

**Pontificia Universidad Católica del Ecuador**

**Facultad De Ingeniería**

**Escuela de Sistemas**



**TEMA:**

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades. Caso de estudio: Estación Científica Yasuní De La PUCE.

**AUTOR:**

CHRISTOPHER ADRIÁN BARRIONUEVO COPPIANO

TRABAJO PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE  
SISTEMAS Y COMPUTACIÓN

**QUITO, JUNIO - 2021**

## **DEDICATORIA**

El presente trabajo dedico a la comunidad universitaria de la PUCE, a la comunidad Waorani del Río Tiputini, A los miembros de la Estación Científica Yasuní que me abrieron las puertas para el proyecto, a mis mentores de carrera de la Escuela de Sistemas que más que una sabiduría, compartieron un estilo de vida en el mundo informático, a mis compañeros de universidad que me enseñaron a ser humilde y compartir los conocimientos adquiridos de forma desinteresada. Finalmente dedico esta disertación a amigos y familiares que estuvieron, que están y siempre estarán en mi vida.

## **AGRADECIMIENTO**

Agradezco a mis padres Orlando y Lida por darme la vida, la crianza y la inculcación de valores, esfuerzo y cariño durante el transcurso de mi vida. A mis hermanos mayores Alejandra y Andres que son mis modelos a seguir, que con su ejemplo han forjado mentalmente y espiritualmente la persona que soy día a día. A mis amigos que han llenado un espacio de afecto en el camino de la vida. Finalmente agradezco a Dios por conocer a mi prometida Daisy y mi hijo Josué que son mi inspiración para superarme.

## RESUMEN

En el presente trabajo de disertación de grado, previo a la obtención del título de ingeniero de sistemas y computación, se manifiesta por la necesidad de mejorar la comunicación remota entre usuarios informáticos de la **Estación Científica Yasuní de la PUCE**, haciendo uso de las Tecnologías de la Información y Comunicación (**TICS**).

El presente trabajo está distribuido en 6 capítulos en los cuales abarca un marco de referencia mencionando la necesidad, marco teórico que sustenta la investigación realizada, la metodología del desarrollo de software empleado en el proyecto, la implementación del proyecto de una aplicación que **Gestione la comunicación por video-conferencia**, haciendo uso de lenguaje de programación, servidor de aplicación, servidor multimedia y herramientas de software libre, y finalmente con las pruebas locales y pruebas de campo.

Una vez culminado el proceso investigativo y desarrollo técnico se procederá con la instalación y uso del personal científico, estudiantil y administrativo de la **Estación Científica Yasuní de la PUCE**.

## Tabla de contenido

|  |    |
|--|----|
| <b>Capítulo 1: Introducción</b> .....                                      | 9  |
| <b>1. Marco de referencia</b> .....  | 9  |
| <b>1.1. Justificación</b> .....  | 9  |
| <b>1.2. Planteamiento del problema</b> .....                               | 10 |
| <b>1.3. Objetivo General</b> .....   | 11 |
| <b>1.4. Objetivos Específicos</b> .....                                    | 12 |
| <b>1.5. Antecedentes</b> .....   | 12 |
| <b>1.6. Alcance</b> .....  | 14 |
| <b>Capítulo 2: Fundamentación teórica</b> .....                            | 14 |
| <b>2. Marco Teórico</b> .....  | 14 |
| <b>2.1. Internet satelital</b> .....                                       | 14 |
| <b>2.2. Video Conferencia</b> .....  | 20 |
| <b>2.2.1. Generalidades</b> .....  | 20 |
| <b>2.2.2. Definición</b> .....   | 20 |
| <b>2.2.3. Antecedentes</b> .....   | 21 |
| <b>2.2.4. Protocolos y clasificación a utilizar videoconferencia</b> ..... | 22 |
| <b>2.2.5. Tecnología síncrona</b> .....                                    | 23 |
| <b>2.2.6. Tecnología asíncrona</b> .....                                   | 23 |
| <b>2.3. Streaming</b> .....  | 24 |
| <b>2.3.1. Generalidades</b> .....  | 24 |
| <b>2.3.2. Definición</b> .....   | 24 |
| <b>2.3.3. Antecedentes</b> .....   | 26 |

