fjärrstyra.

1.2 Problemställning

Att göra en generell applikation för manövrering av ett fjärrstyrningssystem i ett konferensrum. Meningen med generell applikation är att det lätt skall gå att implementera olika fjärrstyrningssystem i applikationen.

1.3 Syfte

Syftet med detta projektet är att ersätta stationära styrpaneler till fjärrstyrningssystem med en fjärrstyrningsapplikation till Android telefoner. Produkten skall sedan evalueras med hjälp av ISO standarden, ISO/IEC 9126 för att mäta kvaliteten på dess generalitet.

1.4 Mål

- Utveckla en prototyp som är en Android applikation.
- Den skall kunna kontrollera ett fjärrstyrningssystem.
- Det skall gå att vidareutveckla den för att även kunna styra flera olika typer av fjärrstyrningssystem.
- En kravspecifikation ska formuleras för att underlätta utvecklingen av applikationen.
- Enheter som första prototypen av applikationen skall kunna styra är: DVD, TV, projektor, projektorduk, ljudanläggning, datorer, ljus, mörkläggningsgardiner.
- Evalueras och få en bra bedömning enligt ISO/IEC 9126.
- En hemsida skall skapas där konfigurationsmallar till Kramers konfigureringsprogram K-config skall finnas tillgängliga för användaren.

1.5 Avgränsningar

- Applikationen kommer endast fungera till Android telefoner.
- Applikationen kommer att utvecklas mot Kramers produkter, dock på ett sådant sätt att det lätt går att utveckla den mot andra liknande produkter.
- Applikationen kommer att vara anpassad för konferensrum.

1.6 Rapportdisposition

2.0 Metod: Detta avsnitt beskriver hur vi lagt upp arbetet.

3.0 Teknisk Bakgrund: Bakgrundsinformation till projektet vilket innefattar fjärrstyrningens historia, kommunikationen mellan produkter, Android och ISO/IEC 9126.

4.0 Design och Implementering: Här förklaras hur arbetet har utvecklats och fungerat.

5.0 Resultat: En värderingsfri presentation av de resultat vi uppnått.

6.0 Slutsats: Detta avsnitt innefattar våra diskussioner om projektet utifrån våra resultat.

2.0 Metod

Detta projekt innefattar som tidigare nämnts en Android applikation som kommer att utvecklas och skrivas med programmeringsspråket Java. Applikationen skrivs med hjälp av verktyget Eclipse med en installerad Android SDK och AVD för att få tillgång till de olika metoder och klasser som behövs vid utveckling av Android applikationer. Utvecklingen av vår produkt sker i ett antal iterationer. Vid varje iteration sker test för att försäkra oss om att allt fungerar som det skall innan vi fortsätter.

När applikationen är färdigställd skall denna evalueras med ISO/IEC 9126. Detta utförs för att undersöka om den uppfyller viktiga krav som en generell produkt innebär. Vi kommer göra en noggrann undersökning om hur kvalitetsstandarden ISO/IEC 9126 fungerar och på vilket sätt andra produkter har evaluerats genom att titta på elektroniska rapporter och bilagor. Applikationen kommer sedan testas utifrån dels ett utvecklaraspekt och dels ett användaraspekt. De saker vi testar från utvecklaraspekten är funktionaliteten, pålitligheten, säkerheten och porteringsmöjligheten. Vid testning från användaraspekten skall vi ta hjälp av ett antal testpersoner som får utföra ett experiment där de testar applikationen på olika sätt. Dessa personer kommer även att intervjuas för att få en uppfattning om vad som kan och behöver ändras i applikationen. Utifrån resultaten från evalueringen och experimentet drar vi våra slutsatser om huruvida vi har lyckats skapa en generell applikation och vilka funktioner som kommer behöva justeras.

2.1 Versioner

För att underlätta arbetet har vi valt att arbeta med iterationer där vi testat varje del för sig för att försäkra oss om att det fungerar som planerat. I första iterationen tänker vi börja med ett GUI för enkel navigering i applikationen tillsammans med en aktivitet för en av enheterna. När detta fungerar börjar vi på andra iterationen som skall innefatta aktiviteter för de resterande enheterna. I denna iteration skall även styrningen av den första enheten fungera.

Efter att vi testat iteration två fortsätter vi i iteration tre genom att implementera styrningen av resterande enheter även förbättra GUI:t något.

När de tre iterationerna är färdigställda och testade är sista steget att finjustera detaljer och eventuella buggar som vi upptäcker på vägen.

Bifogat finns en kravspecifikation som versionerna nedan refererar till.

Version 1:

1. Simpelt Gui som gör det möjligt att navigera i applikationen.
2. DVD aktivitet, knappar för styrning av DVD ska vara implementerade:
play, pause, stop, next, previous, fastforward, rewind, menu, audio, subtitle, power(on/off), eject, navigering.
3. Ikoner för alla enheter som går att styra med denna applikation:
Projektor, Projektorduk, TV, Dator, Mörkläggningsgardiner, Ljudkontroll, Belysning.

Version 2:

1. Aktiviteter för alla enheter, knappar för styrning av dessa:
 - Projektor* : power(on/off), aspect, input, brightness(+/-), contrast(+/-). (krav# 1.3.2)
 - Projektorduk* : upp, ner. (krav #1.3.4)
 - TV* : power(on/off), aspect, input, volym(öka,minska,mute), kanalbyte. (krav #1.3.3)
 - Dator* : kan finnas knappar för olika datorer, den dator man väljer kommer att visas på skärmen. (krav #1.3.8)
 - Mörkläggning* : upp, ner. (krav #1.3.5)
 - Ljudkontroll* : öka, minska, mute. (krav #1.3.7)
 - Belysning* : tänd, släck. (krav #1.3.6)

2. Konfigurationsmall till Kramers switchar för att underlätta för användare;
 - Detta är en konfiguration som laddas in på fjärrstyrningsswitchen för att ställa in vilka enheter som är inkopplade till den. Vi skapar en mall för att underlätta arbetet för användaren.
 - Denna mall skall följas för att applikationen skall fungera utan störningar (krav #2.8).
 - Den kommer finnas tillgänglig på en hemsida tillsammans med en utförligt beskriven installations guide(krav #2.6 och #2.7).

3. Styrning av DVD fungerar. Vi skall testa mot switchen för att kontrollera att en DVD kan styras.

Version 3:

1. Styrning av alla enheter fungerar(krav #1.3). Testa samtliga enheter mot switchen.
2. Utveckla GUI:t till ett snyggare och stilrent utseende.
 - Lättanvänt för användaren(krav #2.4).
 - Färgtema, egna knappar och ikoner, logga.

Version 4:

1. Finslipning av applikationen, korrigerar buggar och diverse problem vi stött på under projektets gång.
 - Arbeta mer på GUI:t.

3.0 Teknisk Bakgrund

Styrssystem för fjärrstyrning använder och har använt sig av många olika tekniker. Detta kapitel tar upp hur fjärrstyrningen har utvecklats och om hur det fungerar. Det tar även upp och förklarar innebörden av ISO/IEC 9126 och hur den används.

3.1 Fjärrstyrning

Fjärrstyrning har funnits sedan Nikola Tesla visade upp den första radiostyrda motorbåten 1898. Det har sedan dess utvecklats till att styra allt från militära fordon, till TV:n, och nu till nästan alla elektroniska enheter som idag existerar. [10]

3.1.1 Historia

Det första tecknet på fjärrstyrning visade Nikola Tesla i en demonstration av sin radiostyrda motorbåt 1898. Han hade sedan 1893 haft patent på just fjärrstyrning och förklarade processen i detta patent. [9]

Detta öppnade en dörr för militära maskiner. I första världskriget använde tyska flottan fjärrstyrning för att styra båtar och störa fiendliga fartygs möjlighet att avancera. Även i andra världskriget användes fjärrstyrning, här för att styra olika typer av vapen som till exempel radiostyrda bomber. [9]

Till en början användes fjärrstyrning för att styra olika maskiner och vapen i militärt syfte men när kriget var slut så började det forskas i hur det kunde användas av allmänheten.

På sena 40-talet utvecklade man den första automatiska garageporten. [4]

Efter det kom grunden till det som används flitigt idag, en fjärrkontroll till TVn. 1952 utvecklade Zenith Radio Corporation fjärrkontrollen "Lazy Bones". Den hade en sladd kopplad från dosan till en motor i TVn som gjorde ändringar på tunern beroende på vad användaren tryckte på. Man kunde även stänga av och sätta på sin TV. Detta blev en stor succé men den fick kritik för sin klumpiga kabel som låg i vägen som många snubblade över. [9, 4]



Figur 1. "Lazy Bones" fjärrkontroll.

Detta ledde till en vidareutveckling av produkten och samma företag skapade 1955 den första trådlösa fjärrkontrollen "Flashmatic". Den fungerade så att man hade en kontroll som vid knapptryck skjuter ifrån sig ett starkt ljus, ungefär som dagens laserpekare, detta ljus skall man sikta på en av de 4 fotocellerna som satt i hörnen på TVn. Varje fotocell hade en funktion när man lös på den. Den ena var att sätta på och stänga av volymen, den andra sätta på och stänga av bilden, den tredje och byta kanal. Problemet med denna produkt var att fotocellerna kunde registrera annat ljus än det som kom från fjärrkontrollen. Hade man sin TV i ett rum där solens strålar kommer åt den så kunde TVn "spöklukt" göra som den ville och helt plötsligt byta kanal. [9, 4]



Figur 2. "Flashmatic" fjärrkontroll.

För att lösa detta utvecklade de fjärrkontrollen "Space command". 1956 kom Zenith med idén att använda ultraljud för att skicka signaler från dosan till TVn. Den hade 4 knappar som bilden 3.1.3 visar och hade samma funktioner som "Flashmatic":s fotoceller hade. Den här varianten av "Space command" var väldigt dyr eftersom det krävdes en speciell mottagare på TVn för att den skulle ta emot och registrera ultraljudet. Den mottagaren använde sig av elektronrör. Dessa kom sedan att ersättas av transistorer vilket gjorde det billigare. Fjärrkontrollen gjordes om så att ultraljudet skapades elektroniskt med hjälp av batterier vilket tidigare fjärrkontroller inte haft. [9, 4]



Figur 3. "Space Command" fjärrkontroll.

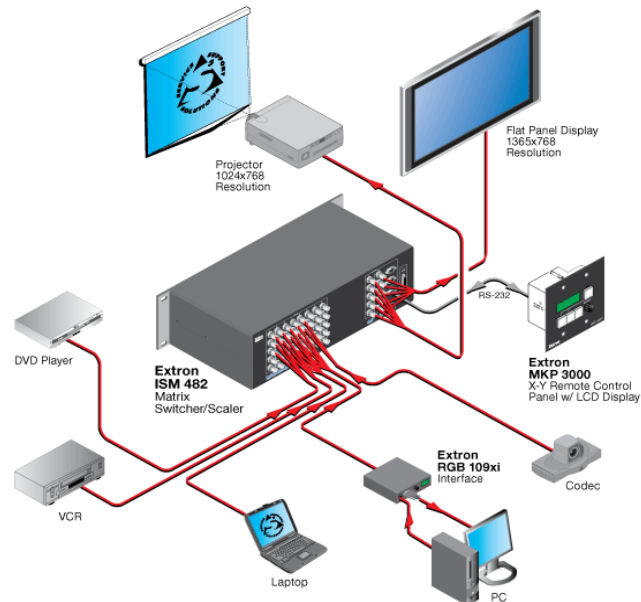
"Space command" fanns i 25 år innan man bytte ut tekniken mot IR. I början på 1980 talet introducerades denna teknik till fjärrstyrningskontroller. IR använder sig av en lågfrekvent ljusstråle som är osynlig för ögat och på så sätt inte störande. Vid varje knapptryck skickas en infraröd stråle som fångas upp av en mottagare som är installerad i TVn. Det är detta som man än idag använder för att styra sin TV och även andra produkter. [9, 4]



Figur 4.. IR fjärrkontroll

3.1.2 Fjärrstyrningssystem

I allt fler hushåll och byggnader idag så installeras någon form av fjärrstyrningssystem. Det kan vara allt ifrån de största systemen där man i princip kan styra allt som finns i byggnaden, till den minsta varianten där man till exempel bara vill styra värmen inomhus.



