



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

IMPLEMENTACIÓN DE PLATAFORMA PARA *STREAMING* DE VÍDEO EN TIEMPO REAL, A PARTIR DE TECNOLOGÍAS LIBRES

Axel Omar Meza Arrecis

Asesorado por el Ing. Mario Joaquín Trujillo Ramírez

Guatemala, noviembre de 2012

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**IMPLEMENTACIÓN DE PLATAFORMA PARA *STREAMING* DE VÍDEO EN
TIEMPO REAL, A PARTIR DE TECNOLOGÍAS LIBRES**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

AXEL OMAR MEZA ARRECIS

ASESORADO POR EL ING. MARIO JOAQUÍN TRUJILLO RAMÍREZ

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, NOVIEMBRE DE 2012

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Juan Carlos Molina Jiménez
VOCAL V	Br. Mario Maldonado Muralles
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Cesar Augusto Fernández Campos
EXAMINADOR	Ing. Oscar Alejandro Paz Campos
EXAMINADOR	Ing. Marlon Francisco Orellana López
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

IMPLEMENTACIÓN DE PLATAFORMA PARA *STREAMING* DE VÍDEO EN TIEMPO REAL, A PARTIR DE TECNOLOGÍAS LIBRES

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha julio de 2011.



Axel Omar Meza Arrecis

Guatemala 20 de abril de 2012

Ingeniero
Carlos Azurdia
Facultad de ingeniería
Universidad de San Carlos de Guatemala

Ingeniero Azurdia:

Deseándole éxitos en sus actividades diarias me dirijo a su persona para informarle que he revisado el trabajo de graduación del estudiante Axel Omar Meza Arrecis, quien se identifica con el número de carné 200312549; trabajo que se titula IMPLEMENTACIÓN DE PLATAFORMA PARA *STREAMING* DE VÍDEO EN TIEMPO REAL, A PARTIR DE TECNOLOGÍAS LIBRES, el cual considero que cumple con los objetivos planteados en el protocolo, por tanto cuenta con mi visto bueno para dar por concluido el mismo.

Sin nada más que agregar, me despido de usted agradeciéndole su atención a la presente.

Atentamente,

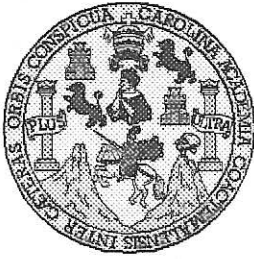


Mario Joaquín Trujillo Ramírez
INGENIERO EN CIENCIAS Y SISTEMAS
Colegiado No. 8831

Ing. Mario Joaquín Trujillo Ramírez

Asesor de trabajo de graduación

Colegiado No. 8831



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 2 de Mayo de 2012


Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **AXEL OMAR MEZA ARRECIS** carné 2003-12549, titulado: "**IMPLEMENTACIÓN DE PLATAFORMA PARA STREAMING DE VIDEO EN TIEMPO REAL, A PARTIR DE TECNOLOGÍAS LIBRES**", y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación titulado **“IMPLEMENTACIÓN DE PLATAFORMA PARA STREAMING DE VIDEO EN TIEMPO REAL, A PARTIR DE TECNOLOGÍAS LIBRES”**, presentado por el estudiante AXEL OMAR MEZA ARRECIS, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

Ing. Marlen Antonio Pérez Turk
Director, Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 27 de noviembre 2012

Universidad de San Carlos
de Guatemala



Facultad de Ingeniería
Decanato

DTG. 634 .2012

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **IMPLEMENTACIÓN DE PLATAFORMA PARA STREAMING DE VÍDEO EN TIEMPO REAL, A PARTIR DE TECNOLOGÍAS LIBRES**, presentado por el estudiante universitario **Axel Omar Meza Arrecis**, autoriza la impresión del mismo.

IMPRÍMASE:


Ing. Murphy Olympo Paiz Recinos
Decano



Guatemala, 27 de noviembre de 2012.

/gdech

ACTO QUE DEDICO A:

- Mi madre** Imelda Arrecis, quien con su esfuerzo y dedicación me permitió concluir mis estudios, me guió por el camino correcto con ejemplo de vida y porque este logro es tanto suyo como mío.
- Mis hermanos** Iván e Irvin Meza Arrecis, con quienes convivo, comparto y han sido un ejemplo en mi camino.
- Mis amigos** Sergio Lemus, Sergio Rodríguez y Cristian Godoy, con quienes compartí durante el camino de la universidad.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
GLOSARIO	VII
RESUMEN.....	XIII
OBJETIVOS.....	XV
INTRODUCCIÓN	XVII
1. DIFUSIÓN DE CONTENIDO MULTIMEDIA, INICIOS Y EVOLUCIÓN.....	1
1.1. Multimedia.....	1
1.2. Primeros pasos	2
1.3. Evolución.....	4
1.4. Modelo OSI y la transmisión de datos	5
1.4.1. Capa física	6
1.4.2. Capa de enlace de datos.....	6
1.4.3. Capa de red.....	6
1.4.4. Capa de transporte.....	7
1.4.5. Capa de sesión.....	7
1.4.6. Capa de presentación	7
1.4.7. Capa de aplicación	8
1.5. Transmisión de multimedia a través de Internet	8
1.5.1. Unidifusión.....	9
1.5.2. Multidifusión	10
1.5.3. Difusión	11
1.5.4. Multidifusión P2P	12
1.6. <i>Streaming</i>	13
1.6.1. Tipos de <i>streaming</i>	15

	1.6.1.1.	Por demanda	15
	1.6.1.2.	En directo	15
1.6.2.		Elementos fundamentales para <i>streaming</i>	16
	1.6.2.1.	Servidor.....	16
	1.6.2.2.	Cliente	16
	1.6.2.3.	Medio	17
1.6.3.		Enfoque de servidor	17
	1.6.3.1.	Flash media server.....	18
	1.6.3.2.	Wowza media server.....	18
	1.6.3.3.	Red5 media server.....	18
1.6.4.		Enfoque de transporte.....	18
	1.6.4.1.	User datagram protocol (UDP).....	19
	1.6.4.2.	Real time streaming protocol	19
1.6.5.		Enfoque de cliente	20
	1.6.5.1.	Precarga de datos.....	20
	1.6.5.2.	Codecs.....	20
	1.6.5.3.	Reproductor	21
2.		HERRAMIENTAS PARA DIFUNDIR CONTENIDO MULTIMEDIA	23
	2.1.	Icecast.....	24
	2.2.	Flash media server.....	25
	2.3.	Wowza media server.....	27
	2.4.	Red5	30
	2.4.1.	Juegos multiusuario	31
	2.4.2.	Aplicaciones SIP	31
	2.4.3.	Aplicaciones de <i>chat</i>	32
	2.5.	Características, ventajas y desventajas	32

3.	DISEÑO Y DESARROLLO DE LA SOLUCIÓN DE SOFTWARE	37
3.1.	Selección de servidor	37
3.1.1.	Necesidades.....	37
3.2.	Descripción de herramientas.....	38
3.2.1.	Java.....	39
3.2.2.	Eclipse Galileo.....	39
3.2.3.	Red5 plugin	40
3.2.4.	Apache	40
3.2.5.	MySQL	41
3.2.6.	Codificador/decodificador de vídeo	41
3.2.7.	Flex.....	42
3.2.8.	Flowplayer	42
3.3.	Esquema de plataforma	43
3.4.	Proceso	45
3.4.1.	Emisor	45
3.4.2.	Receptor.....	46
3.5.	Instalación y configuración de herramientas	47
3.5.1.	Apache	48
3.5.2.	Java.....	49
3.5.3.	Red5.....	50
3.5.4.	Eclipse.....	54
3.5.5.	Red5plugin	54
3.5.6.	MySQL	56
3.5.7.	Flowplayer	57
3.6.	Creando la primera aplicación.....	58
3.6.1.	Nuevo	58
3.6.2.	Tipo de proyecto.....	59
3.6.3.	Configuración de proyecto.....	60
3.6.4.	Nuevo servidor	62

3.6.5.	Tipo de servidor.....	62
3.6.6.	Deploy y scripts de inicio de Red5	63
4.	PUESTA EN MARCHA Y VISIÓN A FUTURO.....	65
4.1.	Ancho de banda.....	65
4.1.1.	Cálculo	66
4.2.	Memoria RAM	70
4.3.	Procesador.....	71
4.4.	Disco duro.....	71
4.5.	Latencia	71
4.6.	Elección del servidor.....	72
4.6.1.	Disco duro.....	72
4.6.2.	Memoria RAM.....	73
4.6.3.	Procesador.....	73
4.6.4.	Ancho de banda	73
4.6.5.	Proveedor.....	74
4.7.	Visión a futuro	76
4.7.1.	Creación de usuarios	76
4.7.2.	Forma de cobro.....	76
	CONCLUSIONES.....	81
	RECOMENDACIONES.....	83
	BIBLIOGRAFÍA.....	85

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Modelo de referencia OSI	8
2.	Transmisión <i>unicast</i>	9
3.	Transmisión <i>multicast</i>	10
4.	Transmisión <i>broadcast</i>	12
5.	Transmisión <i>Multicast person to person</i>	13
6.	<i>Streaming</i> a través de Internet	14
7.	Interacción de tecnologías	44
8.	Página inicial de Red5.....	52
9.	Objeto compartido en Red5	53
10.	Paso 1, instalación red5plugin.	55
11.	Paso 2, Instalación red5plugin.	56
12.	Paso 1, nueva aplicación.	59
13.	Paso 2, nueva aplicación.	59
14.	Paso 3, nueva aplicación.	61
15.	Crear servidor Red5.....	62
16.	Elegir tipo de servidor.....	63
17.	<i>Path</i> de deploy y scripts.	64
18.	Características iniciales de servidor.	75

TABLAS

I.	Alcance de dispositivos , Wowza Media Server.....	29
II.	Características, FMS vs Wowza vs Red5	33
III.	Consumo de ancho de banda en streaming de vídeo.....	68
IV.	Consumo de ancho de banda por semana.	70

GLOSARIO

Ancho de banda	Cantidad de información o de datos que se puede enviar a través de una conexión de red en un período dado.
Aplicación	Programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajo.
Asistente	Software que guía paso a paso durante la ejecución de un procedimiento, generalmente instalaciones.
Bit	<i>Binarydigit</i> (dígito binario). Un bit es un dígito del sistema de numeración binario.
Bitácora	Registro de sucesos.
Byte	Considerado como una secuencia de bits contiguos, equivalente a un octeto de bits.
Cliente	Equipo o proceso que accede a recursos y servicios brindados por otro llamado servidor, generalmente de forma remota.
Códec	Codificador-decodificador.

Codificador	Esquema que regula una serie de transformaciones sobre una señal o información.
Códigoabierto	Término con el que se conoce al software distribuido y desarrollado libremente.
Consola	Método que permite a las personas dar instrucciones a algún programa informático por medio de una línea de texto simple.
Demo	En informática son prototipos, versiones incompletas o de evaluación de un determinado programa informático con el fin de mostrar la idea de funcionamiento y demostrar sus funcionalidades.
Emisor	En la comunicación, es el actor que emite un mensaje y lo envía a través de un canal de comunicación.
Formato	Es la manera en que se almacena y cataloga la información dentro de un sistema informático, cada aplicación utiliza formatos específicos.
Hertz	El hercio, hertzio o hertz (símbolo Hz), es la unidad de frecuencia del Sistema Internacional de Unidades.
Instalador	Programa ejecutable cuya función es realizar todo lo necesario para que un programa funcione en una computadora.

Interfaz	Conexión entre dos ordenadores o máquinas de cualquier tipo dando una comunicación entre distintos niveles.
IP	Una dirección IP es una etiqueta numérica que identifica, de manera lógica y jerárquica, a un interfaz de un dispositivo (habitualmente una computadora) dentro de una red que utilice el protocolo IP.
ISP	Un proveedor de servicios de Internet (o ISP, por la sigla en inglés de InternetServiceProvider) es una empresa que brinda conexión a Internet a sus clientes.
Latencia	En redes informáticas de datos es la suma de retardos temporales dentro de una red.
MAC	Control de acceso al medio, es una dirección única que identifica a un dispositivo físico.
Medio	Es una vía, o canal a través del cual se pretende lograr comunicación.
Megabyte	Mega equivale a 1 millón, en informática Megabyte aproximadamente equivale a 1 millón de bytes.
Multiplataforma	Multiplataforma es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas, por ejemplo, Windows, Linux, Mac OS, etcétera.

NAT	Mecanismo utilizado por enrutadores IP para intercambiar paquetes entre dos redes que se asignan mutuamente direcciones incompatibles.
Navegador	Navegador web (del inglés, web browser) es una aplicación que opera a través de Internet, interpretando la información de archivos y sitios web para que podamos ser capaces de leerla.
<i>Open source</i>	Equivalente a código abierto.
OSI	Es un marco de referencia para la definición de arquitecturas de interconexión de sistemas de comunicaciones.
Receptor	En la comunicación, es el actor que recibe un mensaje enviado por un receptor, a través de un canal de comunicación.
Reproductor	Es un programa informático o un dispositivo capaz de mostrar un abanico de contenidos audiovisuales. Por norma general, esto incluye la reproducción de sonido, vídeo e imágenes.
<i>Router</i>	También conocido como encaminador, enrutador, direccionador o ruteador. Es un dispositivo de hardware usado para la interconexión de redes informáticas que permite asegurar el direccionamiento de paquetes de datos entre ellas o determinar la mejor ruta que deben tomar.

RTSP	El protocolo de flujo de datos en tiempo real (del inglés Real Time Streaming Protocol) establece y controla uno o muchos flujos sincronizados de datos, ya sean de audio o de video.
Servidor	Es una computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes.
Stream	Contenido enviado/recibido por medio de <i>streaming</i> .
Streaming	Distribución de multimedia a través de una red de computadoras de manera que el usuario consume el producto al mismo tiempo que se descarga.
TCP	Es uno de los protocolos fundamentales en Internet, significa Transfer Control Protocol. Está orientado a la conexión, por lo que la transmisión por medio de él es bastante confiable.
Telégrafo	Dispositivo de telecomunicación destinado a la transmisión de señales a distancia.
UDP	Protocolo del nivel de transporte basado en el intercambio de datagramas, se caracteriza por su velocidad, a diferencia de TCP, no está orientado a la conexión, por lo que no se considera confiable.

UTP

Es un cable de par trenzado que no se encuentra blindado y que se utiliza principalmente, para comunicaciones.

RESUMEN

En un inicio, las redes de computadoras eran conformadas por cantidades muy pequeñas de las mismas, y eran pocas las empresas que podían tener más de una. Con el paso del tiempo esto ha ido cambiando, a tal punto que la mayoría de empresas actualmente poseen más de una computadora, y por regla general deberían de tenerlas distribuidas en una red para que interactúen entre ellas.

El transcurrir de los años dio origen a que las computadoras no se comunicaran únicamente dentro de la red privada de una empresa, actualmente con la ayuda de Internet, es posible interactuar con computadoras de cualquier parte del mundo. Este avance tecnológico ha permitido diversidad de nuevas funcionalidades y necesidades; la transmisión de datos ha ido evolucionando a pasos gigantes, si bien antes era un gran avance la posibilidad de enviar una carta a través Internet, hoy es algo totalmente habitual.

Esta época se caracteriza por el auge que están teniendo las redes sociales, así como la transmisión de multimedia a través de la red, para ésta última se puede mencionar como claro ejemplo: youtube que se ha convertido en uno de los sitios más visitados por el público en general. Youtube generó una revolución en la transmisión de medios, desde su aparición muchos han intentado replicarlo.

En la actualidad existen varios servidores que permiten crear plataformas de transferencia de multimedia a través de la red, sin embargo, el más llamativo por su potencia y costo es Red5, que es un servidor de medios de código abierto, el cual es completamente gratuito y ofrece características que se pueden encontrar en otros servidores de medios, pero cuyas licencias son de altos costos.

OBJETIVOS

General

Desarrollar una plataforma para distribución de audio y vídeo de alta calidad a través de Internet, en donde cualquier persona pueda compartir un evento especial, o simple y sencillamente lo que su cámara web esté enfocando en ese momento y enviarlo en cuestión de décimas de segundo a una o varias personas, sin importar en dónde se encuentren, siempre y cuando tengan a su alcance una computadora con conexión a Internet y un navegador que soporte contenido *flash*.

