

Información Importante

La Universidad Santo Tomás, informa que el(los) autor(es) ha(n) autorizado a usuarios internos y externos de la institución a consultar el contenido de este documento a través del Catálogo en línea del CRAI-Biblioteca y el Repositorio Institucional en la página Web de la CRAI-Biblioteca, así como en las redes de información del país y del exterior con las cuales tenga convenio la Universidad.

Se permite la consulta a los usuarios interesados en el contenido de este documento, para todos los usos que tengan **finalidad académica**, nunca para usos comerciales, siempre y cuando mediante la correspondiente cita bibliográfica se le dé crédito al trabajo de grado y a su autor.

De conformidad con lo establecido en el Artículo 30 de la Ley 23 de 1982 y el artículo 11 de la Decisión Andina 351 de 1993, la Universidad Santo Tomás informa que “los derechos morales sobre documento son propiedad de los autores, los cuales son irrenunciables, imprescriptibles, inembargables e inalienables.”

Centro de Recursos para el Aprendizaje y la Investigación, CRAI-Biblioteca

Universidad Santo Tomás, Bucaramanga

**Estudio e Implementación de aplicaciones soportadas en WebRTC que permitan
mejorar la Calidad de Experiencia (QoE) del Audio sobre IP (AoIP)**

Omar Augusto Rincón Rojas

Trabajo de grado para optar el título de Magister en Redes y Sistemas de Comunicación

Director

Tito Raúl Vargas Hernández

**Magister en Tecnologías, Sistemas y Redes de Comunicación en la Especialidad de
Ingeniería Telemática**

Codirector:

Juan Carlos Guerri

Doctor en Telecomunicaciones

Universidad Santo Tomas, Bucaramanga

División de Ingenierías y Arquitectura

Facultad de Ingeniería en Telecomunicaciones

2018

ELABORADO POR:	REVISADO POR:	APROBADO POR:
<hr/> <i>Omar Augusto Rincón Rojas</i>	<hr/> <i>Tito Raúl Vargas Hernández</i>	Comité de posgrado FIT

© Universidad Santo Tomás (USTA) – agosto de 2018

Tabla de Contenido

	Pág.
Lista de apéndices.....	7
Resumen	8
1. Introducción	9
2. Definición del Problema.....	12
2.1. Planteamiento del problema.....	12
2.2. Formulación del problema	13
2.3. Sistematización del Problema	13
3. Objetivos	13
3.1. Objetivo general	13
3.2. Objetivos específicos	14
4. Metodología	14
5. Estado del arte	16
6. Diseño del prototipo de aplicación.....	17
6.1. Especificación de requisitos.....	17
6.1.1. Características de los actores	17
6.1.2. Definición de requisitos	18
6.1.3. Casos de uso.....	19
6.2. Arquitectura.....	20
6.3. Interfaz gráfica de usuario.....	26
7. Análisis de resultados de la evaluación de la calidad del servicio y la experiencia.....	27
7.1. Escenario A.	28
7.2. Escenario B.	30
7.3. Escenario C.	33
8. Conclusiones	35
9. Trabajo Futuro.....	36
10. Referencias bibliográficas	37

Lista de Tablas

	Pág.
Tabla 1. Características de los actores	17
Tabla 2. Requisitos clasificados y priorizados.....	18
Tabla 3. Mensajes de señalización a la medida enviados por el protocolo WebSocket.	25
Tabla 4. Aspectos de calidad de servicio en una red local para el cliente 1.	29
Tabla 5. Aspectos de calidad de servicio en una red local para el cliente 2.	30
Tabla 6. Pruebas con limitación de ancho de banda del cliente 1.	32
Tabla 7. Pruebas con limitación de ancho de banda del cliente 2.	32
Tabla 8. Aspectos de calidad de servicio para pruebas en internet.....	34

Lista de Figuras

	Pág.
<i>Figura 1.</i> Diagrama de casos de uso.....	19
<i>Figura 2.</i> Arquitectura de WebRTC.....	20
<i>Figura 3.</i> Modelo arquitectural triangular de WebRTC.....	21
<i>Figura 4.</i> (a) Diagrama de protocolos usados. (b) Diagrama de protocolos encapsulados en el protocolo WebSocket.....	23
<i>Figura 5.</i> Diagrama de protocolos entre los clientes y el servidor para el escenario de una transmisión de audio exitosa.....	24
<i>Figura 6.</i> Diagrama funcional de la arquitectura del prototipo desarrollado.	26
<i>Figura 7.</i> Vista menú inicial del prototipo de aplicación.	26
<i>Figura 8.</i> Vista menú de sala del prototipo de aplicación.	27
<i>Figura 9.</i> Vista del cambio de códec preferido del prototipo de aplicación.....	27
<i>Figura 10.</i> Topología de pruebas del escenario A.....	29
<i>Figura 11.</i> Topología de pruebas del escenario B.....	31
<i>Figura 12.</i> Topología de pruebas del escenario C.....	33
<i>Figura 13.</i> Mean Opinion Score para cada uno de los códec.....	34

Lista de apéndices

Apéndice A. Anteproyecto.....	39
Apéndice B. Artículo.....	40
Apéndice C. Casos de uso	41
Apéndice D. Código fuente.....	42
Apéndice E. Manual de usuario.....	43
Apéndice F. Evaluación.....	44

Resumen

WebRTC es un nuevo proyecto para estandarizar la comunicación en tiempo de real de audio, video y datos de forma nativa en los navegadores web y aplicaciones móviles, sin el uso de programas adicionales. La medición de la calidad de la transmisión de audio bajo la tecnología que WebRTC ofrece es de importancia de cara a los servicios y aplicaciones que permite crear. Es por esto que en este trabajo se presenta un estudio de la evaluación de la calidad del servicio y experiencia de la transferencia de audio mediante WebRTC por medio de un prototipo de aplicación desarrollado que permite establecer el códec a utilizarse de entre Opus, iSAC y G.722. Se obtuvo información acerca del desempeño en una red local, internet y de ancho de banda limitado de la transmisión de audio mediante WebRTC para cada uno de los códecs y una evaluación de la calidad de la experiencia mediante la medición del Mean Opinion Score (MOS) según la norma ITU-T, en donde se evidencia que el códec Opus es el más preferido por la calidad que ofrece y además requiere un uso de ancho de banda promedio en la relación a los demás códecs considerados.

Palabras claves: WebRTC, streaming de audio, calidad de la experiencia.

1. Introducción

El desarrollo de Internet y su suite de protocolos ha permitido la creación de aplicaciones que conectan computadores en cualquier lugar del mundo, y en donde la demanda de información y contenido requiere de respuestas inmediatas. Los servicios que permiten la participación, colaboración, y comunicación en tiempo real son una necesidad de los consumidores y obligan a los proveedores de soluciones tecnológicas ofrecer servicios de calidad e impulsar el desarrollo de nuevos estándares y un uso eficiente de las tecnologías de red ya implementadas.

Como respuesta a esta demanda, específicamente de contenidos de audio y video, han surgido tecnologías como el Audio sobre IP y WebRTC.

Audio sobre IP permite la distribución de audio digital utilizando la misma infraestructura ya existente de redes IP para Internet y para voz sobre IP (VoIP) [1], [2]. El audio sobre IP ha sido diseñado para proporcionar una alta calidad de audio y tiene aplicaciones actualmente como refuerzo de voz, música de fondo y de primer plano e incluso por las estaciones de radio por Internet, sin embargo, audio sobre IP requiere más ancho de banda que VoIP, lo que lo hace inadecuado para aplicaciones donde el ancho de banda es limitado [3].

Por otra parte, WebRTC (Comunicación en tiempo real para la web), es un nuevo proyecto en desarrollo que tiene código abierto y se basa en estándares abiertos determinados por el W3C (World Wide Web Consortium) y por el IETF (Internet Engineering Task Force) en la RFC 7478, el cual permite estandarizar la comunicación en tiempo real vía audio, video o datos entre navegadores web y aplicaciones móviles de manera nativa mediante el uso de HTML5 y APIs de JavaScript sin necesidad de plug-ins [4], [5].

