



SOVELLUSKEHITYS ÄLYTELEVISION

Petri Mourujärvi
Opinnäytetyö
Kevät 2013
Tietojenkäsittelyn koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma, sähköinen liiketoiminta ja digitaalinen media

Tekijä: Petri Mourujärvi

Opinnäytetyön nimi: Sovelluskehitys älytelevisioon

Työn ohjaaja: Tapani Alakiuttu

Työn valmistumislukukausi ja – vuosi: Kevät 2013

Sivumäärä: 41 + 14 liitesivua

Tämän opinnäytetyön tarkoituksena oli selvittää ja vertailla, minkälaisia eri teknologioita älytelevisio laitteissa käytetään sovelluskehitykseen. Vertailun tuloksena oli saada selville, voiko sama sovellus toimia useamman eri valmistajan älytelevisiolaitteessa tai jopa tietokoneessa. Tavoitteena oli myös toteuttaa demo sovellus yhdessä toimeksiantajan kanssa. Opinnäytetyön toimeksiantajana on BonWell Intelligence Oy.

Suoritetun vertailun perustella kävi selväksi, että sellaista sovellusta ei voida toteuttaa, joka toimisi useamman eri valmistajan älytelevisiossa. Älytelevisiosovellus ei toimisi sellaisenaan myöskään tietokoneessa, vaikka käytettävä teknologia pohjautuu selaimissa käytettäviin web-teknologioihin. Demosovelluksen kehitysalustan valintaa varten tehtiin yhteenveto kehitysalustojen keskeisistä sovelluksen tekemiseen liittyvistä asioista. Yhteenvetoon perusteella demosovellus päätettiin tehdä Samsungin kehitysalustalla.

Demosovellukselle asetettiin tavoitteeksi reaaliaikaisen videokuvan lähettäminen kahden älytelevisiosovelluksen välillä. Käytettävissä ei ollut kuitenkaan kahta älytelevisiota. Demosovelluksen testaaminen oli siten mahdollista ainoastaan kehitysalustan emulaattoria käyttäen. Demosovellusta lähdettiin kehittämään asteittain. Demosovellukselle asetettuja tavoitteita ei kuitenkaan pystytty toteuttamaan opinnäytetyön puitteissa. Samsungin kehitysalusta ei tarjoa sovellukselle suoraa ohjelmointirajapintaa, jolla videokuvan lähettäminen voitaisiin toteuttaa älytelevisiosovellusten välillä.

Opinnäytetyössä on tehty myös katsaus tietokoneiden välityksellä tehtävään videoyhteyteen. Kehitteillä olevat ohjelmointirajapinnat mahdollistavat tietokoneen selainten välisen videoyhteyden muodostamisen HTML5:tä ja JavaScriptiä käyttäen. Samsungin älytelevision selainmoottori ei kuitenkaan tue näitä vielä kehitteillä olevia ohjelmointirajapintoja. Samsungin uusin kehitysalusta mahdollistaa kuitenkin ulkopuolisten web-ohjelmointirajapintojen hyödyntämisen. Uusimmasta kehitysalustasta löytyy tuki myös web-socketin käyttämiseen. Tämän opinnäytetyön puitteissa näitä mahdollisuuksia ei voitu kuitenkaan tutkia.

Asiasanat: älytelevisio, sovelluskehitys, kehitysympäristö, html5, suoratoisto

ABSTRACT

Oulu University of Applied Sciences
Business Information Systems, digital media and electronic business

Author: Petri Mourujärvi

Title of thesis: Smart TV Application Development

Supervisor: Tapani Alakiuttu

Term and year when the thesis was submitted: Spring 2013

Number of pages: 55

The aim of the thesis was to clarify and compare what kind of different technologies are used for the Smart TV application development. The purpose of comparing was to find out if it is possible to develop an application which can be used in different manufactures' devices or even computers. The goal was also to develop a demonstration application together with the client organisation. The client organization for the thesis is BonWell intelligence Oy.

The result of comparing was that the same application cannot be used in different manufactures' Smart TV devices. A smart TV application would not function as such in a computer either even though the technology used is based on web-technologies used in browsers. The central factors related to developing an application were summarised to be able to select the platform for demo application. Based on the summary, Samsung platform was selected for demonstrating application development.

The target for the demonstration application was to implement two-way video communication between the Smart TV applications. Target devices were not available for development. Therefore, the only way to test applications was to use the emulator of the development platform. Development was started step by step. The target set for the demonstration application could not be met within the frame of the thesis. There is no such an application interface available on Samsung development platform which can be used for two-way video communication.

The thesis includes an overview of two-way video communication between computer browsers. Interfaces under development enable implementation of two-way video communication between browsers using HTML5 and JavaScript. These interfaces are not supported by Samsung Smart TV browser engine. However, the newest platform makes it possible to utilise external web interfaces. Also a web-socket is supported by the newest platform. It was not possible to investigate these features within the framework of this thesis.

Keywords: Smart TV, application development, html5, streaming

SISÄLLYS

1 JOHDANTO	5
2 ÄLYTELEVISIO	6
2.1 Älytelevisiosovellus	6
2.2 Sovellusvalikoima	7
2.2.1 Videosovellukset	7
2.2.2 Yhteisöpalvelusovellukset	9
2.2.3 Pelisovellukset	9
2.2.4 Muut sovellukset	9
2.3 Oheislaitteet	10
3 KEHITYSALUSTOJEN VERTAILU	11
3.1 Kehitysalustojen käyttämät tekniikat	11
3.2 Median toistaminen ja siirtotekniikka	15
3.2.1 Median streamaus	16
3.2.2 Median suojaaminen	19
3.2.3 Median pakkaaminen	20
3.3 Muut ominaisuudet	20
4 KEHITYSALUSTAN VALINTA DEMO SOVELLUKSEN TEKEMISEEN	23
5 SOVELLUS KEHITYSTYÖ ÄLYTELEVISIOON	24
5.1 Sovelluksen kehitysprosessi	24
5.2 Kehitysympäristön asentaminen	25
5.3 Kehitysympäristö	26
5.4 Demosovellus	29
6 TULOKSET	34
7 POHDINTA	36
LÄHTEET	39
LIITTEET	42

1 JOHDANTO

Pitkin opiskelua opinnäytetyö on käynyt aika ajoin mielessä. Muutamia vaihtoehtoja on tullut vastaan mutta kynnyks aloittamiselle on ollut liian suuri. Opinnäytetyön tullessa lopulta ajankohtaiseksi mietin vaihtoehtoja, joiden kautta löytäisin kiinnostavan aiheen. Minua motivoivat tehtävät, joissa pääsee ratkaisemaan teknisiä ongelmia. Itse aiheen lisäksi minulle oli tärkeää, että toimeksiantajan kanssa yhteistyö sujuisi. Laitoin Yritystakomon (<http://www.yritystakomo.fi/>) kautta sen piirissä toimiville alkaville yrityksille ilmoituksen, jossa ilmoitin etsiväni itselleni aihetta opinnäytetyöhön. Yllätykseksi yhteydenottoja tuli lukuisia. Vaihtoehtoja puntaroidessani valitsin kaksi eniten itseä kiinnostavaa aihetta, joista kävin paikan päällä keskustelemassa. Harkittuani valitsemiani vaihtoehtoja päädyin sovelluskehitys älytelevisioon vaihtoehtoon. Valintaani vaikutti se, että aihe oli itselleni sopivan haastava. Myös sovellusten kehittämisessä käytettävät teknologiat kiinnostavat minua. Lisäksi toimeksiantajistani saama positiivinen mielikuva helpotti päätöstä.

Opinnäytetyön tarkoitus oli ottaa selville, minkälaisia kehitysalustoja eri älytelevisiovalmistajat käyttävät. Lisäksi työhön kuului kehitysalustojen keskinäinen vertailu ja demosovelluksen tekeminen yhdessä toimeksiantajan kanssa. Älytelevisioiden nopea kehittyminen tuo kuluttajan ulottuville mahdollisuuden käyttää uusia palveluita olohuoneen sohvalta käsin. Tämä tuo vastaavasti sovellusten kehittäjille uusia liiketoimintamahdollisuuksia. Älytelevisioiden sovelluksissa käytettävä teknologia pohjautuu tietokoneen selaimissa käytettäviin teknologioihin. Tässä opinnäytetyössä on tehty myös katsaus tietokoneen selaimilla toteutettavaan kaksisuuntaisen videoyhteyden muodostamiseen. Kirjallista aineistoa liittyen suoraan älytelevisioiden sovellusten kehittämiseen ei ollut saatavilla. Myöskään vastaavanlaisia opinnäytetöitä ei etsimieni tietojen perusteella ole tehty aiheesta. Opinnäytetyössä käytetty teoriapohja perustuukin älytelevisiovalmistajien ilmoittamiin tietoihin sekä teknologiasta kertoviin web-sivuihin. Itse älytelevisiosovellusten käyttämistä tekniikoista sen sijaan löytyy kuitenkin kirjallisuutta.

2 ÄLYTELEVISIO

Älytelevisio on laite, joka tuo käyttäjän ulottuville uusia palveluja perinteiseen televisioon verrattuna. Älytelevisiotoiminto voi olla televisioon sisäänrakennettu, tai se voi olla erillinen laite, joka on yhdistetty televisioon. Älytelevisiosta löytyy internetyhteys, joka on sen eräs keskeisin ominaispiirre. Internet tarjoaakin yhä enemmän palveluja, jotka ovat tarkoitettu myös älytelevisioon käyttöön. Älytelevisio mahdollistaa myös interaktiivisuuden, ja tuo siten uusia käyttömahdollisuuksia kuluttajille. Älytelevisioon käyttö ei rajoitu kuitenkaan vain internetiin tarjoamiin mahdollisuuksiin. Älytelevisioon voi kytkeä esimerkiksi osaksi kotiverkkoa, ja katsoa verkon yli muiden laitteiden jakamaa sisältöä. Älytelevisiolla voi myös pelata pelejä. Lisäksi älytelevisiossa on laajat kytkentämahdollisuudet ulkoisille laitteille monipuolisten liitännöiden avulla. Älytelevisioon avulla on mahdollista esittää internettarjoannon lisäksi koko talouden kotiverkkoon kytkettyjen laitteiden sisältö samalla kuvavuodulla.

2.1 Älytelevisiosovellus

Älytelevisio itsessään ei vielä tuo käyttäjän ulottuville internetin tuomia mahdollisuuksia. Palvelujen käyttämiseen tarvitaan ohjelmia, joita kutsutaan älytelevisiosovelluksiksi. Erilaisia sovelluksia voi olla valmiiksi asennettuna älytelevisioon valmistajien toimesta. Sovelluksia voi myös ladata ja asentaa älytelevisioon internetyhteyden avulla valmistajien ylläpitämistä sovelluskaupoista. Käytettävä sovellus voi myös sijaita pilvipalvelimella. Sovelluksia käytetään älytelevisioon selaimella tai selainmootorilla. Sovellus voi olla ilmainen tai maksullinen. Älytelevisiosovellukset poikkeavat kuitenkin perinteisistä tietokoneen web-sovelluksista, vaikka niiden tekemisessä käytetään samoja teknologioita. Sovelluksien suunnittelussa on otettu huomioon television tuomat erityistarpeet. Näistä tärkeimpiä ovat katseluetäisyys, televisioruudun koko sekä käyttöliittymä.

Älytelevisiosovelluksen katseluetäisyys on hyvin erilainen verrattuna tietokoneen tai älypuhelinsovelluksen käyttämiseen. Käytössä ei myöskään ole tietokoneen tapaan hiirtä. Navigointi tapahtuukin vielä pääsääntöisesti älytelevisioon kauko-ohjaimen nuoli näppäimiä hyväksi käyttäen. Kehitys menee tosin tällä saralla nopeasti eteenpäin ja älytelevisioihin on saatavilla koko ajan lisää erilaisia lisälaitteita. Älytelevisiosovelluksen käyttöliittymä on kuitenkin yksinkertaisempi kuin tie-

tokone sovelluksessa. Käyttöliittymä on lähempänä älypuhelimien käyttöliittymää kuin tietokoneen käyttöliittymää.

2.2 Sovellusvalikoima

Älytelevisiosovelluksilla on luokitus. Sovelluskaupassa ne sijaitsevat luokituksen mukaisissa gallerioissa. Luokituksissa on hieman eroa älytelevisiovalmistajien välillä. Sovellukset voidaan luokitella esimerkiksi viihde-, peli-, uutispalvelu- ja urheilusovelluksiin. Sovelluksilla on nimen lisäksi myös sovellusikoni. Ikonin ulkoasu tulisi olla hyvin sovelluksen käyttötarkoitusta kuvaava. Ikonin voidaan ajatella olevan kirjanmerkki älytelevision selainmoottorille, jolla sovelluksia käytetään. Älytelevisio valmistajien laitteiden teknisistä eroavaisuuksista johtuen sovellukset ovat laitevalmistajariippuvaisia. Lisäksi saman valmistajan mallien välillä on myös eroja. Tästä syystä tietyn valmistajan laitteelle tehdyt sovellukset eivät välttämättä toimi saman valmistajan kaikissa malleissa. Sovellustarjonnassa on myös maakohtaisia eroja. Seuraavaksi teen pienen katsauksen tarjolla oleviin erilaisiin älytelevisiosovelluksiin. Katsaukseen on pyritty ottamaan sovelluksia erilaisista sovellusluokista.

2.2.1 Videosovellukset

Internet välityksellä toimivia videopalveluita on hyvin paljon tarjolla. Älytelevisioihin on kehitetty sovelluksia joilla näitä palveluja voidaan katsoa. Tuotettu palvelu voi olla ilmainen tai maksullinen. Älytelevisiosovellukset ovat rajoittuneempia versioita vastaaviin tietokone sovelluksiin verrattuna. Ne on tarkoitettu pelkästään videon toistamiseen. Niillä ei voi ladata videoita palveluun, kuten tietokonesovelluksilla.

Internetin video palvelut voidaan jakaa useaan eri ryhmään tarkoituksensa mukaisesti. YouTube-palvelu on varmaan lähes kaikille internetin käyttäjille tuttu palvelu. Se tarjoaa ilmaisen kanavan jakaa videosisältöä internetin välityksellä. Videon lataaminen palveluun vaatii kuitenkin rekisteröitymisen. YouTube-palvelussa olevien videoiden katsominen ei vaadi rekisteröitymistä. YouTube-sovellus on ilmainen, ja se löytyy valmiiksi asennettuina lähes kaikilta älytelevisiovalmistajilta.

Vimeo-palvelu on myös YouTube-palvelun kaltainen videon tallentamiseen ja jakamiseen erikoistunut palvelu. Vimeo-palveluun voi ladata myös omia videoita, ja jakaa niitä palvelun käyttäjille.

Palvelu poikkeaa kuitenkin YouTube-palvelusta. Vimeo-palvelussa voi verkostoitua ja määritellä, kenellä on oikeus nähdä palveluun lataamasi video. Vimeo-palvelun käyttö vaatii myös aina rekisteröitymisen. Vimeon perustili on ilmainen mutta Vimeo plus- ja Vimeo Pro -palvelut maksavat. Vimeo-sovellus on saatavilla usean eri valmistajan älytelevisioon. Älytelevisioon ladattava Vimeo-sovellus on ilmainen.

Videovuokraukseen internetin välityksellä tarkoitettuja palveluita löytyy useita. Palvelu voi olla erikoistunut tietyn alan elokuvaan tai esimerkiksi uutuuksiin tai klassikoihin. Maksutapoja on myös useita. Osalla palvelun tuottajilla voi olla kiinteä kuukausimaksu, jolla voi vapaasti katsoa kaikkia saatavilla olevia elokuvia. Osalla on elokuva- tai esityskohtainen hinta, jonka suuruus voi myös vaihdella käyttötavasta. Kertakatselu on halvempi kuin itselle lataaminen. Myös ladattavan videon laatu voi olla eri tasoa. HD-laatuinen video maksaa enemmän kuin standardi-laatuinen video. Palvelujen käyttäminen vaatii rekisteröitymisen.

Netflix-videovuokrauspalvelu perustuu kiinteään kuukausimaksuun. Netflix-palvelussa voi katsoa kiinteällä hinnalla tarjolla olevia elokuvia ja sarjoja. Viaplay on samantyylinen palvelu kuin Netflix. Viaplay-sovelluksen avulla voi katsoa lisäksi reaaliaikaisia lähetyksiä. Molemmat tarjoavat palvelua myös Suomessa, ja niiden käyttämiseen on saatavilla sovellus älytelevisioihin. Digital Theatre -palvelu on erikoistunut teatteriesitysten digitaaliseen esittämiseen. Esityksiä voi vuokrata palvelusta. Hinnoittelu perustuu kertaostokseen. Hinta riippuu siitä, lataako esityksen itselleen vai katsooko sen vain kerran. Myös itselle ladatun esityksen kuvanlaadulla on merkitystä hintaan. Digital Theatre -sovellus on saatavilla ainakin Samsung-älytelevisioon. Maakohtaisessa saatavuudessa voi olla eroja.

