



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y
ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y
TELECOMUNICACIONES**

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN
DEL TÍTULO DE INGENIERO EN ELECTRÓNICA
Y TELECOMUNICACIONES**

**TEMA: IMPLEMENTACIÓN DE VIDEO STREAMING PARA
LA VISUALIZACIÓN EN TIEMPO REAL DE LA
FUNCIONALIDAD DE UNA APLICACIÓN INTERACTIVA
TRANSMITIDA EN BTS (BROADCAST TRANSPORT
STREAM).**

AUTOR: ROGEL CORONEL LISSETH MARILYN.

DIRECTOR: ING. ACOSTA BUENAÑO FREDDY ROBERTO.

SANGOLQUÍ

2016



**DEPARTAMENTO DE ELECTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y
TELECOMUNICACIONES**

CERTIFICACIÓN

Certifico que el trabajo de titulación , **“IMPLEMENTACIÓN DE VIDEO STREAMING PARA LA VISUALIZACIÓN EN TIEMPO REAL DE LA FUNCIONALIDAD DE UNA APLICACIÓN INTERACTIVA TRANSMITIDA EN BTS (BROADCAST TRANSPORT STREAM)”**, realizado por la Srta. LISSETH MARILYN ROGEL CORONEL, ha sido revisado en su totalidad y analizado en el software anti-plagio, el mismo cumple normas estatutarias establecidas por la Universidad de las Fuerzas Armadas ESPE, por tanto me permito acreditarlo y autorizar a la Srta. LISSETH MARILYN ROGEL CORONEL para que lo sustente públicamente.

Sangolquí, 11 de abril de 2016.

A handwritten signature in blue ink is positioned above a horizontal line. The signature is cursive and appears to read 'Freddy Acosta Buenaño'. Below the line, the name 'Ing. Freddy Acosta Buenaño, MSc.' is printed in black text.

Ing. Freddy Acosta Buenaño, MSc.

DIRECTOR



**DEPARTAMENTO DE ELECTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y
TELECOMUNICACIONES**

AUTORÍA DE RESPONSABILIDAD

Yo, LISSETH MARILYN ROGEL CORONEL, con cédula de identidad N° 1003400155 declaro que este trabajo de titulación **“IMPLEMENTACIÓN DE VIDEO STREAMING PARA LA VISUALIZACIÓN EN TIEMPO REAL DE LA FUNCIONALIDAD DE UNA APLICACIÓN INTERACTIVA TRANSMITIDA EN BTS (BROADCAST TRANSPORT STREAM)”**, ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 11 de abril de 2016.

Lisseth Marilyn Rogel Coronel
C.C. 1003400155



**DEPARTAMENTO DE ELECTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y
TELECOMUNICACIONES**

AUTORIZACIÓN

Yo, LISSETH MARILYN ROGEL CORONEL Autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca virtual de la Institución el presente trabajo de titulación “IMPLEMENTACIÓN DE VIDEO STREAMING PARA LA VISUALIZACIÓN EN TIEMPO REAL DE LA FUNCIONALIDAD DE UNA APLICACIÓN INTERACTIVA TRANSMITIDA EN BTS (BROADCAST TRANSPORT STREAM)”, cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Sangolquí, 11 de abril de 2016.

A handwritten signature in blue ink is positioned above a horizontal line. The signature is cursive and appears to read 'Lisseth Rogel Coronel'.

Lisseth Marilyn Rogel Coronel

C.C. 1003400155

DEDICATORIA

“En el mundo somos reconocidos por nuestros logros, pero en el Reino de Dios somos conocidos por nuestro corazón”, Papa Francisco.

A Papito Dios porque cada vez que caí, cada vez que me sentía atrapada, siempre me mostró la luz que me guio a la salida.

A mi abuelo, Eudoro quien me enseñó que no siempre se puede tener lo que deseas, pero debes amar lo que tienes. Estoy segura que hoy en el cielo se regocija por el logro de su nieta favorita.

A mis padres, Cosme y Mónica, quienes me inculcaron valores y con su ejemplo me enseñaron a ser una guerrera, la cual no decae por más fuerte que sea la tormenta.

A mis pequeños, Cosme, Giovanni, Adamaris y Micaela, quienes son mi impulso para persistir sin importar las circunstancias.

Lisseth Marilyn Rogel Coronel.

AGRADECIMIENTO

A Dios por bendecir mi camino, porque yo soy la niña de sus ojos.

*A mis padres quienes me entregaron toda su confianza y me brindaron apoyo
incondicional en este arduo camino.*

A mi familia que se ha mantenido a mi lado en cada alegría y tristeza.

*A mis ingenieros, quienes más allá de ser mis mentores se convirtieron en mi
segunda familia y velaron por mi desarrollo integral.*

A mis amigas y amigos que me han acompañado a lo largo de esta gran aventura.

*Al baloncesto que formó mi carácter y me enseñó que siempre habrá una
revancha.*

El título es mío, el logro es nuestro.

ÍNDICE GENERAL

CERTIFICACIÓN	II
AUTORÍA DE RESPONSABILIDAD.....	III
AUTORIZACIÓN	IV
DEDICATORIA	V
AGRADECIMIENTO	VI
ÍNDICE GENERAL.....	VII
RESUMEN.....	XIII
ABSTRACT.....	XIV
CAPÍTULO 1 INTRODUCCIÓN.....	1
1.1. ANTECEDENTES.....	1
1.2. JUSTIFICACIÓN E IMPORTANCIA DEL PROYECTO	3
1.3. ALCANCE DEL PROYECTO	4
1.4. OBJETIVOS.....	6
1.4.1. General	6
1.4.2. Específicos.....	7
1.5. ESTRUCTURA DE LA MEMORIA	7
CAPÍTULO 2 ESTADO DE LA CUESTION.....	9
2.1. TELEVISION DIGITAL	9

2.1.1. Tipos de televisión digital.....	9
2.1.2. Televisión digital terrestre.....	10
2.2. STREAMING.....	12
2.2.1. Introducción.....	12
2.2.2. Definición de streaming.....	13
2.2.3. Servicios de streaming.....	14
2.2.4. Elementos que forman el servicio streaming.....	17
2.2.5. Arquitectura de servicio streaming.....	21
2.2.6. Protocolos de servicio streaming.....	22
2.2.7. Formatos para servicio streaming.....	25
2.3. HTML5	29
2.3.1. Etiquetas HTML5.....	30
2.4. PAGINA WEB	34
2.4.1. Introducción.....	34
2.4.2. Tipos de páginas web	34
2.4.3. Sistema de Gestión de Contenidos (CMS).....	36
2.5. SERVIDOR PARA DESARROLLO Y FUNCIONAMIENTO DE PAGINAS WEB.	37
2.6. ESTUDIO DEL ESTADO DEL ARTE.....	38
CAPÍTULO 3 DISEÑO	42
3.1. PLATAFORMA DE USABILIDAD DE TDT.....	42
3.2. TOPOLOGÍA.....	44
3.3. SERVIDOR DE USABILIDAD.....	45
3.4. SERVIDOR DE CAPTURA, CODIFICACIÓN Y STREAMING	47
3.4.1. Tarjeta capturadora de video.....	48
3.5. SET TOP BOX	50

CAPÍTULO 4 IMPLEMENTACIÓN, PRUEBAS Y ANÁLISIS DE RESULTADOS .	54
4.1. IMPLEMENTACIÓN	54
4.2. ENCUESTA MOS	56
4.3. PRUEBAS REMOTAS.....	62
4.4. ANÁLISIS DE RESULTADOS.	63
CAPÍTULO 5 CONCLUSIONES Y LINEAS DE TRABAJOS FUTUROS	65
5.1. CONCLUSIONES	65
5.2. LÍNEAS DE TRABAJOS FUTUROS.....	67
CAPÍTULO 6 BIBLIOGRAFÍA.....	69

ÍNDICE DE TABLAS

Tabla 1 Tipos de Televisión Digital	10
Tabla 2. Requerimientos de ancho de banda en diferente.....	
 aplicaciones.	19
Tabla 3. Contenedores y Codecs de A/V.	29
Tabla 4 Etiquetas HTML5.....	30
Tabla 5 Comparación de tarjetas capturadoras (Avermedia.com,	
 2015).....	49
Tabla 6. Resultados pruebas remotas.	63

ÍNDICE DE FIGURAS

Figura 1 Plataforma de Usabilidad de TDT.	5
Figura 2. Escenarios para la TDT	11
Figura 3. Componentes de un sistema de TDT.	12
Figura 4. Streaming vs otros tipos de acceso a medios continuos	13
Figura 5. TS frame. Fuente:(how.com, 2016)(how.com, 2016)	14
Figura 6. Streaming en directo	15
Figura 7. Streaming bajo demanda	16
Figura 8. Streaming en directo con contenidos almacenados	16
Figura 9. Arquitectura básica de servicio streaming.....	20
Figura 10. Elementos de arquitectura streaming	21
Figura 11. Formato contenedor.	26
Figura 12. Estructura lógica de un flujo de bits Ogg.	28
Figura 13. Página web estática.....	35
Figura 14. Página web dinámica.....	35
Figura 15. Página web	42
Figura 16. Prototipo control remoto.....	43
Figura 17. Video streaming	44
Figura 18. Topología de servicio streaming	44
Figura 19. Sevidor de captura, codificación y streaming.....	48
Figura 20. Recepción de señales de TDT.	51
Figura 21. Vista trasera STB Fuente: (Eitv.com.br, 2016).....	52
Figura 22. Conexión de los servidores con el STB.....	53
Figura 23. Escenario para la visualización del video streaming.....	54
Figura 24. Servicio de video streaming implementado.	55

Figura 25. Streaming de una aplicación interactiva.....	56
Figura 26. ¿Cómo considera la calidad del video presentado en la..... página web?	59
Figura 27. Cómo aprecia la sincronización entre el audio y el video en... la página web?	60
Figura 28. Comparación de navegador, dispositivo y nodo.	61
Figura 29. Comparativa PC de escritorio con tv digital.....	62
Figura 30. Comparativa dispositivo portátil con tv digital.	62

RESUMEN

Con la aparición de la Tv digital a mediados de los 90, iniciaron nuevas formas de ver televisión, entre ellas la interactividad del telespectador con el contenido que se está transmitiendo. Con estas nuevas tendencias también vinieron nuevos requerimientos, uno de ellos la transmisión de contenidos en tiempo real. El presente proyecto de investigación abarca estas dos líneas con la **IMPLEMENTACIÓN DE VIDEO STREAMING PARA LA VISUALIZACIÓN EN TIEMPO REAL DE LA FUNCIONALIDAD DE UNA APLICACIÓN INTERACTIVA TRANSMITIDA EN BTS (BROADCAST TRANSPORT STREAM)**. Se presenta un estudio del estado del arte en el cual se muestra las recomendaciones para este tipo de aplicaciones, por otro lado se estudia alternativas para determinar las herramientas que se van a utilizar para la implantación con el fin de dar flexibilidad y escalabilidad a lineamientos futuros. Mediante el uso de una página web se muestra el video streaming al usuario, para su desarrollo se utilizó dos alternativas: una basada en programación en HTML5 y la otra conservando el uso de plugins, en este caso se utilizó el plugin de VLC. Finalmente se aplicó una encuesta tipo MOS con el objetivo de obtener una retroalimentación de la calidad de la experiencia percibida por el usuario en el presente proyecto.

Palabras clave:

- TDT, STB,
- BTS,
- VIDEO STREAMING,
- MOS,
- HTML5,
- HTTP- DASH.

ABSTRACT

The new ways of watching television began with the rise of digital TV in the middle 90s, these include viewer interactivity with the content that it being transmitted. With these new tendencies also came new requirements, one of them is the transmission of real-time content. This research project include these two lines for building STREAMING VIDEO FOR DISPLAY IN REAL TIME OF FUNCTIONALITY OF AN INTERACTIVE APPLICATION TRANSMITTED IN BTS (BROADCAST TRANSPORT STREAM). The project shows a study of the state of the art in which recommendations for such applications, by the other hand this study give other methods in order to provide flexibility and scalability to future guidelines. This project uses a website to shown streaming video to the user, for its developing were used two alternatives: one based on programming in HTML5 and other preserving the use of plugins, for this case was used plugin VLC. Finally was applied a survey MOS type in order to obtain a feedback of the quality of experience perceived by the user in this project

KEYWORDS:

- **TDT, STB,**
- **BTS,**
- **VIDEO STREAMING,**
- **MOS,**
- **HTML5,**
- **HTTP- DASH.**

CAPÍTULO 1

INTRODUCCIÓN

1.1. ANTECEDENTES

A partir de su aparición en la década de los 70 la televisión es sin lugar a duda uno de los medios de comunicación, entretenimiento e información comúnmente más utilizados tanto en nuestro país como a nivel mundial. La mayor cantidad de avances para la televisión se llevó a cabo entre los años de 1920 y 1950, comprendiendo desde la integración del sistema de televisión hasta la televisión a colores, a partir de esto la televisión analógica no dio su siguiente gran evolución sino hasta mediados de los 90's en donde apareció la Televisión Digital Terrestre (TDT).

Para poder transmitir TDT a nivel mundial se cuenta con varios estándares como: DVTB-T¹, ATSC², ISDB-T³, DTMB⁴, estos estándares dependiendo de sus prestaciones han sido adoptados por los distintos países que ya están o tienen proyección a emitir señales de TDT.

En Ecuador se realizó el análisis de los estándares de televisión digital terrestre (TDT) y pruebas de campo utilizando equipos de comprobación

1 Digital Terrestrial Television Broadcasting

2 Advanced Television Systems Committee

3 Digital terrestrial broadcasting system

4 Digital Terrestrial Multimedia Broadcast

técnica de la Superintendencia de Telecomunicaciones. Este análisis indica que se debe adoptar el estándar de Radiodifusión Digital de Servicios Integrados Terrestres versión brasileña (ISDB-Tb⁵), el cual muestra varias ventajas sobre sus competidores en la implementación de todos los modelos de servicio de televisión digital (Sotomayor Jácome, 2009). Lo mencionado es oficial a partir del 26 de marzo de 2010 cuando el entonces Superintendente de Telecomunicaciones Fabián Jaramillo, anunció que la recomendación proporcionada por la SUPERTEL había sido aceptada por el Consejo Nacional de Telecomunicaciones (CONATEL). En la actualidad, estos dos organismos (SUPERTEL Y CONATEL), se encuentran fusionados en la Agencia de Regulación y Control de las Telecomunicaciones – ARCOTEL.

Habiendo adoptado un estándar oficial, nuestro país comenzó el cambio de tecnología de manera progresiva desde el año 2010, por lo que actualmente algunos canales ya transmiten su programación de forma digital como emisiones de prueba y en forma análoga, y se ha planificado que el apagón analógico se realice en 3 fases: para el 31 de diciembre de 2016 las ciudades de Quito, Guayaquil y otras con más de 500.000 habitantes, para el 31 de diciembre de 2017 las ciudades con población entre 200.000 y 500.000 habitantes y finalmente para el 31 de diciembre de 2018 las ciudades con población menor a 200.000 habitantes (Información, 2015).

Adicional a la emisión de audio y video, el estándar ISDB-tb simultáneamente transmite datos, con ello crea un canal de retorno lo cual posibilita la interacción entre el usuario y el contenido, por tal motivo actualmente estudiantes, docentes y personas interesadas en la TDT han desarrollado aplicaciones interactivas con el uso de simuladores disponibles en la Web. Sin embargo, a través de pruebas realizadas por el grupo de investigación de TDT (Wicom/ESPETV) de la Universidad de las Fuerzas

⁵ *Integrated Services of Digital Broadcasting – Terrestrial.*

Armadas ESPE se ha constatado que la visualización obtenida al ejecutar una aplicación interactiva sobre estos simuladores no es la misma que se obtiene sobre un sistema real de TDT. Cuando la aplicación es generada como un archivo Broadcast Transport Stream (*.bts), y se utiliza un decodificador, set top box (STB), y televisiones digitales que incluyen el middleware Ginga⁶, presenta ciertos problemas que no son percibidos en la simulación (ubicación errónea de imágenes, falta de sincronismo entre el audio y el video, funcionalidades equivocadas, entre otros).

1.2. JUSTIFICACIÓN E IMPORTANCIA DEL PROYECTO

La Televisión Digital Terrestre es una oportunidad que posibilita el desarrollo de múltiples programas y aplicaciones como Telegobierno, Telesalud y Teleducación, para el buen vivir de la población (Información, 2015), por lo cual se ha impulsado la investigación de esta nueva tecnología. De acuerdo a los requerimientos surgidos en las III Jornadas Iberoamericanas de difusión y capacitación sobre Aplicaciones y Usabilidad de la Televisión Digital Interactiva (TVDi) de la Red temática en Aplicaciones y Usabilidad de la Televisión digital Interactiva (RedAUTI) en Paraíba, Brasil, actualmente diseñadores gráficos, productores, investigadores, alumnos que se desenvuelven en este medio no pueden llevar a cabo pruebas en tiempo real de sus aplicaciones interactivas para TDT. Estos inconvenientes surgen como consecuencia de la escasez y falta de accesibilidad a los laboratorios disponibles para esta tecnología, por tal motivo los desarrolladores cotidianamente experimentan problemas en la ejecución de sus aplicaciones cuando son transmitidas sobre un sistema real de TDT. Los problemas comúnmente presentados son: la falta de sincronismo entre el audio y el

⁶ Nombre del Middleware Abierto del Sistema Nipo-Brasileño de TV Digital (ISDB-TB).

video, ubicación errónea de las imágenes, mal funcionamiento de las opciones programadas y consecuentemente la visualización es distinta a la simulada. Debido a estos inconvenientes no se permite que la funcionalidad de la aplicación para TV-digital desarrollada se lleve a cabo.

Para solucionar estos requerimientos la Universidad de las Fuerzas Armadas ESPE a través de su grupo de investigación Wicom/ESPETV se encuentra ejecutando un proyecto de Investigación para la Implementación de una plataforma de usabilidad para TDT. La plataforma requiere la implementación de un sistema de video streaming para permitir que los usuarios puedan visualizar en tiempo real la funcionalidad de sus aplicaciones interactivas, en base a la manipulación remota de transmisores y receptores del laboratorio, facilitando a los desarrolladores tener una visión clara de lo que están implementando y no tener inconvenientes cuando su aplicación interactiva este siendo transmitida por un servidor broadcast de TDT.

