



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS



IMPLEMENTACIÓN DE UN SERVIDOR DE VIDEO STREAMING Y
SISTEMA MOVIL PARA LA RETRANSMISIÓN DE “UDLA CHANNEL” Y
EVENTOS RELEVANTES



AUTORES

DARÍO CORELLA TITUAÑA - FRANCISCO SEGURA GAIBOR

AÑO

2020



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

IMPLEMENTACIÓN DE UN SERVIDOR DE VIDEO STREAMING Y
SISTEMA MOVIL PARA LA RETRANSMISIÓN DE “UDLA CHANNEL” Y
EVENTOS RELEVANTES

Trabajo de Titulación presentado en conformidad con los requisitos
establecidos para optar por el título de Ingenieros en Redes y
Telecomunicaciones

Profesor Guía

Mg. José Julio Freire Cabrera

Autores:

Darío Armando Corella Tituaña

Francisco Sebastian Segura Gaibor

Año

2020

DECLARACIÓN DEL PROFESOR GUÍA

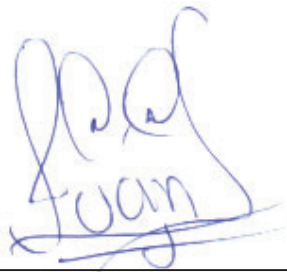
“Declaro haber dirigido el trabajo, Implementación de un servidor de video streaming y sistema móvil para la retransmisión de “UDLA Channel” y eventos relevantes, a través de reuniones periódicas con los estudiantes, Darío Armando Corella Tituaña y Francisco Sebastian Segura Gaibor, en el semestre 202020, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”



José Julio Freire Cabrera
Máster en Gerencia Empresarial
CI: 170973145-7

DECLARACIÓN DEL PROFESOR CORRECTOR

"Declaro haber revisado este trabajo, Implementación de un servidor de video streaming y sistema móvil para la retransmisión de "UDLA Channel" y eventos relevantes, de los estudiantes, Darío Armando Corella Tituaña y Francisco Sebastian Segura Gaibor, en el semestre 202020, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación".



Iván Ricardo Sánchez Salazar
Magister en Calidad, Seguridad y Ambiente
CI: 1803456142

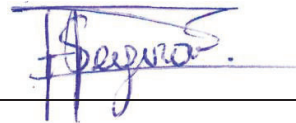
DECLARACIÓN DE AUDITORÍA DEL ESTUDIANTE

"Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes."



Corella Tituaña Darío Armando

C.I.: 2100137674



Francisco Sebastian Segura Gaibor

C.I.:1721045977

AGRADECIMIENTOS

Agradezco en primer lugar a mis padres por su apoyo y esfuerzo incondicional durante toda la vida.

A la universidad de las Américas y todos los docentes que formaron parte de mi formación tanto académica como en valores.

Un especial agradecimiento a mi familia que en todo momento fueron un apoyo incondicional.

Finalmente, a mis amigos y compañeros de clase, quienes fueron parte fundamental de todo este proceso.

RESUMEN

Este trabajo de titulación trata del diseño e implementación de un servidor de video streaming con la finalidad de generar contenido audiovisual y transmitir eventos relevantes hacia Smart TV situados en diferentes puntos.

El servidor será capaz de captar la señal de video generado desde la estación móvil para ser codificada y transmitida al servidor de streaming virtualizado para posteriormente visualizar en televisores Smart TV localizados en diferentes zonas requeridas.

Asimismo, después de ejecutar el video streaming, se utiliza el server como punto RTMP, a diferentes televisores antes mencionados.

El objetivo es optimizar los costos, razón por la cual, se utilizará software libre para cumplir el objetivo.

Para finalizar, se diseñará un sistema móvil, para la realización y emisión de contenido audiovisual, la cual se transmite en vivo a diferentes televisores.

ABSTRACT

The following degree work seeks the design and implementation of a video streaming server with the aim of generated audiovisual content and event relevant transmit to the Smart tv ubicated in any ware place.

The server will be able to capture the video signal generated from the mobile station to be encoded and transmitted to the virtualized streaming server to later display on Smart TVs located in different required areas.

Also, after executing the streaming video, the server is used as the RTMP point, to different televisions mentioned above.

The objective Finally, It will design a mobile system for the production and broadcast of audiovisual content, which is broadcast live to different televisions. is to optimize costs, which is why we will use free software to meet our rason.

ÍNDICE

INTRODUCCIÓN	1
1. Capítulo I. Marco teórico	2
1.1. Introducción Digitalización del Video	2
1.2. Señal de video.....	2
1.2.1 Sistemas de Televisión	3
1.2.2 Creación del video	7
1.2.3 Sensor de imagen.....	7
1.2.4 Medios ópticos	12
1.2.5 Video con Barrido entrelazado VS video con barrido progresivo	13
1.3. Digitalización de Video	14
1.3.1 Recomendación ITU 601.....	15
1.3.2. ¿Por qué comprimir?.....	18
1.3.3. Principios para comprimir	19
1.3.4. Técnicas para comprimir	19
1.3.5. Formatos de video digital y reproductor	21
1.3.6. Modelo de referencia OSI	24
1.3.7. Modelo de referencia TCP/IP	26

1.3.8.	Protocolo de red IPV4	26
1.3.9.	Transmission Control Protocol (TCP).....	28
1.3.10.	User Datagram Protocol (UDP).....	29
1.4.	Video Sobre IP	30
1.4.1.	RTP.....	31
1.4.2.	Video-Streaming	32
1.4.3.	Transmisión de video	35
1.4.4.	Tipos de streaming y tecnologías de trasmisión de video.....	36
1.4.5.	Video sobre redes LAN	38
1.5.	Sistemas Operativos aplicados a servidores	40
1.6.	Virtualización	41
1.7.	Máquinas virtuales	41
1.7.1.	Principales características de las máquinas virtuales.....	42
2.	Capítulo II. Análisis de los requerimientos	43
2.1.	Componentes del servidor de streaming.....	43
2.2.	Software de virtualización	43
2.2.1.	VMware vSphere Enterprise.....	44
2.2.2.	Citrix XenServer Free Edition	45
2.2.3.	Microsoft Hyper-V Server	45

2.2.4. Proxmox.....	46
2.3. Análisis de códecs.....	46
2.4. Análisis de rendimiento (servidor)	48
2.5. Plataforma de video	50
2.6. Características de los códecs para el servicio de video streaming.....	51
2.7. Dimensionamiento de Hardware.	51
2.7.1. Procesador (CPU).....	51
2.7.2. Memoria (RAM).....	52
2.7.3. Ancho de banda (red).....	52
2.7.4. Almacenamiento (disco).....	53
3. Capítulo III. Implementacion del servidor streaming	54
3.1. Diseño de la solución	54
3.2. Instalación de sistema operativo de virtualización Proxmox	56
3.2.1. Creación de una máquina virtual en Proxmox.	60
3.3. Instalación del Mist Server	65
3.3.1. Configuración inicial de MistServer	66
3.3.2. Acceso a la interfaz de configuración Web.....	68

4.	Capítulo IV. Diseño de la solución e implementación de la estación móvil	86
4.1.	Diseño de la solución	86
4.1.1.	Generador móvil de video <i>streaming</i>	86
4.2.	Esquema de la solución	90
4.2.1.	Estación móvil	91
4.2.2.	Computador portátil para sistema móvil	91
4.3.	Implementación del sistema.....	93
4.3.1.	Configuración del emisor de contenidos (OBS Studio).....	94
5.	Capítulo V. Pruebas de funcionamiento.....	95
5.1.	Conexión estación móvil con el servidor	95
5.1.1.	Conexión a red de datos.	95
5.1.2.	PING Request.....	96
5.2.	Calidad de imagen	97
5.3.	Tasa de transferencia de datos	98
5.4.	Tiempo de retardo	99
6.	Capítulo VI. Análisis de factibilidad financiera	101
6.1.	Costos de Implementación	101

6.2. Evaluación Económica.....	102
7. Conclusiones y recomendaciones.....	107
7.1. Conclusiones	107
7.2. Recomendaciones.....	108
Referencias.....	109

INTRODUCCIÓN

Siendo UDLA Channel un medio informativo de alto interés para la comunidad de la Universidad de las Américas y para el público en general, se ve la necesidad de implementar un sistema que permita difundir el contenido del canal dentro de las instalaciones de la universidad considerando que este es un medio que trabaja conjuntamente con carreras como Música, Cine y Multimedia, Gastronomía, Odontología, y en general con todas las carreras de la universidad, el desarrollo de este proyecto busca beneficiar a la institución permitiendo a las carreras publicar su trabajo, haciendo coberturas de los eventos que requieran ser difundidos en la institución.

Por ello se pretende Implementar un servidor de video streaming y un sistema móvil para la retransmisión de contenido de video del canal “UDLA CHANNEL” y eventos relevantes de la institución. Además, analizar protocolos y tecnologías de transmisión de video sobre redes IP y las especificaciones del contenido que se desea retransmitir para la selección de hardware, software y protocolos. Entro otros objetivos se tiene:

- Analizar el levantamiento de situación común del cableado estructurado en ambientes de oficina, hacia cada uno de los monitores existentes y salas comunales de la institución.
- Instalar el sistema operativo y las herramientas de software necesarias para implementación del servidor de streaming de video. Configurar las fuentes del contenido y las salidas o flujos de video que se van a transmitir.
- Diseñar un sistema móvil para generación de audio y video en vivo, que se originen en cualquier lugar.
- Implementar y probar el servidor streaming y el sistema móvil.

1. Capítulo I. Marco teórico

El presente capítulo estudiará la comprensión de audio y video, la digitalización, codificación y protocolos de transmisión de streaming de redes IP. Además, se estudiarán las alternativas de software relacionadas con los servidores de streaming, finalmente, se presentarán características de cada sistema con sus pros y contra en cada caso.

1.1 Introducción Digitalización del Video

Cuando se habla de vídeo digital, se está refiriendo al proceso técnico de digitalización que permite corregir el color, el brillo, obtener un zoom digital, añadir tratamiento, así como también reduce la dimensión de los equipos y refuerza la resistencia a caídas.

Este proceso de digitalización del video se inició con la integración de dispositivos CCD (placas sólidas) los cuales reemplazan a los tubos de vacío que para ese tiempo eran convencionales en sistemas análogos. Al emitir la información, por cable o por onda, la señal ya está digitalizada. Un ejemplo es el de la telefonía que se utiliza en casa (analógico o digital).

Digitalizar la emisión permite corregir errores en la misma y por supuesto incrementa la capacidad de las líneas. Así es como la televisión digital terrestre (TDT) menciona al proceso de transmisión digital de la señal televisiva, la cual se transmite a través de ondas electromagnéticas. (Sendín, 2012, pág. 23)

1.2 Señal de video

“La señal de video se forma con información de brillo, color y sincronismo que genera imágenes en movimiento en una pantalla de televisor, un monitor, etc”. (Joglar, 2009). El video está compuesto de brillo (Sensación de claridad), crominancia (espacios de color de la fotografía y video) y los pulsos que dan sincronismo (llamados barridos de imagen). Estos se definen en colores primarios: azul, rojo y verde (RGB).

“El color se configura por medio de coordenadas circulares (Ver figura 1), la saturación varía de acuerdo a la distancia radial y su ángulo representa tonalidad de color, estos se emplean en los sistemas de televisión”. (Cuenca, 1999)

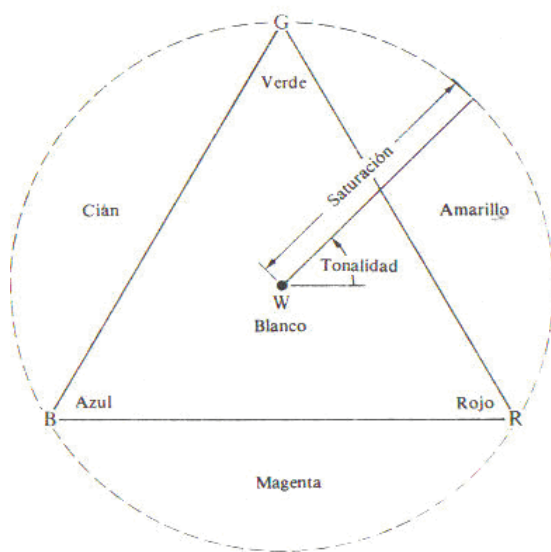


Figura 1. Vector copio de color que define la tonalidad y saturación del color.

Tomado de: (Gualoto, 2008)

1.2.1 Sistemas de Televisión

Los videos analógicos están determinados como el inicio del sistema de televisión que permitieron la recepción y transmisión de la imagen y de las ondas sonoras, actualmente se transmiten a través de redes especializadas que utiliza la radiación de frecuencia. “A nivel global los sistemas de televisión se han estandarizado y

dependiendo del sistema empleado la codificación de la señal se cambia”.
(Pacheco, 2018)

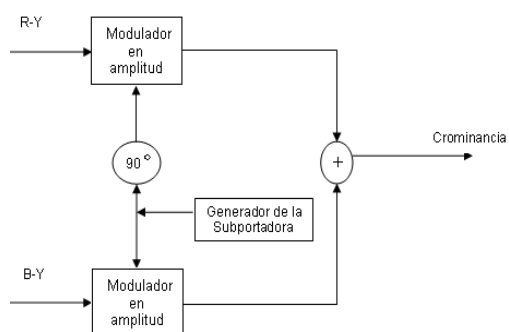
1.2.1.1 NTSC (National Television System Committee)

Tabla 1. Características de NTSC

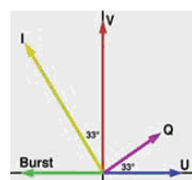
SISTEMA	NTSC
Líneas	525
Campos	60
Frecuencia vertical	60 Hz
Frecuencia horizontal	15.750 KHz
Subportadora de frecuencia de color	3.579545 MHz
Portadora de audio	4.5 MHz
Ancho de banda del video	4.2 MHz

Tomado de: (Gualoto, 2008)

“El comité nacional de sistemas de televisión (NTSC) es el estándar empleado actualmente en Canadá, Estados Unidos, México, Japón y países de América del Sur, entre ellos Ecuador” (Gualoto, 2008).



$$Y = (B - Y)\sin(\omega_c t) + (R - Y)\cos(\omega_c t)$$



Vector de crominancia (R-Y, B-Y)

Figura 2. Vector copio de color que define la tonalidad y saturación del color.

Tomado de: (Gualoto, 2008).

En la figura 3, se determina la modulación en NTSC, se separa la pantalla en 2 campos: a) las líneas que se encuentran de forma horizontal se determinan en pares e impares; y b) representa el número de líneas de sondeo horizontal del campo. “En un segundo pasan 60 campos, el tiempo que corresponde a un campo es 1/60 s o 16.666 ms, cada campo contiene 262.5 líneas, el tiempo por línea es 16.666 ms dividido entre 262.5 líneas, que es igual a 63.5 us, de ahí la frecuencia de exploración horizontal de 15.750 KHz”. (Gualoto, 2008)

El contenido de una imagen a negro y blanco se conforma con en la señal de luminancia, al minimizarla, los componentes RGB obtienen señales de espacios de color de video rojo (V), señales de espacios de color de video azul (U) y señal de espacios de color de video verde.

$$Y = (Cr_R - Y) \cos w_s t + (Cr_B - Y) \sin w_s t \quad \text{ecuacion1.1}$$

En el receptor se detecta:

$$\begin{cases} Cr_R - Y = V \\ Cr_R - Y = U \\ Y \end{cases} \quad \text{ecuacion1.2}$$

En la coordenada de espacios de color de video verde Cr_G se consigue:

$$(Cr_G Y) = -\frac{0.30}{0.59} (Cr_R - Y) - \frac{0.11}{0.59} (Cr_B - Y) \quad \text{ecuacion1.3}$$

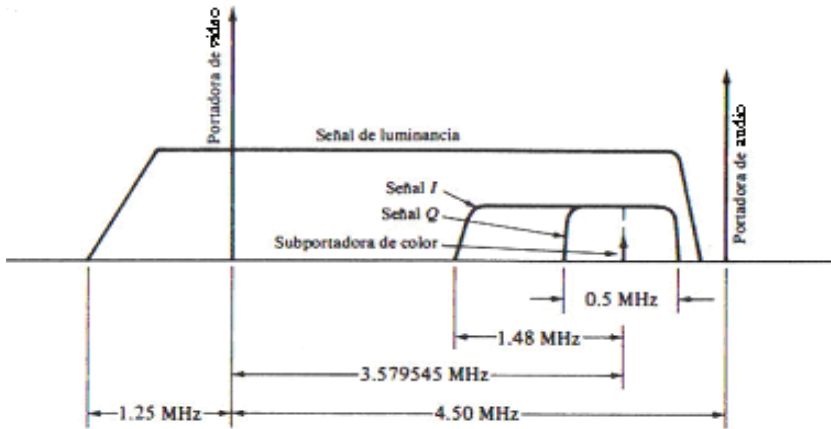


Figura 3. Diagrama espectral de la modulación NTSC

Tomado de: (Gualoto, 2008)

Para obtener (CrG - Y), se debe desarrollar la valoración conocida como “matriciado” componentes de croma en azul y rojo, se localiza en el espacio 6MHz los componentes: video, audio, subportadora de color, en la figura 4 se muestra el diagrama en un analizador de espectros señal TV y NTSC.

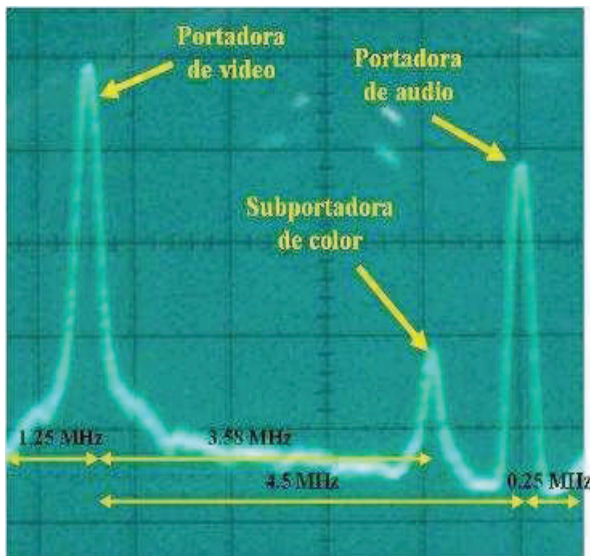


Figura 4. Diagrama en un analizador de espectros.

Tomado de: (Gualoto, 2008)

1.2.2 Creación del video

La creación de un video o fotograma se realiza por medio de la proyección “de medios ópticos en un sensor de imagen dispuesto en el interior de la cámara de video. Las variaciones en la amplitud de las señales son proporcionales a la intensidad de luz acumulada sobre el sensor”. (Gualoto, 2008)

1.2.3 Sensor de imagen

Al sensor de imagen se lo define como un punto de conexión que permite transformar las ondas de luz en corriente eléctrica, se caracteriza por tener un espacio que actúa a la variación de percepción de la luz y forman un proceso basado en puntos que contribuyen a la formación de la imagen. (Cuenca, 1999).

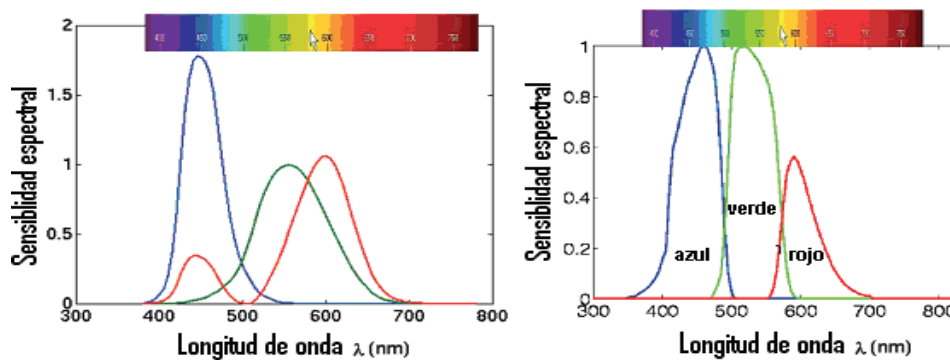


Figura 5. Percepción al ojo del hombre / Percepción de la filmadora.

Tomado de: (Gualoto, 2008)

En tiempos pasados los sensores consistían en tubos al vacío con superficies fotosensible y un generador de electrones con el fin de escanear las imágenes, lo cual por su tamaño dificultaba trasladar las cámaras con facilidad; actualmente, los sensores tienen 2 tecnologías: a) dispositivo de carga CCD, Coupled Device, entre otros. b) semiconductor complementario (óxido metálico).

1.2.3.1 Tecnología CCD

La tecnología CCD, “es un sensor de imagen integrado que permite funciones de almacenamiento y exploración (Scanning)” (Rodríguez, 2018). Permite el arreglo de celdas, cada una acumula carga eléctrica a la luz, esta puede transferir contenido a filas y columnas con el fin de que la imagen se conforme para su procesamiento.

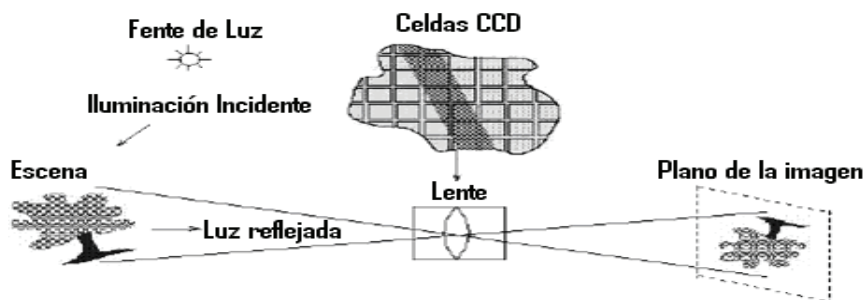


Figura 6. Proceso de captación de una imagen.

Tomado de: (Gualoto, 2008)

1.2.3.2 Tecnología CMOS

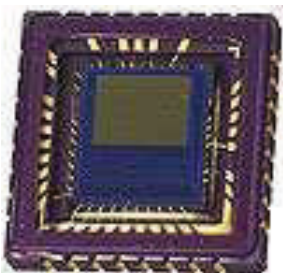


Figura 7. Sensor CMOS.

Tomado de: (Gualoto, 2008)

La tecnología CMOS, son sensores que se utiliza en los chips de memorias (figura 7), permiten soluciones de cámaras económicas en menor tamaño, no son adecuadas para cámaras que exigen una buena calidad de imagen. La peculiaridad fundamental de esta variedad de tecnología es la sensibilidad.

La sensibilidad refiere a la capacidad de captar imágenes en condiciones de poca luz. La sensibilidad mínima en cámaras de luz se da entre 3 y 0,5 lux¹⁷, las cámaras blanca y negro en cambio tiene valores normales de 1 y 0,05 lux. (Rodríguez, 2018).

1.2.3.3 Resolución

La resolución analógica se compone por imágenes que generan rayas horizontales, tienen 256 colores, 16 bits 65.536, 24 bits 16.77.216 y 32 bits 429.4967.296 colores.

La resolución PAL y NTSC, hace referencia a la dimensión del encuadre y a las superficies del video; cuando es una cinta digital se habla de cantidad de pixeles, y el número de pixeles depende de la cantidad de líneas en la TV. La resolución de la norma NTSC, determina que la dimensión máxima de la imagen digital es de 704 horizontal y 480 vertical píxeles; por lo tanto, para la señal en PAL de 704 horizontal y 576 vertical píxeles.

En el siguiente cuadro se muestra los formatos de imagen digital dada en número de pixeles.

Tabla 2. Formatos velocidad y aplicaciones

FORMATO	RESOLUCIÓN NTSC/PAL(Píxeles)
QCIF	176 x 120/144
CIF	352 x 240/288
2CIF	704 x 240/288
4CIF	704 x 480/576

Tomado de: (Campos, 2005)

A continuación, se tiene en la figura 8, la diferencia de relación y aspecto entre los tamaños de video NTSC:

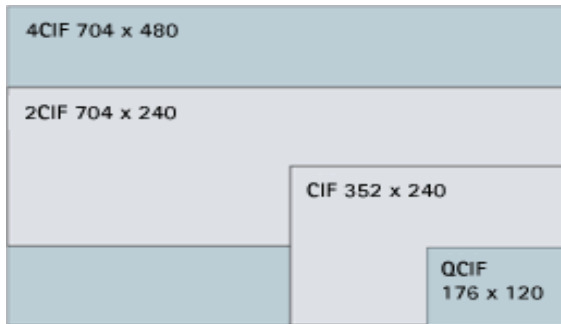


Figura 8. Resolución NTSC

Tomado de: (Gualoto, 2008)

Mientras que, la resolución VGA se utiliza para las cámaras digitales que emplean resoluciones de VGA (640 píxeles x 480 píxeles), o QVGA (320 píxeles x 240 píxeles), y SVGA (1.024 píxeles x 768 píxeles) y de 1.280 píxeles x 960 píxeles cuatro veces VGA la cual ofrece la definición en megapíxeles.

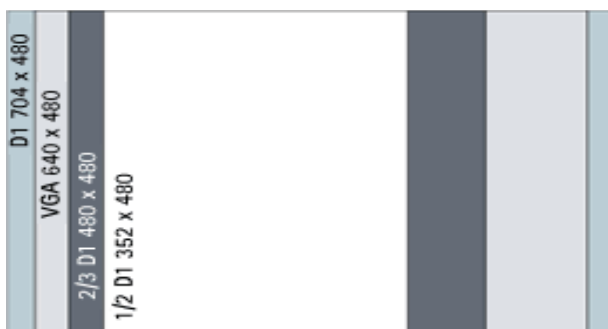


Figura 9. Definición de la imagen calidad Standar.

Tomado de: (Bartolome, 2003)

Entre más alta sea una resolución se mejora la imagen, y se puede observar muchos más detalles, su máxima llega a 400.000 píxeles (704 píxel x 576 píxel = 405.504 píxeles) lo que es igual a 0,4 megapíxeles.

Mientras que la resolución High Definition HD cuenta aproximadamente con 1280 píxeles horizontales y 720 píxeles verticales, se conoce como 720p. Tiene una relación de aspecto 16:9 entre el alto y ancho, siendo la norma utilizada para las futuras pantallas con alta resolución.

La resolución Full HD ofrece 1920 x 1080 píxeles, con un total de 2 730 600 píxeles, lo cual representa una definición de 1080p lo que es mayor a 720p lo que significa una alta definición total o real. “La resolución Full HD es de las más comunes en televisores, monitores y proyectores y cuenta con una relación de aspecto de 16:9.” (Gualoto, 2008).

Otra de las resoluciones importantes es la Ultra HD o Ultra High Definition la cual alcanza un mínimo de 3840 x 2160 píxeles, al igual de las anteriores presenta una relación de 16:9. Al tener estos píxeles, su resolución es 4 veces mejor que la resolución High Definition (1080p).

Finalmente, la resolución 4K refiere a 4096 x 2160 píxeles que conforma el espacio del monitor, cuenta con un total de 8 847 360 píxeles (casi 9 megapíxeles), con el fin de que se usen en las producciones cinematográficas, posee 4 veces más píxeles que 2k.

En la figura 10, se puede ver la diferencia de las resoluciones en alta definición, en futuro será un tipo de resolución 8k. “La 8K recibe esta denominación por el conteo de píxeles dispuesto a lo ancho de la pantalla, los cuales son 7680”. (Bartolome, 2003)

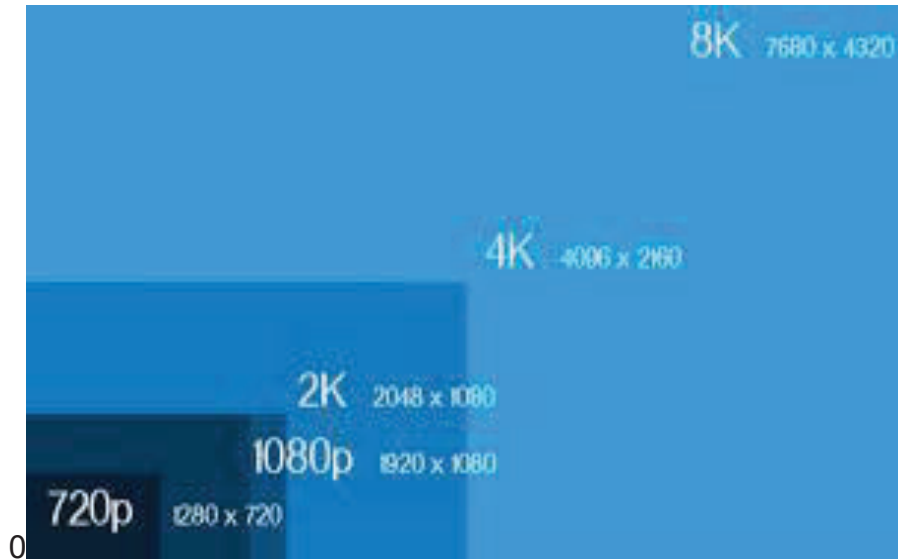


Figura 10. Comparación del tamaño de pantallas por píxel.