|   |    |
|---|----|
| 2.3.4. Protocolo RTMP y RTCP .....                    | 27 |
| 2.3.5. Otros protocolos para el uso de Streaming..... | 27 |
| 2.4. Servidor Media .....                             | 28 |
| 2.4.1. Red5.....                                      | 28 |
| 2.5. Herramientas de desarrollo.....                  | 29 |
| 2.5.1. Red5 pro SDK.....                              | 29 |
| 2.5.2. JavaScript. ....                               | 30 |
| 2.5.3. PHP .....                                      | 30 |
| 2.5.4. Laravel.....                                   | 30 |
| 2.5.5. MySQL.....                                     | 32 |
| 2.5.6. Angular .....                                  | 32 |
| 2.5.7. Guías de diseño.....                           | 33 |
| Capítulo 3: Metodología de Desarrollo.....            | 34 |
| 3. Método y Metodología ágil para el desarrollo.....  | 34 |
| 3.1. Modelo Evolutivo-Espiral.....                    | 34 |
| 3.2. Scrum.....                                       | 38 |
| 3.3. Diseño .....                                     | 41 |
| 3.3.1. Diseño Casos de Uso.....                       | 41 |
| 3.3.2. Diseño conceptual.....                         | 42 |
| 3.3.3. Diseño de arquitectura infraestructura .....   | 42 |
| 3.3.4. Diseño de implementación .....                 | 43 |
| 3.3.5. Diagrama de Actividades.....                   | 44 |
| 3.3.6. Prototipo.....                                 | 45 |

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

|  |    |
|--|----|
| <b>Capítulo 4: Implementación</b> .....  | 48 |
| <b>4. Implementación de la aplicación.</b> .....   | 48 |
| <b>4.1. Implementación de interface.</b> .....   | 48 |
| <b>4.2. Implementación de manejo de sesiones de usuario.</b> .....   | 50 |
| <b>4.3. Implementación de Streaming usando el protocolo RTMP, RTCP y WS con Red5.</b> .....                          | 55 |
| <b>4.4. Implementación del manejo de mensajería.</b> .....   | 58 |
| <b>4.5. Utilización del servicio de Google para búsquedas de artículos científicos o de carácter técnicos.</b> ..... | 59 |
| <b>Capítulo 5: Pruebas</b> .....   | 62 |
| <b>5. Pruebas</b> .....  | 62 |
| <b>5.1. Pruebas funcionales.</b> .....   | 62 |
| <b>5.2. Pruebas no funcionales.</b> .....  | 63 |
| <b>5.2.1. Pruebas de carga.</b> .....  | 63 |
| <b>5.2.2. Pruebas de stress.</b> .....   | 67 |
| <b>Capítulo 6: Pruebas de campo</b> .....  | 68 |
| <b>6. Pruebas en la localidad del caso de estudio</b> .....  | 68 |
| <b>6.1. Pruebas funcionales.</b> .....   | 69 |
| <b>6.2. Rendimiento de la aplicación.</b> .....  | 72 |
| <b>6.3. Conclusiones</b> .....   | 72 |
| <b>6.4. Recomendaciones.</b> .....   | 73 |
| <b>6.5. Bibliografía.</b> .....  | 74 |
| <b>6.6. Glosario de términos.</b> .....  | 75 |

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

|                          |           |
|--------------------------|-----------|
| <b>6.7. Anexos</b> ..... | <b>77</b> |
|--------------------------|-----------|

## **Capítulo 1: Introducción**

### **1. Marco de referencia**

#### **1.1. Justificación**

El internet se caracteriza por ser “una red de computadoras es en esencia un conjunto de computadoras interconectadas entre sí para intercambiar información” (López, 2017) que con este medio se puede transferir, intercambiar y compartir datos de forma remota. Sin embargo “Son muchas las comunidades que están privadas de acceso a la banda ancha por una discriminación del ADSL (Asymmetric Digital Subscriber Line)” (Muñoz, 2006) especialmente en la Amazonía ecuatoriana.

Esta “brecha digital y geográfica implica que los pueblos se distancien los unos con los otros debido a su zona de difícil acceso” (Yolanda Martínez, 2015). Pero por su riqueza en fauna y cultura son motivos de estudio como es el caso del parque nacional Yasuní. Un área protegida por los nativos del lugar y por el gobierno ecuatoriano ubicado en la Provincia de Orellana.