Es importante precisar que con la implementación del video streaming en el laboratorio Wicom/ESPETV, se puede realizar pruebas de TDT y aplicaciones interactivas para la misma, así como también pruebas de funcionalidad de Televisión por Protocolo de Internet (IPTV).

1.3. ALCANCE DEL PROYECTO

Se desarrolló un análisis del estado del arte de los temas relacionados a formatos de audio y video, tarjetas de video, librerías para hacer streaming, video streaming y servidor streaming.

Una vez desarrollado el estado del arte se implementó el escenario que se muestra mediante la línea resaltada en la Figura 1. Se cuenta con un

sistema de producción el cual está formado por el set top box (STB) y el servidor de captura y codificación. El sistema de producción se encarga de capturar el contenido en vivo y lo transmite, en tiempo real y con el formato adecuado, al servidor. La señal de A/V del STB es obtenida a través del puerto HDMI e ingresa a un servidor denominado de captura, codificación y streaming el cuál ha sido implementado sobre un ordenador ubicado en el laboratorio WICOM /ESPETV. Para el servidor de captura y codificación se adquirió la tarjeta capturadora de A/V, AVerMedia C727, la cual permite obtener directamente el video en formato *multimedia framework formats* (ffmpeg)⁷.

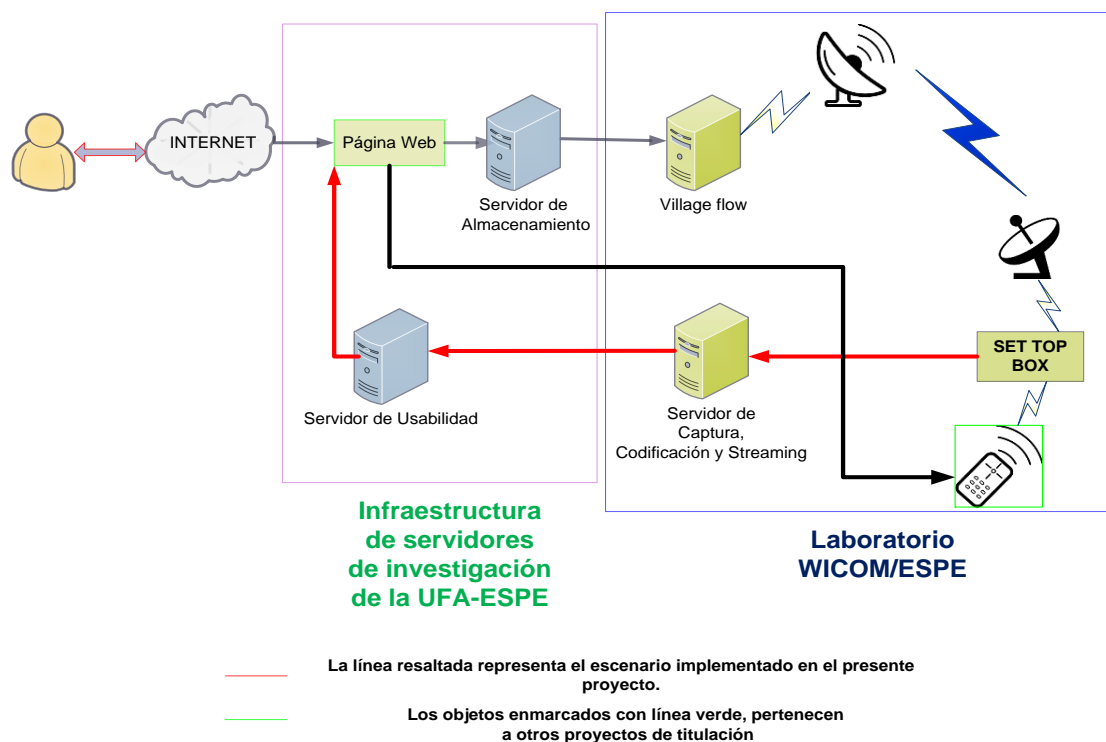


Figura 1 Plataforma de Usabilidad de TDT.

Posterior con la ayuda del software VLC se comprimen los contenidos, utilizando para el audio el códec *theora* y para el video el códec *vorbis*, y se

⁷ Software libre capaz de decodificar, codificar, grabar, transcodificar, reproducir y hacer streaming de A/V.

les da un formato adecuado, contenedor *ogg*, de manera que resultan aptos para su transmisión por medio de streaming sobre un protocolo de transporte.

Adicional utilizando la herramienta HTML5 y el gestor de contenidos Drupal, se implementó la interfaz gráfica de usuario (GUI), sobre la cual se muestra el contenido del video streaming a los usuarios. Habiendo obtenido el video en el formato requerido e implementado la GUI, con la ayuda de herramientas de software libre se creó un servidor virtual de streaming, el cual permite generar un enlace entre el usuario y el laboratorio. Toda la implantación del prototipo para este servidor ha sido realizada utilizando herramientas de virtualización de servidores bajo la infraestructura de investigación de la Universidad de las Fuerzas Armadas ESPE.

Para determinar la calidad de la experiencia de usuario, se han ejecutado pruebas de funcionamiento en tiempo real, utilizando la transmisión de TDT con interactividad, las cuales a través de una encuesta a diferentes usuarios tipo, muestran una evaluación *Mean Opinion Score* (MOS), que han permitido cuantificar de forma subjetiva el trabajo realizado.

1.4. OBJETIVOS

1.4.1. General

Implementar video streaming para la visualización en tiempo real de la funcionalidad de una aplicación interactiva transmitida en BTS (Broadcast Transport Stream).

1.4.2. Específicos

- Desarrollar un estudio del estado del arte.
- Determinar la tarjeta de video a utilizar para la obtención de la señal de video a través del puerto HDMI.
- Determinar el software a utilizar para la implementación.
- Crear un servidor virtual de streaming para generar un enlace entre el usuario y el laboratorio.
- Implementar el video streaming para la plataforma de usabilidad de televisión digital del grupo Wicom/ESPETV.
- Realizar una evaluación MOS para determinar la calidad de la experiencia ofrecida por el sistema.

1.5. ESTRUCTURA DE LA MEMORIA

El presente documento se divide en los siguientes apartados:

Capítulo 1: *Introducción.* Engloba los temas referentes a los antecedentes, justificación e importancia que conlleva plantear el presente trabajo de investigación. En la sección Alcance de Proyecto se detalla los resultados que se han obtenido y finalmente se enumera los objetivos planteados.

Capítulo 2: *Estado de la cuestión.* En este capítulo se describe brevemente aquella información teórica que se ha revisado para poder

desarrollar la investigación. Resume conceptos relacionados a Televisión Digital Terrestre Interactiva, definición, servicios, arquitecturas, servidores, códecs, formatos y protocolos para el servicio streaming. Por otro lado se describe la teoría referente a las herramientas adicionales utilizadas, para implementar el servicio de streaming y la interfaz gráfica de usuario, tales como: HTML5, Drupal, XAMPP y Reproductor VLC. Finalmente se realiza un estudio del estado del arte recopilando información de trabajos relacionados a la investigación desarrollada.

Capítulo 3. *Diseño.* En este apartado se muestra la topología implantada, y el estudio realizado para la selección de hardware y software utilizado.

Capítulo 4. *Implementación, pruebas y análisis de resultados.* Reúne el trabajo realizado en la presente investigación. Por un lado muestra los pasos a seguir en la implementación, mientras que en otra de sus partes incluye una evaluación subjetiva MOS, de la calidad de la experiencia de usuario. Adicional se realizaron pruebas remotas del prototipo implantado. Al final de esta sección se realiza un análisis de los resultados obtenidos.

Capítulo 5. *Conclusiones y líneas de trabajos futuros.* En esta sección se resume los resultados obtenidos y se plantea lineamientos para futuras investigaciones relacionadas al tema desarrollado.

Capítulo 6. *Bibliografía.* Contiene las fuentes de consulta que han contribuido con el desarrollo del proyecto.

Finalmente, se cuenta con los apéndices A, B, C y D, en los cuales se muestran: el manual de instalación, configuración usuario y la encuesta MOS, respectivamente.

CAPÍTULO 2

ESTADO DE LA CUESTION.

2.1. TELEVISION DIGITAL

La televisión digital o digital TV, es un grupo de tecnologías que permite la emisión y recepción de señales de televisión de forma digital, logrando no solo mejorar la calidad en audio y video, sino también ahorrar el espectro electromagnético. Por otro lado, la tv digital a diferencia de su antecesora permite el desarrollo de aplicaciones interactivas, la cuales integran al televidente con la programación presentada.

Las aplicaciones de tv digital día a día van teniendo mayor acogida, y no solo están enfocadas al mundo del ocio y el entretenimiento, sino que han llegado a formar parte importante en programación educativa, cultural y preventiva, por ejemplo con el desarrollo de sistemas de alerta temprana.

2.1.1. Tipos de televisión digital.

Con la aparición de la tv digital se dio lugar a distintas formas para acceder a ella. Actualmente la gran mayoría son servicios pagados, sin embargo con la nueva tendencia del conocido “apagón analógico” se ha dado lugar a un servicio de televisión digital gratuito. A continuación en la Tabla 1 se muestra los principales tipos de televisión digital, su forma de

transmisión, requerimientos mínimos para poder obtenerla y un detalle de los dispositivos adicionales que demandan.

Tabla 1
Tipos de Televisión Digital

Tipo de televisión digital	Forma de transmitirla	de Como se puede recibirla	Equipos adicionales
Por cable	Redes híbridas de fibra óptica y cable coaxial	Acometida desde la red de distribución hasta el domicilio.	Sintonizador.
Por satélite	Satélites de comunicaciones.	Antena parabólica correctamente orientada al satélite de comunicaciones.	Sintonizador de canales o un decodificador.
IP o ADSL	Protocolos asimétricos (xDSL).	Línea telefónica dotada de medios ADSL	Modem y sintonizador de canales.
Terrestre	Ondas hercianas terrestres.	Antenas VHF, UHF convencionales.	Set Top Box (STB), receptores o sintonizadores con el receptor adecuado

2.1.2. Televisión digital terrestre.

La televisión digital terrestre (TDT) o también conocida como la televisión para todos, es la cual llega a cada uno de los hogares propagándose por medio de ondas hercianas terrestres y sin la necesidad de pagos adicionales. Como se puede observar a través de la Figura 2, para poder disfrutar de las prestaciones de la TDT se accede a través de los escenarios:

- Antena convencional + STB + TV analógico convencional
- Antena convencional + TV digital.

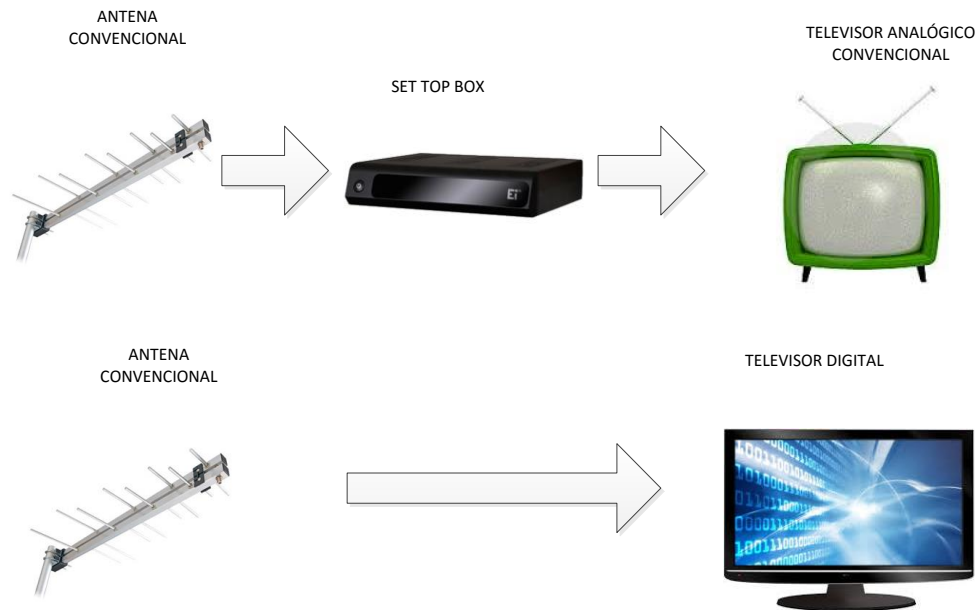


Figura 2. Escenarios para la TDT

2.1.2.1. Televisión digital interactiva.

“Una televisión en la que el telespectador puede hacer algo más que simplemente verla y/o oírla, pasando a ser una especie de teleparticipante, ya que tiene la posibilidad de realizar acciones que pueden cambiar el contenido mostrado por su televisor” (Javier Melechón, URL citado en Martínez, (Acosta, 2012)).

A través de la interactividad en la TDT se puede ofrecer diferentes tipos de servicios, entre ellos aquellos que ofrecen una información independiente de la programación audiovisual que se está emitiendo en ese momento (Servicios de información), también existen los que se complementan con información suplementaria a la programación audiovisual emitida (Servicios ligados a la programación) y por ultimo aquellos que ofrecen la posibilidad de enviar y recibir información de forma personalizada y exclusiva (Servicios transaccionales)(Acosta, 2012).

2.1.2.2. Componentes de un sistema de TDT.

La TDT está formada por distintos componentes como antenas UHF, STB, Televisor convencional o digital, en la Figura 3 se detalla cada uno de ellos.

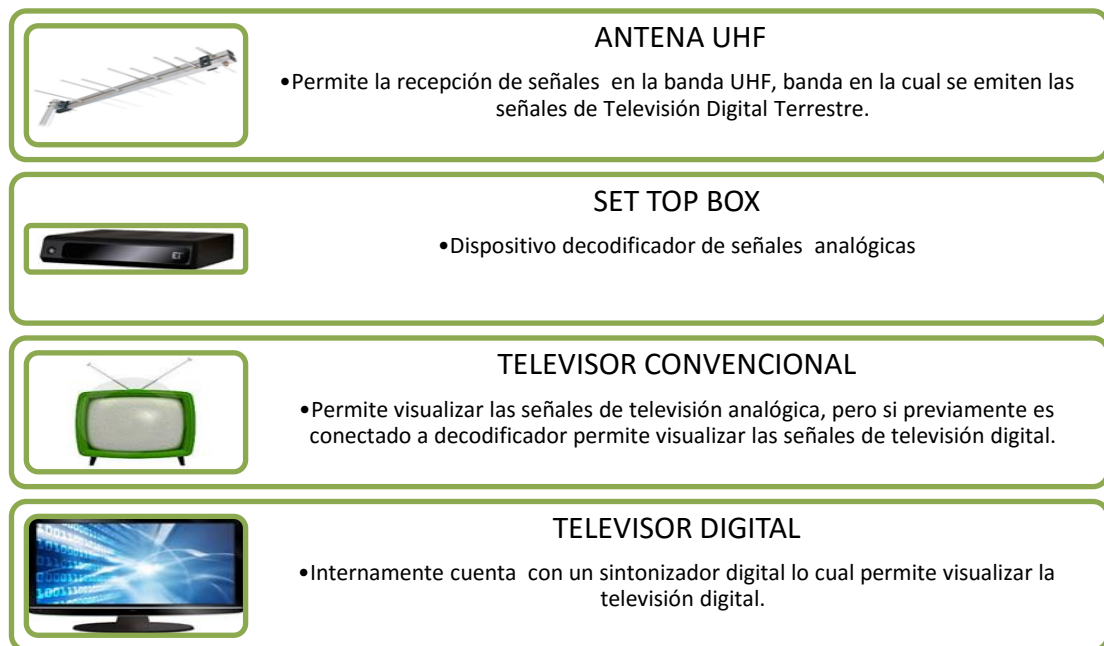


Figura 3. Componentes de un sistema de TDT.

2.2. STREAMING

2.2.1. Introducción.

En los últimos años, el cambio en la forma de acceder a los medios audiovisuales, ha permitido cada vez dar mayor apertura para que el usuario elija el momento y la forma en la cual desea visualizar los contenidos. Precedentemente el usuario accedía mediante la descarga tradicional, descarga toda la información y después la reproduce, y a la

descarga progresiva (pseudo-streaming), reproduce la información según va descargando (YouTube, Google videos, ...) (Alonso, 2009). Para usar estos métodos, el cliente debe disponer de suficiente espacio en el disco para almacenar el contenido, y esperar que termine la descarga para estimar si este cumple o no con sus expectativas, lo cual representa un desperdicio de tiempo y ancho de banda. Como solución a estos inconvenientes nació la tecnología *streaming*.

2.2.2. Definición de streaming.

Como se puede observar en la Figura 4a, streaming es la tecnología en la cual el cliente puede visualizar el contenido de forma paralela mientras realiza su descarga. A diferencia de la transferencia clásica (Ver Figura 4b), este tipo de tecnología funciona mediante el uso de un búfer de datos, que va almacenando una cantidad suficiente del contenido de forma que permite la fluidez de la reproducción sin interrupciones. Sin embargo de existir problemas con la conexión, el cliente podría seguir mostrando el contenido almacenado en el buffer, una vez consumido todo el buffer deberá esperar hasta que se vuelva a llenar para continuar con la reproducción.