Erlaisia uutis- ja sääpalveluun tarkoitettuja sovelluksia löytyy kaikilta älytelevisiovalmistajilta. USA today, Euronews ja Accuweather ovat esimerkkejä tällaisista uutis- ja sääpalveluista joihin on älytelevisiosovellus saatavilla. Uutispalvelusovelluksen avulla on myös mahdollista seurata reaaliaikaisia lähetyksiä.

Skype on videopuheluihin tarkoitettu sovellus. Videopuhelupalvelu poikkeaa kaikista edellä mainituista palveluista siten, että kyseessä on kaksisuuntainen palvelu. Videopuhelu tapahtuu internetin välityksellä. Skype-sovellus on saatavilla usean eri valmistajan älytelevisioon. Videopuhelua varten älytelevisiossa pitää olla kameralle ja mikrofonille tuki. Kamera voi olla sisäänra-

kennettu älytelevisioon, tai se voi olla myös lisälaitte, joka kytketään USB-liitännällä älytelevisioon. Skypen käyttö vaatii rekisteröitymisen ja Skype-tilin luomisen. Skype-puhelu on ilmainen kahden Skype ominaisuuden omaavaan laitteen välillä.

2.2.2 Yhteisöpalvelusovellukset

Yhteisöpalvelusovellukset ovat sosiaalisen median käyttöön suunniteltuja sovelluksia. Facebook on varmaan yleisin tähän tarkoitukseen käytetty sovellus. Facebook-sovellus on saatavilla myös älytelevisioon. Facebook-palvelun käyttäminen vaatii aina rekisteröitymisen. Palvelun avulla voi pitää yhteyttä kaveripiirin kanssa. Älytelevision Facebook-sovelluksessa on myös rajoituksia tietokonesovellukseen verrattuna. Älyteleviolla ei pysty lataamaan kuvia ja videoita palveluun. Myös Twitter-sovellus on saatavilla älytelevisioon. Twitter-palvelu poikkeaa Facebook-palvelusta siten, että kuka tahansa pystyy seuraamaan ketä tahansa palveluun liittynyttä.

2.2.3 Pelisovellukset

Pelisovellukset ovat tulleet nyt myös älytelevisioihin. Uudet älytelevioidot antavat sovelluksen kehittäjille mahdollisuuden luoda interaktiivisuutta erilaisten toimintojen avulla. Angry Birds on Rovion suunnittelema peli, joka on alun perin suunniteltu älypuhelimien. Angry Birds -pelisovellus löytyy nyt myös Samsungin uusiin älytelevisioihin. Pelaaminen tapahtuu käden liikkeellä ilman ohjainta. Pelin ohjaaminen perustuu liiketunnistukseen, joka tapahtuu älytelevision kameralla. Peli on ilmainen mutta ei toimi kaikissa malleissa.

2.2.4 Muut sovellukset

Uusista älytelevioido malleista löytyy myös internetselaimen. Digitaalisten kuvien, musiikin ja radiokanavien esittämiseen löytyy myös sovelluksia älytelevisioihin. Periaatteessa tekniikka mahdollistaa kaikenlaisen digitaalisessa muodossa olevan materiaalin esittämisen älytelevioidossa, ja mikä mielenkiintoisinta, älytelevioidoihin on mahdollista kehittää sovelluksia myös itse.

2.3 Oheislaitteet

Älytelevisio-ominaisuuksien lisääntyminen aiheuttaa älytelevisioiden käytölle omat haasteensa. Tietokoneissa sovellusten käyttämiseen käytetään näppäimistöä ja hiirtä. Pelikonsoleissa käytetään peliohjaimia pelaamiseen. Perinteinen televisiotoimintojen ohjaaminen tapahtuu kaukosäätimellä. Kaikki nämä samat toiminnot ovat monessa älytelevisioissa samassa paketissa. Seuraavaksi teen pienen katsauksen tällä hetkellä älytelevisioon saatavilla oleviin ohjaimiin.

Kauko-ohjain on vakiovaruste kaikissa älytelevisioissa. LG on suunnitellut kauko-ohjaimen jossa on hakupyörä. LG on nimennyt ohjaimen taikaohjaimeksi. Ohjaimella pystyy osoittamaan televisioruudulta haluttua kohtaa, joten se toimii tietokoneen hiiren tapaan. Älytelevisioiden ohjaaminen voi tapahtua myös älypuhelimien tai taulutietokoneiden avulla. Ohjaaminen on mahdollista älypuhelimien tai taulutietokoneeseen erikseen ladattavan sovelluksen avulla. Ladattava sovellus on ilmainen.

Philips lupaa näppäimistö- ja hiirituen uusissa malleissa. Myös Samsung ja LG tukevat tietyissä malleissa erillistä näppäimistöä. Ääniohjaus on myös mahdollista Samsungin ja LG:n tietyissä malleissa. Ääniohjaus tapahtuu älytelevisioiden sisäisellä tai erillisessä laitteessa sijaitsevan mikrofonin avulla. Myös liikeohjaus on mahdollista Samsungin joissain malleissa. Liikeohjaus perustuu älytelevisioiden kameraan. Liikeohjaukseen perustuvia pelejäkin on jo saatavilla. Myös monenlaisille peliohjaimille löytyy tuki, esimerkiksi Google TV -sovelluksille.

3 KEHITYSALUSTOJEN VERTAILU

Suurin osa älytelevisiovalmistajista mahdollistaa ulkopuolisten kehittää sovelluksia älytelevisioihin. Tätä varten älytelevisiovalmistajilta löytyy työkalut, joita kutsutaan kehitysalustoiksi. Kehitysalustojen keskinäisen vertailun helpottamiseksi teknisiä tietoja on kerätty Excel-taulukoon (liite 1). Tiedot on kerätty älytelevisiovalmistajien kehittäjille suunnatuilta kehitysfoorumeiden sivuilta. Vertailussa ei ole mukana kaikkia mahdollisia älytelevisiovalmistajia. Mukaan on otettu tunnettuja valmistajia, joilta löytyy omat kehitysalustat. Mukana ovat Samsung, LG, Philips ja Sharp. Tunnetuista valmistajista Sony ei ole listassa mukana, koska sillä ei ole varsinaisesti omaa kehitysalustaa. Sony käyttää Google TV -alustaa. Panasonic jäi pois siitä syystä, että Panasonicilla on muista poiketen vuosimaksu sovelluskehittäjille. Philips ja Sharp ovat liittoutuneet, ja heiltä löytyy yhteinen kehitysalusta. Philips ja LG ovat kehittäneet yhdessä myös Smart TV Alliance -nimellä kulkevan spesifikaation älytelevisiosovelluksille. Tunnettujen älytelevisio valmistajien lisäksi ainakin Nyxiolta ja Toshibaalta löytyy älytelevisiomallit.

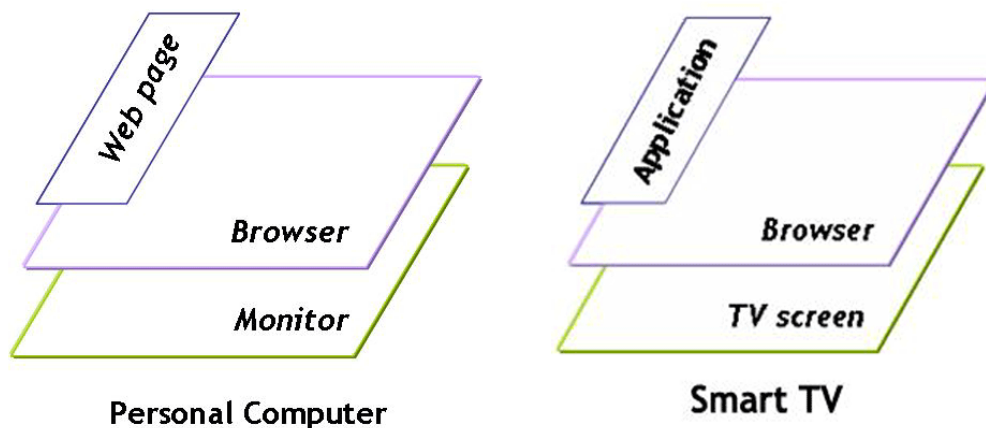
Mukana vertailussa on myös alustan tarjoavia palveluja. Google TV, Yahoo Connected TV ja Espial Browser TV tarjoavat oman alustan älytelevisiovalmistajien käyttöön. Apple TV ei ole mukana, koska se ei ole vielä avannut sovellusten kehitysmahdollisuutta ulkopuolisille. Liitteenä olevaan taulukkoon (liite 1) ei ole kerätty kaikkea mahdollista tietoa. Taulukon tekeminen olisi vienyt liikaa aikaa, ja älytelevisiolaitteiden kehitys menee nopeaa tahtia eteenpäin. Manuaalisesti tehdyn taulukon ajan tasalla pitäminen vaatisi myös jatkuvaa ylläpitoa. Opinnäytetyön kirjoittamisen aikana kaikki laitevalmistajat julkaisivat jo uudemmat kehitysalustaversiot. Myös Smart TV Alliancen vielä kehitteillä oleva spesifikaatio hyväksyttiin opinnäytetyön kirjoittamisen aikana. Vertailussa käytettyjä teknikoita ja terminologiaa on myös pyritty kuvaamaan vertailun ohessa. Vertailussa on mukana siis sekä älytelevisio laitevalmistajia että pelkän alustan tarjoavia ratkaisuja.

3.1 Kehitysalustojen käyttämät tekniikat

Kehitysalustojen SDK (Software Development Kit) tai ADK (Application Development Kit) keskinäistä vertailua hankaloittaa se, että valmistajilla ei ole yhtenäistä tapaa ilmoittaa teknisiä tietoja. Lisäksi samalla älytelevisiovalmistajalla voi olla käytössään useita eri versioita kehitysalustoista.

Kehitysalustaversiot ovat sidoksissa valmistajan eri malleihin. Suunniteltaessa sovellusta älytelevisioon täytyykin huomioida, haluaako kehittäjä saada sovelluksensa toimimaan tietyn valmistajan kaikissa älytelevisio malleissa. Tällaisessa tapauksessa voi joutua turvautumaan myös vanhempaan kehitysalustaan ja tekniikkaan.

Sovelluksien käyttämiseen tarvittavasta ohjelmasta käytetään nimitystä app engine tai browser engine. Opinnäytetyössä näistä on käytetty suomenkielistä termiä selainmoottori. Älytelevisiosovelluksen kehittämisessä käytetään pääsääntöisesti samoja teknikoita kuin web-sovelluksen valmistamisessa. Älytelevisiosovelluksen käyttöä voi verrata web-sivujen katsomiseen tietokoneen selaimella. Kuvassa 1 on esitetty vertaus älytelevisio sovelluksen ja web-sivun välillä. (<http://www.samsungdforum.com/Guide/art00005/index.html>, 28.12.2012.)



Kuva 1. Web-sivun ja älytelevisiosovelluksen vertauskuva

Kaikkien vertailussa olevien laitevalmistajien sekä Espial Browser TV -kehitysalustat perustuvatkin web-tekniologiassa käytettyihin HTML- (Hypertext Markup Language), CSS- (Cascading Style Sheets) ja JavaScript -ohjelmointikieliin. HTML ja CSS eivät kuitenkaan varsinaisesti ole ohjelmointikieliä. HTML- ja CSS -kieliä voidaan kutsua ohjelmointikielten sijasta merkkaukieliiksi. HTML-kieltä käytetään web-sivujen sisällön kuvaamisen ja merkkaamiseen. HTML on yleisin tähän tarkoitukseen käytetty kieli. Merkkaamiseen käytetään elementtejä, joita on paljon erilaisia tarkoituksia varten. Elementit muodostuvat alku- ja lopputageista sekä niiden väliin tulevasta si-

sällöstä. Lisäksi elementtien merkitystä voi tarkentaa määritteillä eli attribuuteilla. (Korpela, Linjama 2005, 70, 72, 74.)

HTML:n alkuaikoina myös sisällön ulkoasua muotoiltiin HTML-kielillä. Web-sivujen ulkoasujen korostamisen myötä koodin luettavuus ja ymmärrettävyys vaikeutuivat. Tähän tarkoitukseen kehitettiin oma kieli CSS. CSS on puhtaasti web-sivun sisällön ulkoasun muokkaamiseen tarkoitettu kieli. CSS-koodi kannattaa tehdä omaan tiedostoon ja linkittää web-sivulle. Näin koodin ylläpidettävyys ja kierrätettävyys helpottuu huomattavasti. Samalla voidaan erottaa web-sivun ulkoasun varsinaisesta sisällöstä. (Collison, Budd, Moll 2007, 3.)

JavaScript on sen sijaan ohjelmointikieli. Sitä käytetään myös EcmaScript nimitystä ja kutsutaan skriptauskieleksi. JavaScript-kieltä käytetään lisäämään interaktiivisuutta web-sivulle. Web-sivun rakenne voidaan kuvata puurakenteena, joka koostuu olioista. Tätä puurakennemallia kutsutaan DOM-malliksi (Document Object Model). JavaScript hyödyntää DOM-mallin olioita sivun tilan muokkaamiseen. JavaScript on tarkoitettu myös pelkästään asiakaspuolen ohjelmointiin. Sen avulla web-sivun sisältöä voi muuttaa lataamatta sivua uudelleen palvelimelta. JavaScriptissä on myös sisäänrakennettuja rajoituksia. Turvallisuussyistä sillä ei ole mahdollista toteuttaa tiedoston lukemista ja kirjoittamista asiakaskoneeseen. JavaScript-koodi on suositeltavaa kirjoittaa omaan tiedostoon ja linkittää web-sivulle. (Smith, Negrino 2007, 2, 7, 13.)

JavaScript-koodin tulkitsemiseen ja suorittamiseen älytelevision selainmoottori käyttää siihen tarkoitukseen tehtyä ohjelmaa. Ohjelmasta käytetään nimitystä JavaScript-moottori. Uusimmissa alustoissa Samsung ja LG käyttävät SquirrelFish JavaScript-moottoria. Osa älytelevisiovalmistajista on sen sijaan ilmoittanut pelkästään mitä JavaScript-versiota kehitysalusta tukee, eikä JavaScript moottorin nimeä. Philips/Sharp Smart TV, Espial TV Browser ja Smart TV Alliance spesifikaatiossa on tuki JavaScript 1.5 versiolle. Yahoo Connected TV sen sijaan ei ole ilmoittanut edes versiota. Uusin JavaScript-versio on Wikipedian mukaan 1.8.5. (http://en.wikipedia.org/wiki/JavaScript_engine, 22.2.2013.)

Web-tekniikan uusimmat versiot kuten CSS3 ja HTML5 ovat varsin hyvin tuettuja kehitysalustoissa. Käytännössä kaikki web-tekniikkaan perustuvat älytelevisiovalmistajat ovat jo siirtyneet näihin uusimmissa kehitysalustoissaan. CSS3 on viimeisin versio web-sivun sisällön ulkoasun muokkaamiseen tarkoitettua kielestä. HTML5 on vastaavasti viimeisin versio HTML-kielen mää-

rittelyksi. HTML5 sisältää paljon uusia ominaisuuksia, kuten median ja dynaamisen grafiikan paremman upottamisen web-sivuille. Median upottamista varten on uudet video- ja audioelementit. Esimerkiksi älytelevisiolle keskeisen median toistaminen onnistuu ilman erillistä playeria käyttämällä videoelementtiä. Mediaesitykselle voidaan tehdä myös ohjauselementit käyttämällä video- ja audioelementtejä. Siten esitystä voi kontrolloida paremmin ja kytkeä esityksen osaksi sivun luonnollista toimintaa. Älytelevisiovalmistajien kehitysfoorumien sivuilta löytyy tiedot mitä HTML5 uusia ominaisuuksia eri alustat tukevat. Älytelevisioiden ilmoittamaan HTML5-tukeen kannattaa suhtautua pienellä varauksella. HTML5 on kuitenkin vielä keskeneräinen luonnos. Sitä kehittävät W3C- ja WHATWG -organisaatiot. HTML5 nimitystä käytetään myös yleisesti kuvaamaan kaikkea, mikä liittyy webin uusiin tekniikoihin. HTML5:n yhteydessä asioita myös yhdistetään useasti kokonaisuuksiksi. Siihen liitetään myös ohjelmointiin ja sovelluksiin kuuluvia asioita (Korpela 2011, 13, 18, 181).

Myös muunlaisia ratkaisuja on käytössä. Yahoo Connected TV ilmoittaa tekniikaksi JavaScriptin ilman HTML-koodia. Google TV on taas valinnut Android-käyttöjärjestelmään pohjautuvan kehitysalustan. Android-käyttöjärjestelmä on monikäyttö-Linux-järjestelmä, jossa jokainen sovellus on oma käyttäjä. Sovellusten ohjelmointikielenä käytetään Javaa. Jokaisella eri sovelluksella on oma Linux-prosessi. Jokaiselle prosessille on taas oma virtuaalikone. Virtuaalikoneen avulla sovellusten koodien suorittaminen on eristetty toisista sovelluksista. Android käynnistää prosessin, kun jonkun sovelluksen toiminto pitää suorittaa. Vastaavasti prosessi sammutetaan kun sitä ei enää tarvita. Järjestelmä voi sammuttaa myös prosessin, jos sen tarvitsee vapauttaa muistia jonkun toisen sovelluksen toiminnon suorittamiseen. Androidilla tehty sovellus voi toimia niin älytelevisiossa, taulutietokoneessa kuin älypuhelimessakin. Sovelluksen luokituksella määritellään, minkälaiseen käyttöön sovellus on tarkoitettu. Rajoituksena on kuitenkin se, että sovelluksessa ei ole toiminnallisuutta, jota ei tueta jossain laitteessa. Lisäksi sovellukselle pitää tehdä migraatio, jotta se toimii oikein eri ympäristöissä. (<https://developers.google.com/tv/>, 7.12.2012.) (<http://developer.android.com/guide/components/fundamentals.html>, 7.12.2012.)