Específicos

1. Transmitir contenido multimedia en tiempo real, es decir, que la recepción y reproducción del contenido se realice al mismo tiempo que está siendo generado del lado del emisor.
2. Enviar vídeo de alta calidad optimizando el ancho de banda requerido por cada receptor.
3. Lograr la mejor sincronización de vídeo con audio, para crear así una reproducción agradable para el receptor.
4. Transmitir a través de múltiples canales paralelamente, esto para crear eventos totalmente independientes.

5. Evitar la compra de licencias de software y demostrar que a partir de tecnologías libres se pueden lograr grandes resultados.

INTRODUCCIÓN

Cuando se habla de telecomunicaciones, se refiere al transporte de información a grandes distancias a través de algún medio o canal de distribución utilizando señales de cualquier tipo.

Con el paso del tiempo el mundo de las telecomunicaciones ha ido evolucionando de manera sorprendente, muchas veces no se observa esta evolución, pero se puede ver el gran avance si se menciona la primera forma de comunicación a distancia: el telégrafo. Avanzando un poco más en el tiempo se encuentran los primeros pasos de la telefonía, que es considerada la mayor contribución al mundo de las comunicaciones. Con el pasar de los años se fueron alcanzando más logros, dando lugar a la aparición de la radio difusión, pasando por la tecnología de la televisión. En la década de los 90 se introdujo la world wide web, cuya infraestructura se expandió en todo el mundo para crear la moderna red de redes que hoy se conoce, el Internet, su utilización ha extendido inmensamente el mundo de las telecomunicaciones, permitiendo de manera rápida y sencilla comunicarse con personas en cualquier parte del mundo.

Desde la aparición de Internet han surgido diferentes formas de comunicación directa o indirecta entre personas, se puede mencionar como comunicación indirecta los sitios web donde se ofrece contenido para el mundo, siendo cada individuo quien decida verlo o no; entre los canales de comunicación directa se tiene el correo electrónico, el cual se dirige a contactos específicos; la mensajería instantánea, la cual ha tenido un gran uso entre los usuarios de Internet por la velocidad de respuesta que puede obtenerse.

Como evolución de la mensajería instantánea surgieron las videoconferencias que consisten en la comunicación directa entre dos personas a través de una computadora con cámara y micrófono, en donde cada una puede ver a la otra en vivo. Más adelante surge la difusión de contenido multimedia a través de la red, más conocida en inglés como *streaming*. Actualmente existen dos tipos de difusión de contenido multimedia a través de Internet, el primero es por demanda, que quiere decir que el contenido se encuentra en Internet y el usuario puede acceder a él y visualizarlo en el momento que lo requiera, como ejemplo de este se hará mención al sitio que revolucionó el contenido de vídeo en Internet, youtube; el segundo tipo es en vivo o en tiempo real, que quiere decir que el contenido es transmitido y visualizado por el o los receptores en el momento en el que está siendo emitido, es este último en el que se enfocará este trabajo, en el cual se investiga sobre la mejor manera de difundir contenido multimedia a partir de tecnologías libres para crear una plataforma capaz de transmitir el contenido de forma óptima y pueda ser visualizado por personas en cualquier parte del mundo.

1. DIFUSIÓN DE CONTENIDO MULTIMEDIA, INICIOS Y EVOLUCIÓN

La difusión de contenido multimedia ha evolucionado a pasos agigantados en los últimos años, sobre todo por el incremento tanto en uso como en capacidad y ancho de banda del Internet. Ante esto es importante conocer qué es multimedia.

1.1. Multimedia

Como su nombre lo indica, multimedia se refiere al uso o combinación de diferentes medios. En informática se conoce la multimedia como la combinación de varios canales de comunicación, entre los cuales se pueden mencionar:

- Audio
- Vídeo
- Texto
- Animaciones
- Cualquier forma de interacción con el usuario

Como multimedia se pueden mencionar varios ejemplos, ya sea que utilicen únicamente un medio, o bien que mezclen varios; a continuación se listan algunos:

- Una pista de audio: puede ser desde un simple MIDI, hasta una mezcla de sonidos; ésta utiliza únicamente un medio: el audio.
- Una película en español: ésta involucra dos medios para la interacción con las personas, audio para voces y efectos de sonido, e imágenes en forma de vídeo para mostrar las diferentes escenas.
- Una película en inglés, subtitulada al español: en este caso se hace uso de tres medios para interactuar con las personas, al igual que el caso anterior, se hace uso de audio y vídeo, pero a diferencia de aquél, éste utiliza un tercer medio, haciendo referencia al texto, el cual está representado por los subtítulos.

1.2. Primeros pasos

Los primeros pasos de la multimedia se dieron hace bastantes años, se pueden mencionar algunos dispositivos como ejemplo de una multimedia arcaica, si se compara con los avances que se tienen a la fecha, pero que representan dispositivos no tan antiguos y que aún están presentes en la memoria de muchos en la actualidad, entre ellos están:

- Casetes de audio
- Vídeo casetes
- Discos longplay

Los anteriores, efectivamente forman parte de la historia del origen de la multimedia, pero los inicios de ésta se remontan a mucho tiempo atrás, podría decirse que la primera muestra de transmisión de datos a través de un medio no manual fue el telégrafo, éste requería de un intermediario que

interpretara las señales eléctricas, para enviar un mensaje. El telégrafo no es considerado como una muestra de transmisión de contenido multimedia debido a que no envía texto puro, por lo tanto, se puede decir que la primera muestra real de transmisión de multimedia se hace con la invención de Graham Bell: el teléfono; éste sin lugar a dudas, es una muestra de transmisión de datos a través de un medio sin necesidad de la intervención de ninguna clase de intermediarios. El teléfono es una de las más grandes invenciones en la historia, que además no ha logrado ser reemplazada, si bien ha evolucionado, no ha podido dejarse atrás, es más, ha abierto campo a nuevas tecnologías.

Con el paso del tiempo nuevas tecnologías fueron apareciendo; como la invención de Marconi: la radiotelegrafía, ésta consistió en ampliar la capacidad de un telégrafo, dejando de utilizar las líneas conductoras de electricidad y además ampliando la forma de comunicación, ya que en lugar de que ésta fuera de un emisor hacia un único punto receptor, esta señal podía ser enviada a varios receptores simultáneamente. Al igual que se mencionó anteriormente, el telégrafo podría generar alguna duda en cuanto a la transmisión de multimedia, la mención de la radiotelegrafía es para dar paso a otro de los inventos más revolucionarios en la historia de la transmisión de audio, es decir: la radio. Ésta surgió cuando las señales de voz provenientes de un micrófono (del teléfono) fueron aplicadas a las portadoras de radio, así nació un nuevo medio de comunicación entre humanos, en el que no era necesaria la intervención de ningún mediador y que, además tenía la capacidad de llegar a un gran número de receptores simultáneamente.

Tiempo después aparecieron nuevos inventos, los cuales permitían grabar la voz, y posteriormente grabar música, este invento se denominó: gramófono.

Más adelante surgió el invento que revolucionaría la transmisión de contenido audio-visual a través de un medio de comunicación, siendo éste: la televisión. Ésta se inició a blanco y negro, y con el paso del tiempo apareció el color. La televisión se ha vuelto un elemento cotidiano en la vida del ser humano, ya sea para ver un noticiero, una película, o bien ver eventos espectaculares que quizás están ocurriendo al otro lado del mundo.

Avanzando más en la línea del tiempo se llega a la era de la digitalización, la cual permitió la integración de aplicaciones aparentemente diferentes, lo cual abrió campo y lo que actualmente se conoce como multimedia.

1.3. Evolución

Con la aparición de Internet, hace ya varios años se ha ido convirtiendo no en una capacidad más, sino en una necesidad la transmisión de multimedia a través de ella, en Internet se pueden encontrar varios ejemplos de transmisión de multimedia, por ejemplo: canciones, vídeos, radio en línea, canales de televisión, etc.

La era digital, los medios y las nuevas características que se incorporan a las computadoras, como también a los dispositivos domésticos, han creado una diversidad de cambios drásticos, se ha ido evolucionando la forma en que las noticias, la información y el entretenimiento se distribuyen, como también la experiencia que éstos proveen. La revolución digital está abarcando todo, tanto en el hogar como en la oficina, en un estudio o en un dormitorio, o incluso en la carretera, no importa dónde, la promesa actual es: donde sea y cuando sea, la información y entretenimiento están donde haya demanda.

Todo empezó con la computadora y el *streaming* de multimedia, a través de Internet o de una red corporativa. Esto creó por primera vez una interactividad verdadera, audio y vídeo por demanda. Con el pasar del tiempo, las tecnologías de *streaming* fueron evolucionando, mucha gente empezó a ampliar su uso con contenido multimedia, lo que generó un impacto dramático en consumidores y trabajadores, como también en las industrias de tecnologías de la información.

La aparición de las conexiones Digital Subscriber Line (DSL) ha revolucionado la forma en que se distribuyen archivos de multimedia. Con el surgimiento del Internet de banda ancha, el *streaming* se ha vuelto el aliado ideal para compañías de infraestructura, desde empresas de telefonía, compañías proveedoras de televisión por cable, hasta proveedores de servicio de Internet (ISPs), esto con el fin de fortalecerse y así proveer enlaces de alta velocidad para hogar y oficina. Si se combina esto con interactividad y el comercio electrónico en Internet, se puede obtener algo irresistible, y es el cobro por uso, algo muy tentador y muy rentable si se logra el objetivo ideal, como también la inclusión y soporte para contenido publicitario.

1.4. Modelo OSI y la transmisión de datos

Éste se encarga de subdividir los sistemas de comunicaciones en partes más pequeñas, a las que se les denomina capas. Las funciones que son similares, se agrupan en capas lógicas; cada una provee servicios hacia la capa superior, mientras recibe servicios de la capa de un nivel inferior. El modelo OSI, dicho en otras palabras, es un estándar para la definición de arquitecturas de interconexión de sistemas de comunicaciones; describe desde el inicio de transmisión de datos (*bits*), hasta llegar a las aplicaciones que se utilizan cotidianamente (correo, navegador de Internet, etc.). El modelo OSI consta de

7 capas, las cuales se describirán a continuación y que ayudará a comprender de mejor forma cómo se transmite la información a través de una red, o bien, a través de la red de redes: Internet.

1.4.1. Capa física

Ésta define todo lo que se refiera a componentes eléctricos y físicos. En particular define la relación entre un dispositivo y el medio de transmisión, por ejemplo: un cable UTP, transmisión inalámbrica, RS-232, etc. La unidad de datos que se maneja en ésta capa es el *bit*.

1.4.2. Capa de enlace de datos

Ésta es la encargada del enrutamiento a nivel físico, a nivel de topología de red. Para ello hace uso de la dirección MAC, la cual provee acceso al medio; ésta dirección es única en cada dispositivo de red, ninguna se repite. La tarea principal de ésta capa es tomar la información tal cual la envía la capa física y transformarla en una abstracción que no contenga errores, para luego enviarla a la capa de red; cada una de éstas abstracciones de datos se denomina trama, nombre que recibe la unidad de datos de ésta capa.

1.4.3. Capa de red

Es la encargada del enrutamiento entre una o varias redes. En ésta capa la unidad de datos que se maneja es denominada: paquete. La capa de red realiza las funciones de enrutamiento, así también, unifica los fragmentos, y reporta errores de envío. Los enrutadores operan en ésta capa, enviando información a través de la red, lo cual hace posible la Internet.

Es decir, el objetivo de la capa 3 es hacer que los paquetes lleguen desde el origen al destino, incluso si éstos no están conectados directamente.

1.4.4. Capa de transporte

Ésta provee una transferencia de datos transparente entre usuarios y de servicios de transferencia confiables hacia las capas superiores. A la unidad de datos de la capa 4 se le denomina segmento o datagrama, esto dependiendo si corresponde al protocolo TCP o a UDP. El protocolo TCP está orientado a la conexión, mientras que UDP no lo está. Ambos trabajan con puertos lógicos para la comunicación, y al unir la capa de transporte con la capa de red se obtienen los *sockets*, que no es más que la unión de una dirección IP y un puerto.

1.4.5. Capa de sesión

Esta capa se encarga de controlar las conexiones entre computadoras. Establece, maneja y termina las conexiones entre la aplicación local y remota. Permite cifrar los datos y también comprimirlos, por lo tanto, puede decirse que actúa también, como un traductor.

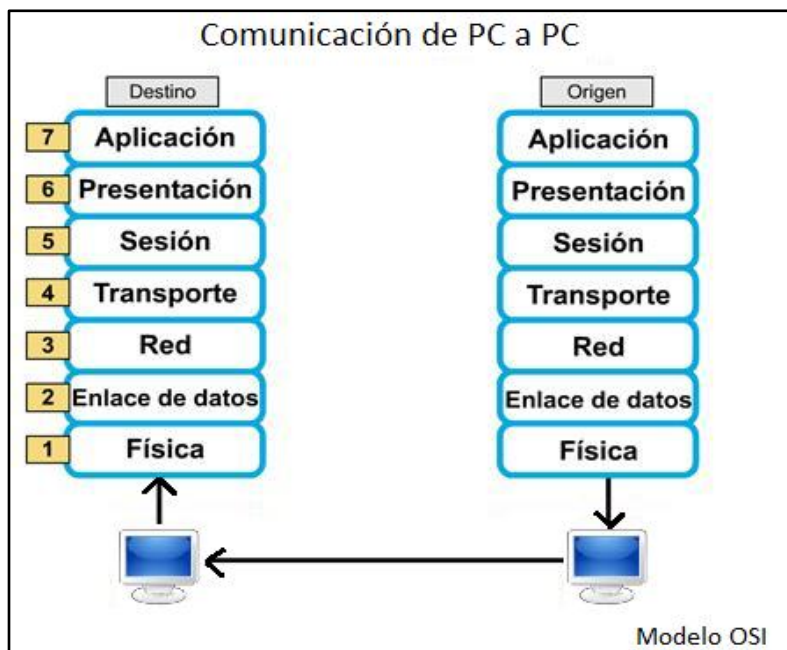
1.4.6. Capa de presentación

Esta capa provee independencia de representación de datos a través de la traducción entre formatos de aplicación y de red. Esta capa transforma datos de manera que la aplicación los acepte. Es la primera capa que se acerca más al contenido de la comunicación en lugar de ver cómo se establece ésta.

1.4.7. Capa de aplicación

Ésta capa es la más cercana al usuario final, lo que significa que la capa de aplicación y el usuario interactúan directamente con las aplicaciones de software que utilizan algún componente para comunicación. Por ejemplo: correo electrónico, servidores de ficheros, gestores de bases de datos, entre otros.

Figura 1. **Modelo de referencia OSI**



Fuente: elaboración propia.

1.5. Transmisión de multimedia a través de Internet

La transmisión de contenido multimedia a través de Internet se ha ido volviendo muy popular en los últimos años. Ésta es más conocida como *streaming*. Youtube es un sitio de *streaming* de vídeo por demanda, es decir,

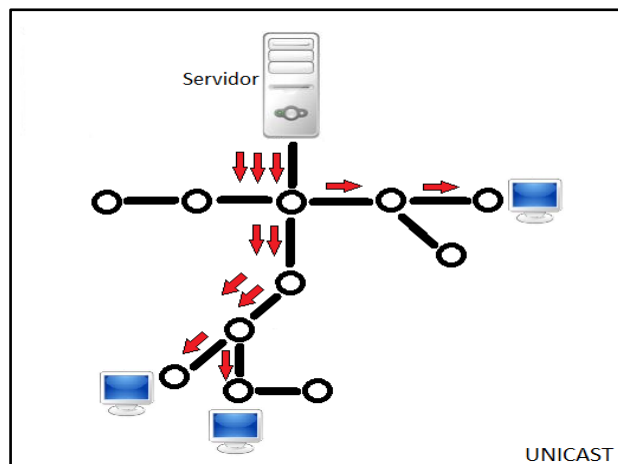
éste está disponible para ser visto en cualquier momento, ya que está colgado en un servidor, el cual mandará los datos cuando estos sean solicitados.

Existen varias formas de transmitir datos de una máquina origen hacia un destino a través de la red, a continuación se describen las más importantes:

1.5.1. Unidifusión

También conocida como *unicast*, es una forma de transmisión en la que por cada usuario conectado al servidor, se establece un canal de comunicación independiente, es decir, si hay 100 usuarios conectados al servidor, se crearán 100 canales independientes, uno para cada usuario conectado, no importando que el contenido que se envía por cada uno de los canales sea el mismo. Por tanto, la unidifusión lo que hace es realizar una conexión punto a punto desde una única máquina origen, hacia cada una de las máquinas destino con las cuales se haya establecido una conexión previa.