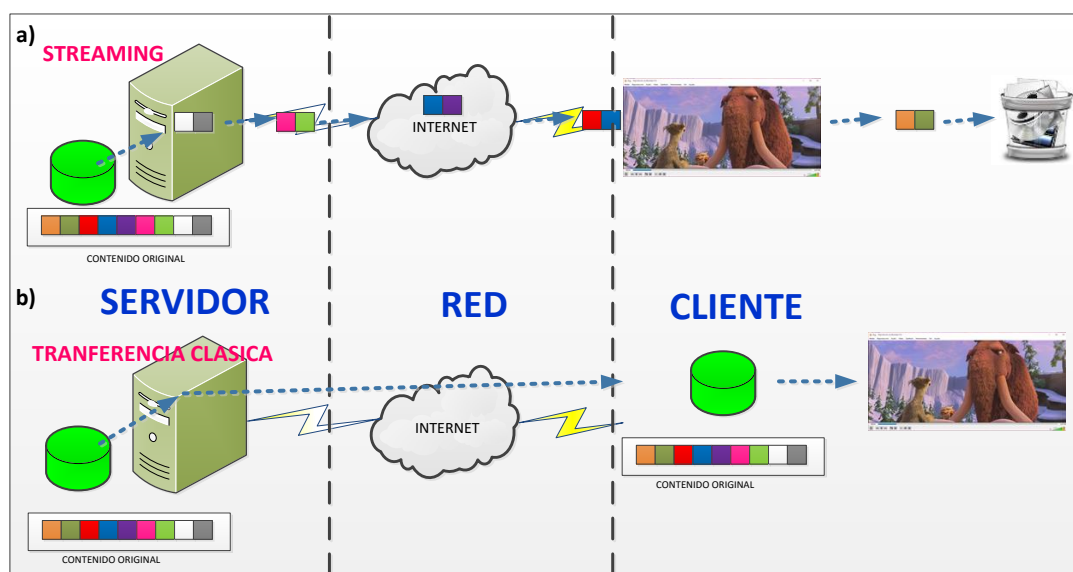


Figura 4. Streaming vs otros tipos de acceso a medios continuos

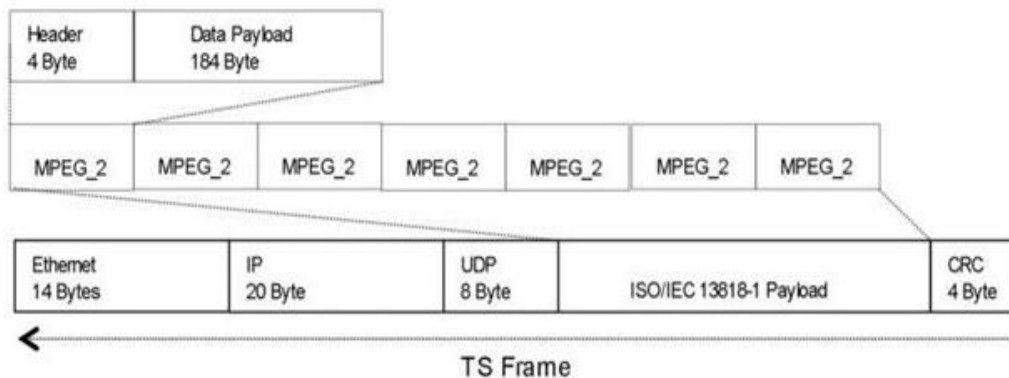


Figura 5. TS frame. Fuente:(how.com, 2016)(how.com, 2016)

Para ofrecer un servicio de streaming, el flujo de datos a ser transmitido se divide en paquetes, y cada uno de estos paquetes para poder ser transportados debe formar una trama TS (Transport Stream), (Ver Figura 5)

La tecnología de streaming muestra ciertas ventajas frente a otros tipos de acceso a medios continuos:

- No es necesario el almacenamiento del contenido, pudiendo descartar la información después de visualizarla.
- Ahorro de ancho de banda, ya que solo se va descargando el contenido suficiente para asegurar la reproducción continua.
- Reproducción en tiempo real.

2.2.3. Servicios de streaming.

2.2.3.1. Servicio de streaming en directo:

El servicio de streaming en directo o también conocido como *Live Streaming*, basa su funcionamiento en la multidifusión. Los usuarios pueden ver la información que se está emitiendo en el punto donde se encuentre

cuando acceden al servicio, siendo esta información un único flujo de audio y video (A/V) para todos, tal como se muestra en la Figura 5.

En el streaming en directo no es posible realizar interacción, pudiendo únicamente pausar la transmisión, sin embargo esta no se detiene. Cuando el cliente retoma su reproducción, visualizará primero el contenido almacenado dentro del buffer, una vez consumido este contenido el cliente obtendrá la información que se está transmitiendo en ese instante dado (Pozueco Álvarez, 2014).

2.2.3.2. Servicio de streaming bajo demanda.

El servicio de streaming bajo demanda o también conocido como *On-demand*, se caracteriza porque realiza un envío punto a punto (*unicast*), en donde, para cada cliente el servidor dispone de un canal de información en exclusiva, el cual asegura la interacción cliente-servidor (pausar, avanzar, retroceder, ...) con un flujo único de datos de A/V para cada usuario (Pozueco Álvarez, 2014).

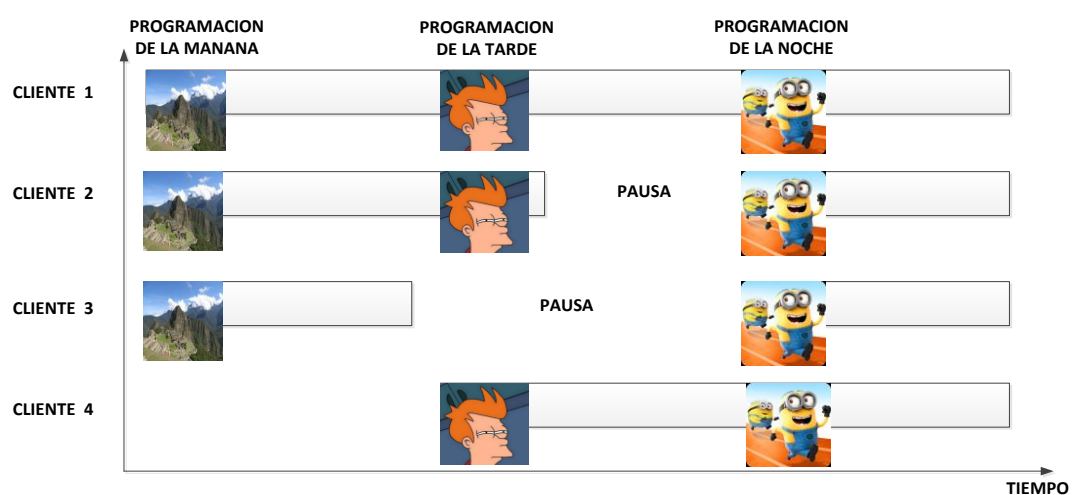


Figura 6. Streaming en directo

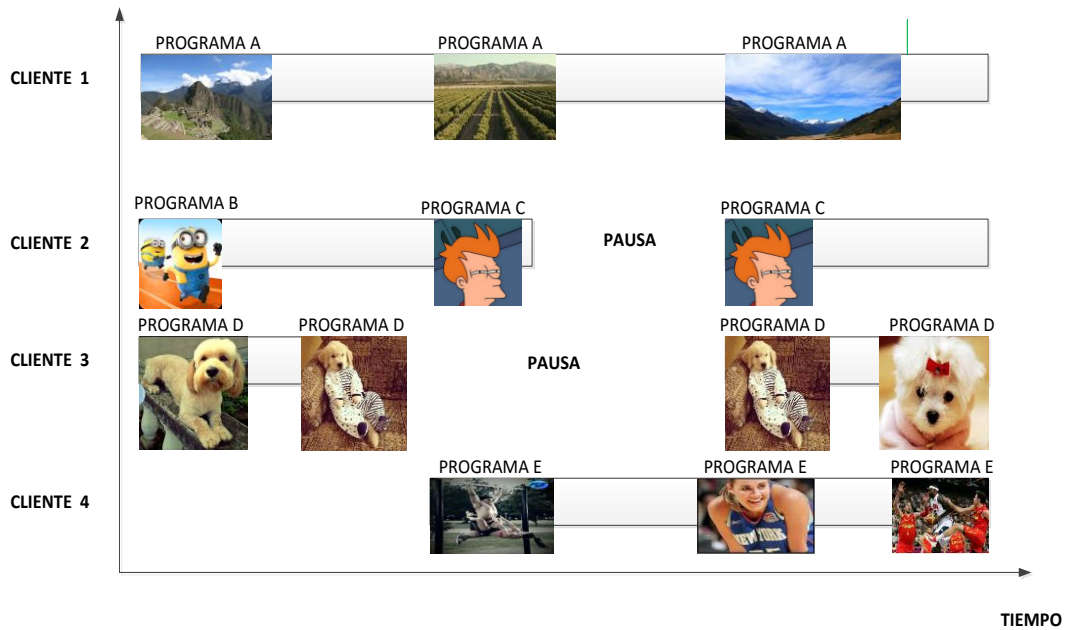


Figura 7. Streaming bajo demanda

Dentro de su estructura el servicio streaming bajo demanda cuenta con un sistema de almacenamiento, en el cual se encuentran previamente archivados los contenidos que van a ser transmitidos, de tal manera que puedan estar a disposición de los clientes en cualquier momento.

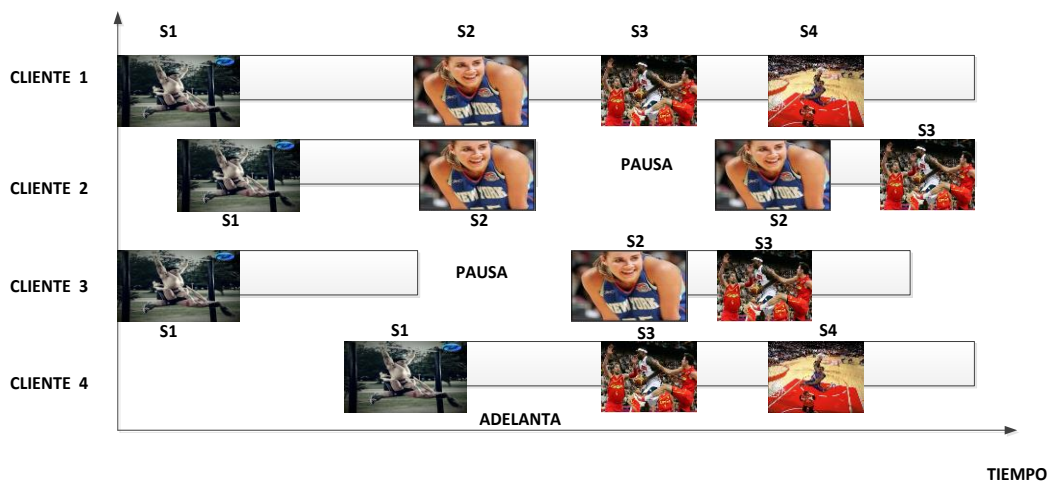


Figura 8. Streaming en directo con contenidos almacenados

Todo lo anteriormente expuesto se muestra en la Figura 7, como ya se mencionó para cada cliente existe un flujo de información distinta la cual puede ser accedida en el momento que el cliente lo solicite y a diferencia del streaming en directo, cuando el cliente pausa la programación, esta no continua, se detiene hasta que el cliente decida reanudar la transmisión (cliente 2 y cliente 3).

2.2.3.3. Servicio de streaming en directo con contenidos almacenados.

Este tipo de servicio streaming es una combinación entre servicio bajo demanda y el servicio en directo ya que, simultáneamente mientras se captura y se emite el flujo de A/V, el contenido también puede ser almacenado. De este modo al mismo tiempo que se puede transmitir el contenido en tiempo real, además se puede asegurar cierto grado de interacción cliente-servidor. Este tipo de servicio streaming es usado con el fin de tratar de aprovechar las características de una transmisión multicast, ya que si el cliente realiza una interacción, cuando se une nuevamente a la programación, se le agrega al flujo que este emitiendo la posición más cercana a la que requiere (Ibnoukhatib, 2012). En la Figura 8 se muestra un ejemplo del funcionamiento de este tipo de servicio streaming.

2.2.4. Elementos que forman el servicio streaming.

La tecnología streaming dentro de su estructura puede estar formada por los siguientes elementos:

- **Sistema de almacenamiento:** Es el componente encargado de almacenar los contenidos para un servicio bajo demanda o de servicio directo con contenidos almacenados, del tal forma que estarán a disposición del cliente en cualquier instante en que los solicite o pueden ser transmitidos por el servidor simultáneamente mientras los almacena

(Ibnoukhatib, 2012). Este elemento no es necesario para ofrecer un servicio streaming en directo.

- **Servidor:** A diferencia de los elementos anteriores este no puede estar ausente en ningún tipo de servicio streaming. Las funciones que desempeña son:
 - Recibir y procesar peticiones de los clientes.
 - Establecer la sesión mediante el intercambio de una serie de parámetros.
 - Procesar el envío de los contenidos multimedia que dependiendo del tipo de servicio puede provenir de un sistema de almacenamiento o de producción.
 - Permitir interactividad entre el cliente y el servidor, mediante las acciones de pausar, adelantar, regresar, etc. El nivel de interactividad consentida dependerá del tipo de servicio que se esté prestando.

- **Proxy:** El elemento proxy es el encargado de mantener la comunicación con el cliente, y en muchas ocasiones se lo utiliza con el fin de mejorar la calidad de servicio. El Proxy dependiendo del tipo de servicio de streaming ofrecido puede realizar las siguientes funciones (Carlos Carvajal Álvarez, 2015):
 - **Splitter:** Utilizado en los servicios en directo, este tipo de Proxy es el encargado de disminuir el flujo entre el cliente y el servidor, mediante la simulación de un proceso multicast, de tal forma que se consigue disminuir el flujo en la red del servidor.
 - **Pass-Through:** Está presente en la oferta de servicios en directo y bajo demanda. Gracias al uso de este tipo de proxy la carga se comparte y se da la posibilidad de usar servidores redundantes, de tal forma que las solicitudes de los clientes son gestionadas a través del

proxy, pero el contenido multimedia es enviado directamente del servidor a cada uno de los clientes.

- **Cache:** En el tipo de servicio *On-Demand*, es normal encontrarnos con un cache, este tipo de proxy es similar al proxy del servidor web. Cuando la información ha sido solicitada por primera vez, el cache se encarga de almacenarla, de tal forma que cuando el cliente u otro cliente la requiera nuevamente no será necesario ir a buscar otra vez la información en el cache, gracias a esta función es posible disminuir el tráfico entre cliente y el servidor, además de disminuir el trabajo al servidor.
- **Red:** El ancho de banda provisto por una red, depende del tipo de conexión a internet (cable, fibra, satelital,...), el cual permite la transmisión de datos a mayores o menores velocidades. La calidad del streaming depende directamente de la capacidad de la red, debido a que si existe un gran número de usuarios accediendo al contenido, la red puede sobrepasar su capacidad de ancho de banda, por lo cual estos usuarios empezaran a sufrir interrupciones hasta llegar a un punto de calidad pésima. A continuación en la Tabla 2 se muestra el ancho de banda requerido para asegurar una QoS (calidad de servicio) de acuerdo a la aplicación.

Tabla 2.

Requerimientos de ancho de banda en diferentes aplicaciones.

Aplicación	BW medio (Mbps) - BW de pico (Mbps)
Streaming HDTV (compresión MPEG-2)	12 - 18
Streaming SDTV (compresión MPEG-2)	3 - 8
Voz digital de alta calidad	< 1
Acceso a Internet de alta velocidad	2 - 4
Streaming de IPTV VoD	4 - 6

Fuente: (Lloret, Garcia, & Boronat, 2008)

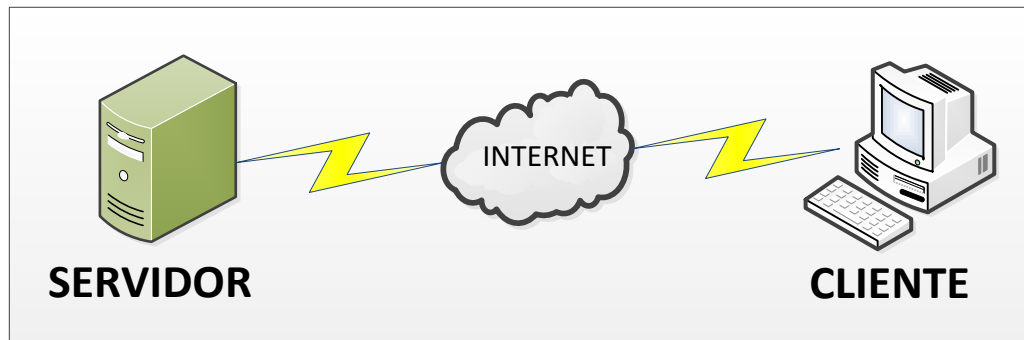


Figura 9. Arquitectura básica de servicio streaming.

- **Protocolos:** Dentro de la tecnología streaming podemos encontrar una serie de protocolos, entre ellos los que no están destinados al control sobre la transmisión (HTTP) y los orientados al control. Para el control de la capa de aplicación tenemos protocolos como RTSP, MMS, RTMP y RTMFP, y para el control del transporte de datos, protocolos como RTP, UDP y TCP. Todos estos protocolos los conoceremos más a fondo en secciones futuras.
- **Cliente:** Es el último elemento en la estructura del servicio streaming, sin embargo dependiendo del tipo de arquitectura que se esté utilizando puede o no estar presente. De estar presente el cliente está conformado por los siguientes elementos (Carlos Carvajal Álvarez, 2015):
 - **Recepción:** Recibe la información solicitada del cliente al servidor.
 - **Buffer:** Es el encargado de almacenar temporalmente en el cliente la información (recepción) que va llegando con el fin de evitar interrupciones en la reproducción y variaciones en el jitter que afecten en la calidad del streaming. Cuando en el buffer existe la cantidad suficiente de frames, inicia la reproducción.
 - **Presentación:** Es la interfaz encargada de la interacción con el usuario permitiendo reproducir la información recibida de forma

temporizada. La presentación puede ser llevada a cabo a cabo a través de reproductores, páginas web, o cualquier otro tipo de GUI (interfaz gráfica de usuario).

2.2.5. Arquitectura de servicio streaming.

La arquitectura básica para brindar servicios de streaming está compuesta por cliente/servidor (Ver Figura 9), sin embargo como se mencionó en la sección 2.2.4, existen varios elementos los cuales pueden formar parte de ella. En la Figura 10, se muestra un esquema con los diferentes componentes que pueden estar presentes dentro de una arquitectura de servicio streaming (Pozueco Álvarez, 2014).

Arquitectura típica: Es la arquitectura más básica implementada para brindar servicios de streaming. Está conformada por los elementos Cliente-Servidor.