Figur 5. Exempel på fjärrstyrningssystem, här från Kramer

3.1.2.1 Äldre arbetssätt

De första fjärrstyrningssystemen man installerade använde man en stationär styrpanel som var placerad på lämplig plats i huset. Denna panel var optimerad för att stödja det styrsystem man hade installerat. Det finns med andra ord unika paneler för varje enhet man vill styra. Ju fler enheter man kopplar till sitt system desto fler paneler är man tvungen att använda.



Figur 6. Styrpanel från Neets.

3.1.2.2 Aktuellt arbetssätt

I många styrsystem idag används fortfarande styrpaneler för att fjärrstyra. Men nu när smartphones och touchpads har blivit aktuella så har de flesta företag utvecklat en applikation för att sköta styrningen. Detta gör att man slipper en styrpanel för varje enskild enhet i systemet. Nu har man istället allt samlat på en enda skärm.



Figur 7. Applikation för Crestron system

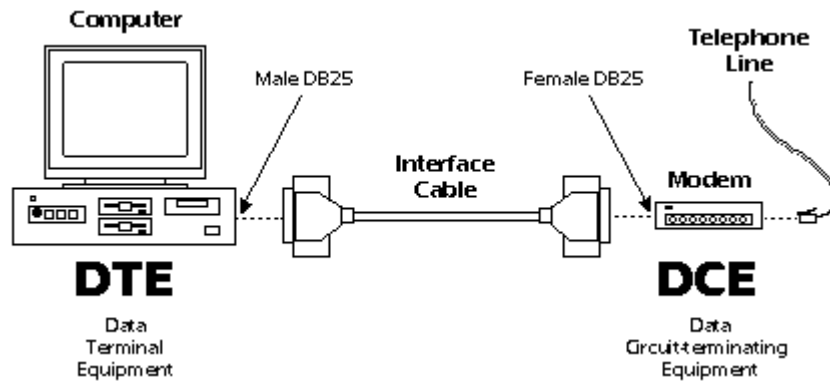
3.2 Kommunikation

3.2.1 RS-232

RS-232 är en ANSI-standard som beskriver signaler mellan till exempel en dator(DTE) och ett modem(DCE). Vanligtvis används det i datorers seriella portar. För sändning av data använder sig RS-232 sig av logiska ettor och nollor där +3V till +12V är en logisk nolla och -3V till -12V är en logisk etta [12, 11].

3.2.1.1 Historia

RS-232 introducerades 1962 och användes först och främst till teleprinterar. Den första DTE:en var en teleprinter och den första DCE:en ett modem. När sedan elektroniska terminaler kom uppdaterades RS-232 standarden år 1969 till RS-232C som stödjer terminalerna. Dessa terminaler var ofta designade för att vara utbytbara med teleprinterarna. Senare började datorer att använda standarden för att kunna ansluta till annan existerande utrustning. I många år var RS-232 standard för seriella anslutningar till datorer. Datorer använde detta fram till i slutet av 1990-talet då det ersattes av USB. RS-232 används dock fortfarande för äldre utrustning, industriell utrustning(så som PLC:er), konsollportar och så vidare [11].



Figur 8. Koppling mellan DTE och DCE.

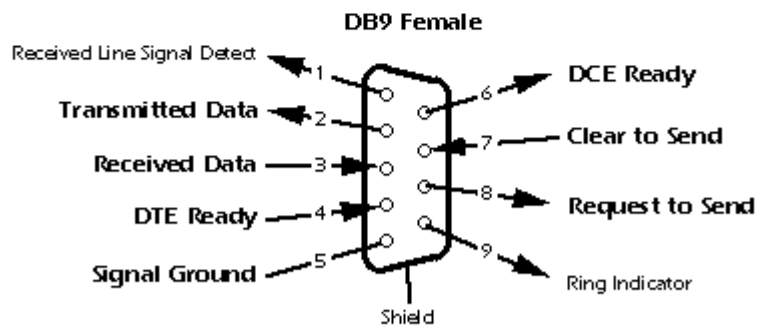
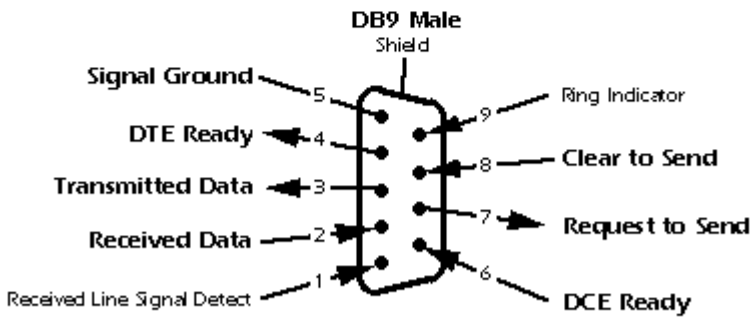
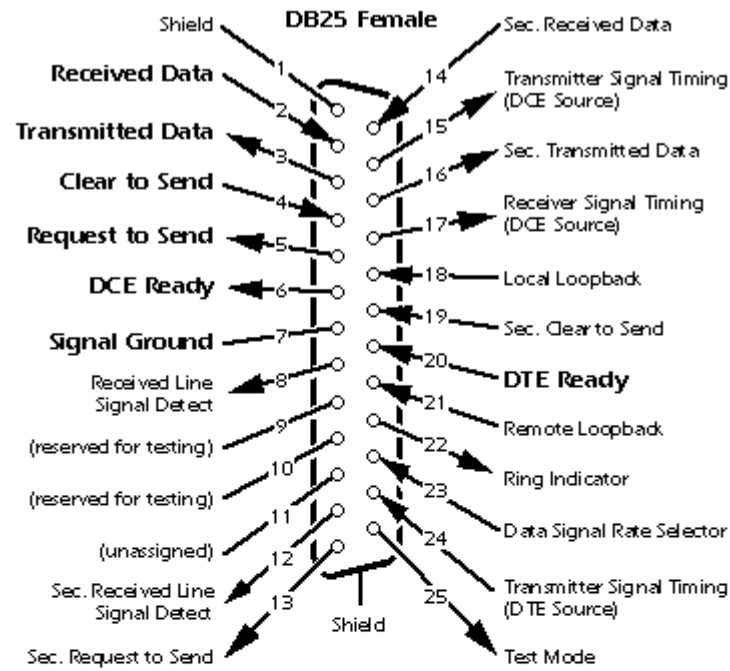
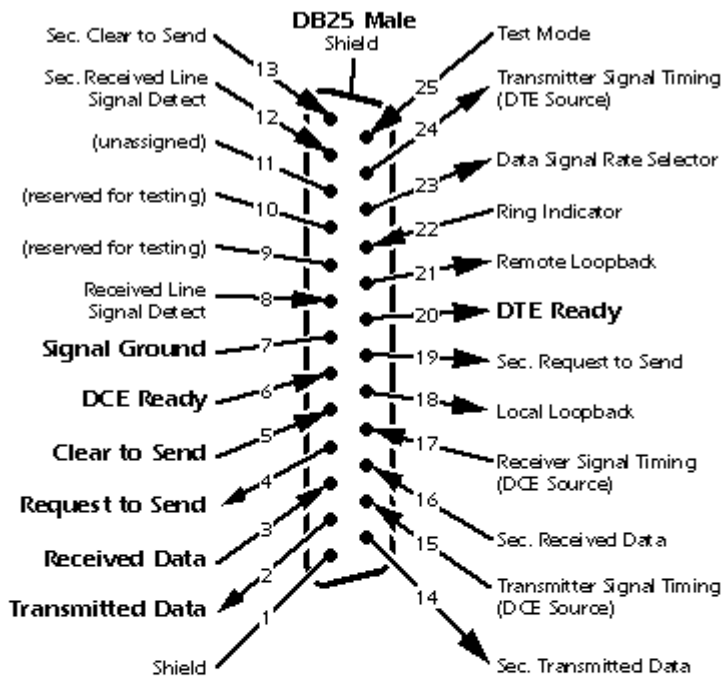
3.2.1.2 Kablar

Istället för att sätta en maximal längd på hur långt signalerna skickas har man satt en maximal kapacitans. Så en vanlig kabel har en max längd på ca 15 meter men om man använder en kabel med låg kapacitans med kabellängder upp till 300 meter ge samma överförings hastigheten [11].

3.2.1.3 Anslutning

Det finns två olika sorters anslutningar för RS-232. Den ena heter ED9 och den andra ED25 och skillnaden mellan dem är att de har olika många stift. ED9 har nio stift och ED25 har 25 stift. Figurerna 9 och 10 visar alla de olika stiftens funktioner både för ED9 och ED25 [12, 11].

Looking Into the DTE Device Connector



- ← Received by DTE Device
- → Transmitted from DTE Device

- ← Received by DCE Device
- → Transmitted from DCE Device

Figur 9. DCE enhetens anslutning

Figur 10. DTE enhetens anslutning

3.2.2 Infraröd strålning(IR)

Infraröd strålning också kallad IR, är en elektromagnetisk strålning med våglängden 700nm till 1mm vilket är strax över våglängden för synligt ljus. Mycket av energin som kommer från solen är infraröd strålning. Infrarött ljus har en rad olika användningsområden så som inom industrin, sjukvården, forskning, mörkerseende, kommunikation med mera [5].

3.2.2.1 Historia

Upptäckten av Infraröd strålning gjordes tidigt under 1800-talet av astronomen William Herschel. Namnet infraröd strålning uppkom i slutet av 1800-talet. Innan dess kallade William den "Calorific Rays". Sedan dess har infrarött ljus använts inom många olika områden [5].



Figur 11. IR ljus i X-box kontroll

3.2.2.2 Kommunikation

Infrarött ljus för överföring av data är kortvågig och används för kommunikation på små avstånd som till exempel att styra TV:n med fjärrkontrollen. Det fungerar på så sätt att en ljusdiod avger infraröd strålning som går genom en lins och fokuseras till en stråle. Strålen fångas upp av en mottagare och översätts sedan till elektrisk ström av en kisel-fotodiod. Även glasfiber kan användas för transporterering för längre sträckor [5].

3.2.3 User Datagram Protocol (UDP)

UDP är ett anslutningslöst transportprotokoll för sändning av datapaketer. Med anslutningslöst menas det att ingen anslutning mellan enheterna görs och alla paketen behöver inte ta samma väg och kan anlända i oordning. UDP har inte heller någon kontroll av paketförlust utan det får programmeraren utforma själv om det är något viktigt som måste komma fram. Det positiva med UDP är dock snabbheten[13].

Ett UDP paket har två delar, en header och ett fält med data.

UDP headern består av 4 delar.

Ursprungsport - Portnummer där paketet skickades ifrån.

Mottagarport - Portnummer till den paketet är till.

Längd - Specifierar hela paketets längd, både headern och datan.