WebRTC está diseñado para alto rendimiento, una comunicación de alta calidad de video, audio y datos arbitrarios. Las aplicaciones WebRTC necesitan un servicio a través del cual pueden intercambiar metadatos de la red y los medios de comunicación, un proceso conocido

como la señalización. Sin embargo, una vez que ha tenido lugar la señalización, vídeo, audio, datos, se transmiten directamente entre los clientes, evitando el uso de un servidor intermediario [6].

Así, WebRTC suministra comunicación entre el usuario final, es decir el usuario que ejecuta el navegador, y el servidor donde se encuentra alojado el servicio web, transmite la información que permite comenzar una comunicación del cliente con el servidor, facilitando la conexión también con otros clientes que estén accediendo a este mismo servicio web [7].

Debido a la naturaleza de las aplicaciones WebRTC, de ofrecer comunicación en tiempo real, los códecs que utiliza están específicamente diseñados para estos propósitos, es decir, deben tener latencias bajas y cómputo operacional pequeño. Los códecs de audio que utiliza actualmente WebRTC para audio son Opus, iSAC, iLBC y G.722, sin embargo, los navegadores web por ahora solo implementan, según las especificaciones definidas en la RFC 7874, Opus con frecuencia de muestreo de 48 kHz, iSAC con frecuencias de muestreo de 16 y 32 kHz, y G.722.

Hasta ahora, todas las plataformas disponibles de comunicación en tiempo real que incluían capacidad para flujos más allá del texto (i.e. audio, video y datos) estaban basadas en clientes dedicados, o en plug-ins (generalmente java o flash) para el navegador, como es el caso de, Skype, entre otros. Sin embargo, las aplicaciones web basadas en WebRTC no requieren de ninguno de esos complementos, por lo que se podría decir que no existe ninguna otra plataforma que sustituya la utilidad proporcionada por el nuevo estándar propuesto por WebRTC. Es por esta razón que existe una oportunidad importante para establecer información y datos de la calidad del servicio de la arquitectura web basada en WebRTC a pesar de su etapa temprana de desarrollo.

En la literatura existe un interés por obtener información acerca de la calidad de contenidos audiovisuales transferidos usando WebRTC y su impacto en la calidad de la experiencia (QoE) [8], [9], incluso de herramientas que permitan una evaluación automatizada de la calidad de contenido audiovisual en aplicaciones desarrolladas mediante WebRTC [10]. Sin embargo, no existe en la literatura un estudio detallado de la calidad del audio en aplicaciones que usan WebRTC.

Es por todo lo anterior que en este trabajo se presenta un estudio del manejo de audio sobre IP empleando WebRTC que permita mejorar la calidad de experiencia por medio del uso de un prototipo de aplicación. Este documento se divide en 7 capítulos. En el capítulo 2 se define el problema de investigación. El capítulo 3 relaciona el objetivo general y objetivos específicos del trabajo. En el capítulo 4 se describe la metodología utilizada y en el capítulo 5, se hace una revisión al estado del arte. La especificación de requisitos del prototipo de aplicación de transmisión de audio necesario para la realización de pruebas de calidad del servicio y de la experiencia son descritas en el capítulo 6, y a partir de esto se obtienen los casos de uso que describen la funcionalidad necesaria de la aplicación, además se propone la arquitectura del prototipo de aplicación junto con la interfaz gráfica del prototipo de aplicación que se desarrolla. En el capítulo 7 se presenta el análisis de los resultados de la evaluación de la calidad del servicio y de la experiencia en tres escenarios diferentes para finalmente en el capítulo 8 y 9 concluir y comentar el trabajo futuro que se considera interesante abordar.

2. Definición del Problema

2.1. Planteamiento del problema

Hoy en día el entorno exige comunicación en tiempo real, las cuales pueden ser implementadas de una forma más rápida, sencilla y eficaz, esto también debido al avance exponencial de aplicaciones web, estableciendo un escenario ideal para la comunicación por medio de WebRTC.

WebRTC permite efectuar capacidades de comunicación en la web, basándose en aplicaciones de Voz sobre IP (VoIP), así mismo el sistema garantiza el envío de datos como captura de pantalla, textos, imágenes, etc. y provee una interfaz de comunicación al usuario de la web, sin ser necesario la instalación de un software externo, softphones y/o plug-ins [11].

Por otra parte, el audio sobre IP ha demostrado ser la infraestructura de red de difusión de radio y televisión en la actualidad. Es por esto, que un número creciente de emisoras están migrando a las redes IP. Las redes de audio IP son más flexibles, con mayor escalabilidad y tan fiable como las tecnologías de red de más edad.

Teniendo en consideración lo anteriormente expuesto, WebRTC es un nuevo estándar propuesto para la comunicación en tiempo real que promete eliminar las diversas soluciones propietarias existentes en la actualidad que son incompatibles entre sí, sin embargo la calidad de experiencia de los usuarios no ha sido investigada usando esta nueva tecnología, lo cual es la razón por la que se intenta establecer información y datos mediante un estudio en el que se puedan presentar las características de la arquitectura web sustentada en WebRTC que permita que la calidad de experiencia de los usuarios mejore, permitiéndonos dar prioridad al audio sobre IP(AoIP) en tiempo real en aplicaciones que ya existen.

2.2. Formulación del problema

El recurso WebRTC permite la comunicación en tiempo real de Audio sobre IP (AoIP), empleando una aplicación web y garantizando la comunicación punto a punto. ¿Cómo establecemos mejoras en la calidad de Experiencia (QoE) del servicio de audio sobre IP(AoIP) empleando WebRTC?

2.3. Sistematización del Problema

¿En qué consiste el recurso WebRTC y cuáles son sus ventajas?

¿Qué herramientas existen actualmente para analizar el tráfico WebRTC?

¿Cómo seleccionar la herramienta adecuada para la transmisión de datos empleando WebRTC bajo condiciones agrestes de red?

¿Qué tipo de pruebas o escenarios de red se deben realizar para evaluar el desempeño de los flujos de audio sobre IP empleando WebRTC?

¿Cómo interpretar los resultados de los análisis anteriormente mencionadas?

3. Objetivos

3.1. Objetivo general

Investigar la funcionalidad y manejo del Audio sobre IP (AoIP) empleando WebRTC en un ambiente Web de Audio conferencia y con base en el estudio establecer mejoras en la Calidad de Experiencia (QoE).

3.2. Objetivos específicos

- Analizar el estado del arte con respecto a los códec de audio, arquitectura de aplicación, protocolos y algoritmos en el ecosistema de WebRTC.
- Realizar el diseño y despliegue de aplicaciones de audio conferencia utilizando WebRTC sobre diferentes navegadores con el fin de obtener un prototipo funcional y medidas de desempeño utilizando diferentes códecs.
- Proponer una estrategia referente a la calidad de experiencia (QoE), basado en los códecs de audio, sobre una plataforma WebRTC sobre redes gestionadas y no gestionadas.

4. Metodología

Para el desarrollo de este proyecto se plantearon las siguientes fases:

- 1) Revisión del estado del arte,
- 2) Desarrollo de un prototipo de aplicación para la transmisión de audio sobre IP mediante WebRTC.
- 3) Planteamiento del método para la captura de datos y lineamientos,
- 4) Análisis e interpretación de datos,
- 5) Verificación y divulgación de resultados

Fase 1. Permite establecer el estado actual referente a la calidad de servicio para voz sobre IP (VoIP) fundamentada en WebRTC. Para ello se consultó las bases de datos científicas realizando una selección y clasificación de los artículos relevantes para la investigación. Para esta fase se plantearon las siguientes actividades:

- Consulta de artículos.
- Selección de artículos e identificación de referencias y autores.

- Identificación de las variables y métricas.
- Clasificación de métodos para la captura de datos referentes al tráfico de voz sobre redes IP.
- Construcción del estado del arte.

Fase 2. Desarrollo de un prototipo de aplicación que permite la realización de pruebas de evaluación de la calidad del servicio y experiencia de la transmisión de audio sobre IP mediante WebRTC. Para esta fase se plantearon las siguientes actividades:

- Análisis de los requisitos del prototipo de aplicación.
- Diseño del prototipo de aplicación
- Implementación del prototipo de aplicación de transmisión de audio mediante el uso de la API de WebRTC.