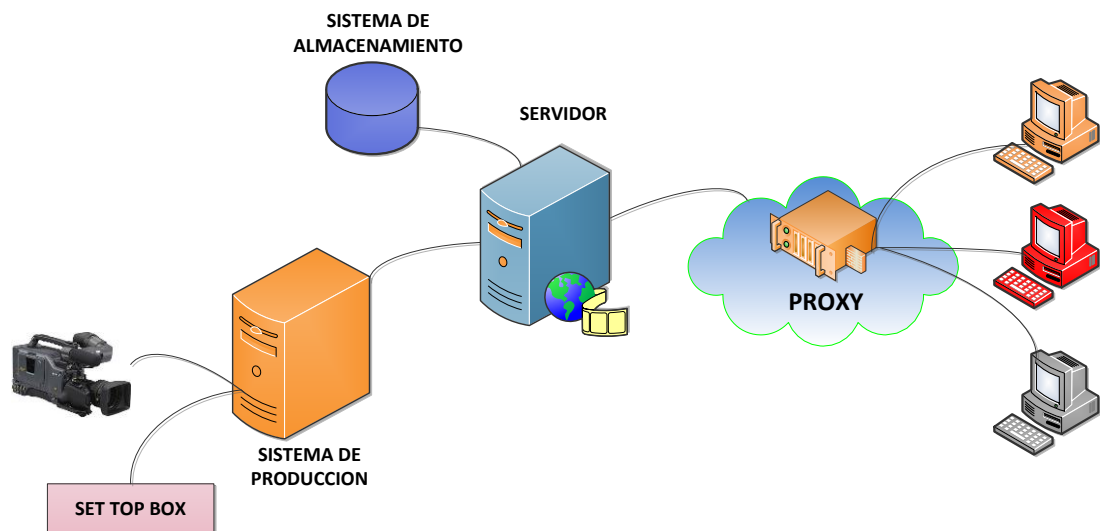


Figura 10. Elementos de arquitectura streaming

Arquitectura sin servidor (Server-Less): Este tipo de arquitectura no cuenta en su estructura con un servidor de A/V, por tanto el archivo es proporcionado al cliente por un servidor Web. Este tipo de arquitectura es también conocida como *pseudo-streaming* o *Fast-Start* (Alonso, 2009)

Arquitectura sin cliente: Este tipo de arquitectura es similar al funcionamiento del servicio de streaming bajo demanda con flujo de datos en directo. Dentro de su estructura no cuenta con el elemento cliente por lo que mediante el uso de un applet java o algún plugin es posible su visualización.

En esta sección se han presentado las arquitecturas generales que se puede encontrar al momento de ofrecer un servicio de streaming, sin embargo dependiendo del servicio que se desee ofertar los elementos presentes en la red de streaming pueden cambiar ligeramente. Por ejemplo para un servicio bajo demanda es primordial contar un sistema de almacenamiento, pero este elemento no es necesario para un servicio en directo (Alonso, 2009).

2.2.6. Protocolos de servicio streaming.

Como se mencionó en la sección 2.2.4, para brindar un servicio de streaming existe una amplia variedad de protocolos, cada uno de ellos cuenta con ciertas características que determinan su uso en los distintos escenarios que se pueden implementar. A continuación se describirá más a fondo los protocolos generalmente utilizados para el servicio de streaming.

Protocolo streaming en tiempo real (RTSP): De sus siglas en inglés, *Real Time Streaming Protocol* (Schulzrinne, 1998), el RTSP es el protocolo mediante el cual se establece y controla la sesión entre el cliente y el servidor. Específicamente se enfoca en el control del envío y la entrega de datos en tiempo real.

Las sesiones establecidas en el RTSP pueden ser una o varias, y en ellas se llevan a cabo acciones de pausar, retroceder o avanzar. En sí, el protocolo RTSP funciona como un control remoto para los servicios de streaming, ya que los datos como tal viajan a través de un canal distinto y sobre otro protocolo, por ejemplo RTP.

Protocolo de transporte en tiempo real (RTP): También conocido como *Real time Transport Protocol* (Sánchez, 2014), a diferencia del anterior, este protocolo se encarga de transportar de extremo a extremo los datos. Trabaja generalmente sobre el protocolo de transporte UDP, sin embargo en ciertos casos es posible utilizar el protocolo TCP.

RTP es un protocolo que no asegura la calidad de servicio, ya que, asume la posibilidad de variaciones en las características de la red y considera la existencia de pérdidas y retardos en el transcurso de la comunicación (Pozueco Álvarez, 2014). Por tanto trabaja conjuntamente con otros protocolos, como por ejemplo RTCP.

Protocolo de control de transporte en tiempo real (RTCP): Generalmente conocido como *Real Time Transport Control Protocol*, es el encargado de monitorear el envío de paquetes RTP, mediante la transmisión periódica de paquetes de control. La información de control responde a las métricas de pérdida de paquete, retardo, jitter, etc., las cuales permiten estimar la calidad del servicio prestado.

Protocolo de descripción de sesión (SDP): Conocido generalmente por sus siglas en inglés *Session Description Protocol* (Handley, Perkins, & Jacobson, 2006). EL SDP proporciona una representación estándar (identificadores, creadores de la sesión, flujos que componen la sesión; nombre del flujo, tipo de encriptación, etc.; direcciones IP, puertos, formatos para acceder a los medios, protocolos de transporte o formato de codificación (Pozueco Álvarez, 2014)), para iniciar sesiones multimedia. La

representación proporcionada es independiente de cómo se transporta la información.

HTTP streaming: En los últimos años el desarrollo tecnológico ha tenido un gran avance y con ellos se ha incrementado notablemente el tráfico de datos a través de la red, perteneciendo un alto porcentaje de este incremento al video streaming. Este aumento ha significado un nuevo desafío para los proveedores de servicio de video, quienes tienen la gran labor de ofrecer su contenido en diferentes formatos y así poder llegar a todos y cada uno de los nuevos equipos (smartphone, tablets, Smart TV...) (Casín Núñez & others, 2013).

Los protocolos descritos anteriormente en esta sección, tienen la necesidad de gestionar una sesión diferente por cada cliente, además varios de ellos requieren trabajar en conjunto (ejemplo RTP funciona junto a RTCP o RTSP), lo cual genera mayor cantidad de información a transmitir. Por otro lado en Internet existen un sin número de redes de distribución de contenidos (CDN⁸), las cuales al igual que varios firewalls no soportan paquetes provenientes de estos protocolos. Estas dificultades han sido solventadas con el HTTP streaming, ya que el tráfico generado por este, es admitido por la gran mayoría de firewalls y CDNs. Adicional con el HTTP streaming ya no es necesario mantener una sesión en el servidor (Casín Núñez & others, 2013).

Estas grandes innovaciones han impulsado a que la industria del streaming de video se esté basando en protocolos HTTP streaming como (Puga Placencia, 2013):

- HTTP Live Streaming (HLS) de Apple (patentada).
- Smooth Streaming de Microsoft (patentada).

⁸ *Content Delivery Network*

- HTTP Dynamic Streaming de Adobe (patentada).
- Dynamic Adaptive Streaming over HTTP.

Dynamic Adaptive Streaming over HTTP (DASH): Fue creado con el objetivo de una entrega adaptativa, eficiente y progresiva de contenidos a través de HTTP (Chanchi, 2015). DASH fue desarrollado por el grupo MPEG y publicado como ISO / IEC 23009-1: 2012, en el año 2012. Actualmente la norma se encuentra en su revisión ISO / IEC 23009-1: 2014 (Pozueco Álvarez, 2014).

Puga Placencia expresa que:

DASH tiene la intención de soportar un modelo de media streaming para la entrega de contenido multimedia en la que el control recae exclusivamente en el cliente. Los clientes pueden solicitar datos mediante el protocolo HTTP desde servidores web estándar que no tienen capacidades específicas DASH. Por consiguiente, la norma emitida por MPEG no se centra en los procedimientos del cliente o servidor, sino en los formatos de datos utilizados para prestar un DASH Media Presentation (p. 27).

2.2.7. Formatos para servicio streaming.

Al hablar de formatos para servicio streaming, es primordial tener claro la diferencia entre un formato contenedor y un códec de audio o video. El contenedor es como una cajita en la cual se transportan los contenidos de A/V. Propiamente el formato contenedor se relaciona con el formato del archivo, mientras que al hablar de un códec, se hace referencia a un formato específico para cada parte, audio o video. Adicional al A/V el contenido puede estar formado por dos elementos extras, los subtítulos y las imágenes. En la Figura 11, se muestra el formato de un contenedor.

Contenedor: Un flujo de streaming es un archivo formado por elementos que pueden ser: audio, video, subtítulos y/o imágenes, el contenedor hace referencia al formato del archivo (AVI, OGG, MP4, MKV, etc.). Un archivo puede incluir uno o varios de estos elementos, por ejemplo se puede tener un contenedor solo de audio, pero con múltiples canales, cada canal dispone de un idioma, en sí, un mismo archivo con varios idiomas.

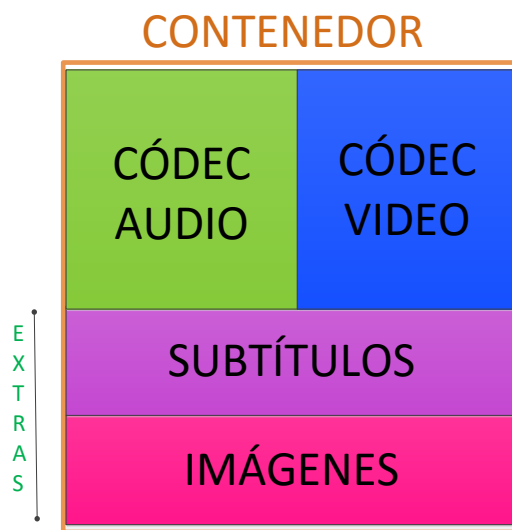


Figura 11. Formato contenedor.

Códec: Es el elemento encargado de la comprensión de un flujo ya sea de audio o de video. Generalmente el tipo de códec es confundido con el formato del archivo total de A/V, pero como ya se explicó son dos conceptos totalmente distintos. El uso de códecs no es mandatorio, sin embargo gracias a ellos se puede mejorar o empeorar la calidad, y a su vez también aumentar o disminuir el tamaño (Emezeta.com, 2015) .

Especificación: También conocida como estándar, son un conjunto de normas que definen las características de los códecs, las generalmente utilizadas son H.262, H.263, H.264.

Para el presente proyecto se analizaron los siguientes contenedores: MPEG-TS, OGG, WEBM.

- **MPEG-2 Transport Stream**, también conocido MPEG-TS, MTS o simplemente como TS, es un formato contenedor para el almacenamiento y transmisión de audio y video multiplexado. Generalmente es usado en tv digital y streaming (Afterdawn.com, 2016) .
- **OGG**, es un formato contenedor multimedia de código libre. Al igual que otros contenedores encapsula los datos comprimidos y unifica el audio y el video en un solo archivo, pero se diferencia de ellos ya que es un contenedor orientado al *stream*, es decir puede escribir y leer simultáneamente (Xiph.org, 2015).

La Figura 12 muestra la estructura lógica de un flujo de bits Ogg. El contenido a transmitir es empaquetado, cada paquete (Paket_1,....., Packet_n) es dividido en segmentos (Seg_1,...., Seg_n), los segmentos son transportados como una secuencia de páginas Ogg (PAGE_1,...., PAGE_n) concatenadas y a cada página se le antepone su encabezado correspondiente. Una página Ogg puede tener hasta 255 segmentos, y cada segmento puede contar hasta con 255 bytes. En una página de Ogg se pueden incluir una carga de hasta 65,025 bytes, no existe restricción en cuanto al tipo de contenido de la carga, es decir esta puede ser audio, video o datos extras. El tamaño de cada página puede variar, sin embargo el tamaño máximo de una página Ogg (carga + encabezado), es 65,307 byte (Kompendium.infotip.de, 2016).

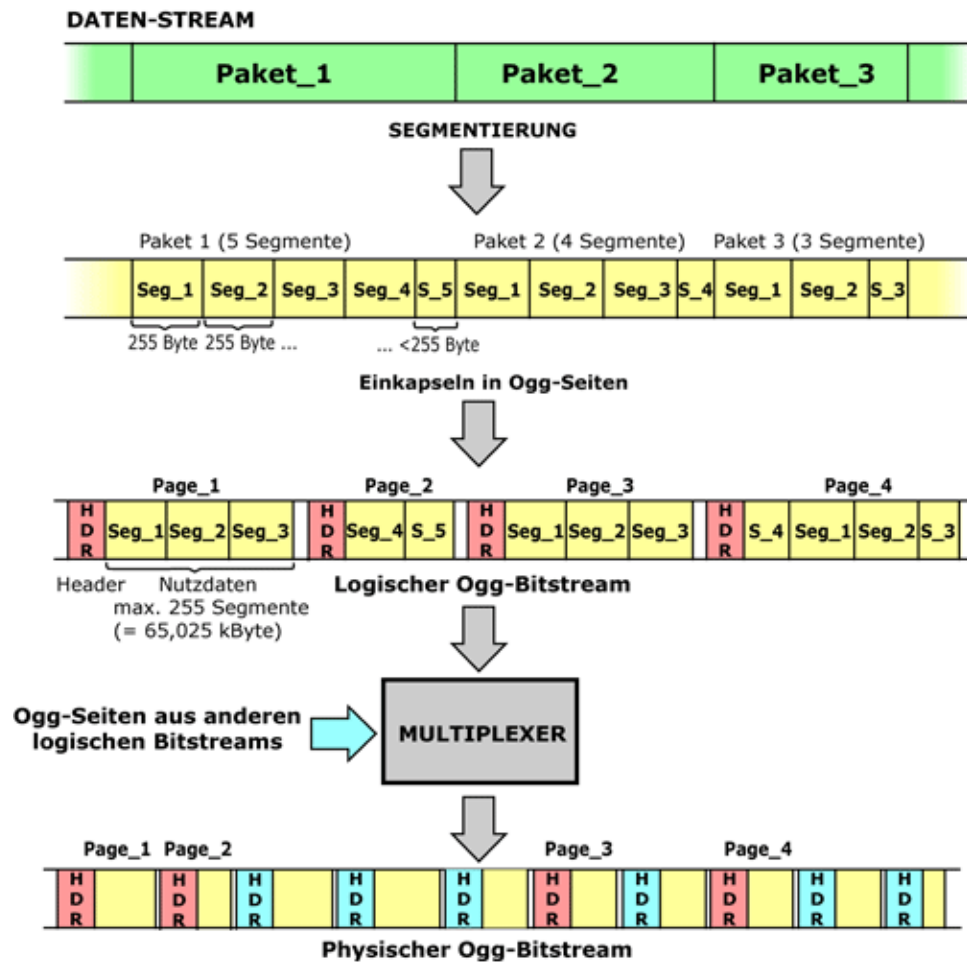


Figura 12. Estructura lógica de un flujo de bits Ogg.

- **WEBM**, al igual que OGG es un formato contenedor de código abierto, desarrollado por google y orientado a usarse con HTML5. Los archivos *.webm están formados por el códec de video VP8 y audio comprimido a través del contenedor Vorbis dentro de la estructura de un contenedor Matroska, Entre sus beneficios más destacados están: la apertura y la innovación (código libre) y su optimización para la web (Webm, 2015).

En la Tabla 3, se muestra una comparación de los principales contenedores con los códecs A/V que pueden albergar.

Tabla 3.
Contenedores y Codecs de A/V.

Contenedor	Códec audio	Códec video
MPEG-TS	mp3, AAC	MPEG-1, MPEG-2 Parte2, MPEG-4, H.264
OGG	Vorbis, Ogg, MPEG Audio	Theora, Dirac, OggUVS, MNG
WEBM	Vorbis, Opus	VP8, VP9

Fuente: (Wikipedia, 2015)

2.3. HTML5

Hypertext Markup Language (HTML) (Raggett, Le Hors, & Jacobs, 1999), o también conocido como el lenguaje de etiquetas. El código HTML es el lenguaje mediante el cual los navegadores pueden leer el contenido (estilo, imágenes, colores, texto...) de las páginas web.

HTML5 es la quinta versión de HTML, al igual que HTML tiene como objetivo interpretar páginas web pero añade ciertas mejoras. Las páginas implantadas con HTML5 son más rápidas y cuentan con diseños modernos, sin embargo el diseño dependerá de la creatividad del diseñador (Web, 2003).

HTML5 ha sido adoptado por los navegadores Google Chrome desde la versión 5, Firefox, Safari y Opera. De igual forma los *smartphone* iPhone, BlackBerry, Google Phone, entre otros han incorporado HTML5 (Van Lancker, 2013).

Las principales características de HTML5 son: estructura, estilo y funcionalidad, básicamente HTML5 es la combinación de HTML, hojas de estilos (CSS) y JavaScript donde (Gauchat, 2012):

- HTML se encarga de la estructura
- CSS presenta la estructura y su contenido
- JavaScript ejecuta tareas.

Uno de los grandes avances de HTML5 fue dejar de lado el uso de plugins, permitiendo una integración directa entre el contenido y el navegador, un claro ejemplo las etiquetas de audio y video. Es importante notar que HTML5 asegura compatibilidad, en términos de publicación web, con todo lo desarrollado anteriormente hasta su aparición (Van Lancker, 2013).

2.3.1. Etiquetas HTML5.

HTML5 es un lenguaje relativamente menos complicado que sus antecesores sin embargo es importante tener en cuenta algunas reglas que se deben respetar al momento de utilizar etiquetas en HTML5 (Van Lancker, 2013).

Tabla 4

Etiquetas HTML5

ETIQUETA	FUNCION
<i>video</i>	Inserta video sin la necesidad de plugins.
<i>object:</i>	Representa un recurso externo (imagen, un sub-documento HTML o un recurso externo a ser procesado por un plugin).
<i>embed:</i>	Inserta un plugin o contenido externo que no es HTML
<i>param:</i>	Define parámetros para los plugins llamados por los elementos <object>.

La etiqueta HTML5 cuenta con los siguientes atributos (Diez, Dominguez, Martinez, & Sáenz, 2012).

- Sintaxis de una etiqueta con atributo:

`<etiqueta atributo= "valor">... </etiqueta>`

- Las etiquetas no distinguen entre el uso de mayúsculas o minúsculas.

`<Etiqueta> = <ETIQUETA> = <etiqueta>`

- Toda etiqueta abierta debe ser cerrada.

`<etiqueta> </etiqueta>`

- Las etiquetas deben ser enlazadas correctamente, es decir en el orden en el cual se abran se deben cerrar.

`<etiqueta1> <etiqueta2> </etiqueta2> </etiqueta1>`

- Los valores de atributos deben estar siempre entre comillas.