Figura 2. Transmisión *unicast*

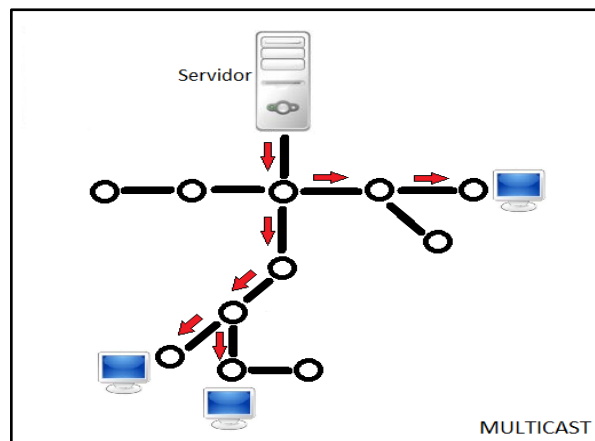


Fuente: elaboración propia.

1.5.2. Multidifusión

Conocida también por su traducción al inglés *multicast*. Es una forma de transmisión de datos en la que el servidor únicamente envía por un canal de comunicación, no importando el número de potenciales máquinas receptoras que haya. La máquina origen envía un *stream* de datos (datagramas) a todas las máquinas destino que posean al menos un miembro del grupo de multidifusión y que, además compartan la misma dirección *multicast*. Estas máquinas pueden estar dispersas geográficamente en múltiples redes en Internet. No importando el número de destinatarios que posea el *stream*, el servidor enviará únicamente una vez la información, para ello los *routers* intermediarios de multidifusión en Internet tienen que tener la capacidad de realizar las copias necesarias de la información transmitida, para así enviarla hacia las máquinas que la solicitan. Como se mencionaba anteriormente, los encargados de realizar la tarea de multidifusión son los *routers*, por tanto, se puede concluir que el proceso de multidifusión pertenece a la capa 3 del modelo OSI.

Figura 3. Transmisión *multicast*



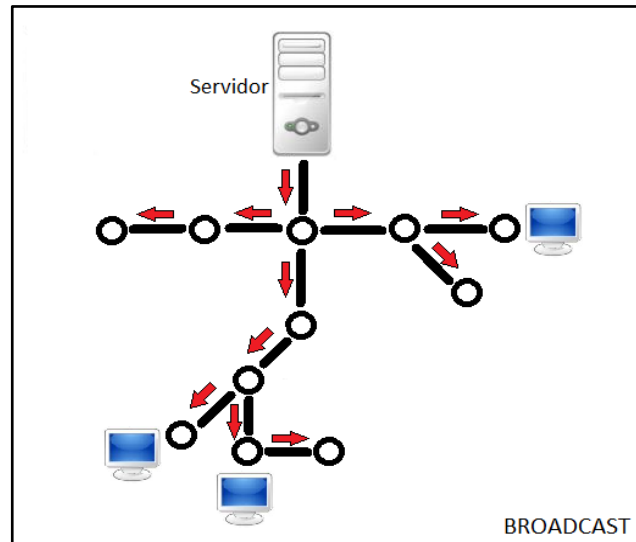
Fuente: elaboración propia.

1.5.3. Difusión

Conocida también por su traducción al inglés como *broadcast*. Es un proceso que envía datos simultáneamente a un grupo de personas o compañías en un área geográfica específica, o a cualquiera que pueda conectarse o recibir señales de un sistema de red de difusión, por ejemplo, televisión satelital o por cable. Generalmente se asocia el *broadcasting* con los sistemas de transmisión de radio o televisión, los cuales envían la misma señal a muchos aparatos receptores en un área geográfica.

La difusión o *broadcast*, también se aplica a los sistemas de distribución donde todos los usuarios que están conectados a la red pueden recibir y reenviar la misma señal de información. Cuando un paquete es recibido de una fuente, éste puede ser copiado y distribuido a todas las conexiones que requieran de la información. La característica principal de los sistemas de difusión es que a éstos no les importa quién está interesado en recibir la información y quién no, la información se envía a todo aquél que esté conectado a la red donde se originó el paquete.

Figura 4. Transmisión *broadcast*

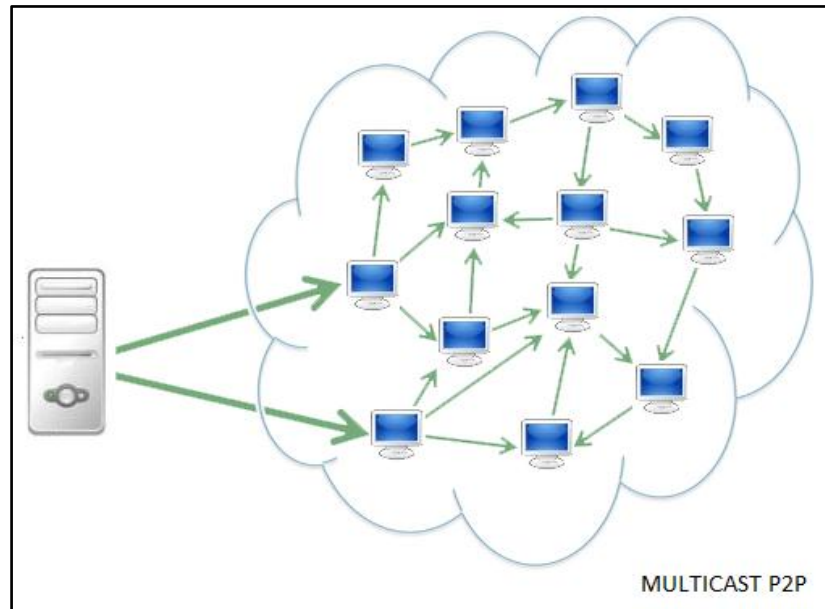


Fuente: elaboración propia.

1.5.4. Multidifusión P2P

Conocida también en inglés como Peer to Peer Multicast. Esta técnica para multidifusión de vídeo es relativamente nueva y no es muy conocida; el propósito principal de ésta es la escalabilidad, es decir, que no importando que se conecten miles de usuarios, éste pueda escalar fácilmente. La forma en que funciona esta técnica es que cuando un nuevo usuario quiera conectarse, debe ser capaz de enviar un mensaje a un nodo al azar, con la condición que éste tiene que ser parte actual del sistema. Cada usuario es responsable de proveer información a un grupo moderado de clientes, esto limitado al ancho de banda que posea.

Figura 5. **Transmisión *Multicast person to person***



Fuente: elaboración propia.

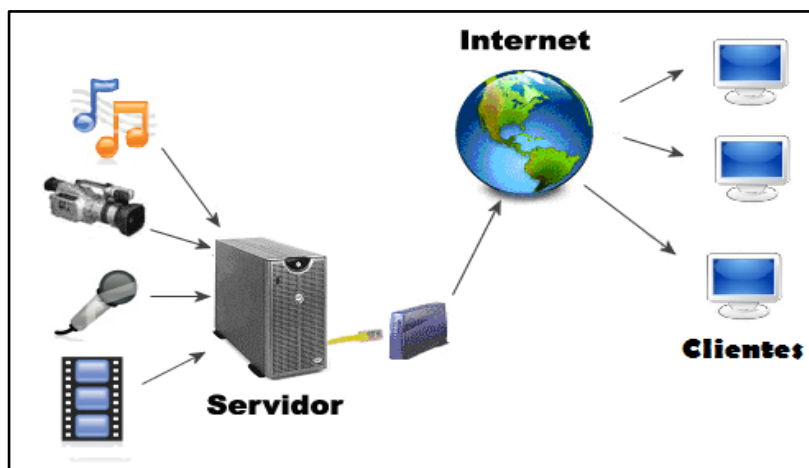
Para definir qué es el *streaming* se mencionó el famoso sitio de vídeos Youtube, si bien es cierto que se podría sacar una conclusión de lo que significa *streaming* a partir de este ejemplo, también es cierto que se podría tener un concepto erróneo, por lo que se definirá lo que significa y los usos que se pueden dar al *streaming* a través de la red.

1.6. **Streaming**

Se conoce como *streaming* a la transmisión continua e ininterrumpida de audio y/o vídeo a través de Internet, donde, ésta se transmite desde una máquina que funge como servidor, y hay una o varias máquinas que actúan como clientes; es decir, José por mencionar algún nombre, puede ser el servidor y realizar *streaming* de vídeo a través de Internet, mientras que en

cualquier lugar del mundo con acceso a Internet pueden estar uno o varios amigos de José recibiendo el *stream* de vídeo que les está enviando desde su computadora. Se puede observar que se mencionan dos conceptos nuevos, *streaming* y *stream*; se puede definir como *stream* a la información que se envía del origen a uno o varios destinos; mientras que *streaming* sería la acción de transmitir esa información.

Figura 6. **Streaming a través de Internet**



Fuente: elaboración propia.

El *streaming* nació en 1995 con el lanzamiento de real audio, una aplicación para transmisión de audio en tiempo real. Antes de que apareciera el *streaming*, si alguien quería escuchar una canción o ver un vídeo, era estrictamente necesario descargar un archivo completo para luego poder reproducirlo en la computadora, mientras que con la aparición del *streaming* todo esto quedó atrás, ahora es posible reproducir contenido multimedia desde Internet, sin necesidad de que éste descargue por completo, es decir, se puede reproducir el *stream* conforme éste está siendo transmitido hacia la computadora.

Una analogía que define de una forma fácil el concepto de *streaming*, y que vale la pena mencionar es la siguiente: se tiene una botella con leche, un vaso y muchas ganas de beber la leche.

- Sin *streaming* (descarga): se debe tomar la botella de leche y servirla en el vaso hasta que éste se llene; una vez lleno puede empezar a consumirse.
- Con *streaming*: se toma la botella de leche, pero en lugar de llenar el vaso completamente, éste se hace a un lado y se consume la leche directamente desde la caja.

1.6.1. Tipos de *streaming*

Existen dos tipos de *streaming*, por demanda y en directo, a continuación se describe cada uno de ellos y cómo funciona:

1.6.1.1. Por demanda

Éste se refiere a la transmisión de contenido multimedia almacenado en un servidor y que puede ser visualizada en cualquier momento. El mejor ejemplo de este tipo es de nuevo Youtube, ya que en él se pueden encontrar millones de vídeos, que han sido subidos por gente de todo el mundo, los cuales se pueden visualizar cuando es requerido sin importar la hora.

1.6.1.2. En directo

A diferencia del *streaming* por demanda que puede ser visto en cualquier momento, el *streaming* en directo puede ser consumido únicamente

en el momento en que se está realizando la transmisión; viene siendo similar a la televisión convencional, donde un televisor sintoniza señales que están siendo transmitidas en tiempo real, no importando si son pregrabados o no, la señal está siendo transmitida en ese momento, de tal forma que al encender el televisor será sintonizado lo que la televisora esté enviando en ese preciso instante.

1.6.2. Elementos fundamentales para realizar *streaming*

Para que todo esto funcione son fundamentales tres partes, comenzando desde dónde se transmite la información, es decir, el servidor; a quién se le enviará la información, es decir, los clientes; y cómo transmitir la información. A continuación se describe brevemente cada una de estas tres partes:

1.6.2.1. Servidor

Este es el encargado de distribuir la información a partir de una fuente, a través de él se generan datagramas que se envían a través de la red. Al hablar de servidor se hace referencia tanto a hardware, como al posible software que éste incluya.

1.6.2.2. Cliente

Es cada una de las máquinas receptoras de la información que se transmite, como mínimo se necesita tener un cliente, sino no tendría sentido la transmisión.

1.6.2.3. Medio

Es a través de dónde se enviará la información, grosso modo puede decirse que es la Internet, pero si se profundiza un poco más y se llega al nivel en donde esto se realiza, se estaría hablando de la capa 4 del modelo OSI, es decir, la capa de transporte; para realizar *streaming* se utilizan datagramas (*UDP*), que son paquetes que se envían sin esperar confirmación de entrega al destinatario, esto permite que la transmisión sea más rápida y fluida.

1.6.3. Enfoque de servidor

No es estrictamente necesario contar con un servidor especial para colocar contenido multimedia, ya que un servidor normal puede enviar información, siendo el cliente el que se encargue de procesarla y mostrarla a medida que está siendo recibida.

Sin embargo, para montar un verdadero servidor de *streaming* es ideal el uso de herramientas de software que provean de funcionalidades y optimizaciones para la realización del objetivo, pudiendo con ellas modificar la calidad del archivo que se envía, para así optimizar el ancho de banda del que se dispone. En el caso del *streaming* en directo, es imprescindible el uso de éstas herramientas de software, la cual al momento de recibir la información del emisor, la enviará a través del servidor hacia los clientes que estén sintonizando en ese momento. En este capítulo se mencionarán las herramientas más conocidas en el medio, sin entrar en mayor detalle sobre ellas, ya que serán profundizadas en el siguiente capítulo.

1.6.3.1. Flash media server

Es un sistema propiedad de Adobe Systems. Este servidor funciona como un distribuidor, se basa en aplicaciones *flash* para conectar los clientes al distribuidor mediante el protocolo de envío de mensajes en tiempo real, RTMP. Funciona en sistemas Windows y Linux.

1.6.3.2. Wowza media server

Este sistema fue desarrollado por Wowza Media Systems. Es utilizado para realizar *streaming* de audio o vídeo, por demanda o en directo. Puede ser utilizado sobre varios sistemas operativos, ya que es una aplicación *java*.

1.6.3.3. Red5 media server

Es un sistema para la distribución de *streaming* que provee funcionalidad de multi-usuario basado en tecnología de *Adobe Flash Player*. Está basado en *Java*, *red5* puede ser utilizado en empresas de cualquier ámbito, sin importar su tamaño. La principal característica de éste es que es código abierto, por esto y por el poder que brinda, será el servidor que se utilizará para el desarrollo de la plataforma que se describirá a partir del capítulo 3.

1.6.4. Enfoque de transporte

Para realizar *streaming* a través de Internet es fundamental la velocidad de transmisión, ya que es necesario que la tasa de llegada de datos sea superior a la tasa de visualización que mantiene el cliente.

Es por ello que para este tipo de transmisiones se utilizan protocolos ligeros, siendo éstos: UDP y RTSP, los cuales se desarrollan en la capa 3 del modelo OSI.

1.6.4.1. User datagram protocol (UDP)

En inglés, User Datagram Protocol, es uno de los miembros núcleo de los protocolos de Internet. Con UDP las aplicaciones de computadora pueden enviar mensajes a través de Internet, sin necesidad de establecer canales especiales de transmisión o rutas de envío. Este protocolo, a diferencia del conocido TCP, no está orientado a la conexión, por lo que, a éste no le interesa saber si los datagramas (paquetes) han sido recibidos por los clientes de forma correcta o no. De ésta forma se logra una transmisión fluida de información a través de Internet, ya que no se pierde tiempo verificando si la información está llegando de forma correcta. Lograr una transmisión fluida ayuda a que el *streaming* se produzca de forma continua y sin interrupciones, algo que es indispensable.

1.6.4.2. Real time streaming protocol (RTSP)

Por sus siglas en inglés, Real Time Streaming Protocol. Es un protocolo no orientado a la conexión, el servidor mantiene sesiones asociadas a un cliente por medio de un identificador. RTSP es un protocolo que puede usar tanto UDP como TCP, dependiendo de la situación; utiliza TCP para los datos de control del reproductor, y utiliza UDP para la transmisión de audio y vídeo, que como se explicó en UDP, esto contribuye a una transmisión más fluida. Este protocolo es el que utiliza Youtube.

1.6.5. Enfoque de cliente

No serviría de mucho tener un servidor funcionando en perfectas condiciones, ni tampoco la optimización en el transporte de la información, si del lado del cliente no se tienen correctamente configuradas las herramientas necesarias para una reproducción óptima del *stream* de datos que se envía, es por ello que se deben de tener claros algunos conceptos que serán manejados del lado del cliente.

1.6.5.1. Precarga de datos

Cuando se transmite la información desde el servidor hacia los clientes, ésta está sujeta a demoras o retrasos, esto ocurre cuando los datos se escasean, ya sea por interrupciones en la conexión, o bien por el poco ancho de banda de cualquiera de las partes. Es por ello que los reproductores multimedia utilizan los conocidos *buffer*, un *buffer* funciona como una memoria, va almacenando los datos que va recibiendo para así disponer de una reserva con el objetivo de evitar que el *stream* se congele. Como se mencionaba antes, esto ocurre en los reproductores multimedia, es decir, del lado del cliente.