Tomado de: (Arrieta, 2020)

1.2.4 Medios ópticos

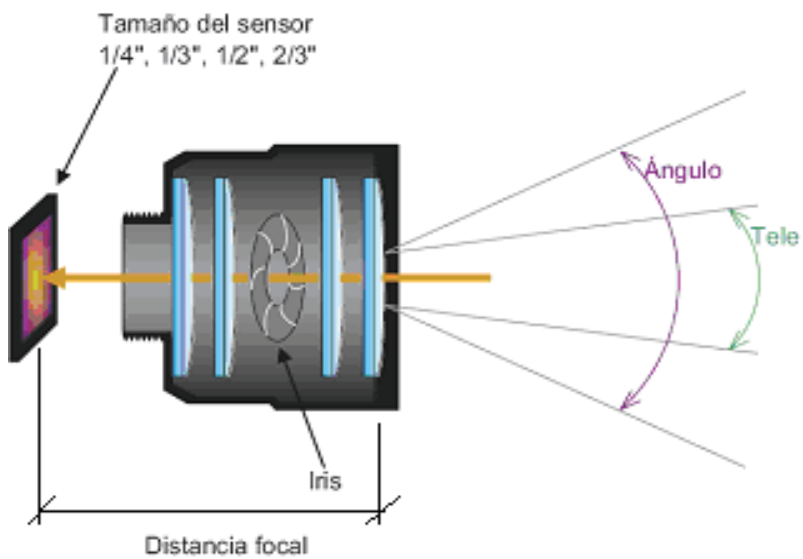


Figura 11. Longitud focal, iris. Sensor de imagen.

Tomado de: (Gualoto, 2008)

Los medios ópticos es parte agregada en las cámaras de video, por lo tanto, es importante definir términos como: Potencia óptica del lente (distancia en milímetros del lente), Iris (controla el nivel de luz), Objetos (aditamentos ópticos, enfoca una mejor imagen).

1.2.5 Video con Barrido entrelazado VS video con barrido progresivo

El movimiento en un vídeo existe gracias al número de imágenes fijas continuas por segundo, para que esto puede ocurrir, se utiliza dos formas: a) interlaced scanning y b) progressive scanning:

a) Video con interlaced scanning



Figura 12. Interlaced Scanning

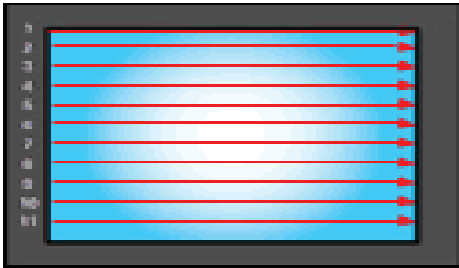
Tomado de: (Perez, s.f.)

El televisor está compuesto por líneas horizontales espaciados al ancho de la pantalla, este se fracciona en dos: a) campo par y b) campo impar. Estas se caracterizan por tener de 30 a 25 fps, dependiendo del sistema NTSC o PAL (figura 12).

b) Video con progressive scanning

En la actualidad y con el pasar del tiempo las ciencias tecnológicas han ido evolucionando con monitores de cristal líquido (LCD), los cuales en la actualidad están basados en transistores de película delgada (TFT22), permitiendo un barrido

progresivo en la imagen de la pantalla. Este barrido toma la muestra entera de línea a línea y no se divide a diferencia del entrelazado. (Figura 13)



Una imagen completa
utilizando barrido
progresivo

Figura 13. Progressive Scanning

Tomado de: (Gualoto, 2008)

1.3 Digitalización de Video

Es un proceso en el video que permite el paso a la multimedia, siendo el vídeo el componente principal, esta consiste en la cuantificación y codificación de la señal.

Se define como “una norma de video digital que no aplica ningún tipo de compresión, razón por la cual los formatos derivados de esta se han mantenido como estándares digitales para el intercambio de información entre equipos y terminales en estudios de grabación y producción de video”. (Aedo, 2014)

1.3.1 Recomendación ITU 601

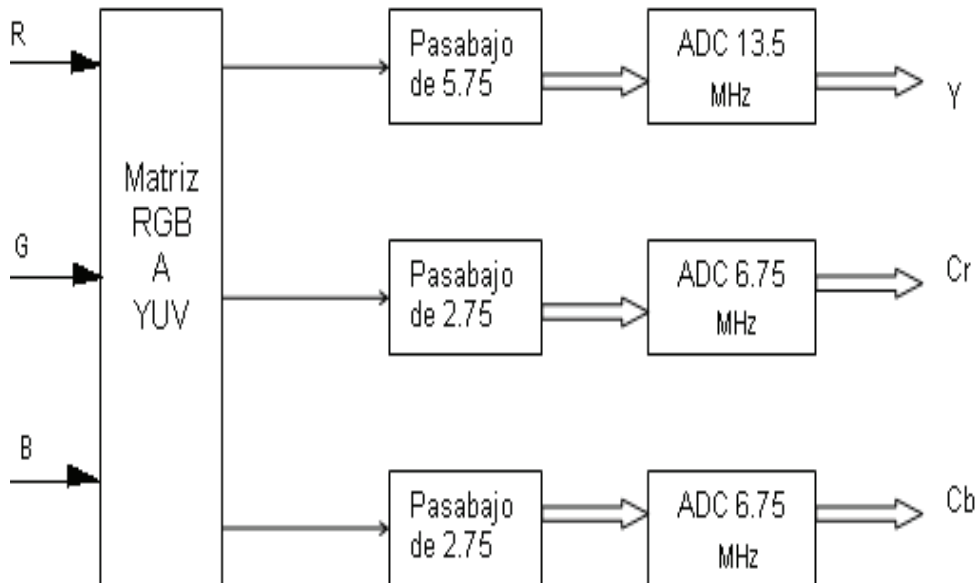


Figura 14. Esquema de obtención de Luminancia (Y), Crominancia (Cr), Tonalidad del color (Cb) en base a RGB.

Tomado de: (Perez, s.f.)

Para obtener una digitalización de video, se utiliza la cantidad de luz o brillo (Y) y la saturación de color (Cr), "en base a las señales RGB, posteriormente se muestrea la señal de luminancia (Y) a 13.5 MHz y la crominancia a 6.75MHz. La luminancia y crominancia se relacionan con RGB, mediante las siguientes relaciones" (Gualoto, 2008):

$$Y = 0,299R + 0,58G + 0,114B$$

$$Cr = 0,713(R - Y)$$

$$Cb = 0,58(B - Y)$$

Donde se emplea 8 bits.

$$Y = 8 * 13.5 = 108Mbps$$

$$Cr = 8 * 6.75 = 54Mbps$$

$$Cb = 8 * 6.75 = 54Mbps$$

Los datos se obtienen de 216 Mbps, los formatos se han dado con forme a la formula (Y, Cr, Cb), donde se tiene:

1.3.1.1. Digitalización 4:4:4

Este tipo de digitalización determina que, por 4 porciones de luminancia, se hallan 4 con croma rojo y azul, según la figura 15 cada cuadro representa píxeles que contiene una porción de luminancia y 2 croma en rojo y azul.

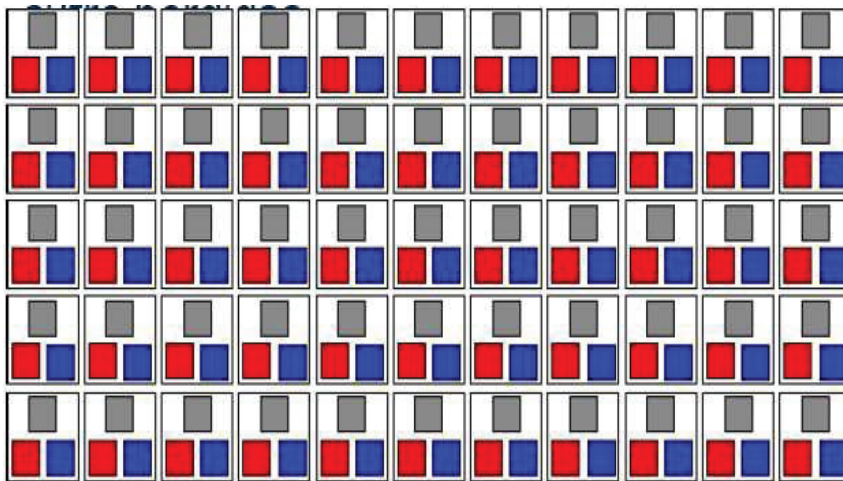


Figura 15. Representación del muestreo 4:4:4.

Tomado de: (Gualoto, 2008)

1.3.1.2. Video Digitalizado en 4:2:2

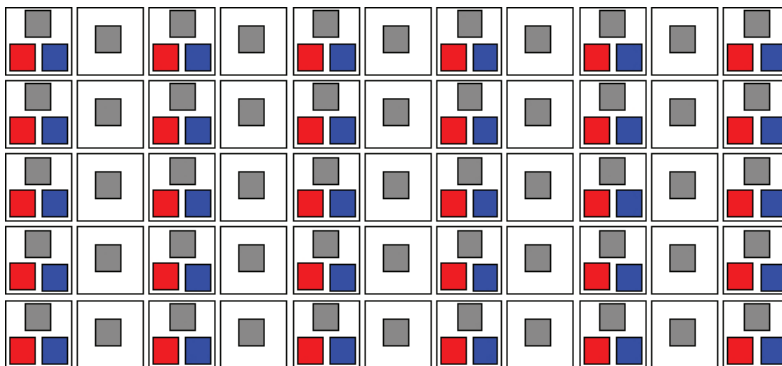


Figura 16. Representación de la muestra 4:2:2

Tomado de: (Gualoto, 2008)

En tanto la representación de digitalización 4:2:2, posee calidad en el video, aprovecha la sensibilidad al brillo del ojo humano más que la coloración, es por lo que la estructura cromática se minimiza a la mitad, cada 4 de luminancia hay 2 de crominancia. (Figura 16).

1.3.1.3. Video Digitalizado en 4:2:0

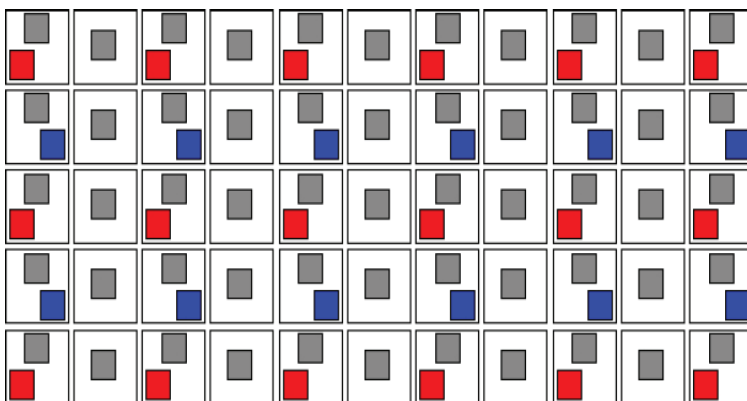


Figura 17. Representación de la muestra 4:2:0

Tomado de: (Gualoto, 2008)

En la figura 17 se determina una representación del muestreo 4:2:0, donde los

colores se exploran en líneas alternas de color rojo y azul, esta información de la cromática se reduce a la cuarta parte en un factor de 2 direcciones: vertical y horizontal.

1.3.1.4. Video Digitalizado en 4:1:1

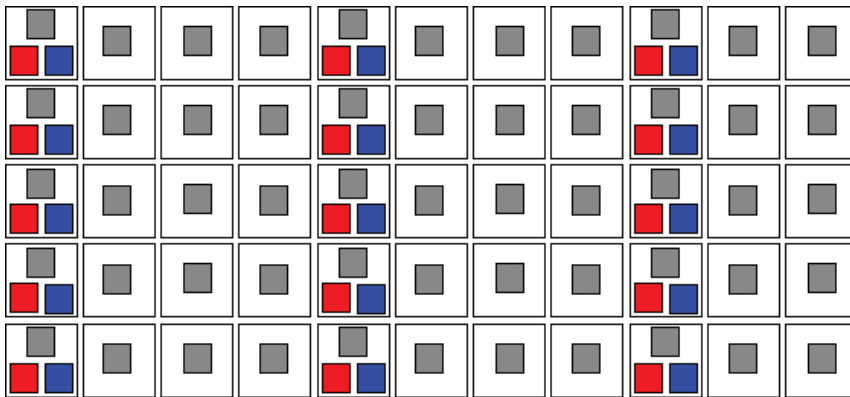


Figura 18. Representación de la muestra 4:1:1

Tomado de: (Gualoto, 2008)

En el presente esquema (figura 18), se determina las muestras de crominancia las mismas que se agrupan 1 cada 4 luminancias horizontales.

1.3.1.5. Video Digitalizado en 4:0:0

Este tipo de estándar suprime la muestra de croma y solo toma la imagen en negro y blanco.

1.3.2. ¿Por qué comprimir?

Es necesario reducir la tasa de bits para la emisión del flujo de video, sin reducir la resolución de la imagen. “Sin compresión no tendría sentido poner imágenes, audio o video en Internet, la calidad de las comunicaciones celulares no sería la misma y desde luego la TV digital no sería posible” (Cuenca, 1999)

1.3.3. Principios para comprimir

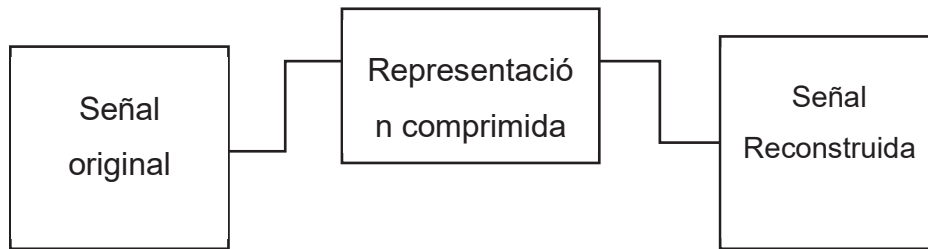


Figura 19. Esquema básico de compresión y descompresión.

Tomado de: (Gualoto, 2008)

En la figura 19, se determina el algoritmo de inicio M la cual forma un contenido comprimido M_c , donde es necesario una menor cantidad de bits, para obtener una representación M_c , generando la muestra original M . Se tiene los siguientes algoritmos de compresión:

- Algoritmos sin pérdida (entrada y salida coinciden)
- Algoritmos con pérdidas (transmite información importante para el ojo humano)

Para que la técnica de compresión sea efectiva, es necesario determinar las dimensiones de la imagen original “dividido para el tamaño que ocupa después de la compresión, a mayor valor, menos ancho de banda se emplea” (Egan, 1991)

1.3.4. Técnicas para comprimir

Las técnicas de compresión o también llamadas tecnologías posibilitadores se definen como algoritmos normados internacionalmente para comprimir secuencias de video sus principales normas son:

1.3.4.1. JPEG

La comprensión de imágenes en JPG se caracteriza por la compresión de imágenes individuales, similitud entre píxeles, la imagen se divide en bloques de 8x8 eliminando componentes de mínima repetición, mientras que los de máxima repetición se codifican.

Al comprimir imágenes en movimiento se usa el mismo principio como cuando se comprime una imagen fija, además de aprovechar el campo repetitivo de las imágenes, este aspecto de compresión se denomina codificación Interframe, sus técnicas se establecen en el contenido de la imagen anterior para obtener la consecutiva ; no obstante, cuando existe un variación en el campo de la imagen se acude a la utilización del codec de imágenes simples. (Pennebaker, 1992)

1.3.4.2. H.261

Este tipo de estándar se utiliza en video conferencia, telemedicina, monitoreo, vigilancia, etc. Utiliza la red digital de servicios integrados (RDSI), la calidad es aceptable ya que hace uso una baja tasa de bit tan solo 64kbps, “cuenta con un mecanismo para controlar la calidad en función del movimiento de la secuencia, cuanto mayor sea el movimiento de las secuencias, menor será su calidad, H.261” (Bartolome, 2003) se determina dos tipos:

- CIF (Formato de intercambio común, 352x288 pixels)
- QCIF (cuarta parte del Formato de intercambio común, 176x144 pixels).

Este sistema aprovecha la redundancia del área y tiempo y es el fundamento de los formatos MPEG.

1.3.4.3. MPEG

El estándar MPEG es un grupo de trabajo de expertos que se formó por la Organización Internacional de Normalización y la Comisión Electrotécnica Internacional para establecer estándares para el audio y la transmisión video, “es uno de los más populares estándares de compresión de video para imágenes en movimiento, se captan imágenes en el tiempo, las imágenes intermedias (imágenes-I) serán calculadas a partir de estas, que se intercalan con imágenes”. (Gualoto, 2008)

- **Imágenes-I:** para codificar un video de esta forma se requiere pocas imágenes, se hace uso uno de 10 o 15 frames, se denomina imagen clave y continuamente llegan a ser el espacio de entrada a la tasa de bytes de la imagen en movimiento.
- **Imágenes-P:** este códec predice la imagen I previa y emplea un componente de resarcimiento de movimiento. El desarrollo del códec detona en la repetición de campos como en función del tiempo.
- **Imágenes-B:** la codificación inicia empleando un frame I ó P previo y el frame I ó P consecuente ejerce como antecedente para la reparación y pronóstico de tendencia, al comprimir estas imágenes logran las proporciones más prominentes, razón por la cual tienden a ser menos pesadas.

Se halla otro modelo de imágenes denominadas imágenes intraframe las cuales tienen similares particularidades como los frames I, pero bajan su definición. Estas son usadas en sistemas que no es requerido una resolución alta.

1.3.5. Formatos de video digital y reproductor

En la actualidad los formatos digitales como la televisión por internet han permitido que se usen otros tipos de estándares, así como códigos. “Las arquitecturas para multimedia incluyen distintos módulos de software que permiten la creación,

almacenamiento, distribución y reproducción de medios digitales, cada arquitectura tiene sus formatos de archivos específicos, para el almacenamiento de la información y de módulos para visualizar distintos tipos de formato”. (Bartolome, 2003)

1.3.5.1. QuickTime (QT)

“Es la arquitectura de Apple para la gestión de datos multimedia, es empleado en aplicaciones informáticas, cámaras digitales, CD's de audio con suplementos multimedia y en la distribución de contenidos multimedia a través de la Web”. (Cuenca, 1999)



Figura 20. Reproductor Quick Time.

Tomado de: (Gualoto, 2008).

1.3.5.2. RealMedia

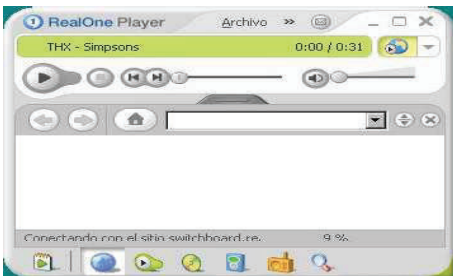


Figura 21. Reproductor Real Player

Tomado de: (Dordoigne, s.f.)

“Es la arquitectura de la compañía RealNetworks para la distribución de audio y video en la Web. Existe un reproductor que se puede instalar como una aplicación cliente o como plugin del navegador”. (Dordoigne, s.f.)

1.3.5.3. DivX



Figura 22. Reproductor DIVX.

Tomado de: (Muñoz, s.f.)

Los formatos de archivos la mayor parte de veces son coligados a una o varios diseños multimedia, permite el uso de archivos con formatos diferentes. A continuación, se detalla formatos más usuales para multimedia.

- MOV: arquitectura QuickTime de Apple.
- AVI: secuencias de video a 15 imágenes por segundo.
- ASF/.WMV. formatos desarrollados por Windows para video, se encuentran más comúnmente en streaming y descarga convencional.
- RM/.RA/.RAM. plataforma de RealNetworks de multimedia en la internet
- MPG/.MPEG/.MP2/.MP3. audio y video.

1.3.6. Modelo de referencia OSI

La diversificación de tecnologías en los primeros años de evolución de las redes hizo que los productos de las diferentes marcas sean incompatibles entre sí, a partir de este precedente la ISO (International Organization for Standardization) estableció un modelo de referencia que permita el desarrollo de dispositivos, aplicaciones o servicios. Este modelo fue llamado OSI (Open System Interconnection) y consta de 7 capas que definen funcionalidades específicas.

El modelo especifica funciones puntuales de cada una de las capas, y cada una presta servicios a la capa de nivel superior, Para enviar información a un dispositivo remoto la información pasa por cada capa del modelo, cada una de las cuales encapsula la información agregando cierta información que permite al dispositivo remoto actuar según la información se añade en forma de cabeceras.

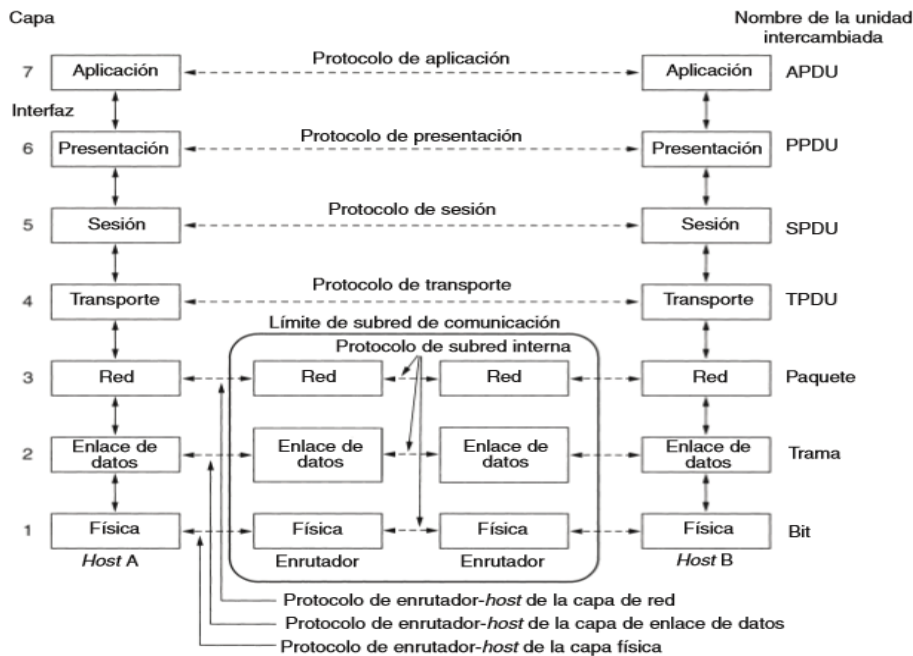


Figura 23. Modelo OSI.

Tomado de: (Tanenbaum, 2003)

Para clarificar el funcionamiento del modelo OSI se pondrá como ejemplo el envío de un paquete desde el host A al host B. El proceso inicia cuando el usuario genera la información en la capa de aplicación esta se encapsula con la información de cabecera correspondiente a esa capa y se entrega a la capa de presentación, la capa de presentación toma esa información y agrega una cabecera donde se agrega información que será leída por la capa de presentación en el otro host, y el proceso se repite de manera similar en cada una de las cámaras hasta que se trasmite por el medio físico, se recibe en el host B y se realiza el proceso inverso, el paquete llega es recibido por la capa física se lee la información de cabecera, se procesa según las instrucciones que constan en la cabecera y se entrega a la capa superior, hasta que llega a la capa de aplicación y el mensaje es presentado al usuario del host B.

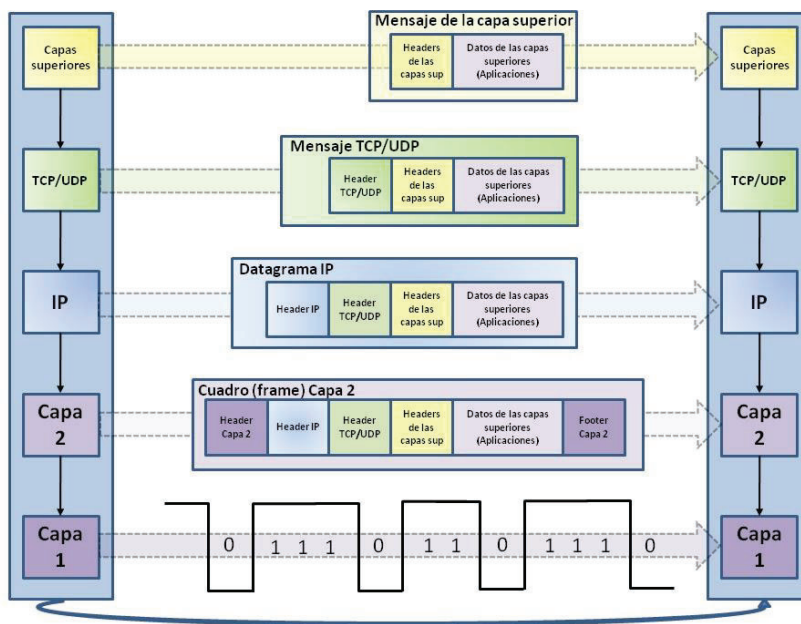


Figura 24. Encapsulamiento modelo OSI.

Tomado de: (Perez, s.f.)

1.3.7. Modelo de referencia TCP/IP

El modelo TCP / IP es el conjunto de protocolos de red que permite la comunicación entre sistemas heterogéneos sin importar la tecnología usada en la capa física.

Hay que notar que el modelo TCP / IP apareció 10 años antes que el modelo OSI y consta de 4 capas ya que agrupando las capas 5, 6, 7 como una sola capa de aplicación y la capa 1 y 2 en una sola, pero básicamente el proceso de comunicación entre capas es el mismo que en el modelo OSI.

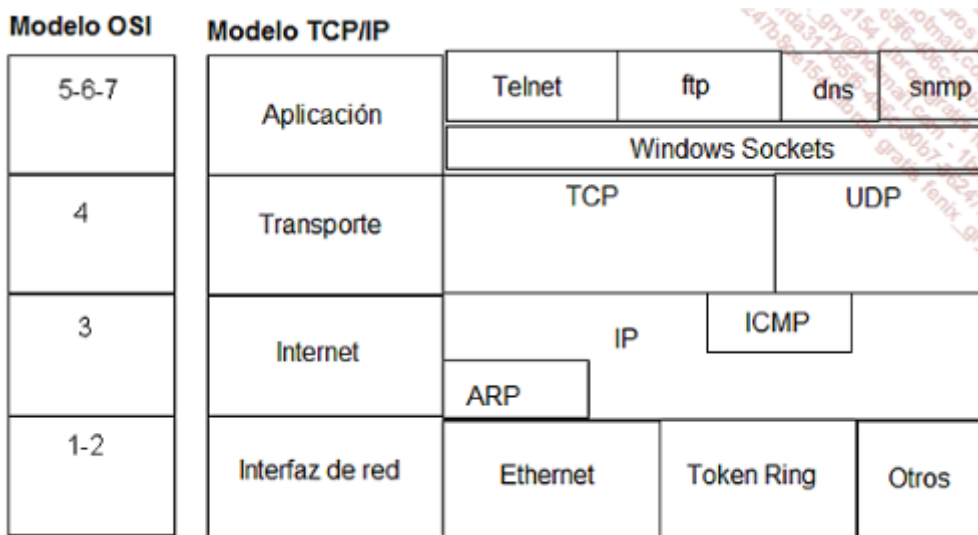


Figura 25. Modelo TCP /IP y OSI.

Tomado de: (Dordoigne, s.f.)

1.3.8. Protocolo de red IPV4

IPV4 se encarga de proporcionar un direccionamiento lógico a todos los hosts conectados a una red de tal forma que se pueda identificar cada equipo, esta dirección es un número binario de 32 bits separado en 4 octetos y se compone de una parte que al host y red.

El protocolo IPV4 permite la comunicación de host dentro de redes locales, así como también permite comunicar entre diferentes segmentos de redes ya que provee de

mecanismos de enrutamiento interredes, esta es la base de lo que se conoce como internet.

1.3.8.1. Direccionamiento IPV4.

IPV4 es un direccionamiento lógico que se asigna a cada host de una red. Su representación es de modo decimal separado por puntos, por ejemplo. 192.168.44.12. una dirección IP puede ser de clase A, B o C.

Clase A, 8 bits de red y 24 bits de host.

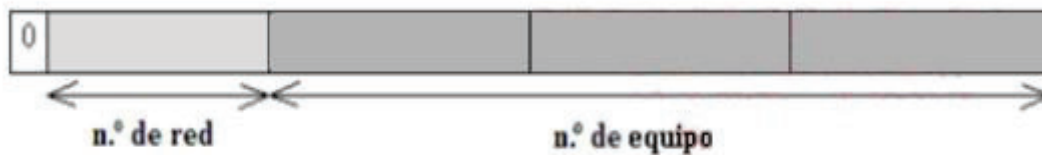


Figura 26. Clase A.