Fase 3. Se abordó el problema de calidad de servicio y experiencia para audio sobre IP fundamentada en WebRTC que permite determinar la forma en la cual se va a capturar datos de audio sobre redes IP y las métricas actualmente existentes para este tipo de tráfico. Para esta fase se plantearon las siguientes actividades:

- Análisis del problema de la calidad de servicio para audio sobre IP fundamentado en WebRTC.
- Planteamiento de la arquitectura o modelo para captura de datos.
- Determinación de las métricas actuales del tráfico para este tipo de datos.

Fase 4. Se utilizó el prototipo de aplicación desarrollado para la realización y análisis de pruebas de evaluación de la calidad del servicio y experiencia de la transmisión de audio sobre IP mediante WebRTC. Para esta fase se plantearon las siguientes actividades:

- Configuración y despliegue del prototipo de aplicación desarrollado.

- Configuración de los escenarios a evaluar y captura de datos de la transmisión de audio.
- Procesamiento y análisis de las capturas de datos de los escenarios planteados.

Fase 5. Se realizó la verificación de las diferentes métricas o variables concernientes al tráfico de voz en redes IP con respecto a las métricas establecidas anteriormente. Para esta fase se plantearon las siguientes actividades:

- Verificación de resultados obtenidos por medio de la captura de datos.
- Comparación de los datos con respecto a las variables de tráfico determinadas.
- Evaluar y proponer técnicas que permitan mejorar la calidad de audio sobre IP.
- Divulgación de resultados y entrega del documento final del proyecto de grado.

5. Estado del arte

La nueva tecnología de WebRTC, y la comunicación en tiempo real que estandariza, ha atraído interés debido a las oportunidades que ofrece y el impacto que tiene en las comunicaciones de las empresas y los consumidores en general [12].

La calidad de la experiencia (QoE) es una de esas áreas de WebRTC que atrae interés debido al ambiente competitivo por ofrecer a los consumidores la mejor calidad posible. Existen varios factores que pueden influenciar la calidad de la experiencia al usar servicios de audio y video, como lo son la disponibilidad del servicio, costo, seguridad, entre otros [13], [14]. Sin embargo, la identificación de los factores cruciales que influyen la calidad de la experiencia en transmisiones de contenido audiovisual en diferentes condiciones para comunicaciones mediante WebRTC es una tarea importante que está bajo estudio [9], [15]. Dentro de esos factores considerados, en la literatura para las transmisiones de video en dispositivos móviles se

encuentran el hardware, el ancho de banda disponible, las resoluciones de video, entre otros [8], [16], [17]. A pesar de este esfuerzo de obtener conocimiento acerca de la calidad de experiencia en WebRTC existen pocos trabajos que estudian el audio transmitido mediante WebRTC, y se centran en la evaluación de la calidad de la voz en WebRTC, al estudiar el efecto del escalamiento temporal del tráfico de voz [18], o en comunicaciones inalámbricas por medio de LTE [19], [20], evidenciando la falta de un estudio detallado al respecto que permita a las empresas ofrecer soluciones de audio basadas en WebRTC de calidad que satisfagan las necesidades de los consumidores.

6. Diseño del prototipo de aplicación

En este capítulo se presenta el análisis y diseño de un prototipo de aplicación que permite la realización de pruebas de evaluación de la calidad del servicio y experiencia de la transmisión de audio sobre IP mediante WebRTC.

6.1. Especificación de requisitos

En esta sección se especifican los requerimientos del sistema basado en las necesidades de evaluación de la transmisión de audio entre dos clientes mediante WebRTC.

6.1.1. Características de los actores

Tabla 1. *Características de los actores.*

Actor	Cliente
Tipo	Primario
Características	Representa a los usuarios de la aplicación que transmiten, a través de una red, audio generado por un dispositivo a un cliente remoto. Además pueden seleccionar el códec de audio a utilizarse durante la transmisión, y grabar el audio remoto recibido.

6.1.2. Definición de requisitos

Los requisitos obtenidos de la aplicación son clasificados como lo indica la norma IEEE 830 en la Tabla 2, en requisitos de interfaz, requisitos funcionales y no funcionales, además son priorizados acorde al grado de importancia para el desarrollo del prototipo de la aplicación. Se les asigna números del 1 al 5, siendo 1 el grado de menor importancia y 5 el grado de mayor importancia.

Tabla 2. *Requisitos clasificados y priorizados.*

Número de requisito	Requisito	Clasificación	Prioridad	Tipo de requisito
R1	La aplicación debe transmitir audio bidireccionalmente entre dos computadoras.	Funcional	5	Funcionalidad
R2	La aplicación debe transmitir audio por medio de navegadores web mediante el uso de WebRTC.	No funcional	5	Restricción
R3	La aplicación debe permitir escoger el códec preferido a utilizarse en la transmisión de audio.	Funcional	5	Funcionalidad
R4	La aplicación debe permitir grabar el audio transmitido remotamente por la otra computadora.	Funcional	3	Funcionalidad
R5	Se requiere de un servidor web y de señalización que permita ofrecer la aplicación a los navegadores web de las computadoras que realicen la transmisión de audio.	No funcional	4	Restricción
R6	La aplicación debe funcionar en computadoras conectadas a través de redes de área local e internet.	No funcional	5	Restricción
R7	Se requiere que los navegadores web a utilizarse con la aplicación tengan soporte para WebRTC.	No funcional	5	Restricción
R8	La interfaz de usuario debe ser amigable.	No funcional	3	Uso y factores humanos

Los requisitos se dejaron sujetos a modificaciones e incorporación de nuevos requisitos si se consideraba necesario.

6.1.3. Casos de uso

A partir de la definición de requisitos y las características de los actores del sistema se consideró el desarrollo del prototipo de aplicación. A continuación, en la Figura 1 se muestran los casos de uso para la aplicación desarrollada.

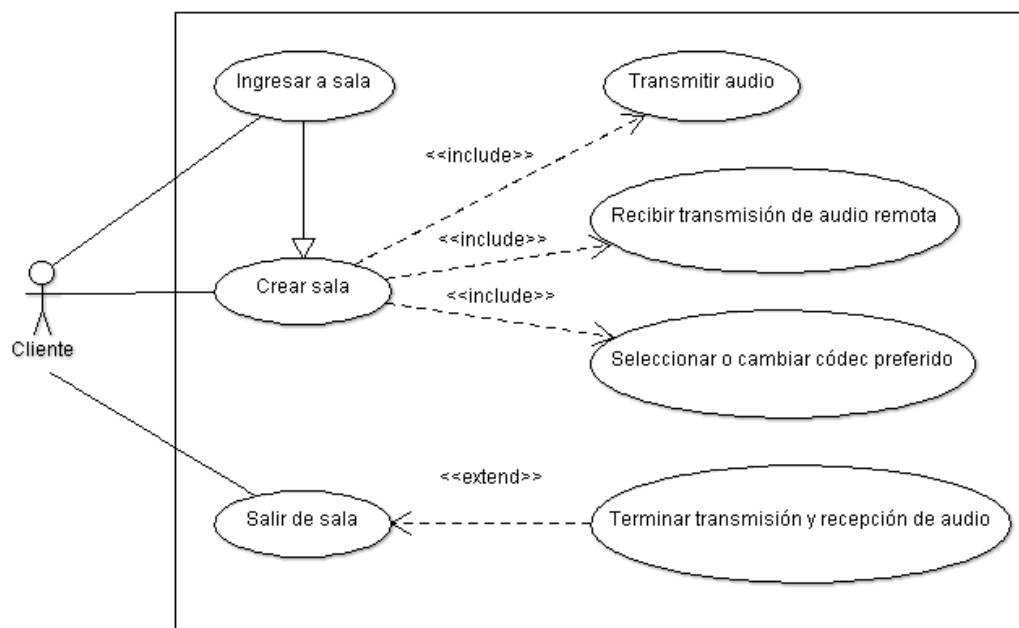


Figura 1. Diagrama de casos de uso.

Las respectivas descripciones de los casos de uso presentados en la Figura 1 se encuentran en el Apéndice C.