Nyxio-laitevalmistaja on yhdistänyt tietokoneen ja älytelevision konkreettisesti samaan pakettiin. Nyxiolta löytyy kaksi eri mallia: VioSphere ja Vuzion. VioSphere on yhdistelmä HD-televisiosta ja tietokoneesta. VioSpheren käyttöjärjestelmänä on Windows7 Ultimate. Vuzion toimii puolestaan Android-käyttöjärjestelmällä. VioSpheressä näytön voi jakaa kahteen osaan, joita voi käyttää samanaikaisesti. VioSphere on mahdollista saada myös kosketusnäytöllä.

(<http://www.nyxio.com/>, 28.12.2012.) Panasonicin Viera kehitysalustan käyttämä tekniikka on HTML5 tai JavaScript ilman HTML-koodia. Panasonic- ja Nyxio -valmistajien tarkempia tietoja ei ole kerätty liitteenä olevaan Excel-taulukkoon (liite 1).

Älytelevisio sovellus joko ladataan laitteeseen tai sitä käytetään pilvipalvelimelta. Samsung, LG, Google TV ja Yahoo Connected TV älytelevisiolaitteissa sovellus ladataan älytelevisioon. Philips/Sharp ja Panasonic älytelevisioissa sovellus sijaitsee pilvipalvelimella. Espial Browser TV perustuu selaimelle suunnattujen palvelujen käyttämiseen älytelevisiossa.

3.2 Median toistaminen ja siirtotekniikka

Elokuvien, videoiden ja tv-sarjojen katsominen internetin välityksellä on älytelevisioyksi tärkeimpiä ominaisuuksia. Myöhemmin tässä luvussa käytettävä termi, video, voi tarkoittaa mitä tahansa edellä mainituista lähetyksimuodoista. Videon katsominen älytelevisiossa tapahtuu erillisellä playerilla tai selainmoottorin avulla. Player voi toimia itsenäisesti tai olla älytelevisioselaimen moottorin liitännäinen. HTML5 tuki mahdollistaa myös videon upottamisen sovellukseen videoelementin avulla. Videoelementin avulla voi toteuttaa videon toistamisen ilman erillistä playeria. Samsung, LG, Google TV ja Espial TV Browser ilmoittavat tukevansa Adoben flash player versiota 10.1. Google TV tukee myös beta versiota 10.2. LG-älytelevisiossa tuki on ainoastaan playerin ”stand alone” -versiolle, mikä tarkoittaa että se toimii itsenäisesti ilman selainmoottoria. Yahoo Connected TV käyttää KONtx-mediaplayeria.

Flash player tuen lisäksi, Samsung ja LG ilmoittavat tukevansa myös Adobe AIR (Adobe Integrated Runtime) ajoympäristöä. Adobe Air mahdollistaa kehittäjän yhdistää HTML-, JavaScript- ja ActionScript -tekniikat sekä Adoben Flash- ja Flex -teknologiat tuottaakseen vuorovaikutteisia internetsovelluksia. Samsung tukee versioita AIR for TV 2.5.1 ja LG tukee versiota AIR 3.0 for TV. (<http://www.adobe.com/fi/products/air/faq.html>, 22.2.2013.)

3.2.1 Median streamaus

Videon toistamiseen internetin välityksellä älytelevisiossa käytetään streamaustekniikkaa. Streamaus eli suoratoiston tuki löytyy kaikista kehitysalustoista. Streamaus mahdollistaa videon katsomisen ilman, että sitä ladataan älytelevisioon. Streamaus mahdollistaisi myös sen, että lataamisen yhteydessä toistamisen voisi aloittaa jo itse lataamisen aikana. Streamaus voidaan jakaa live- ja VOD (Video On Demand) -streamaukseen. Live streamaus tarkoittaa reaaliaikaisen lähetksen katsomista. VOD tarkoittaa esitallennetun videon katsomista. Älytelevisiossa videon lataaminen on rajoitettua niistä puuttuvan talletuskapasiteetin johdosta.

Videon siirtämiseksi internetin välityksellä tarvitaan myös tiedonsiirtoprotokollaa. Http/https protokollien lisäksi, älytelevisio alustat tukevat useita eri protokollia. Flash player käyttää RTMP/RTMPe (Real Time Messaging Protocol) -protokollaa. Protokolla on alun perin Macromedian kehittämä ja tarkoitettu audion-, videon- ja tiedonsiirtoon flash playerin ja palvelimen välillä. Adobe on julkaissut siitä julkisen version yleiseen käyttöön. Julkaistu versio on kuitenkin vaillinaisen versio alkuperäisestä. RTMP-protokollasta on useita variaatioita. RTMPe-versio sisältää salauksen ja käyttää Adoben omaa suojaus mekanismia. RTMP pohjautuu TCP (Transmission Control Protocol) -protokollaan, missä ylläpidetään jatkuvaa yhteyttä. Tämä mahdollistaa kommunikaation, jossa on vain pieni viive. Lähetettävä stream pilkotaan palasiksi, joiden koko neuvotellaan dynaamisesti käyttäjän ja palvelimen välillä. Tämä mahdollistaa mahdollisimman sujuvan ja maksimaalisen tiedon lähettämisen. Pilkotut palaset eri streameista, voidaan lomittaa ja lähettää ne multipleksaamalla ne yhden yhteyden yli. ([http://en.wikipedia.org/wiki/Real Time Messaging Protocol](http://en.wikipedia.org/wiki/Real_Time_Messaging_Protocol), 22.2.2012.)

MMS (Microsoft Media Server) on Microsoftin kehittämä streamaus protokolla. MMS-protokolla käyttää UDP- (User Datagram Protocol) tai TCP -protokollaa tiedon siirtämiseen yhteen kohteeseen (unicast) Windows media palvelimelta. MMS-protokolla on vanhentunut ja Microsoft käyttää RTSP (Real Time Streaming Protocol) -protokollaa uudemmassa Windows media palvelin 9 sarjassa. MMS-protokolla on kuitenkin tuettu myös tässä palvelimessa. Älytelevisio valmistajista ainoastaan Samsung ilmoittaa tukevansa MMS-protokollaa. ([http://en.wikipedia.org/wiki/Microsoft Media Server](http://en.wikipedia.org/wiki/Microsoft_Media_Server), 22.2.2013.)

RTP/RTSP (Real-time Transport Protocol/Real-time Streaming Protocol) on sovellustason protokolla, joka tarjoaa sovelluksille media lähettämisen yhteen tai useampaan kohteeseen samanaikaisesti. RTSP-protokolla tarjoaa yhteen toimivuuden erilaisien asiakas- ja palvelintoimijoiden välille. Protokolla luo ja kontrolloi yhtä tai useampaa ajallisesti synkronisoitua streamia tietystä mediasta. Protokolla on kaksisuuntainen ja mahdollistaa asiakkaan kontrolloida media lähetystä verkon yli kuten käynnistää, pysäyttää ja pika kelata sitä. Protokolla käyttää lähetykseen TCP- tai UDP -protokollaa riippuen tilanteesta. RTP-protokollaa käytetään lisäämään aika- ja jaksotiedot jokaiseen lähetettävään pakettiin, jotta mediatiedosto voidaan kasata vastaanotto päässä uudelleen. (Minoli 2012, 242.)

Edellä esitettyjen protokollien lisäksi löytyy myös protokollia, jotka mahdollistavat adaptiivisen streamauksen. Adaptiivisista protokollista käytetään lyhennettä HAS (Http Adaptive Streaming). Adaptiivisessa streamauksessa siirtonopeus voi vaihdella verkon kapasiteetin mukaan. Kaikkien älytelevisioiden uusimmista alustoista löytyy tuki HLS (Http Live Streaming) -protokollalle. HLS-protokolla on Applen toteuttama adaptiivinen protokolla, joka pohjautuu http-protokollaan. Http:een pohjautuvan protokollan etu on siinä, että sillä pystytään siirtämään minkä tahansa palomuurin tai proxy serverin läpi standardin mukaista http-liikennettä. HLS-protokolla tarjoaa myös salauksen ja käyttäjän todentamisen. Tämä mahdollistaa siten julkaisijan suojata julkaisunsa. HLS-protokollalla on mahdollista toteuttaa myös live streamausta.

HLS-videon jakeluun voidaan käyttää ihan tavallista web-palvelinta. Video pilkotaan pieniksi tiedostoiksi erillisellä ohjelmalla, jota kutsutaan segmentoijaksi. Segmentoija voi olla joko stream segmentoija live-lähetyksille tai file segmentoija VOD-lähetyksille. Pilkoituja tiedostoja kutsutaan mediasegmenttiedostoiksi. Pilkotut mediasegmenttiedostot tallennetaan web-palvelimelle. Tallennetuista tiedostoista tehdään lista, jota päivitetään. Listasta käytetään nimitystä playlist tai index file. Lista sisältää URL (Uniform Resource Locator) -tiedon kulloinkin ladattavissa olevista mediatiedostoista. Lista ladataan vastaanottajan toimesta aika ajoin. Mediatiedoston keston voi määrittellä segmentoijan asetuksissa, ja tyypillisesti se on 10 sekuntia. Live streamauksessa pilkotut mediatiedostot voidaan poistaa palvelimelta lähetyksen jälkeen. VOD-streamauksessa vastaavasti ne tallennetaan jolloin ne ovat käytettävissä milloin tahansa.

Lähetettävästä videosta voidaan tehdä myös vaihtoehtoisia listoja. Tämä mahdollistaa eri laatuisten videoiden lähettämisen käytettävissä olevan verkon kapasiteetin mukaan. Tämä tapahtuu

siten, että alussa ladataan isäntälistä (master index file). Tämä lista sisältää tiedon vaihtoehtoisista listoista. Vastaanottajan pään ohjelmisto päätelee heuristisesti verkon suorituskykyä. Sen perusteella se lataa sitten parhaan mahdollisen käytettävissä olevan listan käyttöönsä. Segmentoijan voi määritellä tekemään tarvittavat vaihtoehtoiset mediasegmenttiedostot. Huomioitavaa on että äänen laatu ja kuvasuhteen koko täytyy olla sama kaikissa vaihtoehtoisissa listoissa. (http://developer.apple.com/library/ios/#documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/Introduction/Introduction.html#//apple_ref/doc/uid/TP40008332-CH1-DontLinkElementID_39, 16.1.2013.)

Mpeg-Dash (Moving Picture Experts Group-Dynamic Adaptive Streaming over HTTP) on ensimmäinen kansainvälisen standardin saanut http-perusteinen adaptiivinen streamaus protokolla. Mpeg-Dash -protokollan tuki löytyy Samsung-, Philips/Sharp Smart TV-, Smart TV Alliance- ja Espial TV Browser -alustoista. Mpeg-Dash on perustoiminnaltaan edellä esitetyn HLS-protokollan kaltainen mutta se on tarkoitettu pelkästään VOD-streamaukseen. Mpeg-Dash on yleiseen käyttöön tarkoitettu protokolla. (http://en.wikipedia.org/wiki/Dynamic_Adaptive_Streaming_over_HTTP, 23.2.2013.)

Widevine on myös http-perusteinen adaptiivinen streamaus protokolla, joka HLS-protokollan tapaan on tarkoitettu sekä VOD- että live -streamaukseen. Widevine-protokollan tuki löytyy Samsungin ja LG:n alustoista. Widevine on perustoiminnaltaan myös HLS-protokollan tapainen. Widevine mahdollistaa käyttäjän laittaa kirjainmerkkejä sekä pika kelata katsottavaa videota eteen- tai taaksepäin. Protokolla on suojattu patentilla. (http://www.widevine.com/video_optimization.html, 23.2.2013.)

MS Smooth on Microsoftin Smooth Streaming -protokolla joka on julkaistu vuoden 2008 kesäolympialaisten yhteydessä. MS Smooth on myös adaptiivinen protokolla joka mahdollistaa sekä VOD- että live -streamauksen. Smart TV Alliance- ja Espial TV Browser -alustat tukevat tätä protokollaa. MS Smooth pääerot HLS-protokollaan että se käyttää aikaperusteista pakettien latausta, joten asiakkaan ei tarvitse säännöllisesti ladata index-tiedostoa. Pilkkottujen mediatiedostojen suositeltu kesto on 2 sekuntia kun se HLS suosituksessa oli 10 sekuntia. Myös stream tiedostojen formaatit eroavat toisista. (<http://www.iis.net/downloads/microsoft/smooth-streaming>, 23.2.2013.)

Muutama muukin vähemmän käytetty adaptiivinen protokolla on mainittu liitteenä olevassa Excel-taulukossa (liite 1). MMSH (MMS over HTTP) on aikaisemmin esitetyn MMS-protokollan adaptiivinen versio. Se on kuitenkin vanhentunut protokolla, joka on korvattu uudemmillä protokollilla. RTMP/RTMPE on myös adaptiivinen siirtoprotokolla. Se on esitelty aikaisemmin flash playerin käyttämänä protokollana. Älytelevisio valmistajista LG ilmoittaa tukevansa näitä protokollia. Lisäksi Philips/Sharp Smart TV tukevat vanhemmissa malleissa WMS (Windows Media Services) -protokollaa. Uusimmassa alustassa tätä ei enää tueta ja pyydetään käyttämään HLS-protokollaa.

Samsung referoi adaptiivisen streamauksen yhteydessä myös OIPF (Open IPTV Forum) -spesifikaatioon. OIPF on teknologia teollisuudesta riippumaton organisaatio. Perustaja jäseniä ovat Samsung, Ericsson, Sony Corporation, France Telecom, Telecom Italia ja Philips. Lisäksi lukematon joukko muita toimijoita on liittynyt mukaan perustamisen jälkeen. OIPF mahdollistaa ja kiihdyttää IPTV (Internet Protocol Television) -laitteiden massamarkkinoiden kasvua määrittelemällä ja julkaisemalla ilmaisia standardeihin perustuvia spesifikaatioita tuleville "end to end" IPTV-palveluille. "End to end" -spesifikaatiot ovat välttämättömiä tehokkaan ekosysteemin kehittämiseksi ja että kuluttajalle saadaan helppokäyttöisiä "plug and play" -yhteensopivia palveluja. (<http://www.oipf.tv/>, 23.2.2013.)

3.2.2 Median suojaaminen

Video- tai musiikkipalvelun käyttöön liittyy oleellisesti myös niiden suojaus DRM (Digital Rights Management). Suojauksella voidaan estää tai rajoittaa käyttäjän oikeuksia kopioida, tallentaa tai uudelleen lähettää esitettävää mediaa. Käytettäviä tekniikoita on useita, ja älytelevisioalustojen tuki vaihtelee. PlayReady suojaukselle löytyy tuki useimmista älytelevisioalustoista. Vain Philips/Sharp Smart TV -alustasta tuki puuttui vertailun teko hetkellä. WMDRM (Windows Media Digital Rights Management) suojausta käyttävät LG, Philips/Sharp Smart TV sekä Samsung vanhemmissa alustoissa. Marlin-BB (Marlin-Broad Band) suojausta käyttää pelkästään Philips/Sharp Smart TV. Widevine tarjoaa streamaus protokollan lisäksi myös suojauksen lähetettävälle medialle. Widevine suojausta käyttävät Samsung ja LG. Google TV ja Yahoo Connected TV ilmoittavat tukevansa DRM-suojausta. Tuki onkin riippuvainen näitä alustoja käyttävistä laitteista.

3.2.3 Median pakkaaminen

Videon siirtämiseen internetin välityksellä käytetään pakkaustekniikoita siirrettävän informaation määrän pienentämiseksi. Pakkaamiseen lähetyspäässä ja purkamiseen vastaanottopäässä käytetään koodekkeja. Erilaisia koodekkeja on valtava määrä, ja tuki vaihtelee myös kehitysalustakohtaisesti. Liitteenä olevaan taulukkoon koodekkien tietoja ei ole kuitenkaan kerätty. Esimerkkinä mainittakoon H.264 videokodekki, joka on tuettu kaikissa kehitysalustoissa. H.264 videokodekki onkin yksi suosituimmista videon pakkaukseen käytetyistä formaateista, jota käytetään HD-videon jakamiseen. (http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC, 18.1.2013).