Tomado de: (Dordoigne, s.f.)

Clase B, 16 bits de red y 16 bits de host.



Figura 27. Clase B.

Tomado de: (Dordoigne, s.f.)

Clase C, 24 bits de red y 8 bits de host.

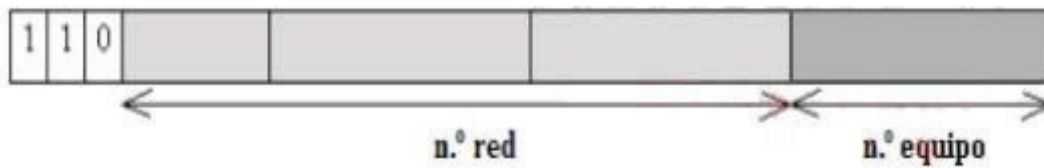


Figura 28. Clase B.

Tomado de: (Dordoigne, s.f.)

a) Mascara de red.

Identifica la parte de una dirección IP que corresponde a la red y la parte de la dirección que identifica al host. Representando en forma binaria los bits representados con el valor 1 indican que pertenecen a la red y con valor 0 que no están asociados a la red, esto se comprenderá mejor con el siguiente ejemplo.

Dirección IP: **192.168.44.12**

Mascara de red: **255.255.255.0**

La misma dirección IP en representación binaria

Dirección IP: 11000000.10100000.00101010.00001100

Mascara de red: 11111111.11111111.11111111.00000000

La parte de la dirección IP que corresponde a los valores 1, de la máscara identifican la red que pertenece, y los valores 0 de la máscara identifican al host.

Protocolos de transporte TCP y UDP

1.3.9. Transmission Control Protocol (TCP)

TCP se encarga de establecer y mantener una sesión entre un cliente y un servidor. El proceso de establecimiento de sesión se da por medio del envío de petición de conexión de un cliente el cual es respondido por el servidor, el cliente recibe la

confirmación y emite su propia confirmación en el servidor quedando establecida la sesión para iniciar una comunicación.

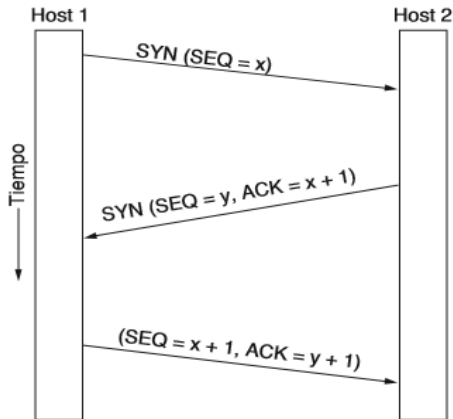


Figura 29. Establecimiento de sesión TCP.

Tomado de: (Tanenbaum, 2003)

Durante el proceso de establecimiento de sesión se establecen parámetros de la comunicación y uno de ellos es la ventana de transmisión que se define como el tamaño máximo de los segmentos de información que serán enviados, por cada segmento recibido se envía un mensaje de confirmación al emisor, si el emisor no recibe la confirmación reenvía la información en una ventana más pequeña con la finalidad de que se asegure que la información llegue a su destino en caso de que se encuentre saturado el canal. A este proceso se lo llama ventana deslizante y es el mecanismo que ofrece confiabilidad al protocolo.

Tras la culminación de una sesión la comunicación termina mediante un mensaje (END) su correspondiente acuse de recibo. De igual forma se puede informar sobre la desconexión de una sesión mediante un mensaje (RST-reset)

1.3.10. User Datagram Protocol (UDP)

Al contrario de TCP, UDP es un protocolo no encaminado a la conexión y prescinde de los métodos de establecimiento de sesión y control de flujo, esto le convierte en

un protocolo que no asegura la entrega de los datos, pero lo convierte en un protocolo ideal para la transmisión de voz y video, elimina la información de cabecera de TCP haciendo que los segmentos sean más ligeros y viajen de forma rápida por las redes IP.

1.4 Video Sobre IP

Las aplicaciones de video sobre IP en la actualidad están muy difundidas a nivel corporativo y residencial con masificación de dispositivos móviles que disponen de cámara de video, sistemas de video vigilancia, y dispositivos de video conferencia y video telefonía que es a donde apunta la tendencia actual.

Según, (Joskowicz, 2013) “es una aplicación típicamente punto a punto, con imágenes del tipo cabeza y hombros, con poco movimiento”. El poco movimiento que genera una llamada de video exige una tasa de transferencia más baja en comparación a video con alto movimiento en la imagen y puede verse como un flujo continuo y uniforme de información.

Al igual que la voz el video necesita ser digitalizado y codificado para que pueda ser transmitido, para ello se dispone de una serie de codecs que ofrecen mayor o menor compresión y calidad de video.

Tabla 3. Códec de video.

Característica	MPEG-1	MPEG-2	MPEG-4	H.264/MPEG-4 Part 10/AVC
Tamaño del macrobloque	16x16	16x16 (frame mode) 16x8	16x16	16x16
Tamaño del bloque	8x8	8x 8	16x16 8x8, 16x8	8x8, 16x8, 8x16, 16x16, 4x8, 8x4, 4x4

Transformada	DCT	DCT	DCT/DWT	4x4 Integer transform
Tamaño de la muestra para aplicar la transformada	8x8	8x8	8x8	4x4
Codificación	VLC	VLC	VLC	VLC, CAVLC, CABAC
Estimación y compensación de movimiento	Si	Si	Si	Si, con hasta 16 MV
Perfiles	No	5 perfiles, varios niveles en cada perfil	8 perfiles, varios niveles en cada perfil	3 perfiles, varios niveles en cada perfil
Tipo de cuadros	I,P,B,D	I,P,B	I,P,B	I,P,B,SI,SP
Ancho de banda	Hasta 1.5 Mbps	2 a 15 Mbps	64 kbps a 2 Mbps	64 kbps a 150 Mbps
Complejidad del codificador	Baja	Media	Media	Alta
Compatibilidad con estándares previos	Si	Si	Si	No

Tomado de: (Joskowicz, 2013)

1.4.1. RTP

RTP es un protocolo de transporte el cual no garantiza la eficacia de servicios en aplicación es en tiempo real tanto en vídeo como voz. Se establece todos los parámetros que permite un flujo de datos confiable de acuerdo a las características del canal de datos.

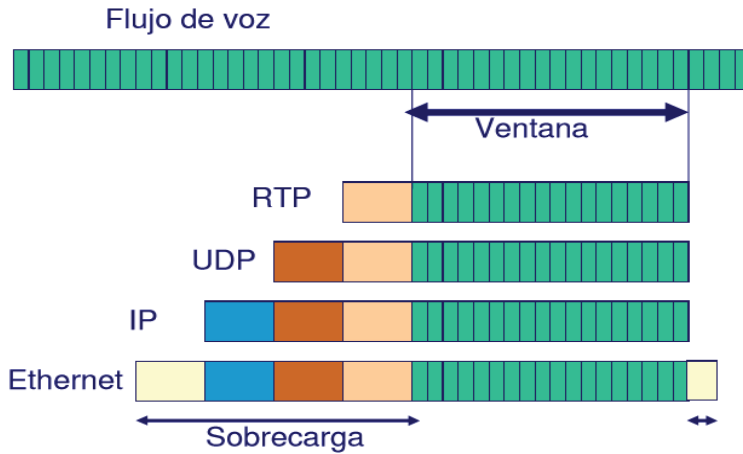


Figura 30. Encapsulado de voz.

Tomado de: (Joskowicz, 2013)

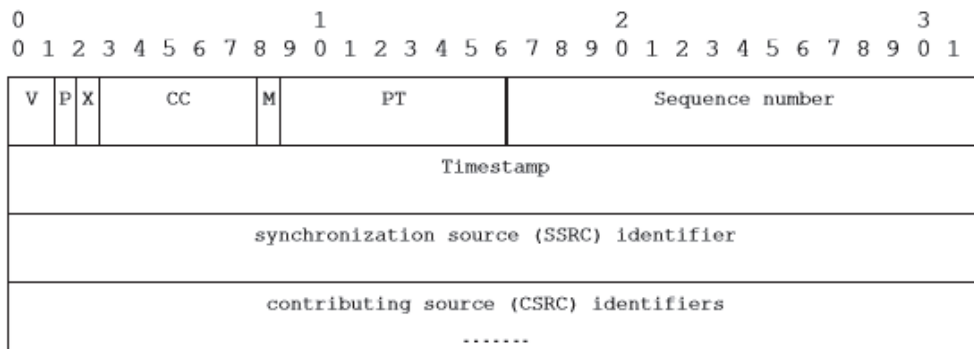


Figura 31. Cabecera RTP.

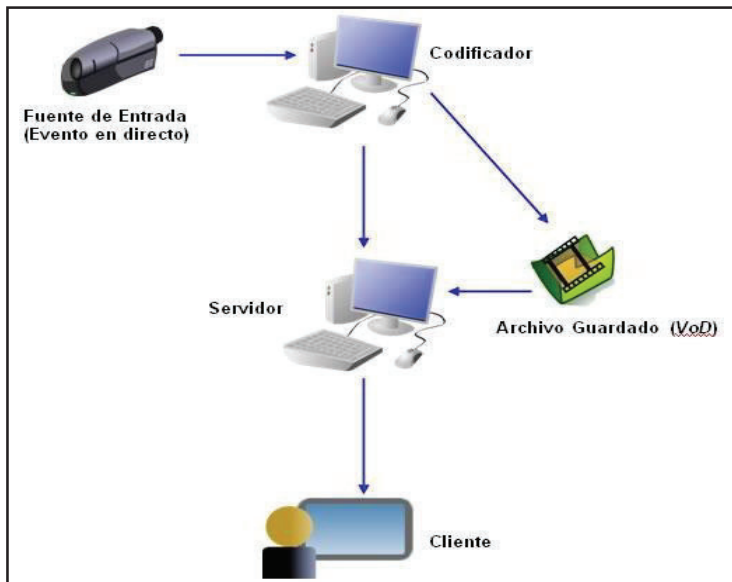
Tomado de: (Joskowicz, 2013)

1.4.2. Video-Streaming

Las transmisiones de vídeo y audio son parte determinante en la entrega de recursos en línea a los usuarios. Este tipo de tecnología trae consigo varias técnicas para permitir la correcta adquisición de conceptos complejos y procesos difíciles. El streaming de video permite ayudar a entender conceptos y procedimientos que son difíciles de explicar simplemente con textos o figuras, añadiendo interacción entre los usuarios y el moderador.

La accesibilidad a este servicio es cada vez mayor y su demanda está en aumento. Con el acceso y el apoyo adecuado, los instructores en línea serán capaces de ayudar a sus estudiantes a comprender y entender los difíciles conceptos que involucren el aprendizaje (Hartsell & Yuen, 2006). En la presente figura se puede apreciar el proceso para disponer del servicio de video-streaming:

Figura 32. Proceso del video-streaming



Tomado de: (Educalab, 2016)

El servicio de streaming en la educación, se puede describir en los siguientes pasos: Primero, el instructor prepara el contenido que será transmitido o grabado con la ayuda de una video cámara o cualquier otro software y hardware especializado, mientras que para el audio se puede utilizar un micrófono externo. Segundo, se debe tener en cuenta los códecs de vídeo, puesto que son una parte importante dentro de este proceso, dado que ayudan a reducir el espacio de almacenamiento y aprovecharlo eficientemente.

Se puede considerar principalmente estos tres factores para el servicio de streaming:

- El formato adecuado

- El servidor de video-streaming
- El ancho de banda (Hartsell & Yuen, 2006)

Existen básicamente tres criterios que conforman la calidad de un video, estos son:

- La velocidad de fotogramas, es decir la cantidad de imágenes por segundo, por ejemplo, 30 o 60 cuadros por segundo.
- La profundidad de color asigna a cada píxel, 24 bits de cuantificación.
- La resolución, la cual depende del número de píxeles, por ejemplo, HD, Full HD, 4K.

Estos tres parámetros pueden afectar primordialmente la calidad de vídeo y tamaño, que será visto por el usuario. A continuación, se muestra una descripción de algunas resoluciones de video:

Tabla 4. Comparación de las tasas de datos para las distintas resoluciones

Resolución	Cuadros por segundo (fps)	Bit rate (Kbps)	Espacio de almacenamiento/por minuto de video
640x480	30	1050	-
320x240	30	235	-
1280x720 (HD)	30 y 60	2350 Y 3000	60 MB
1920x1080 (Full HD)	30 y 60	4300 Y 5800	130 MB Y 375 MB

Tomado de: (Benson; Goldfain; Gupta, 2017)

Como ya se mencionó, el uso de un servidor es un aspecto importante que se debe mencionar, puesto que se necesita un servidor para almacenar el archivo multimedia

y un software que permita la transmisión de los datos por la web. Así, la elección más adecuada de estos elementos es un aspecto muy significativo porque si el servidor no posee los suficientes recursos (ancho de banda), estos recibirán una mala calidad de vídeo y con ello imágenes degradadas, sonidos bruscos e incluso la pérdida de la conexión (Hartsell & Yuen, 2006). La adición de video streaming puede ayudar a mejorar y completar el proceso de aprendizaje de los usuarios, permitiéndoles así fomentar la adquisición de aprendizaje del contenido. El uso de esta herramienta permite sumar autenticidad y realidad al contexto de aprendizaje.

Una imagen en movimiento puede ayudar a visualizar un proceso o ver cómo funciona algo. El video puede tomar información o conocimiento tácito que puede ser demasiado difícil de describir en el texto de un artículo. Con estos beneficios en mente, la transmisión de vídeo es una nueva oportunidad que los educadores poseen para gestionar sus cursos en línea (Castro & Santo, 2010).

1.4.3. Transmisión de video

Existen ciertas limitantes a la hora de emitir video por red, factores como la recepción de imágenes y errores al transmitir, bajo ancho de banda o pérdidas de datos.

El video digital permite manipular flujos de bits más pequeños y con ello una mayor protección contra fallos, se disminuye el efecto de la interferencia en los canales de comunicación, se consigue códecs óptimos, se da paso a la encriptación de los datos y se puede mezclar el video con varias características de contenido en un propio canal.

No existe una sola red tecnológica para ser aplicada adecuadamente en video, ya que depende del medio en el cual se trabaje, se muestra que; al existir una alta tasa de bits disponible, la topología de red adecuada derivaría a MPLS o para una topología LAN se podría utilizar Fast Ethernet o xDSL aunque al presente el interesado ya puede contar con FFTH.

Como ya se mencionó el vídeo sirve como fichero, o en tiempo real, conocida como streaming. El stream asegura al cliente la más alta calidad de vídeo dependiendo de la velocidad. Últimamente el interesado era forzado a detener la reproducción de su video puesto que la red estaba congestionada, y almacenaba en el buffer la información para lograr continuar la vista del video sin detenerse. Actualmente al darse dificultades de atascos de red, el servidor de vídeo actúa disminuyendo la calidad de la imagen que envía para mantener la emisión e ir llenando el buffer sin saturarlo. Al empeorar los escenarios, el servidor deja de emitir frames de vídeo, y conserva la calidad del audio (Castro & Santo, 2010).

1.4.4. Tipos de streaming y tecnologías de transmisión de video

A continuación, se enlista los diferentes tipos de streaming:

- Streaming tradicional.
- Streaming alternativo.
- Descarga progresiva.
- HTTP Pseudo-streaming.
- Dynamic Adaptive Streaming over HTTP (DASH)

El que será utilizado en este proyecto será el streaming adaptativo dinámico (DASH, por sus siglas en inglés), la H de DASH significa HTTP. Desplegado por MPEG convertido en norma internacional en noviembre de 2011 y publicado en abril de 2012 como ISO/IEC 23009-1:2012.

DASH incorpora los mejores elementos de adaptación de streaming diseñadas para resolver los problemas de los usuarios al acceder a la secuencia de vídeo como la intermitencia de la señal, la mala calidad de vídeo en condiciones de red cambiantes, y el retardo significativo de vídeo en arranque. El objetivo del streaming adaptativo es modular la tasa binaria del vídeo, en función del estado de la red (Beatriz, 2012).

A continuación, se detallan las ventajas más sobresalientes de este tipo de streaming:

- HTTP ofrece solución robusta.
- Mejoras en el contenido, el video se divide en pequeñas partes (Mariano & Jorge, 2013).

En la tabla 6 se describe los diferentes streaming con más detalle:

Tabla 5. Comparación de los tipos de streaming

TIPOS DE STREAMING	Progressive Download	HTTP Pseudo streaming	DASH
Protocolo de aplicación	HTTP		
Protocolo de transporte	TCP		
Protocolo de red	IP		
Soporte Unicast	Si		
Ante una pérdida de un paquete	Retransmisión del mismo		
Permanencia en cache	Si	Si	Solo en fragmentos de tamaños reducido
Soportado por navegadores web	Si		

Control sobre la transmisión	No	Si	Si
Necesidad de servidores	Servidores web		
Capacidad de adaptación al ancho de banda	No	No	Si

Tomado de: (Mariano, G. & Jorge, O., 2013)

1.4.5. Video sobre redes LAN

El video sobre redes LAN está basado en la arquitectura TCP/IP. Utiliza aplicaciones de internet como: ftp y http. Se realiza una descarga progresiva del contenido, de modo que pueda empezar a reproducir, la utilización de streaming, en el que utilizan protocolos para la emisión de contenido multimedia en tiempo real (RTP) con un control de sesión dinámico (RTSP), con esta elección el cliente para hacer la descarga y visualización no usa al máximo su ancho de banda disponible, más bien sólo se usa la tasa de bits necesario para que la multimedia se visualice en tiempo real, aunque la mayoría de aplicaciones que utilizan la transmisión en tiempo real son las video llamadas y las video conferencias, este tipo de servicios no son los que se están abordando en este proyecto de tesis aunque es importante resaltar sus diferencias (Universidad Politécnica de Valencia, 2016).

El servicio que se estudia en este proyecto consiste en la difusión de contenido multimedia a través de streaming a varios clientes usando el modelo cliente servidor y el protocolo TCP/IP. La técnica de transmisión para el video streaming sobre redes IP es Unicast debido a que esta técnica permite el control de contenido por parte del cliente a diferencia de Multicast que no permite este tipo de control (Universidad Politécnica de Valencia, 2016).

La transmisión Unicast permite que el servidor envíe un stream independiente a cada usuario cuando éste así lo solicite, en cambio en la transmisión Multicast se realiza una copia de los datos para enviarlos a los usuarios, pudiendo sacrificar de esta manera la calidad del video, además cabe resaltar que este tipo de transmisión está principalmente orientada a televisión en donde se puede elegir el canal, pero no el contenido en sí, en cambio en Unicast si existe un control de contenido.

En la figura 33 se muestra una comparación en la transmisión de streams de video Unicast y Multicast, donde en el envío Multicast se envía un único stream mientras que en el unicast se tramiten varios (flujo de datos independiente):

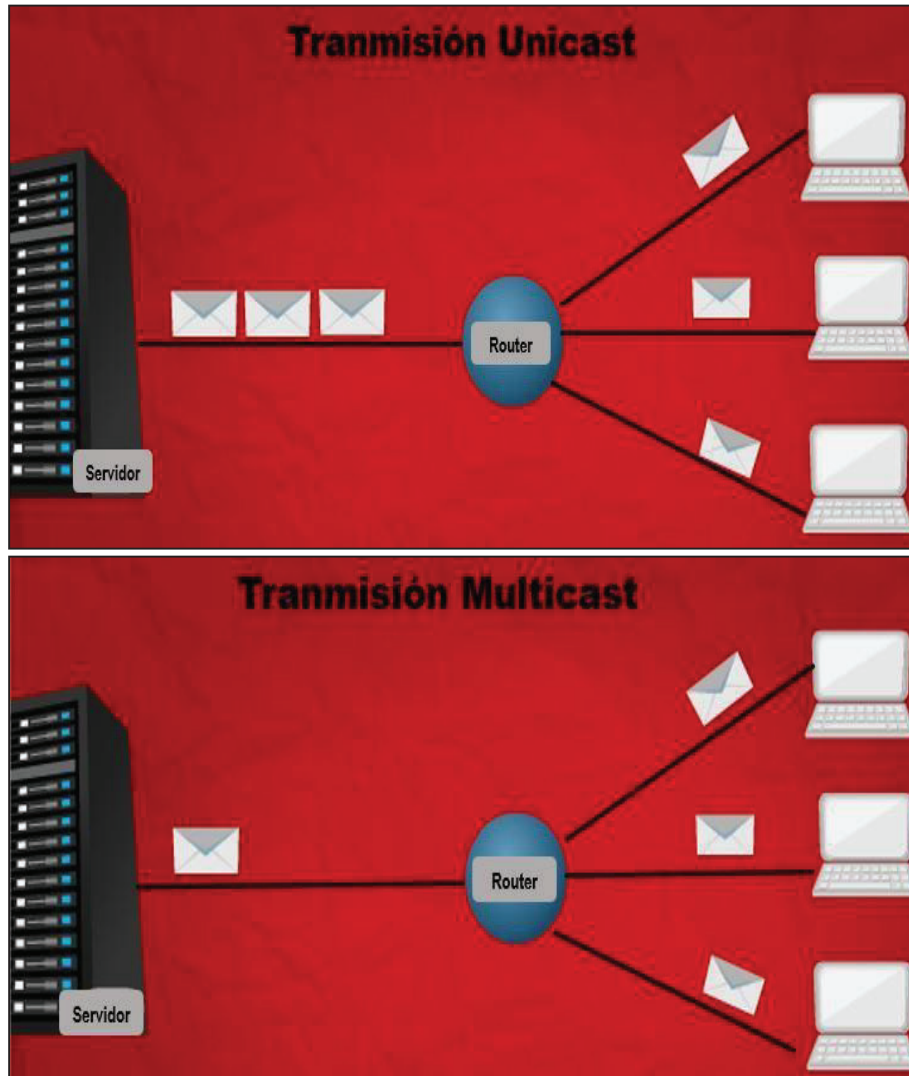


Figura 33. Tranmisión unicast vs multicast

1.5 Sistemas Operativos aplicados a servidores

FreeBSD se define como un sistema libre, cuenta con procesadores Intel 80486 y 80386, versiones DX y SX, además funciona procesadores que son compatibles con Intel como yrix y AMD. Linux es un sistema operativo basado en Unix, representa uno de los principales softwares libre y de código abierto. Se utiliza junto a un empaquetado de software, llamado distribución GNU/Linux y servidores.

Mac OS X Server sistema operativo que se desarrolla en Apple Inc. Incluye herramientas gráficas que permiten la gestión de redes, servicio de red, usuarios LDAP, DNS, servidor Samba y de correo, incorpora herramientas como Wiki, servidor web, Servidor iChat, y otros más.

Microsoft Servers marca que ocupa líneas de productos como: Microsoft Windows y productos dirigidos al mercado. Entre algunas versiones se tiene Windows Sever 2003, Windows Server 2012, Windows Essential Business Server, Windows Home Server, Windows Server 2008 R2, etc.

Unix se determina como un sistema operativo que se desarrolló a principios del 69, sus retribuciones fueron puestos a la venta por AT&T a Novell, Inc. En 1995, se vendió el software a Santa Cruz Operation en 1995, en 2001 la empresa se convirtió en el grupo SCO. En 2010 pasa a ser propiedad de Novell.

1.6 Virtualización

Es una tecnología que ayuda a la creación de servicios TI ligados al hardware, además Red Hat, (s.f). Imaginar que se tiene 3 servidores con propósitos individuales como: servicio de correo, aplicaciones heredadas y un servidor web, cada uno utiliza el 30% de capacidad, una parte de su potencial actual, pero como las aplicaciones son heredadas tienen que ser conservadas junto con el tercer servidor.

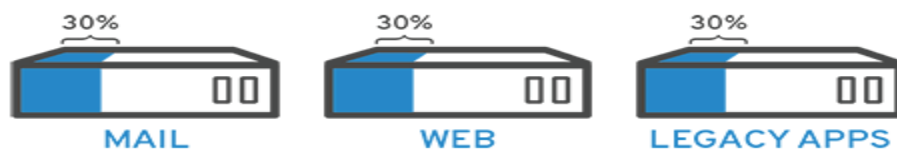


Figura 34. Servidores físicos

1.7 Máquinas virtuales

Las máquinas virtuales son un sistema informático que contiene un sistema operativo y una aplicación, es independiente y se pueden instalar varias máquinas virtuales en un servidor o «host». (Vmware, s. f.), estas con el fin de ejecutar varios sistemas operativos, “es viable arrancar varios sistemas operativos y aplicativos en un solo servidor físico o una capa ligera de software, llamada «hipervisor», desvincula las máquinas virtuales del host y asigna recursos informáticos de forma dinámica a cada máquina virtual según las necesidades” (Bartolome, 2003).

1.7.1. Principales características de las máquinas virtuales

Las principales características de las máquinas virtuales son:

Creación de particiones

Ejecuta sistemas operativos en una máquina física.

Distribuye los recursos del sistema.

Aislamiento

Aísla la seguridad y fallos en el hardware.

Controles avanzados de recursos.

Encapsulación

Guarde el estado completo en archivos.

Transfiere y copia con la misma facilidad que si fueran archivos.

Independencia del hardware

Suministre o migre cualquier máquina virtual a un servidor físico.

2. Capítulo II. Análisis de los requerimientos

En este capítulo se analizarán las características de equipamiento orientado a proveer el mejor resultado posible en la implementación del servidor de streaming. También se analizarán las características de los programas de captura y procesamiento de video en vivo, y transmisión de video en varios protocolos.

2.1 Componentes del servidor de streaming

Para la elaboración de este análisis se necesitarán los siguientes recursos:

- Máquinas Virtuales
- Servidor físico con sistema de virtualización Proxmox
- Software de compresión de video
- Plataforma de streaming MistServer
- Plataforma de streaming Red5
- Reproductor de video VLC
- Aplicación streaming de software libre OBS
- Navegadores Mozilla Firefox y Google Chrome

2.2 Software de virtualización

Se conoce como software de virtualización al conjunto de programas que permiten generar un recurso computacional virtual a partir de un recurso físico que puede ser, capacidad de disco duro, memoria RAM, y procesador. esto se logra creando una capa de abstracción entre la maquina física (host) y la máquina virtual MV.

2.2.1. VMware vSphere Enterprise

Es una compañía que se encarga de la virtualización de PCs como sistemas de servidores, este sistema requiere uso de licencia y entre sus importantes características se tiene:

- Virtualización completa
- Virtualización de hardware asistido
- Migraciones en caliente
- Conversión P2V
- Medidas e informes de rendimiento.
- Control de energía
- Alertas en tiempo real
- Almacenamiento fino
- Restauración y backup de las MV
- Migraciones de MV

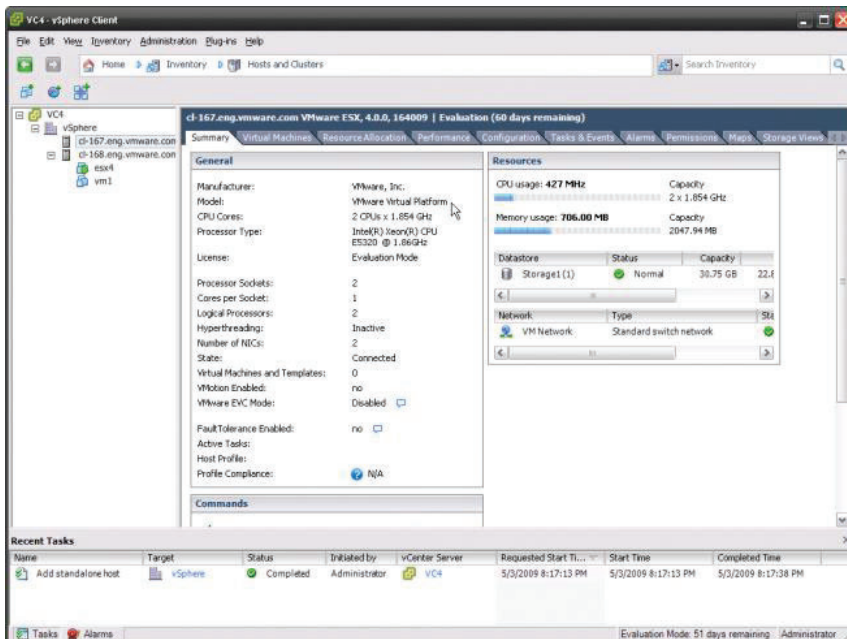


Figura 35. VMware

2.2.2. Citrix XenServer Free Edition

Citrix XenServer Free Edition, es una compañía más de virtualización, cuenta con 2 opciones, la libre y la licencia pagada, lo usan pequeñas y medianas empresas, entre sus características se tiene:

- Virtualización de hardware asistido
- Migración en caliente
- Informes de rendimiento.
- Almacenamiento fino.
- Capacidad de realizar snapshots

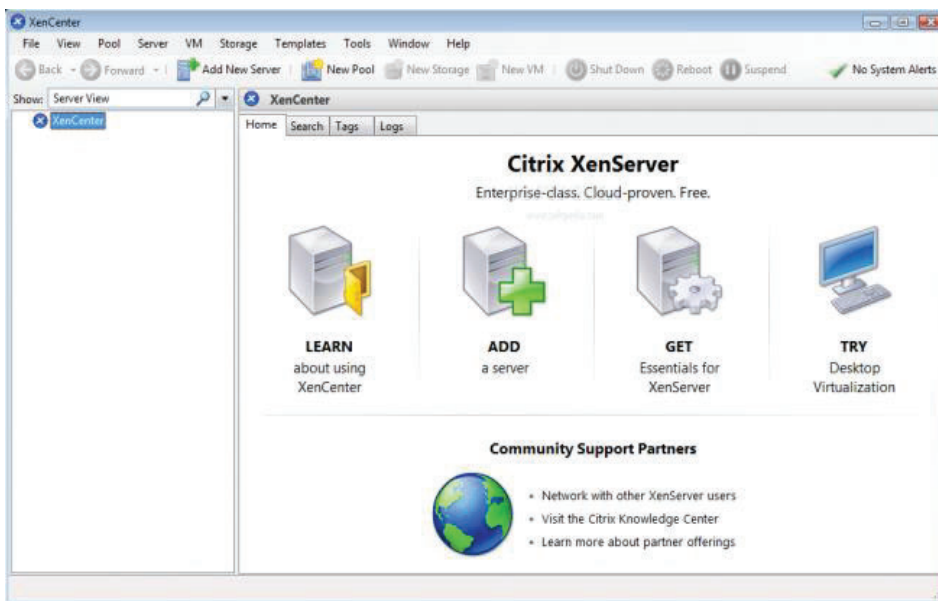


Figura 36. Xen Server

2.2.3. Microsoft Hyper-V Server

Microsoft Hyper-V Server, es un sistema de Microsoft, funciona bajo licencia entre los sistemas más comunes se tiene: "Ubuntu, Suse, RedHat, CentOS y Fedora. Los usuarios que más lo utilizan son las pequeñas y medianas empresas" (Dordoigne, s.f.)