“En 1994 se crea la Estación Científica Yasuní, centro primordial de investigaciones y descubrimientos de la PUCE situado en la orilla derecha del Río Tiputini” (PUCE, 2017). Y como centro de investigación es factible poseer conexión a Internet para facilitar los estudios.

Una solución de conexión fue el uso de Internet vía Satélite. Este tipo de acceso se adapta para solventar en:

- “Áreas con baja densidad de población, en las que soluciones terrestres no llegan” (Grasa, 2017).

- “Áreas pobladas en las que no es técnicamente viable una solución de banda ancha de tipo ADSL o de Cable Modem por distancia de proveedor o por carencia de infraestructura” (Grasa, 2017).

Este trabajo de disertación se justifica en el desarrollo de un sistema que gestione Streaming que podrá ayudar a establecer una forma de comunicación satelital entre los biólogos del centro Yasuní con otras entidades, facilitando la transmisión de actividades y eventos en tiempo real ayudando de manera remota a los investigadores huéspedes de la estación con otras entidades. Además, servirá como base para el futuro análisis socio-cultural de la región de forma remota.

## **1.2. Planteamiento del problema**

Conectarse a Internet hoy en día se ha vuelto una necesidad más que una recreación. Necesidad en el ámbito investigativo por el cual se necesita un buen ancho de banda y que su conexión sea continua. Sin embargo al “utilizar Internet satelital en áreas remotas, la conexión será afectado por condiciones climatológicas” (Lagatree, 2006).

“Las conexiones satelitales utilizan frecuencias de radio microondas que viajan en línea recta y estas no pasan a través de objetos sólidos” (Richard, 2017). El clima afecta el aire, que está siempre presente, entre el plato de la antena y el satélite. “No obstante la humedad reduce la señal” (Richard, 2017). Por ende, mientras más fuerte sea la señal recibida, más inmune esta la señal a las condiciones climatológicas.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.

Caso de estudio: Estación Científica Yasuní De La PUCE

Dado que la estación científica se encuentra en la Amazonía ecuatoriana, la región es de selva tropical donde el clima es caluroso y húmedo. En temporada de lluvias las precipitaciones son paulatinas que pueden durar días enteros.

Según Richard “Las lluvias ligeras no afectan el servicio de Internet satelital, pero si el disco no está correctamente alienado con el satélite, la señal no soportará el desvanecimiento de la misma.” Además, mantiene que las tormentas y lluvias pesadas pueden bloquear toda la señal del servicio de Internet.

Por lo tanto, se plantea el presente trabajo de disertación de grado, dado el nivel de inconveniente que presenta La estación científica Yasuní de la PUCE a los factores climáticos, la ausencia de técnicos y de un sistema propio de comunicación directa en tiempo real. Para solventar tal requerimiento se pretende presentar un prototipo de comunicación vía streaming utilizando el servidor de Red5 el cual nos facilita utilizar transmisiones en formatos Flash que es un formato liviano.

### **1.3. Objetivo General**

- Implementar un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.

#### **1.4. Objetivos Específicos**

- Documentar los beneficios e inconvenientes de utilizar Internet satelital en la región amazónica.
- Analizar herramientas que faciliten la transmisión de datos multimedia para la implementación adecuada de la aplicación.
- Aplicar una metodología de desarrollo que se ajuste a la implementación de la aplicación.
- Implementar un sistema basado en metodologías de desarrollo que gestione streaming utilizando el protocolo RTMP y que permita hacer uso de las herramientas de Google para la búsqueda de artículos científicos.
- Validar el sistema con pruebas.

#### **1.5. Antecedentes**

Desde 1927 las tecnologías de conferencia comenzaron su pleno desarrollo, desde tecnologías unidireccionales de televisión y su empujón a las tecnologías de transmisión bidireccional utilizando receptores de audio y video en ambos extremos en 1930.