1.6.5.2. Codecs

Éstos funcionan como programas residentes en memoria de la computadora del cliente, están encargados de descifrar o interpretar la información que llega desde el servidor, ésta es descifrada basada en el formato de vídeo que se utilizó para cifrarla, una vez hecho esto se envía al reproductor para que éste lo reproduzca al usuario final.

Hay dos tipos de *codecs*, éstos se dividen según la forma en que realizan su función.

- *Codecs* transformadores: éstos basan su funcionamiento en el ordenamiento eficiente de la información, para así poder ahorrar espacio.
- *Codecs* predictores: éstos operan con base en la eliminación de información innecesaria para la codificación.

1.6.5.3. Reproductor

Éste es la interfaz que está entre el usuario y el contenido o *stream* que se recibe, toma como soporte los *codecs* que hayan instalados en la computadora para así descifrar los datos y mostrarlos como una imagen o bien como audio, dependiendo de lo que se esté transmitiendo.

Existen reproductores para la computadora, como también reproductores para sitios web, en este caso se utilizará *FlowPlayer*, que es un reproductor web, a base de *javascript*, a este tipo de reproductores basta con decirles qué tipo de información deben leer y cómo mostrarla.

2. HERRAMIENTAS PARA DIFUNDIR CONTENIDO MULTIMEDIA

Cuando surgió la idea de desarrollar una plataforma para realizar *streaming* de vídeo, se comenzó a elaborar una investigación, la primera herramienta que se cruzó por el camino fue IceCast, un servidor de medios código abierto multiplataforma.

Utilizando el reproductor de multimedia VLC se envió audio hacia el servidor IceCast, el cual enviaba exitosamente el *stream* de audio para ser escuchado en cualquier computadora que contara con un reproductor como VLC y que tuviera alcance a la red que estaba siendo utilizada. Si bien las pruebas fueron satisfactorias, no fueron lo que se esperaba, ya que se producía un retardo bastante alto, siendo el contenido de únicamente audio, fue por ello que se continuó la búsqueda de otras alternativas, sin saber que se encontraría la herramienta perfecta para los planes que se tenía en mente.

Fue así como se encontró Flash media server, el cual se veía como la herramienta perfecta para realizar *streaming* de vídeo, por demanda y en tiempo real; todo iba marchando muy bien, hasta que se llegó a la parte donde fue referido el precio de ésta herramienta, la cual tenía un precio aproximado de Q. 36 000,00 (US\$. 4 500,00), un precio no esperado en el plan original.

Al ver un tanto frustrado el plan con que se contaba, se pensó que generalmente existen alternativas libres, por lo cual se inició nuevamente la búsqueda y fue entonces cuando se encontró Red5, una alternativa libre a Flash Media Server y que será la herramienta a tratar en este documento.

Ahora bien, no por ser libre esta herramienta tiene que ser pobre o mala en cuanto a su funcionamiento, al contrario, es una herramienta bastante poderosa; es por ello que a continuación se explicará un poco más sobre qué hace cada una de ellas y qué funcionalidades posee una que no posean otras, para tener una mejor comparación.

2.1. Icecast

Es una herramienta para realizar *streaming* de medios, se puede utilizar para crear estaciones de radio por Internet o para el uso que se quiera dar, siempre basado en *streaming*. Es bastante versátil, ya que se pueden agregar nuevos formatos fácilmente, como también soporta estándares abiertos para comunicación e interacción.

Actualmente Icecast soporta *streams* de varios tipos:

- MP3
- Ogg FLAC
- Ogg Theora
- AAC
- Ogg Vorbis
- Ogg Speex

Icecast nació como una herramienta para *streaming* de audio, desde su versión 2.2.0 y es capaz de soportar *streaming* de vídeo.

Aunque, actualmente existe una versión para Microsoft Windows, Icecast nació como una herramienta para GNU/Linux, es por ello que su configuración se realiza desde archivos de texto plano, quizás un poco rústica, pero está bien organizado y no importando la plataforma la configuración es estándar.

Icecast funciona con puntos de montaje, o *mountpoints*, los cuales conforman cada una de las líneas de distribución que éste tiene en un momento específico. Para enviar datos hacia el servidor Icecastse necesita de alguna herramienta que permita enviar un *stream* hacia él, mediante *hypertext transfer protocol*, más conocido como http.

Como se mencionó anteriormente, la configuración es fácil, a través de archivos de texto, pero se debe tener mucho cuidado en algo, y esto es el cambio de la contraseña que tiene por *default*, ya que si no se cambia ésta, cualquier persona que sepa del famoso admin...admin podrá acceder al servidor y modificar lo que desee a su antojo.

2.2. Flash media server

Conocido también por sus siglas FMS, es un servidor de medios, propiedad de Adobe Systems. El objetivo de este servidor es proveer servicio a las aplicaciones basadas en *flash*, para conectarse a él; éste puede enviar y recibir datos desde y hacia los usuarios conectados a él. Los clientes conectados remotamente pueden hacer llamadas a procedimientos remotos del lado del servidor, y el servidor puede llamar métodos específicos de los clientes.

Actualmente Flash media server maneja varios formatos de transmisión, siendo estos:

- FLV
- H.264
- MP3
- AAC
- LC-AAC
- HE-AAC
- Speex

Este servidor puede manejar objetos compartidos, o conocidos dentro de la programación como *SharedObjects*; estos se utilizan para sincronizar estructuras complejas de datos para llamar a un mismo método en múltiples clientes, en todos los que estén suscritos al objeto compartido.

FMS utiliza también, conexiones de red, conocidas como *NetConnections*, las cuales se utilizan para transportar objetos en código *ActionScript*, estos mensajes los maneja transparentemente el cliente y servidor. Además, el Servidor también permite a los usuarios recibir y publicar *Streams*, gracias a los *NetStreams*, éstos permiten a los usuarios conectarse a un *stream* que esté siendo transmitido por el servidor, a medida que el *stream* avance y nuevos usuarios se conecten a él, todos verán lo mismo conforme este avanza.

Flash media server nació el 9 de julio de 2002 con su versión 1.0 actualmente éste se encuentra en su versión 4.0 y como se ha mencionado, su precio aproximado es de Q. 36 000,00 o bien US\$. 4 500,00.

2.3. Wowza media server

Es un servidor de medios, propiedad Wowza *Media Systems*. Al igual que los anteriores es un servidor de medios, con la diferencia que éste está basado en Java, lo cual ofrece una plataforma escalable y altamente eficiente, facilitando enormemente el *streaming* en directo y por demanda en diferentes formatos, por ejemplo: FLV y H.264; además se pueden integrar a las aplicaciones servicios como *Chat* en directo o grabaciones en vídeo.

Wowza puede funcionar también para publicar archivos de música, los cuales podrán escucharse a través de un reproductor *flash* en la página *web*, más no podrán ser descargados.

Wowza aplica todas las variantes del protocolo RTMP:

- RTMP
- RTMPT
- RTMPS
- RTMPE
- RTMPTE

Como también los protocolos Real Time:

- RTP
- RTSP
- MPEG-TS

Los servidores Wowza son muy poderosos como se puede observar, además de manejar muy bien el protocolo RTMP (Real Time Messaging Protocol), cubren una buena cantidad de formatos de salida, los cuales se listan a continuación:

- FLV
- MP4, incluyendo:
 - .mp4
 - .f4v
 - .mov
 - .m4v
 - .mp4a
 - .3gp
 - .3g2
- MP3

Este servidor es realmente poderoso, comparado con el Flash Media Server, Wowza gana y por mucho, no sólo en el precio que actualmente es de aproximadamente Q. 8 000,00 o US\$. 995,00, sino que también en la cantidad de características que éste proporciona. Realmente es sorprendente ver todas las características que provee.

Con la revolución que se está dando actualmente de Computación en la nube, Wowza no quiere quedarse atrás y ha ingresado a este mundo. Actualmente se puede encontrar Wowza Media Server 2 dentro del servicio de nube que ofrece *Amazon*, se está hablando de Elastic Cloud Computing, más conocida como Amazon EC2. Wowza se ofrece como un servicio más dentro

de la amplia gama que ofrece *Amazon EC2*, brindando la oportunidad de utilizarlo sin necesidad de adquirir una licencia, sino que solamente con tener una cuenta de EC2 se puede hacer uso de éste.

Wowza, a diferencia de Flash Media Server y *Red5*, puede realizar *streaming* de video no solo a computadoras que tengan instalado Flash Media Server, actualmente Wowza da la posibilidad de hacer *streaming* de vídeo prácticamente a cualquier dispositivo, esto gracias a que no utiliza únicamente el protocolo RTMP, sino que utiliza un protocolo diferente dependiendo del dispositivo que esté accediendo al *stream*. A continuación se mencionan los dispositivos y de qué forma es enviada la información para su respectiva reproducción:

Tabla I. **Alcance de dispositivos, Wowza media server**

Dispositivo	Vía
PC's	Flash
iPhones	http
Smarphones	RTSP/RTP
Silverlight	RTSP/RTP
Mac	RTSP
IPTV	RTMP, HTTP, HLS

Fuente: elaboración propia.

2.4. Red5

Red 5, al igual que las herramientas anteriores, es un servidor de medios, con la diferencia que éste no es propiedad de ninguna empresa, nace a partir de una comunidad liderada por Dominick Accattato y Chris Allen.

Al ser éste un servidor código abierto, o libre, quiere decir que puede ser descargado y utilizado de la forma que a cada quien le convenga, compilándolo desde cero si así lo desea, o bien puede descargar un instalador y utilizar la versión estable más reciente.

El servidor Red5 es una herramienta potente que permite la distribución de *streaming* de vídeo orientado a aplicaciones multiusuario que utilicen *Flash Player* y también algunas otras tecnologías cliente. Está basado en Java y actualmente Red5 está colocado como una solución bastante buena para empresas de cualquier tamaño.

Red5 incluye soporte para las funcionalidades que ofrece la tecnología flash, como *NetConnection*, *NetStream* y *SharedObjects*, a la vez provee una poderosa implementación de RTMP. Red5 posee un contenedor Tomcat, esto con el fin de poder tener aplicaciones Java EE en él.

Éstos son los formatos que maneja Red5:

- FLV
- MP3
- F4V
- MP4
- AAC

- M4A
- Sorenson
- VP6
- h.264
- Nelly Moser
- MP3
- Speex
- AAC
- NSV

Red5 no se limita únicamente a hacer *streaming* de vídeo, red5 puede ser utilizado para muchas otras cosas, por ejemplo: juegos multiusuario, aplicaciones SIP y aplicaciones de chat.

2.4.1. Juegos multiusuario

Red5 es una herramienta poderosa para realizar este tipo de juegos, ya que provee un API para las conexiones multiusuario, como también se puede utilizar el poder que provee para animaciones la tecnología *Flash*, a través de *Flash Builder* o bien del mismo *Macromedia Flash*. Tal es el caso de un equipo de desarrollo de juegos llamado *Red5 Studios*, el cual es un equipo completo y formal que se dedica al desarrollo de juegos multiusuario, es tan grande el equipo y el éxito que están teniendo, que incluso en su página web tienen puestos vacantes para más miembros para su tribu.

2.4.2. Aplicaciones SIP

Estas aplicaciones se basan en el protocolo SIP, o *Session Initiation Protocol*. Una aplicación de este tipo puede ser cualquiera que involucre

sesiones con un inicio y un fin, y que incluyan contenido multimedia, es decir, puede considerarse un juego, o el *streaming* como tal, pero en este caso lo enfocaremos a un proyecto realmente novedoso llamado *Red5Phone*, este proyecto está orientado a realizar llamadas a través de VoIP, este es un proyecto que se encuentra en constante desarrollo y que al igual que la herramienta padre, éste también es código abierto, lo que significa que se pueden descargar las fuentes y luego utilizarlas según las necesidades.

2.4.3. Aplicaciones de *chat*

Con la implementación de la API para conexiones multiusuario, Red5 brinda la oportunidad de mezclar tecnologías, flash, java y si fuera necesario alguna otra, para realizar aplicaciones para chat de texto con otros usuarios, incluso éstas pueden ser incluidas en cualquier aplicación desarrollada con Red5, se puede tener un *streaming* de vídeo con chat incluido, o bien incluir un Chat en un juego multiusuario, etc.

2.5. Características, ventajas y desventajas

Cada servidor tiene sus ventajas y desventajas respecto a los otros. Desde el punto de vista personal, los 3 son muy potentes, e indiferentemente de que para este documento en particular se vaya a utilizar Red5, según sus características y tendencias a futuro se piensa que el que se lleva el premio al mejor y más completo servidor de medios es Wowza, realmente es sorprendente todas y cada una de las características que éste posee, sobre todo la tendencia a futuro, con la inclusión de éste a la nube y la capacidad de realizar *streaming* a cualquier dispositivo, desde una computadora, hasta un *smartphone*, o *iPhone*.

Tabla II. **Características, FMS vs Wowza vs Red5**

Característica	Flash Media Server	Wowza Media Server 2	Red5
Protocolos soportados	RTMP RTMPT RTMPS RTMPE RTMPTE RTMFP	RTMP RTMPT RTMPS RTMPE RTMPTE	RTMP RTMPT RTMPS RTMPE RTMPTE
Edición de prueba	10 conexiones (gratis)	10 conexiones (gratis)	Ilimitadas.
Precio	Q. 36 000,00 US\$. 4 500,00	Q. 8 000,00 US\$. 995,00	LIBRE.
Plataformas soportadas	Microsoft® Windows Server® 2003 with Service Pack 2 Windows Server 2008 Linux® Red Hat® 4 or 5.2 Corre como una aplicación de 32 bits, no importando si el sistema es de 32 bits o de 64 bits.	Windows Mac OS X Linux Solaris Unix Soporte para 64 bits en todas las plataformas listadas.	Windows Debian/Ubuntu Mac OSX WAR Gentoo

Continuación de la tabla II.

Audio / Vídeo Streaming	FLV	FLV	(Por demanda) FLV MP3 F4V MP4 AAC M4A
	H.264	H.264	
	FLV	FLV	
	MP3	MP3	(En directo) Sorenson VP6 h.264 Nelly Moser MP3 Speex AAC NSV
	AAC, LC-AAC, HE-AAC Speex	AAC, LC-AAC, HE-AAC Speex	
CStreaming Multicliente/ Multiprotocolo	Flash (RTMP)	Flash (RTMP) iPhone (HTTP) Silverlight (Smooth) QuickTime/3GPP (RTSP/RTP) IPTV (MPEG-TS)	Flash (RTMP)
Grabación	H.264/AAC a contenedor FLV MPEG-4	H.264/AAC a contenedor FLV H.264/AAC a contenedor MP4 (Quicktime)	FLV
Lenguaje del lado del Servidor	AS2	Java	Java

Fuente: <http://askmeflash.com/article/10/comparison-wowza-vs-fms-vs-red5>. Consulta: 15 de julio de 2011.

Se puede observar en la tabla II que Flash Media Server y Red5 están muy parejas, lo que deja una buena sensación en el plan de levantar la plataforma para *streaming* a partir de este servidor, ya que si Flash Media Server es reconocido por su gran capacidad para este tipo de tareas, quiere decir que Red5 no se queda atrás en esta tarea. Los tres servidores funcionan tanto en plataformas Windows como en GNU/Linux, pero no todos funcionan en Mac, ahí es donde empieza a perder puntos el Flash Media Server.

Pero sin lugar a dudas, el que más sobresale de los 3 es Wowza, en la tabla II se pudo observar más a detalle la ventaja que mantiene ante sus rivales, siendo la más fuerte, a criterio propio, la variedad de protocolos que maneja para realizar *streaming* multiusuario, definitivamente eso le suma varios puntos para dejar un tanto abajo a la competencia que únicamente maneja RTMP.

3. DISEÑO Y DESARROLLO DE LA SOLUCIÓN DE SOFTWARE

3.1. Selección de servidor

Ésta es la primera decisión que se debe hacer, sin duda es difícil la elección de un servidor para realizar *streaming*, ya que como se observó en el capítulo anterior cada uno ofrece características diferentes e innovadoras. Si bien se mencionó que el servidor Wowza es el que sobresale entre los mencionados, lo que se hará primero es un listado de funcionalidades que tendrá la plataforma, para ver si se ajusta a las necesidades el servidor Red5.

3.1.1. Necesidades

Para la selección del servidor se deberá hacer un listado con las necesidades que se tienen, esto con el fin de realizar la mejor selección y no tener problemas en un futuro cercano.