Figura 37. Microsoft Hyper-V

Tomado de: (Arrieta, 2020)

2.2.4. Proxmox

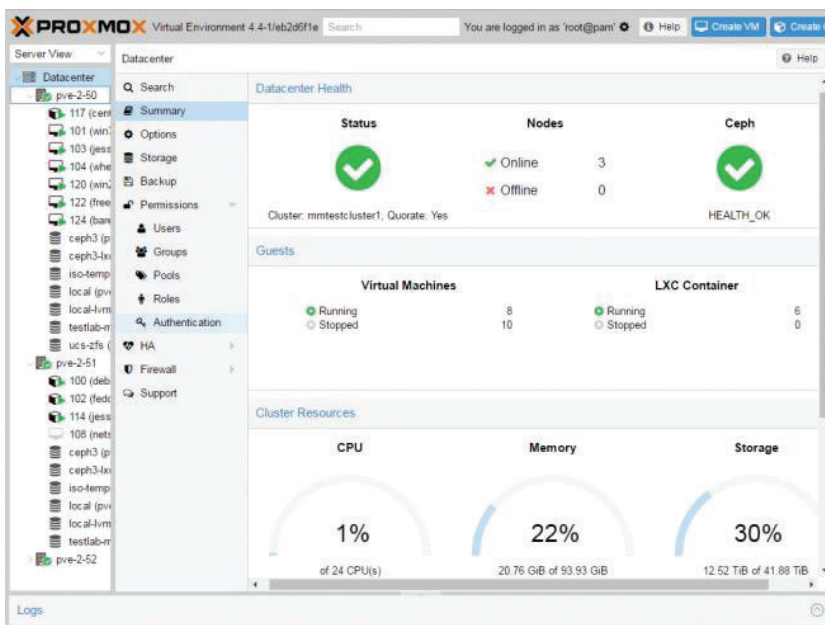


Figura 38. Proxmox

Dado que Proxmox está basada en los sistemas operativos Debian, uno de los más estables y robustos para aplicaciones tanto de usuario final como servidor, sumado a que es una plataforma software libre con una amplia comunidad a nivel mundial, se ha elegido a Proxmox como el sistema de virtualización para el desarrollo del proyecto.

2.3. Análisis de códecs

Muestra. Se tomó en cuenta los siguientes aspectos: la tasa de transferencia de la

información, la tasa de compresión, y el tamaño del archivo comprimido, para lo cual se hizo el análisis correspondiente y pruebas cuyo proceso se menciona a continuación.

Contexto y diseño. El análisis cuantitativo se hizo utilizando softwares de compresión de video tanto para H.264, H.265 y VP9 como también para VP8. Para los softwares empleados no fue necesario pagar ningún tipo de licencia.

Se analizaron dos tipos de videos, unos con fondo estático y otros con fondo en movimiento con la misma persona y con la misma duración, en este caso fue de 1 minuto, esto con el propósito de comparar la tasa de compresión y el tamaño del archivo luego del proceso de compresión. Se consideraron las resoluciones de 640, 720 y 1080 tanto para los videos con fondo estático como los videos con fondo en movimiento.

Instrumentos de medición. - Se utilizaron los siguientes softwares de compresión para según esto determinar cuál es la opción que mejor se ajusta a los requerimientos del canal.

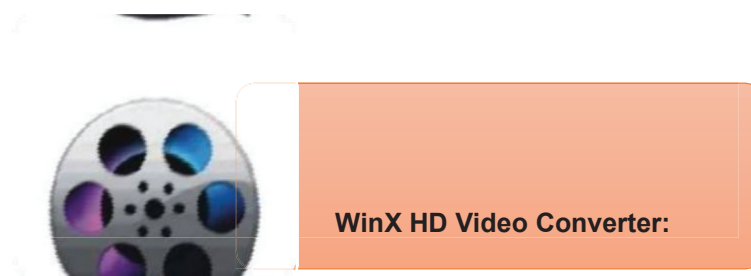


Figura 39. Software de simulación para los códecs

Tomado de: (Arrieta, 2020)

Los programas antes mencionados son para determinar la tasa de compresión y el tamaño final del archivo comprimido, mientras que para el análisis de la tasa de datos necesaria para cada tipo de video se utilizó la plataforma YouTube y para el

estudio del consumo energético se hizo uso del programa monitor de actividad. La plataforma de YouTube, al tener los servidores para los diferentes códecs permite configurar VP9 en sus videos.

La configuración se hizo en el navegador Mozilla Firefox como se muestra en la figura 40 que se muestra a continuación:

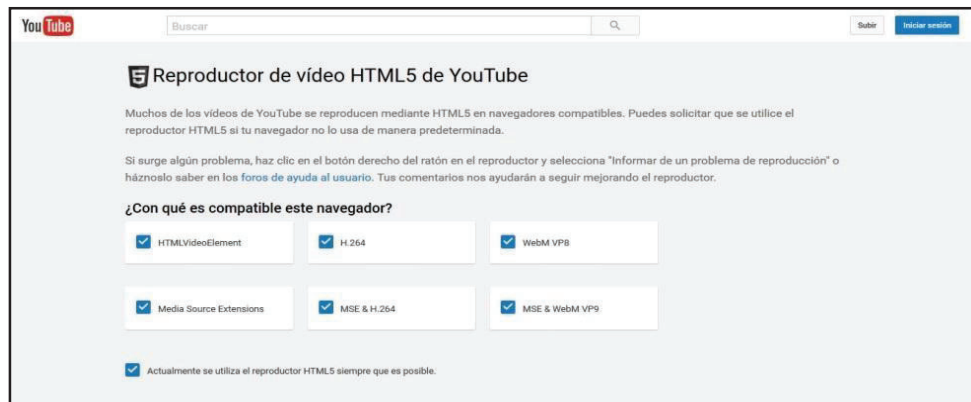


Figura 40. Códec VP9 activado en YouTube

Tomado de: (Arrieta, 2020)

Donde la eficiencia de cada códec estudiado se describe con detalle en el apartado de resultados, mencionando todas las estadísticas estudiadas.

2.4. Análisis de rendimiento (servidor)

Recursos utilizados. - en la parte de conectividad se dispone de un servidor virtual con salida al internet de 40 Mbps simétricos por fibra óptica. El servidor que se utilizará es una máquina virtual con 4 núcleos a 4Ghz, 4 Gigabytes en RAM, corriendo sobre un servidor físico Huawei tecal con dos procesadores de 6 núcleos, 24 GB en RAM y conexiones físicas Giga Ethernet.

Se conoce que un video requiere de 3 a 5 Mbps con resolución media, por lo cual se puede afirmar que la universidad puede brindar el servicio de video streaming sin problema ya que la red consta de puertos de acceso fast y giga ethernet:

$$\frac{\text{capacidad de la red}}{\text{capacidad de video}} = \frac{1\text{Gbps}}{3\text{Mbps}} = 333 \text{ usuarios o streams de video}$$

$$\frac{\text{capacidad de la red}}{\text{capacidad de video}} = \frac{1\text{Gbps}}{5\text{Mbps}} = 200 \text{ usuarios o streams de video}$$

Se considera que en la universidad el número de usuarios esta alrededor de 200 entre televisores y computadores de estudiantes o profesores, en este sentido se obtiene el siguiente análisis:

- Si se considera el 20% de usuarios conectados en una hora pico, se tiene:

Calculo para una tasa de video de 5Mbps

$$\frac{20 \times 200}{100} = 40 \text{ usuarios en una hora pico}$$

Calculo para una tasa de video de 3Mbps

$$\frac{20 \times 333}{100} = 67 \text{ usuarios en una hora pico}$$

- Aun si se considera un 50% de usuarios conectados en una hora pico:

Calculo para una tasa de video de 5Mbps

$$\frac{50 \times 200}{100} = 1000 \text{ usuarios en una hora pico}$$

Calculo para una tasa de video de 3Mbps

$$\frac{50 \times 333}{100} = 167 \text{ usuarios en una hora pico}$$

En resume se puede mantener 167 conexiones concurrentes en alta definición,

estos datos pueden varían en función de la configuración de la calidad de video.

2.5. Plataforma de video

Esta sección hace mención a las alternativas consideradas como posibles soluciones para la plataforma de video.

La primera alternativa considerada fue la arquitectura híbrida de Red5 que permitía un almacenamiento local de archivos de vídeo y en la nube. Tiene una línea muy amplia de productos que es particularmente modular. Está orientado hacia plataformas empresariales, para la educación, telecomunicaciones, entre otros; sin embargo, los servicios de Red5 son más caros que los de la mayoría de otras soluciones.

La segunda alternativa considerada fue MistServer que es afín a muchas aplicaciones, y soporte muchos protocolos de video streaming.

Entre sus características están:

- Versión gratuita: si
- Prueba gratis: si
- Disponible para Windows, Mac y Linux
- Móviles como Android o iOS
- Documentación
- Soporte En línea (de pago)
- Versión licenciada

Para el despliegue del servidor de streaming se hizo uso de un servidor privado virtual (VPS), a continuación de esto se procedió a generar las claves de administrador y configuración de accesos. El acceso al servidor en este caso se hace con una PC que posee sistema operativo Windows, y para entrar al servidor se hace uso de putty y del protocolo SSH para acceso remoto y de esta manera poder operarlo remotamente.

2.6. Características de los códecs para el servicio de video streaming

Hay muchos factores a considerar al seleccionar una solución que adapte a las necesidades de la presente investigación. La tabla 7 resume las características generales de los principales códecs de compresión de video.

Tabla 6. Tabla de análisis de los códec

CÓDEC	Estándar abierto	Adopción	Ahorro de AB	Soporte 720p, 1080p	Soporta Ultra HD
H264	SI	SI	MEDIO	SI	NO
H265	NO	PARCIAL	BAJO	SI	SI
VP8	SI	NO	MEDIO	PARCIAL	NO
VP9	SI	PARCIAL	BAJO	SI	Si

Adaptado de: (Arrieta, 2020)

2.7. Dimensionamiento de Hardware.

Se dividirá el hardware en 4 categorías: CPU, RAM, ancho de banda y almacenamiento. Cada categoría es importante, pero dependiendo de las necesidades, una puede ser más importante que la otra.

2.7.1. Procesador (CPU)

La CPU es obviamente importante ya que maneja todos los cálculos/peticiones del servidor. Tanto en la ejecución del propio software transmisión de video como en el procesamiento de información y requerimientos de los clientes.

Para tener algo tangible, Mist Server indica que empleando el sistema de prueba de procesadores cpubenchmark. Cada 3.4 puntos del registro de CPU que se emplea equivale a un espectador. Por ejemplo, el Intel Xeon E5-2679 v4 a 2.50GHz viene con una marca de CPU de 25236 y será capaz de manejar 7010 espectadores al mismo tiempo.

2.7.2. Memoria (RAM)

La RAM, es un tipo de memoria que almacena los datos de ejecución del sistema operativo, así como los programas que se ejecutan a información que requiera procesamiento rápido. Esta memoria solo almacena información cuando el computador se encuentra encendido, cuando el sistema se apaga se borra la memoria.

Para calcular el uso de la memoria en el peor de los casos se necesita aproximadamente 12,5 MB por cada megabit de ancho de banda del flujo entrante bajo la configuración predeterminada del servidor de streaming. Por lo tanto, obviamente, cuanto más flujos o pistas de streaming, más memoria necesita. Además de esto, se necesitan 2MB constantes de memoria por cada conexión activa (en cualquier dirección).

Así que en 50 flujos entrantes de 2mbps y 600 espectadores se necesita: $12.5 \times 2 \times 50 = 725 \text{ MB} + 2 \times 650 = 2025 \text{ MB}$. Así que aproximadamente 2 GB, ahora recomendaría un margen de seguridad de al menos el 10%, así que ir con al menos 2,2 GB sería prudente.

2.7.3. Ancho de banda (red)

El ancho de banda es a menudo el principal cuello de botella cuando se trata de medios de transmisión, especialmente cuando se utilizan calidades superiores como 4K. El ancho de banda es simplemente la cantidad de tráfico que su servidor puede manejar antes de que la conexión de la red se sature. Una vez que su red se satura significará que los usuarios tendrán que esperar por sus datos, lo que a

menudo lleva a una muy mala experiencia del espectador cuando se trata de medios de comunicación. Así que es definitivamente una de las principales cosas a evitar y por lo tanto necesario para calcular lo que puede manejar.

Para calcularlo se necesita saber la calidad del stream y multiplicarlo por cada conexión (tanto entrante como saliente) para cada stream que tienes o planeas tener y sumarlo todo. Tenga en cuenta que incluso si la calidad de un flujo o el propio flujo no se ve, la conexión entrante seguirá consumiendo ancho de banda de la red si se empuja desde una fuente externa, así que no descuide esos flujos.

Por ejemplo, si tengo 6 streams, uno de 1mbps, dos de 2mbps y 2 de 5mbps con 50 espectadores en el de 1mbps, 300 espectadores en el de 2mbps y 150 espectadores en el de 5mbps tendré que ser capaz de manejar:

$$1Mbps \times (50 + 1) + 2Mbps \times (300 + 2) + 5Mbps(150 + 3) = 158Mbps$$

2.7.4. Almacenamiento (disco)

Para calcular el almacenamiento se debe multiplicar la calidad del flujo de video por el tiempo de duración. Se debe prestar atención a que las calidades de los flujos se miden en bits mientras que el almacenamiento se mide en bytes. Hay 8 bits en un byte, por lo que el almacenamiento necesario es 8 veces menor que el ancho de banda x duración.

Siguiendo el ejemplo del ancho de banda, si se tiene los mismos 6 flujos, uno de 1mbps y dos de 2mbps y 5mbps y quisiera grabar todos ellos durante 20 minutos, necesitaría:

$$\frac{(1 \times 20 \times 60) + (2 \times 2 \times 20 \times 60) + (5 \times 2 \times 20 \times 60)}{8} = 2250Mbps$$

Así que se necesita un poco más de 2GB.

3. Capítulo III. Implementacion del servidor streaming

3.1. Diseño de la solución

Este capítulo tiene la finalidad de describir el proceso de implementación streaming en un entorno virtualizado, y con los recursos necesarios de acuerdo con el análisis de requerimientos.

Es importante señalar que la solución que se presenta es del tipo cliente-servidor, ya que un equipo estará encargado distribuir el contenido multimedia a múltiples clientes a través de una red de datos IP.

El sistema consta de varias etapas que se definen a continuación

Captura. – comprende el proceso de obtención de la información mediante el uso de cámaras de video con salida HDMI, y capturadoras de video que toman esa señal y la transportan a un computador mediante USB.

Codificación. – en este paso la señal de video es procesada en un computador de portátil o de escritorio, el cual mediante un software adecuado procesa la señal de video obtenido por la capturadora y la codifica con códecs y formatos soportados por servidores de streaming, para su distribución en redes IP.

Transporte. – el transporte de video en tiempo real se realiza sobre una red de datos IP, la cual comunica de extremo a extremo al computador encargado de la codificación y el servidor de streaming.

El transporte en IP también se usa para la comunicación desde el servidor de streaming hacia los host clientes, es decir los equipos de reproducción de video.

Terminales de reproducción. – son los terminales que reciben el contenido de video en tiempo real, estos pueden ser de tipo fijo como estaciones de trabajo,

televisores Smart TV, o móviles como computares portátiles o celulares. Estos equipos deben tener el software adecuado para la reproducción de video en red.

En la figura 41 se muestra el diagrama de bloques del diseño propuesto, junto con los elementos constitutivos.

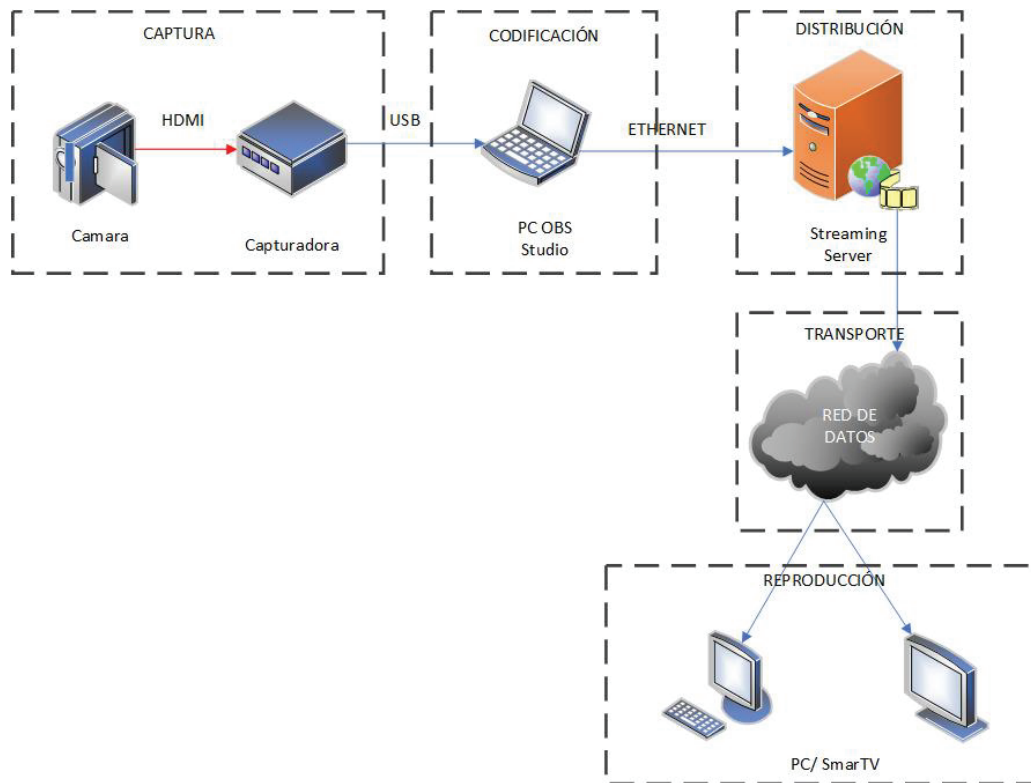


Figura 41. Diagrama de Bloques

El sistema de streaming server al estar basado en IP, requiere que todos los elementos tengan comunicación punto a punto entre sí, para asegurar que todos los protocolos de red de sistema se comuniquen adecuadamente. El sistema, al estar destinado a operar dentro de las instalaciones de la UDLA, se han considerado los campus Granados, Queri, y UDLA Park como una red privada de administración única es decir que están interconectados directamente mediante enlaces de datos de alta capacidad, y que el servicio de streaming podrá ser alcanzado desde cualquier punto de la red como se muestra en la figura 42.

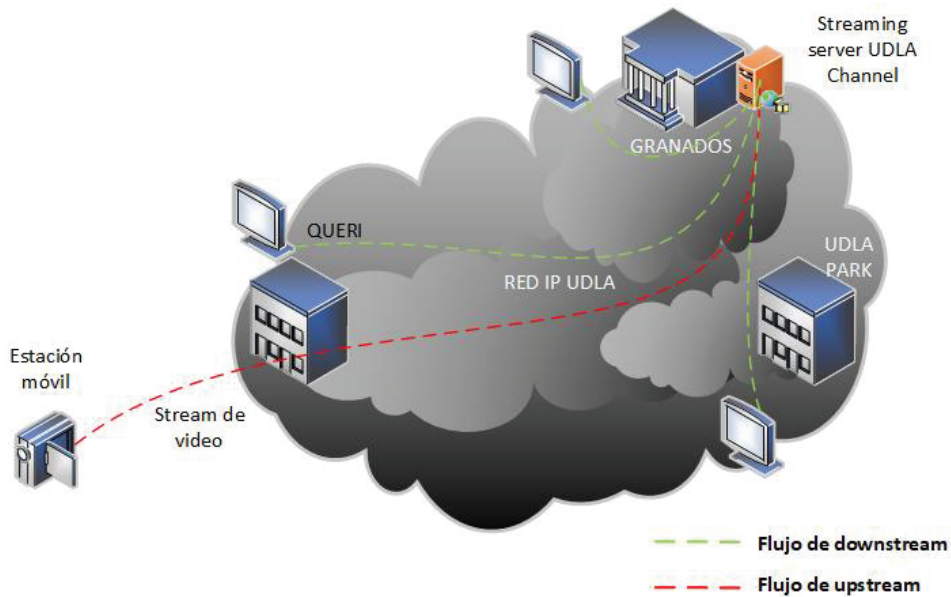


Figura 42. Diagrama de Red

3.2. Instalación de sistema operativo de virtualización Proxmox

El primer paso es grabar una imagen del sistema operativo Proxmox en un DVD para posteriormente arrancar el servidor con el disco de instalación insertado en el lector. Inmediatamente se muestra una ventana con el logotipo del SO el cual indica que se ha iniciado correctamente la instalación.

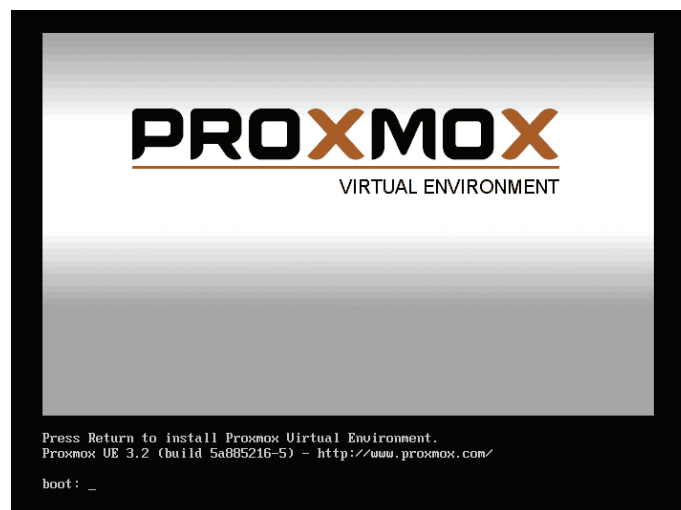


Figura 43. Ventana inicio Proxmox

Tomado de: (Arrieta, 2020)

La siguiente ventana muestra un acuerdo de licencia el cual debe tomar en cuenta para continuar con el proceso de instalación.

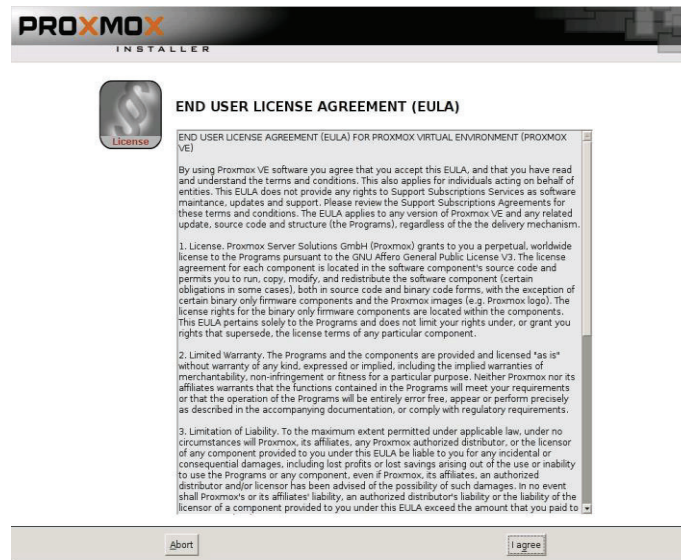


Figura 44. Ventana acuerdo de licencia

Tomado de: (Arrieta, 2020)

El siguiente punto es configurar la zona horaria del servidor, que para el caso de Ecuador es América/Guayaquil -5.

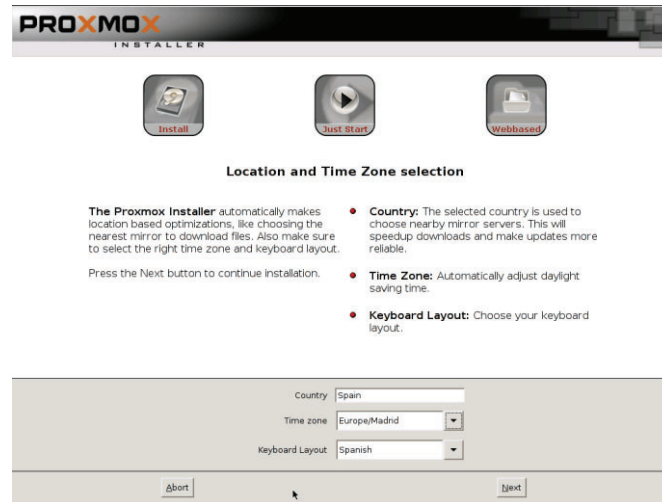


Figura 45. Ventana zona horaria.

Tomado de: (Arrieta, 2020)

El siguiente paso es definir una contraseña de root para el servidor ya que posteriormente se debe acceder a la interfaz de configuración con la contraseña definida.

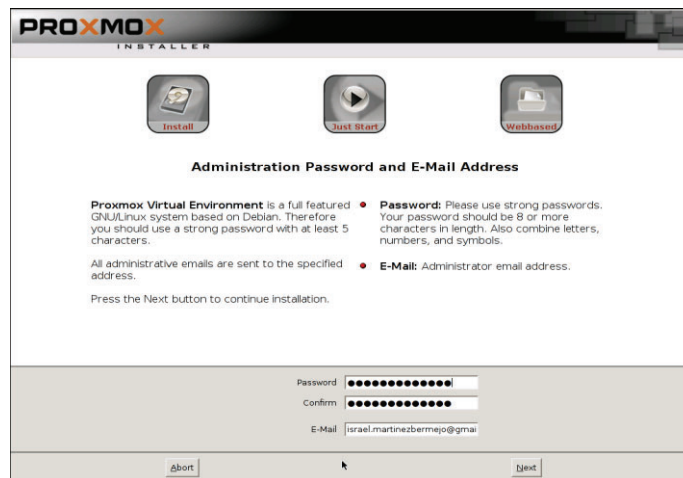


Figura 46. Ventana definición de contraseña

Tomado de: (Arrieta, 2020)

A continuación, se debe definir la dirección IP de gestión para el servidor, así como la puerta de enlace y los DNS.