Checksumma - Ett fält för att upptäcka om fel uppstått[13].

offset (bits)	0 – 15	16 – 31
0	Source Port Number	Destination Port Number
32	Length	Checksum
64+	Data	

Figur 12 Ett UDP pakets delar

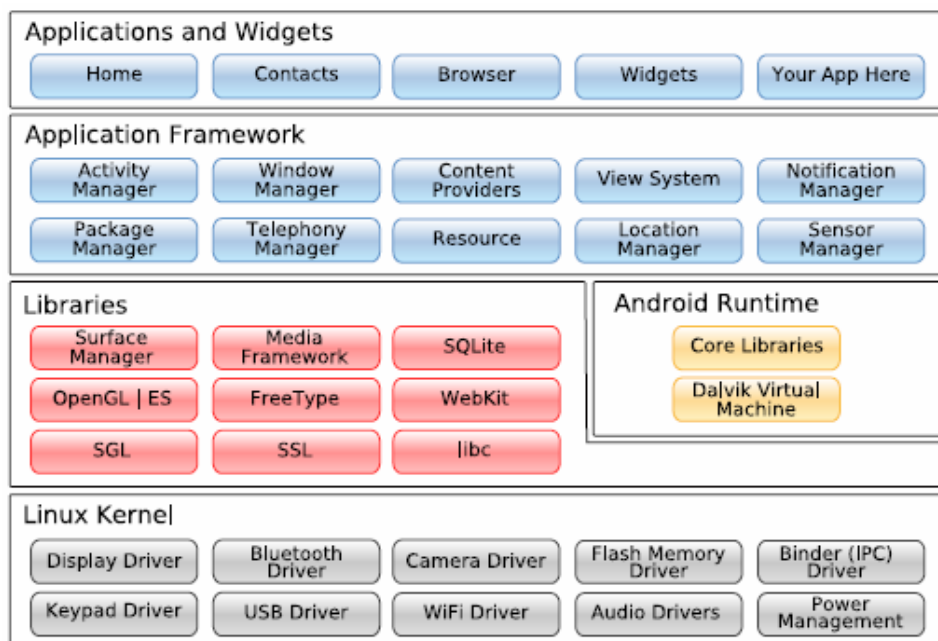
3.3 Android

Android är ett operativsystem som används till mobiltelefoner och surfplattor. Det har framställts av Google tillsammans med flera andra företag och är uppbyggt i ett antal lager enligt bild nedan.

Android har som grund operativsystemet Linux. Linux bidrar med huvuddelar så som säkerhet, minneshantering, processhantering. Linux arbetar även som ett kommunikationslager mellan hårdvara och mjukvara.

Vid utveckling i Android så använder man programmeringsspråket Java. Biblioteken som används skiljer sig något från Javas standard bibliotek för att optimera koderna för mobiltelfoner.

En Android applikation byggs upp av ett antal komponenter. Dessa komponenter bildar tillsammans en applikation [1,14,13].



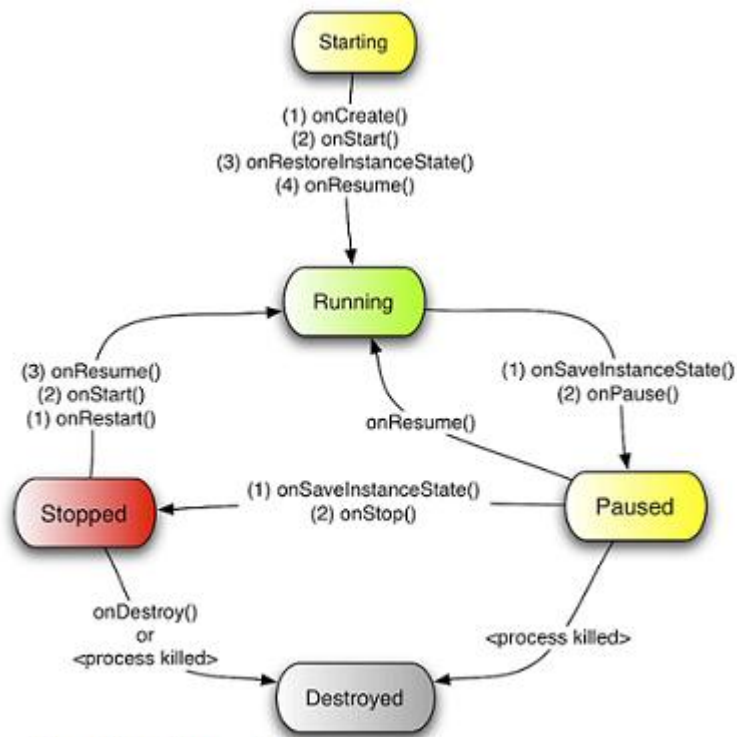
Figur 13. Androids Systemarkitektur

3.3.1 Komponenter

Vid utveckling av Android appikationer finns det några komponenter en utvecklare bör göra sig välbekant med. De viktigaste är Activities, Intents, Services och Content Provider [3].

3.3.1.1 Activities

Allt som har ett användargränssnitt representeras av en så kallad Activity, eller aktivitet. Varje aktivitet har en egen livscykel och kan vara i fem typer av tillstånd. När man först startar en aktivitet försätts den i tillståndet "Starting" och anropar metoden onCreate() som initierar aktiviteten. En aktivitet är i tillståndet "Running" när den körs. onPause() anropas när en annan aktivitet skall startas och sätter den i läget "Paused". Aktiviteten är i läget "Stopped" när den inte skall användas på länge och minskar därav minnesupptagning. Tillslut när man vill döda aktiviteten så anropas onDestroy() och den hamnar då i läget "Destroyed" [1,3].



Figur 14. Activity livscykel

3.3.1.2 Intents

En Intent är ett meddelande som tillåter Android komponenter att begära funktionalitet från andra komponenter. Ett exempel på en Intent kan vara att en aktivitet startar en annan aktivitet [3].

3.3.1.3 Services

I Android finns det flera processer som kan köras i bakgrunden utan att användaren ser den. En sådan funktion kallas Service. En service kan vara ett program så som en musikkspelar som spelar upp musik och samtidigt kan användaren köra andra program utan att störa musikkspelaren [3].

3.3.1.3 Content Provider

Content Provider är ett sätt att dela global information mellan olika applikationer. Det är en klass som implementerar olika metoder för att ta hand om data som man vill skall vara tillgänglig för andra applikationer. Det skulle kunna vara data så som olika personers kontaktinformation. Det finns en standard Content Provider implementerad i Android systemet men vill man dela specifika data är det bästa sättet att göra en egen [3, 2].

3.4 ISO/IEC 9126

ISO/IEC 9126 är en internationell kvalitetsstandard för produktvärdering inom mjukvaruutveckling. Den är framställd av organisationen International Standardization Organisation som utvecklar patentskyddade, kommersiella och industriella standarder. ISO/IEC 9126 är uppdelad i fyra parter: "quality model, external metrics, internal metrics, quality in use metrics." [6]

3.4.1 Quality model

Quality model klassificerar kvaliteten på mjukvaran med hjälp av sex specificerade egenskaper: functionality, reliability, usability, efficiency, maintainability, portability. Varje egenskap har i sin tur underenskaper som detaljerat specificerar vad som skall uppnås [6].

3.4.1.1 Functionality

"A set of attributes that bear on the existence of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs." [6]

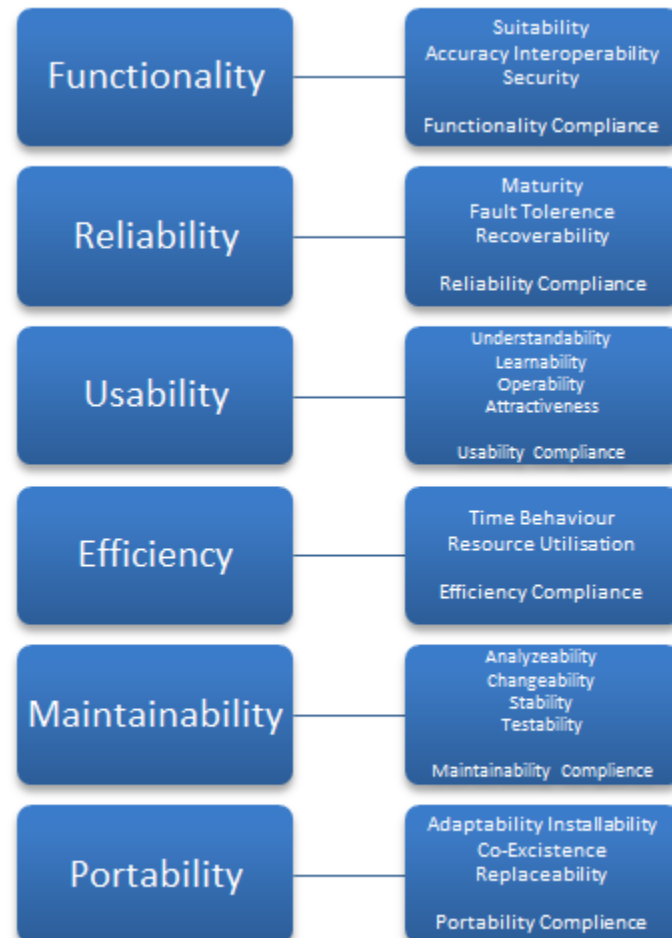
Sub-characteristics

Suitability - Kan mjukvaran utföra den krävda uppgiften?

Accuracy - Är resultatet som förväntat?

Interoperability - Kan systemet interagera med andra system?

Security - Är systemet intrångssäkert? [7]



Figur 15. Quality model, egenskaper samt underenskaper

3.4.1.2 Reliability

"A set of attributes that bear on the capability of software to maintain its level of performance under stated conditions for a stated period of time." [6]

Sub-characteristics

Maturity - Är de flesta felen borta?

Fault Tolerance - Kan mjukvaran hantera fel?

Recoverability - Kan man fortsätta med sitt arbete efter krasch utan att ha förlora data? [7]

3.4.1.3 Usability

"A set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users" [6].

Sub-characteristics

Understandability - Förstår användare hur systemet fungerar?

Learnability - Lär sig användaren fort hur system ska användas?

Operability - Kräver programmet stora kunskaper för användning?

Attractiveness - Ser systemet attraktivt ut? [7]

3.4.1.4 Efficiency

"A set of attributes that bear on the relationship between the level of performance of the software and the amount of resources used, under stated conditions." [6]

Sub-characteristics

Time Behaviour - Responstid?

Resource Utilisation - Använder sig programmet av systemets resurser effektivt? [7]

3.4.1.5 Maintainability

"A set of attributes that bear on the effort needed to make specified modifications." [6]

Sub-characteristics

Analyzeability - Kan fel hittas lätt?

Changeability - Kan man ändra i koden?

Stability - Efter ändringar gjorts, fungerar systemet ännu?

Testability - Kan mjukvaran testas lätt? [7]

3.4.1.6 Portability

"A set of attributes that bear on the ability of software to be transferred from one environment to another." [6]

Sub-characteristics

Adaptability - Kan systemet porteras till andra miljöer?

Installability - Är det lätt att installera programmet?

Co-Excistence - Fungerar systemet tillsammans med andra system?