- Capacidad de envío de vídeo en tiempo real
- Capacidad de envío de audio en tiempo real
- Múltiples canales de transferencia
- Control de recursos
- Personalizar tamaño de vídeo
- Personalizar calidad de vídeo
- Personalizar formato de vídeo
- Personalizar volumen de transmisión
- Menor costo posible en licencia

- Capacidad de distribuir contenido flash
- Sea capaz de correr sobre plataforma Linux, a fin de bajar costos
- El servidor ofrezca posibilidad de escalar sin mayores complicaciones

Según se puede observar en la tabla II, los tres servidores en mención ofrecen características diferentes, siendo Wowza el más completo, compitiendo por el segundo lugar Flash Media Server y Red5; los cuales ofrecen características similares, y por tanto un potencial similar. Ambos podrían utilizarse perfectamente para desarrollar una herramienta de *streaming* de vídeo en tiempo real; para este caso se elige Red5, tanto por su potencial, como por ser código abierto y sin ningún costo.

3.2. Descripción de herramientas

Una vez se ha elegido el servidor a utilizar, es hora de ver qué otras herramientas serán necesarias para el desarrollo, ya que no basta con la instalación del servidor de medios para continuar, se necesitan diferentes herramientas de integradas para lograr el resultado esperado.

- Java
- Eclipse Galileo
- Red5plugin
- Apache
- MySql
- Codificador/Decodificador de vídeo
- Flowplayer
- Flex

Para lograr el objetivo, estas herramientas bastan para la realización de la plataforma, pero antes de pasar a la instalación y configuración de cada una de éstas, se dará una breve explicación de cómo funcionan y para qué sirve cada una de ellas.

3.2.1. Java

Java es un lenguaje de programación orientado a objetos, desarrollado por *Sun Microsystems* a principios de los años 90”.

Para el desarrollo de la plataforma, Java tendrá un papel muy importante, por ser un lenguaje de alto nivel y con un potencial tan grande, fue el elegido para ser parte de Red5 del lado del servidor, es decir que, mediante este lenguaje se hará toda la lógica del servidor. Del lado del servidor se programará la aplicación a la cual se conectará cada cliente, es decir que aquí se verá si un cliente tiene acceso o no, de igual forma si se le quiere quitar el acceso en un momento determinado.

3.2.2. Eclipse Galileo

Eclipse es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama *Aplicaciones de Cliente Enriquecido*, opuesto a las aplicaciones *Cliente-liviano* basadas en navegadores.

Se ha elegido Eclipse, ya que aparte de ser un IDE bastante completo, ofrece la facilidad de que se puede agregar un *plugin* para facilitar el uso de Red5 en él.

Si bien es cierto, que no es estrictamente necesario el uso de un IDE para programar en Java, en este caso se utilizará por la facilidad que brinda, y que además del *plugin* de Red5 que se puede agregar, también se puede integrar Flash builder, o Flex a él, para poder desarrollar también el código *ActionScript* (AS) para el desarrollo de los clientes que se conectarán al servidor Java.

3.2.3. Red5plugin

Es una herramienta altamente productiva para el desarrollo en Eclipse, aportando la generación de código interactivo para red5, capacidad de debugear paso a paso. Además, el *plugin* soporta y ayuda en el desarrollo al utilizar bibliotecas de terceras personas, tal es el caso de *jedai*, *flexlib*, *FlexMDI*, entre otras.

La mejor de todo, además del poder que ofrece, es que Red5plugin es libre, no hay que pagar nada por utilizarlo.

3.2.4. Apache

El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.12 y la noción de sitio virtual.

Es utilizado principalmente para enviar páginas web a través del Internet. Es capaz de ser utilizado en el envío de páginas estáticas y dinámicas, muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache.

Para el caso de la plataforma de *streaming*, se estará utilizando Apache, tanto por el potencial que tiene, como por ser una herramienta de código abierto y que corre perfectamente en cualquier plataforma.

3.2.5. MySql

Es un sistema gestor de bases de datos relacionales de código abierto, que está bajo los términos de la licencia GNU. Existen versiones de pago para uso comercial, pero si se hace uso de las versiones no comerciales no se debe pagar por el uso de éste. En este caso se hará uso de una versión gratuita, ya que el poder que éstas brindan es más que suficiente para lo que se necesita para la plataforma.

La base de datos que se utilizará en la plataforma estará más que todo encargada de almacenar la información sobre los usuarios registrados, tanto usuarios distribuidores de contenido, así como los consumidores del mismo. Será únicamente accesible desde la aplicación servidor, es decir únicamente desde el servidor java.

3.2.6. Codificador/decodificador de vídeo

Es el encargado de tomar el *stream* de datos que estará recibiendo como entrada el servidor Red5, decodificarlo y codificarlo de nuevo a un formato diferente, con el fin de optimizar el vídeo que se distribuirá a los clientes conectados.

Existe una gran variedad de codificadores de vídeo, para este caso se utilizará Xuggle. El mismo se empleará por varias razones, por su potencial, ya que con él se puede codificar, decodificar y manipular audio y/o vídeo, tanto en

tiempo real como almacenado previamente, se pueden agregar efectos al vídeo, subtítulos, imágenes, prácticamente se puede hacer lo que sea de forma rápida y óptima; además es una herramienta de código abierto, lo cual encaja perfectamente en las necesidades de este documento.

3.2.7. Flex

Es una plataforma de código abierto altamente productiva, utilizada para la elaboración de aplicaciones móviles, web y de escritorio, con la característica de ser bastante expresivas. Permite la elaboración de aplicaciones web y móviles que compartan un código base, esto ayuda a reducir el tiempo y costo de la creación de aplicaciones y el tiempo de mantenimiento.

En este caso, Flex será utilizado para la creación de las aplicaciones cliente que se conectarán al servidor, explotando el potencial que posee para la elaboración de aplicaciones bonitas y llamativas, como también la capacidad que brinda de utilizar tecnología Flash para realizar conexiones con un servidor como Red5.

3.2.8. Flowplayer

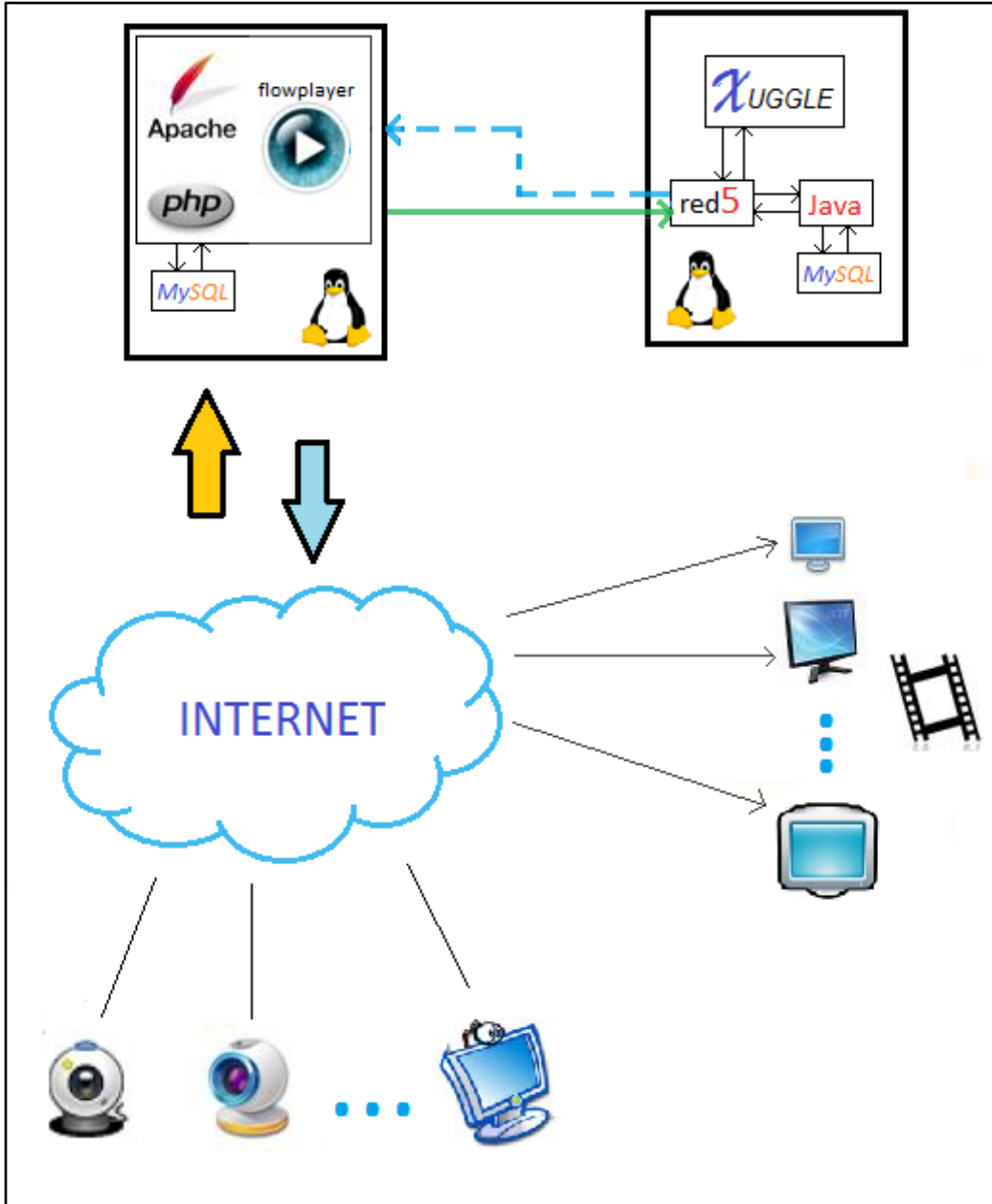
Es un reproductor de contenido multimedia que funciona embebido en páginas HTML. En el caso de la plataforma servirá para desplegar el contenido enviado por el servidor de *streaming*. Este reproductor estará del lado del servidor, por lo que no es necesario que el cliente lo instale para que pueda funcionar correctamente, ya que la instancia estará corriendo de una manera remota.

Flowplayer tiene varias versiones, incluyendo una gratuita, la cual fue liberada bajo la licencia GPL 3.0, cuenta con una nota de licenciamiento, donde se permite el uso del código, con la condición de no remover el *Copyright*.

3.3. Esquema de plataforma

A continuación se muestra un esquema, en el cual se logra apreciar de buena forma la interacción entre tecnologías, cabe mencionar que de las tecnologías descritas anteriormente, únicamente se muestran las que conforman parte de la plataforma, más no así, las que únicamente servirán como puente para su desarrollo.

Figura 7. Interacción de tecnologías



Fuente: elaboración propia.

3.4. Proceso

Se asumirá que la plataforma para difusión de *streaming* de vídeo se encuentra totalmente terminada, esto con el fin de poder detallar el proceso completo desde el principio, cuando el usuario emisor inicia una nueva transmisión, hasta el final, del lado del usuario receptor. Esto para entender de mejor forma la función que tiene cada una de las herramientas dentro de este proceso, que en realidad no es complejo, pero sin una explicación completa quizás no se llegaría a entender del todo.

3.4.1. Emisor

- a) El usuario emisor conecta su computadora a Internet.
- b) El usuario emisor ingresa al sitio web (alojado en el servidor web) e ingresa sus credenciales para iniciar transmisión.
- c) Se le presenta una aplicación Flash (embebida en HTML) en la cual se puede iniciar y terminar una transmisión.
- d) Al iniciar la transmisión, el *stream* es enviado desde la computadora del usuario emisor, hacia el servidor web (aplicación *Flash*) a través de Internet.
- e) El servidor web reenvía la información hacia el servidor de medios, Red5.
- f) Red5 recibe la información (video FLV) y la envía a Xuggle.
- g) Xuggle recibe el vídeo FLV y lo decodifica.
- h) Xuggle procesa el vídeo y lo vuelve a codificar con propósito de optimizar la salida de vídeo.
- i) Se almacena en la bitácora el inicio de transmisión.
- j) Red5 recibe el vídeo recodificado de Xuggle y lo envía de regreso al servidor web.

- k) El servidor web recibe la información y la envía al reproductor de vídeo Flowplayer.
- l) Se crea un nuevo perfil de transmisión, con un identificador único de canal.

3.4.2. Receptor

- a) El usuario receptor conecta su computadora a Internet.
- b) El usuario receptor ingresa al sitio web (alojado en el servidor web) e ingresa al canal que desea observar.
- c) Se le abre una nueva página HTML, en la cual se encuentra embebido un reproductor multimedia, *Flowplayer*.
- d) Flowplayer inicia el proceso de sincronización con el servidor web, mientras tanto se activará un GIF, para mostrar que se encuentra aún en proceso de carga.
- e) Una vez *Flowplayer* logra la sincronización de datos, este muestra el canal solicitado, esto gracias al identificador de canal que creó del lado del servidor.

3.5. Instalación y configuración de herramientas

En ésta sección se mostrará de una manera superficial la instalación y configuración de las diferentes herramientas necesarias para iniciar el desarrollo de la plataforma de *streaming*, para este caso sí se tomará en cuenta todas las herramientas necesarias, ya que sin alguna de ellas podría dar problema en el futuro.

Se dará por hecho que se cuenta con una computadora con cualquier sistema operativo GNU/Linux, Microsoft Windows, o incluso Mac OS. En este caso no se tomará en cuenta que sea una tecnología libre, ya que esto es independiente de la puesta en marcha de la plataforma, es únicamente para la construcción de la misma.

Cada sistema operativo tiene su forma de instalación y configuración de herramientas, algunas sencillas, algunas complicadas; para la instalación y configuración de herramientas, en este caso se utilizará el sistema operativo Microsoft Windows por varias razones:

- Instalaciones vía asistente, más aptas para un tutorial.
- No interfiere en el funcionamiento de la plataforma, únicamente en el desarrollo de la misma.

La configuración que se hará para cada herramienta será la configuración base, sin profundizar más de lo necesario, únicamente lo fundamental para el correcto funcionamiento e interacción entre tecnologías

3.5.1. Apache

Lo primero en hacerse es descargar el instalador del servidor web, para esto se debe dirigir al sitio web de apache, a continuación el enlace para la descarga del instalador.

<http://httpd.apache.org/download.cgi>

Una vez descargado, se debe instalar en el sistema ejecutando el archivo obtenido.

- a. Se abrirá un asistente el cual guiará durante el proceso de instalación del servidor.
- b. El asistente preguntará por el nombre del dominio del servidor, el cual se debe llenar con *localhost*.
- c. Pedirá una dirección de correo, ésta puede ser llenada con cualquiera que se desee.
- d. Preguntará si se desea que el servidor responda a través del puerto 80 para todos los usuarios, o solamente para el usuario actual por el puerto 8080 al iniciar manualmente; se marca la primera opción.

Con estos pasos queda configurado inicialmente *Apache*, para el propósito de desarrollo está bien hasta aquí, ahora, si quisieran hacerse configuraciones adicionales más adelante, se pueden hacer vía el archivo de configuración `httpd.conf` que se encuentra en el fichero de instalación de Apache, este fichero se tuvo que haber indicado en el proceso de instalación.

Es importante mencionar que, para que *Apache* funcione correctamente, no debe haber ningún otro servidor web corriendo a la vez, al

menos no ambos escuchando por el mismo puerto. Generalmente, en las versiones profesionales de Microsoft por defecto está configurado el servidor IIS, de ser así, basta con deshabilitar IIS ingresando al panel de control, luego seleccionar Programas, y luego elegir deshabilitar o habilitar características de Windows; se desplegará un listado con varias características de Windows, se debe ubicar Internet Information Services y deshabilitar el cheque correspondiente a ésta opción.

Para comprobar el funcionamiento del servidor, basta con abrir un navegador y colocar la dirección `http://localhost`, debería de mostrarse una página con el mensaje *It Works!*, indicando éxito.

3.5.2. Java

Ya que se tiene el servidor web, es necesario pasar a la instalación y configuración de Java, para ello se debe descargar la JDK más reciente de la página de Oracle.

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

- a. Ya con el instalador de la JDK se debe pasar a la instalación, ejecutando el archivo que se descargó.
- b. Ahora, prácticamente es seguir el asistente que guiará a través del proceso, deben de aceptarse los términos de la licencia.
- c. Una vez instalado, deben configurarse las variables de entorno; aunque puede ser que no sea necesario, para ello hay que ejecutar el comando `javac` en una ventana de símbolo del sistema, si al hacer esto salen

varias líneas de ayuda, quiere decir que no es necesario, de lo contrario hay que pasar al punto 4.