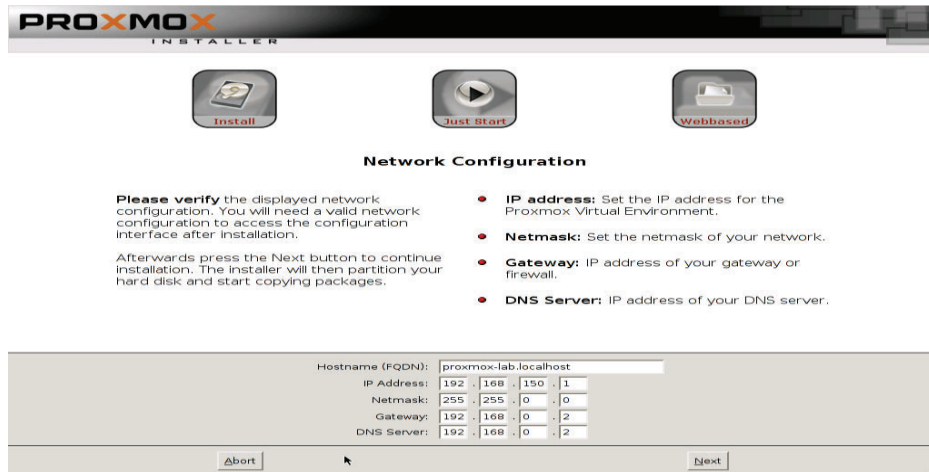


Figura 47. Ventana definiendo IP, Mascara de red, Gateway, DNS

Tomado de: (Arrieta, 2020)

El instalador se encarga de realizar las particiones necesarias en el disco duro, por lo cual durante este último paso transcurrirán alrededor de 10 minutos.

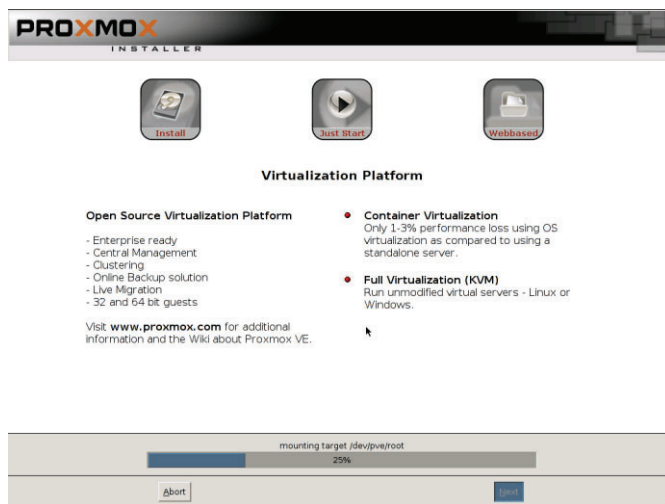


Figura 48. Ventana iniciando particiones

Tomado de: (Arrieta, 2020)

Minutos después, el instalador muestra un mensaje que indica que la instalación ha sido satisfactoria y se deberá reiniciar el servidor con el botón Reboot.

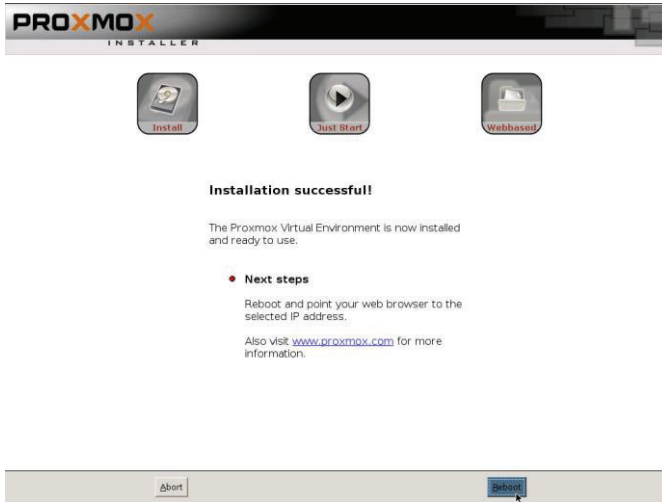


Figura 49. Ventana finalización de instalación.

Tomado de: (Arrieta, 2020)

Finalmente, y si todo ha salido bien se mostrará una pantalla de login al CLI del sistema operativo y adicionalmente se muestra la IP y el puerto para la administración web del servidor.

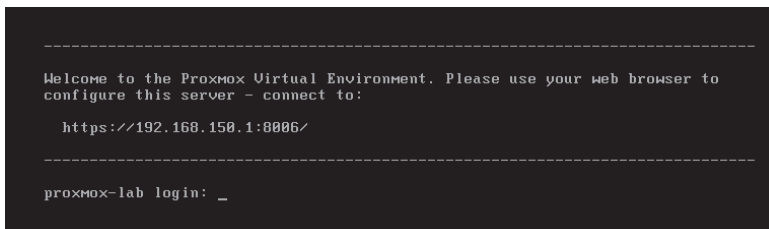


Figura 50. Ventana de acceso después del reboot.

3.2.1. Creación de una máquina virtual en Proxmox.

Como primer paso se creará un contenedor con el sistema operativo que se instalará, en este caso Debian por su estabilidad. Se da click en Create CT, donde se muestra una ventada que detalla los parámetros que siguientes.

Create: LXC Container ✕
 General | Template | Root Disk | CPU | Memory | Network | DNS | Confirm
 Node: Resource Pool:
 CT ID: Password:
 Hostname: Confirm password:
 Unprivileged container: SSH public key:
Load SSH Key File
 Help Advanced Back Next

Figura 51. Ventana creación Promox

A continuación, en la siguiente ventana se elige la imagen del sistema operativo que se desea instalar, esta imagen debe ser previamente descargada en la carpeta de ISOs o “Templates” que el sistema operativo ofrece.

Create: LXC Container ✕
 General | Template | Root Disk | CPU | Memory | Network | DNS | Confirm
 Storage:
 Template:
 Help Advanced Back Next

Figura 52. Ventana de elección de sistema operativo.

El siguiente paso es determinar el tamaño del disco duro de la máquina virtual. 20 GB en este caso ya que será un servidor de streaming dedicado.

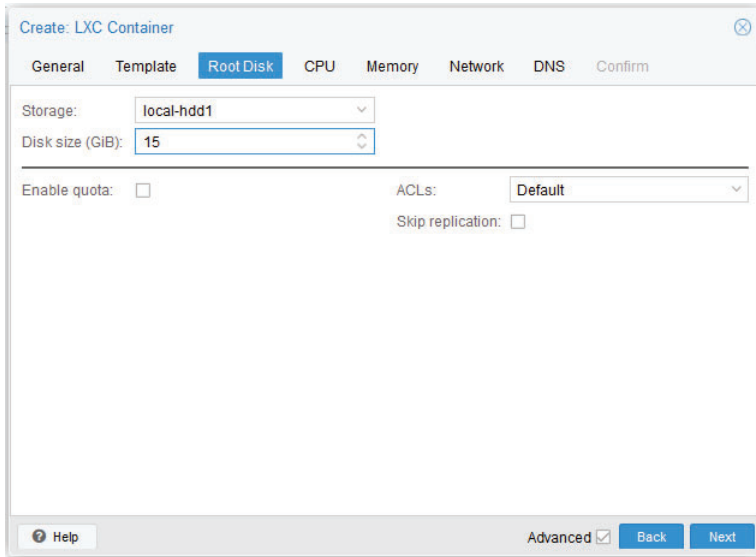


Figura 53. Ventana de Tamaño del disco duro.

A continuación, se detalla el número de núcleos del procesador, se escoge 4 núcleos, ya que garantizan una buena capacidad de procesamiento.

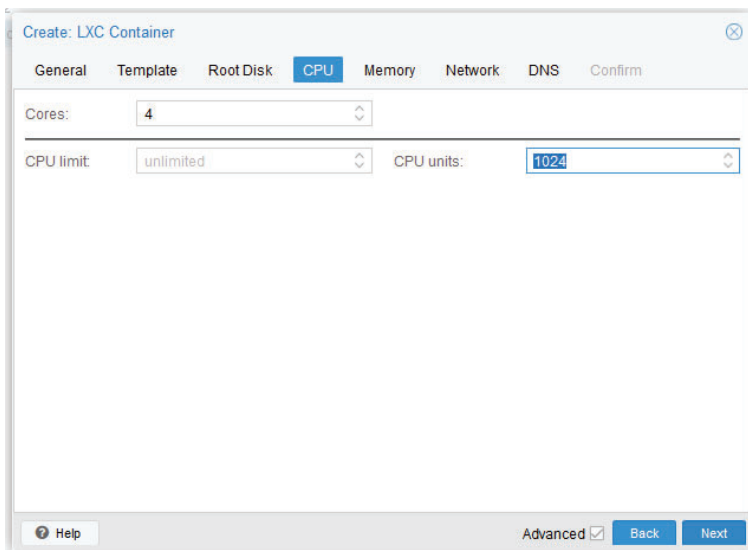


Figura 54. Ventana de selección de números de procesadores del CPU.

Como ya se analizó los parámetros del servidor en el capítulo 2 la memoria RAM para el servidor es suficiente con 4094MB, y el swap que debe ser el doble de la RAM es decir 8192.

The screenshot shows the 'Create: LXC Container' window with the 'Memory' tab selected. The 'Memory (MiB)' field is set to 4096 and the 'Swap (MiB)' field is set to 8192. The window has tabs for General, Template, Root Disk, CPU, Memory, Network, DNS, and Confirm. At the bottom, there are buttons for Help, Advanced (checked), Back, and Next.

Figura 55. Ventana de definicion de la Memoria RAM.

Se continúa con la configuración del direccionamiento IP del servidor de streaming, mascara de subred, puerta de enlace y DNS, para este proyecto se usó el direccionamiento privado: 192.168.100.225 / 24 con puerta de enlace 192.168.100.1.

The screenshot shows the 'Create: LXC Container' window with the 'Network' tab selected. The 'Name (i.e. eth0)' field is set to eth0. The 'IPv4' configuration is set to Static with IP address 192.168.100.226/24 and Gateway (IPv4) 192.168.100.1. The 'IPv6' configuration is set to Static. The 'Bridge' is set to vmbri0 and 'VLAN Tag' is set to no VLAN. The 'Rate limit (MB/s)' is set to unlimited and 'Firewall' is unchecked. At the bottom, there are buttons for Help, Advanced (checked), Back, and Next.

Figura 56. Ventana de configuración de direccionamiento IP.

Finalmente se configura la dirección del servidor DNS, en este caso 8.8.8.8, o en su defecto las direcciones DNS correspondientes al proveedor de servicio.

The screenshot shows the 'Create: LXC Container' window with the 'DNS' tab selected. The 'DNS domain' field contains '8.8.8.8' and the 'DNS servers' field also contains '8.8.8.8'. At the bottom, there is an 'Advanced' checkbox which is checked, and 'Back' and 'Next' buttons.

Figura 57. Ventana de configuración de DNS.

Para finalizar, en la última ventana se muestra todas las características del servidor para confirmar la instalación.

The screenshot shows the 'Create: LXC Container' window with the 'Confirm' tab selected. It displays a summary table of configuration parameters:

Key ↑	Value
cores	4
hostname	streaming
memory	4096
nameserver	8.8.8.8
net0	bridge=vibr0,name=eth0,ip=192.168.100.226/24,gw=192.168.100.1
nodename	pve
ostemplate	local:vztmpl/debian-9.0-standard_9.5-1_amd64.tar.gz
rootfs	local-hdd1:15
searchdomain	8.8.8.8
swap	8192
vmid	103

At the bottom, there is an 'Advanced' checkbox which is checked, and 'Back' and 'Finish' buttons.

Figura 58. Ventana Final de la configuración.

3.3. Instalación del Mist Server

Para instalar el programa de Mist Server se debe descargar el archivo instalador de la página oficial, el archivo correcto será el de extensión **.deb** que es el correspondiente a Debían, O instalar mediante Curl como se muestra en la captura.

```

root@streaming:~# apt install curl
Reading package lists... Done
Building dependency tree... Done
The following additional packages will be installed:
  libcurl3
The following NEW packages will be installed:
  curl libcurl3
0 upgraded, 2 newly installed, 0 to remove and 93 not upgraded.
Need to get 219 kB of archives.
After this operation, 993 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://security.debian.org stretch/updates/main amd64 libcurl3 amd64 7.52.1-5+deb9u10 [292 kB]
Get:2 http://security.debian.org stretch/updates/main amd64 curl amd64 7.52.1-5+deb9u10 [227 kB]
Fetched 519 kB in 0s (572 kB/s)
Selecting previously unselected package libcurl3:amd64.
(Reading database ... 20399 files and directories currently installed.)
Preparing to unpack .../libcurl3_7.52.1-5+deb9u10_amd64.deb ...
Unpacking libcurl3:amd64 (7.52.1-5+deb9u10) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.52.1-5+deb9u10_amd64.deb ...
Unpacking curl (7.52.1-5+deb9u10) ...
Setting up libcurl3:amd64 (7.52.1-5+deb9u10) ...
Processing triggers for libc-bin (2.24-11+deb9u3) ...
Processing triggers for man-db (2.7.6.1-2) ...
Setting up curl (7.52.1-5+deb9u10) ...
root@streaming:~# curl -O - https://releases.mistserver.org/is/2333/f4e229cealb8bdac6c419b8a0c9d695/mistserver_64_Pro1.tar.gz 2>/dev/null | sh
Downloading and installing binaries
./MistAnalyzerDESC
./MistAnalyzerEMML
./MistAnalyzerFLV
./MistAnalyzerE64
./MistAnalyzerHLS
./MistAnalyzerMP4
./MistAnalyzerOGG
./MistAnalyzerRIFF
./MistAnalyzerRMF
./MistAnalyzerRTSP
./MistAnalyzerTS

```

Figura 59. Instalación de Mist Server

El proceso de instalación toma varios minutos ya que descarga e instala todos los paquetes de software necesario secuencialmente. Una vez instalado se muestra un mensaje como el siguiente:

```

detected systemd
Created symlink /etc/systemd/system/multi-user.target.wants/mistserver.service -> /etc/systemd/system/mistserver.service.
Successfully installed MistServer using systemd. Starting MistServer.
root@streaming:~#

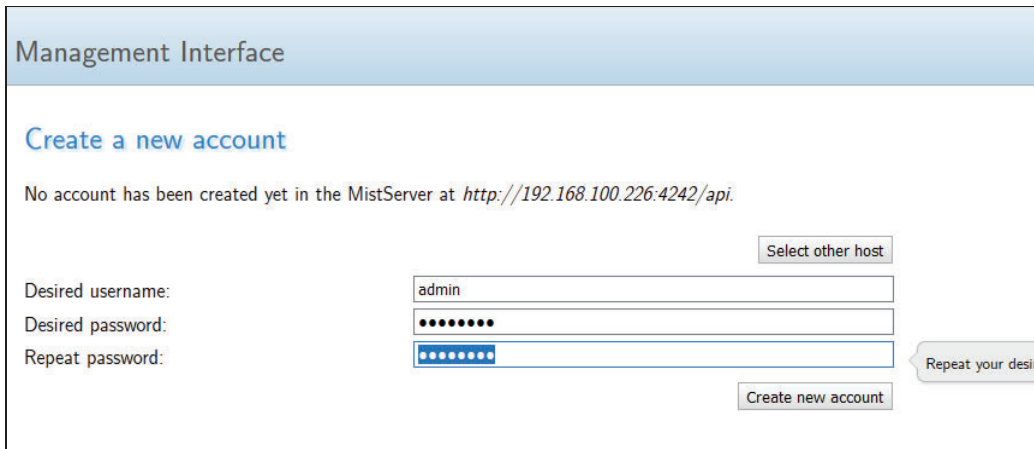
```

Figura 60. Ventana de finalización de instalación

Finalmente, si la instalación fu exitosa se podrá acceder a la gestión del servidor con la siguiente URL 192.168.100.226:4242 o la dirección la se haya definido anteriormente para el servidor y el puerto 4242.

3.3.1. Configuración inicial de MistServer

El primer paso de configuración es la creación de un usuario administrador como se muestra en la captura.



The screenshot shows the 'Management Interface' for MistServer. The main heading is 'Create a new account'. Below this, a message states: 'No account has been created yet in the MistServer at <http://192.168.100.226:4242/api>'. There are three input fields: 'Desired username:' with the value 'admin', 'Desired password:' with masked characters, and 'Repeat password:' with masked characters. A 'Select other host' button is located above the password fields. A 'Repeat your desired password' tooltip is visible over the 'Repeat password:' field. A 'Create new account' button is at the bottom right.

Figura 61. Ventana de creación de Usuario y Password

Se habilita todos los protocolos que van a ser usados en el servidor, lo recomendable es habilitar todos los protocolos soportados así se tiene la seguridad que cualquier tipo de stream funcionará apropiadamente como se muestra en la figura 62.

Finalmente, la opción "Overview" de la interface grafica del servidor muestra el estado de la configuración realizada, así como la información de la licencia y la versión del servidor. Cabe mencionar que la licencia que se entrega a la UDLA es de tipo profesional para uso no comercial. Dispone de todas las características profesionales de software Mist Server, no tiene ningún tipo de restricción, y no tiene caducidad es decir es perpetua como se muestra en la figura 63.

MistServer Management Interface CONNECTED
admin @ http://181.198.186.169:4242/api
Last communication with the server at 18:09:35
Last log entry: 18:05:38 [WARN] Actually seeking to 66, for 0 is not available any more

Protocols

You can find an overview of all the protocols and their relevant information here. You can add, edit or delete protocols.

Protocol	Status	Settings		
DASH (fMP4) over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
DTSC	Active	interface: 0.0.0.0, port: 4200, username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
WebM/MKV over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
Flash progressive over HTTP (FLV)	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
H264 over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
Flash segmented over HTTP (HDS)	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
Apple segmented over HTTP (HLS)	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
Microsoft segmented over HTTP (HSS)	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
HTTP	Active	interface: 0.0.0.0, port: 8080, username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
TS over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
JSON over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
MP3 over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
MP4 over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
OGG over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
RTMP	Active	interface: 0.0.0.0, port: 1935, username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
RTSP	Active	interface: 0.0.0.0, maxsend: 1472, port: 5554, username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
SubRip/WebVTT over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
WAV over HTTP	Enabled	username: root	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

Figura 62. Protocolos del servidor.

MistServer Management Interface

Last log entry: 18:10:35 [ACCS] Session <eea363a769584bb9f26c74b019b8187f> udlastream (HTTP) from 190.152.81.170 ended

Overview

You can find most basic information about your MistServer here.
You can also set the debug level and force a save to the config.json file that MistServer uses to save your settings.

Version: 2.17-Pro Pro_64_2333
Version check: **Your version is up to date.**
Server time: Thu 11 Jun 2020, 18:10:30
Licensed to: 14h

Active products:

Product	Updates until	Use until	Max. simul. instances
MistServer Non-commercial Perpetual	∞	∞	1
MistServer Non-commercial	2020-06-22	2020-06-22	1

[More details](#)

Configured streams: 1
Active streams: 1
Current connections: 1

Enabled protocols: DASH, DTSC, EBML, FLV, H264, HDS, HLS, HSS, HTTP, HTTPTS, JSON, MP3, MP4, OGG, RTMP, RTSP, SRT, WAV
Disabled protocols: HTTPMinimalServer, HTTPS, Push, TS

Recent problems:
20:03:32 Sortset is empty!
20:28:06 onFail 'udlastream': Stream open failed
20:28:06 onFail 'udlastream': Stream open failed
20:28:06 onFail 'udlastream': Stream open failed
20:28:06 onFail 'udlastream': Stream open failed

Human readable name: Udla
Debug level: Default
Access log: Log to MistServer log
Prometheus stats output: Disabled

You can set the amount of information MistServer saves to log. A full reboot of MistServer is required before some changes take effect.

Figura 63. Ventana de estado del servidor

3.3.2. Acceso a la interfaz de configuración Web.

Como primer paso se debe probar la conectividad hacia el servidor de streaming mediante el uso de la consola de Windows y el comando ping a la dirección del servidor, como se muestra en la figura 64. La IP puede variar dependiendo la configuración del servidor.

Ping 192.168.100.226 (cambia la dirección IP en caso de ser diferente)

```
C:\Users\franc>ping 192.168.100.226

Pinging 192.168.100.226 with 32 bytes of data:
Reply from 192.168.100.226: bytes=32 time=14ms TTL=62
Reply from 192.168.100.226: bytes=32 time=186ms TTL=62
Reply from 192.168.100.226: bytes=32 time=12ms TTL=62
Reply from 192.168.100.226: bytes=32 time=19ms TTL=62

Ping statistics for 192.168.100.226:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 12ms, Maximum = 186ms, Average = 57ms

C:\Users\franc>
```

Figura 64. Prueba de ping.

Si la respuesta al ping no es exitosa se debe confirmar el estado del servidor y la conectividad.

En caso de una respuesta exitosa al ping, el próximo paso será acceder a la web GUI con un navegador WEB. para el ejemplo se usa Firefox como se muestra en la figura 65.

Nombre de usuario: admin

Contraseña: Udla2020



192.168.100.226:4242/#&http:// 80%

Management Interface **DISCONNECT**

Login

Please provide your account details.
You were asked to set these when MistController was started for the first time. If you did not yet set any account details, log in with your desired credentials to create a new account.

Host:

Username:

Password:

Figura 65. Ventana de login.

3.3.3. Configuración de un stream en vivo

La configuración de un stream en vivo se realiza accediendo al submenú “stream” de la parte izquierda de la pantalla. En este submenú se encuentran las siguientes opciones:

Switch to thumbnail view: cambia al modo de vista de previsualización.

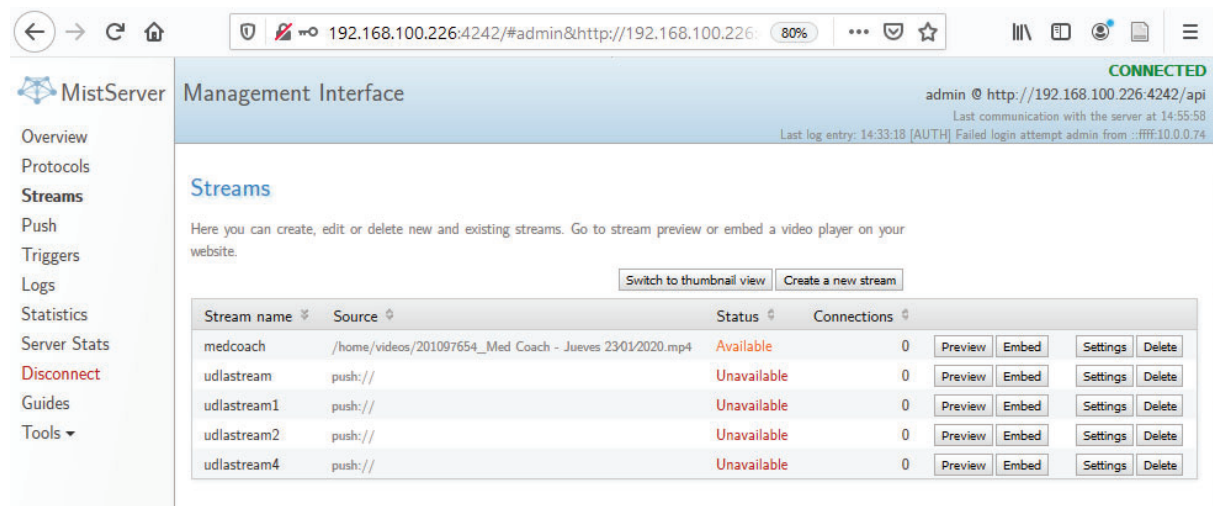
Create new stream: accede a la ventana de configuración de un stream de video.

Preview: muestra una vista previa del stream de video seleccionado.

Embed: muestra las opciones de integración del stream, en páginas o reproductores externos.

Settings: permite acceder a la ventana de configuración de un stream creado previamente.

Delete: borra el stream seleccionado.



The screenshot shows the MistServer Management Interface. The top navigation bar includes a home icon, a search bar with the URL '192.168.100.226:4242/#admin&http://192.168.100.226', a 80% zoom level, and a 'CONNECTED' status. The left sidebar contains a menu with items: Overview, Protocols, Streams (highlighted), Push, Triggers, Logs, Statistics, Server Stats, Disconnect, Guides, and Tools. The main content area is titled 'Streams' and contains the text: 'Here you can create, edit or delete new and existing streams. Go to stream preview or embed a video player on your website.' Below this text are two buttons: 'Switch to thumbnail view' and 'Create a new stream'. A table lists the following streams:

Stream name	Source	Status	Connections	Preview	Embed	Settings	Delete
medcoach	/home/videos/201097654_Med Coach - Jueves 23/01/2020.mp4	Available	0	Preview	Embed	Settings	Delete
udlastream	push://	Unavailable	0	Preview	Embed	Settings	Delete
udlastream1	push://	Unavailable	0	Preview	Embed	Settings	Delete
udlastream2	push://	Unavailable	0	Preview	Embed	Settings	Delete
udlastream4	push://	Unavailable	0	Preview	Embed	Settings	Delete

Figura 66. Opciones del submenú stream.

A continuación, acceder al submenú de creación de un nuevo stream para configurar los parámetros de conexión de un stream en vivo. Los datos que deben configurar son:

Stream name: Nombre de la transmisión el cual debe ser contener solo caracteres alfabéticos en minúsculas y numero sin espacios ni caracteres especiales a excepción de el guion bajo (_) que es el único carácter especial permitido.

Source: especifica el tipo de fuente que contendrá el stream. Para streams e vivo deberá ser **push://**

Stop sessions: al guardar la configuración detendrá las conexiones establecidas hasta ese momento.

Parámetros opcionales

Buffer time (ms): tiempo de buffer disponible para retroceder el stream.

Debug: nivel de registro de sucesos en la transmisión.

Resume support: soporte para reanudar la transmisión desde cuando esta se desconecta.

The screenshot shows the MistServer Management Interface. The browser address bar displays the URL: 192.168.100.226:4242/#admin&http://192.168.100.226:4242/api. The page title is "Management Interface" and it shows a "CONNECTED" status for the user "admin" at the URL "http://192.168.100.226:4242/api". The last communication with the server was at 14:57:43. The page content includes a sidebar with navigation options like Overview, Protocols, Streams, Edit, Push, Triggers, Logs, Statistics, Server Stats, Disconnect, Guides, and Tools. The main content area is titled "New Stream" and contains the following configuration fields:

- Stream name:
- Source:
- Stop sessions:
- Configure your source to push to:
 - RTMP full url:
 - RTMP url:
 - RTMP stream key:
 - RTSP:
- Buffer Input options:

This input type is both used for push- and pull-based streams. It provides a buffer for live media data. The push://[host][:password] style source allows all enabled protocols that support push input to accept a push into MistServer, where you can accept incoming streams from everyone, based on a set password, and/or use hostname/IP whitelisting.
- Optional parameters:
 - Buffer time (ms):
 - Cut time (ms):
 - Debug:
 - Resume support:
 - Segment size (ms):
 - Verimatrix PlayReady Server:
- Encryption:

Figura 67. Opciones de configuración de una transmisión en vivo.

De este menú de configuraciones se debe copiar los parámetros que permitirán la conexión del software de transmisión hacia el servidor, los cuales a continuación resaltan con negrita.

RTMP url: **rtmp://192.168.100.226/live**
RTMP stream key: **udlastream01**

A continuación, se debe configurar la fuente de la transmisión en vivo, en este ejemplo se usa la aplicación OBS studio. La aplicación OBS estudio es un programa de código abierto disponible para Windows, Linux y Mac, se puede descargar de la página oficial www.obsproject.com/es que permite la transmisión de video de varios tipos de fuentes como cámaras web, capturadoras de video, archivos pregrabados, etc.

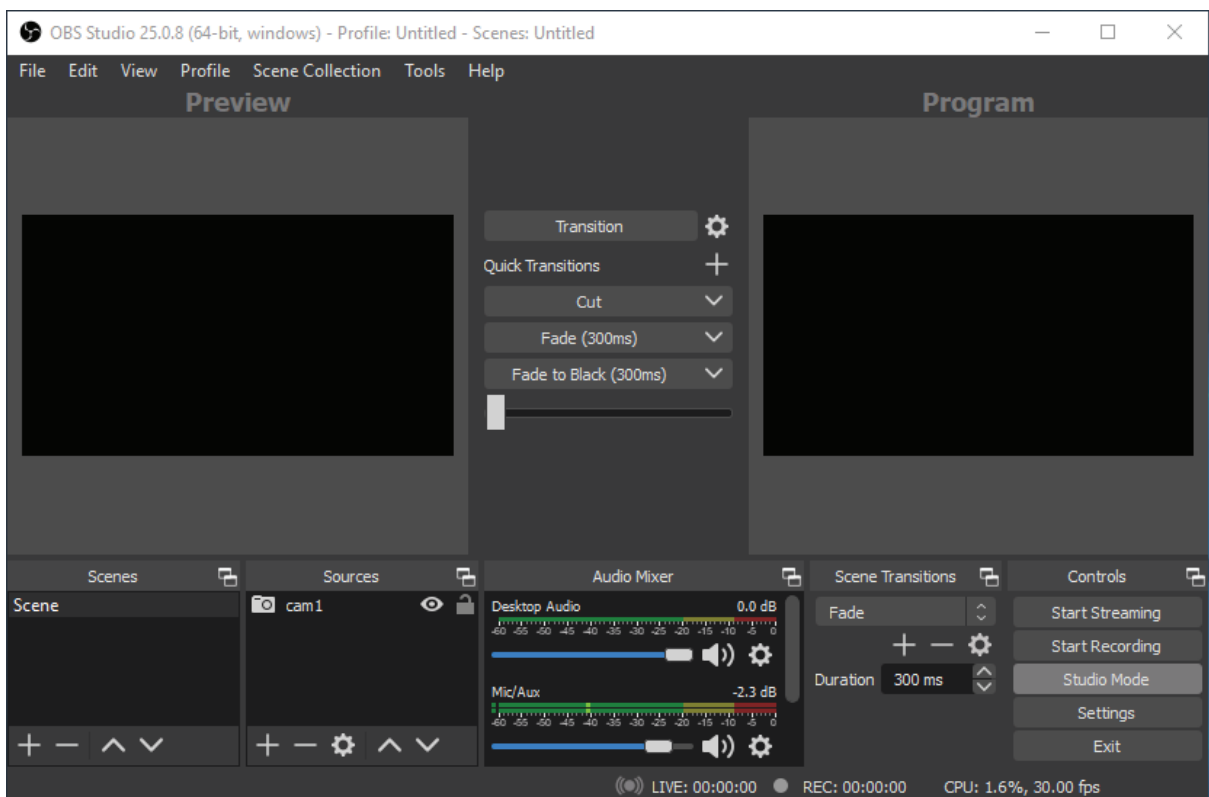


Figura 68. OBS studio.