Replaceability - Är andra system utbytbara med ditt system? [7]

3.4.2 Intern kvalitet, extern kvalitet och användarberoende kvalitet

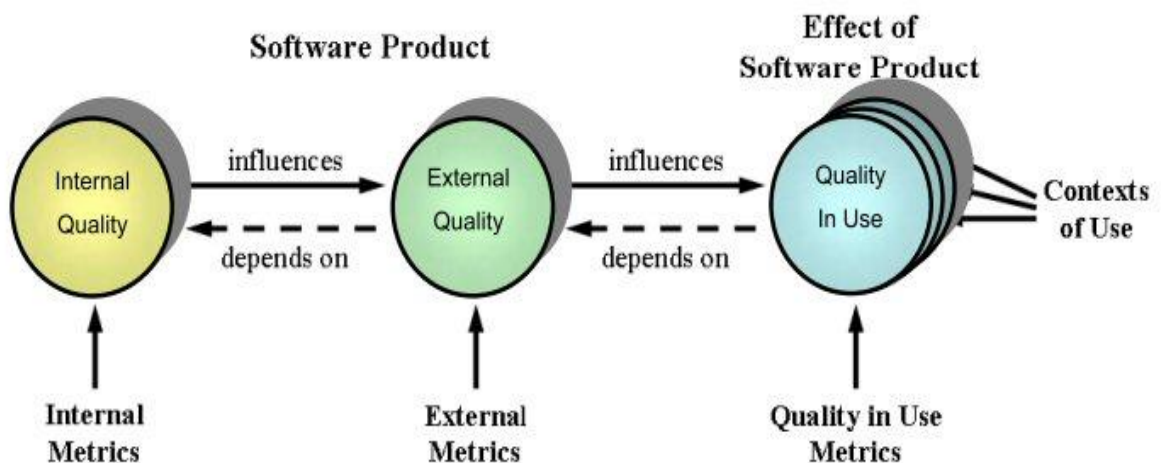
Intern-, extern-, användarberoende kvalitet är olika mätsätt att bestämma ett programs kvalitet.

Intern kvalitet - Mäts av kodens statiska egenskaper, oftast genom granskning av till exempel path length.

Extern kvalitet - Mäts av kodens dynamiska egenskaper vid exekvering, till exempel responstid.

Användarberoende kvalitet - Mäts genom till vilken grad programmet uppfyller användarens behov[8].

Mätsätten är alla del av kvalitetsmodellen men representerar olika egenskaper av den. De beror alla av varandra vilket visas i figur 18. Om koden är dåligt konstruerad påverkas de externa egenskaperna vilket i sin tur påverkar användarens upplevelse av programmet[8].



Figur 16. De olika mätsättens påverkan på varandra.

4.0 Design och Implementering

Arbetet med applikationen är uppdelat i två större parter, Android och Kommunikation. I det här avsnittet beskriver vi hur vi utvecklat och designat vår applikation.

4.1 Android

4.1.1 Design

Med erfarenheter från tidigare arbeten om hur vi enklast skall utveckla en applikation har vi upptäckt att det är bäst att sätta sig tillsammans för att diskutera vad som behövs och rita upp hur vi vill att det skall se ut. Vilka funktioner och funktionaliteter som måste finnas med. En början till utseende och funktionalitet hade vi diskuterat fram tillsammans med Kramer och experter inom styrsystemsinstallation sedan tidigare. Utefter det fick vi göra några få avgränsningar gentemot vår kunskap och tidsram. Med samlad information kom vi fram till att applikationen skall kunna styra de vanligaste enheter som finns i ett konferensrum vilket innefattar: dvd, tv, projektor, projektorduk, belysning, mörkläggnings, ljudkontroll, datorhantering. Det vi fokuserade mest på i början var att skapa ett simpelt grafiskt gränssnitt och kontroller för nämnda enheter. Nästa steg var uppkoppling mot fjärrstyrningssystemet. Detta går vi igenom mer ingående i nästa del av kapitlet. När detta var klart så åkte vi till Kramer för testning.

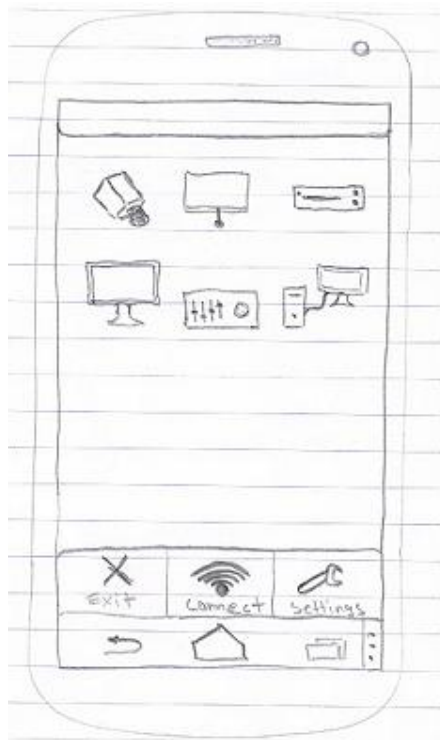
Fjärrstyrningsmomentet var nu klart. Nu skulle funktionaliteten för olika scenarion implementeras.

För Kramers system kan man välja att ha en inloggning för autentisering så en inloggningsfunktion krävdes. Som användare vill man inte ha onödiga knappar till enheter som man inte kan styra och för att lösa det skapade vi en "Device setup". Här väljer användaren själv vilka enheter som finns i systemet.

Därefter implementerade vi en ljudkontroll som man alltid kan komma åt var man än befinner sig i applikationen.

De flesta komponenter vi har använt oss av är standardkomponenter i Androids system. Det var ett fåtal som vi behövde läsa på mer om för att göra skräddarsydda komponenter som passade vårt syfte.

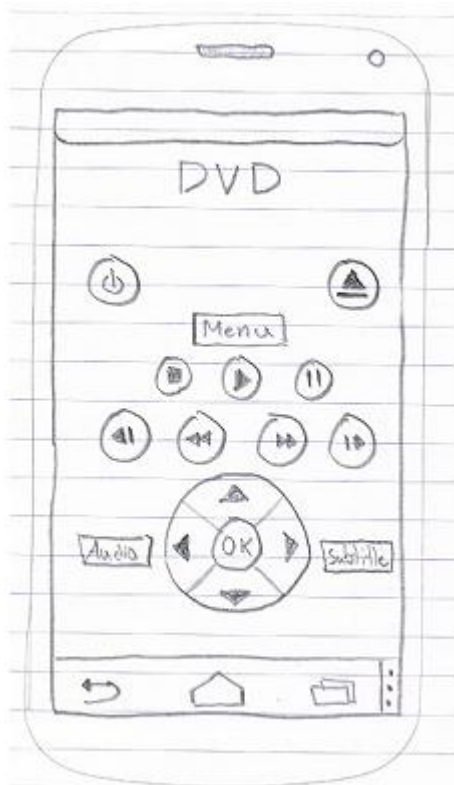
I figurerna nedan visas några av de tänkta bilderna till vänster och hur de tillslut blev till höger.



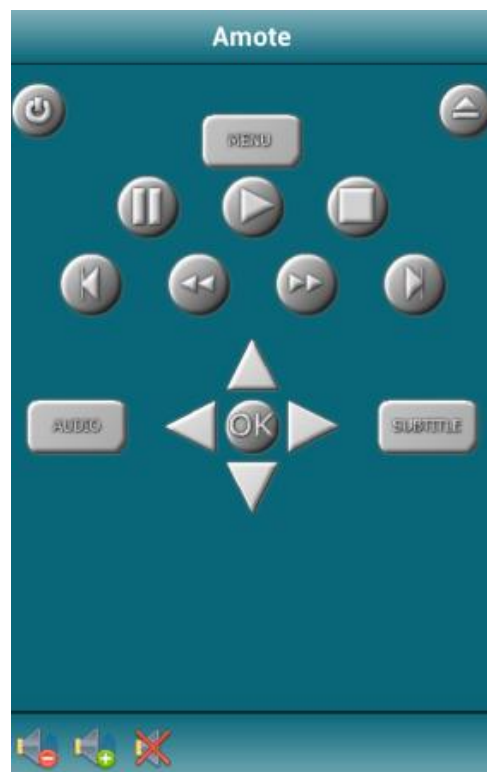
Figur 17. Planerad layout för ikonlistan.



Figur 18. Ikonlistans slutresultat.



Figur 19. Planerad layout för DVD.



Figur 20. DVD kontrollens slutresultat.

4.1.2 Implementering

4.1.1 Implementering av "Device setup"

Idé

Till en början hade vi ikoner för alla enheter som vår applikation kan styra. Men sen kom vi på att det kan vara förvirrande för användaren och införde då en "Device setup" där man får välja vilka enheter man har kopplat in i systemet som sedan ritas upp, se figur 21.

Funktion

På denna delen av applikationen får användaren välja genom att kryssa i vilka enheter som är inkopplade i styrsystemet. På vissa av enheterna finns ytterligare funktioner. Om man till exempel har datorer inkopplade så får man även välja hur många. När användaren är klar och trycker sig vidare i applikationen så skapas ikoner för styrning av de valda enheterna på nästa sida.

Implementation

Det var inte så smidigt att implementera denna funktion som vi hade förväntat oss. Standardklasser för att göra denna typ av "Settings" sida finns klart i Android. Vi behövde dock skraddarsy den något för att det skulle fungera som planerat. När man trycker på en enhet i listan skall tillhörande kryssruta fyllas i. Detta gäller för alla enheter. På vissa skall det även öppnas ytterligare en funktion där man ställer in hur många datorer man har till exempel. Detta löste vi genom att titta på exempel och läsa på mer om hur Adapters och Keylisteners fungerar.

4.1.2 Implementering av ikonplacering

Idé

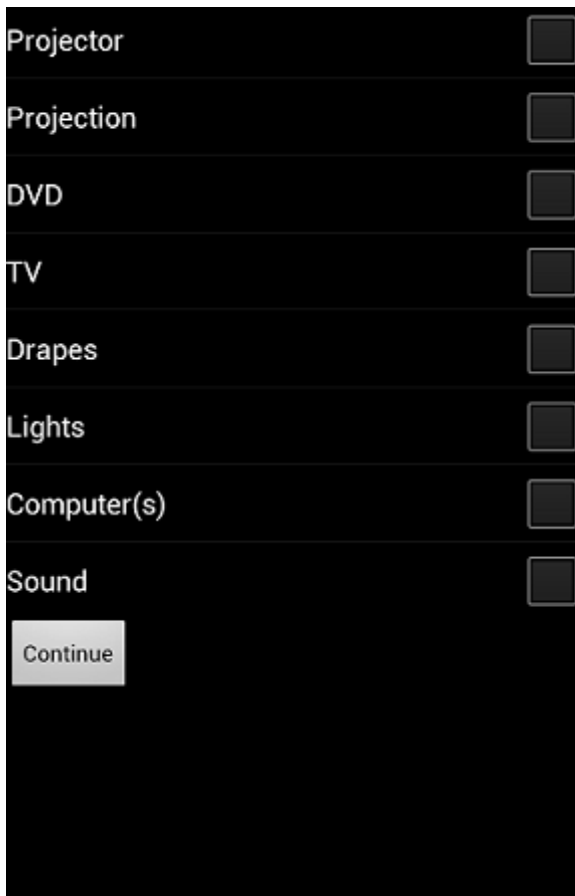
Efter att man valt vilka enheter som är inkopplade till systemet ritas respektive ikon upp. Dessa går sedan att trycka på för att komma till deras aktiviteter och sedan styra dem.