6.2. Arquitectura

En esta sección se presenta la arquitectura del prototipo de aplicación utilizado para hacer el estudio de audio sobre IP empleando WebRTC.

Teniendo en cuenta la arquitectura y los protocolos utilizados para la comunicación entre pares mediante WebRTC, se han analizado los recursos necesarios para generar y estudiar tráfico de audio sobre IP generado y transmitido mediante WebRTC.

WebRTC permite a las aplicaciones construidas con su API el acceso al audio en tiempo de real de los dispositivos conectados de los clientes para su transmisión. Sin embargo, se necesita de un mecanismo para coordinar la transmisión y enviar mensajes de control en un proceso conocido como señalización. En la Figura 2 se muestra la arquitectura de WebRTC.

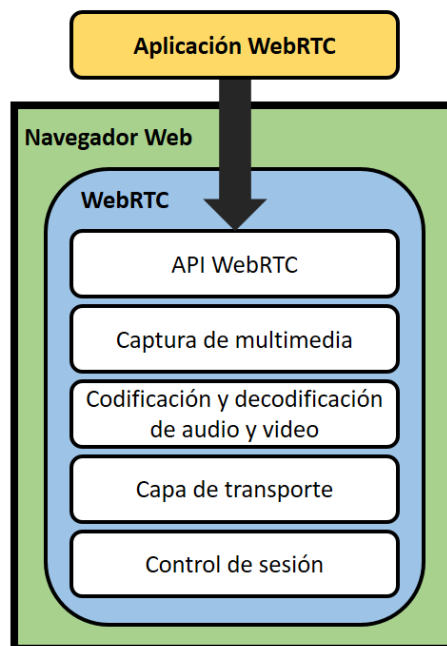


Figura 2. Arquitectura de WebRTC. Adaptado de “Real-Time Communication with WebRTC: Peer-to-Peer in the Browser”, por Loreto S y Romano S, 2014.

Los métodos y protocolos para llevar a cabo el proceso de señalización no son parte de la especificación de WebRTC ya que se considera que es parte de la aplicación.

En la Figura 3 se describe el modelo arquitectural en que se basa el prototipo de aplicación de pruebas desarrollado, que se denomina el modelo triangular de WebRTC [21], el cual se compone de un sitio ofrecido por un servidor web y de señalización que por medio de WebRTC permite la comunicación entre 2 clientes por medio de una red entre pares (P2P, por sus siglas en inglés).

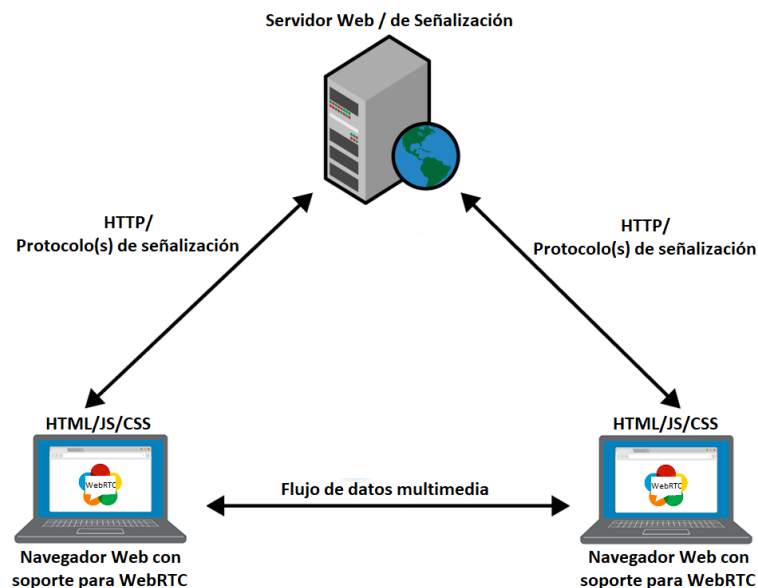


Figura 3. Modelo arquitectural triangular de WebRTC. Adaptado de “Real-Time Communication with WebRTC: Peer-to-Peer in the Browser”, por Loreto S y Romano S, 2014.

La página web del prototipo de la aplicación para la transmisión de audio entre dos clientes fue desarrollada en los lenguajes de programación HTML, CSS y JavaScript y es ofrecida usando el servidor web Nginx. El código fuente del prototipo de aplicación desarrollado se encuentra en el Apéndice D. En el prototipo de aplicación creado, el proceso de señalización fue desarrollado con el entorno de ejecución para JavaScript, Socket.io, el cual está diseñado

para construir aplicaciones en red con contenido dinámico que permite usar JavaScript para programación del lado del servidor y del cliente, y que posibilita el uso del protocolo WebSocket para el intercambio de mensajes de control de sesión, configuración de red de ambos clientes, y de negociación de formato de medios a utilizar en la sesión. El prototipo de aplicación requiere únicamente para funcionar que el cliente acceda a la página web la cual descarga una secuencia de comandos que contiene la programación del lado del cliente y se comunica por medio de Node.js con el servidor para ofrecer contenido dinámico y enviar y recibir los mensajes de señalización.

El prototipo de aplicación permite, mediante la manipulación de los mensajes del protocolo SDP, establecer la preferencia, antes de establecer una transmisión entre los clientes, del códec de audio a utilizar de entre Opus, ISAC y G.722, esto si están disponibles por parte de los navegadores involucrados. Además, permite visualizar que códec está activo en la transmisión, la dirección IP del otro par y su número de puerto, aparte de otros datos estadísticos de la conexión proveídos por la API de WebRTC.

WebRTC mediante el uso de Interactive Connectivity Establishment (ICE) determina la mejor ruta para intercambiar los paquetes de la transmisión. ICE se basa en los protocolos STUN y TURN para esta tarea. En el caso del prototipo de aplicación se usan servidores Session Traversal Utilities for NAT (STUN) de libre uso de Google para resolver el potencial problema de NAT transversal y permitir el descubrimiento entre los clientes, con el fin de establecer la comunicación.

El prototipo de aplicación permite además de usar el protocolo WebSocket para el envío de los mensajes de señalización, de encapsular mensajes de los protocolos ICE y SDP que se intercambian entre los clientes por medio del servidor de señalización. Cuando se establece la comunicación entre los clientes se transmite de manera bidireccional el audio de ambos clientes

codificado en un códec disponible (o en el seleccionado como preferido) en los navegadores web por medio del uso de los protocolos RTP y RTCP.

En la Figura 4 (a) se muestra un diagrama de protocolos del prototipo desarrollado en donde se observan los protocolos involucrados, y en (b) los protocolos encapsulados en el protocolo WebSocket.

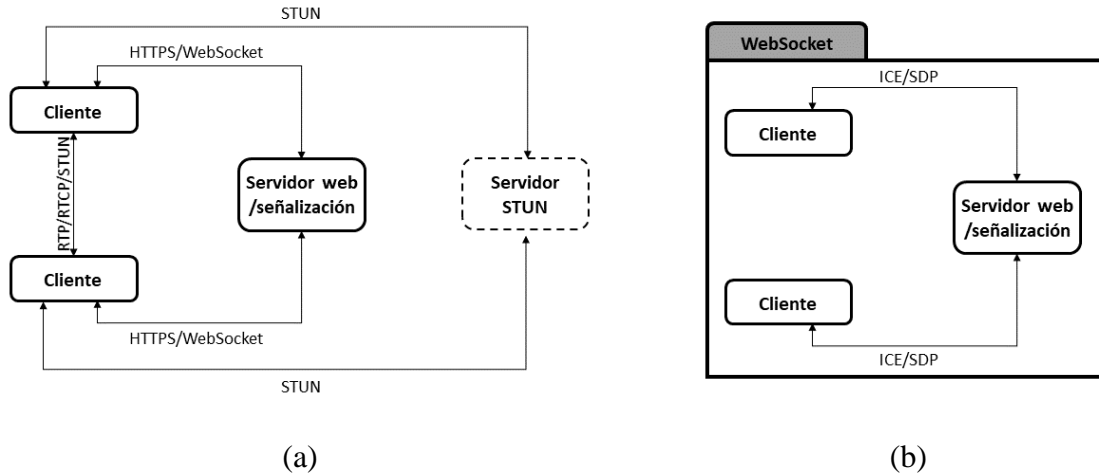


Figura 4. (a) Diagrama de protocolos usados. (b) Diagrama de protocolos encapsulados en el protocolo WebSocket.