Dentro del programa OBS studio en la sección inferior izquierda, se debe configurar los parámetros que se habían copiado anteriormente del servidor de streaming (RTMP url y RTMP stream key), además de los parámetros de calidad de video que se muestran en la figura 69.

Stream

Service: Custom

Server: rtmp://192.168.100.226/live

Stream key: udlastream01

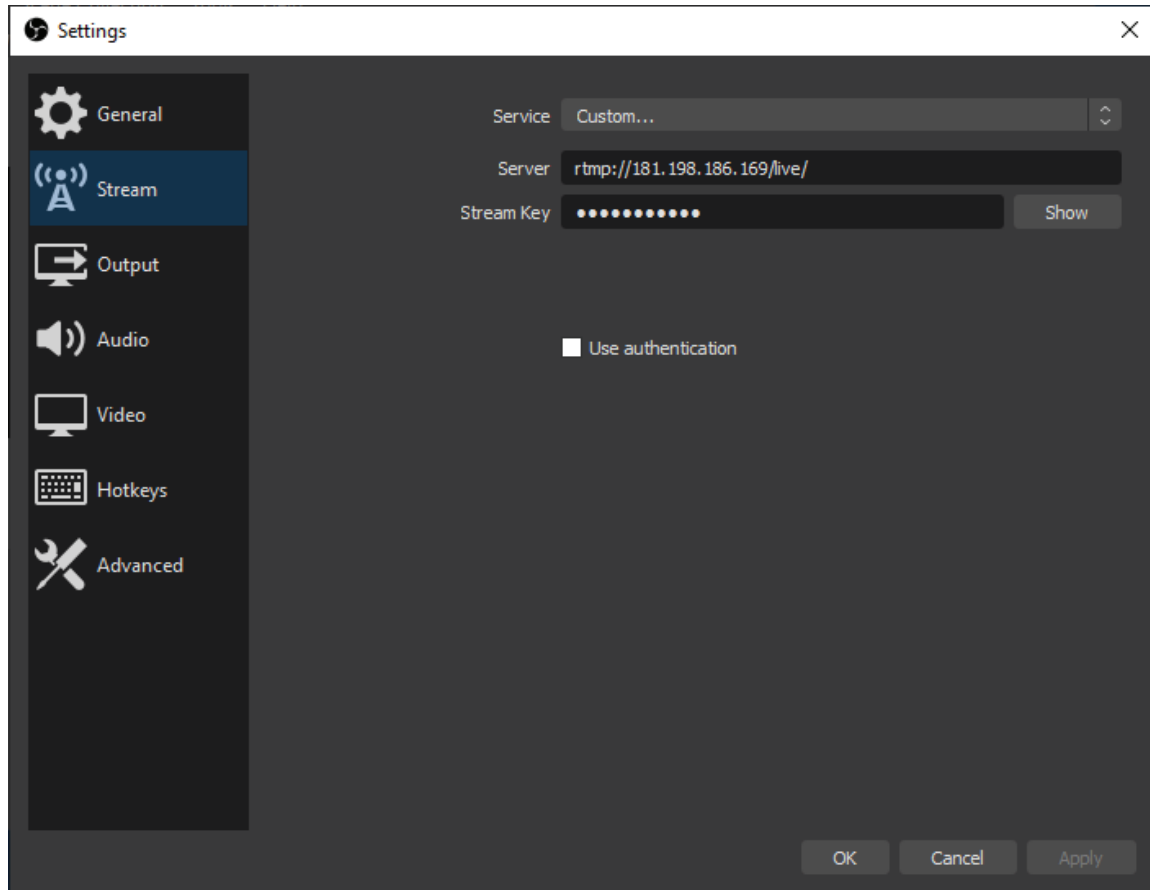


Figura 69. Parámetros del servidor

Output.

En este apartado se configura el tipo de encoder de video, control de ancho de banda y re escalado de resolución entre otras.

Encoder: X.264

Rate Control: CBR (constant bit rate)

Bitrate: 1024 (puede ser mayor o menor dependiendo de la capacidad del canal de comunicación al servidor)

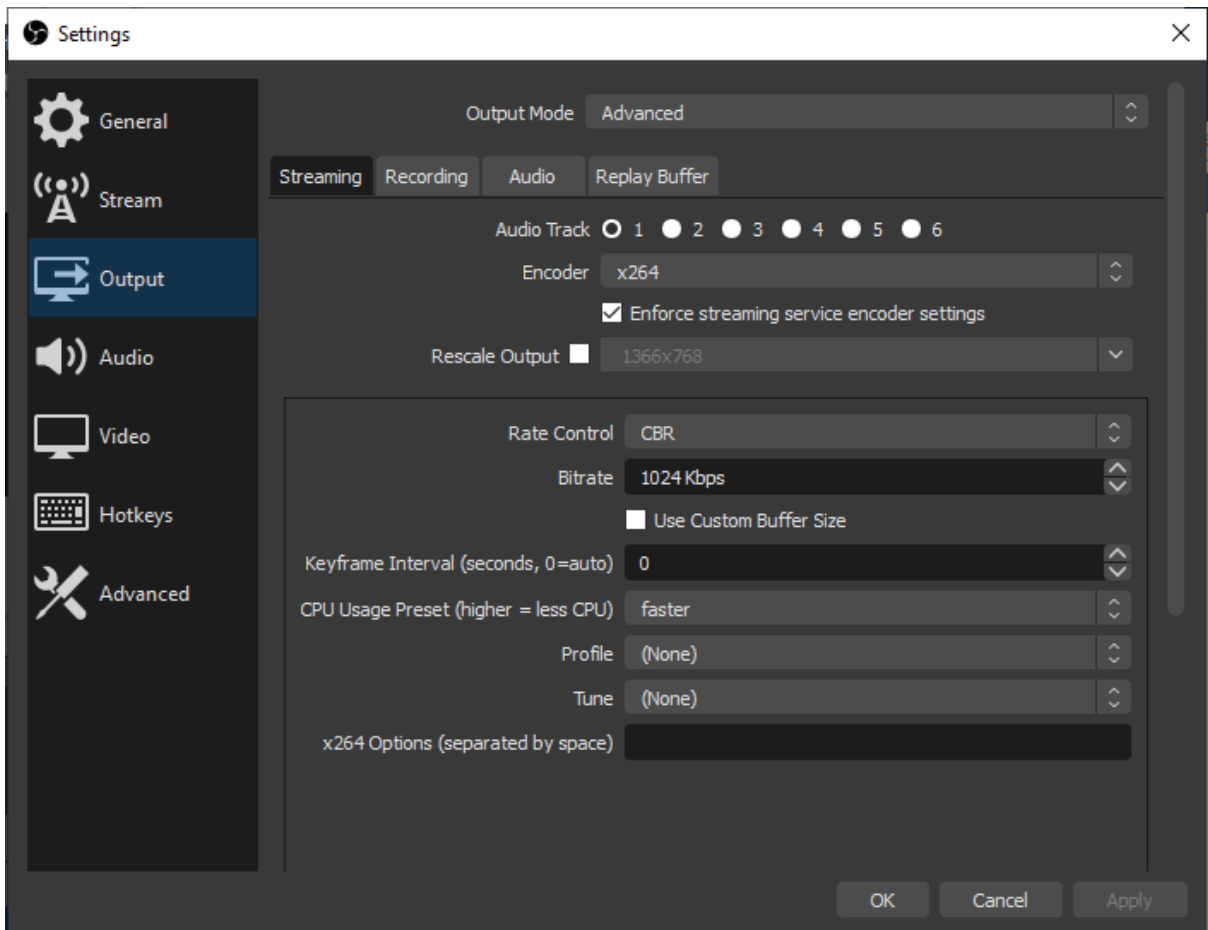


Figura 70. Parámetros de codificación

Posteriormente dar click en Ok para guardar y aplicar los cambios

A continuación, en el menú principal de OBS, en la opción de sources se debe elegir la fuente del video que se va a transmitir, en este ejemplo se transmitirá un video pregrabado y la imagen de una webcam como se muestra en figura 71.

El programa OBS cuenta con una ventana de previsualización donde se realizan todas las pruebas de video y una ventana de programación que muestra como está saliendo el contenido. En medio de las dos se encuentra el botón “Transition” que aplica los cambios realizados en previsualización para que sean mostrados en la salida de la señal. figura 72.

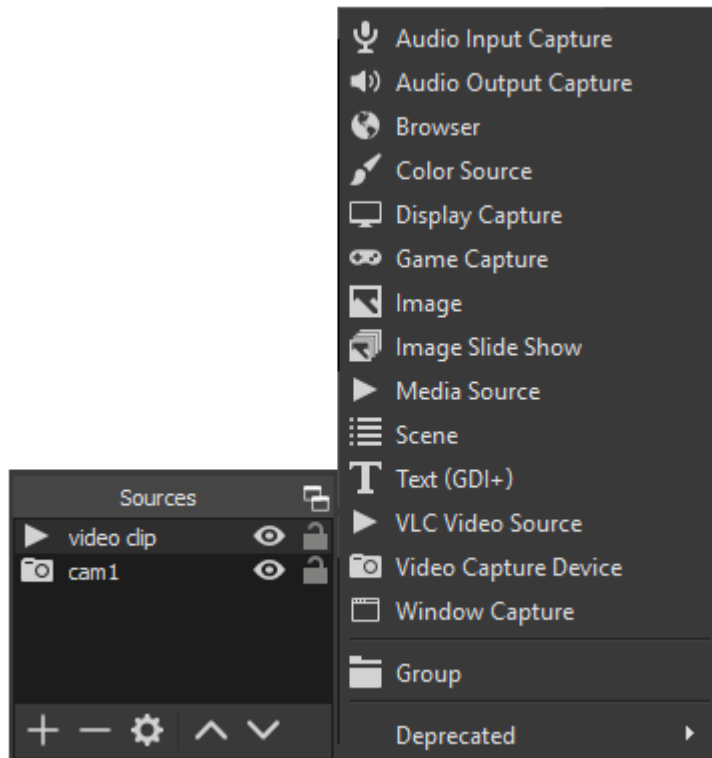


Figura 71. Entradas de video

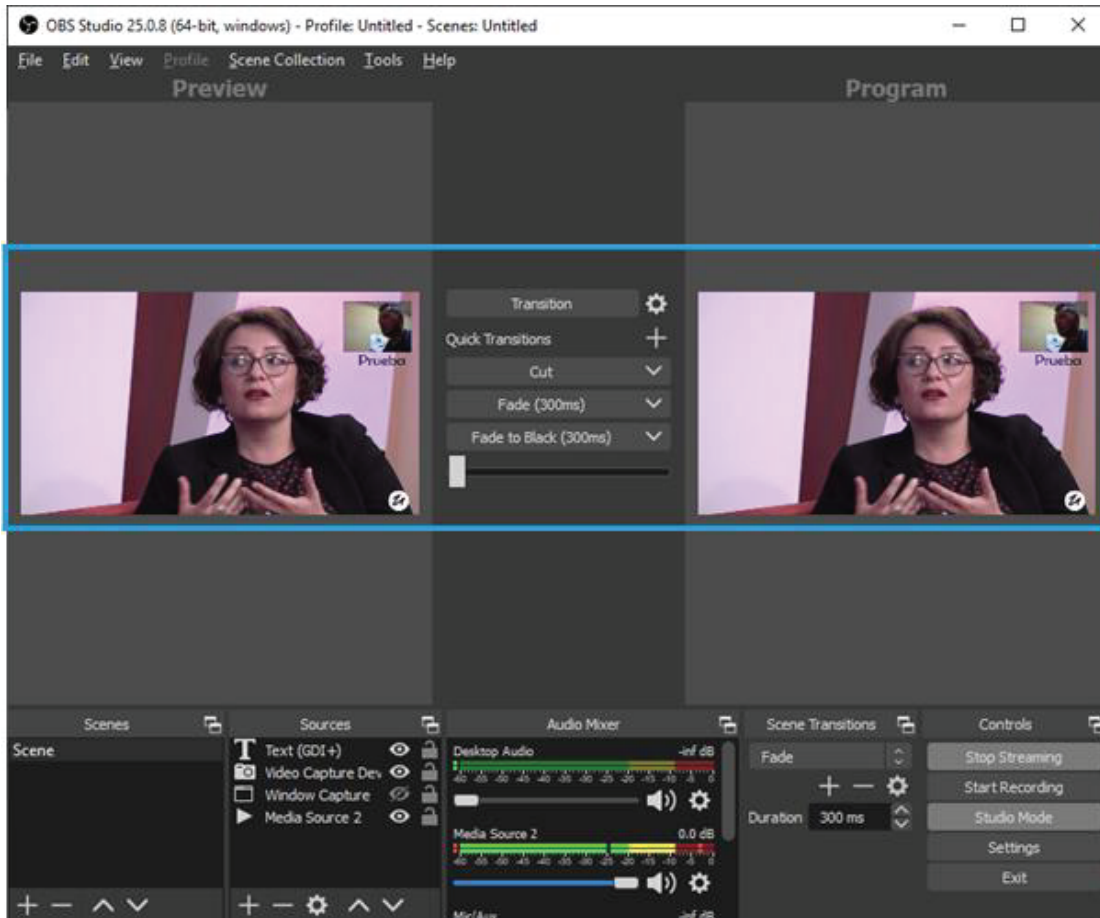


Figura 72. Ventana de control

El siguiente paso es el inicio de la transmisión mediante el botón de “Start Streaming” ubicado en el cuadro de control en la parte inferior derecha de la ventana de OBS. Al establecer la conexión con el servidor, en la barra de estado ubicada en la parte inferior de la ventana se mostrará en verde el estado de la conexión, el tiempo de transmisión, el estado del CPU, y el porcentaje de cuadros descartados mientras más cercano a cero este valor, significa que la transmisión es de mejor calidad como se muestra en la figura 73.

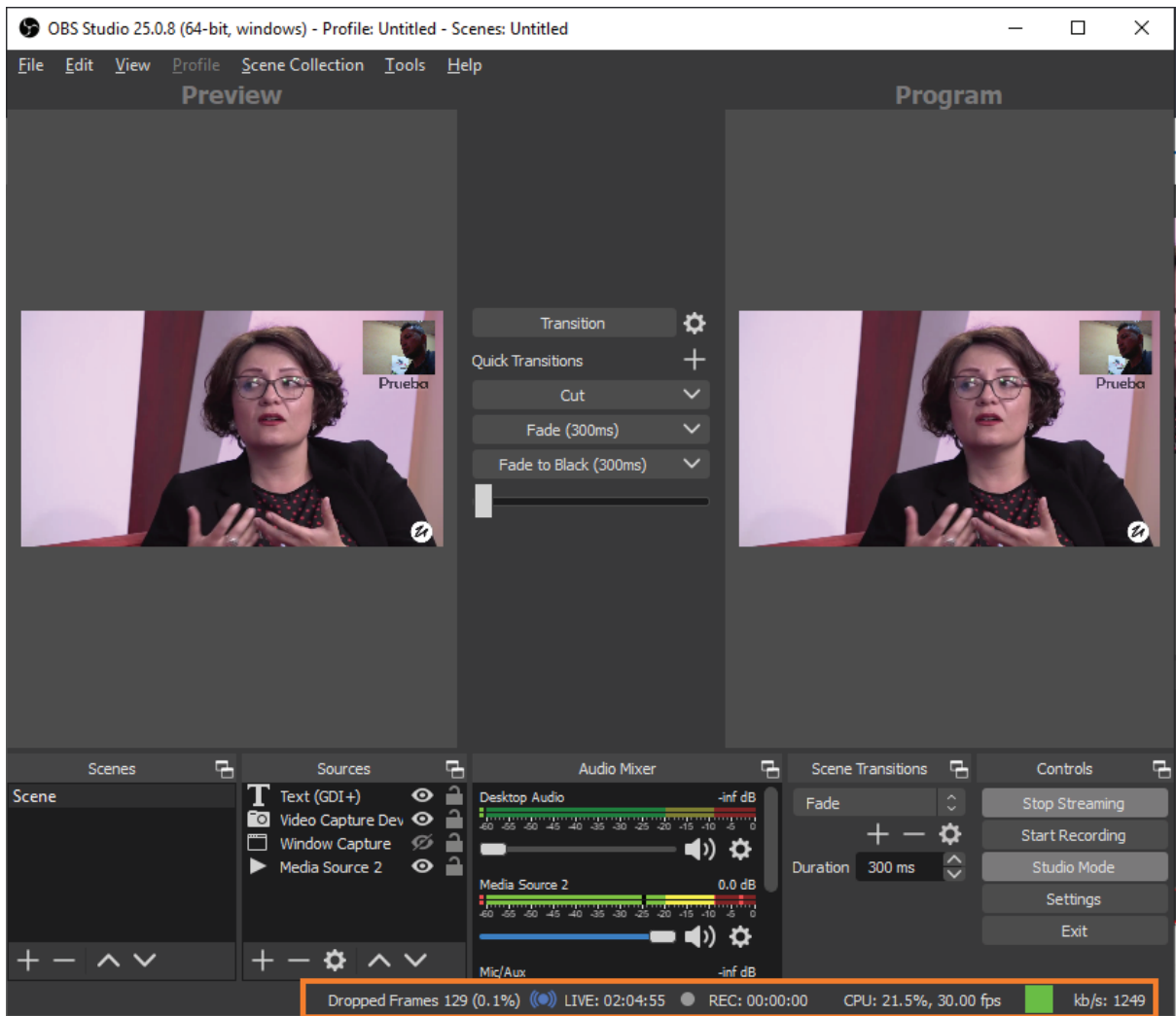


Figura 73. Streaming en video de más de 2 horas

En este punto la transmisión está establecida hacia el servidor, y para verificar que es lo que el usuario vería en su reproductor se debe regresar a la ventana de configuración de streams en el servidor, para obtener las URL de conexión hacia el stream configurado. Figura 75.

Un parámetro adicional que se puede configurar es la IP pública del servidor en caso de tener salida hacia la internet mediante un router o firewall intermedio. Figura 74.

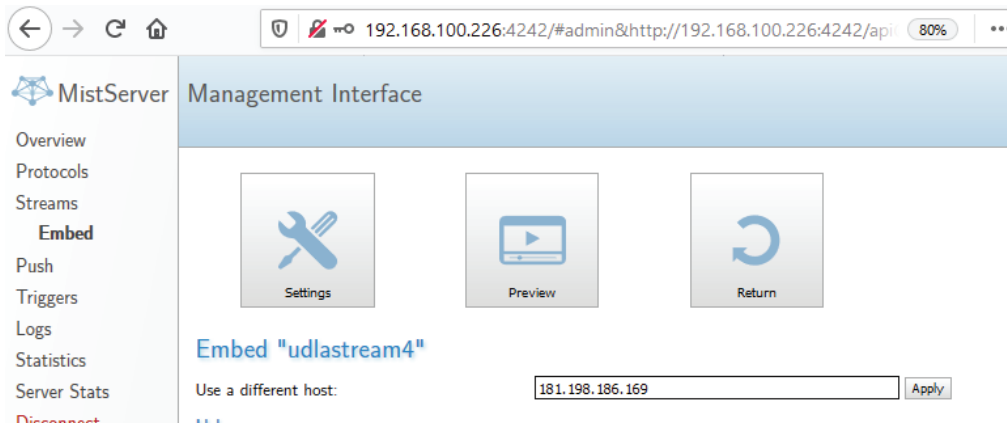


Figura 74. Configuración de IP publica

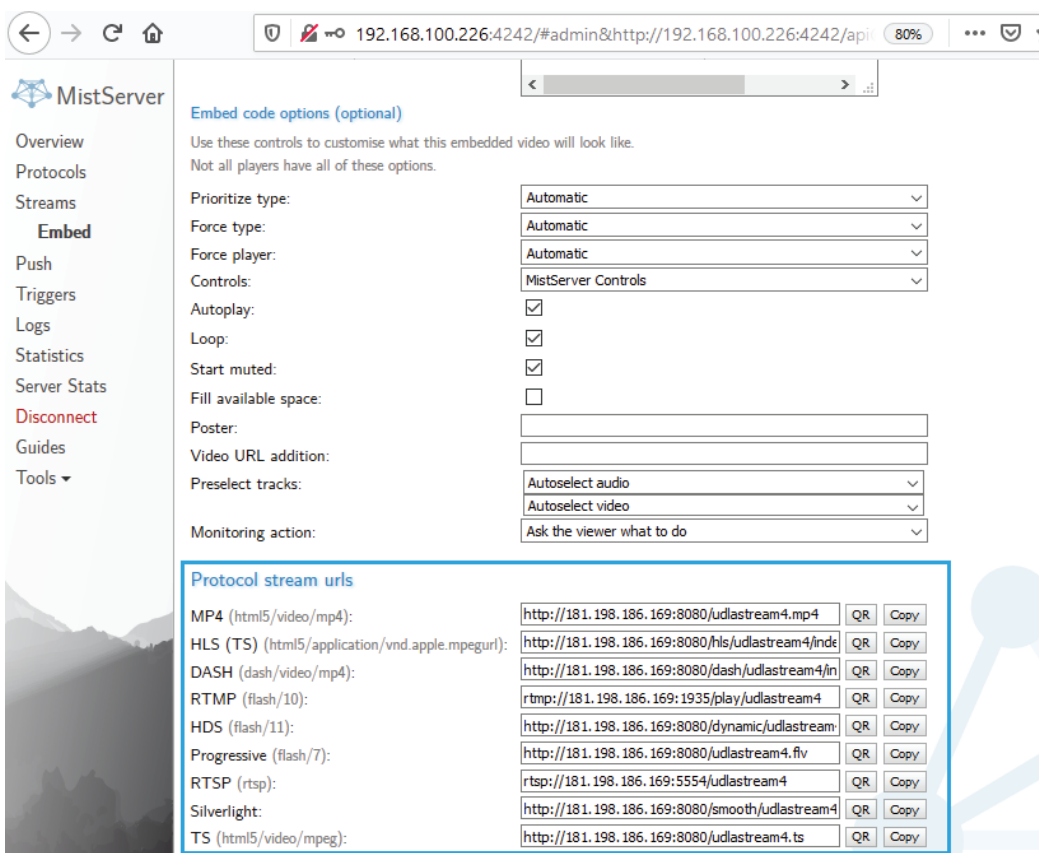


Figura 75. URLs de integración con reproductores de video.

Como se muestra en la imagen anterior el sistema genera diferentes URL con varios tipos de protocolo lo que permite integrar la señal de video con reproductores multimedia para:

- Smart TVs
- Dispositivos móviles como celulares y tablets.
- Aplicaciones de reproducción multimedia para Windows, Mac o Linux.
- Páginas web con reproductores de video “embebidos”

Para determinar cuál URL utilizar se debe verificar los protocolos de streaming soportados por el dispositivo.

Para el presente ejemplo se ha empleado el primer URL correspondiente a MP4. Se debe ejecutar la aplicación VLC e ingresar al menú de reproducción de contenido de red mediante la combinación de teclas Ctrl + n para pegar la URL copiada del stream MP4. Como se muestra en la figura 76.

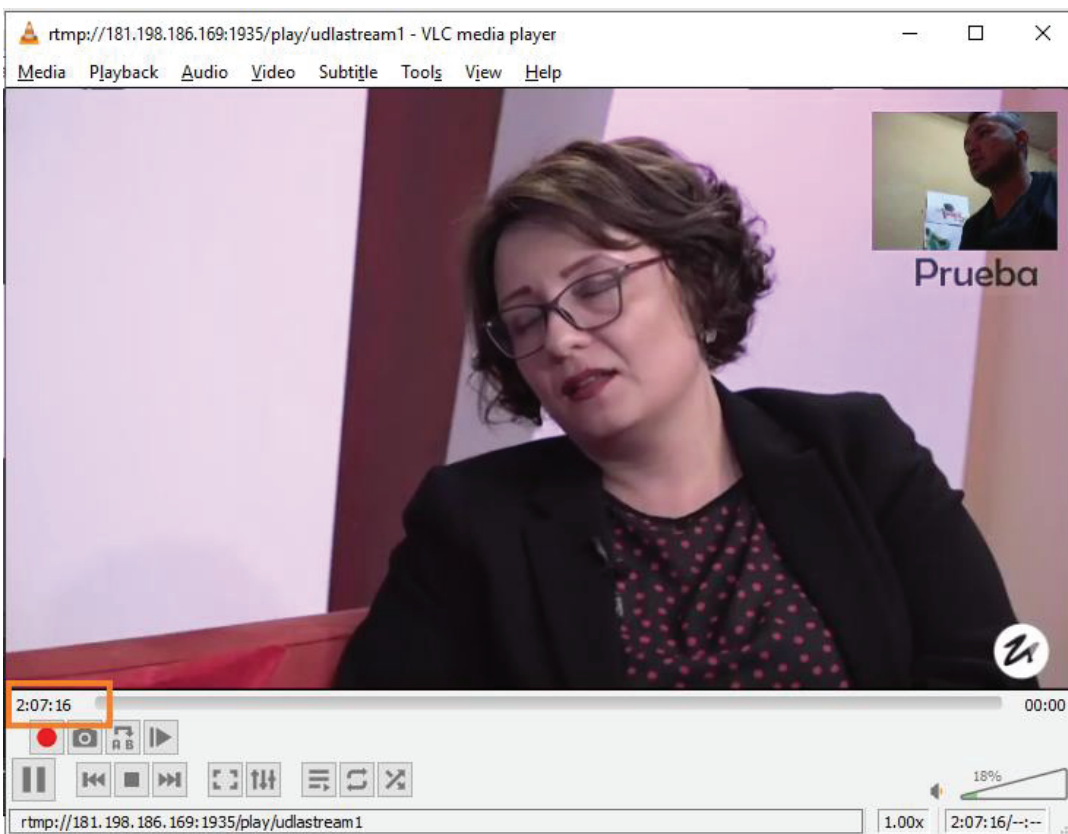


Figura 76. Reproducción de stream MP4.

De igual forma se puede reproducir este mismo stream en un dispositivo móvil copiando la URL antes mencionada como se muestra en la figura 77.

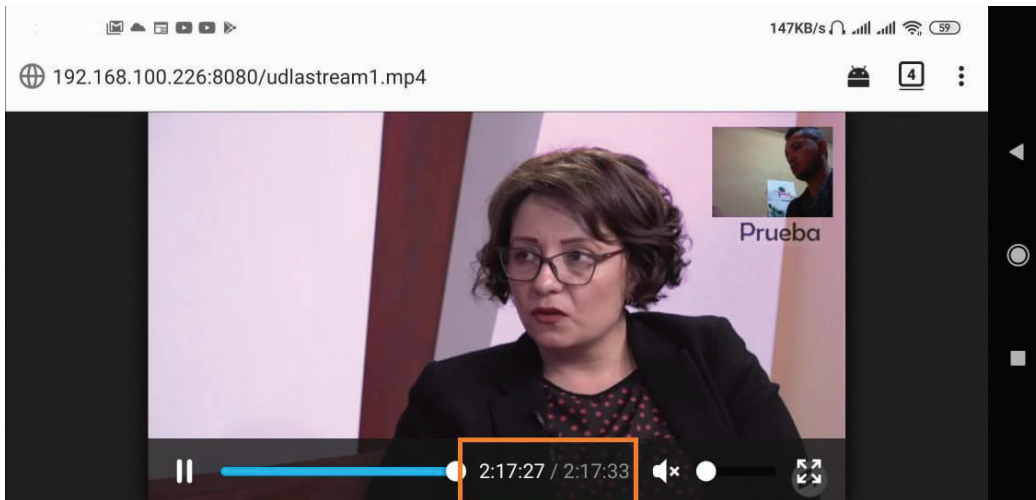


Figura 77. Reproducción en un dispositivo celular.