Funktion

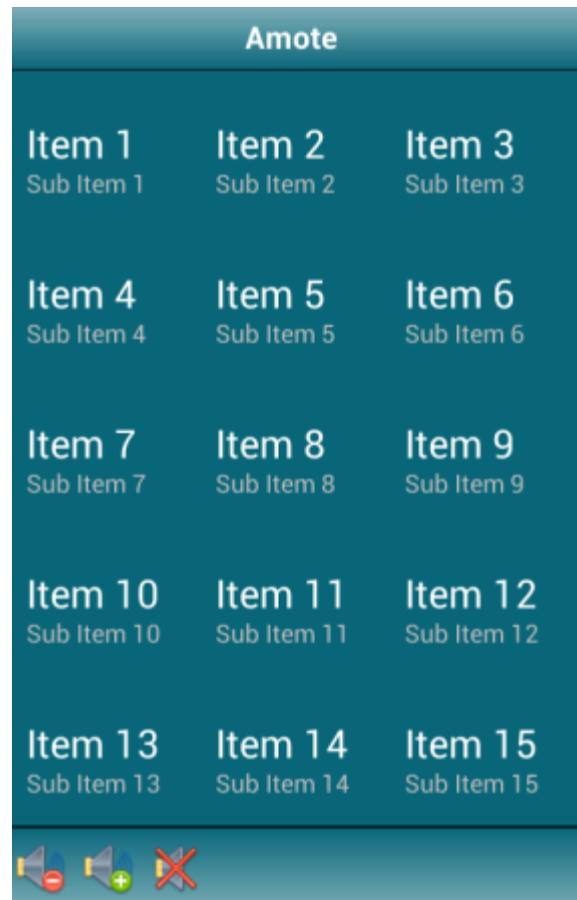
Varje vald enhet ritas upp som knappar, en efter en. I figur 22 visas en lista innan den fylls med knappar till de enheter man valt. De resterande platserna förblir tomma utan texten som syns i figuren. Dessa går sedan att trycka på för att komma till respektive aktivitet.

Implementation

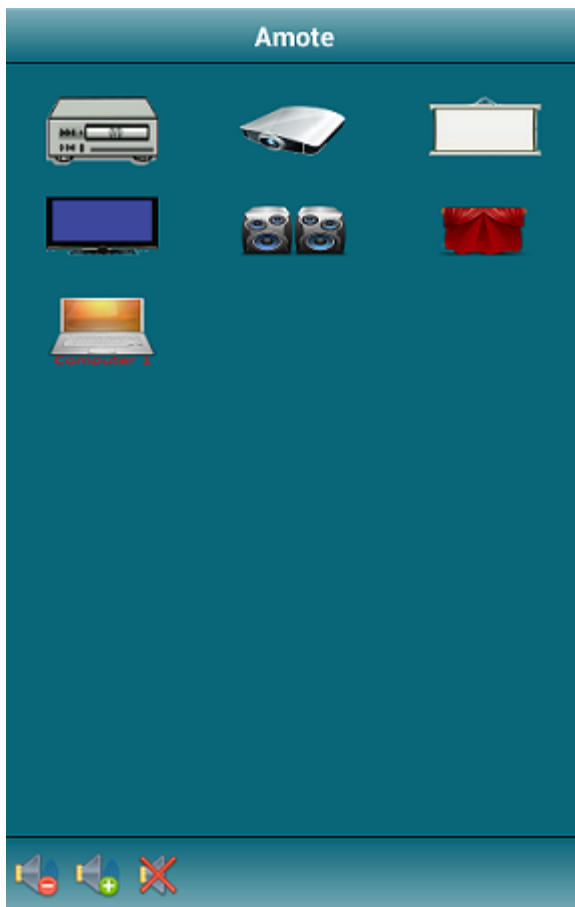
I början hade varje enheten en specifik plats. Men vi insåg snabbt att detta blev väldigt fullt när inte alla enheter var valda. Ikoner skapades då med olika stora mellanrum till varandra. Vi valde då att istället använda en GridView som vi fyllde med knappar till de enheter man valt. GridView var något vi var ganska osäkra på och fick även här göra en skraddarsydd Adapter och Keylistener.



Figur 21. "Device setup". Enheter man kan lägga till.



Figur 22. GridView. Listan på enheter innan de ritats ut.



Figur 23. De valda enheterna utritade i en lista.



Figur 24. Styrknappar för en DVD.

4.1.2 Implementering av "Device managing"

Idé

När man trycker på någon av de uppritade ikonerna så kommer man till dess aktivitet för styrning av motsvarande enhet. Beroende på vilken enhet det är så finns det olika styrknappar. Som framgår av figur 24 så har DVD aktiviteten knapparna: play, pause, stop, fast forward, rewind, next, previous, menu, power, knappar för navigering, audio, subtitle, input, eject.

Funktion

Varje enskild knapp i alla enheters aktiviteter skickar ett unikt kommando med hjälp av "Command send helper" till styrsystemets centralenhet som sedan utför kommandot.

Implementation

När vi implementerade dessa knappar så var strukturen för kommunikationen mellan klient och server redan byggd så detta gick snabbt och enkelt. Vi har valt att inte implementera alla knappar som existerar för de olika enheterna utan istället ta med de vanligaste knapparna som används. Detta för att minska antalet knappar att rita upp på GUI:t och underlätta för användaren.

4.1.3 Implementering av ljudkontroll

Idé

Oftast så har man någon form av ljudkälla i sitt styrsystem men istället för att trycka på en ikon till en ljudenhet så införde vi en central ljudkontroll som är tillgänglig i alla vyer.

Funktion

Denna ljudkontroll ligger som en "bottom bar" i applikationen och innehåller 3 knappar. En för att höja ljudet, en för att sänka och den sista för att stänga av och sätta på ljudet.

Implementation

Här stötte vi inte heller på några större hinder. Allt som behövs här är en fungerande kommunikation mellan telefon och centralen för styrsystemet. Att hålla ljudkontrollen central och alltid tillgänglig underlättar användandet.

4.2 Kommunikation

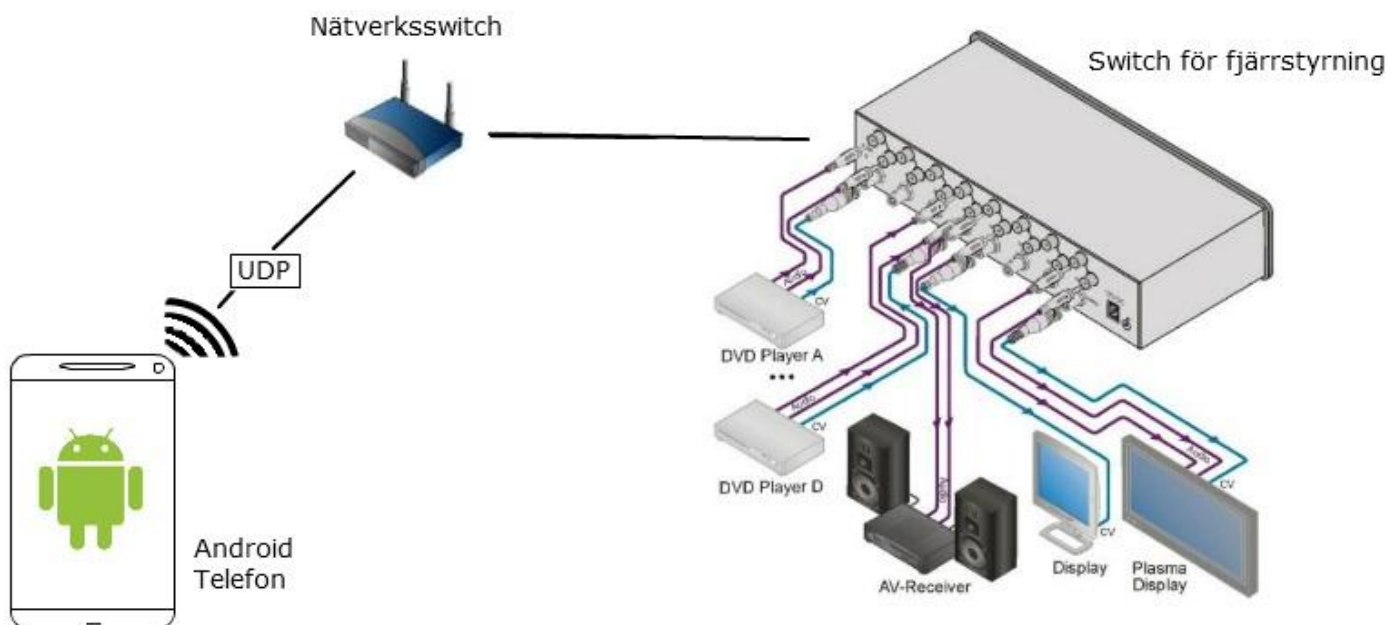
4.2.1 Design

När vi skulle utveckla kommunikationen mellan Android applikationen och switchen läste vi först den specifikation vi fått av Kramer om hur det skall vara uppbyggt. Sedan diskuterade vi även med Kramer för att få en djupare förståelse för hur det fungerar. Det protokoll som var bäst lämpat och skulle användas i systemet var UDP. Vi konstruerade kommunikationsdelen oberoende av resten av applikationen för att enkelt kunna byta ut protokollet om det skulle bli aktuellt.

I switchen som Kramer utvecklar kopplar man in de enheter man vill styra, till exempel en projektor.

För att enheter och knappar skall fungera måste switchen konfigureras, vilket görs med Kramers programvara K-config. Där specificerar man vad som är anslutet till switchens portar och vad Android applikationens knappar ska göra. Knapparna adresseras sedan genom knappens k-config nummer som är mellan 1-60.

Android telefonen kommer att skicka textsträngar som innehåller knapp-id med transportprotokollet UDP över trådlöst nätverk. Om meddelandet som switchen tar emot godkänns utförs kommandot och ett "SUCCES" meddelande skickas tillbaka till telefonen. Skulle meddelandet vara felaktigt eller misslyckas skickas ett felmeddelande tillbaka.



Figur 25. Kommunikationen från telefon till switch.

4.2.2 Implementering

4.2.2.1 Implementering av "Communication"

Idé

För att applikationen ska vara så generell som möjligt är Communication klassen oberoende av applikationens andra delar.

Funktion

Funktionen av denna klass är att skapa en kommunikation mellan telefon och switch. IP adress och port har användaren tidigare fyllt i för att skapa en anslutning. Denna klass skickar även iväg kommandon till centralen som användaren vill utföra.

Implementation

Implementeringen av denna klass gick väldigt bra. Det var en lätt klass att implementera då man visste precis vad den ska göra. Att den skulle vara generell skapade inga som helst problem utan gjorde snarare klassen mindre och mer specifik vilket förenklade utvecklingen.

4.2.2.2 Implementering av "Command send helper"

Idé

En hjälpklass till Communication klassen som skapar kommandon i form av strängar som ska skickas till switchen.

Funktion

Efter att användaren valt att utföra ett visst kommando skickas det in i denna klass för att sedan vidarebefordra rätt sträng till Communication klassen. Command send helper klassen får in vilket kommando som valts och rätt enhet letas upp och en specificerad sträng vidarebefordras till Communication klassen.

Implementation

När vi implementerade denna klass var vi klara med vad alla knappar i de olika aktiviteterna skulle skicka för kommandon vilket gjorde skapandet av denna klass väldigt enkelt. Med tiden blev klassen större då fler knappar kom till.

5.0 Evaluering

Evaluering av detta arbete kommer ske med ISO standarden ISO/IEC 9126. Vi valde att göra en evaluering utefter denna standard med avseende på att den är internationell och en av de mest välkända ISO standarder som används idag. Med hjälp av denna kommer vi kunna utvärdera vår prototyp från olika aspekter och med resultatet veta hur vi skall förbättra den.

Innan evalueringen utfördes en utvärdering av produktens viktiga egenskaper. Med detta som bakgrund valde vi att lägga störst fokus på kvaliteten hos själva produkten istället för kvaliteten hos datahanteringen i produkten.

Vi realiserade kvalitetsbedömningen i två moment. Först i form av ett experiment där vi testar applikationen på personer som är vana och personer som är ovana vid att använda en Android telefon. Sedan gjordes en intern testning för datahantering i telefonen.