3.3 Muut ominaisuudet

Kuten aikaisemmin tuli esille, liitteenä oleva Excel-taulukko (liite 1) ei sisällä kaikkea mahdollista tietoa. Älytelevisiovalmistajilla on omat tapansa ilmoittaa kehitysalustan tarjoama tuki. Tiedot on laitettu taulukkoon vain, jos ne on erikseen ilmoitettu. Tyhjänä oleva tieto ei siis välttämättä tarkoita puuttuvaa tukea kyseiselle ominaisuudelle. Ei olisi tarkoituksenmukaista tehdä vertailua sellaisista ominaisuuksista, joista ei ole tietoa kaikilta valmistajilta. Vertailua ei ole myöskään järkevää tehdä epäoleellisille asioille. Edellisissä kappaleissa oli esitelty kehitysalustojen käyttämät perustekniikat, sekä median toistamiseen liittyviä asioita. Tässä kappaleessa käydään läpi vielä niitä ominaisuuksia, mitkä ovat opinnäytetyön toimeksiantajan näkökulmasta oleellisia asioita.

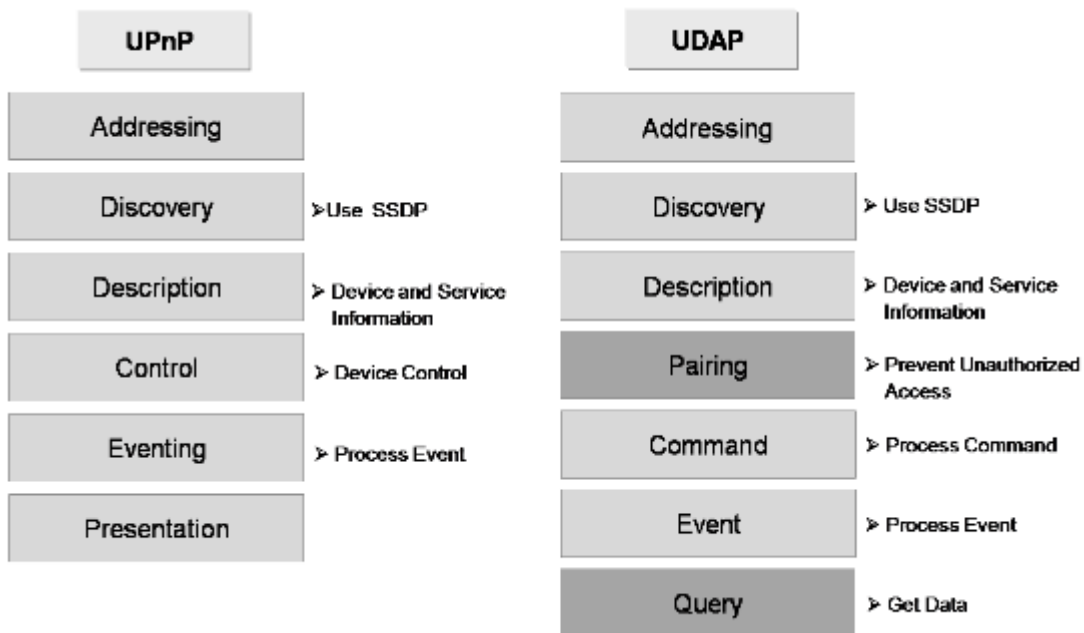
Kameran tuki sekä älytelevisio sovelluksen ohjattavuus älypuhelimella olivat asioita, mistä on etsitty vielä erikseen tietoa, vaikkei niitä älytelevisiovalmistajan sivuilla ole suoraan ilmoitettu. Usealla älytelevisiovalmistajalla onkin malleja joihin on sisäänrakennettu kamera. Kaikki älytelevisiovalmistajat eivät kuitenkaan ole antaneet kehitysalustoille avointa ohjelmointirajapintaa, jolla voi käyttää sitä. Vertailun teko hetkellä kehitysfoorumien tietojen mukaan ainoastaan Samsungin kehitysalustasta löytyy tuki kameralle. LG-kehitysfoorumien keskustelupalstalle esitetyn kysymyksen perustella kameratuki tulee vasta vuoden 2013 alussa. Kehitysfoorumien keskustelupalstat ovatkin keskeisiä paikkoja, mistä voi saada tietoa älytelevisiosovelluksen kehittämiseen.

Sovelluksien ohjaamiseen älypuhelimella tai taulutietokoneella löytyy tuki kaikista kehitysalustoista. Älypuhelimien tai taulutietokoneeseen pitää asentaa kuitenkin erillinen sovellus. Älytelevisio-

ohjaaminen tapahtuu lähiverkon välityksellä. Lähiverkko mahdollistaa myös tietojen jakamisen älytelevisioon ja muiden siihen kytkettyjen laitteiden välillä. Älytelevisiovalmistajat käyttävät eri protokollia kommunikointiin, samaan lähiverkkoon kytkettyjen laitteiden välillä. Philips ja Espial TV Browser eivät mainitse spesifikaatiossa mitä protokollaa laitteiden väliseen kommunikaation käytetään. Näiden osalta tietoa käytetystä protokollasta ei ole erikseen etsitty.

Samsung älytelevisiossa tämä kommunikaatio tapahtuu UPnP (Universal Plug and Play) -protokollalla, joka muodostuu joukosta eri verkkoprotokollia. UPnP mahdollistaa verkkolaitteiden tunnistaa toiset verkkolaitteet ja avata funktionaalinen verkkopalvelu tiedonjakamista ja kommunikointia varten. (http://en.wikipedia.org/wiki/Universal_Plug_and_Play, 21.2.2013.)

LG älytelevisioon ja muiden laitteiden väliseen kommunikaatioon lähiverkossa käytetään UDAP (Universal Discovery & Access Protocol) -protokollaa. UDAP on LG:n määrittelemä protokolla, joka pohjautuu http-protokollaan. UDAP pohjautuu myös UPnP-arkkitehtuuri konseptiin, mutta poikkeaa siitä. Alla on kuva UPnP- ja UDAP -protokolla pinoista (kuva 2). (http://developer.lgappstv.com/TV_HELP/index.jsp?topic=%2F%2Fge.tvsdk.references.book%2Fhtml%2FUDAP%2FUDAP%2FLG+UDAP+2+0+Protocol+Specifications.htm, 21.2.2013.)



Kuva 2. Kuva UPnP- ja UDAP -protokollien konseptuaalisesta erosta

Google TV käyttää taas vastaavasti omaa Anymote- ja Google TV Pairing Protocol -protokollia laitteiden väliseen kommunikaatioon lähiverkossa. Google TV Pairing Protocol -protokollaa käytetään yhteyden muodostamiseen ja Anymote-protokollaa käytetään tiedon siirtämiseen. Anymote-protokolla edellyttää kytketyn laitteen käyttävän TLS/SSL (Transport Layer Security/Secure Sockets Layer) -protokollaa. TLS/SSL-protokollan avulla voidaan suojata laitteiden välillä siirrettävä kommunikaatio, samaan verkkoon kytketyiltä muilta laitteilta tai sovelluksilta. (<https://developers.google.com/tv/remote/docs/communication?hl=fi#discovery>, 21.2.2013.) (<https://developers.google.com/tv/remote/docs/communication?hl=fi>, 21.2.2013.)

Yahoo Connected TV -älytelevisio ja ulkoisen laitteen kommunikaation lähiverkossa käytetään Yahoohon kehittämää uutta protokollaa. Protokolla käyttää SSL-protokollaan paketoitua pistoke yhteyttä TCP/IP (Transmission Control Protocol/Internet Protocol) -lähiverkon yli. (http://developer.yahoo.com/connectedtv/kontxapiref/YCTV_KONTX_Remote_Device.html, 21.2.2013.)

Eräs oleellinen asia sovellusta kehitettäessä on myös se, pystyykö sitä testaamaan keskeneräisenä kohdetelevisiossa. Testaaminen on mahdollista keskeneräisenä kaikissa laitevalmistajien älytelevisioissa lukuun ottamatta Philips/Sharp Smart TV:tä. Testaaminen Philips/Sharp Smart TV -sovelluksille tapahtuu käyttäen erillistä validointityökalua.

4 KEHITYSALUSTAN VALINTA DEMO SOVELLUKSEN TEKEMISEEN

Opinnäytetyö on jaettu kahteen osaan: teoria- ja käytäntöosuuksiin. Teoriaosuudessa on tehty yleiskatsaus älytelevisioihin, niissä käytettäviin sovelluksiin, ja vertailtu keskenään niiden käyttämiä tekniikoita sovellusten kehittämiseen. Käytäntöosuudessa on sitten perehdytty demosovelluksen tekemiseen valitulla kehitysalustalla. Kehitysalustojen teknisen vertailun tarkoitus oli auttaa valitsemaan sopiva kehitysalusta demosovelluksen tekemiseen. Vertailun tarkoituksena oli myös perehtyä siihen, onko mahdollista tehdä yksi sovellus, joka toimisi useassa eri älytelevisiossa. Vaikka sovelluksien tekemiseen käytetään samaa teknologiaa eri kehitysalustoissa, niin ongelmaksi muodostuvat laitekohtaiset ohjelmointirajapinnat. Täten yhden älytelevisiovalmistajan alustalla tehty sovellus ei toimisi muiden valmistajien älytelevisioissa.

Kehitysalustojen vertailussa otettiin huomioon toimeksiantajaa kiinnostavat älytelevisioiden ominaisuudet. Täten ei ollut tarkoituksenmukaista käydä kaikkia kehitysalustojen yksityiskohtia läpi. Päätöksenteon helpottamiseksi on tehty power point -yhteenveto (liite 2). Yhteenvetoon on kerätty myös tietoja älytelevisiovalmistajista, joista ei ollut tietoja Excel-taulukossa (liite 1). Yhteenvetoon tarkoituksena oli tiivistää toimeksiantajaa kiinnostavat yksityiskohdat helpommin luettavaan muotoon.

Toimeksiantajan näkökulmasta alustan tarjoamia tärkeimpiä ominaisuuksia sovelluskehitykselle ovat kamera tuki sekä älypuhelimella ohjattavuus. Kamera tuen puuttuminen useasta kehitysalustasta helpotti päätöksentekoa. Samsung on ainoa joka tarjosi varmuudella päätöksentekohetkellä tuen kameralle kehitysalustassaan. Sen sijaan sovelluksen ohjaamiseen älypuhelimella löytyy tuki kaikista alustoista. Tästä syystä demosovellus päätettiin tehdä Samsungin kehitysalustalla.

5 SOVELLUS KEHITYSTYÖ ÄLYTELEVISION

Tässä luvussa kerrotaan Samsungin kehitysalustalla tehtävään demosovelluksen kehittämisprosessiin liittyvistä asioista. Kuten edellisessä kappaleessa todettiin, päätettiin demosovellus tehdä Samsungin kehitysalustalla. Kehitysalustaversioita oli useita tarjolla. Päätöksentekohetkellä viimeisin julkaistu versio oli SDK 3.5.2. Sovelluskehityksessä käytettiin aluksi SDK 3.5.2 -alustaa mutta myöhemmin kokeiltiin myös SDK 4 -kehitysalustaa.

5.1 Sovelluksen kehitysprosessi

Aluksi on hyvä käydä läpi, mitä eri vaiheita älytelevisiosovelluksen kehitysprosessiin kuuluu. Se on hyvin samantapainen kaikilla älytelevisiovalmistajilla. Tässä on kuvattu Samsungin käyttämä prosessi. Kehittäjäksi ryhtyminen vaatii rekisteröitymisen kehitysfoorum sivustolle. Rekisteröitymisen yhteydessä täytyy hyväksyä samalla valmistajan asettamat ehdot. Rekisteröityminen on myös vaatimuksena kehitysympäristön lataamiselle ja erilaisen tuen saamiselle. Samsung tarjoaa myös kumppanuutta sovelluksen kehittäjille. Kumppanuutta pitää anoa, ja kumppanuus tuo lisäetuja kehittäjään verrattuna. Rekisteröityminen on kaikilla kehitysfoorumeilla ilmainen mutta Panasonic perii vuosimaksun jäseniltään.

Rekisteröitymisen jälkeen sovelluskehittäjällä on oikeus ladata kehitystyökalut itselleen. Tämän jälkeen alkaa itse sovelluksen kehittämisvaihe. Kehitysalustojen etu on siinä, että ne pitävät sisälleen emulointimahdollisuuden. Emulointi mahdollistaa kehitettävän sovelluksen simulointia tietokoneessa. Kehitettävää sovellusta voi testata myös keskeneräisenä älytelevisiossa. Tämä onkin suotavaa, sillä emulointi tietokoneympäristössä ei koskaan vastaa oikeata käyttötilannetta. Lisäksi emulaattorit eivät tue kaikkia älytelevisio ominaisuuksia. Sovelluksen kehittämisen vaiheeseen kuuluu myös älytelevisiovalmistajan sovelluksille asettamien vaatimuksien läpi käyminen.

Sovelluskehityksen loputtua sovellukselle annetaan nimi ja yksilöllinen tunnus. Tätä varten sovellus pitää rekisteröidä Samsungin Seller Officessa. Seller Office vahvistaa rekisteröinnin sähköpostiviestillä, minkä jälkeen sovellus on ladattavissa Samsungin kehitysfoorumille. Tämän jälkeen Samsung evaluoi sovelluksen. Älytelevisiovalmistajilla on oma hyväksyntäprosessinsa, joka sovelluksen tulee läpäistä. Läpäistyään valmistajan hyväksyntäprosessin sovellus tulee näkyviin

valmistajan ylläpitämään sovelluskauppaan. Tämän jälkeen sovellus on ladattavissa käyttäjien toimesta televisiomalleilla, joilta löytyy tuki kehitetylle sovellukselle. (<http://www.samsungforum.com/Support/BeginningforSmartTV>, 12.2.2013.)

5.2 Kehitysympäristön asentaminen

Kehitysalustan lataaminen vaati rekisteröitymisen ja samalla Samsung-tilin luomisen. Koneessa johon kehitysalusta asennetaan pitää olla myös DirectX- ja Java SE -ohjelmistot asennettuina. Omassa koneessani nämä ohjelmat oli valmiiksi asennettu. SDK 3.5.2 -versiossa on vielä valittavana joko Samsungin IDE- tai Eclipse IDE -ympäristö. Itse valitsin Eclipse IDE -ympäristön, koska se oli tullut tutuksi aikaisemmillä ohjelmoinnin kursseilla.

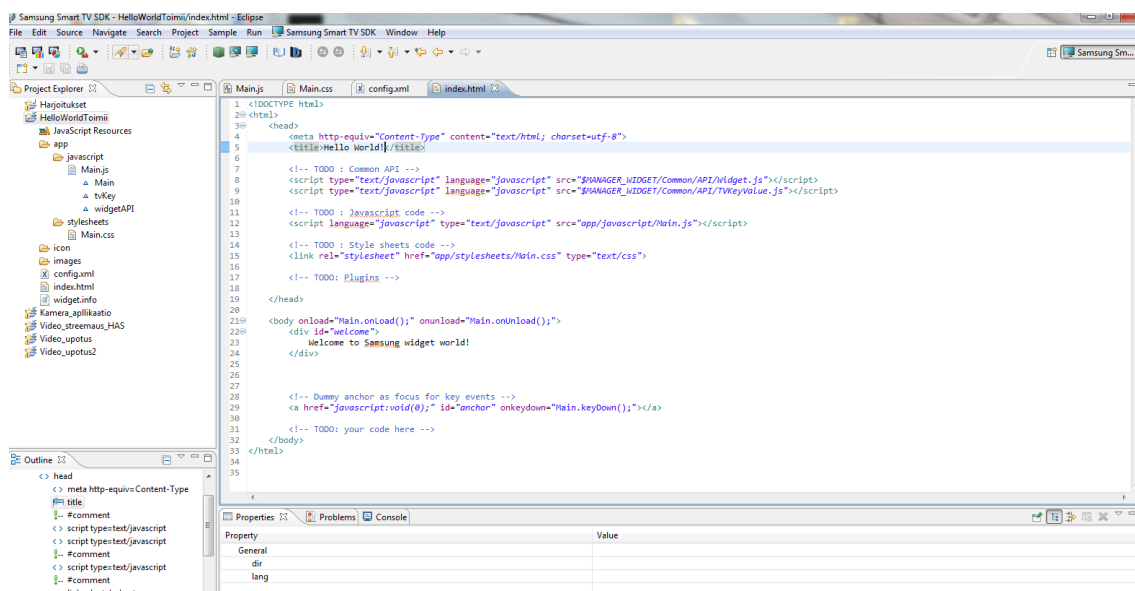
Ohjelmiston lataamisen jälkeen on sen asentamisen vuoro. Asennuspaketti sisältää Eclipse-ympäristön lisäksi kolme eri emulaattoria. SDK 3.5.2 -paketin emulaattorit emuloivat vuoden 2010, 2011 ja 2012 älytelevisiomalleja. SDK 4 -paketin emulaattorit taas vuoden 2011, 2012 ja 2013 älytelevisiomalleja. Asennusohjelma ehdottaa lopuksi myös Apache http-palvelimen asentamista koneelle. Palvelin on välttämätön, kun halutaan testata kehitettävää sovellusta oikeassa älytelevisiossa. Asensin palvelimen, vaikka en omista itse sellaista älytelevisiota, jossa oikeata sovellusta voisi testata.