- d. Para agregar una variable de entorno se debe ir a las propiedades del equipo e ir a opciones avanzadas, luego variables de entorno.
- e. Debe editarse la variable *Path*, agregando al final de ella la ubicación donde fue instalada la JDK, se debe tener cuidado de no borrar nada de lo que ya existía en la variable, solo agregar la parte nueva precedida por un punto y coma.
- f. Ahora se debe hacer lo mismo que en el punto 5, sólo que ésta vez con la variable llamada *CLASSPATH*.
- g. Aceptar los cambios, y ahora debería de estar correctamente instalado y configurado para utilizarse la *JDK*.

3.5.3. Red5

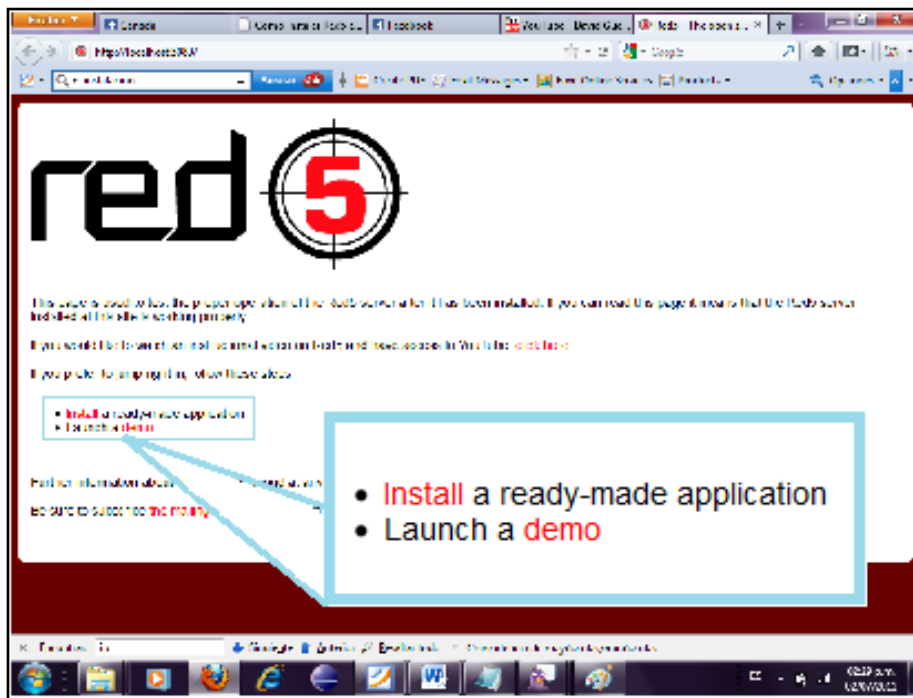
Una vez instalado y configurado correctamente el servidor web y Java, se puede pasar a la instalación del servidor de medios. Primero debe descargarse el instalador del sitio oficial de Red5.

<http://www.red5.org/red5-server/>

Actualmente se encuentra en la versión 0.9.1 Final, o bien la 1.0 como candidato a ser nueva versión. Una vez descargado, se ejecuta el instalador descargado y se configura de la siguiente forma:

- a. Abrirá un asistente, el cual guiará a través del proceso de instalación.
- b. Pedirá la carpeta de instalación, puede ser en cualquier lugar, sin embargo, se hará en la carpeta archivos de programa.
- c. Solicitar la IP en la cual se desea instalar el servidor, a diferencia de como se hizo con Apache, en este caso se debe dejar en blanco y aceptar, esto con el fin de que se reconozca cualquier interfaz de la computadora y no dejarlo atado a una única IP.
- d. Pedirá el puerto a utilizar para peticiones HTTP, en ésta casilla se debe especificar el puerto 5080 y aceptar.
- e. Después de un momento terminará la instalación, para comprobar que esté instalado correctamente se debe abrir un navegador y colocar la dirección `http://localhost:5080`, si todo está bien aparecerá una ventana con el logotipo de Red5.
- f. Una vez dentro de la página, se puede ver que hay 2 opciones, una para instalar demos y otra para correr los demos.

Figura 8. Página inicial de Red5

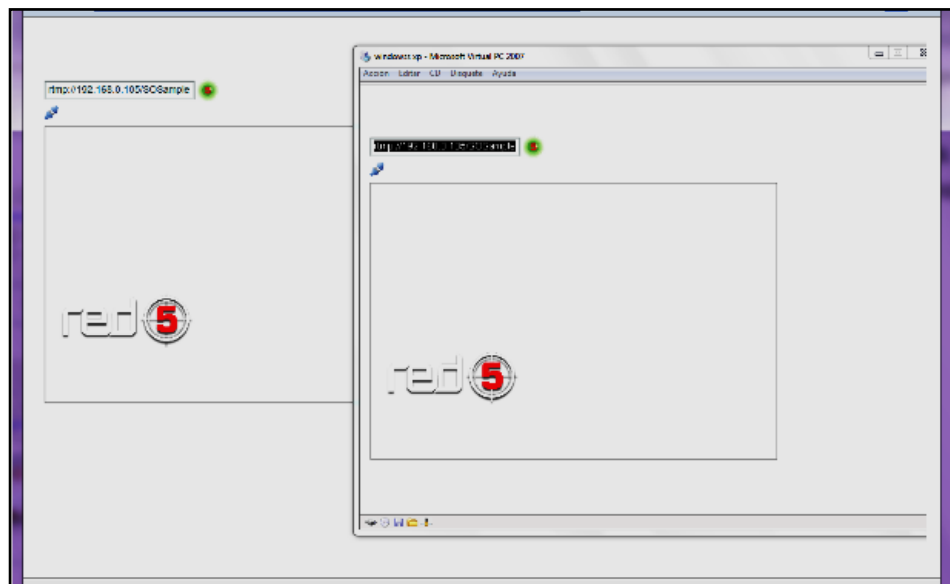


Fuente: elaboración propia.

- g. Ingresar a la sección de instalaciones (primer enlace) mostrada en la imagen.
- h. Instalará la demo llamada SOSample, para probar el correcto funcionamiento, para ello basta con seleccionar la aplicación y luego hacer clic en *Install*.
- i. Hecho esto, se podrá tener el primer contacto con la tecnología de Red5, en este caso se mostrará cómo funciona un objeto compartido; habrá que abrir 2 ventanas de un navegador y colocar ésta dirección <http://localhost:5080/demos/BallControl.html> en ambas.

- j. Se observará una aplicación sencilla con el logo de Red5 en cada ventana. A continuación debe hacerse clic al botón conectar en ambas aplicaciones para sincronizarlas entre sí.

Figura 9. **Objeto compartido en Red5**



Fuente: elaboración propia.

- k. Si todo va bien es la hora de probar, basta con arrastrar el logo red5 de cualquiera de las 2 aplicaciones hacia una nueva posición y podrá verse que automáticamente se mueve también en la otra aplicación, esto muestra la sincronización que existe entre 2 aplicaciones cliente y 1 aplicación servidor que viene lista para usar en el servidor.

El servidor de Red5 puede iniciarse y detenerse manualmente, para ello debe de ingresar a los servicios de Microsoft Windows, ejecutando Services.msc, adentro seleccionando Red5 y luego iniciar/detener.

3.5.4. Eclipse

Eclipse será el ambiente con el que se estará en contacto durante la mayor parte del desarrollo de la plataforma, es por eso que se vuelve una parte fundamental. Para este propósito debe descargarse la versión Eclipse Galileo, puede funcionar con otras versiones; sin embargo, para ésta explicación se utilizará la versión mencionada. Debe descargarse el instalador de la página oficial. Ya se mencionó la versión de Eclipse, pero no se ha dicho que tiene que ser para desarrollo en *Java Enterprise Edition*, JEE.

<http://www.eclipse.org/downloads/>

- Para tener una instalación de eclipse, primero se debe descomprimir el archivo descargado.
- Ahora basta con ubicar la carpeta obtenida de la descompresión, en cualquier lugar del disco duro, ya que Eclipse da esa facilidad, no necesita instalarse y esto permite tener varias instancias de él, en diferentes versiones, lo cual no provocará problema.
- Ya que no es una instalación como a las que se está acostumbrado, ahora quizás resulte necesario, y a la vez cómodo crear un acceso directo para poder acceder más fácilmente a él.

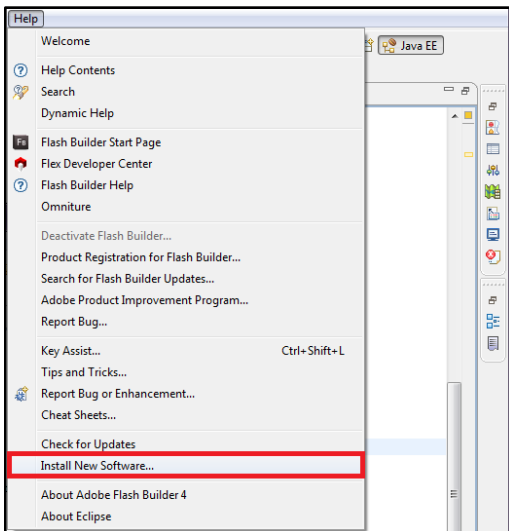
3.5.5. Red5plugin

Este agregado, quizás es el más fácil de instalar, ya que basta con agregar la siguiente dirección como nueva instalación de software:

http://trac.red5.org/chrome/site/projects/red5plugin/update_0.2.x/

- a. Hacer clic en el menú de Ayuda y luego elegir la opción de Instalar Nuevo Software.

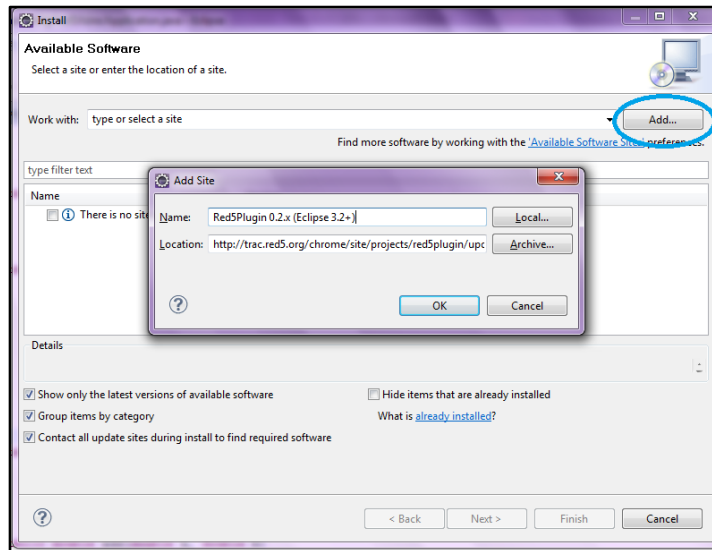
Figura 10. Paso 1, instalación red5plugin



Fuente: elaboración propia.

- b. Debe pulsar agregar.
- c. Ahora se solicitan los datos de la instalación, en la casilla de nombre puede ser cualquier identificador, en la localidad se debe poner la dirección especificada al inicio de la configuración de red5plugin, aceptar.

Figura 11. Paso2, Instalación red5plugin



Fuente: elaboración propia.

3.5.6. MySQL

Para la instalación de MySQL se debe ir al sitio oficial y descargar la última versión gratuita de este gestor de bases de datos, la versión gratuita se denomina *MySQL community server*, la dirección es:

<http://dev.mysql.com/downloads/mysql/>

- a. Una vez descargado el instalador, hay que ejecutarlo.
- b. Se abrirá un asistente, el cual lo guiará a través de toda la instalación.
- c. En el caso de desarrollo, no se necesita de una base de datos totalmente configurada, por eso solo se cambiarán algunas de las opciones de la instalación típica:

Detailed Configuration

Server Machine

Transactional Database Only

Colocar la contraseña de usuario Root

Dejar todo lo demás por defecto

Para comprobar que se ha realizado correctamente la instalación, basta con abrir una ventana de símbolo de sistema y teclear: *Mysql -u root -p*

Seguido de esto *MySQL* solicitará la contraseña de usuario Root, si todo está bien se mostrará el número de versión que se ha instalado y se dará acceso a la consola de comandos de *MySQL*.

3.5.7. Flowplayer

El reproductor, es por decirlo de alguna forma, el menos importante hasta el momento, ya que es el último que se agrega a la plataforma; sin embargo, se explicará desde ya, la instalación y configuración del mismo.

Antes que todo se debe descargar los archivos necesarios para la adaptación de *flowplayer* a una página HTML; el sitio oficial de *flowplayer* para realizar la descarga es:

<http://flowplayer.org/download/>

Deben descargarse los archivos siguientes, cabe mencionar que las versiones pueden variar:

- Flowplayer-3.2.7

- Flowplayer.controls-3.2.5
- flowplayer.rtmp-3.2.3

Una vez descargados los archivos deben de pegarse dentro de la carpeta donde se encuentre la página en la cual se va a embeber el reproductor multimedia.

3.6. Creando la primera aplicación

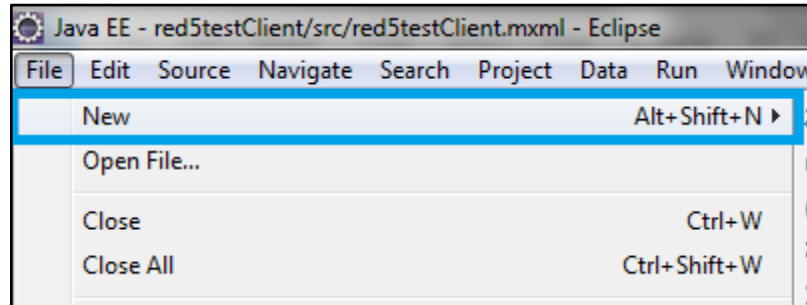
Una vez se tengan configuradas todas las herramientas necesarias es hora de realizar la primera aplicación real, para esto es necesario abrir Eclipse, al inicio se pedirá crear un nuevo espacio de trabajo, o *workspace*, aquí se puede elegir un nombre cualquiera que represente el proyecto que se trabajará.

La forma en que se mostrará cómo crear una nueva aplicación desde cero se hará mediante una serie de imágenes, ya que de esta forma se puede mostrar fácilmente paso a paso, esto con el fin de que no se cree confusión ni dudas en ninguno de los pasos.

3.6.1. Nuevo

Dentro de Eclipse, hacer clic en el menú *File*, se desplegarán varias opciones, elegir *New*.

Figura 12. Paso 1, nueva aplicación

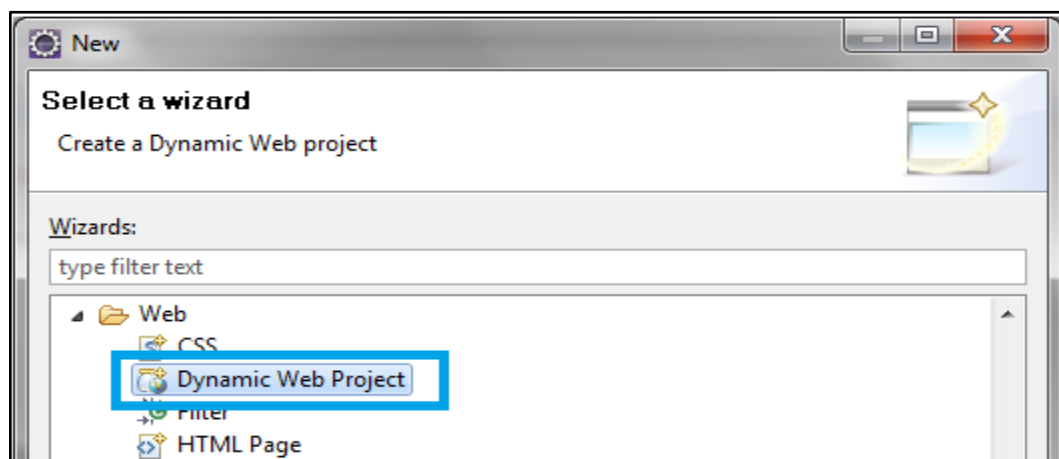


Fuente: elaboración propia.

3.6.2. Tipo de proyecto

Ahora se mostrará una nueva ventana, en la cual se puede elegir el tipo de proyecto nuevo a realizar, en ésta se debe seleccionar la opción *Dynamic Web Project*.