5.1 Evalueringsattribut

I denna delen har vi gjort ett urval av de kvalitetsattribut som är mest relevanta för produkten. De är valda utifrån företagets och våra egna slutsatser och diskussioner om vad som bäst lämpar sig för en mobilapplikation. De egenskaper som bedömningen utförs utefter är: Functionality, Usability, Reliability och Portability.

5.1.1 Functionality

Functionality är möjligheten för produkten att bidra med funktioner som möter de bestämda behoven när den används under speciella förhållanden [12].

Valda underregnskaper: Suitability, Security

Suitability

Målet med prototypen är att få den att fungera för de beslut vi gjorde i början av arbetet. Därför anser vi att denna underregnskap är bra att ta med i evalueringen för att se så att vi levt upp till dessa mål.

Security

Säkerhet är en viktig underregnskap för alla produkter som skall ha åtkomst till personlig information och data. Det är viktigt att applikationen tar hänsyn till switchar där inloggning krävs så att inte otillåten anslutning kan inträffa.

5.1.2 Usability

Usability är möjligheten för användare att förstå, lära sig och gilla produkten under speciella förhållanden [12]. Detta är den viktigaste egenskapen hos prototypen därför är alla dess underegenskaper med i bedömningen av kvalitén.

När man designar en applikation till telefoner så finns tre hot som påverkar Usability egenskapen.

- Tangentbordets storlek: Om tangentbordet är väldigt litet kan inmatning av text bli svårt och tidskrävande.
- Skärmstorlek: Olika telefoners skärmstorlekar och upplösning kan göra att det grafiska gränssnittet skiljer sig något och gör det svårt för användaren att se och manövrera i applikationen.
- Nätverksåtkomst: Eftersom produkten måste vara uppkopplad mot samma nätverk som fjärrstyrningssystemet så kan störningar i nätverket skapa förhinder.

Med hänsyn till detta utformade vi ett experiment för att testa Usability på bästa sätt, vilket beskrivs i nästa del nedan.

5.1.3 Reliability

Reliability är möjligheten för mjukvaran att upprätthålla sin prestationsförmåga vid användning under speciella förhållanden [12].

Valda underegenskaper: Recoverability

Recoverability

I vårt fall har datan som hanteras inte någon större betydelse. Men det är viktigt att kunna fortsätta arbeta efter att en krasch av till exempel switchen inträffat.

5.1.4 Portability

Portability är en uppsättning egenskaper som gör systemet överförbart till andra miljöer.[12]

Vi vill att systemet skall fungera till flera olika styrsystem vilket gör detta till en viktig egenskap. Men för detta krävs mer arbete.

Valda underegenskaper: Replaceability, Adaptability

Replaceability

Kan vårt system ersätta ett annat system. Ofta används stationära styrpaneler för styrning av styrsystem. Det vi vill med vår applikation är att ersätta dessa för att förenkla användandet.

Adaptability

Kan systemet användas till andra styrsystem. Målet var att skapa en prototyp som går att fortsätta utveckla för olika typer av styrsystem.

5.2 Experiment

5.2.1 Testgrupp

Våra testpersoner är uppdelade i två grupper: vana och ovana Androidanvändare. Grupperna kommer att bestå av ca 4-5 personer. Tanken är att vi ska få ut så mycket information om användarvänligheten hos applikationen som möjligt. Innan testet börjar kommer alla få en kort genomgång av Androids kortkommandon för att testet ska vara jämnt. Speciellt så att de ovana Androidanvändarna får reda på vilka kortkommandon som existerar.

5.2.2 Testmoment

Testmomentet kommer att utföras i tre steg.

1. Personen får en kort genomgång av Android samt vad det är för applikation som presenteras. Personen får även en text med givna uppgifter att utföra.
2. Personen blir bedd att utföra givna uppgifter.
3. Efter varje uppgift intervjuar vi personen angående svårigheter med förståelse och utförande.

Testuppgifter

Uppgift 1 - Anslutningstest

I denna testsituation skall testpersonen starta applikationen, klicka sig vidare till anslutningsskärmen och sedan ansluta till angiven adress och logga in med tillhandahållen inloggningsinformation. Figur 26 och 27 tillhör denna uppgift

Uppgift 2 - Enhetstest

I denna testsituation befinner man sig på device setup aktiviteten. En lista tillhandahålls om vilka enheter som skall användas i systemet för att få tillförlitliga mätvärden. När testpersonen satt upp vilka enheter som skall vara med går denna vidare till huvudskärmen med uppritade enheter. Testpersonen skall sedan gå in på en specificerad enhet och utföra ett kommando. Figur 28, 29 och 30 tillhör denna uppgift.

Uppgift 3 - Ljudtest

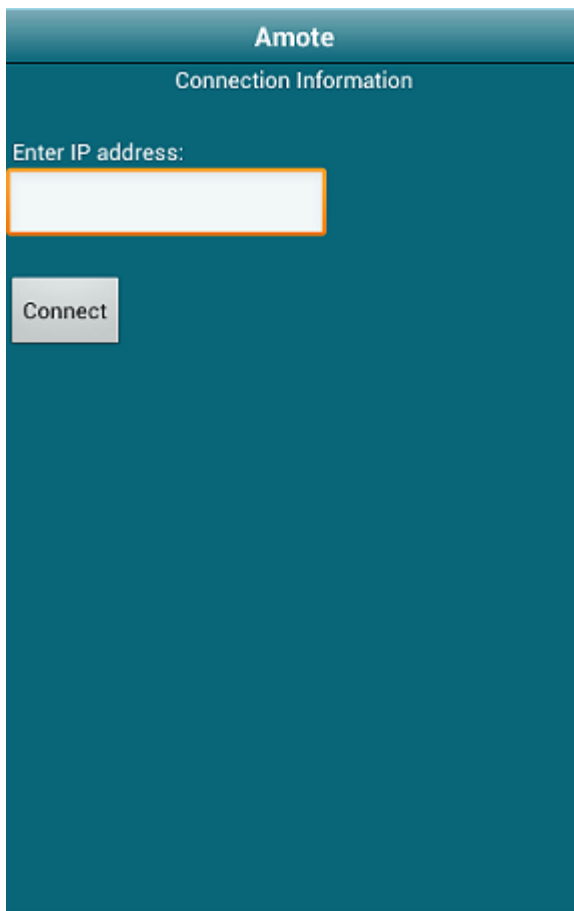
I denna testsituation börjar testet där uppgift 2 slutade. Volymen skall höjas på specificerad enhet vilket utförs genom den globala volymkontrollen. Då volymen höjts skall testpersonen gå in på inställningssidan via menyknappen och byta den globala ljudkontrollen till en annan enhet. Figur 31 och 32 tillhör denna uppgift.

Uppgift 4 - Ändra enheter

I denna testsituation skall testpersonen från avslutad skärm i uppgift 3 gå tillbaka till device setup aktiviteten och ändra enheter enligt anvisningar. Sedan gå vidare till huvudskärmen och gå in på en enhet och utföra ett kommando. Denna uppgift är i princip likadan som *Uppgift 2*. Det vi testar här är om personen förstår att man kan gå tillbaka och ändra de enheter man har inkopplade.

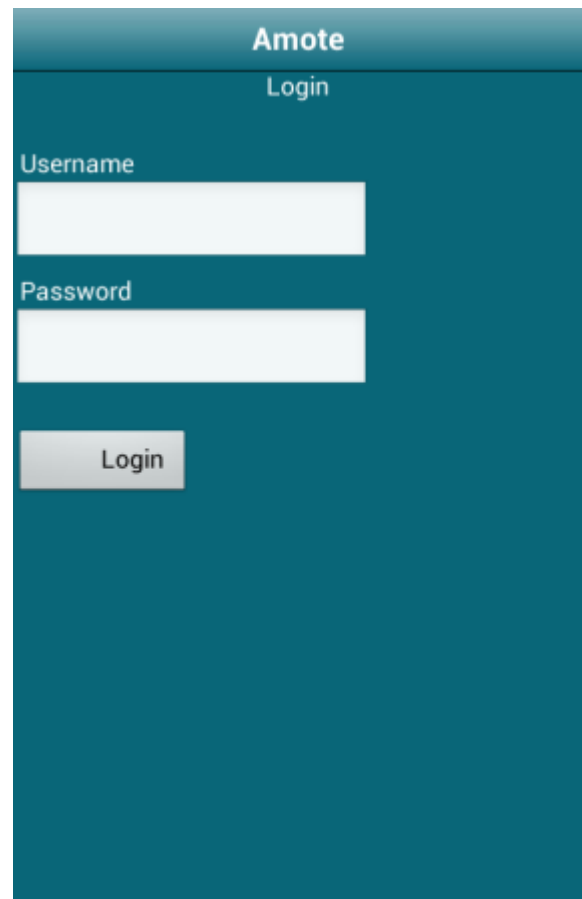
Uppgift 5 - Avsluta applikationen

Till sista skall testpersonen avsluta applikationen genom menyknappens exit-funktion. Figur 31 tillhör denna uppgift.



The screenshot shows the 'Amote' application interface for connection. At the top, the title 'Amote' is displayed. Below it, the section is titled 'Connection Information'. There is a text prompt 'Enter IP address:' followed by a white text input field with an orange border. Below the input field is a grey button labeled 'Connect'.

Figur 26. Grafiska gränssnittet för anslutning.

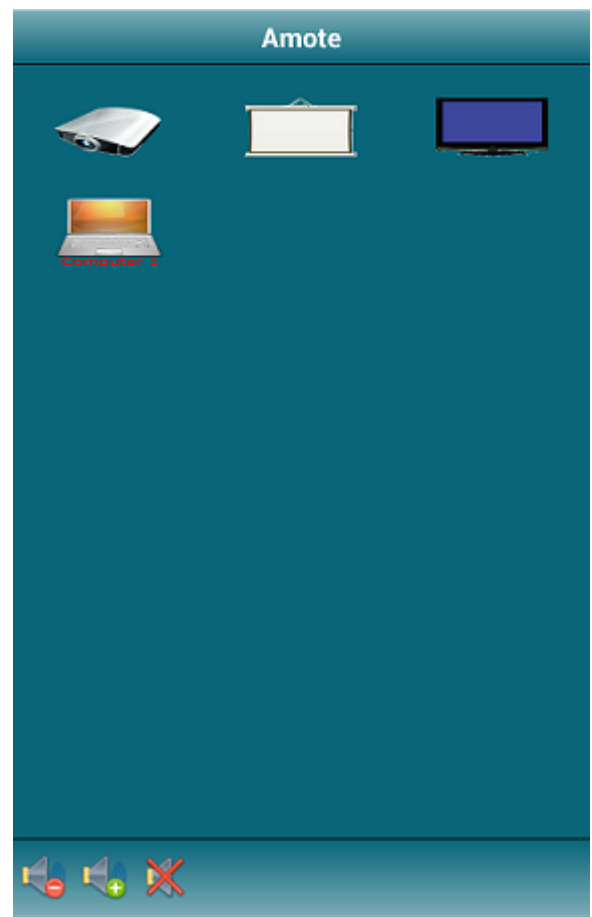


The screenshot shows the 'Amote' application interface for login. At the top, the title 'Amote' is displayed. Below it, the section is titled 'Login'. There are two white text input fields: the first is labeled 'Username' and the second is labeled 'Password'. Below the password field is a grey button labeled 'Login'.