Kun asennus on suoritettu, täytyy SDK käynnistää ”suorita järjestelmänvalvojana” komennolla. Ohjelmisto asentaa Samsungin tarvitsemat työkalut vasta ensimmäisen käynnistyksen yhteydessä. Ohjelma pitää muutenkin käynnistää järjestelmän valvojan oikeuksilla ainakin, jos käyttöjärjestelmä on Windows7, kuten itselläni. Työtiedostot tallennetaan käyttäjän hakemistoon, johon oikeudet ovat kunnossa. Kuitenkin, jos sovellusta testaa emulaattorissa, SDK yrittää paketoita sovelluksen oletusasetuksilla sellaiseen kansioon, johon normaalikäyttäjällä ei ole oikeuksia. Tein sen virheen aluksi, etten muistanut käynnistää ohjelmaa ohjeiden mukaisesti ja Samsung työkalut eivät olleet käytettävissä.

5.3 Kehitysympäristö

Samsung Eclipse IDE -kehitysympäristö on perusolemukseltaan samanlainen kuin normaali Eclipse-kehitysympäristö. Siihen on vain lisätty älytelevisiosovelluksen kehittämiseen vaadittavat ominaisuudet. Kehitysympäristön asentamisen jälkeen on hyvä testata kehitysympäristön toimivuus ”hello world” sovelluksella. Samsungin kehitysfoorum sivuilta löytyy tutor-ohjelma tällaisen sovelluksen luomiseen. Tutor-ohjelmasta löytyy myös mallitiedostot, jotka voi halutessaan suoraan ladata itselleen. Helpoin tie kehitysympäristön toiminnan testaamiseen onkin ladata valmis paketti omalle koneelle.

Hello world tutor -ohjelasovellus koostuu index.html-, main.js-, main.css- ja config.xml -tiedostoista. Index.html sisältää sovelluksen rakenteen. Main.js on JavaScript-tiedosto, ja sisältää sovelluksen kontrolliosan. Main.css määrittelee sovelluksen ulkoasun. Config.xml-tiedosto sisältää toimintaympäristön asetuksia. Config.xml-tiedostoa tarvitaan, jotta sovellusta voi käyttää älytelevisiossa. Edellä mainittujen tiedostojen lisäksi tarvitaan JavaScriptin ohjelmointirajapintakirjastot. Ne on kuitenkin asennettu koneelle valmiiksi kehitysympäristön asentamisen yhteydessä. Alla kuva kehitysympäristön perusnäköymästä (kuva 3).

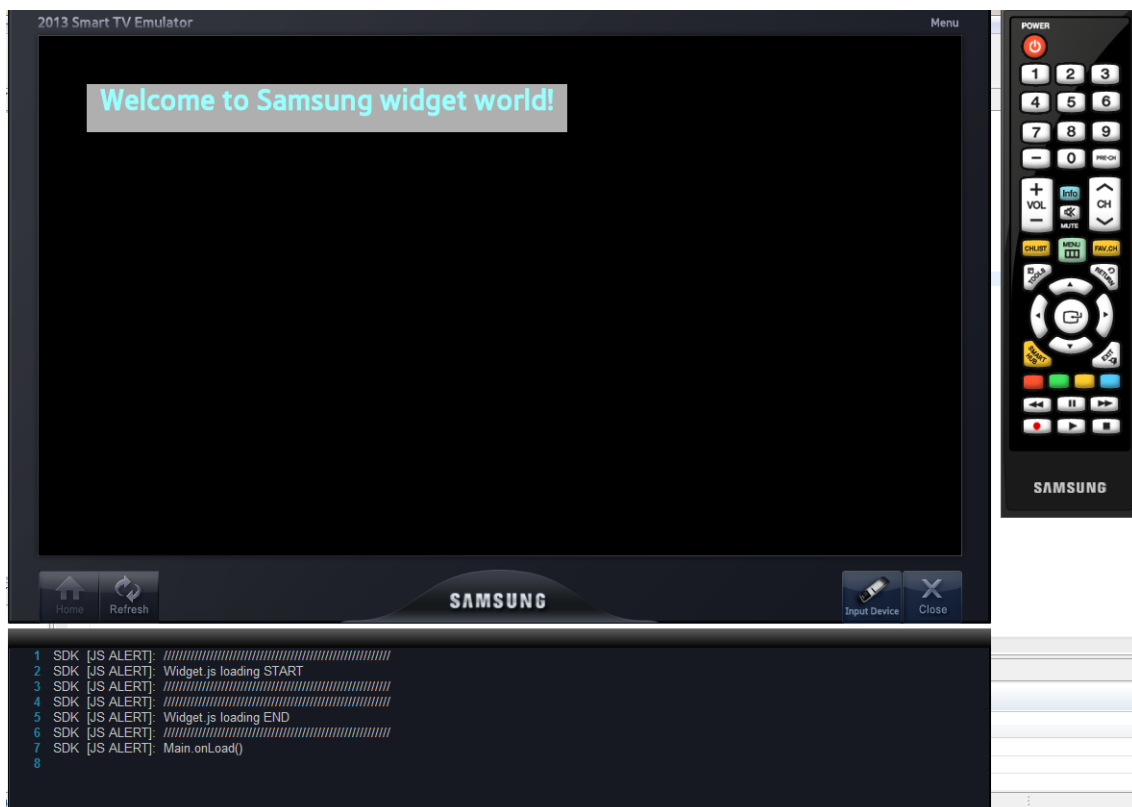


Kuva 3. Kehitysympäristön perusnäköymä

Sovelluksen kehittäminen aloitetaan luomalla uusi projekti. Uutta projektia luotaessa valitaan että kyseessä on Samsung Smart TV -sovellus. Valittavana on vielä, onko kyseessä perus-, JavaSc-

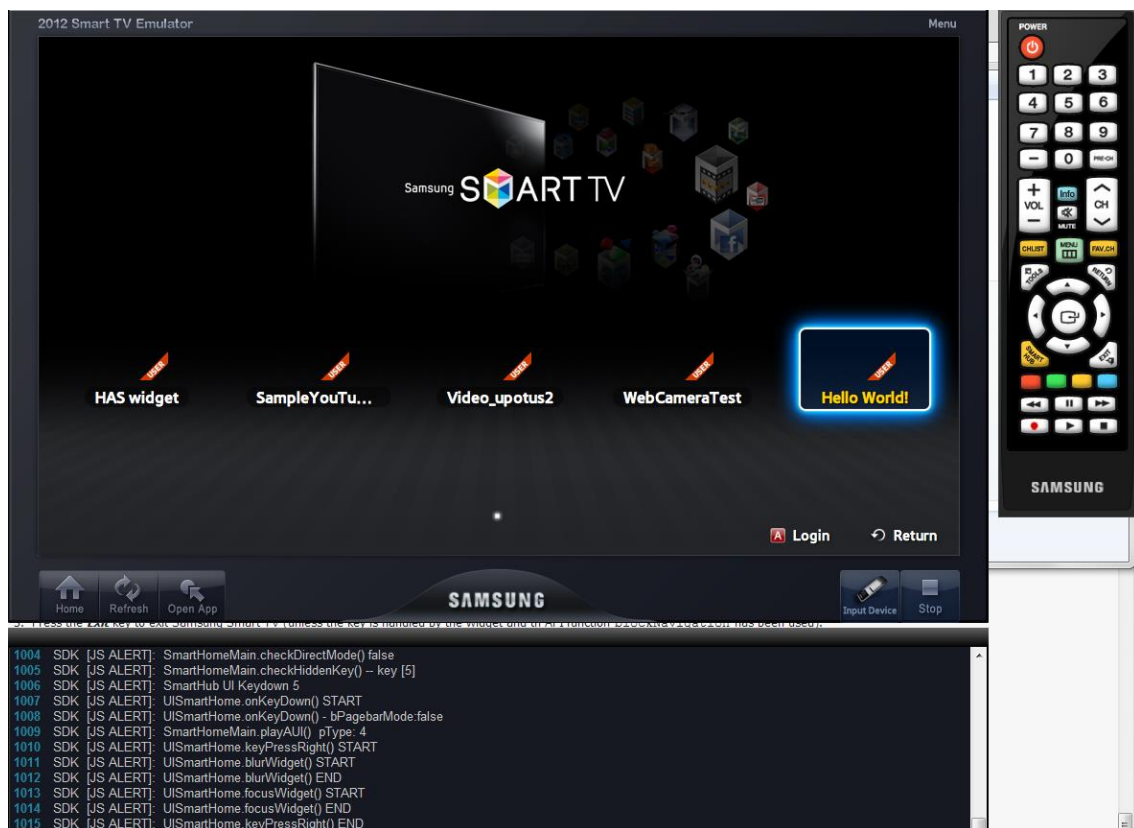
ript- tai Flash -sovellus. Basic-sovellus mahdollistaa visuaalisen editorin käytön. Flash-sovellus voi taas käyttää älytelevision flash playeria. Flash-sovelluksen toteutukseen on vielä kaksi eri tapaa. Flash-sovellus voi joko käyttää suoraan älytelevision flash playeria, tai sitten se käyttää flash playeria selainmoottorin liitännäisenä. Suositus on käyttää liitännäistä, sillä muuten sovellus ei voi hyödyntää laiteohjelmisto rajapintaa. Kehitysalustassa ei ole myöskään tukea pelkälle playerille mutta älytevisiolaitteesta tuki löytyy. Hello world -sovellusta luotaessa tyypiksi valitaan JavaScript-sovellus.

Tämän jälkeen kehitysympäristö luo tarvittavat tiedostot ja tiedostorakenteen automaattisesti projektikansioon. Hello world tutor -ohjelman ladatut koodit voi kopioida vastaaviin tiedostoihin. Ladatussa paketissa tiedostojen polkurakenne voi olla eri kuin projektikansioon luodun sovelluksen. Polkurakenteen tarkistamisen ja tarvittaessa muuttamisen jälkeen sovellus on valmis testattavaksi. Sovelluksen pystyy käynnistämään emulaattorissa Samsung Smart TV SDK -valikosta. Alla kuva emulaattorin ruudulta hello world -sovelluksen käynnistymisen jälkeen (kuva 4).



Kuva 4. Hello World -sovellus käynnistetty emulaattorissa

Älytelevisiosovelluksen voi testata myös toisella tavalla. Sovellus paketoidaan, kuten se paketoitaisiin oikeaan älytelevisioon testattavaksi. Sen jälkeen paketoitu sovellus kopioidaan emulaattorin valmiit sovellukset-kansioon. Emulaattori käynnistetään ja siirrytään emulaattorin home näkymään. Tämä näkymä vastaa älytelevisioon Smart Hub -sovellusvalikoimanäkymää. Home-näkymässä on kaikki emulaattorin sovellushakemistossa olevat sovellukset. Sovellus-ikonissa oleva user teksti tarkoittaa, ettei sovellus ole virallisesti hyväksytty. Liikkuminen sovellusten välillä tapahtuu kauko-ohjaimen nuolinäppäimillä. Haluttu ohjelma käynnistetään nuolinäppäinten välillä olevalla enter-näppäimellä. Alla on kuva emulaattorin home-näkymästä (Kuva 5).



Kuva 5. Emulaattorin home-näkymä

Kehitettävä sovellus pitää aina testata myös kohde älytelevisiossa. Älytelevisiossa on vähemmän muistia käytettävissä, joten älytelevisioon muisti voi loppua kesken. Myös sovelluksen graafinen ulkoasu ja näppäinten vasteajat pitää varmentaa oikeassa ympäristössä. Testaamista varten kehitysympäristökoneessa pitää olla asennettu palvelin. Asennuspaketti ehdottaakin asennuksen yhteydessä Apache-palvelimen asentamista koneelle. Kehitysympäristön asetuksiin pitää asettaa tämän jälkeen palvelinasetukset kohdalleen. Kehitettävä sovellus pakataan, ja config.xml-tiedoston asetuksissa sovelluksen tyyppiä valitaan "user". Kohde älytelevisioon asetukset pitää

myös asettaa kohdalleen. Verkkoasetukset muutetaan siten, että älytelevisio voi kytkeytyä kehitysympäristökoneen palvelimeen. Älytelevisioon pitää luoda developer-käyttäjätili. Developer-käyttäjätilin avulla saa älytelevisioon näkyville asetuksiin kehitysasetukset. Kehitysasetuksia muokkaamalla älytelevisio voi kytkeytyä kehitysalustakoneen palvelimeen ja ladata sovelluksen älytelevisioon.

5.4 Demosovellus

Kehitysympäristön toimivuus oli saatu siis varmistettua hello world -sovelluksella. Seuraavaksi asetettiin vaatimukset itse demosovellukselle. Tavoitteeksi asetettiin reaaliaikaisen videokuvan lähettäminen kahden älytelevisiosovelluksen välillä (live streamaus). Demosovelluksen käyttöliittymän ulkoasulle sen sijaan ei asetettu mitään tavoitteita. Reaaliaikaisen videokuvan lähettämiseen kahden älytelevisioon välillä käytetään streamaustekniikkaa. Lisäksi kehitysympäristön lisäksi tarvittaisiin kaksi älytelevisiota, joissa molemmissa olisi kamera. Käytettävissä ei ole kuitenkaan sellaista ympäristöä, johon demosovellusta ollaan tekemässä. Oli myös epäselvää, onko tavoitteeksi asetetun demosovelluksen toteuttaminen ylipäättänsä mahdollista.

Demosovelluksen tekeminen piti pilkkoa osiin. Ensimmäisessä vaiheessa oli tarkoitus saada videostreamaus onnistumaan emulaattorin ruudulle. Videostreamaukseen löytyy tutorohjelma Samsungin kehitysfoorumien sivuilta. Se sisältää myös valmiit koodit, jotka voi ladata omalle koneelle. Tutorohjelmassa käytettävät testivideot on toteutettu HAS- ja HLS -protokollia käyttäen. Tieto streamattavista videoista sijaitsee erillisessä XML (Extensible Markup Language) -tiedostossa. XML-tiedostossa ei ole suoraa linkkiä videoihin vaan index-listaan. Index-lista, joka sijaitsee palvelimella, sisältää tiedon streamattavan videon tiedostoista. Koska kyseessä on esitallennetun videon katsomista, puhutaan VOD-streamauksesta. XML-tiedostoa muokkaamalla voi testivideoiden lisäksi kokeilla itse tallennettujen videoiden streamausta omalta palvelimelta. Esitallennetun videon streamaus onnistuikin tutorohjelmalla.

Seuraavassa vaiheessa oli tarkoitus saada live-streamaus onnistumaan emulaattorin ruudulle. Tämä olikin lähtökohdaltaan astetta vaativampi tehtävä. Vaihtoehtoja puntaroidessa ensimmäisenä ajatuksena oli käyttää tietokoneen kameraa, josta video sitten streamattaisiin emulaattorin ruudulle. Samsungin kehitysfoorumien sivuilta löytyy tutorohjelma myös kameran käyttämiseen. Valmiit kooditkin oli ladattavissa. Vaikka kameran tutorohjelma käynnistyy emulaattorissa nor-

maalisti, sen käyttämä ohjelmointirajapinta ei kuitenkaan tue tietokoneen kameraa. Täten tutorohjelman kameran sovelluksella ei voinut toteuttaa video-streamausta emulaattorin ruudulle. Tutorohjelman koodi sinänsä oli toimiva emulaattorissakin. Kameran videokuvan sijasta sillä pystyy katsomaan esitallatettuja videoita.

Demo-ohjelman suunnittelu ja testaaminen kävivät tätä myötä entistä hankalammaksi. Seuraava idea oli testata live-streamausta käyttäen HLS-protokollaa. HLS-protokollaan pohjautuvasta streamauksesta on hyvät ohjeet Applen developer-sivustolla. Ideana oli edelleen käyttää oman tietokoneen kameraa, josta oli ajatus tehdä live video streamaus Applen työkaluilla. Applen työkalujen lisäksi tarvittaisiin tavallinen palvelin. Palvelin oli asennettuna koneelle. Applen sivustolla olevat työkalut olivat kuitenkin MAC-tietokoneille tarkoitettu. Täten niitä ei voinut asentaa omaan tietokoneeseen jossa on windows7-käyttöjärjestelmä.

Hakukoneiden avulla löytyi myös windows7-käyttöjärjestelmään suunniteltuja ohjelmia, joilla voi tehdä videosta HLS streamauksen. Tällaisia ohjelmia ovat muun muassa Aimersoftin Video Converter ultimate -ohjelma ja MarkelSoftin HTTP Live Streaming (HLS) Segmenter -ohjelma. Aimersoftin Video Converter kokeiluversiolla voi muuttaa videon TS (Transport Streaming) -muotoon. Ohjelmisto muuttaa kuitenkin ainoastaan videon formaatin, eikä segmentoi niitä pienimmiksi tiedostoiksi. Ohjelma ei myöskään tee index-tiedostoa. MarkelSoftin HTTP Live Streaming Segmenter -ohjelma sen sijaan segmentoi videon pienimmiksi tiedostoiksi. Se tekee myös tarvittavan index-tiedoston, joka sisältää tiedot segmentoiduista tiedostoista. MarkelSoftin kokeiluversiossa on asetettu raja videon pituudelle. Testaamista aikaraja ei kuitenkaan estänyt. Live-streamausta kummallakaan ohjelmistolla ei voi kuitenkaan toteuttaa.