Como se muestra en las figuras 80 y 81 se han levantado transmisiones de más de dos horas con reproducción de dispositivos móviles y PCs de escritorio. La estabilidad del sistema se puede evidenciar en el tiempo de reproducción mostrado en las capturas de imagen y el consumo de recursos se visualizará más adelante.

3.3.4. Configuración de stream de un archivo almacenado

EL servidor de video permite configurar un stream de video de un archivo de video almacenado localmente en el servidor. Al estar ejecutado en una distribución e Linux, la aplicación accede al sistema de archivos local y distribuye un archivo de video previamente almacenado y configurado en el servidor según se muestra en la figura 78.

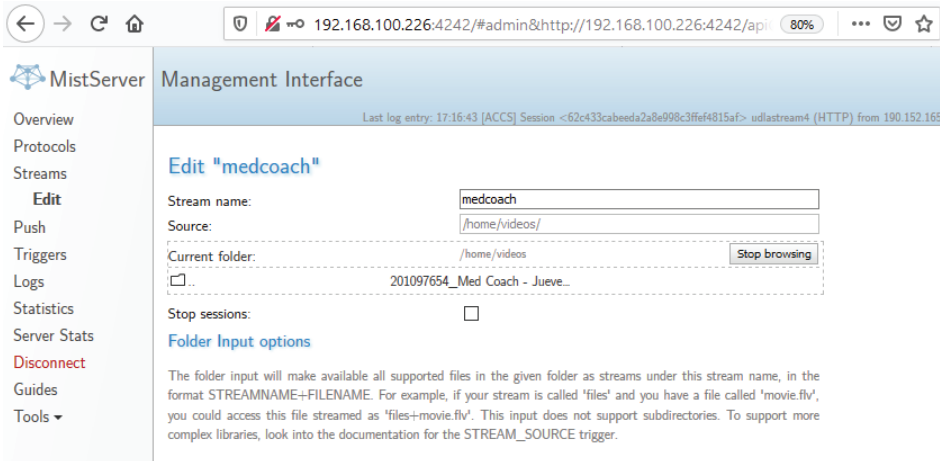


Figura 78. Configuración de stream de video almacenado en el servidor.

El proceso de transferencia de archivos al servidor se realiza mediante el programa WinSCP, el cual permite transferir archivos al servidor mediante el protocolo SFTP. La ruta del servidor en la que se ha almacenado el video es: /home/videos. (Figura 79)

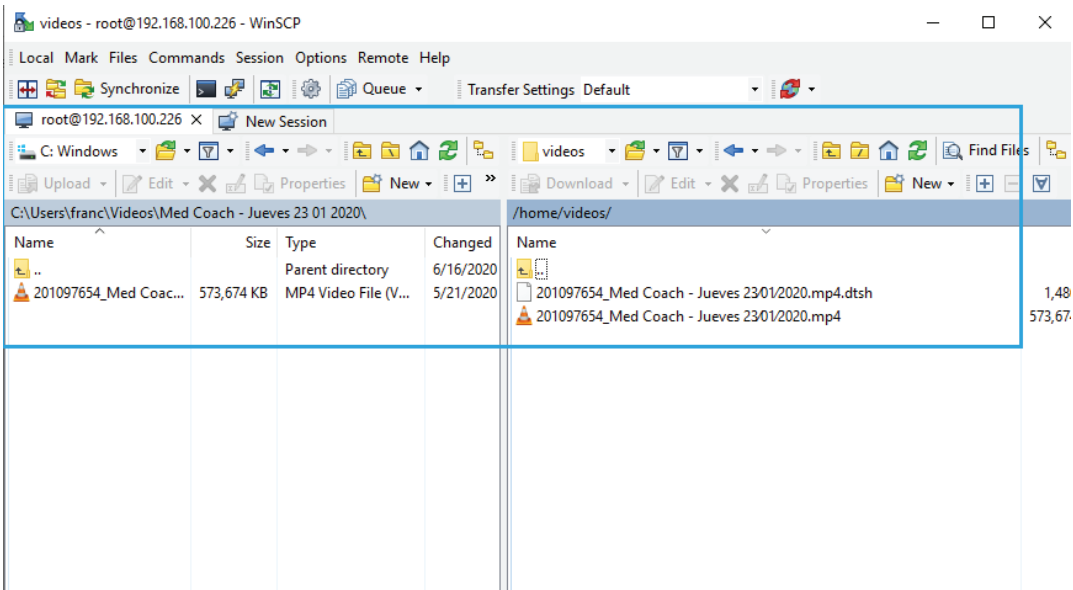


Figura 79. Transferencia de archivos con WinSCP.

En la ventana de configuración de nuevo stream se deben asignar un nombre del stream y se incluye la ruta donde se encuentra almacenado el video.

Stream name: **medcoach**

Source: **/home/videos/201097654_Med Coach - Jueves
23/01/2020.mp4**

A partir de este punto el proceso de verificación de operatividad del video se realiza con el mismo reproductor de video VLC, y las URL de conexión desde el reproductor de video se obtienen desde el submenú Embed del área de Streams del servidor.

3.3.5. Verificación de conexiones y estado del servidor.

El servidor genera un reporte grafico del estado de las conexiones en tiempo real ingresando en el submenú “Statistics” como se muestra en la figura 80.

En esta sección se pueden crear varios gráficos estadísticos para analizar el comportamiento en vivo de muchos streams de video, para esto se debe definir los parámetros que describen a continuación.

Add to: Nombre del gráfico en el que se van a presentar los resultados

Data type: y tipo de información que se va a analizar, puede ser: número de conexiones, ancho de banda de subida y de bajada, uso de CPU, y ocupación de memoria RAM.

Data origin: Se pueden analizar la información de todos los stream, o individualmente.

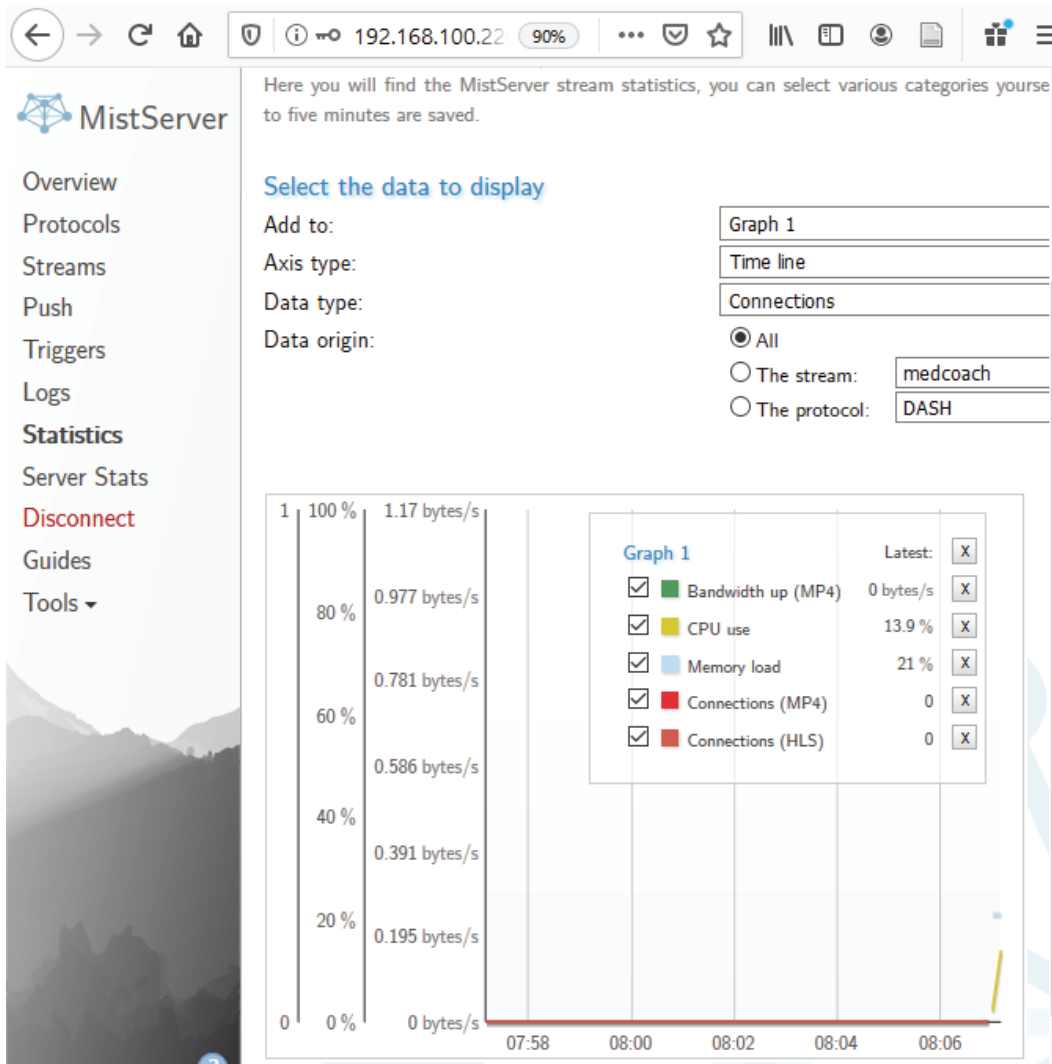


Figura 80. Área de estadísticas.

En la figura 81 imagen se muestra el consumo de recursos de una transición con 8 visualizaciones en vivo.

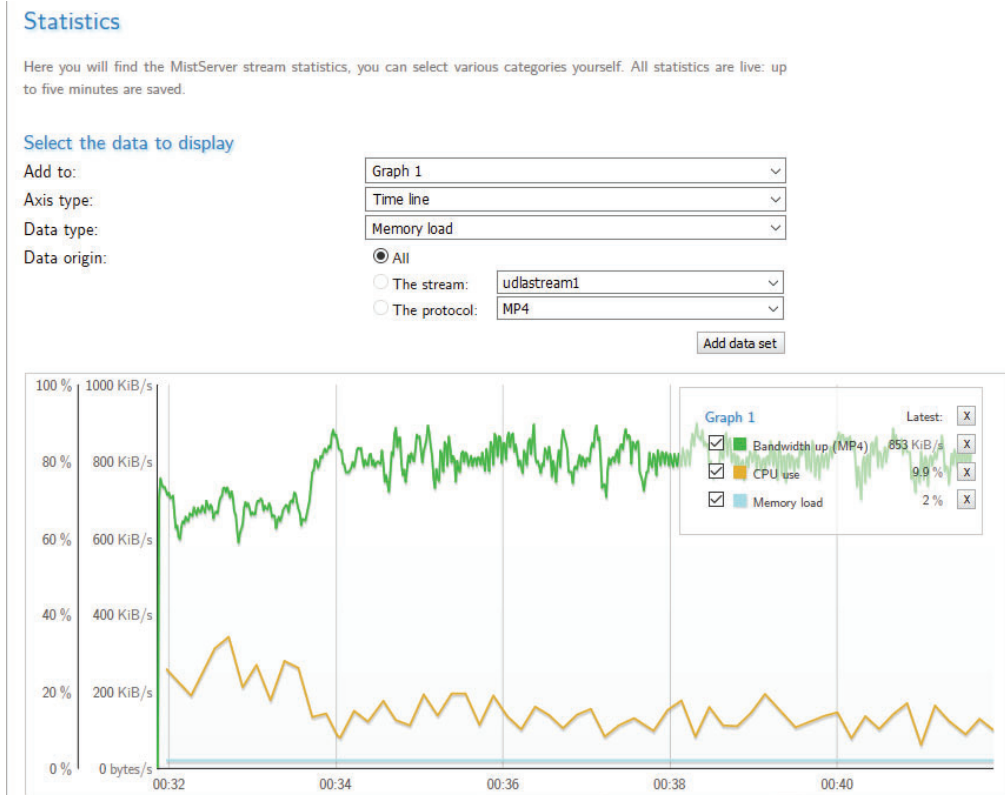


Figura 81. 8 visualizaciones en tiempo real.

3.3.6. Verificación Monitorización de recursos por la línea de comandos

Otro método para monitorear los recursos generales del sistema es ingresando a la línea de comandos de servidor mediante SSH y ejecutando el comando “**top**” como se muestra en la figura 82.

```

top - 05:43:28 up 14 days, 5:27, 1 user, load average: 3.33, 3.70, 4.69
Tasks: 29 total, 1 running, 28 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 4096.0 total, 3979.1 free, 70.6 used, 46.3 buff/cache
MiB Swap: 8192.0 total, 8177.6 free, 14.4 used. 4025.4 avail Mem

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
31335	root	20	0	74980	10632	9848	S	0.7	0.3	1:12.29	MistOutRTMP
31339	root	20	0	66532	11592	10892	S	0.7	0.3	1:21.70	MistOutMP4
31414	root	20	0	66416	11524	10920	S	0.7	0.3	0:12.55	MistOutMP4
31418	root	20	0	66416	11500	10900	S	0.7	0.3	0:11.25	MistOutMP4
31433	root	20	0	66412	11420	10828	S	0.7	0.3	0:04.34	MistOutMP4
31434	root	20	0	66408	11240	10656	S	0.7	0.3	0:03.39	MistOutMP4
31338	root	20	0	66024	3900	3448	S	0.3	0.1	0:28.39	MistInBuff+
31432	root	20	0	66412	11396	10796	S	0.3	0.3	0:04.38	MistOutMP4
1	root	20	0	170352	4268	2852	S	0.0	0.1	0:08.52	systemd
71	root	20	0	19304	136	0	S	0.0	0.0	0:02.02	systemd-loc
72	message+	20	0	9236	724	312	S	0.0	0.0	0:01.23	dbus-daemon
97	root	20	0	6924	112	112	S	0.0	0.0	0:00.01	login
98	root	20	0	2420	212	104	S	0.0	0.0	0:00.00	agetty
270	root	20	0	4660	1464	1168	S	0.0	0.0	0:00.04	bash
3185	root	20	0	29680	4088	3856	S	0.0	0.1	0:04.67	systemd-jo+

Figura 82. Monitorización de recursos por la línea de comandos.

4. Capítulo IV. Diseño de la solución e implementación de la estación móvil

Este capítulo tiene la finalidad de proponer el diseño e implementación de estación móvil funcional. El diseño servirá de base para la implementación del prototipo, así como la configuración general para que el servicio de video *streaming* pueda ser retransmitido el contenido de UDLA Channel o auditorios de la universidad y transmitido en lugares de interés.

Este proceso consiste en la identificación de hardware y software que se usará tanto en el servidor como en el cliente y también la instalación y configuración de los componentes seleccionados.

4.1. Diseño de la solución

El proceso del diseño de la solución consta de tres pasos esenciales que ayudarán a identificar y justificar la selección de los componentes. Estos pasos son:

Primero, se identifican las dos dependencias: el generador de contenidos “UDLA Channel” en conjunto con el servidor de video streaming y móvil. Segundo, se determina el hardware y software a utilizar en la implementación del prototipo. Por último, se presenta un esquema gráfico del prototipo y su funcionamiento.

4.1.1. Generador móvil de video *streaming*

El generador móvil de video es un sistema para capturar y codificar el contenido multimedia que se crea en cualquier lugar, para esto se empleará varios factores como software y hardware. En Hardware tenemos: ordenador, capturadora de video, cámara de video y micrófono.

Software se tiene en cuenta el programa a utilizar para realizar el streaming y las características del ordenador para que el video y audio tengan la mejor calidad para el usuario final. Este generador móvil de contenido se conectará vía enlace de red al servidor de retransmisión para poder visualizar en los televisores.

4.1.1.1 Selección de Software

Para la implementación del sistema móvil de *stream* de video en el estudio, es necesario la selección de programas esenciales para el funcionamiento del sistema. Este apartado cumple con el propósito de utilizar herramientas de software libre para la implementación del prototipo funcional.

4.1.1.2 OBS Studio

Como dicen sus siglas en inglés, OBS Studio es un software libre de código abierto para la grabación y transmisión de video en vivo, además, permite la captura de audio y video de diversas fuentes en tiempo real y su incorporación.

Su código fuente está disponible en GitHub, escrito en la programación C y C++ con soporte para varias plataformas de streaming como Twitch, YouTube, servidores personalizados, entre otras. (UBUNLOG, 2017)

El principal requerimiento de sistema de OBS es la capacidad y rendimiento del procesador, el cual variará dependiendo del encoder, resolución, FPS y la complejidad de la escena, a su vez posee la herramienta de configuración automática y de esta manera se ajusta al sistema que se usa. Además, se necesita del API DirectX en el caso de Windows para facilitar las tareas de multimedia.

La selección de este programa de software libre está ligada a la captura multimedia de varias fuentes y la posibilidad de realizar la transmisión a través del protocolo RTMP (Real Time Messaging Protocol), dicho protocolo se usará en la implementación de Nginx como servidor web de video streaming.

4.1.1.3 XSPLIT

XSplit Broadcaster: se define como “un software de transmisión y grabación en vivo simple y potente que brinda innumerables transmisiones en vivo y grabaciones” (Bartolome, 2003) y es la elección perfecta para producir contenido de vídeo enriquecido con los videojuegos que amas. Esta es la forma más sencilla de iniciar la transmisión o grabación de juegos en tu PC o consola.

4.1.1.4 Selección de Hardware

El sistema puede ser instalado y configurado en un computador de escritorio o laptop, por el simple hecho que los requerimientos de software son medios, sin la necesidad de adquirir equipos natos para servidores o de gran rendimiento, logrando abaratar costos en el prototipo. Además, se necesita de una pequeña tarjeta para capturar la señal de UDLA Channel para poder ser interpretada por el computador.

4.1.1.5 Computador

El computador es el encargo de alojar todos los componentes seleccionados en el punto, ya que se ha tratado utilizar software de gran capacidad a bajas prestaciones, se puede decir que el computador debe tener las siguientes características:

Tabla 7. Requerimientos de hardware para la estación móvil.

Requerimientos	Mínimos		Recomendados	
CPU	Intel i5	AMD Ryzen 5	Intel i7	Nvidia 4gb
RAM	8GB		8 GB	
Almacenamiento	250 GB HDD		250 GB SSD	
USB	3.0 x3			

4.1.1.6 Capturadora de video

La Capturadora de video tiene la finalidad de como su nombre lo dice, capturar el video proporcionado por el sistema móvil, este a su vez trasmite una señal de salida que se puede interpretar por una computadora y con el software indicado proyectar esta señal de video en el computador y a su vez transmitirla a servicios de video streaming.

La tarjeta está compuesta por un canal simple de entrada de video HDMI y por dos interfaces de salida, un USB 3.0 para comunicación al PC y un HDMI para transmitir la señal de entrada a una pantalla. En la figura 87, se puede observar un tipo de uso con una filmadora.



Gráfico 83. Capturadora de video.

Tomado de (MIraBox, 2019)

Cámara de video (**CAMCORDER**)

Es un sistema de captura de imágenes en movimiento la cual permite recibir el contenido generado en vivo en cualquier lugar. Para el sistema no es necesario una

cámara de alta gama. Se necesita una cámara que capture en 1080i 60Hz es decir full HD, las cuales existen muchas variedades disponibles a la venta y también existen modelos adquiridos por la facultad de comunicación y se puede pedir prestadas para la coyuntura.

4.1.1.7 Micrófono

Para una producción en vivo, este dispositivo debe captar la mayoría de las frecuencias que tiene la señal de voz, para que al transmitirla sea de mayor agrado del espectador.

4.1.1.8 Smart TV

Smart tv es el último componente del prototipo cuya función es la captación del video y sonido a través de un navegador web o aplicativo y la emisión de este contenido audiovisual en las cafeterías. Para el prototipo basta con que el smartv tenga acceso a internet o puerto de red.

4.2. Esquema de la solución

El resultado de la selección de Software y Hardware para el sistema móvil se ve esquematizado en la figura 84, lo cual permite identificar los componentes necesarios para entender el correcto funcionamiento del prototipo.

Además, el servidor tiene la funcionalidad de capturar el contenido multimedia generado por UDLA Channel o de fuentes externas y realizar el control remoto de los dispositivos clientes que se encargan de emitir el streaming de video.

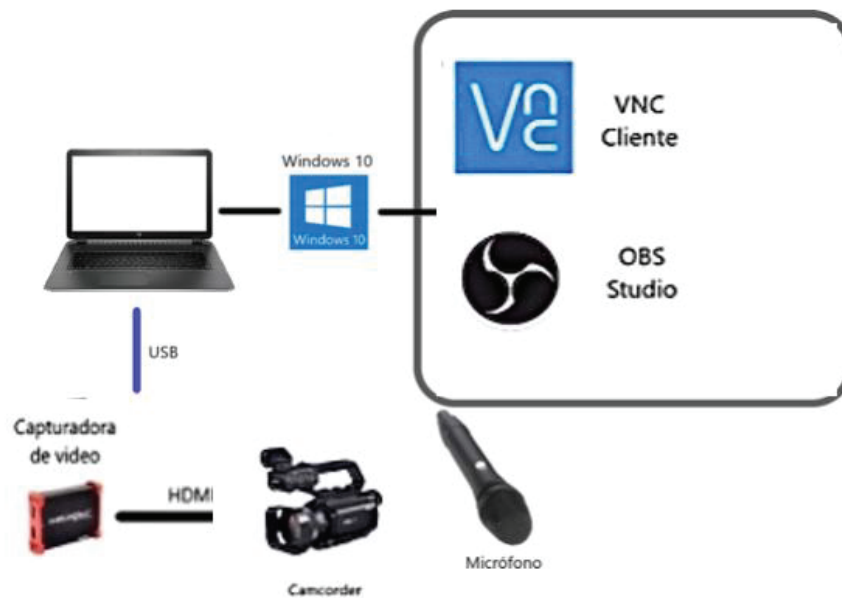


Figura 84. Diseño del sistema móvil.

4.2.1. Estación móvil

La funcionalidad de la estación móvil es generar contenidos de eventos realizados en la universidad y transmitir el mismo por streaming de video. En el apartado 4.1.1. Se definió los elementos necesarios para el servidor tanto estático como móvil los cuales se implementarán en la estación móvil a diferencia que se debe utilizar un computador portátil para garantizar la portabilidad de la estación y la adquisición de dispositivos para la captura de audio y video.

4.2.2. Computador portátil para sistema móvil

El computador portátil debe tener ciertas características técnicas para garantizar la codificación del video, la trasmisión de este y la virtualización del servidor. Por lo tanto, en la tabla 4 se detallan los requerimientos de hardware necesarios que debe tener el computador portátil:

Tabla 8.Requerimientos de computador para unidad móvil

Hardware	Capacidad o Modelo
Procesador	Intel Core i7 o Nvidia 4gb
Memoria RAM	8 GB
Almacenamiento	SSD 250 GB
Periféricos	USB 3.0

Con el hardware definido la estación móvil tendrá la capacidad de trabajar sin inconvenientes garantizando el video streaming para que pueda llegar al usuario final.

4.2.2.1 Video cámara o camcorder

La video cámara de mano de marca Sony y modelo HXR-MC88 es una *camcorder* profesional con la capacidad de capturar video a una resolución máxima del estándar Full HD o 1920 x 1080 pixeles. Está diseñada para tener un gran rendimiento de imagen a un precio asequible, ideal para centros educativos para la enseñanza de producción de video, para presentaciones empresariales y *videoblogs*. (Sony, 2019)

Esta cámara tiene características que la hacen esenciales para implementar en la solución móvil con el objetivo de que la calidad de imagen sea igual de buena que la transmitida por UDLA Channel, también posee atributos como la salida por micro USB de la imagen captada a un computador que permite interpretar el video y transmitirlo por el servidor.

4.2.2.2 Micrófono

El micrófono ayuda a captar el audio generado por el presentador, en el presente proyecto se utilizará un micrófono de cable pues actualmente es el más estable en transmisiones, ya que los micrófonos inalámbricos son muy inestables porque

existen muchas señales de Radio Frecuencia (RF) que trabajan en la misma frecuencia de estos y se interfieren muy fácilmente. Se utiliza el micrófono de la marca Shure modelo sm58 ya que tiene una amplia recepción de frecuencia auditivas, el cual enriquecerá con una gran definición a la transmisión en lo que audio se trata.

4.2.2.3 Diseño general de la estación móvil

La estación móvil se asemeja mucho a la estación fija ubicada en UDLA Channel, por lo tanto, la funcionalidad y los componentes de software siguen siendo los mismo, de esta manera se presenta en la figura 89 la propuesta para estación móvil:

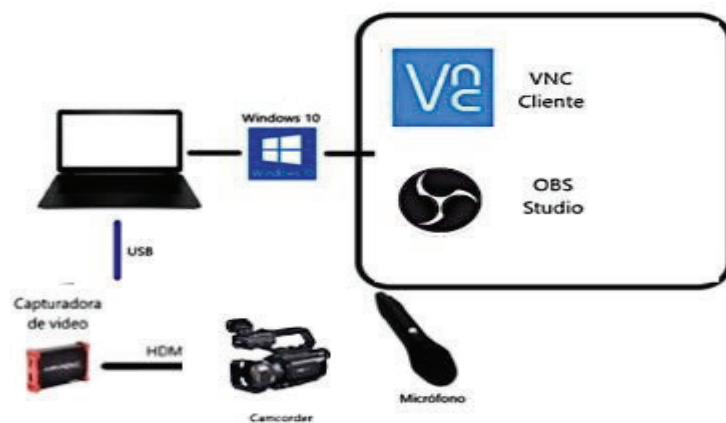


Figura 85. Solución móvil de video *Streaming*

Tomado de Elaboración Propia

4.3. Implementación del sistema

En la implementación del prototipo se hará énfasis en los parámetros de instalación y de configuración con mayor relevancia de los componentes seleccionados en el apartado 3.1 “Diseño de la solución”, en dicho apartado se especificó programas de

gran utilidad con bajos requerimientos de hardware con la capacidad de lograr los objetivos del proyecto.

4.3.1. Configuración del emisor de contenidos (OBS Studio)

La emisión de contenido multimedia por parte de OBS Studio se la realiza con la configuración de dos aspectos: El primero, es la configuración de una fuente multimedia. OBS Studio permite una variedad muy amplia de fuentes multimedia desde internas como externas como periféricos de videos, por ejemplo. Entonces, para el caso de UDLA Channel se configura una nueva fuente desde un dispositivo de captura de video, en este se selecciona parámetros de resolución y fps. En la figura 14, se puede observar la forma de operar OBS Studio.

El segundo, trata de la configuración y conexión al servidor de video streaming. En la ventana de configuración en la pestaña de emisión se selecciona un servicio personalizado, donde se indica la URL del servidor de video streaming que el cual tiene el siguiente formato “rtmp://[Dirección IP]/\$name/”, donde:

- **Rtmp://.** - Indica el servicio web.
- **Dirección IP.** – Se apunta a la dirección IP del servidor WEB.
- **/\$name.** – Es el nombre del aplicativo del servicio web RTMP que se haya configurado.

Además, permite establecer una clave de trasmisión, la cual el servidor valida y da paso a la conexión por parte del cliente. En la siguiente figura se detalla los campos a llenar:

Servicio: Personalizado

Servidor: rtmp://181.198.186.169/live/udlastream

Clave: udlastream

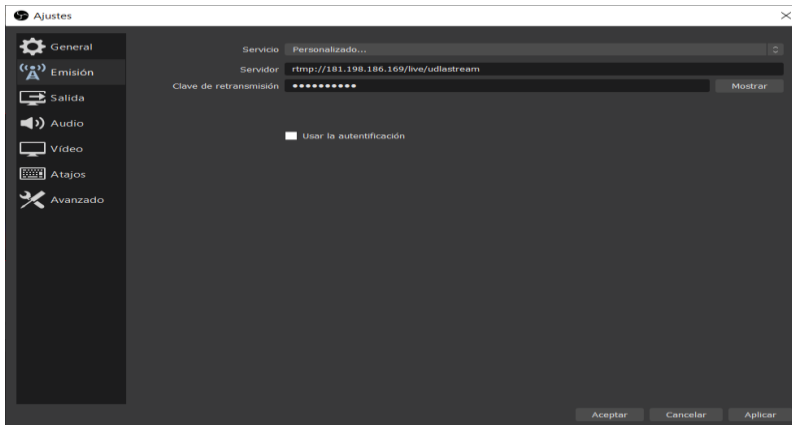


Figura 86. Configuración OBS.

Una vez configurado los dos puntos, OBS está listo para la transmisión al servidor de video streaming.

5. Capítulo V. Pruebas de funcionamiento

Al finalizar el proceso de diseño e implementación del sistema es importante realizar un análisis de resultados o pruebas. Esto conlleva a realizar un análisis en la calidad del servicio, es decir, analizar la calidad de imagen, el tiempo de retardo y el consumo de recursos tanto del servidor como de red. Además, se analiza el manejo del prototipo por parte del usuario administrador que se ubica en el servidor y el usuario cliente que se ubica en el sistema móvil.

5.1. Conexión estación móvil con el servidor

Esta conexión para el sistema móvil es fundamental, ya que es el primer enlace antes de salir a los monitores finales, por esta razón la siguiente prueba se debe realizar antes de cada transmisión. Estar conectados a una red de datos, prueba de ping request.

5.1.1. Conexión a red de datos.

Para conectarse desde el sistema móvil al servidor se puede hacer de 2 formas:

- **Puerto LAN:** esta conexión se realiza con cable de red (UTP), a un Router el cual proporcionará la IP por DHCP.
- **WIFI:** el computador portátil buscará la red WIFI disponible y se conectara mediante una contraseña predefinida. La dirección se obtiene por DHCP.

5.1.2. PING Request

El objetivo del ping es determinar si el host destino (servidor de streaming), , es accesible desde otro host (sistema móvil). Esto ayuda a verificar si se tiene conexión de extremo a extremo. En el sistema Windows se debe digitar lo siguiente: botón de windows + r. Aparece la siguiente ventana como se muestra en la figura 87.

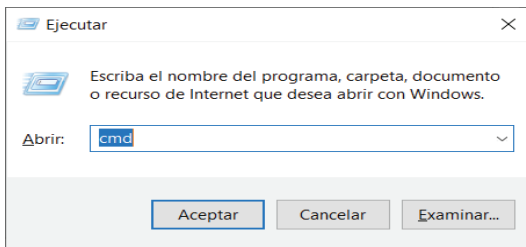


Figura 87. Venta ejecutar.

Digitar cmd y la tecla tecla Enter, aparece la ventana que se muestra en la figura 89.

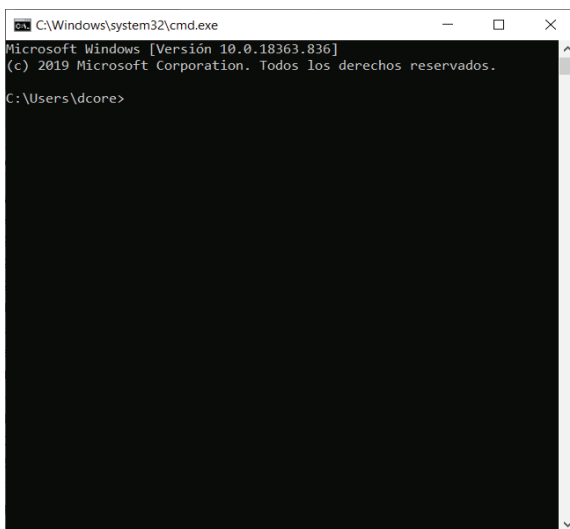
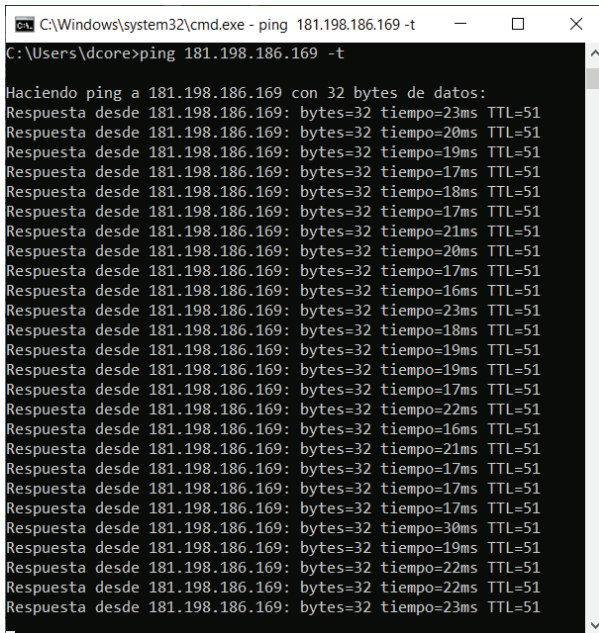


Figura 88. Línea de comandos Windows.

En esta ventana teclear el comando “ping (ip del servidor) - t y ejecutar como se muestra en la figura 90.