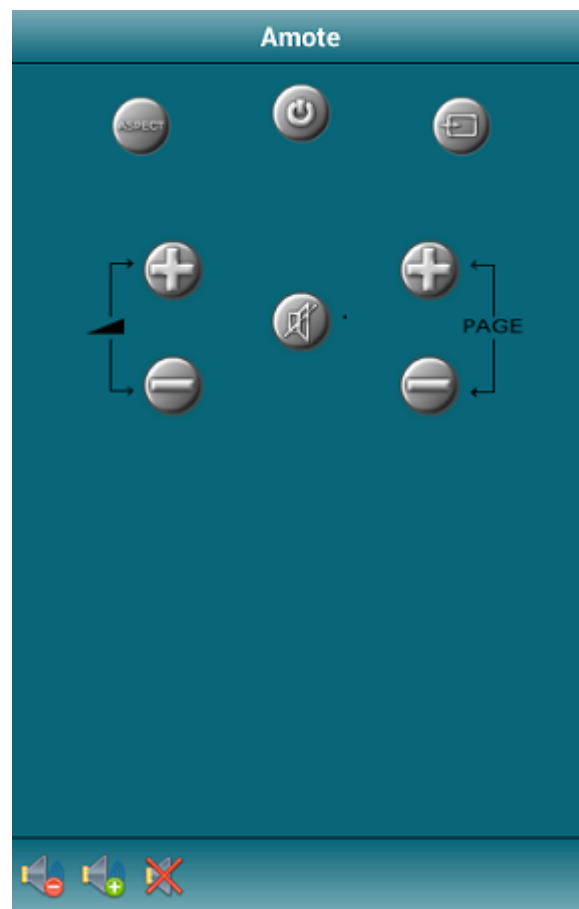
Figur 27 Grafiskt gränssnitt för inloggning



Figur 28. Några valda enheter i "Device setup".



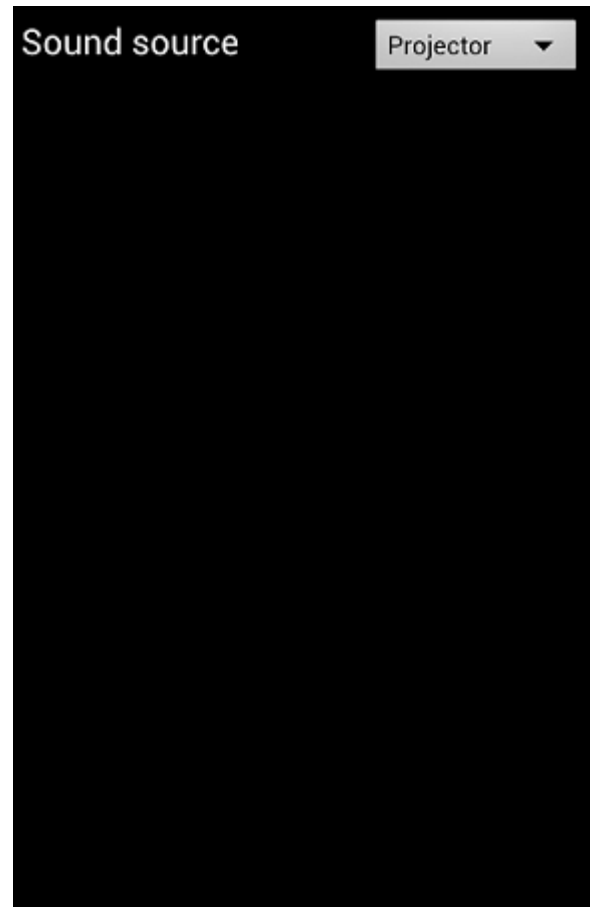
Figur 29. De valda enheterna uppritade i en lista.



Figur 30. TV kontrollen. Här skall ett kommando utföras.



Figur 31. Android-menyns 3 olika valmöjligheter.



Figur 32. Inne på Android-menyns "Settings" sida.

5.0 Testresultat

I detta testresultat presenterar vi hur evaluering av vår prototyp med standarden ISO/IEC 9126 gick till. Med de valda egenskaperna Functionality, Usability, Reliability och Portability och några av dess underenskaper har vi utformat ett antal olika testfall. Detta för att testa så att prototypen lever upp till våra krav i kravspecifikationen. Resultatet är uppdelat i två mindre delar. Utvecklaraspekt beskriver de test som vi själva endast kan utföra medan i avsnittet Användaraspekt tar vi upp hur vårt utförda experiment gick.

5.1 Utvecklaraspekt

Functionality

Suitability:

För att testa Suitability gjorde vi en jämförelse mellan de funktioner vi från början hade som mål och de funktioner vi nu har åstadkommit.

De funktionella mål vi satte upp för applikationen var:

Den skall kunna kontrollera ett fjärrstyrningssystem.

Enheter som prototypen skall kunna styra:

DVD, TV, projektor, projektorduk, ljudanläggning, datorer, ljus, mörkläggningsgardiner.

Prototypen har uppnått dessa mål och därmed klarat av Suitability testet.

Security:

På switchen i fjärrstyrningssystemet kan man ställa in ett inloggningskrav. Om inte detta är satt så kan vem som helst som befinner sig i det lokala nätverket få åtkomst till switchen. Det är så deras produkt är utvecklad. Så säkerhetstestet vi utförde var om man kunde komma åt fjärrstyrningssystemet, när inloggning krävdes, när man angav fel inloggningsuppgifter.

Vi gjorde 10 testfall med olika användarnamn och lösenord. Bara det korrekta användarnamnet och lösenordet fick åtkomst till fjärrstyrningssystemet. Denna funktion fungerade med andra ord utan konstigheter och får därmed egenskapen Security godkänd.

Reliability

Recoverability

För att testa underenskapen Recoverability behövde vi utföra någon form av avbrott under körning av applikationen. Det lättaste sättet att skapa en sådan situation var att helt enkelt genom att stänga av switchen. Då försvinner anslutningen mot den. Sedan satte vi på den igen för att se om vi fortfarande kunde styra något. Resultatet av detta test var negativt. Vi kunde inte efter ett sådant avbrott fortsätta styra. Detta för att switchen i systemet kräver först en form av "handshake" för att tillåta åtkomst. Man måste med andra ord börja om från början i applikationen och skapa en ny anslutning till switchen.

Vi anser dock att detta inte är något större problem då det inte är särskilt svårt att starta om applikationen och återansluta. De senaste enhetsinställningarna sparas vilket gör det lätt för användaren att åter komma in i sitt arbete.

Portability

Replaceability

För test av utbyttbarheten körde vi först ett styrsystem med en stationär styrpanel och testade så att allt fungerade. Vi bytte sedan ut styrpanelen och ersatte den med vår applikation. Sedan utförde vi samma tester som gjordes med testpanelen och konstaterade att alla funktioner fungerade på samma sätt.

Resultatet av våra tester för utbyttbarheten är att applikationen kan ersätta det gamla sättet att styra vilket var det tänkta målet.

Adaptability

Som tidigare nämnt är applikationen tänkt att kunna styra flera olika fjärrstyrningssystem. Dock är detta en prototyp som fungerar till endast ett sådant. Men i framtida versioner ska den fungera till fler.

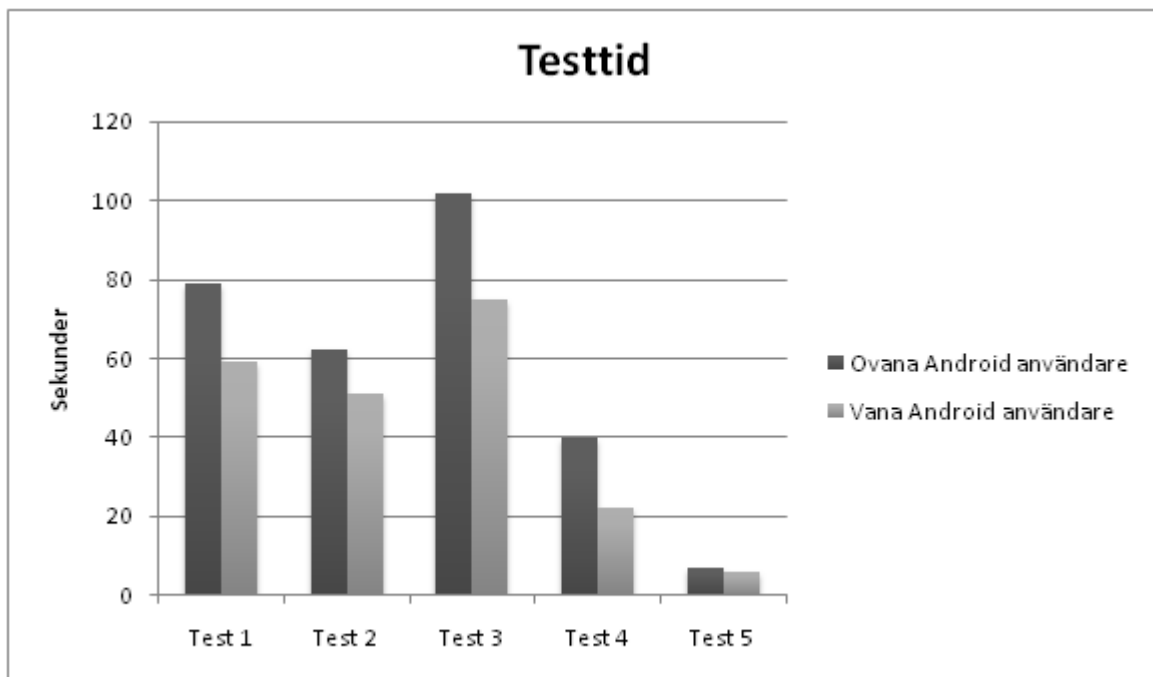
Vi försökte komma i kontakt med ytterligare ett företag liknande Kramer men utan resultat. Därav blev det endast en typ av fjärrstyrningssystem och egenskapen Adaptability underkänd.

5.2 Användaraspekt

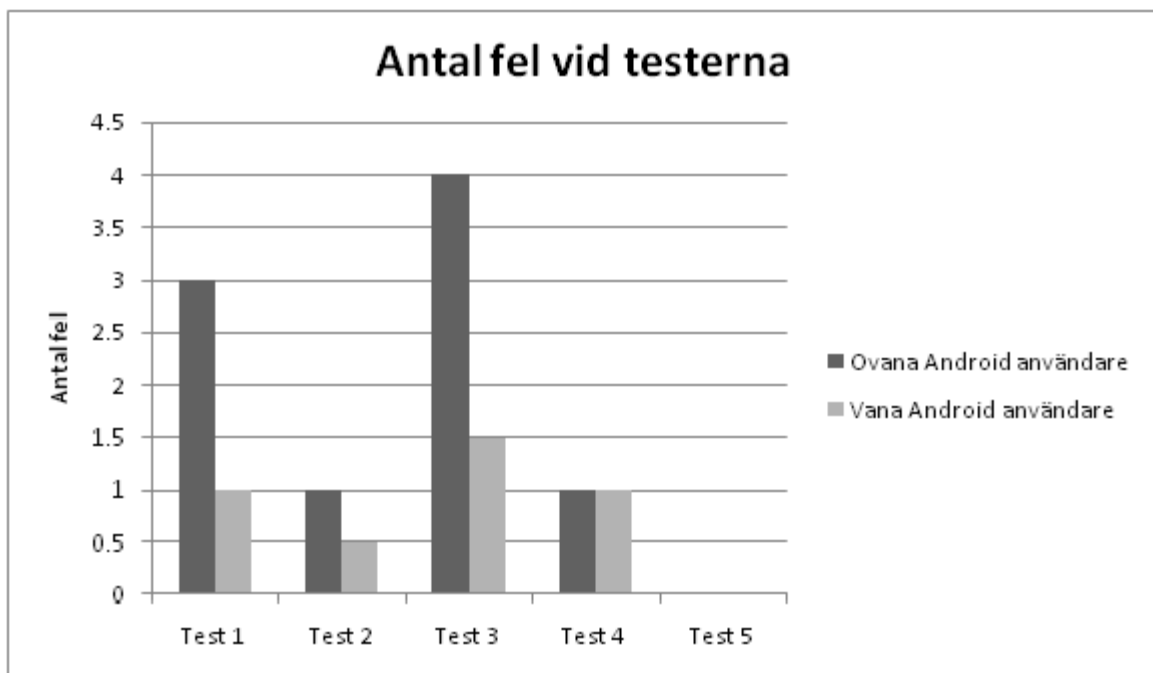
Vi hade kommit fram till att egenskapen Usability behövde testas genom att låta olika personer testköra applikationen. Som tidigare nämnts utformade vi fem olika typer av uppgifter för att genomgå alla de olika moment i applikationen som kan uppstå vid användning. Vi studerade testpersonernas sätt att ta itu med uppgiften och efter varje uppgift intervjuade vi dem om saker de uppfattade som svåra och även om möjliga förslag till förbättringar.