HLS live videon testaaminen olisi sinänsä ollut mielenkiintoista. Kuten luvussa 3.2.1 on esitetty, streamattava video pitää ensin segmentoida pienemmiksi osiksi. Tämän jälkeen segmentoiduista tiedostoista pitää tehdä index-lista. Index-listan sisältö pitää päivittää jatkuvasti ja toimittaa se vastaanottopäähän. Vastaanottopää lataa sitten pilkotut tiedostot ja esittää ne jatkuvana ketjuna älytelevisiion ruudulla. Applen suositus segmenttien pituudeksi on 10 sekuntia. Segmenttien pituuden voi toki määritellä lyhyemmäksi. Esimeriksi suoraa TV lähetystä katsottaessa ei ole merkitystä sillä, onko lähetyksessä jonkinasteinen viive. Kaksisuuntaisessa kommunikaatiossa sen sijaan viiveellä on suuri merkitys. HLS vaatii kuitenkin toimiakseen tavallisen palvelimen. Älytelevisio ei voi toimia kuitenkaan palvelimena, joten HLS-protokolla ei sovellu videon jakamiseen.

Tässä vaiheessa alkoi näyttää jo tosi vaikealta toteuttaa demosovellusta opinnäytetyön puitteissa. Se, että live streamauksen olisi saanut onnistumaan tietokoneen kamerasta emulaattorin ruudulle, ei olisi vielä riittänyt. Live streamaus olisi pitänyt saada vielä lähtemään verkon yli toiselle laitteelle. Samsungin kehitysfoorumin dokumentaatiota tutkimalla selvisi, että kehitysalusta ei tarjoa tähän tarkoitukseen valmista ohjelmointirajapintaa. Toimeksiantajan kanssa pidetyn palaverin jälkeen päätettiin etsiä vielä muita ratkaisuja. Eräs mahdollisuus oli vielä tutkia, onko ulkopuolisen ohjelmointirajapinnan käyttäminen mahdollista älytelevisiossa. Tarpeellista teknistä tietämystä video streamauksesta ei kuitenkaan ollut. Siksi tässä vaiheessa piti perehtyä video streamaukseen tarkemmin. Lähestyin asiaa perehtymällä siihen, miten se on toteutettu kahden tietokoneen välillä. Tarkoitus oli selvittää, pystyykö samaa tekniikkaa hyödyntämään älytelevisiossa.

Etsin hakukoneen avulla ilmaisia sovelluksia, joilla voi siirtää videokuvaa kahden tietokoneen välillä. Tarkoitus oli selvittää, mitä tekniikkaa ja siirtoprotokollaa videon siirtämisessä käytetään. Ilmaisia Web-kamera sovelluksia löytyy paljonkin. Ilmaisissa sovelluksissa on kuitenkin yleensä rajoitteita maksullisiin versioihin verrattuna. Päädyin testaamaan Digi-Watcherin Watcher- ja RemoteView -ohjelmia. Ilmaisversio-ohjelmissa käyttöaika on rajattu 30 minuuttiin. Aikarajan umpeuduttua ohjelma on käynnistettävä uudelleen. Testausmielessä se ei ole ongelma. Asensin ohjelmat kahteen eri kotitietokoneeseen. Molemmissa tietokoneissa on web-kamerat.

Watcherin asetukset mahdollistivat ohjelmalla nauhoittamisen ja kuvien oton tietyn aikasyklein. Otetut kuvat voi tallettaa palvelimelle, josta niitä voi seurata selaimella. Watcherin asetuksissa asetetaan myös portti, joka mahdollistaa sovelluksen kytkeytyä verkon kautta Watcherin video streamiin. Videoinformaation tiedonsiirtoon käytetään TCP/IP-protokollaa. Watcher-ohjelma käyttää kuitenkin videossaan sen omaa DGW-formaattia. Formaatti ei ole tuettu Samsungin kehitysalustassa, joten sillä ei voi kytkeytyä Watcherin tuottamaan video streamiin. Testasin kytkeytymisen toiselta koneelta ainoastaan Digi-Watcherin omalla RemoteView-ohjelmalla. Testauksessa molemmat koneet olivat samassa lähiverkossa. Jos ohjelmia käyttää internetin yli, pitää oma lähiverkon reititin konfiguroida tarkoitusta varten. Tämä tutkimus ei tuonut ratkaisua demosovelluksen tekemiseen. Muita tutkimuksia en suorittanut ilmaisilla web-kamera sovelluksilla. (<http://www.digi-watcher.com/>, 7.2.2013)

Seuraavaksi etsin tietoa siitä, miten live video streamaus voidaan toteuttaa tietokoneen selainten avulla. Lähestyin asiaa etsimällä hakukoneen avulla demo-ohjelmia. Demo-ohjelmia löytyy useammanlaisia. <http://www.html5rocks.com/en/tutorials/getusermedia/intro/> sivulta löytyy yksinkertaisia demoja. Lisäksi sivun alaosasta löytyy linkkejä muihin demoihin. Demon testaamisessa käytin useampaa eri selainta. Demot toimivat parhaiten uusimmalla Chrome-selaimella. Yksinkertaisimmillaan oman tietokoneen videokuva streamataan ainoastaan oman selaimen ruudulle. Löytyy myös demo-ohjelmia, joissa selaimen ruudulle streamattua videokuva pystyy muokkaamaan. Tämä on mahdollista käyttäen HTML5-videoelementtiä sekä JavaScriptiä. Lisäksi löytyy myös demoja, joissa videokuva streamataan kahden tietokoneen selaimen välillä. Selainten välinen streamaus on toteutettu joko paikallisena tai internetin yli olevana streamauksena. Miellenkiinto kohdistuikin kahden selaimen internetin välityksellä toteutettuun ratkaisuun. Käyttävähän älytelevision selainmoottorit samoja teknologioita kuin tietokoneiden selaimet.

Selaimilla toteutetut demosovellukset pohjautuvat kahteen työn alla olevaan W3C:n spesifikaatioon. Ensiksi tarvitaan ohjelmointirajapinta, jolla selain voi kytkeytyä paikalliseen kameran tai mikrofoniin tuottamaan mediaan tai molempiin yhtäaikaaisesti. Tämä rajapinta on määritelty W3C:n "Media Capture and Streams" -spesifikaatiossa. Tästä rajapinnasta käytetään myös nimitystä getUserMedia API. Spesifikaatiossa määritellään myös ohjelmointirajapinta, jolla tuotettavaa mediaa voi muokata ja prosessoida JavaScript-koodin avulla. (<http://dev.w3.org/2011/webrtc/editor/getusermedia.html>, 19.1.2013.)

Lisäksi tarvitaan ohjelmointirajapinta, joka mahdollistaa selainten välisen kommunikaation. Tämä rajapinta on määritelty W3C:n "WebRTC 1.0: Real-time Communication Between Browsers" -spesifikaatiossa. Tästä rajapinnasta käytetään myös nimitystä WebRTC API. Spesifikaatiossa määritellään useita vaiheita, jotka liittyvät videoneuvottelun toteuttamiseen HTML:n välityksellä. Tarvitaan tekniikoita, kuten ICE (Interactive Connectivity Establishment), STUN (Session Traversal Utilities for NAT) ja TURN (Traversal Using Relays around NAT). Näiden teknikoiden avulla voidaan välittää palomuurien ja reitittimien takana olevien koneiden IP-osoitteiden tieto. Spesifikaatiossa määritellään myös, miten paikallisesti tuotettu media stream lähetetään ja vastaanotetaan. Siinä määritellään myös koneiden välille tarvittavan peer-to-peer -yhteyden luonti. Peer-to-peer -yhteydessä jokainen kone voi toimia sekä palvelimena että asiakkaana. Myös muun kuin mediatiedon lähettäminen peer-to-peer -yhteydellä on määritelty spesifikaatiossa. (<http://dev.w3.org/2011/webrtc/editor/webrtc.html>, 19.1.2013.)

Hyödyntämällä näitä kahta ohjelmointirajapintaa on mahdollista toteuttaa videopuhelu kahden selaimen välillä. Selaimen ei myöskään tarvitse käyttää liitännäisiä videon toistamiseen. Videopuhelu voidaan toteuttaa HTML5-videoelementillä sekä JavaScript-koodilla. Tietokoneiden selainten tuki tarvittaville ohjelmointirajapinnoille on kuitenkin puutteellinen vielä. Selaimilla on lisäksi omat nimet käytettäville rajapinnoille. JavaScriptin ehtolauseerakennetta hyödyntäen pystyy kuitenkin kiertämään helposti selainspesifiset eroavaisuudet. Samsungin älytelevisiossa tätä ratkaisua ei kuitenkaan voi sellaisenaan toteuttaa. Älytelevision selainmoottori ei tue tarvittavia vielä kehitysasteella olevia ohjelmointirajapintoja. On myös epätodennäköistä, että älytelevision selainmoottorit tulevat koskaan tukemaan näitä ohjelmointi rajapintoja.

HTML5-videoelementin toimivuus tuli kuitenkin testattua Samsungin uusimmassa kehitysalustassa. Video ei kuitenkaan käynnistynyt automaattisesti videoelementin autoplay attribuutilla. Videon käynnistäminen täytyikin toteuttaa JavaScript-koodin avulla käyttämällä tapahtumankäsittelijää. Kokeilumielessä upotin samaan ohjelmaan useamman videoelementin. Kahta yhtäaikaista videota ei kuitenkaan pystynyt toistamaan samanaikaisesti. Toiminta poikkesi tässä mielessä tietokoneen selaimen toiminnasta. Tämä asettaa myös rajoituksia sovelluksen tekemiselle.

Viimeisenä vaihtoehtona mietittiin, onko mahdollista käyttää Skype-ohjelman käyttämää rajapintaa. Älytelevision Skype-sovellus on toteutettu myös natiivi sovelluksena, joten sen käyttämää rajapintaa ei myöskään voinut hyödyntää. Tämän jälkeen tehtiin toimeksiantajan kanssa päätös että opinnäytetyön puitteissa tavoitteeksi asetettua demosovellusta ei voida toteuttaa.

6 TULOKSET

Web-tekniologiassa käytettävät HTML-, CSS- ja JavaScript -ohjelmointikielien ovat selvästi yleisin käytetty ratkaisu älytelevisioiden kehitysalustoissa. Yhdelle laitteelle suunniteltu sovellus ei kuitenkaan toimi sellaisenaan eri valmistajien laitteissa. Laitekohtaiset ohjelmointirajapinnat estävät tämän mahdollisuuden. Myös saman valmistajan laitteiden välissä on eroja. Tästä syystä kehitysalustaversioita löytyykin yhdeltä valmistajalta useita, eri vuosien älytelevisio-malleille. Sovelluskehittäjä voi joutua turvautumaan vanhempiin kehitysalustoihin ja ratkaisuihin, jos haluaa sovelluksensa toimivan myös vanhemmissa laitteissa. Älytelevisioon suunnitellut sovellukset eivät toimi sellaisenaan myöskään tietokoneen selaimilla.

Älytelevisioiden sovelluksia käytetään selainmoottorilla. Älytelevisiovalmistajilla on erilaisia ratkaisuja siihen, miten sovelluksia käytetään. Sovelluksen voi joko ladata älytelevisioon, tai sitten se voi sijaita pilvipalvelimella. Sovelluskehittäjä voi tarvittaessa päivittää pilvipalvelimella sijaitsevan sovelluksen, esimerkiksi siitä löytyvän virheen takia. Päivitetty sovellus on tämän jälkeen kaikkien käytettävissä. Vastaavasti älytelevisioon ladattavaan sovellukseen pitää rakentaa jonkinlainen tarkistusalgoritmi päivitystä varten. Tämän avulla sovellus voi esimerkiksi ilmoittaa käyttäjälle uudesta ladattavissa olevasta sovellusversiosta.

Älytelevisiosovelluksen kehittäjäksi ryhtyminen on sinänsä helppoa. Sovelluksen kehittäminen on mahdollista käyttäen älytelevisio valmistajien ilmaisia kehitysalustoja. Poikkeuksena on Panasonic, joka perii vuosimaksun kehittäjiltä. Kehitysalustoissa olevan emulaattorin avulla pystyy testaamaan myös kehitettävää sovellusta ilman että omistaa älytelevision. Käytännössä kehittäjäksi ryhtyjän on pakko testata sovellus myös oikeassa älytelevisio laitteessa. Pelkästään emulaattorissa testaaminen ei riitä, koska se ei vastaa todellista tilannetta. Emulaattorista ei myöskään löydy kaikkea tarvittavaa tukea, kuten kameran osalta tuli esille. Täten sovellus-kehittäjän tulee hankkia tarvittava älytelevisiolaitte itselleen.

HTML5-tuki mahdollistaa myös uusien web-ominaisuuksien käyttöönoton älytelevisioissa. Älytelevisioiden selainmoottoreilla voi olla kuitenkin erilainen toteutus ominaisuuksille laitteen erilaisesta luonteesta johtuen. Esimerkkinä HTML5-videoelementin toiminta. Tietokoneen selaimella pystyy toistamaan useaa eri mediaa yhtäaikaaisesti. Samsungin kehitysalustalla eikä älytelevisiossa-

kaan vastaavaa voida toteuttaa. Tämä rajoittaa sovellusten tekemistä tietyissä asioissa. Esimerkiksi videokuvaa ei voi toistaa käyttäen eri lähteestä tulevaa taustamusiikkia.

Älytelevision sovelluksilla on myös rajoituksia tietokoneen vastaaviin sovelluksiin verrattuna. Puutteellisen talletuskapasiteetin johdosta niihin ei voi tallentaa esimerkiksi ladattavia videoita. Toisaalta niillä ei pysty myöskään lataamaan palveluihin videoita tai kuvia. Kehitysalustat ja älytelevioliitteet muuttuvat kuitenkin nopeasti. Tämän opinnäytetyön kirjoittamisen aikana kaikilta vertailussa mukana olleilta älytelevioliitevalmistajilta tuli uudet kehitysalustaversiot. Tämä mahdollistaa yhä monipuolisempien sovelluksien kehittämisen.

Toimeksiantajan kanssa asetettua tavoitetta demosovelluksen kaksisuuntaiselle videoyhteydelle ei voitu toteuttaa opinnäytetyön puitteissa. Lähtökohtaisesti ongelmaksi muodostui oikean testiympäristön puuttuminen. Pelkän emulaattorin varassa aloitettu testaaminen pysähtyi alkuunsa puutteellisen kameratuen takia. Samsungin alusta ei myöskään tarjoa suoraa ohjelmointirajapintaa, joka mahdollistaisi videokuvan lähettämisen älytelevioliitteella. Koska älyteleviio ei voi toimia palvelinlaitteena, se asettaa myös rajoituksia video lähettämiseen älyteleviioista internetin yli.

7 POHDINTA

Opinnäytetyön tarkoituksena oli selvittää mitä eri tekniikoita käytetään älytelevisioiden sovellusten kehittämiseen. Tutkittiin onko myös mahdollista tehdä yhtä sovellusta, joka toimisi eri valmistajien laitteissa. Käytettäviä tekniikoita piti vertailla keskenään ja valita kehitysalusta demosovelluksen tekemiseen. Kehitysalustojen käyttämien tekniikoiden tiedot etsittiin älytelevisiovalmistajien kehitysfoorumien sivuilta. Tehdyn vertailun perusteella selvisi, ettei voi toteuttaa yhtä sovellusta, joka toimisi eri valmistajien laitteissa. Demosovelluksen tekemiseen valittiin Samsungin kehitysalusta. Tavoitteeksi asetettua kaksisuuntaista videoyhteyttä älytelevisio sovellusten välillä ei kuitenkaan voitu toteuttaa opinnäytetyön puitteissa. Samsungin kehitysalustasta ei löytynyt valmista ohjelmointirajapintaa, joka olisi mahdollistanut videon lähettämisen televisiosta.

Opinnäytetyön aiheen valintahetkellä opinnäytetyön sisältö ei ollut tarkkaan tiedossa. Alun perin oli tarkoitus tutkia pelkästään Samsung älytelevisio mahdollisuuksia ja rajoituksia sovelluskehitykselle. Tutkimuksen kuluessa aihe muuttui laajemmaksi, ja opinnäytetyön teoriapohjaksi tuli vertailla eri älytelevisiovalmistajien kehitysalustoja keskenään. Kehitysalustojen vertailu oli haastavaa. Älytelevisiovalmistajilla on useita eri versioita kehitysalustoista. Lisäksi älytelevisiovalmistajilla on omat tapansa ilmoittaa tekniset tiedot. Älytelevisiolaittevalmistajien kehitysalustojen lisäksi on olemassa palvelun tarjoavia kehitysalustoja, kuten Google TV, Yahoo Connected TV ja Espial Browser TV. Tähän kun vielä lisää eri tekniikoista käytettävän teknisen termistön, tuli paljon uutta sisäistettävää asiaa.