En la Figura 5 se puede observar el intercambio de mensajes que ocurre en el escenario de una transmisión de audio exitosa entre dos clientes. Los clientes realizan una petición HTTPS para obtener la página web en sus navegadores web, la cual contiene el prototipo de aplicación, de allí se establece una conexión por medio de Socket.io con el servidor intercambiando mensajes de señalización por medio del protocolo WebSocket los cuales se listan en la Tabla 3. Estos mensajes de señalización que en algunos casos se envían entre clientes mediante la retransmisión por parte del servidor, e incluyen mensajes encapsulados de los protocolos SDP y ICE, permiten coordinar la transmisión de audio y enviar mensajes de control para luego establecer una comunicación entre pares (P2P) entre los dos clientes que pertenecen a una misma

sala. Cuando esta comunicación entre pares ocurre se intercambian paquetes con el audio codificado, encapsulado mediante los protocolos RTP y RTCP, sin intervención alguna del servidor hasta que se envíe un mensaje de señalización que marque el final de la transmisión.

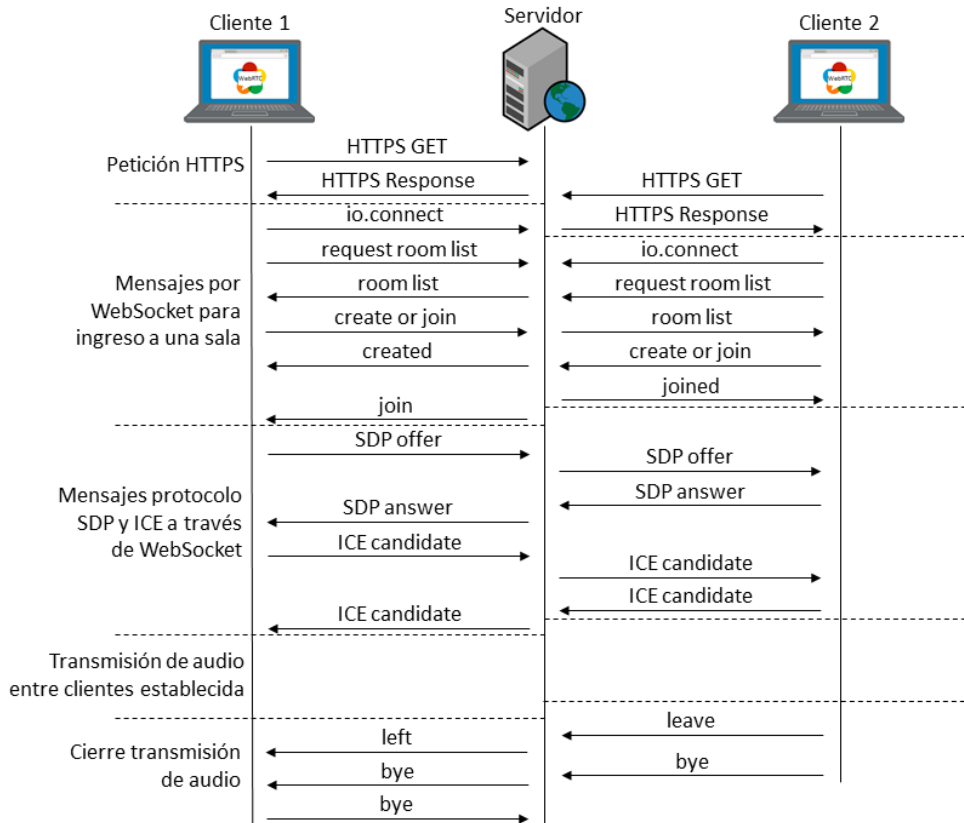


Figura 5. Diagrama de protocolos entre los clientes y el servidor para el escenario de una transmisión de audio exitosa.

Tabla 3. Mensajes de señalización a la medida enviados por el protocolo WebSocket.

Nombre mensaje	Descripción
request room list	Pide la lista de salas creadas en el servidor.
room list	Envía la lista de salas activas en el servidor al cliente.
create or join	Envía una petición de la creación de una sala, o de unirse a ella. Cada sala tiene un límite de dos clientes.
created	Informa de que el cliente es el creador de la sala.
join	Informa de que otro cliente pidió unirse a la sala creada.
joined	Informa de que el cliente se unió a una sala ya existente.
full	Informa de que la sala a la que intenta unirse el cliente está llena.
leave	Petición de un cliente de salirse de una sala a la que pertenece.
left	Informa de que el cliente se salió de la sala especificada.
bye	Marca el final de la comunicación con el cliente remoto.

En la Figura 6 se puede apreciar un diagrama funcional de la arquitectura del prototipo realizado, en donde dos clientes utilizan un navegador web que soporta WebRTC para usar la aplicación realizada en JavaScript y Node.js. Node.js permite a los clientes comunicarse con el servidor web y de señalización para intercambiar información de la sala y obtener información del otro cliente para establecer la comunicación. Y en caso de tener problemas de NAT transversal usan servidores STUN de libre uso para ayudar el descubrimiento de los clientes.

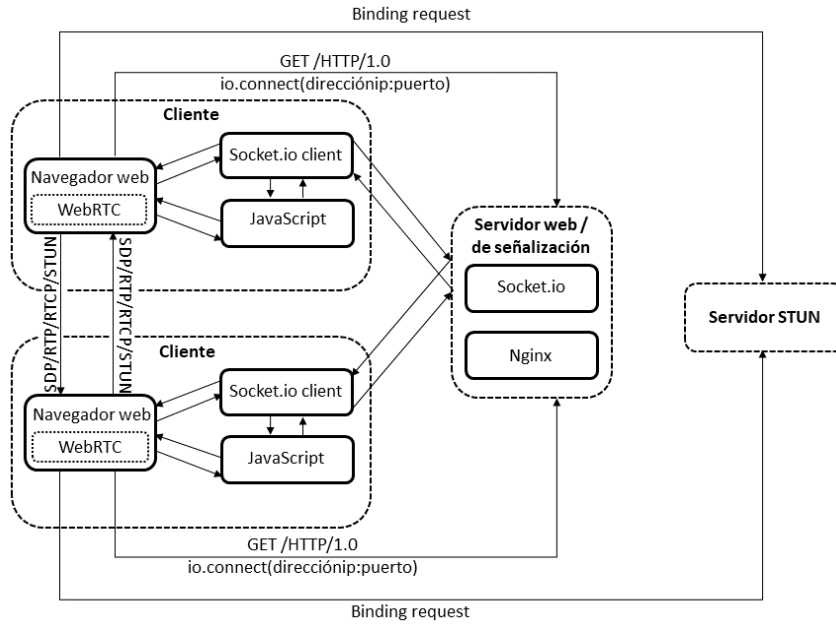


Figura 6. Diagrama funcional de la arquitectura del prototipo desarrollado.

6.3. Interfaz gráfica de usuario

En esta sección se presenta la interfaz gráfica de usuario (Front-end) del prototipo de aplicación desarrollado para hacer el estudio de audio sobre IP empleando WebRTC.

Las Figura 7, Figura 8 y Figura 9 muestran capturas de la visualización final para el usuario del prototipo de la aplicación de pruebas.