`<etiqueta="video" width="400" </etiqueta>`

Como se mencionó anteriormente con el fin de mejorar su estructura, HTML5 ha introducido nuevos elementos como: *section*, *figure*, *template*, *article*, *audio*, *video*, *embed*, *source* entre otros (Html-differences.whatwg.org, 2016). El desarrollo del presente proyecto se enfocará en los elementos mostrados en la Tabla 4.

- **Etiqueta <video>**

La etiqueta <video> cuenta con soporte para los códecs más utilizados (Ogg Theora, H.264/MPEG-4, VP8/WebM), sin embargo actualmente existe el inconveniente de que no todos los navegadores admiten los mismos

- Src: URL del archivo que contiene el flujo a reproducir.
- Autoplay: Inicia la reproducción del video cuando la página termina de cargarse.

- Controls: Permite implementar los controles de un reproductor por defecto (play, pause, stop, volumen)
- Width: Delimita el ancho del cuadrilátero donde se mostrara la reproducción.
- Height: Delimita la altura del cuadrilátero donde se mostrara la reproducción
- Loop: Finalizada la reproducción, se inicia de nuevo automáticamente),
- Muted: Suprime el sonido de la reproducción.
- Poster: Imagen que se presenta mientras la reproducción está pausada o antes que la misma inicie.

El ejemplo del uso de la etiqueta de video:

```
<video width="600" height="500" src="video.mp4" controls autoplay preload>  
</video>
```

- **Etiqueta <objet>**

La etiqueta <objet> permite incluir cualquier tipo de contenido en las páginas HTML y cuenta con los siguientes atributos (Librosweb.es, 2016):

- Data: URL del archivo que contiene el flujo a reproducir.
- Width: Delimita el ancho del cuadrilátero donde se mostrara la reproducción.
- Height: Delimita la altura del cuadrilátero donde se mostrara la reproducción
- Type: Define el tipo de contenido de los datos

El ejemplo muestra el uso de la etiqueta objet::

```
<object data="video.mp4" type="application/mpeg" width="600"
height="500" />
```

A los objetos <object> se les puede definir información adicional en forma de parámetros con el uso de la etiqueta <param>.

- **Etiqueta <param>**

El uso de esta etiqueta está determinado estrictamente si se utiliza la etiqueta <object>, <param> siempre va dentro de <object> y tiene los atributos (Librosweb.es, 2016):

- name = Denota el nombre del parámetro
- value = Denota el valor del parámetro

Ejemplo:

```
<object data="video.mp4" type="application/mpeg" width="600" height="500"
/>
<param name="wmode" value="transparent" />
```

- **Etiqueta <embed>**

La etiqueta embed tiene un desempeño similar a la etiqueta de video, y al igual que ella cuenta con atributos como width, height, src, autoplay, preload (Aulaclie.es, 2016). El ejemplo de la etiqueta embed:

```
<embed width="600" height="500" src="video.mp4" autoplay preload>
</embed>
```

2.4. PAGINA WEB

2.4.1. Introducción.

Página web, página electrónica o ciberpágina (R.A.E, 2015), es un conjunto de información electrónica formada por texto, imágenes, videos, sonidos y muchos más elementos. Este contenido está descrito con el uso del lenguaje HTML. Una página puede estar compuesta por información sencilla, la cual no requiere la ayuda de un servidor, o información más compleja, teniendo que acceder a bases de datos, algoritmos de programación, entre otros.

2.4.2. Tipos de páginas web

- ***Página web estática.***

Una página web estática es aquella que está enfocada a mostrar contenido permanente e invariable, restringiendo al usuario la posibilidad de interactuar con ella. La Figura 13 muestra su estructura, como se observa básicamente está conformada por CSS, código HTML y JavaScript; no utiliza bases de datos ni programación compleja (E-duca.eu, 2015).

Este tipo de páginas representan un significativo ahorro de recursos para su creación y alojamiento, sin embargo constituyen un arduo trabajo si se desea modificar algún contenido, teniendo que cambiar el código HTML manualmente. Por lo general están presentes en sitios web que se limitan a mostrar información general de las actividades que desarrollan y de los productos o servicios que ofrecen.



Figura 13. Página web estática.

- **Página web dinámica.**

-

Al igual que las páginas web estáticas están formadas por HTML, CSS y JavaScript pero cuentan adicionalmente con PHP y MySQL, los cuales permiten agregar funciones y características interactivas en su diseño.

El lenguaje PHP se permite incluir programación compleja a nuestro sitio, mientras que con el uso del gestor de base de datos MySQL, se posibilita guardar el contenido de nuestra web en una base de datos en el servidor. Pudiendo disfrutar de aplicaciones como encuestas, votaciones, foros, adquisición de productos, chat on-line, enviar emails, entre otras (E-duca.eu, 2015).



Figura 14. Página web dinámica

Una página web dinámica se puede comprender como la unión de una página estática (HTML, CSS y JavaScript) más componentes dinámicos. La parte estática se alberga implícitamente en el cliente, mientras que el PHP y MySQL se encuentran alojados en el servidor, tal como se muestra en la Figura 14.

2.4.3. Sistema de Gestión de Contenidos (CMS).

Conocido como *Content Management System*, es una herramienta para crear y mantener de forma más fácil una página web. Permite administrar, organizar y presentar la información del sitio (E-duca.eu, 2015).

El CMS controla simultáneamente todas las bases de datos que contienen la información que se coloca en la página web, permitiendo manejar de forma más sencilla el contenido y el diseño. Gracias al uso de un CMS se evita contenido duplicado, archivos innecesarios en el servidor y enlaces rotos.

En sí, un sistema de gestión de contenidos es una página web dinámica de administración, la cual permite al usuario construir su propio contenido, gestionarlo, organizarlo y actualizarlo de manera más eficiente.

Los CMS más usados en internet según ALEXA⁹ son: WordPress, Drupal y Joomla (McClurg, 2015).

WordPress no es directamente un CMS, es más bien una herramienta enfocada para blog que ha evolucionado rápidamente. Es generalmente escogido por diseñadores de páginas web que están empezando, por su simplicidad y facilidad para ejecutarlo. WordPress no es muy escalable y es muy poco robusto en seguridad (BSIDESTUDIOS, 2015).

⁹ Motor de búsqueda especializado en categorización de páginas por ranking.

Drupal es preferentemente utilizado por los desarrolladores, es un CMS con un código claro y un diseño muy organizado. Cuenta con una gran cantidad de módulos, fáciles de instalar y los cuales facilitan un sin número de funcionalidades. Drupal está enfocado en sitios que manejan muchos usuarios, nodos con información, tiendas on-line, una red social masiva en sí, sitios que son muy dinámicos y deben ser actualizados frecuentemente (BSIDESTUDIOS, 2015).

Joomla es un sitio mayormente utilizado por diseñadores, es muy sencillo de instalar y cuenta con una interfaz fácil de usar. Este CMS basa su funcionamiento en plantillas permitiendo hacer web bastante atractivas, sin embargo los módulos que ofrece son bastante especializados y presentan ciertos problemas de integración con otros módulos, lo cual probablemente signifique el desarrollo de módulos direccionados a una función específica que se necesite, restando flexibilidad en el diseño (BSIDESTUDIOS, 2015).

2.5. SERVIDOR PARA DESARROLLO Y FUNCIONAMIENTO DE PAGINAS WEB.

Este tipo de servidor es un sistema unificado de infraestructura de internet que cuenta con herramientas para desarrollo y funcionamiento de páginas web, tales como: servidor web Apache, servidor de base de datos MySql y lenguaje PHP. Para la plataforma de Windows los más utilizados son WAMP y XAMPP.

WAMP solo se puede utilizar en Windows, cuenta con un menú que permite fácilmente realizar cambios en la configuración, sin embargo para activar o desactivar servicios es necesario ingresar a la configuración de cada servicio en el menú de Windows. Adicional el servidor WAMP cuenta un explorador que muestra los archivos de nuestro proyecto.

XAMP, también conocido como XAMPP porque adicional incluye el lenguaje de programación Perl. A diferencia del anterior, XAMP es multiplataforma y adicionalmente viene con el servidor FTP FileZilla.

2.6. ESTUDIO DEL ESTADO DEL ARTE.

Hoy por hoy hay un proceso de migración de televisión analógica a televisión digital, de ahí surge la importancia de la implementación de herramientas que permitan desarrollar mejoras en el sistema de TDT. Con el desarrollo de esta tecnología se han ido impulsando una gran cantidad de aplicaciones la mayoría de ellas direccionadas a ofrecerle al usuario interactividad con el contenido que se encuentra visualizando, sin perder su atención en el contenido que se está transmitiendo. Varios estudios describen sugerencias que pueden ser de utilidad en el diseño y pruebas de aplicaciones para TVDi, estas abarcan el modelo de interacción, la interfaz y el modelo de navegación (Febles Estrada & Pozo, 2013), permitiendo evaluar la usabilidad de las mismas.

WICOM/ESPETV forma parte de la RedAUTI (Red temática en Aplicaciones y Usabilidad de la Televisión digital Interactiva) y de la RiTdT (Red Iberoamericana para el desarrollo de la televisión digital terrestre y las aplicaciones interactivas) (Espetv.espe.edu.ec, 2016).

La RedAUTI está formada por 238 investigadores de 39 grupos de investigación de 13 países iberoamericanos, su objetivo es:

Apoyar la formación de los recursos humanos, estimular la investigación de forma transdisciplinaria y fortalecer los grupos de investigación de las Universidades iberoamericanas en temas relacionados con TVDI, avanzando en el diseño, implementación y despliegue de aplicaciones, servicios y

producción de contenidos para TVDI, en sus múltiples plataformas, de código abierto, de interés colectivo, para dar solución a problemas del contexto iberoamericano. (Redauti.net, 2016)

La RITDT, está conformada por 8 países entre ellos: Argentina, Bolivia, Brasil, Chile, Ecuador, España, Perú y Venezuela. Su propósito es: *“fomentar la creación e integración de grupos de investigación para el intercambio de conocimientos en el área de la Televisión Digital Terrestre (TDT), centrándose en el desarrollo de aplicaciones interactivas que impulsen la inclusión digital en Latinoamérica”*(Cytod.org, 2015).

Dentro del marco de colaboración con estas entidades y comprendiendo la necesidad de herramientas para colaborar con el desarrollo de la TDT, el grupo Wicom/ESPETV desarrolla un proyecto de investigación que permita mostrar la usabilidad de aplicaciones para televisión digital, este proyecto presenta un gran reto para mostrar al usuario el funcionamiento de su aplicación en tiempo real mediante un servicio de video streaming.

Los estudios encaminados a brindar un servicio de video streaming buscan proponer proyectos desarrollados en código abierto con la finalidad de que cualquier desarrollador o empresa enriquezca las funciones del actual sistema (Carrera Troncoso, 2015). Estos sistemas proponen funcionalidades que son iguales o mayores a sistemas disponibles en el mercado (Laura, Quiroz, & Córdova, n.d.), calidad de la experiencia (Carrera Troncoso, 2015) y su implementación en diferentes escenarios con el fin de solventar las distintas necesidades de los usuarios (Escobar, Martínez, & Cortés, n.d.).

Para brindar un correcto servicio de video streaming se debe seleccionar el protocolo y la arquitectura que mejor se adapte a los requerimientos necesarios, Carlos S. a través de su trabajo de maestría evalúa distintas arquitecturas (arquitectura típica, sin servidor, entre otras) y técnicas

(SVC,DASH,...) que adecuan desplegar servicios de calidad sobre la red LTE (Sánchez, 2014).

Es importante tomar en cuenta que la tecnología de streaming de video representa un elevado consumo de recursos y sugiere la necesidad de condiciones de transmisión estables para poder garantizar una cierta calidad (Pozueco Álvarez, 2014). Relacionado a estos problemas se han hecho muchas investigaciones para mejorar el nivel de operación de red. Laura Pozueco propone un estudio mediante el cual se aporta soluciones para la adaptación de contenidos multimedia transmitidos sobre un sistema de video streaming a través de la red (Pozueco Álvarez, 2014), por otro lado Beatriz Casín, estudia específicamente el protocolo DASH sobre HTTP, permitiendo tener una visión de las condiciones que se requiere para su implantación.

Por otro lado, concerniente al campo de aplicaciones interactivas existen un sin número de estudios que muestran desde los lineamientos para desarrollarlas hasta otros más avanzados que permiten la generación automática de las mismas. Por ejemplo Pillajo, plantea un Template generator, este es un software para la generación de plantillas interactivas GINGA-NCL, el cual a partir de plantilla prediseñadas, es capaz de crear aplicaciones interactivas Ginga-NCL para la TDT (Pillajo Bolagay, 2016), de forma similar Cevallos Salas, plantean la Generación Automática de Contenido para Aplicaciones Interactivas de Televisión Digital con Ginga-NCL (Cevallos David, 2016). Sin embargo la usabilidad de estas aplicaciones no puede ser mostrada al usuario de forma remota sobre un sistema real de televisión digital.

En la actualidad, el enfoque de desarrollo de una plataforma de usabilidad para TDT es un estudio propuesto en las JAUTI¹⁰, tiene como objetivo impulsar la investigación de esta nueva tecnología y permitir a

¹⁰ *Jornadas Iberoamericanas de Difusión y Capacitación sobre Aplicaciones y Usabilidad de la Televisión Digital Interactiva*

diseñadores gráficos, productores, investigadores, alumnos que se desenvuelven en este medio llevar a cabo pruebas en tiempo real de sus aplicaciones interactivas cuando son transmitidas sobre un sistema real de TDT, sin embargo este tema aún no está completamente definido y por ende poco probado.

Los estudios precedentemente citados, direccionan los lineamientos que deben tomarse en cuanto a usabilidad, interactividad así como para ofrecer un servicio de video streaming en tiempo real, no obstante no existe un estudio que unifique las tres líneas y se dirija en la implantación de una plataforma de usabilidad de aplicaciones interactivas para televisión digital. Por tanto en el presente trabajo de titulación, a través de la combinación de estos conocimientos, ha desarrollado como parte de la fase 3 de dicha plataforma, la implementación de video streaming para la visualización en tiempo real de la funcionalidad de una aplicación interactiva transmitida en BTS (Broadcast Transport Stream).

CAPÍTULO 3

DISEÑO

3.1. PLATAFORMA DE USABILIDAD DE TDT.

En la sección 1.2 se mencionó los requerimientos surgidos en las III Jornadas Iberoamericanas de difusión y capacitación sobre Aplicaciones y Usabilidad de la Televisión Digital Interactiva (TVDi) de la RedAUTI, como una solución a ellos la Universidad de las Fuerzas Armadas ESPE planteo el proyecto de investigación para la implantación de una Plataforma de Usabilidad de TDT, el cual se muestra en la Figura 1. El proyecto ha sido dividido en 4 fases. Cada fase ha sido planteada como un trabajo de titulación. A continuación vamos a conocer de forma breve cada uno de ellos.

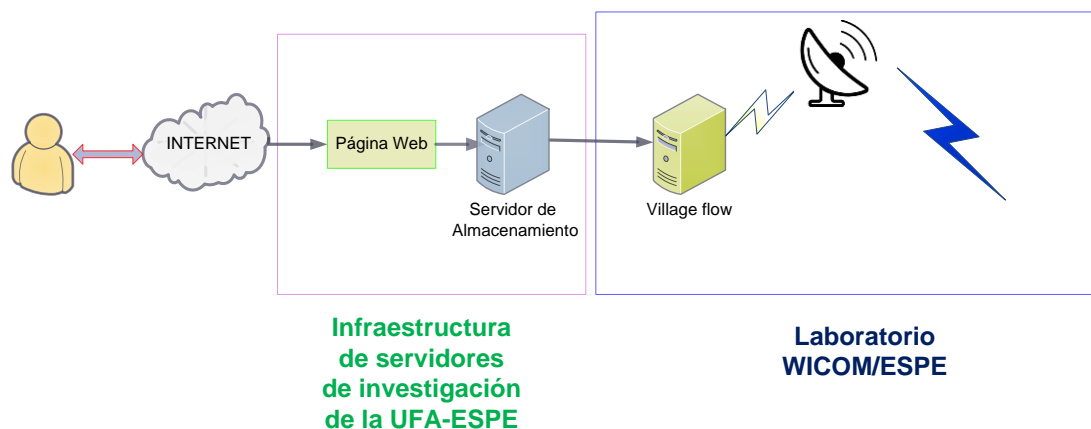


Figura 15. Página web

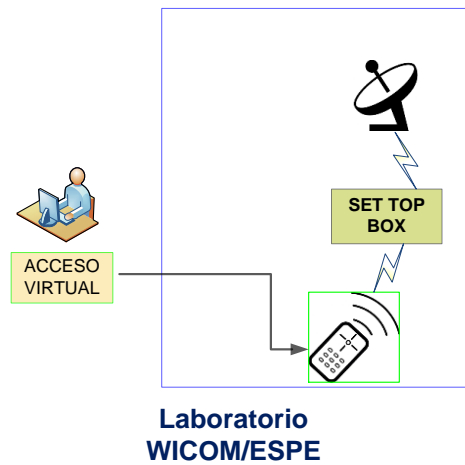


Figura 16. Prototipo control remoto

La primera fase se muestra a través de la

Figura 15, este contempla la creación de una página web que permita subir la información de aplicaciones, se guarden en el servidor de almacenamientos y posteriormente sean adicionadas de A/V y transmitidas por medio de ondas UHF.

La segunda fase (Ver Figura 16), se trata de un control remoto con acceso virtual, este será capaz de sustituir al control original que viene con el decodificador (STB). El control remoto esta implementado físicamente en el laboratorio WICOM/ESPE

La tercera fase consiste en implementar video streaming para la visualización en tiempo real de la funcionalidad de una aplicación interactiva transmitida en BTS. En la Figura 17 se muestra que parte de la Plataforma de Usabilidad de TDT abarca esta fase. Su implementación se describe en el presente proyecto de investigación

Finalmente la cuarta fase consiste en la integración de las tres fases anteriores, lo cual proporcionará como resultado la plataforma de usabilidad de TDT (Ver Figura 1).