Estos hitos históricos fueron esenciales para el paulatino desarrollo de nuevos prototipos de video conferencia, como fue el PICTURE-PHONE creado por Bell Labs 1956 (prototipo con un ancho de banda de 1 MHz y transmisión de imagen cada 2 segundos). Sin embargo, estos prototipos históricamente no

tuvieron una fuerte acogida debido a sus limitaciones en cuanto a pantalla y falta de información por los usuarios (En aquella época a la gente no les gustaba aparecer mientras hacia una llamada.).

A pesar de tal inconformidad por parte de la gente, el desarrollo de las videoconferencias fue necesario para el ámbito empresarial “para impulsar la interoperabilidad de salas publicas entre países” (Ordoñez, 2009). Una gran parte de sistemas de videoconferencias en los años 70s fueron implementados con la aparición de nuevos protocolos de empaquetamiento de audio y video como NVP (Network Voice Protocol) y PVP (Packet Video Protocol), a pesar de su capacidad el software que solventaba esta solución resulto ser caro para el público “250 000\$ sistema y 1000\$/hora” (Ordoñez, 2009).

Fue preciso que en 1993 los sistemas de videoconferencia fueron adaptándose a los ordenadores de escritorios con la nueva incorporación de tarjetas de red, audio y video “Esto permitía a los usuarios ejecutar aplicaciones como si estuviesen en su equipo, añadiendo una funcionalidad muy útil en las herramientas de trabajo colaborativo” (Ordoñez, 2009).

Desde el siglo XX hasta la actualidad, con la masificación de redes IP de Internet, en conjunto a nuevos protocolos poco a poco se insertó en la parte domestica como empresarial con costes cada vez más baratos.

En el presente plan de disertación año 2017 se requiere un proceso similar al video conferencia por el cual se hará uso de las redes IP para la transmisión síncrona de streaming para la Estación Científica Yasuní.

## **1.6. Alcance**

El presente estudio e implementación de la aplicación tiende una investigación exploratoria para el análisis teórico y aplicación práctica de las herramientas de desarrollo que nos facilitarán la comunicación vía Streaming. Además de ir analizando las guías de diseño para el entorno Web. Con tal conocimiento se prevé implementar el sistema en el entorno Web para dispositivos de escritorio. El sistema estará albergado en los servidores de la Facultad De Ingeniería en el laboratorio de LTIC facilitando el acceso al sistema a través de Internet. Tal investigación se realizará con estudios descriptivos y correlacionales. En los descriptivos se buscará las propiedades y característica de la conexión satelital que posee la Estación Científica Yasuní. En los correlacionales se conocerá el aporte de la aplicación en la comunidad científica de biólogos, tanto de la PUCE como con otras entidades y verificar el rendimiento de la aplicación localizado en el caso de estudio.

## **Capítulo 2: Fundamentación teórica**

### **2. Marco Teórico.**

#### **2.1. Internet satelital**

##### **2.1.1. Generalidades**

“Internet por satélite ha resurgido en los últimos años como una alternativa viable contra el consumo de tiempo y energía, de armar miles de kilómetros de cable de fibra óptica. Basados en constelaciones de órbitas no sincrónicas innovadoras de satélites pequeños producidos en masa que reducen los

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

costos de producción y mejoran la latencia para competir mejor con la banda ancha, las nuevas empresas relacionadas con Silicon Valley, como OneWeb, SpaceX y O3b, han prometido acceso a Internet de alta velocidad y barato. alrededor del mundo” (Graydon, 2019). Al mejorar la latencia y reducir los costes de instalación promete ser la alternativa ideal para sectores remotos como en la Amazonía o sectores ambientales mundialmente protegidos.

La conexión a Internet satelital utiliza un satélite geostacionario, posicionado en órbita terrestre, encargado de por frecuencia proveer paquetes de información entre usuarios y servidores. En este medio se maneja 2 canales de enlace:

- Enlace de subida (up-link): se define una señal que viaja desde un nodo terrestre hacia el satélite en conexión.
- Enlace de bajada (down-link): aquella señal que baja desde el satélite a los nodos receptores terrestres.