Estudio e Implementación de aplicaciones soportadas en WebRTC que permitan mejorar la Calidad de Experiencia (QoE) del Audio sobre IP (AoIP)

Crear una sala

Nombre sala Codec de preferencia a usar

Unirse a una sala

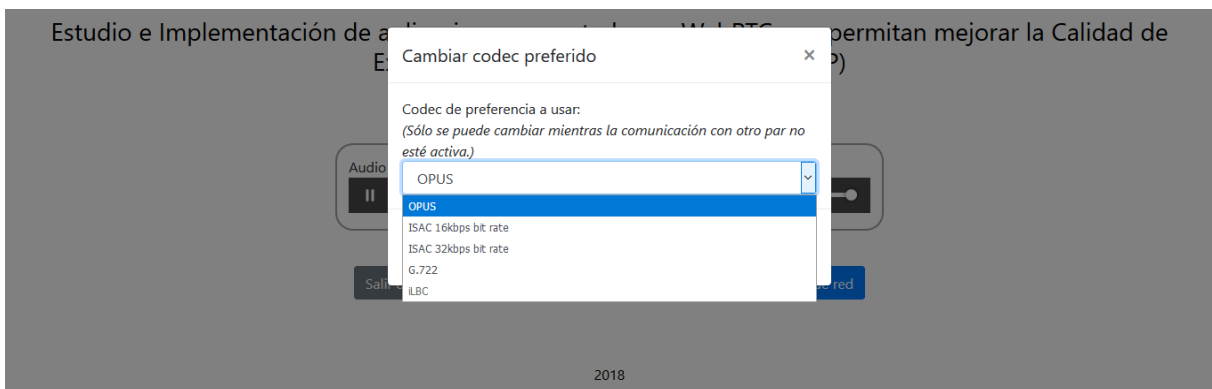
Figura 7. Vista menú inicial del prototipo de aplicación.

Estudio e Implementación de aplicaciones soportadas en WebRTC que permitan mejorar la Calidad de Experiencia (QoE) del Audio sobre IP (AoIP)



2018

Figura 8. Vista menú de sala del prototipo de aplicación.



2018

Figura 9. Vista del cambio de códec preferido del prototipo de aplicación.

El respectivo manual de usuario del prototipo de aplicación de pruebas para la transmisión de audio mediante WebRTC se encuentra en el apéndice E.

7. Análisis de resultados de la evaluación de la calidad del servicio y la experiencia

En este capítulo se presentan resultados del estudio del manejo de AoIP empleando WebRTC. El prototipo de aplicación desarrollado se desplegó con la ayuda de la empresa Vortice Store en su red corporativa en un servidor accesible desde cualquier lugar por medio de internet en la dirección <https://190.254.213.124>. Se utilizó el navegador web Google Chrome

para las pruebas del prototipo de aplicación desarrollado. Se realizaron mediciones de la calidad de servicio (QoS) como el retardo promedio, jitter, paquetes perdidos y tasa de transferencia efectiva (en inglés throughput), y una medición de manera subjetiva en términos del Mean Opinion Score (MOS), para así analizar la relación que existe entre la calidad de experiencia del usuario (QoE) y la calidad de servicio (QoS).

Cabe resaltar que el Mean Opinion Score (MOS) es una prueba subjetiva basada en las recomendaciones de la norma ITU-T, que permite medir la percepción de cada individuo en el rango de 1 a 5, donde 1 es la más baja calidad percibida, y 5 es la percepción de más alta.

Además, para generar un entorno real de interconexión entre los clientes y los servidores, como lo sería Internet, para simular momentos de congestión, cambios de ancho de banda disponible, y provocar pérdidas de paquetes que afecten la calidad del servicio, se ha incluido en esta evaluación la herramienta Dummynet [22].

Esta herramienta se instala y se activa un proceso en los clientes que permite asignar canales de transmisión del tráfico generado, en este caso el tráfico de WebRTC entre los clientes, con la particularidad de tener la posibilidad de controlar el ancho de banda del canal asignado para la transmisión.

Haciendo uso de la topología presentada en la sección anterior, se han configurado y puesto en marcha tres escenarios de transmisión de audio.

7.1. Escenario A.

En una red local se realizaron capturas del tráfico de red en ambos clientes para transmisiones de audio en tiempo real con cada uno de los codecs considerados. Las duraciones de las transmisiones fueron de 5 minutos. En la Figura 10 se puede observar el escenario considerado.

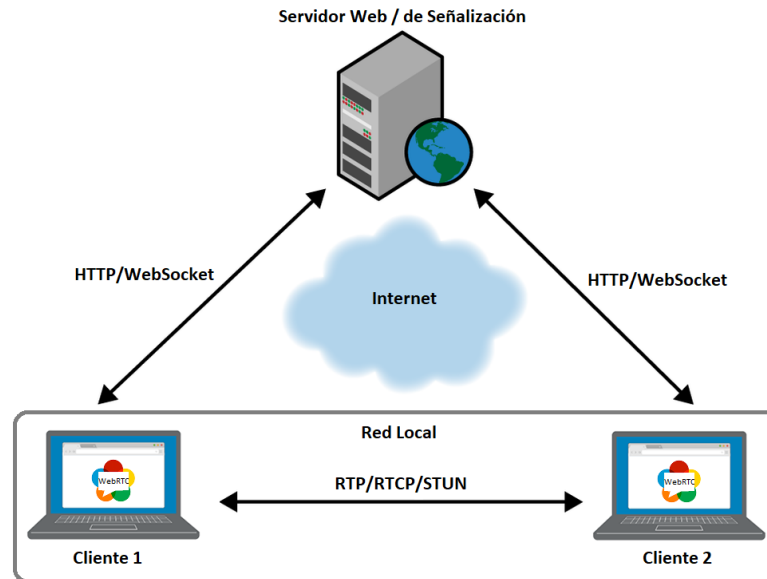


Figura 10. Topología de pruebas del escenario A. Adaptado de “Real-Time Communication with WebRTC: Peer-to-Peer in the Browser”, por Loreto S y Romano S, 2014.

Se presentan a continuación en las Tabla 4 y Tabla 5 los aspectos de calidad de servicio observados para los dos clientes involucrados en una transmisión de audio por medio de la aplicación desarrollada de AoIP con WebRTC.

Tabla 4. Aspectos de calidad de servicio en una red local para el cliente 1.

	CLIENTE 1			
	OPUS	ISAC 16kHz	ISAC 32kHz	G.722
Retardo promedio (ms)	3,905	3,195	5,638	3,998
Paquetes perdidos	0,02%	0%	0%	0,026%
Jitter (µs)	42	62	51	45
Throughput (bytes/s)	7105,7	3387,6	4359,3	11297,5

Tabla 5. Aspectos de calidad de servicio en una red local para el cliente 2.

	CLIENTE 2			
	OPUS	ISAC 16kHz	ISAC 32kHz	G.722
Retardo promedio (ms)	5,460	4,075	3,267	2,497
Paquetes perdidos	0,046%	0,039%	0,108%	0,020%
Jitter (µs)	35	96	6	34
Throughput (bytes/s)	7361,2	5882,4	5663,3	11298,3

Se puede observar en este escenario un buen desempeño y calidad de la conexión con un retardo promedio de paquetes inferior a 6ms y pérdida de paquetes inferior al 0.05%. Además, se puede observar los valores de throughput típicos de cada uno de los códec utilizados en la transmisión de audio.

7.2. Escenario B.

En la red local se realizó una limitación del ancho de banda en uno de los clientes con la herramienta DummyNet, para luego realizar transmisiones de audio en tiempo real con cada uno de los codecs considerados. Esto con el propósito de conocer el efecto en la transmisión de audio de una conexión con ancho de banda limitado.

El ancho de banda se limitó en el rango de 8 a 128 Kilobit/s incrementando el ancho de banda de a 8 Kilobits/s para cada nueva transmisión con cada códec utilizado.

Se observó que al limitar la transmisión por debajo de un valor de ancho de banda específico para cada códec la transmisión de audio se detiene por parte del navegador web. En la Figura 11 se puede observar el escenario considerado.