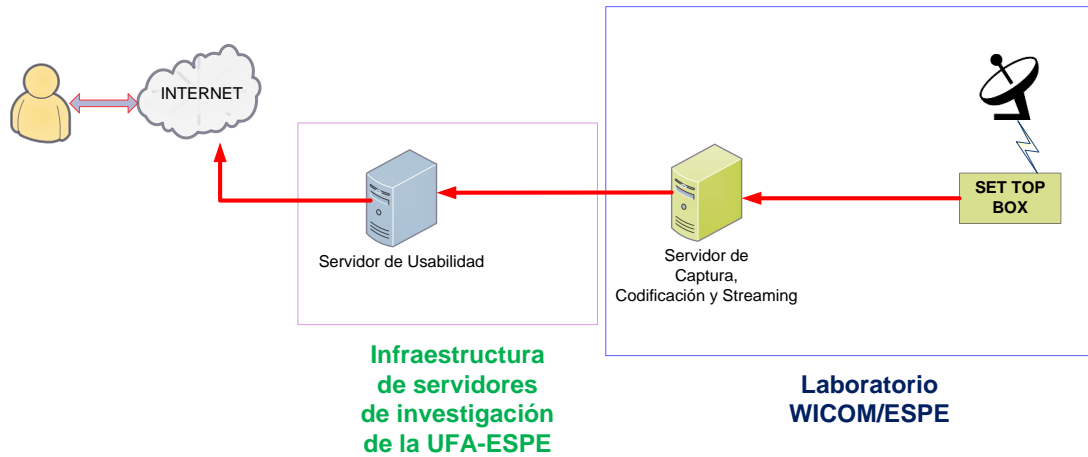


Figura 17. Video streaming

3.2. TOPOLOGÍA

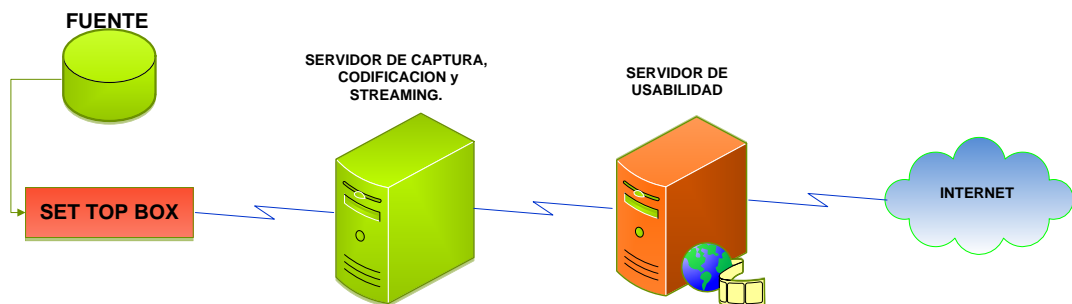


Figura 18. Topología de servicio streaming

La demanda del servicio streaming crece día a día y con ello surge la necesidad de encontrar nuevas alternativas para poder obtenerlo. Actualmente existen compañías dedicadas a brindar este tipo de servicio, sin embargo sus servicios están destinados netamente al consumo masivo de la teleaudiencia. Por ello para el presente proyecto tuvo la necesidad de

diseñar un prototipo que brinde las particularidades del video streaming, a un costo conveniente, sea usado para la transmisión de contenidos investigativos y expanda la línea de investigación para trabajos futuros. Al ser un proyecto netamente de aporte a la comunidad, sin esperar rentabilidad del mismo, uno de los grandes retos es trabajar con herramientas de bajo costo (PC de fáciles de encontrar en el mercado, software libre,...). Entre las características más complicadas a conseguir, está el obtener calidad de experiencia (QoE) de usuario en tiempo real.

Este proyecto busca brindar un interfaz que permita a los desarrollares de aplicaciones interactivas para TDT, tener un panorama claro sobre la usabilidad de su trabajo, impulsar a que continúen con su desarrollo y promover a nuevas personas a formar parte de este gran grupo de investigación. Tomando en cuenta estos requerimientos se ha planteado la topología que se muestra en la Figura 18. Como se observa la arquitectura seleccionada para la implantación del servicio streaming cuenta con un STB, el servidor de captura, codificación y streaming y el servidor de usabilidad, a continuación se detalla cada uno de estos elementos.

3.3. SERVIDOR DE USABILIDAD.

El servidor de usabilidad tiene como objetivo crear una interfaz entre el usuario y el laboratorio. Para su implantación, en primera instancia se debió elegir el sistema operativo sobre el cual se trabajaría, para ello se consideró el soporte de la tarjeta controladora para este sistema (Ver sección 3.4.1).

El sistema operativo Linux, fue descartado debido a que el soporte para Linux debe ser solicitado como herramienta adicional, por tal motivo se decidió utilizar el sistema operativo Windows, en su versión Windows 7, por su compatibilidad con aplicaciones (VLC, Drupal, XAMPP), interfaz,

estabilidad, entre otros. En base al sistema operativo se prosiguió a elegir las herramientas de software libre que ayudaron a desarrollar el servidor de usabilidad, para el diseño de una página WEB que sirva como GUI, se lo hizo mediante el uso del CMS Drupal y para montar el servidor que permita su desarrollo y funcionamiento se usó XAMPP.

Adicional al uso de Drupal y XAMPP, con el objetivo de dar flexibilidad a la visualización del presente proyecto, se utilizó la herramienta HTML5. El uso de esta herramienta para el servicio streaming, permite obtener la interfaz gráfica de usuario (página web) a través de los buscadores Chrome, Opera y Mozilla Firefox. Por otro lado en el navegador Firefox también se puede visualizar el resultado del presente proyecto mediante el uso del plugin de VLC, esta implementación se desarrolló en un nodo de la página web denominado Streaming con Plugin VLC.

En la sección 2.4.3 se menciona las principales características de Drupal, de igual forma los conceptos teóricos referentes a Drupal y XAMPP, se puntualizan en la sección 2.4 y 2.5 respectivamente. Detalles de su configuración consultar en Manual de Instalación adjunto.

Con la finalidad de ofrecer alta disponibilidad el servidor de usabilidad fue implementado de dos formas, físicamente sobre una PC de escritorio, en la laboratorio WICOM/ESPE (Ver Figura 19; **Error! No se encuentra el origen de la referencia.**), y virtualizado bajo la infraestructura de investigación de la Universidad de las Fuerzas Armadas ESPE. Actualmente el servidor de usabilidad físico se encuentra funcionando bajo la IP pública: 190.15.140.13 y el servidor virtual con la IP pública: 190.15.140.9.

Para determinar las particularidades del servidor de usabilidad físico se siguió la referencia de los requerimientos mínimos de hardware sobre los cuales se desempeña correctamente la tarjeta controladora (Ver sección 3.4.1), por tanto este servidor cuenta con las siguientes características:

- Sistema operativo: Windows 7 Professional
- Procesador: Intel® Core(TM) i7-2600 CPU @ 3.40GHz 3.40 GHz
- Memoria instalada (RAM): 6,00 GB
- Tipo de sistema: Sistema operativo de 32 bits.

En base a estos parámetros, se determinó los requerimientos para el servidor de usabilidad virtualizado:

- Sistema operativo: Windows 7
- Procesador: Intel® Xeon® CPU x5460 @3.16GHz (2 procesadores, 8 cores)
- Memoria instalada (RAM): 8,00 GB
- Tipo de sistema: Sistema operativo de 64 bits.

El contenido publicado en la página web, del servidor de usabilidad, es proveniente del flujo de datos que transmite el servidor de captura, codificación y streaming.

3.4. SERVIDOR DE CAPTURA, CODIFICACIÓN Y STREAMING.

El servidor de captura codificación y streaming, se encarga de tomar el flujo de datos, provenientes de los periféricos de salida del STB (Ver sección 3.5), para desarrollar esta función se analizó dos alternativas, la primera fue usar un extensor multiplicador HDMI a través de Ethernet Tx y Rx, por medio de esta opción se encontró limitaciones referentes a la fuente (únicamente cuenta con periférico de entrada tipo HDMI), además de no contar con gran escalabilidad, por tanto se analizó una segunda opción el usar una tarjeta capturadora de video (Ver sección 3.4.1). Posteriormente por medio del software VLC, toma en tiempo real la señal capturada, transforma la fuente original a otro formato de A/V y por medio del protocolo DASH over HTTP, la

emite para que sea tomada por el servidor de usabilidad y pueda formar parte del servicio streaming

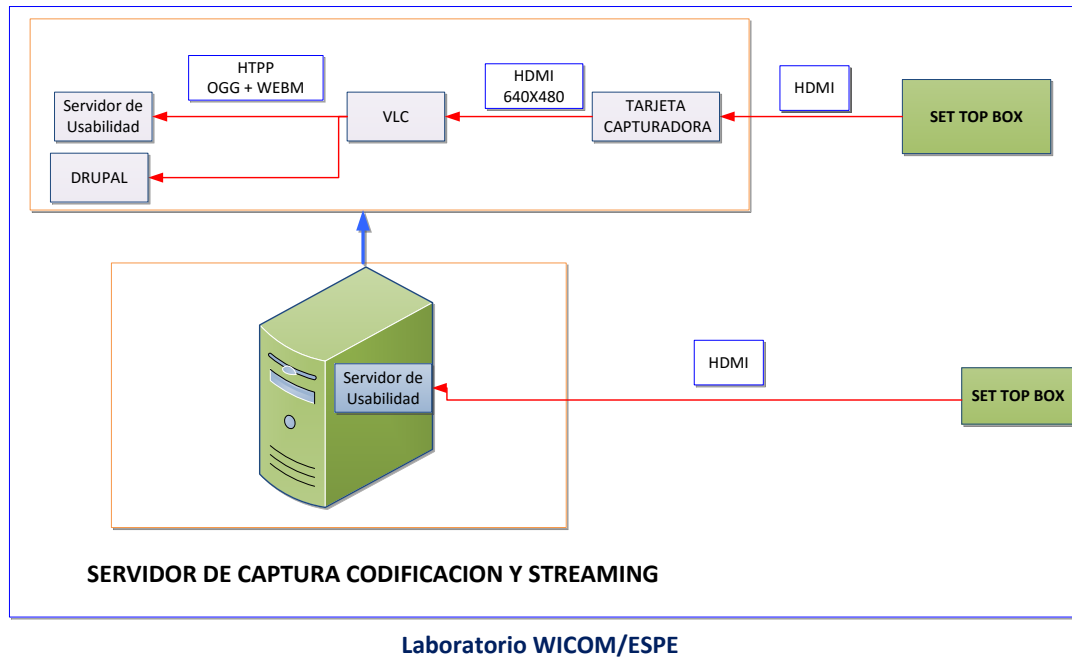


Figura 19. Sevidor de captura, codificación y streaming.

Al igual que el servidor de usabilidad fue desarrollado sobre el sistema operativo Windows 7 e implantado sobre la misma PC de escritorio en el Laboratorio WICOM/ESPE, tal como se muestra en la Figura 19

3.4.1. Tarjeta capturadora de video.

Para el desarrollo del presente proyecto de investigación fue necesaria la adquisición de una tarjeta capturadora, la cual por medio de su entrada HDMI permite capturar la señal de video que genera el STB. En la búsqueda de este elemento se encontró una gran cantidad opciones, sin embargo tomando en cuenta la facilidad para su adquisición y considerando aquellas que mejor se adaptan los requerimientos necesarios, se planteó tres opciones: C027 DARKCRYSTAL HD CAPTURE PRO, C129

DARKCRYSTAL HD CAPTURE SDK DUO, C727 DARKCRYSTAL HD CAPTURE SDK. Las tres propuestas pertenecen a AVerMedia Technologies. AVerMedia Technologies que es una empresa especializada en video de alta definición y en tiempo real.

Tabla 5

Comparación de tarjetas capturadoras (Avermedia.com, 2015).

Modelo	C027			C129			C727		
Interfaz	PCI-e x1			PCI-e x1			PCI-e x1		
Entradas	HDMI			2 HDMI			HDMI		
	YPBPR (Componentes)			2 YPBPR (Componentes)			YPBPR (Componentes)		
	RCA (Compuesta)			2 RCA (Compuesta)			RCA (Compuesta)		
	S-Video (Súper Vídeo)			2 S-Video (Súper Vídeo)			S-Video (Súper Vídeo)		
Sistema Operativo	Mac OS			Linux, Windows® 8/7/ Vista™/XP (32/64-Bit)			Windows 8/7/ Vista/ XP (32/64bits) , Linux		
Requerimientos mínimos	Captura en SD			Intel® Pentium® 4 3.0GHz	4	Intel Core2 Duo 2.4 GHz o superior	2	Intel® Pentium® 4 3.0GHz	4
				AMD Athlon™ 64 3200+	64	Athlon II X3 450 3.2 GHz o superior.	3	AMD Athlon™ 64 3200+	64
	Captura en HD (MPEG-2)			Intel® Core™2 Duo 2.4GHz	4	Intel Core i5 750 o superior.	2	Intel® Core™ 2 Duo 2.4GHz	4
				AMD Athlon™ Dual Core 2.8GHz	64x2	AMD equivalente o superior.	o	AMD Athlon™ Dual Core 2.8GHz	64x2
	Captura en HD (H.264)			Intel® Core™ i5 750 o AMD equivalente		Intel Core i5 3570 o AMD FX-8150 eight-core o superior.		Intel® Core™ i5 750 o AMD equivalente	
Captura de contenido 3D							Intel® Core™ 2 Duo o AMD Athlon x2 o superior		
Códec de video	AVI, MPEG-2, WMV, H.264.			AVI, WMV, MPEG2, H.264			AVI, WMV, MPEG2, H.264		
Compatible con el software	Restringido a software para el operativo			Telestream Wirecast, Adobe Flash Media Live Encoder 3, Adobe Premiere Pro CS, Ustream, Xsplit, Windows Encoder 9, Microsoft GraphEdit			VMix, Microsoft DirectShow, Adobe Premiere, Pro CS6, Corel, VideoStudio, CyberLink PowerDirector, CyberLink PowerProducer, Real Producer (v13), VLC Media Player, Sony Vegas, Pro 10, Adobe Flash Media, Live		

La Tabla 5, muestra una comparativa de las principales características de las propuestas seleccionadas. Los aspectos tomados en cuenta para la selección de la tarjeta capturadora de video fueron: tipo de interfaz, tipos de entradas, sistema operativo, requerimientos de mínimos de hardware, códec de video que ofrece y compatibilidad de software.

Posterior a la comparación de los parámetros, se escogió la tarjeta C727 DARKCRYSTAL HD CAPTURE SDK.

3.5. SET TOP BOX

Como se muestra en la Figura 20, para poder obtener las señales de TDT, se utilizó un STB el cual se conecta a una antena UHF que recepta las señales de televisión digital terrestre. EL STB que se utilizó el EITV Developer Box, el cual es *“un set-top box híbrido (ISDB-T e IPTV) enfocado en desarrolladores de aplicativos Ginga (DTV_i), compatible con los estándares declarativo (Ginga-NCL / Lua) e imperativo (Ginga-J / JavaDTV) que fueron especificados para el Sistema Brasileño de TV Digital (SBTVD)”*(Eitv.com.br, 2016).

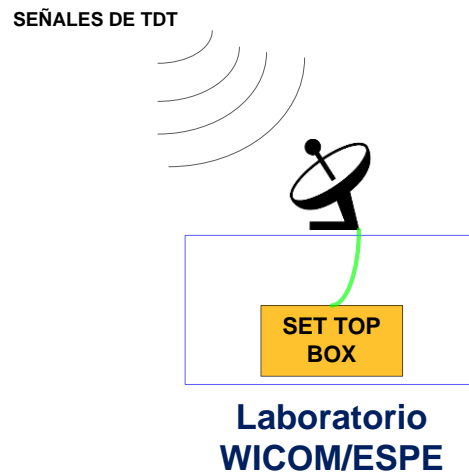


Figura 20. Recepción de señales de TDT.

El EiTV Developer Box cuenta con las siguientes características (Eitv.com.br, 2016):

- UHF: 470MHz(CH14) a 806MHz (CH69)
- VHF: 174MHz(CH7) a 216MHz (CH13)
- Señal: Compatible con el sistema ISDB-T
- Estándar: Rec. ITU-T H.264 (MPEG-4 AVC)
- Formatos: 480i y 1080i
- Sensor infrarrojo para el control remoto
- Interfaces de salida (Ver Figura 21):
 - Salida Digital de Audio y Vídeo (HDTV)
 - Salida de Vídeo Componente (YPbPr)
 - Salida de Audio Estéreo 1 (D + Y)
 - Salida de Vídeo Compuesto (CVBS – A/V)
 - Salida de Audio Estéreo 2 (D + Y)

- Salida de Audio Digital (SPDIF coaxial)
- Interactividad completa (DTVi – Ginga);
- Muestra sólo vídeo y audio del canal de difusión;
- Carga de aplicativos por aire, vía Internet o red local;
- Soporte simultáneo a canales ISDB-T e IPTV (vía UDP y RTP);
- Ajuste de la imagen en la pantalla del televisor;
- Compatible con los estándares de colores PAL-M y NTSC.



Figura 21. Vista trasera STB Fuente: (Eitv.com.br, 2016).

El STB cuenta con una gran cantidad de periféricos de salida, compatibles con las entradas de la tarjeta controladora, la conexión entre el STB y los servidores se realizó a través del puerto HDMI, como se puede observar en la Figura 22.

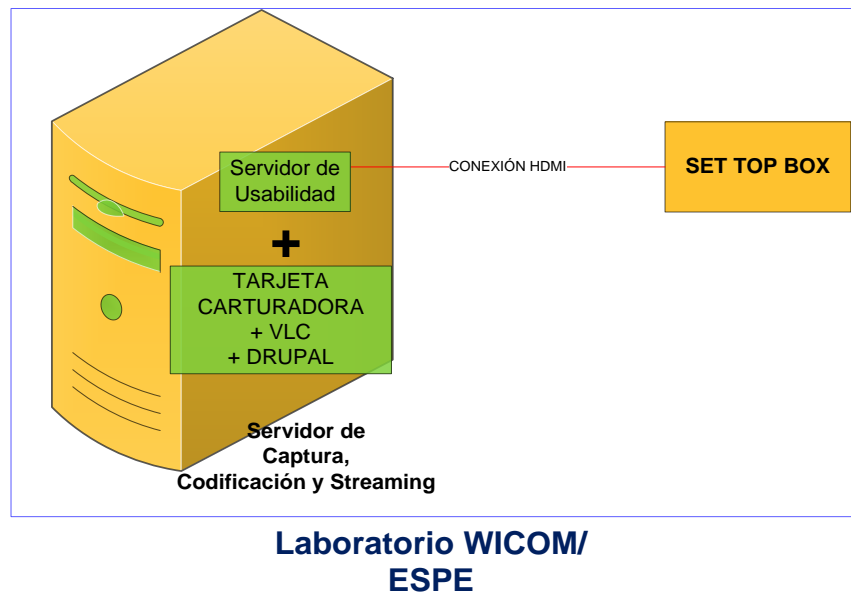


Figura 22. Conexión de los servidores con el STB.