Vi arbetade med två olika typer av grupper. Den ena gruppen med fem vana Android användare och den andra med fem ovana Android användare. Detta gjorde vi för att få två olika synfall på prototypen och därav förbättra evalueringen. De mätvärden utöver intervjun vi tog tillvara under testfasen var de antal fel som gjordes vid varje uppgift och tiden det tog att utföra den. I det tidigare delkapitlet 5.2 *Experiment* beskrivs specifika uppgifter som varje testperson utförde.

I figur 33 och 34 visas testresultaten vi fick från experimentet.



Figur 33 Genomsnittligt antal sekunder att utföra testerna för testpersonerna.



Figur 34 Genomsnittligt gjorda fel under testerna.

Från dessa figurer kan vi se att uppgift 3 var svårast. Vid intervjuerna med personerna efter uppgiften fick vi reda på varför.

Testpersonerna hade svårt att hitta volymkontrollen på grund av att den var för liten. Vägarna att gå för att byta källan till sin ljudanläggning var inte uppenbar för de ovana användarna.

Detta tillsammans ledde till många felaktiga försök och lång tid.

Den första uppgiften var näst svårast. Vissa av testpersonerna förstod inte att man skulle trycka på Kramer- ikonen för att komma vidare. Istället försökte de ta andra vägar genom till exempel Androids menyknapp. Det som tog längst tid i den uppgiften var att skriva in all textinformation som behövdes för att ansluta och logga in. Detta hade vi räknat med och kan variera stort beroende på vilka telefon man har. Telefoners tangentbord finns i många olika storlekar som kan göra det svårt att trycka på rätt tangent.

Uppgift 4 och 5 var väldigt enkla och medförde inga större problem för användarna. Det enda märkbara var hos de ovana användarna. Androids "back"-knapp behövde användas i denna uppgift men de tänkte inte på att denna fanns, så de hittade inte vägen tillbaka till enhetshanteraren.

Uppgift 2 var den enklaste att utföra. Personerna tyckte den var väldigt rakt på sak. Det handlade egentligen bara om att läsa uppgiften ordentligt.

5.3 Diskussion

Efter denna testfas har vi kommit fram till att vår applikation är på god väg mot att bli en generell applikation. ISO standarden kände vi var en mycket bra kvalitetsstandard att följa om man vill utveckla en generell produkt. ISO/IEC 9126 finns tillgänglig på Internet men den finns även att köpa för att få djupare kunskaper om den. Vi hade lite problem med att hitta information om det fanns någon guide att följa för att testa de olika egenskaperna. Men utifrån undersökning på andra produktevalueringar fick vi den uppfattningen att man i princip hittade på egna typer av tester som relaterar till vad själva egenskapen handlar om. Detta gjorde att vi lätt kunde utföra relevanta och bra tester på applikationen. Den grund till att vi ansåg att dessa tester faktiskt var acceptabla låg mestadels på den forskning på hur andra produktevalueringar har utförts. Med tanke på att detta är en stor internationell kvalitetsstandard så kan vi med resultatet säga att vi har lite kvar att förbättra för att få den generell men vi är på god väg att lyckas med det. Det enda testet vi är missnöjda med är porteringsmöjligheten. Det faktum att vi inte har implementerat ytterligare ett företag. Anledningen till det är att vi inte lyckats komma i kontakt med något annat företag än just Kramer. Hade vi bara fått beviset att 1 till företag lyckats implementeras så hade även detta test blivit godkänt. Dock skall själva implementeringen inte var några svårigheter då det enda som egentligen behövs är kunskap om hur fjärrstyrningssystemet fungerar, på vilket sätt skall man kommunicera med företagets centralenhet och vilka koder används. Grundkomponenterna för själva styrningen finns redan i applikationen för återanvändning.

Vid intervjuerna under experimentet fick vi en del förslag på ändringar som kan göras för att förbättra egenskapen Usability på prototypen. Grafiska komponenter är något man hela tiden kan ändra på, därför har inte så mycket tid lagts ner på den biten. Det är en sak som vi från början bestämde oss för att inte fastna i.

Här följer en lista med förslag från testpersoner på funktioner och metoder som kan ändras på:

- Grafiska komponenter såsom ikoner, färgteman, menyer etc.
- Namn under enheternas ikoner för underlätta för användaren(den ikonen vi har ser kanske inte ut som deras enhet de har i hemmet).
- Lägga till en knapp för att återvända till enhetshanteraren under Androids menyknapp.
- Förtydliga volymkontrollen(större ikoner som är lättare att se och trycka på)
- Möjligheten att kunna ställa in fler ljudfunktioner.
- Implementera automatiska punkter vid inmatning av IP adress.
- Lägga till en knapp för avstängning i form av en kryssruta.

Övriga delar vi inte har hunnit med under arbetet samt framtida implementeringar:

- Skapa hemsida för nedladdning av K-konfig filer till Kramers switchar.
- Utöka enhetsutbudet som applikationen kan styra.
- Tillämpa fler företags fjärrstyrningssystem.

6.0 Slutsats

Med en växande marknad inom mobiltelefoni och fjärrstyrning har en nytt sätt för att fjärrstyra uppstått. Smartphones blir mer och mer populära i hemmet samt fjärrstyrningen kan styra i princip allt man har. För att inte förbruka onödigt mycket resurser och skapa barriärer mellan olika styrsystem har det här examensarbetet fokuserat på att framställa en kvalitetsevaluerad prototyp i form av en Android applikation för fjärrstyrning av elektroniska enheter.

6.1 Kritisk diskussion

Huvudsyftet med arbetet var att designa och utveckla en prototyp för en generell applikation med målet att sedan evaluera prototypen. Den första prototypen utvecklades för att kunna fjärrstyra enheter som kan existera i ett konferensrum. Fjärrstyrning fungerar för enheterna: DVD, TV, projektor, projektorduk, ljudanläggning, datorer, mörkläggningsgardiner. Evaluering utfördes enligt ISO/IEC 9126 kvalitetsmodell. Efter noggrann genomgång av prototypens kvalitéer gjordes ett urval av egenskaper från ISO/IEC 9126 kvalitetsmodell. Från diskussioner med vår industriella partner Kramer kom vi fram till att Usability är den mest intressanta och viktigaste egenskapen. För att undersöka denna egenskap har vi utfört ett experiment där vi testat applikation på ett antal personer. Efter testerna intervjuades varje enskild person angående eventuella brister och svårigheter.

6.2 Fortsatt forskning

Med resultaten från evalueringen har vi fått en vägledning om vad vi kan ändra och förbättra på applikationen i framtiden. För ytterligare utveckling av applikationen skulle GUI:t behöva förbättras för att attrahera fler användare. Fler enheter behövs läggas till för att nå ut till en större grupp av användare så att applikationen blir användbar på fler platser än i ett konferensrum. Vårt sätt att implementera enheter kanske inte var optimalt, dock tror vi att metoden förenklar användandet. Istället för att hårdkoda alla enheter i applikationen skulle man kunna utveckla ett program till en dator som sätter upp ett grafiskt gränssnitt till Android applikationen. Därifrån får man sen lägga till enheter och knappar till dessa just för sitt egna specifika styrsystem. Detta ställer dock högre krav på användarens tekniska kunnande.

Med mer tid och arbete kan det bli en användbar applikation som Kramers användare först och främst kan använda till sina konferensrum men senare även på andra platser. I framtida versioner av applikationen kan också användare av andra styrsystem ha nytta av den.

Referenser

- [1] Android Development Tutorial
<http://www.vogella.com/articles/Android/article.html>(acc 2012-05-14, Lars Vogel)
- [2] Content Provider <http://developer.android.com/guide/topics/providers/content-providers.html> (acc 2012-05-03, Android developers)
- [3] "Hello, Android", Introducing Google's Mobile Development Platform, Third Edition, 2010 , Ed Burnette
- [4] History of the remote controller http://www.beoworld.org/article_view.asp?id=76 (acc 2012-05-08, Beolink Magazine: 'The first 50 Years of Television' 2002, Created 2007-01-11, Modified 2007-03-22)
- [5] Infrared <http://en.wikipedia.org/wiki/Infrared>(acc 2012-05-08)
- [6] ISO/IEC 9126 http://en.wikipedia.org/wiki/ISO/IEC_9126#Quality_Model (acc 2012-05-17)
- [7] ISO 9126 external systems quality characteristics, subcharacteristics and domain specific criteria for evaluating e-Learning systems
<http://web.up.ac.za/ecis/SACLA2010PR/SACLA2010/Papers/SACLA027.pdf> (acc 2012-05-17,
I. Padayachee, P. Kotze, A. van Der Merwe)
- [8] Quality in Use: Meeting User Needs for Quality
<http://www.usabilitynet.org/papers/qiuse.pdf> (acc 2012-05-30, Nigel Bevan)
- [9] Remote Control <http://www.ideafinder.com/history/inventions/remotectl.htm> (acc 2012-05-08)
- [10] Remote Control http://en.wikipedia.org/wiki/Remote_control (acc 2012-05-08)
- [11] RS-232 <http://en.wikipedia.org/wiki/RS-232>(acc 2012-05-05)
- [12] The RS232 Standard
http://www.camiresearch.com/Data_Com_Basics/RS232_standard.html (acc 2012-05-08, Christopher E. Strangio)
- [13] UDP http://en.wikipedia.org/wiki/User_Datagram_Protocol (acc 2012-05-10)
- [14] What is Android? <http://developer.android.com/guide/basics/what-is-android.html>
(acc 2012-05-10)

Figurer

- 1, 2, 3: <http://morecontrol.com/2009/08/lazy-bones-to-redeye-a-brief-history-of-the-tv-remote/>
- 4: <http://www.miccastore.com/micca-digital-media-player-remote-control-select-model-p-64.html>
- 5: <http://www.designintegration.co.uk/technologies/switching-and-distribution/>
- 6: http://www.neets.dk/products/control_system/307-0002/index.php
- 7: <http://navcom-solutions.com/audio-and-video-systems/details/crestron-systems>
- 8, 9, 10: http://www.camiresearch.com/Data_Com_Basics/RS232_standard.html
- 11: <http://en.wikipedia.org/wiki/Infrared>
- 12: http://en.wikipedia.org/wiki/User_Datagram_Protocol
- 13: <http://kebomix.wordpress.com/2010/08/17/android-system-architecture/>
- 14: http://ofps.oreilly.com/titles/9781449390501/Main_Building_Blocks.html
- 16: <http://glossar.hs-augsburg.de/Gebrauchstauglichkeit>