Figura 13. Paso2, nueva aplicación



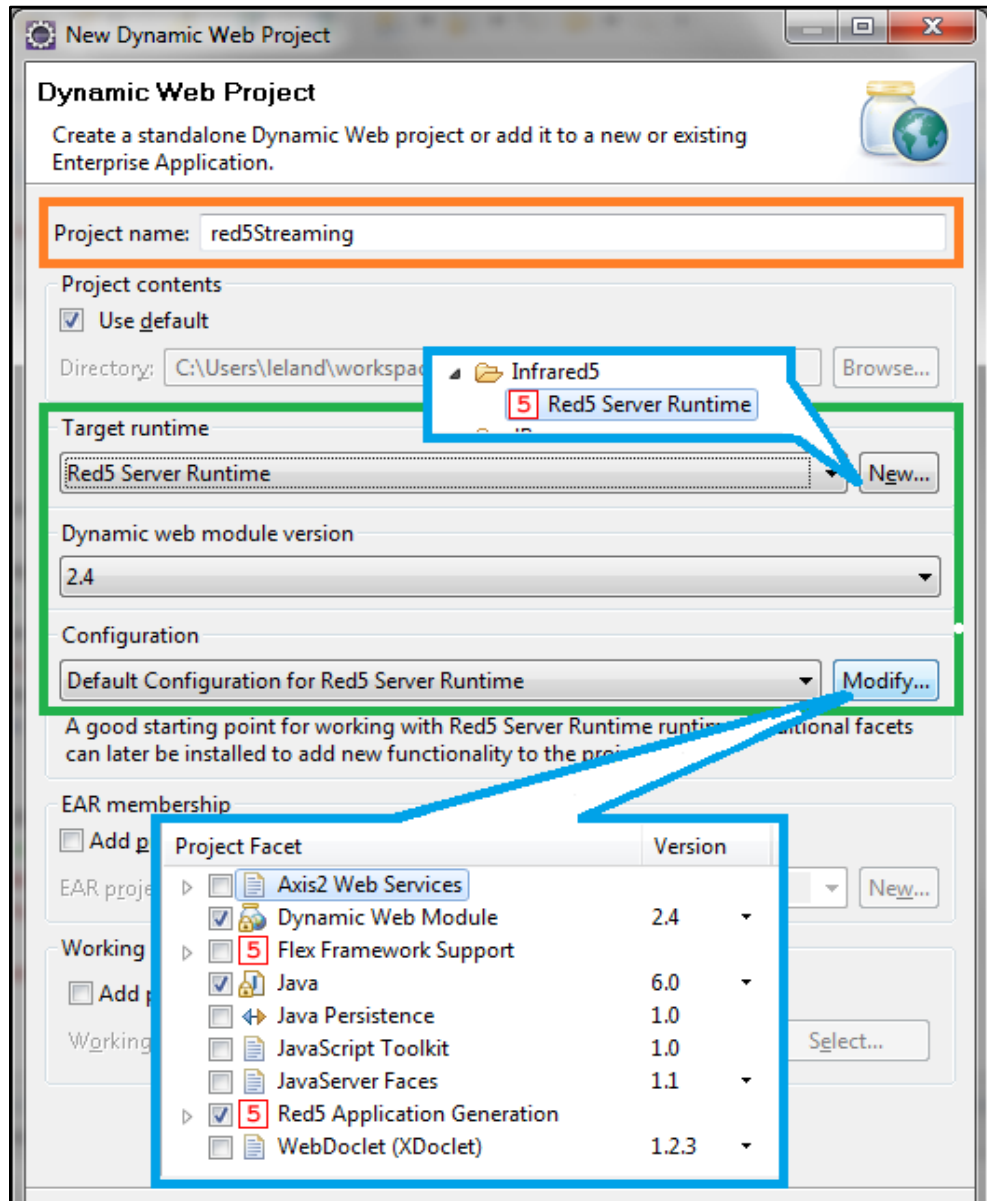
Fuente: elaboración propia.

3.6.3. Configuración de proyecto

En esta ventana se deben configurar 4 opciones:

- a. Colocar el nombre de la aplicación, puede ser el que se desee.
- b. Cambiar el objetivo, o *Target runtime* a *Red5 Server Runtime*.
- c. Elegir 2.4 como versión de *Dynamic web module*.
- d. Por último, configurar como *Default configuration for Red5 Server Runtime*.

Figura 14. Paso 3, nueva aplicación



Fuente: elaboración propia.

Hecho esto deberían de aparecer 2 proyectos, *Java* y *Flex*, el primero servirá como el servidor donde se creará la lógica de conexiones entrantes, en el segundo irán los clientes que se conectarán al servidor java.

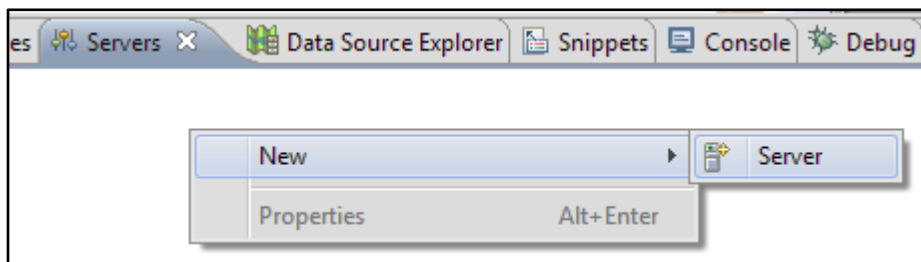
Si se explora en el árbol del proyecto servidor, se encontrará el archivo llamado `Application.java`, para fines de una primera aplicación que no necesita mayor funcionalidad, éste se dejará tal y como fue generado.

A pesar de no necesitar algo extra, se debe generar un contenedor para el servidor. A continuación los pasos para crearlo.

3.6.4. Nuevo servidor

Localizar la pestaña *Servers* en la parte inferior de Eclipse, luego dar clic derecho dentro de ella y se desplegará un menú, ahí se debe elegir *New*, y luego *Server*.

Figura 15. Crear servidor Red5

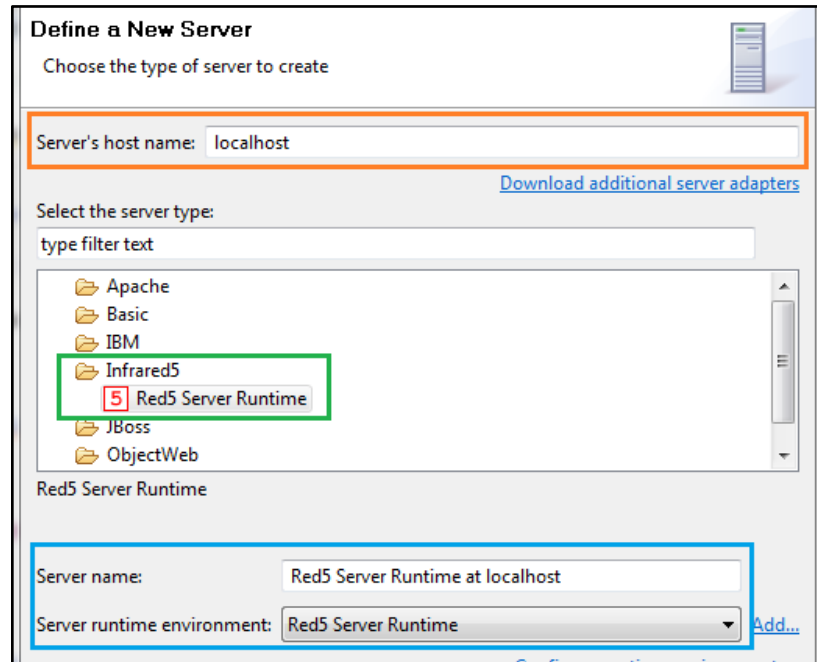


Fuente: elaboración propia.

3.6.5. Tipo de servidor

Se abrirá una nueva ventana, en ésta se debe colocar un nombre para el servidor, como también elegir un tipo de servidor, en este caso la opción *Red5 Server Runtime*.

Figura 16. Elegir tipo de servidor



Fuente: elaboración propia.

3.6.6. Deploy y scripts de inicio de Red5

Indicar dónde desea hacerse el *deploy* del proyecto, en este caso en la carpeta webapps de la instalación de Red5. También debe indicarse en dónde se encuentran los *scripts* (.bat) para iniciar y detener Red5.

Figura 17. **Path** de deploy y scripts

New Red5 Server Runtime Server	
Create a new Red5 Server Runtime server	
Auto Deploy Directory	C:/webapps <input type="button" value="Browse..."/>
Start Script	C:/red5-debug.bat <input type="button" value="Browse..."/>
Stop Script	C:/red5-shutdown.bat <input type="button" value="Browse..."/>
Server Port	5080
Debug Port	8787

Fuente: elaboración propia.

4. PUESTA EN MARCHA Y VISIÓN A FUTURO

Con base en lo que se ha plasmado en los capítulos previos, debería de ser factible el desarrollo y puesta en marcha de un servidor de medios de forma local, es decir, en la computadora del desarrollador; pero no serviría de mucho tener un servidor únicamente en una computadora si no tiene acceso a Internet, o bien, si no tiene la capacidad de funcionar como servidor, es por ello que en este capítulo se abordará sobre este tema y algunos factores que se consideran importantes para lograr una puesta en marcha satisfactoria.

4.1. Ancho de banda

El avance en infraestructura por parte de los proveedores de servicio de Internet ha logrado que con menos y mejor equipo puedan transferirse cantidades mucho mayores de datos cada día, esto ha permitido a los proveedores brindar un mejor servicio a un costo cada vez menor, permitiendo al usuario final adquirir enlaces de alta capacidad a costos accesibles; sin embargo, cuando se transmite vídeo a través de Internet hay que ser cuidadoso, ya que puede saturarse el enlace si no se optimiza el vídeo y su envío.

En el caso de *streaming* de vídeo se considera una tasa aceptable transmitir a 256Kbps. Dependiendo de la modalidad de transmisión que se desarrolle, así será el ancho de banda de subida requerido para el servidor.

Es realmente determinante la cantidad de tráfico que se puede llegar a tener cuando se transmite vídeo en Internet, por ejemplo, si se tuviera la

plataforma funcionando 1 hora, con 1 persona conectada a ella recibiendo el audio y vídeo, y con una tasa de transferencia de 256 Kbps, el servidor necesitaría enviar 112.5 MB de información, al ver esta cantidad talvez no supone una alta cantidad de datos, pero esto se complica al aumentar el número de personas que verán el evento. Si el número de personas se incrementara a 10, esto cambiaría bastante, se estaría teniendo que transmitir 1125MB (1.09GB) por cada hora que se transmita el evento, una cantidad bastante considerable, para un número bajo de personas. Si la plataforma mantuviese un promedio de 10 personas conectadas no importando la hora y ésta estuviese funcionando las 24 horas del día, esto significaría 1125MB x 24 horas, es decir 27 000 MB (26,37GB). Ver ésta cantidad es impactante, ya que consumir 27,37 Gigabytes de ancho de banda para un solo día implica un costo elevado en la adquisicion del servicio, es por ello que es sumamente importante realizar estos cálculos antes de contratar un servicio, ya que si se falla desde el punto de partida, esto puede originar fallos en las siguientes etapas.

4.1.1. Cálculo

Es relativamente sencillo calcular cuanto ancho de banda se necesita, siempre y cuando se tenga el número promedio estimado de usuarios que estarán conectados a la plataforma.

Previamente se mencionó cuál es el ancho de banda recomendado para la transmision de vídeo por la red, 256 kbps, o bien 256 kb por segundo. Ésta es la base para calcular el ancho de banda necesario mensualmente; se debe tener claro que 1 kb equivale a 1024 b (bits), entonces:

$$256 \text{ kbps} = 256 \times 1024 \text{ bits}$$

Es decir, una tasa de 256 kbps equivale a 262 144 bits por segundo, dicho de otra forma, 262 144 señales binarias, por segundo; dichas señales son enviadas por el servidor hacia el cliente. Sabiendo esto se tiene la base para continuar con los cálculos del ancho de banda que se necesita para aproximar lo que se necesita mensualmente, entonces:

$$256 \text{ kbps} = 262\,144 \text{ bps}$$

$$262\,144 \times 60 \text{ segundos} = 15\,728\,640 \text{ bits por minuto}$$

$$15\,728\,640 \times 60 \text{ minutos} = 943\,718\,400 \text{ bits por hora}$$

Se puede observar que la cantidad de bits para la transmisión de una hora es de casi 1 gigabit, esto para un sólo cliente. Para tener esta cantidad de una forma más familiar se pasará de bits a bytes, para esto lo único que debe hacerse es dividir por 8.

$$943\,718\,400 / 8 = 117\,964\,800 \text{ bytes}$$

Pero aún así, es recomendable pasarlo a una cantidad mucho más familiar, ésta vez se dividirá por 1024 para pasarlo a kilobytes y después otra vez por 1024 para pasarla a Megabytes, que es la dimensional más popular.

$$117\,964\,800 / 1024 = 115\,200 \text{ kB}$$

$$115\,200 / 1024 = 112,5 \text{ MB}$$

Se puede observar la cantidad final de una forma más familiar a la primera observada. La cantidad total de datos es de 112,5 Megabytes, para

transmitir una hora de vídeo a una sola persona, una cantidad que talvez no aparenta ser tan alta, pero si se multiplica por el número de clientes conectados, por varias horas al día, ésta sube considerablemente.

A continuación se muestra la tabla III en la cual se puede observar la gran cantidad de ancho de banda que podría llegar a darse dependiendo de los clientes y tiempo que permanezcan dentro.

Tabla III. **Consumo de ancho de banda en streaming de vídeo**

<i>1 Cliente</i>		<i>10 Clientes</i>		<i>50 Clientes</i>	
Tiempo	GB	Tiempo	GB	Tiempo	GB
1h	0,11	1h	1,10	1h	5,49
2h	0,22	2h	2,20	2h	10,99
3h	0,33	3h	3,30	3h	16,48
4h	0,44	4h	4,39	4h	21,97
5h	0,55	5h	5,49	5h	27,47
6h	0,66	6h	6,59	6h	32,96
7h	0,77	7h	7,69	7h	38,45
8h	0,88	8h	8,79	8h	43,95
9h	0,99	9h	9,89	9h	49,44
10h	1,10	10h	10,99	10h	54,93
11h	1,21	11h	12,08	11h	60,42
12h	1,32	12h	13,18	12h	65,92
13h	1,43	13h	14,28	13h	71,41
14h	1,54	14h	15,38	14h	76,90
15h	1,65	15h	16,48	15h	82,40

Continuación de la tabla III.

16h	1,76	16h	17,58	16h	87,89
17h	1,87	17h	18,68	17h	93,38
18h	1,98	18h	19,78	18h	98,88
19h	2,09	19h	20,87	19h	104,37
20h	2,20	20h	21,97	20h	109,86
21h	2,31	21h	23,07	21h	115,36
22h	2,42	22h	24,17	22h	120,85
23h	2,53	23h	25,27	23h	126,34
24h	2,64	24h	26,37	24h	131,84

Fuente: elaboración propia.

En la tabla III puede observarse la cantidad de GB que pueden llegar a transferirse en un solo día, desde 0,11GB que significa un cliente por una hora, hasta 50 clientes conectados simultáneamente las 24 horas, que consumirían 132GB aproximadamente, una cantidad enorme para un solo día.

¿Y qué pasa si en lugar de mostrar datos para un sólo día, se muestran datos de semanas completas?, es ahí en donde se logra ver el consumo a gran escala y de porqué es que se debe de hacer un estudio previo, sería terrible tener una caída de servidor porque se terminó el ancho de banda disponible.

En la la tabla IV se muestra la cantidad de GB que serían utilizados en ancho de banda basado en los datos la tabla III.

Tabla IV. **Consumo de ancho de banda por semana**

<i>1 Cliente</i>		<i>10 Clientes</i>		<i>50 Clientes</i>	
Tiempo	GB	Tiempo	GB	Tiempo	GB
1 semana	18,48	1 semana	184,59	1 semana	922,88
2 semanas	36,96	2 semanas	369,18	2 semanas	184576
3 semanas	55,44	3 semanas	553,77	3 semanas	2768,64
4 semanas	73,92	4 semanas	738,36	4 semanas	3691,52

Fuente: elaboración propia.

Viéndolo de forma más real, se puede observar el enorme ancho de banda necesario para transmisión de vídeo si se estuviera utilizando a tiempo completo durante un mes completo, quizás esto no sea así, pero en algún momento podría suceder, es por ello que es obligatorio realizar un estudio previo para contratar el servicio de ancho de banda

4.2. Memoria RAM

La memoria RAM es muy importante para que se puedan correr varios programas simultáneamente en una computadora, no representa un costo elevado como hace algunos años lo hacía, por lo que no representa problema en su adquisición para instalar en un equipo personal. En el caso de un servidor de *streaming* no se requiere de mucha memoria RAM para su buen funcionamiento, es cierto que van a haber muchos usuarios conectados a él, por lo que tampoco se puede dejar una cantidad tan baja, lo que sí es cierto es que por cada usuario que se conecte no se otorga gran cantidad de memoria, lo más importante es el ancho de banda.