CAPÍTULO 4

IMPLEMENTACIÓN, PRUEBAS Y ANÁLISIS DE RESULTADOS

4.1. IMPLEMENTACIÓN

El escenario implementado para obtener el servicio de video streaming para la visualización en tiempo real de la funcionalidad de una aplicación interactiva transmitida en BTS (Broadcast Transport Stream), se muestra en la Figura 23.

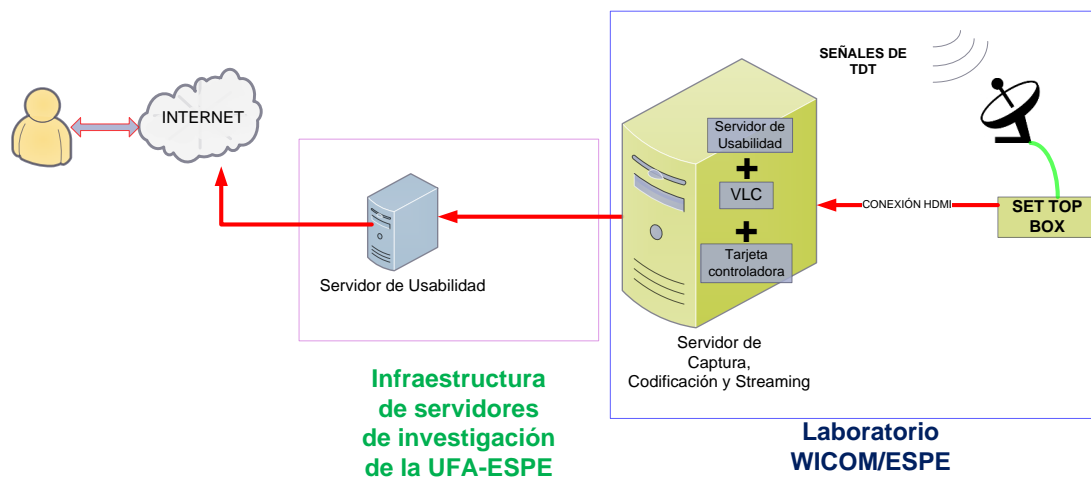


Figura 23. Escenario para la visualización del video streaming.

Para la implantación del presente proyecto de investigación se realizó:

1. **Conexión del STB con la PC de escritorio:** La tarjeta controladora fue instalada sobre un slot PCI express de la PC y por medio del puerto HDMI se obtiene la señal proveniente del STB.
2. **Desarrollo e implementación del servidor de captura, codificación y streaming:** Para su funcionamiento se instaló VLC en su versión 2.1.5, XAMPP en su versión 5.6.12 y Drupal versión 7.39.
3. **Desarrollo e implementación del servidor de usabilidad:** Al igual que el anterior se desarrolló sobre XAMPP en su versión 5.6.12 y Drupal versión 7.39.
4. **Virtualización del servidor de usabilidad:** Desarrollado sobre la infraestructura de servidores de la UFA-ESPE, utilizando la herramienta CİTRIX.

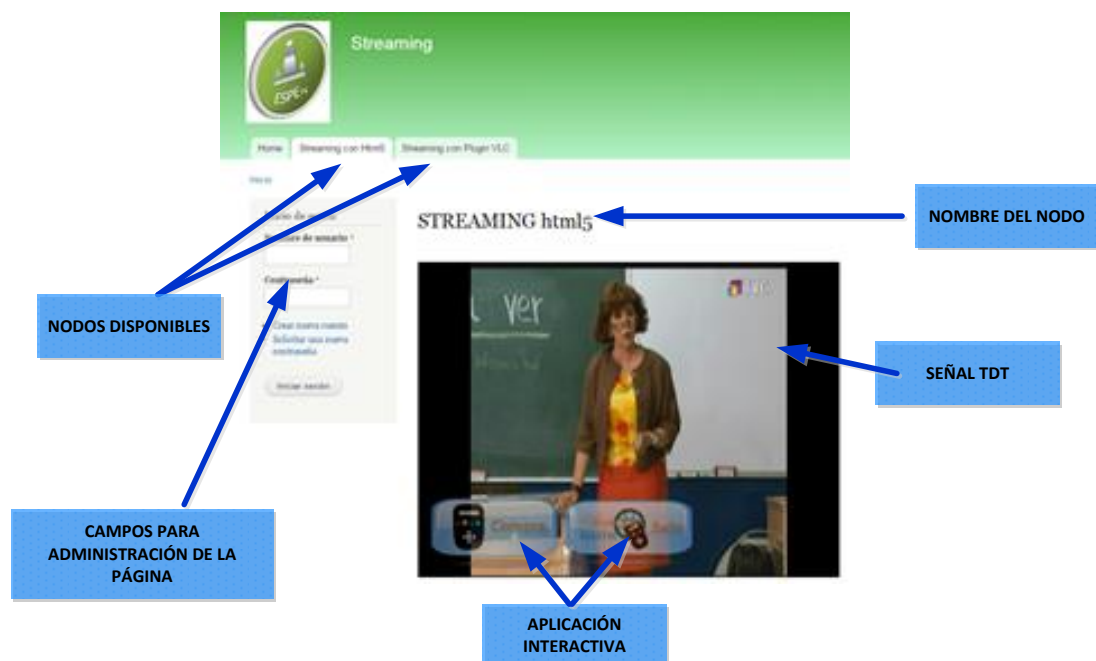


Figura 24. Servicio de video streaming implementado.



Figura 25. Streaming de una aplicación interactiva.

En los archivos adjuntos se puede observar los detalles de configuración para la implementación del presente proyecto. Las Figura 24 y Figura 25 muestran el servicio de video streaming implementado sobre el cual se realizó las pruebas de QoE. Como se puede observar se realizaron pruebas con aplicaciones interactivas, lo cual permitió comprobar la usabilidad de las mismas.

4.2. ENCUESTA MOS

Mediante la encuesta MOS que se muestra en el Apéndice D, se tiene como objetivo determinar la calidad de la experiencia (QoE) del presente proyecto de titulación.

Se utilizó una PC de escritorio conectada por cable Ethernet a la misma red sobre la cual está montado el servidor de usabilidad y una laptop conectada por conexión de red inalámbrica a una red externa.

Los navegadores que se seleccionaron fueron: Google Chrome y Mozilla Firefox. La página web cuenta con dos nodos (pestañas), Streaming con HTML5 y Streaming con Plugin de VLC (Ver Figura 24), el segundo únicamente puede ser visualizado utilizando Mozilla Firefox.

Como base a lo anteriormente mencionado, se determinó las posibilidades que se puede llegar a visualizar, estableciendo los siguientes escenarios:

1. ESCENARIO 1:

Buscador: Google Chrome

Nodo: Streaming con HTML5

Dispositivo: PC de escritorio

2. ESCENARIO 2:

Buscador: Mozilla Firefox.

Nodo: Streaming con Plugin VLC

Dispositivo: PC de escritorio

3. ESCENARIO 3:

Buscador: Mozilla Firefox.

Nodo: Streaming con Plugin VLC.

Dispositivo: PC de escritorio

4. ESCENARIO 4:

Buscador: Google Chrome

Nodo: Streaming con HTML5

Dispositivo: Laptop.

5. ESCENARIO 5:

Buscador: Mozilla Firefox.

Nodo: Streaming con Plugin VLC

Dispositivo: Laptop.

6. ESCENARIO 6:

Buscador: Mozilla Firefox.

Nodo: Streaming con Plugin VLC.

Dispositivo: Laptop.

La PC de escritorio cuenta con las siguientes características:

- Sistema operativo: Windows 7 Professional
- Procesador: Intel® Core(TM) i7-2600 CPU @ 3.40GHz 3.40 GHz
- Memoria instalada (RAM): 6,00 GB
- Tipo de sistema: Sistema operativo de 32 bits.
- Conexión a internet: Ethernet (misma red que los servidores)
- Upload: 9,80 Mbps
- Download: 79.91 Mbps
- Latencia: 8ms

Por otro lado, el dispositivo portátil (laptop) cuenta con las siguientes características:

- Sistema operativo: Windows 10 Pro.
- Procesador: Intel® Core(TM) i5-2450M CPU @ 2.50GHz 2.50 GHz
- Memoria instalada (RAM): 6,00 GB
- Tipo de sistema: Sistema operativo de 64 bits.
- Conexión a internet: Wifi (red externa)
- Upload: 2.41 Mbps
- Download: 0.05 Mbps
- Latencia: 165ms

Para determinar bajo qué escenario de los implementados (nodo, navegador, dispositivo) se desempeña mejor para el servicio de video streaming, se plantearon las siguientes preguntas:

1. ¿Cómo califica su experiencia de usuario al momento de cargar la página web?(Ver Figura 25)
2. ¿Cómo considera la calidad del video presentado en la página web? (Ver Figura 26)
3. ¿Cómo aprecia la sincronización entre el audio y el video en la página web?(Ver Figura 27)

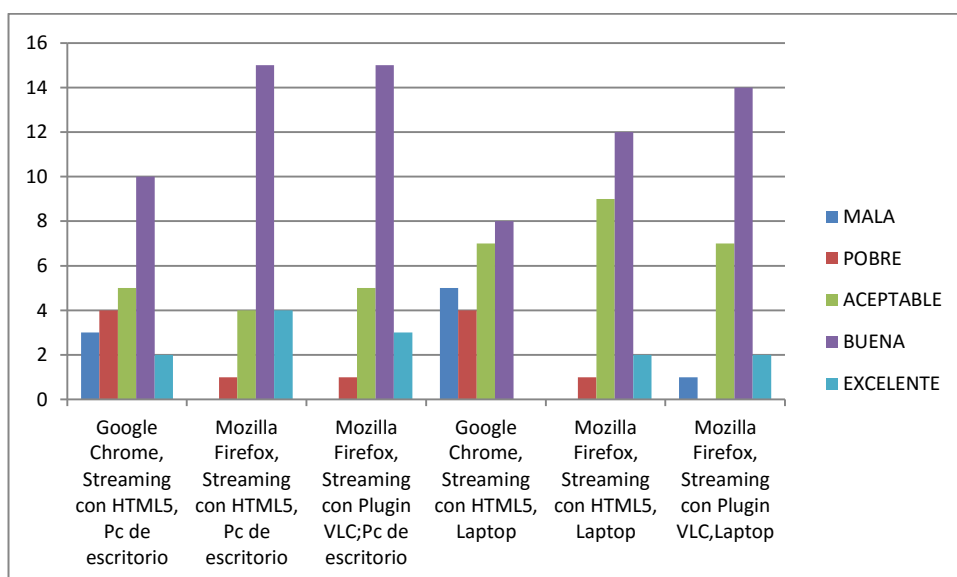


Figura 26. ¿Cómo considera la calidad del video presentado en la página web?

Adicionalmente se pidió a los encuestados que realizaran las siguientes comparaciones:

4. ¿De acuerdo a la experiencia apreciada que navegador considera que se acopla mejor al proyecto implementado?(Ver Figura 28)
5. ¿Acorde a lo experimentado en que dispositivo usted considera se obtuvo una mejor calidad de video streaming? (Ver Figura 28)

6. ¿Conforme a la experiencia percibida en que nodo pudo apreciar una mejor QoE? (Ver Figura 28)
7. ¿De los escenarios planteados, cual usted considera que se asemeja mejor a la señal mostrada en la tv digital? (Ver Figura 29 y Figura 30)

Las pruebas se desarrollaron aplicando la encuesta a 24 personas, actualmente son estudiantes de la Carrera Ingeniería Electrónica y Telecomunicaciones, obteniéndose los siguientes resultados.

La velocidad y la latencia del internet, son factores importantes que determina la calidad de la experiencia de usuario (Ver sección 4.3), sin embargo este no es el único factor determinante sobre la QoE. En las ilustraciones: Figura 25, Figura 26 y Figura 27, se puede apreciar que los escenarios implantados sobre el navegador Google Chrome son los más variantes y no presentan una inclinación hacia una QoE MALA o EXCELENTE, mientras que aquellos implantados sobre Mozilla Firefox tienen tendencia a presentar una QoE, BUENA o EXCELENTE.

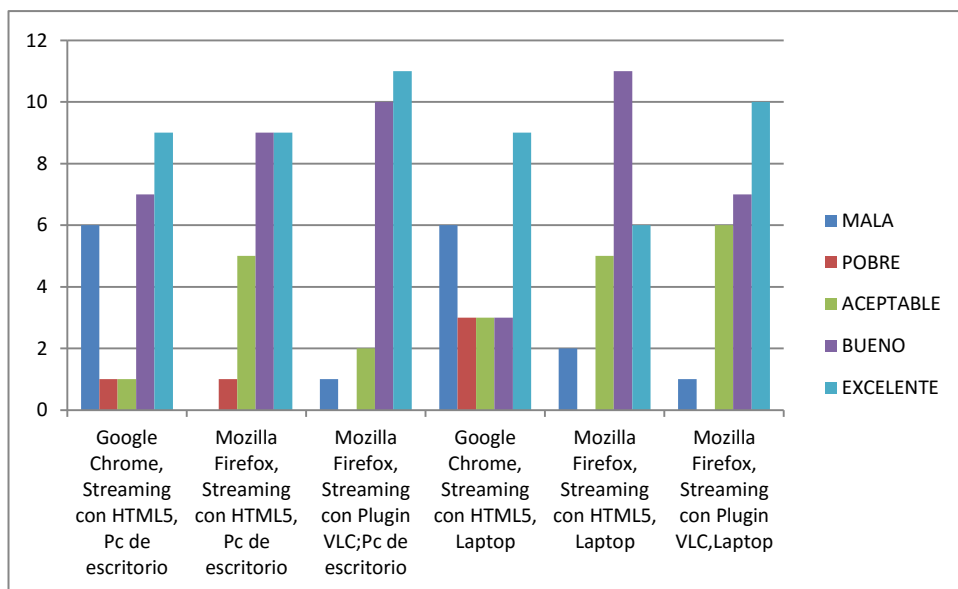


Figura 27. Cómo aprecia la sincronización entre el audio y el video en la página web?

Mediante un segundo bloque de preguntas se les pidió a los encuestados que determinen cual navegador, dispositivo y nodo permite una mejor QoE. En la Figura 28 se observa los resultados de la percepción obtenida, se evidencia que para 15 personas el mejor navegador fue Mozilla Firefox y para 19 de los 24 encuestados la PC de escritorio mostro mejores prestaciones. En cuanto al nodo la valoración de los encuestados mostro igualdad en la elección de los nodos.

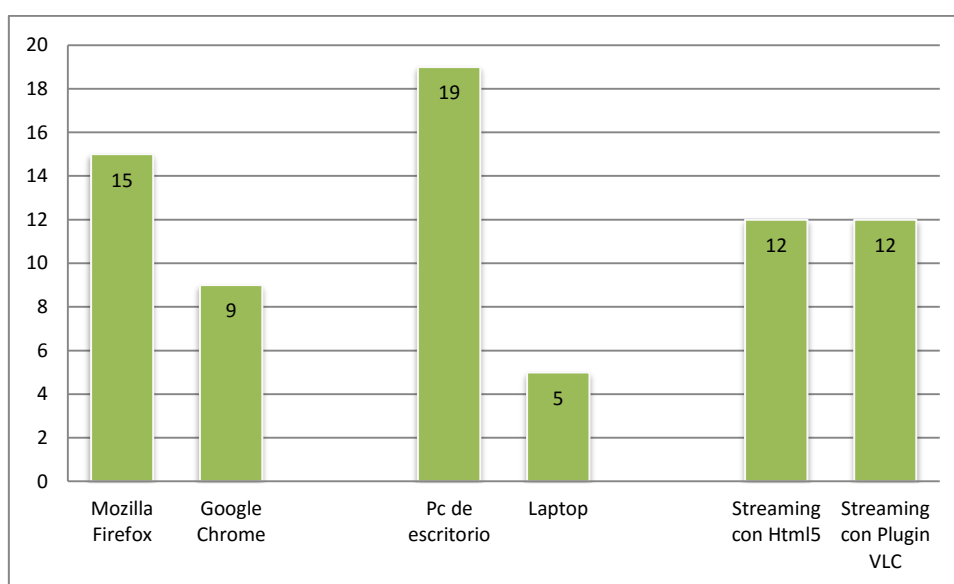


Figura 28. Comparación de navegador, dispositivo y nodo.

Finalmente se solicitó a los encuestados que evaluaran cuál de los escenarios tanto en la PC de escritorio como en dispositivo portátil se asemeja más a la señal de TDT directamente mostrada sobre una tv digital. En las PC de escritorio existieron dos escenarios que predominaron, el escenario 1 y el escenario 2, con un 33% y 50 % de aceptación respectivamente (Ver Figura 29). Por otro lado cuando se visualizó utilizando el dispositivo portátil, el escenario 6 (similar al escenario 3) sin lugar a duda mostro mejores resultados, con un 62% de aprobación, (Ver Figura 30).

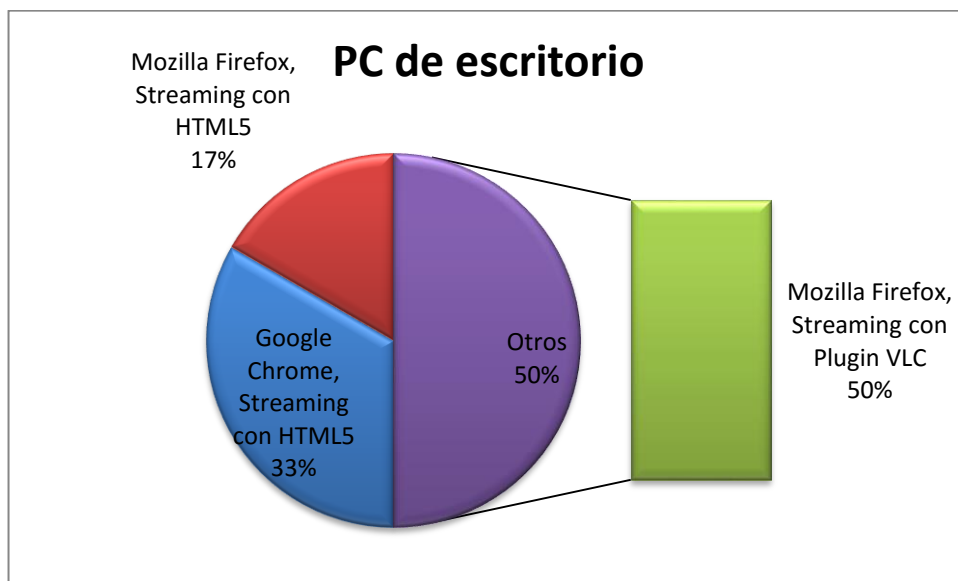


Figura 29. Comparativa PC de escritorio con tv digital.