Kehitysalustojen vertailun helpottamiseksi aloin kerätä teknisiä tietoja erilliseen Excel-taulukkoon. Taulukkoon tuli kerättyä valmistajien ilmoittamat tiedot siinä muodossa kuin ne oli ilmoitettu kehitysfoorumien sivuilla. Excel-taulukon luettavuus ei kuitenkaan ollut ihan mutkatonta suuren tietomäärän vuoksi. Jälkikäteen mietittynä olisi alun perin ollut parempi määrittää ne tiedot, joita haetaan, eikä kerätä kaikkea ilmoitettua tietoa. Tietojen keräämisen yhteydessä tulikin mietittyä, onko kaikella tiedolla merkitystä vertailun kannalta. Jossain vaiheessa alkoi tuntua, ettei tietojen kerääminen Excel-taulukkoon ollut mielekästä. Tästä syystä tulikin myöhemmin tehtyä erillinen yhteenveto. Yhteenvedossa keskityttiin pelkästään oleellisten asioiden vertailemiseen.

Aiheen rajauksen onnistumista olen pohtinut myös useamman kerran. Varsin aikaisessa vaiheessa oli tiedossa, mitä demosovellukselta tullaan haluamaan. Kuitenkin opinnäytetyön teoriapohjaksi valittiin kehitysalustojen keskinäinen vertailu. Olisiko ollut parempi alun perin keskittyä pelkäämään Samsungin kehitysalustaan, ja teoriapohjaksi ottaa video streamaukseen perehtyminen kehitysalustojen keskinäisen vertailun sijasta? Demosovelluksen lopputuloksen kannalta se ei kuitenkaan välttämättä olisi tuonut ratkaisua. Puuttuihan Samsungin kehitysalustasta tarvittava suora ohjelmointirajapinta videon lähettämiseen televisiosta. Tämä olisi tosin voinut selvitä paljon aikaisemmassa vaiheessa ja johdattanut muunlaisen ratkaisun etsimiseen. Toisaalta ilman keskinäistä vertailua olisi jäänyt älytelevisiokentän kokonaiskuva hahmottumatta. Itselleni tuli ainakin parempi käsitys siitä, minkälaisia erilaisia mahdollisuuksia älytelevisiolaitteet mahdollistavat sovellusten kehittämiseen.

Onko sitten yleensäkin mahdollisuutta toteuttaa videon lähettäminen televisiosta puuttuvan suoran ohjelmointi rajapinnan takia? Miten Skype-videopuhelu on sitten toteutettu? Mitä mahdollisuuksia JavaScript tarjoaa? Onhan Samsungin kehitysalustassa kuitenkin tuki kameralle. Näitä kysymyksiä nousi pintaan mahdollisia vaihtoehtoja puntaroidessa. Opinnäytetyöhön käytettävä aika on kuitenkin rajallinen. Toisaalta asia kiehtoi itseä kovasti. Olihan kysymyksessä tekninen ongelma, joka on itselleni suuri motivoiva tekijä. Puuttuvan oikean testiympäristön johdosta painu duin tietokoneiden välillä tehtävään video-streamaukseen. Kaikkia kokeiluja ei ole edes kirjattu opinnäytetyöhön. Välillä mietinkin tutkiessani tietokoneiden välistä video-streamausta, että onko yhteys opinnäytetyön aiheeseen liian ohut.

Omasta mielestäni oli perusteltua kuitenkin tutkia tietokoneiden välistä video-streamausta. Varsinkin mahdollisuus selainten väliseen streamaukseen käyttäen kehitteillä olevia ohjelmistorajapintoja oli kiinnostava. Älytelevisioiden sovellukset toimivat selainmoottoreilla ja käytettävä teknologia on sama. Myös kehitysalustoista tuli uusia versioita opinnäytetyön kirjoittamisen aikana. Samsungin uusin kehitysalusta mahdollistaa ulkopuolisten web-ohjelmointirajapintojen hyödyntämisen. Uudesta alustasta löytyy myös tuki web-socketin käyttämiseen. Löytyisikö näiden tekniikoiden avulla ratkaisu kaksisuuntaisen videon toteuttamiseen älytelevisiosovellusten välillä? Toisaalta toteuttamiseen tarvitaan JavaScript-osaamista. Oma JavaScript osaaminen perustui opintojen aikana tehtyihin pieniin demoihin. Opinnäytetyön aikana sain myös siitä lisää arvokasta tietämystä. JavaScript-kielen hallitseminen ja sen tuomien mahdollisuuksien käyttäminen on oleellista älytelevisiosovellusten kehittäjälle.

Opinnäytetyön uutuusarvoa pohdittaessani mielestäni aiheeni on hyvin ajankohtainen. Uskon että opinnäytetyön toimeksiantaja sai myös arvokasta tietoa älytelevisioiden sovelluskehityksestä sekä myös rajoituksista. Kirjallisuutta älytelevisio sovelluksen kehittämiseen ei ole saatavilla. Toisaalta tällainen kirja vanhenisi melko nopeasti, sillä tekniikka menee nopeaa vauhtia eteenpäin. Selainten välisestä videoyhteyden muodostamisesta on kuitenkin julkaistu kirja vuoden 2012 lopussa (<http://www.webrtcbook.com/>). Kirjan aihe on kuitenkin viitekehyksen ulkopuolella.

Opinnäytetyön tekeminen oli pitkä prosessi. Itse valitsin opinnäytetyön kirjoittamisen tietoisesti siihen vaiheeseen, että voin keskittyä sen tekemiseen täysillä. Opinnäytetyötä tehdessä tuli koettua niin onnistumisen kuin epäonnistumisen hetkiä. Monta kiveä tuli käännettyä etsiessäni ratkaisua demosovelluksen tekemiselle. Opinnäytetyön puitteissa ei tullut kuitenkaan otettua yhteyttä laitevalmistajaan. Olisiko Samsung kertonut kaksisuuntaisen videoyhteyden mahdollisuudesta avoimesti? Kuten kaikkiin oppimisen prosesseihin kuuluu, törmäsin tässäkin prosessissa tuttuun ilmiöön. Mitä enemmän aiheesta alkoi tietää, huomasi, kuinka paljon on vielä opittavaa.

LÄHTEET

Kirjat

Collison, S & Budd, A & Moll, C. 2007. CSS – Tehokas hallinta. 1. painos. Jyväskylä: Gummerus.

Korpela, J. 2011. HTML5 – Uudet ominaisuudet. Jyväskylä: Docendo.

Korpela, J. & Linjama, T. 2005. Web-suunnittelu. 1. painos. Jyväskylä: Docendo.

Minoli, D 2012. Linear and nonlinear video and tv applications: Wiley

Smith, D & Negrino, T. JavaScript – Tehokas hallinta. 1. painos. Jyväskylä: Gummerus.

Digitaaliset lähteet

Adobe AIR 3 / FAQ. Hakupäivä 22.2.2013. <http://www.adobe.com/fi/products/air/faq.html>.

Android Application Fundamentals. Hakupäivä 7.12.2012,
<http://developer.android.com/guide/components/fundamentals.html>.

Apple HTTP Live Streaming. Hakupäivä 16.1.2013,
http://developer.apple.com/library/ios/#documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/Introduction/Introduction.html#//apple_ref/doc/uid/TP40008332-CH1-DontLinkElementID_39.

Communication from Remote Device to Google TV. Hakupäivä 21.12.2012.
<https://developers.google.com/tv/remote/docs/communication?hl=fi#discovery>.

Digi-Watcher kotisivut. Hakupäivä 7.2.2013, <http://www.digi-watcher.com/>.

Google TV Pairing Protocol. Hakupäivä 21.12.2012.
<https://developers.google.com/tv/remote/docs/communication?hl=fi>.

How to develop for Google TV. Hakupäivä 7.12.2012, <https://developers.google.com/tv/>.

LG Smart TV UDAP Specifications. Hakupäivä 21.2.2013.
http://developer.lgappstv.com/TV_HELP/index.jsp?topic=%2F%2Fge.tvsdk.references.book%2Fhtml%2FUDAP%2FUDAP%2FLG+UDAP+2+0+Protocol+Specifications.htm.

Microsoft IIS Smooth Streaming. Hakupäivä 23.2.2013,
<http://www.iis.net/downloads/microsoft/smooth-streaming>.

Nyxio kotisivut. Hakupäivä 28.12.2012, <http://www.nyxio.com/>.

Open IPTV forum. Hakupäivä 23.2.2013, <http://www.oipf.tv/>.

Samsung, D TV Developers FORUM. Hakupäivä 28.12.2012,
<http://www.samsungdforum.com/Guide/art00005/index.html>.

Samsung D TV Developers FORUM, Beginning for Smart TV. Hakupäivä 12.2.2013,
<http://www.samsungdforum.com/Support/BeginningforSmartTV>.

Yahoo Connected TV Remote Device. Hakupäivä 12.2.2013,
http://developer.yahoo.com/connectedtv/kontxapiref/YCTV_KONTX_Remote_Device.html.

Widevine video optimization. Hakupäivä 23.2.2013,
http://www.widevine.com/video_optimization.html.

Wikipedia Dynamic Adaptive Streaming over http. Hakupäivä 23.2.2013,
http://en.wikipedia.org/wiki/Dynamic_Adaptive_Streaming_over_HTTP.

Wikipedia JavaScript engine. Hakupäivä 22.2.2013,
http://en.wikipedia.org/wiki/JavaScript_engine.

Wikipedia H.264/MPEG-4 AVC. Hakupäivä 18.1.2013, http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC.

Wikipedia Real Time Messaging Protocol. Hakupäivä 22.2.2013.
http://en.wikipedia.org/wiki/Real_Time_Messaging_Protocol.

Wikipedia Universal Plug and Play. Hakupäivä 21.2.2013,
http://en.wikipedia.org/wiki/Universal_Plug_and_Play.

W3C Media Capture and Streams. Hakupäivä 19.1.2013,
<http://dev.w3.org/2011/webrtc/editor/getusermedia.html>.

W3C WebRTC 1.0: Real-time Communication Between Browsers. Hakupäivä 19.1.2013,
<http://dev.w3.org/2011/webrtc/editor/webrtc.html>.

Platform Features		Feature		Description				
		OS	Linux 2.6	Graphic Resolution	App: 960 x 540, 32 bpp Smart Hub: 1280 x 720 Video: 1920 x 1080 32 bpp	Samsung smart tv		
Type	Feature	2010		2011		2012		
		TV/AV	SDK 1.5	TV/AV	SDK 2.5	TV/AV	SDK 3.0	
APP ENGINE	HTML	HTML 4.01	HTML 4.01	HTML 5	HTML 5	HTML 5	HTML 5	
	DOM	DOM 2	DOM 2	DOM 2	DOM 2	DOM 3	DOM 3	
	CSS	CSS 2.1	CSS 2.1	CSS 3	CSS 3	CSS 3	CSS 3	
	Javascript	Javascript 1.6	Javascript 1.6	Javascript 1.8	Javascript 1.8	SquirrelFish	SquirrelFish	
Flash	Browser Plug in	SWF	Flash8.0/FlashLite3.1/ ActionScript2.0	Flash8.0/FlashLite3.1/ ActionScript2.0	Flash8.0/FlashLite3.1/ ActionScript2.0	Flash10.1/ ActionScript2.0/3.0	Flash 10.1 / ActionScript 3.0	
	Standalone	SWF	Flash8.0/FlashLite3.1/ ActionScript2.0	Flash8.0/FlashLite3.1/ ActionScript2.0	Flash8.0/FlashLite3.1/ ActionScript2.0 Flash10.1 / ActionScript3.0(TV D6000 or higher, BD D6700 or higher only)	Flash10.1/ ActionScript 3.0	Flash 10.1 / ActionScript 3.0	
	AIR	AIR	Not Supported	Not Supported	AIR for TV 2.5.1	Not Supported	AIR for TV 2.5.1 (TV D6000 or higher, BD D6700 or higher only)	
	Streaming	Streaming	RTMP/RTMPE	RTMP/RTMPE	RTMP/RTMPE	RTMP/RTMPE	RTMP/RTMPE	
DRM	WMDRM 10 PD	Supported	Not Supported	Replaced with (PlayReady)	Not Supported	Replaced with (PlayReady)	Not Supported	
	Marlin-BB	Not Supported	Not Supported	Supported	Not Supported	Supported	Not Supported	
	PlayReady	Not Supported	Not Supported	Supported	Not Supported	Supported	Not Supported	
	Widevine	Supported	Not Supported	Supported	Not Supported	Supported	Not Supported	
VOD	Streaming	HTTP	Supported	Supported	Supported	Supported	Supported	
		HTTPS	Supported	Supported	Supported	Supported	Supported	
		MMS	Supported	Supported	Supported	Supported	Supported	
		RTP/RTSP	Not Supported	Not Supported	Not Supported	Not Supported	Supported	
	Adaptive streaming	HAS (xml metadata)	Supported - OIPF Rel2 / MPEG2 TS only / AES-128	Not Supported	Supported - OIPF Rel2 / MPEG2 TS only / AES-128	Supported	Not Supported	Not Supported
		Mpeg-Dash (xml metadata)	Not Supported	Not Supported	Not Supported	Not Supported	Supported OIPF Rel2 / MPEG2 TS only / AES-128	Supported
		HLS (m3u8 metadata)	Supported - OIPF Rel2 / MPEG2 TS only / AES-128	Not Supported	Supported - OIPF Rel2 / MPEG2 TS only / AES-128	Supported	Supported OIPF Rel2 / MPEG2 TS only / AES-128	Supported
		MMSH						
		RTMP/RTMPE						
		Widevine						
Live Streaming	Adaptive Streaming	MS smooth						
		WMS						
		HLS audio	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported
		HAS LS	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported
Live Streaming	Adaptive Streaming	HLS LS	Not Supported	Not Supported	Supported - HLS v3 / MPEG2 TS only / AES-128	Supported	Supported - HLS v3 / MPEG2 TS only / AES-128	
		WMS						
		WidevineLS	Supported	Not Supported	Supported	Not Supported	Supported	Not Supported

Platform Features		Samsung smart tv					
Type	Feature	2010		2011		2012	
		TV/AV	SDK 1.5	TV/AV	SDK 2.5	TV/AV	SDK 3.0
Browser Engine more features	Core						
	AJAX						
	JSON						
	Cookie						
	XML						
	XSLT						
	XPath						
	Canvas						
	XVHLHttpRequest						
	Java						
userAgent String							
Image	JPEG						
	PNG						
	GIF						
	BMP						
Subtitle	SAMI	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported
	CineCanvas	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported
	GIF	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported
	Popcorn library	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported
Play List	ASX	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported	Not Supported
Input Device	Traditional TV RCU						Supported
	Magic RCU (pointing device)						
	Smart phone						Smart Text
	HID/Xinput game pads (USB keyboard)						Supported
Media Player	player						
	HTML5 video/audio						
HTTPS	Server certificate						
	Client certificate						
Font	Font type	SDK allows users to add new fonts to the application	SDK allows users to add new fonts to the application	SDK allows users to add new fonts to the application	SDK allows users to add new fonts to the application	SDK allows users to add new fonts to the application	SDK allows users to add new fonts to the application

Hardware Features							
Type	Feature						
HW	Location						
	USB						Supported
	WiFi						
	Bluetooth						
	Camera						Supported
	Locatio (GPS)						
	Microphone						Supported
	NFC						
	Sensors						
	Telephony						
Touchscreen							
SW	OpenGL for Java						
	Application Widgets						
	System feature						
	SIP/VOIP						
SW testaus smart TV:ssä						Supported, SMART HUB needed	Supported, SMART HUB needed