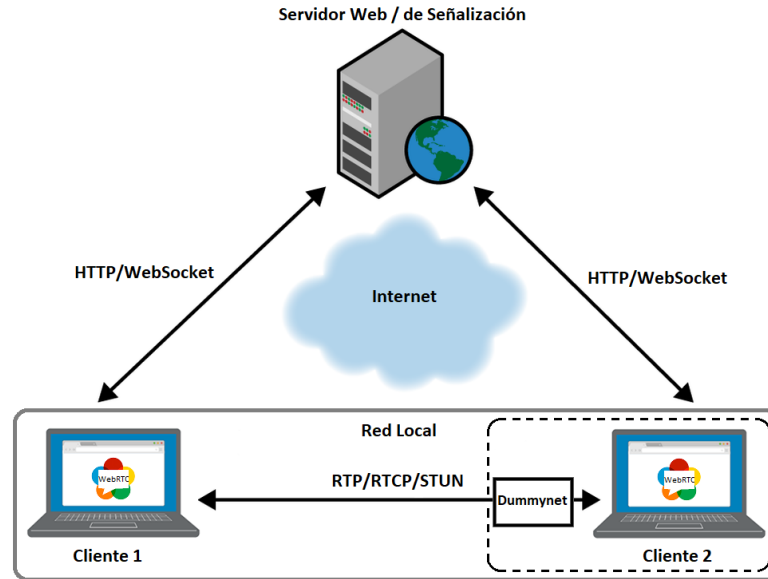


Figura 11. Topología de pruebas del escenario B. Adaptado de “Real-Time Communication with WebRTC: Peer-to-Peer in the Browser”, por Loreto S y Romano S, 2014.

En las Tabla 6 y Tabla 7 se observan las pruebas de transmisión de audio en las que el ancho de banda no es suficiente para mantener que la transmisión de audio se realice por un tiempo considerable, y en donde el cliente 2 realiza la limitación de ancho de banda. Para valores menores de ancho de banda a los considerados en las tablas la transmisión de audio no es posible para cada uno de los códec, y para valores mayores que los considerados la transmisión es satisfactoria. Las transmisiones de audio en la mayoría de los casos en las que no es satisfactoria dura menos de 10 segundos, pero excepcionalmente en otros casos la transmisión puede seguir con un retardo que se incrementa con el tiempo hasta que la transmisión se detiene.

Tabla 6. Pruebas con limitación de ancho de banda del cliente 1.

CLIENTE 1						
Códec	Ancho de banda (Kbit/s)	Retardo promedio (s)	Paquetes perdidos	Jitter (μ s)	Throughput (bytes/s)	Tiempo de transmisión (s)
OPUS	56	0,160	0,068%	33	6822,2	146
	64	0,037	0,036%	37	6862,1	441
	72	0,024	0,295%	39	6876,9	1834
ISAC 16kHz	24	1,224	0,820%	21406	3230,0	2
	32	0,897	0,392%	86	4277,1	6
	40	0,039	0,029%	7	4220,4	1834
ISAC 32kHz	24	1,119	0,794%	72	3241,1	2
	32	0,635	0%	77	4269,0	7
	40	0,215	0,014%	74	4535,3	624
G.722	64	0,722	0,543%	38	8396,6	2
	72	0,667	0,350%	37	9464,0	5
	80	0,529	0,381%	34	10421,7	9

Tabla 7. Pruebas con limitación de ancho de banda del cliente 2.

CLIENTE 2						
Códec	Ancho de banda (Kbit/s)	Retardo promedio (s)	Paquetes perdidos	Jitter (μ s)	Throughput (bytes/s)	Tiempo de transmisión (s)
OPUS	56	0,018	0%	42	7209,7	146
	64	0,046	0,031%	40	7292,3	441
	72	0,032	0,093%	39	7297,9	1834
ISAC 16kHz	24	1,943	28,346%	1452	3111,7	2
	32	1,664	22,753%	233	3977,8	6
	40	1,360	5,236%	81	5071,7	1834
ISAC 32kHz	24	2,243	36,829%	397	3070,4	2
	32	1,634	21,575%	219	4012,6	7
	40	1,361	8,370%	80	5078,6	624
G.722	64	0,999	20,388%	79	8112,4	2
	72	0,831	13,219%	61	9464,0	5
	80	0,662	5,177%	54	10162,6	9

A partir esta información se puede observar que la implementación actual de WebRTC para transmisión de audio no realiza una adaptación al ancho de banda disponible por parte del cliente, además de que los requerimientos de ancho de banda varían para cada códec, Opus requiere aproximadamente de al menos 80 Kbps para funcionar bien, ISAC de 48 Kbps

aproximadamente y G.722 de 88 Kbps aproximadamente. Esto coloca a G.722 como el códec utilizado que más ancho de banda requiere para funcionar satisfactoriamente.

7.3. Escenario C.

Se realizaron transmisiones de audio en tiempo real en internet con cada uno de los códec. Para esto se contó con doce (12) voluntarios los cuales usaron cada uno de los códec para conversar con otro de los voluntarios por una duración aproximada de 2 minutos y al final de la transmisión midieron el MOS de la misma, además se realizaron capturas del tráfico de red en ambos clientes para medir la calidad de servicio. En la Figura 12 se puede observar el escenario considerado.

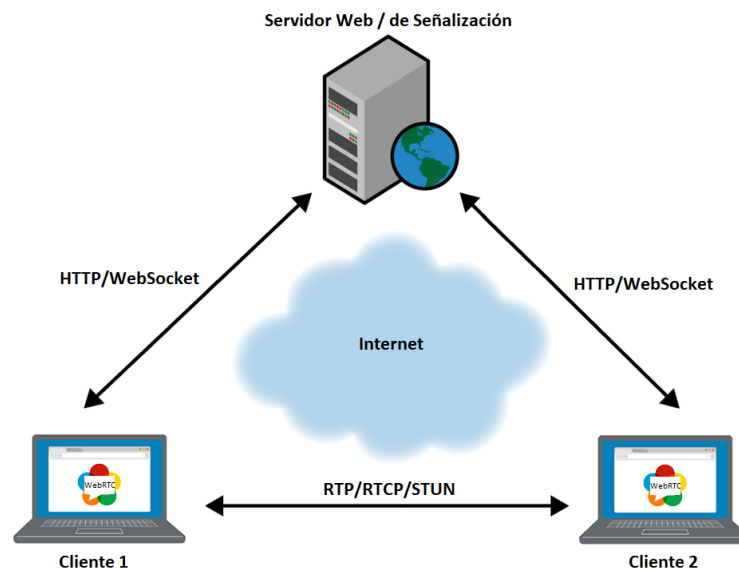


Figura 12. Topología de pruebas del escenario C. Adaptado de “Real-Time Communication with WebRTC: Peer-to-Peer in the Browser”, por Loreto S y Romano S, 2014.

En la Tabla 8 se presentan los datos promedio de la calidad de servicio observada en este escenario para las transmisiones de audio efectuadas por cada uno de los voluntarios.

Tabla 8. Aspectos de calidad de servicio para pruebas en internet.

Número de usuarios	Códec	Retardo promedio (ms)	Paquetes perdidos promedio	Jitter promedio (µs)	Throughput promedio (bytes/s)
12	OPUS	32,38	0,128%	33	7129,8
	ISAC 16kHz	32,95	0,380%	67	5529,3
	ISAC 32kHz	34,64	0,862%	67	5486,4
	G.722	33,75	0,394%	51	11258,1

Además, en la Figura 13 se puede observar el comportamiento de las evaluaciones del MOS para cada una de las transmisiones de audio con los codecs evaluados.

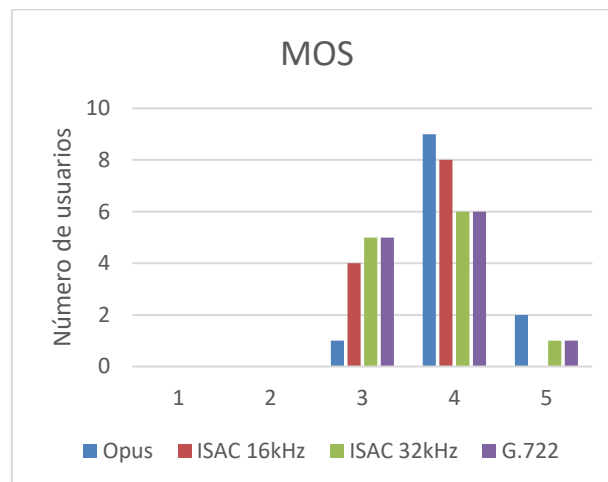


Figura 13. Mean Opinion Score para cada uno de los códec.