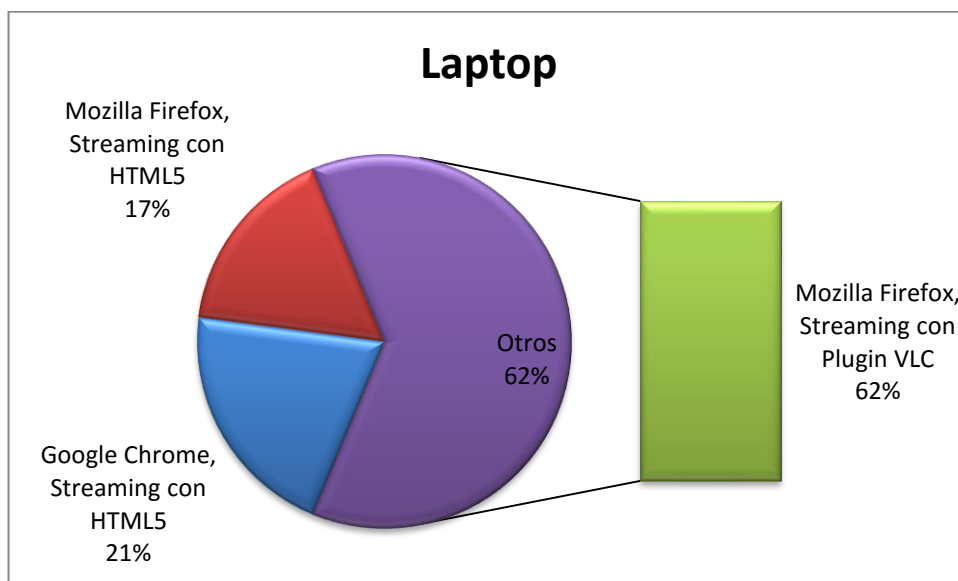


Figura 30. Comparativa dispositivo portátil con tv digital.

4.3. PRUEBAS REMOTAS.

Durante el desarrollo del presente proyecto se mencionó que la calidad del servicio streaming depende de la estabilidad de la red, haciendo referencia a métricas como la velocidad y latencia. Por tal motivo con el

objetivo de analizar la influencia de la red se desarrollaron pruebas remotas de los escenarios implementados sobre redes externas al laboratorio Wicom/ESPETV, obteniendo los siguientes resultados (Ver Tabla 6 **Error! No se encuentra el origen de la referencia.**)

Tabla 6.
Resultados pruebas remotas.

DOWNLOAD	UPLOAD	Latencia	Nro. intentos para cargar la página	Tiempo aprox. de carga.	Percepción de molestias una vez cargada la página.	Navegador
2.15Mb/s	0.35Mb/s	10 ms	4	2seg	no	Chrome
2.15Mb/s	0.35Mb/s	10 ms	1	1seg	no	Mozilla
2,46 Mb / s	0,38 Mb / s	10 ms	2	1.5seg	no	Chrome
2,46 Mb / s	0,38 Mb / s	10 ms	1	2seg	no	Mozilla
9,80 Mb / s	79.91 Mb / s	8ms	1	inmediato	no	Chrome
9,80 Mb / s	79.91 Mb / s	8ms	1	inmediato	no	Mozilla
9.99Mb/s	18,28 Mb / s	10ms	1	1.5seg	no	Chrome
9.99Mb/s	18,28 Mb / s	10ms	1	2seg	no	Mozilla
1.11Mb/s	1,27 Mb / s	50ms	1	5seg	si	Chrome
1.11Mb/s	1,27 Mb / s	50ms	1	2seg	no	Mozilla
1.98Mb/s	1,04 Mb / s	49ms	2	3seg	si	Chrome
1.98Mb/s	1,04 Mb / s	49ms	1	2seg	no	Mozilla
8.23Mb/s	9,68 Mb / s	11ms	1	15seg	no	Chrome
8.23Mb/s	9,68 Mb / s	11ms	1	20seg	no	Mozilla

4.4. ANÁLISIS DE RESULTADOS.

De acuerdo a los resultados obtenidos en la evaluación MOS, se observa que los escenarios que utilizaron el dispositivo portátil, nodo Streaming con HTML5 y el navegador Google Chrome tuvieron percepciones más variadas,

es decir de acuerdo a los encuestas su QoE no tuvo tendencia hacia una calidad MALA o EXCELENTE, mientras que aquellos que utilizaron la PC de escritorio, el nodo Streaming con Plugin VLC y el navegador Mozilla presentaron una QoE mayormente BUENA o EXCELENTE, lo cual se debe a inicialmente al tipo de conexión a internet que se utilizó.

La PC de escritorio al estar conectada directamente a la misma red que los servidores, y por medio de una conexión Ethernet tuvo mayor velocidad y menos latencia, por tal como era de esperarse los resultados de las pruebas desarrolladas sobre ella son similares y con mejor QoE, mientras que el dispositivo portátil estaba conectado a una red externa por medio del uso de la tecnología wifi, presento menor velocidad y mayor latencia, consecuentemente menor QoE.

Por otro lado referente al nodo y al tipo de navegador utilizado, mejor QoE se obtuvo utilizando el Mozilla y el nodo Streaming con Plugin VLC esto debido a que la sincronización con los metadatos en estos es más rápida y no exige menor velocidad de la conexión internet

De los resultados obtenidos se observa en la , la variación en la velocidad y latencia de red influyen inicialmente en el número de intentos para cargar la página, sumado a esto los requerimientos del nodo y navegador utilizados, el tiempo de carga de la página y las molestias una vez cargada la página también se ven influenciados Sin embargo si las condiciones de estabilidad de la red son pobres existe similar influencia en la QoE independientemente del navegador o nodo que se esté utilizando.

CAPÍTULO 5

CONCLUSIONES Y LINEAS DE TRABAJOS FUTUROS

5.1. CONCLUSIONES

- Se logró visualizar en tiempo real aplicaciones interactivas transmitidas en BTS, por medio del servicio de video streaming implementado sobre una página web.
- Mediante un análisis técnico que consideró, tipos de interfaz, tipos de entradas, sistema operativo, requerimientos de mínimos de hardware, códec de video ofertados y compatibilidad de software, se seleccionó la tarjeta capturadora AVerMedia C727 para obtener el flujo de datos desde el STB hacia el servidor de captura, codificación y streaming.
- Una característica fundamental que determinó la selección del software Drupal, XAMPP, HTML5 y VLC fue la escalabilidad y flexibilidad que estos permiten. HTML5 elimina el uso de plugins, mientras que VLC, Drupal y XAMPP son multiplataforma y de código abierto, lo cual aumenta la posibilidad de mejorar las funciones del actual sistema en trabajos futuros.
- Se implementó el servidor de streaming de forma física sobre un PC de escritorio en el laboratorio WICOM/ESPETV, y a su vez sobre la infraestructura de servidores virtuales de la UFA-ESPE, gracias a ello

se consiguió brindar alta disponibilidad del enlace entre el usuario y el laboratorio.

- Para la implementación de la plataforma de usabilidad de televisión digital del grupo Wicom/ESPE se requiere la visualización de interactividad de aplicaciones en tiempo real, mediante la implantación del servicio de video streaming, en el presente proyecto se ha finalizado la tercera fase y se ha logrado satisfacer dicho requerimiento.
- De los resultados obtenidos en la evaluación MOS sobre la calidad de la experiencia percibida en el presente proyecto, se constató que la variación en la velocidad y latencia de la red genera molestias en la calidad del video streaming.
- Se evidenció problemas al mostrar el contenido en la página web, principalmente al utilizar el nodo Streaming con HTML5 o el navegador Google Chrome esto debido a que para ambos casos la sincronización de los metadatos sugiere mayor velocidad y menor latencia en la red.
- Tanto el nodo Streaming con Plugin VLC como el navegador Mozilla Firefox presentaron mejores resultados en calidad en la experiencia percibida por el usuario, sin embargo si la velocidad y latencia de la red es demasiado deficiente al igual que en los anteriores no se puede asegurar calidad en la presentación de contenidos.
- Referente al dispositivo utilizado, el usuario presenció menos molestias al utilizar la PC de escritorio consecuencia de que esta estaba conectada mediante interfaz Ethernet a la misma red que el servidor de usabilidad. Tomar en cuenta que el tipo de conexión a

internet influye en la velocidad y latencia de la red y esta a su vez en la QoE del servicio streaming.

- Contrastando los resultados de la encuesta tipo MOS con los obtenidos de las pruebas remotas, se puede corroborar que la calidad del video streaming depende directamente de la velocidad y latencia de la red. La prueba remota con mejor velocidad y menor latencia evitó presenciar interrupciones.
- Comparando las dos pruebas se constató que el escenario más inestable se presenta cuando se utiliza el navegador Google Chrome, sin embargo si las prestaciones de la red son demasiado limitadas, a pesar de que Mozilla Firefox tiene mayor velocidad de respuesta para mostrar el contenido, también influyen en la apreciación de pixelado al cargar la página, lo cual toma un mayor número de intentos o un mayor tiempo de espera hasta que la pagina este totalmente cargada. Adicional también se pueden evidenciar problemas de calidad posteriores a la carga de la página.

5.2. LÍNEAS DE TRABAJOS FUTUROS.

- Con la culminación del presente trabajo de investigación se da paso para iniciar la integración de la plataforma de usabilidad de televisión digital.
- La evaluación del actual proyecto se ha enfocado únicamente en una estimación subjetiva, por tanto en trabajos futuros se puede incluir una evaluación objetiva a través de un estudio de las métricas de calidad de servicio.

- La tecnología de video streaming constituye un elevado consumo de recursos, por tanto para garantizar cierta calidad de servicio es importante considerar el estudio de métodos para mejorar la estabilidad de las condiciones bajo las cuales se realiza la transmisión.
- En el presente trabajo de investigación no se dimensionó el estudio de la marca de tiempo de presentación o también conocido como *presentation timestamp* (PTS), sin embargo se puede considerar mejoras futuras mediante la configuración de esta característica para solventar los inconvenientes de la sincronización de los metadatos con la carga de la página web, principalmente para mejorar la calidad de presentación tanto el nodo Streaming con HTML5 como el navegador Google Chrome.

CAPÍTULO 6

BIBLIOGRAFÍA

Acosta, F. (2012). Televisión Interactiva. *Seminario de Televisión Digital Terrestre - TDT*.

Afterdawn.com. (2016). MPEG-2 Transport Stream - AfterDawn: Glossary of technology terms & acronyms. Afterdawn.com. Retrieved from http://www.afterdawn.com/glossary/term.cfm/mpeg2_transport_stream

Alonso, F. J. S. (2009). Tecnologías de Streaming.

Aulaclíc.es. (2016). Tema 09 HTML. Retrieved from http://www.aulaclíc.es/html/t_9_1.htm

Avermedia.com. (2015). Spectacular Audiovisual Technologies | AVerMedia. Retrieved from <http://www.avermedia.com/>

BSIDESTUDIOS. (2015). CMS Battle Royale - Drupal/Wordpress/Joomla. Retrieved from <http://www.bsidestudios.com/blog/cms-battle-royale-drupalwordpressjoomla>

Carlos Carvajal Álvarez, M. G. P. y C. M. Á. Sergio González Pérez. (2015). Streaming Real Media Networks + SMIL. Retrieved from [https://sites.google.com/site/smirealmedia/home/arquitectura-del-servicio-](https://sites.google.com/site/smirealmedia/home/arquitectura-del-servicio-1)

- Carrera Troncoso, J. A. (2015). Diseño e implementación de una plataforma de estudio de calidad de experiencia, en streaming de video.
- Casín Núñez, C. B., & others. (2013). DASH: un estándar MPEG para streaming sobre HTTP.
- Cevallos David, R. M. Cevallos Fernando Bernal Ivan. (2016). Generación Automática de Contenido para Aplicaciones Interactivas de Televisión Digital con Ginga-NCL. Retrieved from <http://es.slideshare.net/RedAUTI/jauti-2014>
- Chanchi, M. G. E. (2015). ARQUITECTURA BASADA EN CONTEXTO PARA EL SOPORTE DEL SERVICIO DE VOD DE IPTV MÓVIL, APOYADA EN SISTEMAS DE RECOMENDACIONES Y STREAMING ADAPTATIVO.
- Cyted.org. (2015). Red Iberoamericana para el desarrollo de la televisión digital terrestre y las aplicaciones interactivas (RITDT) | CYTED. Programa Iberoamericano de Ciencia y Tecnología para el Desarrollo. Retrieved from <http://www.cyted.org/es/node/4634>
- Diez, T., Dominguez, M., Martinez, J., & Sáenz, J. (2012). Creación de páginas web accesibles con HTML5, 26.
- E-duca.eu. (2015). Diseño y Formación de páginas web con Drupal en Madrid | Asociación de apoyo al desarrollo web. Retrieved from <http://www.e-duca.eu/>

Eitv.com.br. (2016). EITV Developer Box | EITV. EITV - Entretenimiento y Interactividad para Televisión Digital. Retrieved from <http://www.eitv.com.br/es/produtos/eitv-developer-box/>

Emezeta.com. (2015). Emezeta.com. Retrieved from <http://www.emezeta.com/articulos/formatos-de-video-todo-lo-que-deberias-saber>

Escobar, R. F., Martínez, C., & Cortés, E. (n.d.). Implementación de Servicios de media Streaming sobre IPV6 en Protocolos HTTP y RTP en una Red Académica de Tecnología Avanzada. *Cadernos de Informática*, 6(1), 261–264.

Espetv.espe.edu.ec. (2016). Noticias. Retrieved from <http://www.espetv.espe.edu.ec/webespetv/Inicio/Inicio.html>

Febles Estrada, A., & Pozo, D. (2013). Evaluando la usabilidad en aplicaciones para la TVDi. *I Jornadas Iberoamericanas de Difusión y Capacitación sobre Televisión Digital Interactiva*.

Gauchat, J. D. (2012). *El gran libro de HTML5, CSS3 y JavaScript*. Marcombo.

Handley, M., Perkins, C., & Jacobson, V. (2006). SDP: session description protocol.

how.com, W. when. (2016). Traffic Types (QOS-Enabled Networks) Part 3. Retrieved from <http://what-when-how.com/qos-enabled-networks/traffic-types-qos-enabled-networks-part-3/>

Html-differences.whatwg.org. (2016). Differences from HTML4. Retrieved from <https://html-differences.whatwg.org/>

Ibnoulkhatib, G. (2012). *Sistema de captura de imágenes de streaming para el mantenimiento de la continuidad de la señal de TV en la UPV-TV.*

Información, M. de Telecomunicaciones y Sociedad de la. (2015). Televisión Digital Terrestre en el Ecuador. Retrieved from <http://www.telecomunicaciones.gob.ec/television-digital-terrestre-en-el-ecuador/>

Kompendium.infotip.de. (2016). Codecs und Container - InfoTip Kompendium. Retrieved from <http://kompendium.infotip.de/codecs-und-container.html>

Laura, J., Quiroz, J. L., & Córdova, C. (n.d.). Diseño e implementación de un Sistema de Video Streaming para la capacitación a distancia de profesionales y familias de personas con habilidades diferentes del Centro Ann Sullivan del Perú-CASP Design and development of a video streaming system. *ECIPERU*, 28.

Librosweb.es. (2016). 6.3. Objetos (Introducción a XHTML). Retrieved from http://librosweb.es/libro/xhtml/capitulo_6/objetos.html

Lloret, J., García, M., & Boronat, F. (2008). IPTV: la televisión por Internet. *Editorial Vértice, Málaga, España*, 230.

McClurg, F. R. (2015). CMS Comparison. Retrieved from <https://www.icts.uiowa.edu/confluence/display/ICTSit/CMS+Comparison>

Pillajo Bolagay, C. A. (2016). Template generator, software para la generación de plantillas interactivas GINGA-NCL. Retrieved from <http://repositorio.espe.edu.ec/handle/21000/10265>

Pozueco Álvarez, L. (2014). Diseño y evaluación de sistemas de estimación de ancho de banda disponible para servicios adaptativos de video streaming.

Puga Placencia, D. F. (2013). Análisis de los servicios y tecnologías de video over the top para móviles.

R.A.E. (2015). Retrieved from <http://lema.rae.es/dpd/?key=web>

Raggett, D., Le Hors, A., & Jacobs, I. (1999). HTML 4.01 Specification. *W3C recommendation, 24*.

Redauti.net. (2016). Información | redauti.net. Redauti.net. Retrieved from <http://redauti.net/info>

Sánchez, C. M. L. (2014). EVALUACIÓN DE ARQUITECTURAS Y PROTOCOLOS EN SERVICIOS DE STREAMING UNICAST Y MULTICAST SOBRE LTE.

Schulzrinne, H. (1998). Real time streaming protocol (RTSP).

Sotomayor Jácome, P. F. (2009). Análisis de los estándares de televisión digital terrestre (TDT) y pruebas de campo utilizando los equipos de comprobación técnica de la Superintendencia de Telecomunicaciones.

Van Lancker, L. (2013). *HTML5 y CSS3: Domine los estándares de las aplicaciones Web [2textordfeminine edición]*. Ediciones ENI.

Web, S. (2003). Web page.

Webm. (2015). Retrieved from <http://www.webmproject.org/about/>

Wikipedia. (2015). Comparison of video container formats. Retrieved from https://en.wikipedia.org/wiki/Comparison_of_video_container_formats

Xiph.org. (2015). Xiph. Retrieved from <https://xiph.org/ogg/>