4.3. Procesador

Éste servirá, principalmente, para procesar el contenido de vídeo recibido, procesar la decodificación y posteriormente volver a codificarlo en el formato adecuado para ser enviado al cliente.

La capacidad del procesador es proporcional al número de clientes emisores que estén conectados a la plataforma, no tanto de los clientes receptores, ya que éstos no envían nada a la plataforma, únicamente reciben; en cambio los clientes emisores estarán enviando una cadena de datos a la plataforma para que ésta la procese y la envíe hacia los clientes receptores, quienes verán el resultado final.

4.4. Disco duro

Éste sirve para almacenamiento de programas instalados en el servidor, como también para las configuraciones de los mismos. El espacio en disco duro no es problema, ya que no se necesita de una gran cantidad si únicamente se está transmitiendo *streaming*, no así si también se estuviera almacenando cada uno de los eventos que pasan por la plataforma, ya que si así fuera, debería de tenerse un espacio bastante amplio para cada canal de transmisión, cosa que no sale a bajo costo en absoluto.

4.5. Latencia

Un término no tan usual, pero que es de tomar muy en cuenta a la hora de contratar un servidor externo. Latencia se refiere al tiempo que tarda en viajar un paquete desde la computadora cliente, hacia el servidor, y de regreso.

Es importante que la latencia del servidor sea baja, ya que entre más baja sea, significa que más rápida será la transmisión de datos, para ello debe seleccionarse un servidor que no se encuentre tan lejos. Sería un error contratar un servicio de otro continente, por ejemplo, ya que el tiempo que toma en viajar un paquete desde Guatemala a Europa es mayor al tiempo que toma desde Guatemala hacia un país del continente americano.

4.6. Elección del servidor

Como se pudo ver en los aspectos a tomar en cuenta para contratar un servidor externo, lo más importante para brindar *streaming* de vídeo es el ancho de banda, a continuación se detallarán las características que se utilizarán para el montaje a futuro de esta plataforma, éstas características serán mencionadas desde la menos importante, hasta la que mayor importancia presenta.

4.6.1. Disco duro

No es tan importante en el caso de ésta plataforma, ya que únicamente se tiene pensado transmitir vídeo, no es necesario que se almacene; tampoco se descarta que en el futuro vaya a proveerse este servicio, por lo que el se eligió un proveedor de una fácil escalabilidad en este sentido.

La cantidad elegida finalmente es de 40GB, suficiente para almacenamiento de programas y configuraciones, como también para realizar los *backups* reglamentarios por cualquier problema que surja.

4.6.2. Memoria RAM

Como se mencionó anteriormente tampoco es tan importante, debe de ponérsele cuidado, no debe descuidarse ni dejarse en el olvido. El uso de la memoria se irá midiendo a lo largo del tiempo, se verá si es necesario un incremento o si todo está funcionando correctamente y no se necesita más memoria. Inicialmente el servidor tendrá 1GB de memoria RAM, una cantidad que de aceptable tiende a suficiente.

4.6.3. Procesador

Al igual que se mencionó anteriormente, éste entrará en funcionamiento sobre todo cuando hayan más clientes emiores conectados y transmitiendo, ya que por cada uno que haya tendrá que procesar la entrada y devolver una salida. Para el caso del servidor para la plataforma de *streaming* se iniciará con un procesador de 1024MHz.

4.6.4. Ancho de banda

Éste es el factor más importante, porque todo el tráfico pasará por acá, y si el ancho de banda llega a terminarse, ocasionaría pérdidas de dinero, es por ello que debe considerarse un proveedor que posea una escalabilidad fácil y no tan cara en este aspecto.

Se eligió para iniciar el servicio de la plataforma un ancho de banda de 900 GB de transferencia mensuales, el proveedor ofrece crecimiento de hasta 3000 GB mensuales, una cantidad excelente de transferencia mensual y cuyo costo no es elevado.

4.6.5. Proveedor

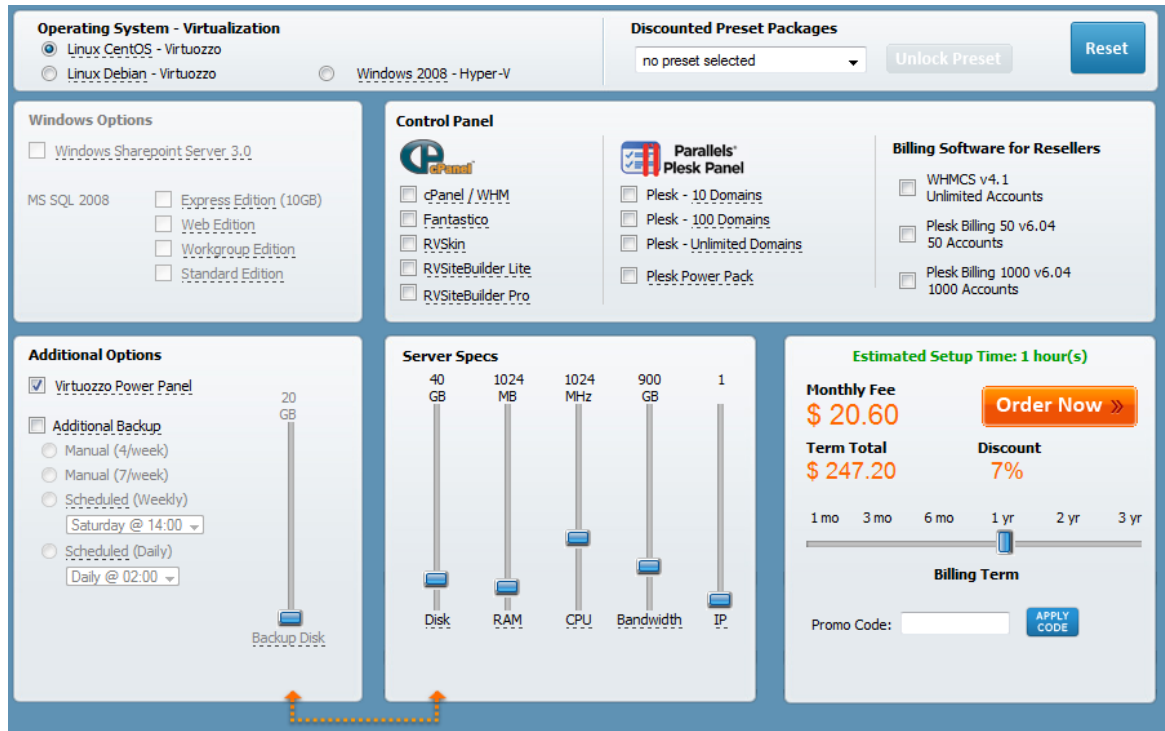
Para la elección de proveedor, se investigó las características que ofrecen varias empresas, entre ellas GoDaddy, Hostgator y MyHosting, siendo éste último el que llenó las expectativas, brindando mayor capacidad de almacenamiento y transferencia de datos, por un precio menor; además MyHosting da la opción de elegir entre dos distribuciones de Linux, lo cual es ideal para continuar con el uso de tecnologías libres.

www.myhosting.com

Éste provee servicios de calidad, con mejores precios que varios de sus competidores, además tiene una latencia baja, incluso se comparó con sistemas de renombre y *myhosting* respondió con latencia menor.

A continuación se presenta una imagen de la configuración elegida en el proveedor, junto con el precio final, el cual no resultó ser muy alto, cabe destacar que se estaría contratando un servidor privado virtual (VPS), y no un servidor dedicado, ya que estos son bastante caros y no entran en el presupuesto inicial.

Figura 18. Características iniciales de servidor



Fuente: elaboración propia.

Puede observarse que las características elegidas son fácilmente escalables en el proveedor *myhosting*, por lo que, si fuese necesario incrementarlas en un momento determinado, se hará fácilmente a través de una solicitud al Technical Assistance Center (TAC) del proveedor.

No debe dejarse por un lado la elección del sistema operativo para el servidor, basado en rendimiento y precio, se ha elegido GNU/Linux, en su distribución CentOS, combinado con una interfaz gráfica Virtuozzo para su fácil administración. El precio del servicio mensual será de 20,60 dólares, que traducido a quetzales serían aproximadamente Q.160,00, un precio bastante accesible.

4.7. Visión a futuro

Todo en la vida debe evolucionar, y no es la excepción en el caso del software, ya que el Software que no evoluciona se queda relegado y tarde o temprano quedará obsoleto, es por ello que debe de pensarse en los recursos que serán necesarios para agregar nuevas funcionalidades.

Actualmente se tiene una transmisión de audio/vídeo que funciona de forma continua y sin interrupciones, a futuro se busca optimizar la transmisión, es decir, buscar la mejor forma de transmisión e ir innovando en cuanto a formatos de vídeo con el fin de transmitir cada vez a mejor calidad y de la forma más fluída posible.

4.7.1. Creación de usuarios

Una mejora que puede hacerse es en la forma de manejar los múltiples canales, ya que actualmente éstos se crean de forma manual desde el código, aún no existe una forma automatizada de realizarlo, algo fundamental para un mejor funcionamiento de la plataforma, si bien es cierto que la implementación de esto no es complicada, será hasta un siguiente release que entre en funcionamiento.

4.7.2. Forma de cobro

Se realizó investigación de las diferentes formas de cobro que existen para aplicaciones de Internet, las que se adecúan más para implementar en la plataforma son:

- Paypal

- MicroPagos vía SMS
- Depósito a cuenta bancaria

Los más interesantes son Paypal y Micropagos vía SMS, ya que ambos pueden funcionar de forma automática en conjunto con la plataforma, mientras que los depósitos bancarios deben ser manejados por un tercero quien está encargado de revisar que en realidad haya sido realizado el depósito y luego abonar saldo a la cuenta del usuario.

La forma más llamativa y más factible en Guatemala es realizando pagos vía SMS, ya que la mayoría de personas en el país posee un celular, y qué más fácil que tomar un celular y realizar un pago enviando un mensaje de texto a un número específico, aparte de fácil, es seguro, ya que esa modalidad no transmite la inseguridad que en su momento llega a transmitir el ingresar un número de tarjeta de crédito en un sitio de Internet por cualquier forma de engaño y/o ataque, por mencionar algunos, *phishing* que consiste en hacerse pasar por alguien más, o bien por el ataque *Man-in-The-Middle* que consiste en interceptar los datos en el medio de la transmisión.

Para realizar cobros vía mensaje de texto existen 2 formas de realizarlo, la primera y más fácil es subcontratando a un tercero, por ejemplo:

- Interacel
- Quetzalotto
- Telepromos

Estas empresas cuentan con la infraestructura para realizar este tipo de cobros, la desventaja es que aparte de lo que cobraría la empresa de telefonía, también estaría el cobro de ésta tercera empresa.

La segunda forma de realizar cobros vía mensaje de texto es montando un Gateway SMS propio, para luego ponerse de acuerdo con las compañías de telefonía móvil para el cobro especial que debería realizarse en el número adquirido.

Para montar un Gateway SMS se necesita de:

- Servidor físico (computadora).
- Software para servicio de SMS Gateway.
- Modem 3G, que puede ser un modem de los que se utilizan para acceder a Internet, o bien algún celular compatible.

Una vez montado el Gateway SMS, basta con comunicarlo con la base de datos que llevará el control de usuarios, esto con el fin de realizar la acreditación de saldo a la cuenta de cada usuario que envíe su mensaje de compra de saldo.

El montar un servidor propio para recibir mensajes de texto tiene ventajas y desventajas, entre las ventajas que se tienen están:

- Mejor control sobre los mensajes que se reciben, pudiendo manipular el texto como se desee, sin limitaciones ni pagos adicionales por funcionalidades solicitadas.
- Mejor ganancia, ya que únicamente estaría de por medio la empresa de telefonía quien cobraría su porcentaje por el servicio prestado.

Entre las desventajas están:

- Inversión de tiempo
- Inversión monetaria (Adquisición de módem y computadora)
- Consumo de energía eléctrica de servidor
- Espacio físico para servidor

Indiferentemente de la forma que se utilice para realizar cobros, unavez se tenga la plataforma en línea, con una vía para realizar cobros se puede empezar a realizar transmisiones, con el cuidado y monitorización pertinentes para llevar un mejor control de la evolución que éste va teniendo con el tiempo.

CONCLUSIONES

1. Gracias al avance de tecnología que existe en la actualidad, es posible desarrollar aplicaciones de software poderosas sin invertir demasiado tiempo para su realización, por medio de este avance tecnológico se logró desarrollar un prototipo de plataforma capaz de transmitir vídeo de alta calidad a través de la red, pudiendo éste ser visto desde cualquier computadora que tenga acceso al servidor, sea por vía red o NAT mediante Internet.
2. Los servidores de medios que existen en la actualidad, y en este caso Red5, son herramientas poderosas, que permiten realizar transmisión de vídeo con un retraso mínimo, lo cual permite tener una mejor experiencia a los receptores de la imagen.
3. Red5, con la ayuda de *Xuggle*, proveen la facilidad de enviar vídeo en forma compacta y optimizada, esto con el fin de ocupar la menor cantidad posible de ancho de banda para enviar vídeo a través de la red.
4. Red5 brinda la facilidad de tener dentro de un objeto *netStream* audio y vídeo, por lo que se logra una sincronización excelente de éstos, lo cual garantiza una reproducción agradable para el receptor.
5. Con Red5 es posible crear *multiroom* utilizando un mismo servidor, esto facilita la implementación de múltiples canales de conexión integrados a

la plataforma, lo cual hace posible la transmisión de eventos independientes.

6. Se puede observar que mediante la utilización de software *Código Abierto* es posible crear aplicaciones de alto nivel, no es necesario invertir en licencias costosas, ya que el software libre brinda la oportunidad de realizar grandes cosas, quizás un poco más difícil de desarrollar, ya que no existe tanta documentación como con herramientas privativas, pero con investigación y paciencia es posible.
7. Si bien se pretendía utilizar únicamente software libre, también es cierto que existen herramientas poderosas que pertenecen a empresas privadas, tal es el caso de Wowza Media Server, el cual actualmente es el servidor que provee características de mayor nivel; sin embargo, Red5 es también poderoso, pero por el hecho de ser muy reciente no ha logrado evolucionar lo suficiente.

RECOMENDACIONES

1. Para la transmisión de vídeo de alta calidad con una velocidad de transmisión alta, se recomienda utilizar codecs que optimicen la calidad y espacio a través de formatos de vídeo livianos, esto permitirá que se transmita de mejor forma el *streaming*.
2. Al utilizar Red5 como servidor de medios, es recomendable se trabaje con las fuentes más recientes, las cuales pueden ser descargadas desde el sitio oficial www.red5.com, o bien sincronizar al repositorio de fuentes, de esta manera se podrá tener fuentes con la menor cantidad de bugs, y funciones optimizadas.
3. Es altamente recomendable el uso de objetos de tipo netStream, estos nos dan un manejo eficiente de audio y vídeo sin necesidad de separarlos, lo cual permitirá una sincronización efectiva de ambos sin necesidad de utilizar varios objetos.
4. Cuando se maneja multiroom se recomienda, a nivel de programación, crear una clase que cuente con todos los objetos necesarios para una sala, de esta forma al ser necesario crear una nueva sala, únicamente tenga que ser instanciada la clase, sin necesidad de algo extra.
5. El desarrollo de una plataforma para transmisión de audio y vídeo en Internet puede realizarse con cualquiera de las aplicaciones mencionadas, sin embargo, se recomienda realizar previo al inicio de ésta, un análisis de fortalezas y debilidades de las diferentes

herramientas, con el fin de asegurar que la que sea elegida, sea la idónea para el objetivo final.

BIBLIOGRAFÍA

1. DEMETRIADES C., Gregory. *Building and implementing a complete streaming system*. Indianapolis, Indiana: Wiley, 2003. 359 p. ISBN: 0-471-20950-3.
2. SIMPSON, Wes. *Video over IP*. 2a ed. Inglaterra: Focal press, 2008. 501 p. ISBN: 978-0-240-81084-3.
3. SYMES, Peter. *Video compression demystified*. USA: McGraw-Hill, 2001. 340 p. ISBN: 0-07-136324-6.
4. TOPIC, Michael. *Streaming media demystified*. USA: McGraw-Hill, 2002. 546 p. ISBN: 0-07-140962-9.
5. ZHU, Ce; LI, Yuenan; NIU, Xiamu. *Streaming media architectures, techniques and applications*. USA: Information Science Reference, 2011. 503 p. ISBN: 978-1-61692-833-9.