Se puede observar que el códec mejor evaluado de los considerados es el Opus y el peor evaluado el ISAC de 16 kHz teniendo en cuenta que los usuarios expresaron la aparición de ecos en la transmisión con dicho códec. El detalle de las evaluaciones presentadas en la Tabla 8 y la Figura 13 se encuentra en el Apéndice F.

8. Conclusiones

Este trabajo permitió investigar la transmisión de audio sobre IP mediante WebRTC al desarrollar un prototipo de aplicación que hiciera posible la realización de pruebas de evaluación de la calidad de servicio y experiencia.

Se analizó el estado del arte de las investigaciones sobre los diferentes aspectos de la tecnología de WebRTC, y en específico de la evaluación de la calidad de la experiencia en WebRTC.

Se desarrolló un prototipo de aplicación que permite la transmisión de audio, mediante WebRTC, entre navegadores web o aplicaciones móviles soportados de dos clientes, que hace posible la realización de pruebas de evaluación de servicio y experiencia utilizando los diferentes códecs de audio implementados.

El prototipo de aplicación desarrollado se desplegó en un servidor accesible por medio de internet que permitió la realización de pruebas bajo un entorno real de interconexión entre los clientes y los servidores.

El prototipo de aplicación desarrollado permitió analizar múltiples escenarios de transmisión de audio empleando WebRTC. El uso de usuarios permitió mediante el uso de las pruebas MOS según la norma ITU-T vislumbrar la preferencia por el códec Opus. La limitación de ancho de banda para la transmisión de audio permitió observar que WebRTC tiene el potencial para desempeñarse bien en redes de área local y corporativas, en donde el ancho de banda disponible supera por mucho los requerimientos mínimos de los codecs para poder funcionar satisfactoriamente, sin embargo, en internet es posible que se presenten momentos de congestión en la red que puedan ocasionar una detención de la transmisión de audio. Estas situaciones deben ser manejadas por las aplicaciones que transmitan audio haciendo uso de la API de WebRTC para detectar una degradación de la conexión y así responder de acuerdo.

9. Trabajo Futuro

La implementación de un prototipo que permita probar las estrategias para manejo de la degradación de la conexión en la transmisión de audio mediante WebRTC. A su vez de una transmisión de archivos de audio por los canales de audio de WebRTC, y la grabación en un archivo de audio de la transmisión de audio recibida remotamente que permita una evaluación con referencia de la calidad de la experiencia.

10. Referencias bibliográficas

- [1] S. Church y S. Pizzi, *Audio Over IP: Building Pro AoIP Systems with Livewire*. Focal Press, 2009.
- [2] P. Stevens y M. Zemack, «Standardising Audio Contribution over IP Communications», *British Broadcasting Corporation*, oct. 2008.
- [3] «A guide to Audio Over IP», 2016. [En línea]. Disponible en: <https://shop.cie-group.com/blog/8/audio-over-i-p-aoip>. [Accedido: 03-jun-2018].
- [4] A. Bergkvist, D. C. Burnett, C. Jennings, A. Narayanan, y B. Aboba, «WebRTC 1.0: Real-time communication between browsers», *Working draft, W3C*, vol. 91, 2012.
- [5] C. Holmberg, S. Hakansson, y G. Eriksson, «Web Real-Time Communication Use Cases and Requirements», *RFC 7478, Internet Engineering Task Force*, p. 29, mar. 2015.
- [6] J. V. Rodrigo Delgado, «Estudio de WebRTC y su implementación con J2EE/JavaEE», Universidad de Valladolid, 2016.
- [7] J. Nightingale, Q. Wang, C. Grecos, y S. Goma, «Modeling QoE for streamed H.265/HEVC content under adverse network conditions», en *5th IET International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN 2013)*, 2013, pp. 249-253.
- [8] D. Vučić, L. Skorin-Kapov, y M. Sužnjević, «The impact of bandwidth limitations and video resolution size on QoE for WebRTC-based mobile multi-party video conferencing», presentado en PQS 2016 5th ISCA/DEGA Workshop on Perceptual Quality of Systems, Berlin, Germany, pp. 59-63.
- [9] K. D. Moor, S. Arndt, D. Ammar, J. N. Voigt-Antons, A. Perkis, y P. E. Heegaard, «Exploring diverse measures for evaluating QoE in the context of WebRTC», en *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, 2017, pp. 1-3.
- [10] B. Garcia, L. Lopez-Fernandez, F. Gortazar, y M. Gallego, «Analysis of Video Quality and End-to-End Latency in WebRTC», en *2016 IEEE Globecom Workshops (GC Wkshps)*, 2016, pp. 1-6.
- [11] S. Pfeiffer, «Video Conferencing in HTML5: WebRTC via Web Sockets», 04-jun-2012.
- [12] A. Johnston, J. Yoakum, y K. Singh, «Taking on webRTC in an enterprise», *IEEE Communications Magazine*, vol. 51, n.º 4, pp. 48-54, abr. 2013.
- [13] K. Brunnström *et al.*, *Qualinet White Paper on Definitions of Quality of Experience*. 2013.
- [14] S. Jelassi, G. Rubino, H. Melvin, H. Youssef, y G. Pujolle, «Quality of Experience of VoIP Service: A Survey of Assessment Approaches and Open Issues», *IEEE Communications Surveys Tutorials*, vol. 14, n.º 2, pp. 491-513, Second 2012.
- [15] D. Ammar, K. D. Moor, M. Xie, M. Fiedler, y P. Heegaard, «Video QoE killer and performance statistics in WebRTC-based video communication», en *2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)*, 2016, pp. 429-436.
- [16] D. Vucic y L. Skorin-Kapov, «The impact of mobile device factors on QoE for multi-party video conferencing via WebRTC», en *2015 13th International Conference on Telecommunications (Con^{TEL})*, 2015, pp. 1-8.
- [17] F. Fund, C. Wang, Y. Liu, T. Korakis, M. Zink, y S. S. Panwar, «Performance of DASH and WebRTC Video Services for Mobile Users», en *2013 20th International Packet Video Workshop*, 2013, pp. 1-8.

- [18] M. Al-Ahmadi, Y. Cinar, H. Melvin, y P. Pocta, «Investigating the Extent and Impact of Time-Scaling in WebRTC Voice Over IP Traffic Under Light, Moderate and Heavily Congested Wi-Fi APs», en *PQS 2016 5th ISCA/DEGA Workshop on Perceptual Quality of Systems*, 2016, pp. 87-91.
- [19] N. Majed, S. Ragot, X. Lagrange, A. Blanc, J. Dufour, y G. Grao, «Experimental evaluation of WebRTC voice quality in LTE coverage tests», en *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, 2017, pp. 1-6.
- [20] G. Carullo, M. Tambasco, M. D. Mauro, y M. Longo, «A performance evaluation of WebRTC over LTE», en *2016 12th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, 2016, pp. 1-6.
- [21] S. Loreto y S. Romano, *Real-Time Communication with WebRTC: Peer-to-Peer in the Browser*. O'Reilly Media, 2014.
- [22] M. Carbone y L. Rizzo, «Dummysnet Revisited», *SIGCOMM Comput. Commun. Rev.*, vol. 40, n.º 2, pp. 12–20, abr. 2010.

Apéndice A.

Anteproyecto. Estudio e Implementación de aplicaciones soportadas en WebRTC que permitan mejorar la Calidad de Experiencia (QoE) del Audio sobre IP (AoIP). (Ver archivo en medio digital)

Apéndice B.

Artículo. Estudio e Implementación de aplicaciones soportadas en WebRTC que permitan mejorar la Calidad de Experiencia (QoE) del Audio sobre IP (AoIP). (Ver archivo en medio digital)

Apéndice C.

Casos de uso del prototipo de aplicación de pruebas de transmisión de audio mediante WebRTC. (Ver archivo en medio digital)

Apéndice D.

Código fuente del prototipo de aplicación de pruebas de transmisión de audio mediante WebRTC. (Ver archivo en medio digital)

Apéndice E.

Manual de usuario del prototipo de aplicación de pruebas de transmisión de audio mediante WebRTC. (Ver archivo en medio digital)

Apéndice F.

Evaluación de la calidad del servicio y de la experiencia para el escenario C. (Ver archivo en medio digital)