```

C:\Windows\system32\cmd.exe - ping 181.198.186.169 -t
C:\Users\dcore>ping 181.198.186.169 -t

Haciendo ping a 181.198.186.169 con 32 bytes de datos:
Respuesta desde 181.198.186.169: bytes=32 tiempo=23ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=20ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=19ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=17ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=18ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=17ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=21ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=20ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=17ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=16ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=23ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=18ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=19ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=19ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=17ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=22ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=16ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=21ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=17ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=17ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=17ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=30ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=19ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=22ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=22ms TTL=51
Respuesta desde 181.198.186.169: bytes=32 tiempo=23ms TTL=51

```

Figura 89. Uso de PING

En la figura 90 se muestra que existe conexión a la IP destino servidor, el cual responde a un paquete de 32 bytes ICMP con una media de tiempo de 23ms.

5.2. Calidad de imagen

El contenido audiovisual proporcionado por el sistema móvil es capturado por una tarjeta de video externa a una resolución de 1920 x 1080 pixeles, denominada comúnmente como Full HD siendo el estándar de video de alta calidad. En la actualidad es el estándar comúnmente utilizado. Se estableció la calidad de imagen en Full HD por los siguientes detalles técnicos:

- La capturadora de video que permite al servidor interpretar la fuente de video proporcionada por UDLA Channel, es capaz de capturar las imágenes con resoluciones de hasta 1920 x 1080 pixeles, o, Full HD.

- Los televisores existentes en son full HD, es decir, se presentarían las imágenes sin ningún inconveniente.
- El estándar 4K conlleva al uso de una cantidad mayor de recursos como: el procesador de un computador para la codificación del video y el ancho de banda para la difusión del video a los diferentes dispositivos clientes.

5.3. Tasa de transferencia de datos

Para poder tener un rendimiento óptimo del *streaming* de video se procede a realizar un análisis de la tasa de transferencia. Esta característica es de vital importancia ya que permite ver la tasa de transferencia de datos de un punto a otro en un determinado periodo. A pesar de que a nivel local el ancho de banda se presupone que es igual para los dispositivos, es importante determinar un valor que permite realizar las configuraciones necesarias para su óptimo rendimiento en la red.

Para determinar la magnitud de transferencia de datos por parte del servicio de video *streaming*, se utilizó el aplicativo de monitoreo y control de tráfico de red *NetBalancer*, éste muestra el consumo de red tanto de subida como de bajada de cada aplicación del sistema.

Con el uso del programa se tomó 169 muestras del aplicativo OBS Studio, encargado de realizar la emisión de un contenido audiovisual con un intervalo de cuatro minutos y medio. Los resultados de las muestras se pueden observar en la Figura 94, donde:

- La máxima cantidad de transferencia de datos fue de 645 KB/s.
- La mínima cantidad de transferencia de datos fue de 241 KB/s.
- El promedio de transferencia de datos fue de 416 KB/s.

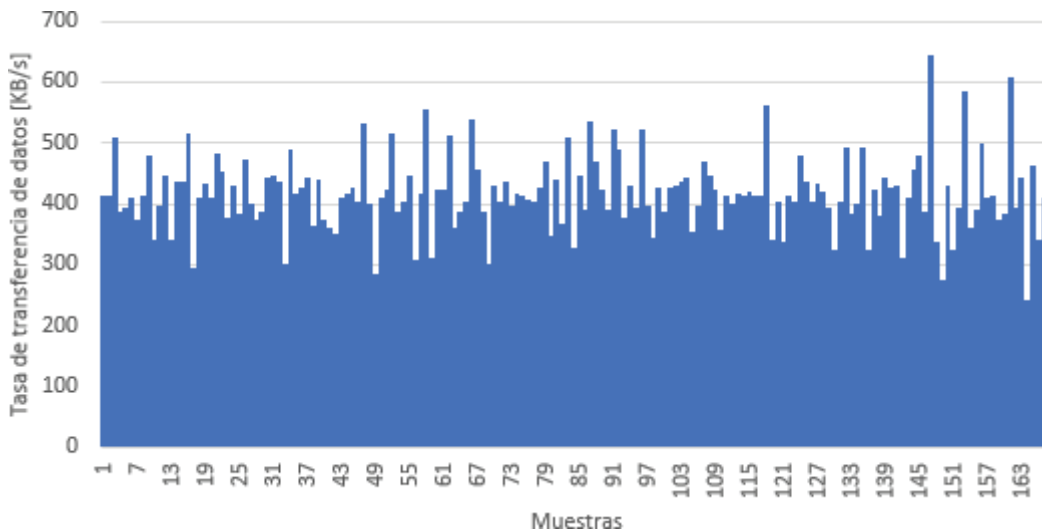


Figura 90. Consumo de ancho de banda

Por lo tanto, con los datos obtenidos se determina que existirá un consumo de un máximo de 645 KB/s de subida y el mismo valor por cada dispositivo cliente.

5.4. Tiempo de retardo

El tiempo de retardo o latencia se analizó localmente, midiendo el tiempo desde la emisión del contenido por parte de OBS Studio hasta su transmisión en la televisión, con el servidor montado en dos diferentes tecnologías de unidad de almacenamiento, estas son: unidad de disco duro y unidad de estado sólido, obteniendo los siguientes resultados detallados en la tabla 9.

Tabla 9. Tiempo de retardo de transmisión de video streaming con RTMP.

Tipo de Almacenamiento	Hard Disk Drive (HDD)	Solid State Drive (SSD)
Tiempo de retardo [s]	6.0	3.6
Nota: Los tiempos medidos se obtuvieron con el uso de un cronómetro.		

Para esta prueba de medición de retardo, la virtualización del servidor se la realizó en el mismo computador con los mismos recursos físicos de hardware, a diferencia

que, uno se alojó en la unidad de almacenamiento de disco duro y el otro en la unidad de almacenamiento de estado sólido.

Por lo tanto, con los resultados obtenidos en la tabla, se puede determinar que la latencia a más de verse afectada por el tipo de conexión de red y las distancias a recorrer, se ve afectada también por los recursos del computador como la unidad de procesamiento ya que se necesita codificar el video al codec H.264, pero el cambio más notorio se observó con el uso de distintos tipos de almacenamiento.

Entonces, el mejor resultado se logró con el uso de una unidad de almacenamiento de estado sólido logrando un tiempo de 3.6 segundos desde su emisión hasta su transmisión.

6. Capítulo VI. Análisis de factibilidad financiera

Este proyecto está pensado como implementación de negocio y ahorro de un rubro por la emisión de contenido a uno o varios sistemas.

Ajustándose a las necesidades de la institución y luego del análisis del proyecto, es factible la retransmisión a varias plataformas siendo necesario la adquisición de una licencia para la retransmisión, denomina Restrim.io. Esta licencia tiene un costo adicional al valor de gastos fijos del proyecto, así como también al costo de transmisión, los cuales se detallan a continuación.

6.1. Costos de Implementación

Costos de sistema móvil

- Portátil: \$1300
- Capturadora: \$120
- Cámara: \$1250
- Micrófono: \$120
- Cables de video y audio: \$120
- UPS: \$90
- Convertidor SDI-HDMI: \$170

Total :\$3170

Costos servidor streaming

- Servidor Proliant: \$990
- Software licenciado Mist Server: \$300

Total: \$1290

COSTO TOTAL

\$4460

Este proyecto se toma como a corto plazo puesto que los equipos electrónicos llegan a la obsolescencia cada 3 años.

6.2. Evaluación Económica

Realizar una evaluación económica para la implementación de servidor streaming con sistema móvil.

Tabla 10. Descripción de los costos

DATOS	
COSTO DE IMPLEMENTACION	\$4460
PRECIO DE TRANSMISION NETO	\$250 POR TRANSMISIÓN A UN SISTEMA \$300 POR TRANSMISIÓN A VARIOS SISTEMAS
LICENCIA RESTRIM.IO PARA RETRANSMISIÓN A VARIOS SISTEMAS	\$50 MENSUALES
LICENCIA DE SOFTWARE DE TRANSMISIÓN	\$16 MENSUALES
GASTOS FIJOS	\$80 POR TRANSMISION
GASTOS VARIABLES: movilización	\$20

Tasa de descuento	10%
-------------------	-----

En la tabla 11, se detallan los costos de implementación, este costo es real con precios en el mercado ecuatoriano. Costo por transmisión el costo de una transmisión a plataformas como Youtube, Facebook, Periscope o simplemente a un servidor propio. Costos de fijos de transmisión tales como comida para camarógrafos y sus honorarios por día. Costos variables como son los gastos de movilización y licencia de software.

Para el siguiente calculo se tomará en cuenta los costos por transmitir a varias plataformas simultáneamente mínimo 3 plataformas.

Tabla 11. Descripción de egresos del sistema móvil.

PRODUCCION	CANTIDAD	GASTOS FIJOS	GASTOS VARIABLES
AÑO 1	16	1280	1376
AÑO 2	16	1280	1376
AÑO 3	16	1280	1376

En la tabla 11, se detalla el valor total de los gastos fijos y variables que se hacen anualmente, teniendo en cuenta el número de transmisiones.

Tabla 12. Cálculo de flujo de fondos

	AÑO 0	AÑO 1	AÑO 2	AÑO 3	TOTAL

	AÑO 0	AÑO 1	AÑO 1	AÑO 3	TOTAL
INGRESOS		4800	4800	4800	
EGRESOS	-4460	-1856	-1856	-1856	
FLUJO DE FONDOS	-4460	2144	2144	2144	1972
FLUJO DE FONDOS DESCONTADOS	-4460	1949,0909	1771,9008	1610,8189	871,8106

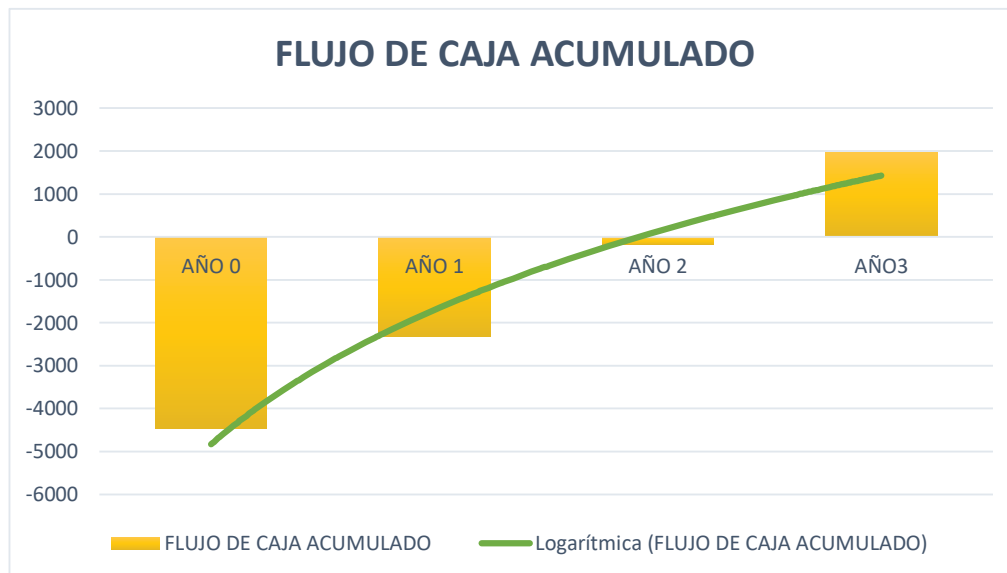
En la tabla 12 se muestra los valores del flujo de fondos, que se hacen desde el año inicial en este caso año 0 que es la inversión en el sistema, hasta el año 3 dando un valor total de \$1972 USD y un flujo de caja descontado es decir traídos al valor actual con una tasa de descuento del 10%, dando un valor de \$871.8106 USD.

Tabla 13. VAN, IVAN, TIR

VAN (10%)	871,8106
IVAN	0,1954
TIR	21%

En la tabla 13 se muestra el valor del VAN (Valor Actual Neto), se entiende que cuando el valor de VAN es positivo el proyecto es viable. Se tiene el valor de IVAN 0,1984 esto quiere decir que por cada 1 dólar invertido en el proyecto se obtiene 1,1954 USD. El TIR del proyecto es de 21%, este es mayor que la tasa de descuento 10% es decir el proyecto es viable.

Figura 91. Flujo de caja acumulado



La figura 96 muestra el año de cambio de -1 a 1 es decir, muestra el punto de cambio de un margen negativo a un positivo, el cual indica el punto de inicio de ganancia proyectado para el proyecto. El periodo de repago es aproximadamente en 2 años de iniciado el proyecto.

Tabla 14. Tasa de descuento Variable

Tasa de descuento TC	0%	0.5%	10%	15%	20%	25%	30%
VAN	1972	1378,643	871,81	435,23	56,296	-274,91	-566,2

En la tabla 14 se muestra los valores netos de cada una de las tasas desde el 0% hasta el 30%. Con esta tabla se muestra la Tasa de Interés de Retorno (TIR) y con la figura 97 se evidencia que la tasa tiende a bajar a cero y así se muestra el valor de la TIR del proyecto.

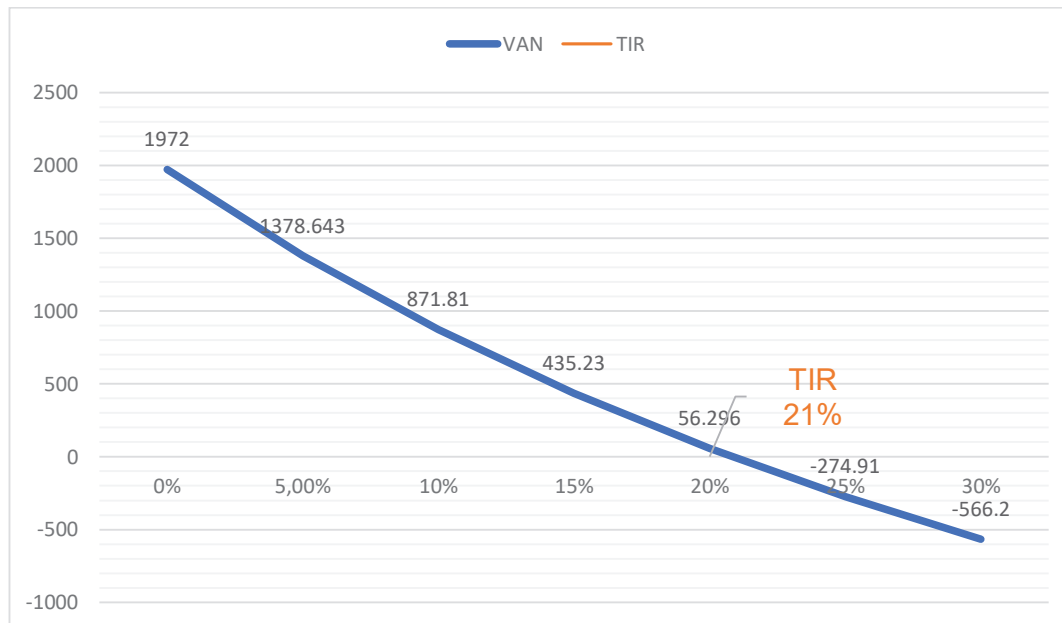


Figura 92. TIR

La TIR de 21% es mayor que la Tasa de descuento TC, lo cual nos muestra que es conveniente ejecutar el proyecto puesto que genera valor agregado y el periodo de repago es aproximadamente 2 años. Es así como se demuestra que el proyecto es viable y con una confianza fenomenal en lo que a cálculos financieros se presenta.

7. Conclusiones y recomendaciones

7.1. Conclusiones

Las características del sistema de streaming implementado permite la integración con diferentes terminales como PCs, TV boxes, o dispositivos móviles, ya que provee de URLs con para varios protocolos de video streaming entre ellos: MP4, HLS, DASH, RTMP, RTSP; los cuales se adaptan a cada tipo de dispositivo según sea el caso. Todos estos tipos de stream trabajan paralelamente si necesidad de realizar ajustes adicionales en el servidor.

La unidad móvil diseñada se integra totalmente con el servidor de streaming implementado, puesto que permite configurar los parámetros de audio y video soportados por el sistema, y dispone de características como codificación h.264 por hardware que aprovecha la GPU del PC para acelerar el proceso de codificación de una transmisión de video en vivo. Las pruebas realizadas permitieron constatar que la latencia máxima en una transmisión en vivo era de 12s con computador portátil de características medias y sobre internet, lo cual es adecuado para un sistema de estas características.

El sistema de streaming que comprende el sistema móvil y el servidor, es ideal para los requerimientos del canal UDLA Channel y transmisión de eventos de interés de la universidad, ya que los elementos seleccionados mostraron ser estables en transmisiones de más de una hora, con diferentes tipos de fuentes de contenido como, video de cámaras en vivo, contenido pregrabado en diferentes calidades, resolución y combinando estas fuentes con la aplicación OBS.

El sistema es escalable puesto que se puede manejar fuentes virtuales como las cámaras de los teléfonos inteligentes. Esto permite que cualquier teléfono inteligente con la aplicación requerida se conecte con el sistema móvil y funcione como fuente de video y pueda ser utilizado en cualquier instante.

El sistema puede escalar cuando así lo requiera puesto que se puede integrar un mezclador de video profesional.

7.2. Recomendaciones

Existen varias alternativas de software para la transmisión de video en red, de las cuales se recomienda el uso de OBS Studio ya que es una herramienta de software libre con bajo consumo de recurso, gran estabilidad y la posibilidad de mezclar múltiples fuentes de audio y video en tiempo real.

Para transmisiones en vivo se recomienda disponer de una conexión hacia el servidor con una velocidad constante de 3Mbps lo cual permite establecer un streaming HD con resolución 720p, se recomienda que la conexión de la unidad móvil sea por un puerto ethernet para asegurar la estabilidad del canal. Evitar las conexiones wifi ya que es un medio de acceso compartido con alto jitter y latencia lo cual afecta a este tipo de aplicaciones.

Se recomienda montar e implementar el servidor de streaming en una Maquina Virtualizada, con el sistema operativo Debian debian 8GB de memoria RAM, ya que con este tipo de características el servidor trabajara sobre el mínimo requerido por el fabricante. También se recomienda una conexión a la red Giga Ethernet para soportar alrededor de 333 streams de video concurrentes. La capacidad de almacenamiento se debe asignar en función de la cantidad contenido pregrabado que se desee almacenar en el servidor.

Se recomienda que en las políticas de la red se permitan los puertos 8080, 1935, 5554 para la transmisión multimedia, y el puerto 4242 para la administración vía Web del servidor. Así como también configurar el enrutamiento necesario para que el servidor de streaming sea accesible desde cualquier lugar de la universidad.

REFERENCIAS

- Aedo, J. (2014). *Sistemas Multimedia: análisis, diseño y evaluación*. Madrid, España: UNED.
- Arrieta, E. (2020). Comparación visual de las diferentes resoluciones de pantalla en alta definición.
- Bartolome, A. (2003). *video Digital*. Huelva: Grupo Comunicar.
- Campos, E. (2005). *Trabajo final de carrera - Consola digital de citófonos*. Obtenido de <http://repositorio.uis.edu.co/jspui/bitstream/123456789/3163/2/118743.pdf>
- Cuenca, P. (1999). *Codificación y transmisión robusta de señales de vídeo MPEG-2 de caudal variable sobre redes de transmisión asíncrona ATM*. Valencia, España: Univ de Castilla La Mancha.
- Dordogne, J. (s.f.). *Redes informáticas nociones fundamentales. (4ta ed.)*. Recuperado el 2 de 10 de 2016, de <https://www.caja-pdf.es/2016/02/25/redes-informaticas-nociones-fundamentales-4ta-edicion/redes-informaticas-nociones-fundamentales-4ta->
- Egan, K. (1991). *La comprensión de la realidad en la educación infantil y primaria*. Madrid, España: Morata.
- Elastixtech. (s.f.). *Protocolo IAX +*. Recuperado el 2016, de <http://elastixtech.com/protocolo-iax/>
- Gualoto, R. (2008). *Diseño e implementación de un sistema de video-vigilancia, en base a un sistema de video embebido, para monitoreo remoto*. Quito.
- Joglar, J. (2009). *Video Compresion*. Barcelona, España.
- Joskowicz, J. (2013). *VOZ, VIDEO Y TELEFONIA SOBRE IP*. Recuperado el 3 de 10 de 2016, de <https://es.scribd.com/doc/305283320/Voz-Video-y-Telefonia-Sobre-IP>
- Marini, E. (2012). Argentina.
- MiraBox. (2019). capturadora MiraBox. *MiraBox*.
- Muñoz, A. (s.f.). *Elastix al Ritmo de Merengue*. Recuperado el 2016, de <http://blogs.elastix.org/libros-sobre-elastix/>

- Pacheco, R. (2018). *Televisión y Nuevas Realidades Tecnológicas y de Mercado: Hacia el sistema de la comunicación Audiovisual*. Colombia: U. Externado de Colombia.
- Pennebaker, W. &. (1992). *JPEG: Still Image Data Compression Standard*. EE.UU: Springer Science & Business Media.
- Perez, E. (s.f.). *Encapsulado y formato de datagramas IP*. Recuperado el 2016, de <http://www.tecnodelinglesalcastellano.com/2011/07/encapsulado-y-formato-de-datagramas-ip.html>
- Rodríguez, J. (2018). *Circuito cerrado de televisión y seguridad electrónica*. Madrid, España: Paraninfo.
- Sendín, A. (2012). *Televisión digital y telecomunicaciones en comunidades de propietarios*. Barcelona, España: Experiencia.
- Tanenbaum, A. (2003). *Redes de computadoras.(4ta ed.)*. México: Pearson Educacion.
- Aedo, J. (2014). *Sistemas Multimedia: análisis, diseño y evaluación*. Madrid, España: UNED.
- Arrieta, E. (2020). Comparación visual de las diferentes resoluciones de pantalla en alta definición.
- Bartolome, A. (2003). *video Digital*. Huelva: Grupo Comunicar.
- Campos, E. (2005). *Trabajo final de carrera - Consola digital de citófonos*. Obtenido de <http://repositorio.uis.edu.co/jspui/bitstream/123456789/3163/2/118743.pdf>
- Cuenca, P. (1999). *Codificación y transmisión robusta de señales de vídeo MPEG-2 de caudal variable sobre redes de transmisión asíncrona ATM*. Valencia, España: Univ de Castilla La Mancha.
- Dordoigne, J. (s.f.). *Redes informáticas nociones fundamentales. (4ta ed.)*. Recuperado el 2 de 10 de 2016, de <https://www.caja-pdf.es/2016/02/25/redes-informaticas-nociones-fundamentales-4ta-edicion/redes-informaticas-nociones-fundamentales-4ta->
- Egan, K. (1991). *La comprensión de la realidad en la educación infantil y primaria*. Madrid, España: Morata.

- Elastixtech. (s.f.). *Protocolo IAX* +. Recuperado el 2016, de <http://elastixtech.com/protocolo-iax/>
- Gualoto, R. (2008). *Diseño e implementación de un sistema de video-vigilancia, en base a un sistema de video embebido, para monitoreo remoto*. Quito.
- Joglar, J. (2009). *Video Compresion*. Barcelona, España.
- Joskowicz, J. (2013). *VOZ, VIDEO Y TELEFONIA SOBRE IP*. Recuperado el 3 de 10 de 2016, de <https://es.scribd.com/doc/305283320/Voz-Video-y-Telefonia-Sobre-IP>
- Marini, E. (2012). Argentina.
- MlraBox. (2019). capturadora MiraBox. *MiraBox*.
- Muñoz, A. (s.f.). *Elastix al Ritmo de Merengue*. Recuperado el 2016, de <http://blogs.elastix.org/libros-sobre-elastix/>
- Pacheco, R. (2018). *Televisión y Nuevas Realidades Tecnológicas y de Mercado: Hacia el sistema de la comunicación Audiovisual*. Colombia: U. Externado de Colombia.
- Pennebaker, W. &. (1992). *JPEG: Still Image Data Compression Standard*. EE.UU: Springer Science & Business Media.
- Perez, E. (s.f.). *Encapsulado y formato de datagramas IP*. Recuperado el 2016, de <http://www.tecnodelinglesalcastellano.com/2011/07/encapsulado-y-formato-de-datagramas-ip.html>
- Rodríguez, J. (2018). *Circuito cerrado de televisión y seguridad electrónica*. Madrid, España: Paraninfo.
- Sendín, A. (2012). *Televisión digital y telecomunicaciones en comunidades de propietarios*. Barcelona, España: Experiencia.
- Tanenbaum, A. (2003). *Redes de computadoras.(4ta ed.)*. México: Pearson Educacion.
- Red Hat, (s.f) *¿Qué es la virtualización?*. Recuperado 7 de mayo de 2020, de <https://www.redhat.com/es/topics/virtualization/what-is-virtualization>

VMware. (s. f.). *Virtualization Technology & Virtual Machine Software: What is Virtualization?* (s. f.). Recuperado 7 de mayo de 2020, de <https://www.vmware.com/solutions/virtualization.htm>