Platform Features		Feature Description		Feature Description			Feature Description		
		OS	Linux	OS	1280 X 720 V.L.U At least RGB888, 1b1bits v.z.u onwards RGB8888 27kbit		OS	Android Honeycomb MR1	
		Graphic Resolution	1.280 X /2U Recommended	Graphic Resolution				Graphic Resolution	720p and 1080p
		LG smart tv		Philips, Sharp smart tv			Google TV		
Type	Feature	NetCast 2.0 (2011) SDK V2.2.1	NetCast 3.0 H12 (2013) SDK V2.2.1	NetTv v1.0 SDK v.x	NetTv v2.0 SDK v.x	NetTv v3.1 SDK v.x	Web pages	Android app	
APP ENGINE	HTML	HTML 4.1 HTML5 partly XHTML s supported	HTML 4.1 HTML5 partly XHTML s supported	XHTML 1.0	XHTML 1.0	XHTML 1.0	HTML5 partly		
	DOM	DOM 2 DOM 3 partly	DOM 2 DOM 3 partly	DOM level 2	DOM level 2	DOM level 2	Supported		
	CSS	CSS 2.1 CSS 3 partly	CSS 2.1 CSS 3 partly	CSS TV Profile 1.0 / CSS 3 partly	CSS TV Profile 1.0 / CSS 3 partly	CSS TV Profile 1.0 / CSS 2 partly / CSS 3 partly	CSS 3		
	Javascript	SquirrelFish Extreme	SquirrelFish Extreme	ECMA-262 (Ecma script, Javascript 1.5)	ECMA-262 (Ecma script, Javascript 1.5)	ECMA-262 (Ecma script, Javascript 1.5)	Supported		
Flash	Browser Plug in	SWF					Adobe Flash Player 10.2 Beta Flash 10.1		
	Standalone	SWF	Flash Lite 3.1 / ActionScript 2.0	Flash Player 10.1 Action Script 2.0					
	AIR	AIR	Not Supported	AIR 3.0 for TV / ActionScript 3.0					
	Streaming	Streaming	HTTP(s) RTMP/RTMP-e	HTTP(s) RTMP/RTMP-e					
DRM	WMDRM 10 PD	Supported	Supported	Not Supported	Not Supported	Supported	Supported		
	Marlin-BB			Not Supported	Supported	Supported			
	PlayReady	Not Supported	Supported						
	Widevine	Supported	Supported						
VOD	Streaming	HTTP	Supported	Supported	Supported	Supported	Supported	Supported	
		HTTPS	Supported	Supported				Supported	
		MMS							
		RTP/RTSP							Supported
	Adaptive streaming	HAS (xml metadata)	Supported	Supported					
		Mpeg-Dash (xml metadata)			Not Supported	Not Supported	Supported		
		HLS (m3u8 metadata)	Supported	Supported	Not Supported	Not Supported	Supported for 2012+ devices onwards (WMS- HTTP no longer supported in 2012+ devices)	Supported	Supported
		MMSH	Supported	Supported					
		RTMP/RTMPE	Supported	Supported					
		Widevine	Supported	Supported					
MS smooth									
WMS			Supported	Supported	Not Supported				
HLS audio	Not Supported	Not Supported							
Live Streaming	Adaptive Streaming	HAS LS							
		HLS LS	Supported	Supported			Supported for 2012+ devices onwards (WMS- HTTP no longer supported in 2012+ devices)	Supported	Supported
		WMS			Supported	Supported	Not Supported		
		WidevineLS	Supported	Supported					

Platform Features		LG smart tv		Philips, Sharp smart tv			Google TV	
Type	Feature	NetCast 2.0 (2011)	NetCast 3.0 HTML2 (2013)	NetTV v1.0	NetTV v2.0	NetTV v3.1	Web pages	Android app
		SDK V2.2.1	SDK V2.2.1	SDK v.x	SDK v.x	SDK v.x		
Browser Engine more features	Core	Webkit 531.2+	Webkit 534.26+	CE-HTML	CE-HTML	CE-HTML	Google Chrome web browser	
	AJAX	Supported	Supported				Supported	
	JSON	Supported	Supported				Supported	
	Cookie	Supported	Supported					
	XML	Supported	Supported					
	XSLT	Supported	Supported				Supported	
	XPath	Supported	Supported					
	Canvas	Supported	Supported					
	XMLHttpRequest (AJAX)	Supported	Supported					
	Java	Not Supported	Not Supported					
userAgent String	Supported	Supported	Not Supported	Not Supported	Supported			
Image	JPEG	Supported	Supported	Supported	Supported	Supported	Supported	Supported
	PNG	Supported	Supported	Supported	Supported	Supported	Supported	Supported
	GIF	Supported	Supported	Supported	Supported	Supported	Supported	Supported
	BMP						Supported	Supported
Subtitle	SAMI	Supported	Supported	Not Supported	Not Supported	Supported		
	CineCanvas	Supported	Supported					
	GIF	Supported	Supported					
	Popcorn library	Not Supported	Not Supported					
Play List	ASX	Supported	Supported	Not Supported?	Not Supported?	Not Supported?		
Input Device	Traditional TV RCU	Supported	Supported	Supported	Supported	Supported		Supported
	Magic RCU (pointing device)	Supported	Supported	New platforms	New platforms	New platforms	Supported	Supported
	Smart phone	Smart Text	Smart Text			Supported		
	HID/Xinput game pads (USB keyboard)	Not Supported	Supported					Supported
Media Player	player	LG media object	LG media object					
	HTML5 video/audio	Supported	Supported				Supported	
HTTPS	Server certificate	Supported	Supported					
	Client certificate	Supported	Supported	Not Supported	Not Supported	Supported		
Font	Font type	LG Display font / Tiresias	LG Display-Regular preferred / Tiresias	Tiresias / Letter Gothic 12 pitch	Tiresias / Letter Gothic 12 pitch	Tiresias / Letter Gothic 12 pitch / Akhbar/MTOTS	Droid Sans / Droid Serif families	Droid Sans / Droid Serif families
Hardware Features								
Type	Feature							
HW	Location							Supported
	USB							Supported (host)
	WiFi					New platforms		Supported
	Bluetooth							Not supported
	Camera							Not supported
	Locatio (GPS)							Not supported
	Microphone							Not supported
	NFC							Not supported
	Sensors							Not supported
	Telephony							Not supported
SW	Touchscreen							Not supported
	OpenGL for Java							Supported
	Application Widgets							Supported
	System feature SIP/VOIP							Supported
								Not supported
SW testaus smart TV:ssä		Supported, via USB, SDK 1.5 =>	Supported, via USB, SDK 1.5 =>	Not Supported, only by validation tool	Not Supported, only by validation tool	Not Supported, only by validation tool	Supported, via IP connection by ADB	Supported, via IP connection by ADB

Platform Features		Feature Description		Feature Description		Feature Description		
		Feature	Description	Feature	Description	Feature	Description	
		OS	640 x 360 — (16:9 medium quality) 640 x 480 — (4:3 medium quality) 720 x 480 — (high quality)	OS	1280 X 720, 32bit	OS		
		Graphic Resolution		Graphic Resolution		Graphic Resolution		
		Yahoo connected TV		Smart TV Alliance v2 (LG, Philips)		Espial TV Browser		
Type	Feature	Feature	Description	Feature	Description	Feature	Description	
APP ENGINE	HTML	XML		HTML 1.1 HTML5 partly		HTML5		
	DOM			DOM level 2		DOM2, DOM 3		
	CSS	Supported		CSS 2.1 CSS 3 partly		CSS3, CSS 3D		
	Javascript	Supported		ECMA-262 (Ecmascript, Javascript 1.5)		JavaScript 1.5		
Flash	Browser Plug in	SWF				Adobe Flash 10.1		
	Standalone	SWF						
	AIR	AIR						
	Streaming	Streaming						
DRM	WMDRM 10 PD		Supported					
	Marlin-BB			Supported		Supported		
	PlayReady			Optional				
	Widevine							
VOD	Streaming	HTTP	Supported		Supported		Supported	
		HTTPS			Supported			
		MMS	Supported					
		RTP/RTSP	Supported					
	Adaptive streaming	HAS (xml metadata)						
		Mpeg-Dash (xml metadata)			Supported		Supported	
		HLS (m3u8 metadata)	Supported		Supported		Supported	
		MMSH						
		RTMP/RTMPE						
		Widevine						
		MS smooth			Supported		Supported	
		WMS						
HLS audio								
Live Streaming	Adaptive Streaming	HAS LS						
		HLS LS	Supported		Supported		Supported?	
		WMS						
		WidevineLS						

Platform Features		Yahoo connected TV	Smart TV Alliance v2 (LG, Philips)	Espial TV Browser
Type	Feature			
Browser Engine more features	Core			WebKit-based core
	AJAX		Supported	Supported
	JSON			
	Cookie		Supported	
	XML		Supported	
	XSLT			
	XPath			
	Canvas		Supported	Supported
	XHTML requests (AJAX)		Partly	?
	Java			
userAgent String		Supported	?	
Image	JPEG	Supported	Supported	
	PNG	Supported	Supported	
	GIF	Supported	Supported	
	BMP			
Subtitle	SAMI			Subtitle supported
	CineCanvas			
	GIF			
Popcorn library		Supported		
Play List	ASX			
Input Device	Traditional TV RCU	Supported	Supported	
	Magic RCU (pointing device)			
	Smart phone	Supported		
	HID/Xinput game pads (USB keyboard)			
Media Player	player	KONtx mediaplayer		
	HTML5 video/audio		Supported	Supported
HTTPS	Server certificate		Supported SSL / TLS	Supported, SSL
	Client certificate			
Font	Font type		Tiresias	
Hardware Features				
Type	Feature			
HW	Location			
	USB			
	WiFi			
	Bluetooth			
	Camera			
	Locatio (GPS)			
	Microphone			
	NFC			
	Sensors			
	Telephony			
Touchscreen				
SW	OpenGL for Java			
	Application Widgets			
	System feature			
	SIP/VOIP			
SW testaus smart TV:ssä			LG Supported, Philips not supported	



Teknologia vertailun yhteenvedo

Petri Mourujärvi

30.12.2012

Sovelluskehitys älytelevisioon

1

Samsung Smart TV

- **Pääteknologia: HTML5+CSS3+JavaScript**
 - SDK 2.5 ja televisio 2011 eteenpäin
- **Live streemaus tuettu (HLS&Widevine)**
- **Sovellus ladataan televisioon ja käytetään selaimella**
 - Sovellus on web sivu joka on ladattu älytelevisioon
 - Sovellus voi olla JavaScript- tai Flash-sovellus
- **Kamera tuettu ainakin 2012 malleissa**
 - Playerilla pystyy toistamaan yhtä mediaa kerrallaan
 - <Video> elementeillä pystyy periaatteessa toistaa useita, ei selvillä kykeneekö selain
 - Kamera api ohjeessa löytyy tieto miten kuvan saa ruudulle

15.2.2013

Sovelluskehitys älytelevisioon

2

Samsung Smart TV

- **Sovelluksien ohjaaminen**
 - Älypuhelimella tuettu
 - HTTP protokollalla (UPnP discovery)
 - Älypuhelimeen pitää asentaa Samsung TV Remote Applikaatio
 - BT kytkeytymisestä ei mitään informaatiota, oletus että ei pysty
 - Smart touch control
 - Ääni ohjaus
 - SDK emulaattori ei tue vielä ominaisuutta
 - Interaktiivinen ohjain
 - <http://www.samsungdforum.com/Guide/tut00015/index.html>
- **Kehitettävää sovellusta voi testata älytelevisiossa**
 - Smart hub pitää olla älytelevisiossa asennettuna

15.2.2013

Sovelluskehitys älytelevisioon

3

LG

- **Pääteteknologia: HTML5+CSS3+JavaScript**
- **Live streemaus tuettu (HLS&Widevine)**
- **Sovellus ladataan televisioon**
 - Sovellus voi olla pelkkä linkki joka sisältää URL osoitteen
 - Hosted (URL-based)
 - Soveltuu sovelluksille jotka sisältää palvelin pohjaista web palvelua
 - Sovellus voi myös sisältää pienen määrän resursseja kuten ikoni, html, JavaScriptiä ja mediaa
 - Packaged (downloadable)
- **Kamera tuki (API) tulee vuoden 2013 alusta**
 - Tieto perustuu keskustelu foorumiin kysymykseen
- **Sovelluksien ohjaaminen älypuhelimella tuettu**
 - UDAP protokollalla (pohjautuu HTTP-protokollaan)
- **Kehitettävää sovellusta voi testata älytelevisiossa**
 - USB yhteys

15.2.2013

Sovelluskehitys älytelevisioon

4

Philips/Sharp Net TV

- Beta versio
- Pääteknologia: CE-HTML (kuuluu HbbTV standardiin)
- Live streemaus tuettu (HLS 2012 mallista alkaen, WMS aikaisemmin)
- Sovellus sijaitsee pilvipalvelimella
 - Sovellusta voi päivittää helposti
- Kamera tuesta ei tietoa
 - Skype saatavilla kuitenkin joihinkin 2012 malleihin, tarvitsee erillisen USB-kameran

15.2.2013

Sovelluskehitys älytelevisioon

5

Philips/Sharp Net TV

- Sovelluksien ohjaaminen
 - Pointing laite
 - USB näppäimistö ja hiiri (tuettu uusissa alustoissa, viittaa Smart TV Alliance speksiin)
 - Android ja iOS älypuhelimella TV remote apps:lla. Laitteet kytketty samaan kotiverkkoon
- Kehitettävää sovellusta voi testata ainoastaan validointi työkalulla

15.2.2013

Sovelluskehitys älytelevisioon

6

Google TV

- **Valmistajat**
 - Toimii tällä hetkellä ainoastaan Sonyn kaveri boksilla
 - Sony, Samsung ja LG kehittää laitteita jotka tulevat tukemaan google TV:tä
- **Pääteteknologiat:**
 - Java (Android sovellus Google TV:lle)
 - HTML5 (Web sovellus Google TV:lle)
- **Live streemaus tuettu (HLS)**
- **Sovellus ladataan televisioon (Android sovellus)**
- **Web sivua käytetään Chrome selaimella (Web sovellus)**
- **Kamera tukea ei ole**

15.2.2013

Sovelluskehitys älytelevisioon

7

Google TV

- **Tuetut HW ominaisuudet**
 - Wifi
 - USB (host)
- **Sovelluksen ohjaaminen**
 - Älypuhelimella tuettu
 - Hiiri, hiiripallo, joystick, peliohjain, näppäimistö, ym..
- **Kehitettävää sovellusta voi testata älytelevisiossa**
 - IP yhteys ADP:lla
 - https://developers.google.com/tv/android/docs/gtv_hardware_devices

15.2.2013

Sovelluskehitys älytelevisioon

8

Yahoo connected TV

- **Valmistajat**
 - Samsung, Sony, VIZIO, Toshiba
- **Pääteteknologia: JavaScript, XML, CSS**
- **Live streemaus tuettu**
 - Simulaattorissa
 - Tuki televisiossa vaihtelee valmistajan mukaan
 - Tuettu protokolla vaihtelee myös valmistajan mukaan
- **Sovelluksen ohjaaminen**
 - Älypuhelimella tuettu
 - Liike
 - Useampi voi ohjata samaan aikaan
- **Rajoitettu kelpuutettuja sovellus kehittäjiä**

15.2.2013

Sovelluskehitys älytelevisioon

9

Smart TV Alliance v2

- **Alustava versio**
- **Valmistajat**
 - LG, Philips (toistaiseksi)
- **Pääteteknologia: HTML5, CSS3, JavaScript**
- **Live streemaus tuettu (HLS)**
- **Sovellusta käytetään selain moottorilla**
- **Kamera tuesta ei mainintaa**
- **Älypuhelin ohjauksesta ei mainintaa**
- **Kehitetty sovellus pitää hyväksyttää tällä hetkellä televisio valmistajan foorumin kautta**

15.2.2013

Sovelluskehitys älytelevisioon

10

Panasonic Viera connect

- **Pääteteknologia:** JavaScript ainoastaan tai HTML5
- **Live streemaus tuettu (HLS)**
- **Sovelluksessa graafinen koti käyttöliittymä televisiossa, muu toiminta kehittäjän palvelimella (HTTP protokolla)**
- **Sovellusta käytetään**
 - JavaScript ainoastaan käytössä => ei-selain moottorilla
 - HTML5 => selaimella
- **Kamera tuesta ei mainintaa**
- **Sovelluksen ohjaaminen**
 - USB näppäimistö
 - Älypuhelin ohjauksesta ei mainintaa

15.2.2013

Sovelluskehitys älytelevisioon

11

Panasonic Viera connect

- **Rekisteröinti ilmainen evaluointia varten**
- **Kehittäjäksi ryhtyminen maksullinen (vuosimaksu)**
- **Kehittämiseen tarvitaan älytelevisio ja oma palvelin (Apache)**
- **Sovellus sijaitsee pilvipalvelimella**
 - Sovellusta voi päivittää helposti

15.2.2013

Sovelluskehitys älytelevisioon

12

Apple TV

- Erikseen myytävä laite
 - Kytetään HDMI-yhteydellä televisioon
 - Langaton tai langallinen yhteys verkkoon
 - Kauko-ohjain ja Remote applikaatio saatavilla iPhone, iPad, iPod touch laitteisiin jolla voi kytkeytyä ja ohjata Apple TV:tä
- Laitteella voidaan:
 - Toistaa videota iTunes, Netflix, Hulu Plus, YouTube ja urheilu kanavilta
 - Kuunnella musiikkia ja katsoa kuvia
 - Kytkeytyä iPad-, iPhone-, iPod touch, Mac- tai PC-koneseen. Toistaa tv ruudulla edellä mainituista laitteista tai päinvastoin
- SDK ei ole saatavilla

15.2.2013

Sovelluskehitys älytelevisioon

13

Espial TV Browser

- TV valmistajat (asiakkaita)
 - Sharp, Toshiba, Hitachi, Panasonic Semi, Mitsubishi ja Iomega
- Pääteknologia: HTML 5, CSS 3, CSS 3D, DOM2&3
- Live streemaus tuettu (HLS)

15.2.2013

Sovelluskehitys älytelevisioon

14

Nyxio

- **Pääteknologia:**
- **PC+HDTV samassa paketissa**
 - VioSphere Smart TV
 - Windows7 Ultimate
- **Android OS ja HDTV samassa paketissa**
 - Vuzion SmartTV with Android OS
 - Voi käyttää android marketin pelejä
- **Sisäänrakennettu kamera**
- **Kosketusnäyttö**
 - Android versiossa vaihtoehtoinen
- **Mikrofoni**
- **Selain**
- **BT**

15.2.2013

Sovelluskehitys älytelevisioon

15