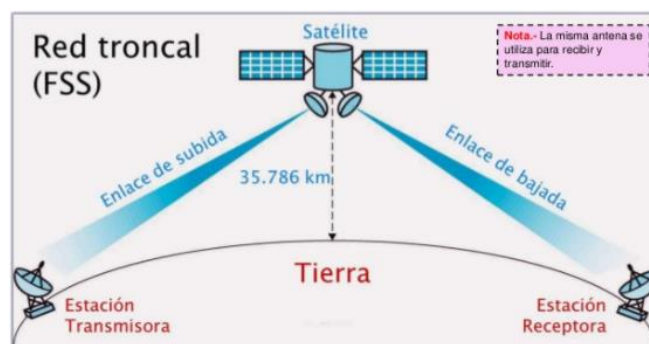


Figura extraída de presentación digital. (G., 2011)

### **2.1.2. Componentes terrestres**

**Antena:** “Dispositivo de aparato emisores o receptores, que sirve para emitir o recibir ondas electromagnéticas”. (REAL ACADEMIA ESPAÑOLA, 2020).  
Dicho esto, las antenas para su correcto funcionamiento deben estar en sincronía con el satélite para tener una conexión sin ruido y con velocidades aceptables.

**Módem:** “Aparato que convierte las señales digitales en analógicas y viceversa”. (REAL ACADEMIA ESPAÑOLA, 2020). Se encarga de enviar/recibir señales analógicas a la antena y receptor hacia un medio digital.

**Tele puerto:** “Estación conformada por varias antenas de radio frecuencia para tratar los servicios de telecomunicaciones por televisión, voz y datos”. (AXESSNET, 2020)

**Centro de operaciones de red:** “Instalaciones que gestionan el control de múltiples conexiones satelitales. Se encargan de monitorear y dar soporte a las señales provenientes del satélite.” (AXESSNET, 2020)

### 2.1.3. Componentes espaciales

**Satélite geoestacionario:** “Vehículo espacial, tripulado o no, que se coloca en órbita alrededor de la Tierra o de otro astro, y que lleva aparatos apropiados para recoger información y transmitirla”. (AXESSNET, 2020).

**Transpondedor:** “Dispositivo con el fin de recibir y enviar información a cualquier espacio que se le ordene, se encarga de limpiar el ruido, amplifica la señal y desplaza la frecuencia”. (AXESSNET, 2020)

**Backbone (Columna Vertebral):** “Principales conexiones de internet, donde se transmiten a altas velocidades”. (AXESSNET, 2020). Generalmente se le conoce como la conexión directa interoceánica internacional.

### 2.1.4. Conexión unidireccional.

#### 2.1.4.1. Definición.

Se refiere al enlace de red que garantiza solo el recibimiento de datos a partir de un nodo emisor para lograr enviar y recibir datos desde Internet o medio de comunicación. (AXESSNET, 2020)

#### **2.1.4.2. Ventajas.**

- Los proveedores de telecomunicaciones ofertan servicios dedicados de acuerdo a la necesidad de los usuarios, es decir, si un usuario necesita televisión por cable, éste necesitará solo la instalación y propagación de canales de televisión.
- La administración de ancho de banda es fácil de manejarlo.

#### **2.1.4.3. Desventajas.**

- Este tipo de conexión requiere de conexiones terrestres adicionales ya sea por medio telefónico, cable coaxial, cable de red o por cable de fibra óptica.
- Depende del sitio topográfico para su instalación.
- Los canales de difusión suelen saturarse a mayor demanda.
- Si existe daños físicos en los equipos en el cableado estructurado la cobertura zonal se quedará incomunicado.

#### **2.1.5. Conexión bidireccional.**

##### **2.1.5.1. Definición.**

Enlace de red que es capaz de recibir y enviar datos sin una conexión adicional. Los datos viajan de forma inalámbrica. (AXESSNET, 2020).  
Generalmente esto hace referencia a una conexión netamente satelital, en

donde el equipo digital se conecta a un enrutador que lo comunica con la antena que está en sincronización con el satélite geoestacionario.

#### **2.1.5.2. Ventajas.**

- No depende de soluciones de cableado estructurado.
- Rápida instalación.
- El soporte y restablecimiento de red es rápido y fácil.

#### **2.1.5.3. Desventajas.**

- Presenta mayores retardos y pérdidas en la transmisión de paquetes, por lo que su latencia es mayor.
- El ancho de banda es inferior a una solución cableada.
- La eficacia depende de la situación meteorológica de la zona con cobertura.

#### **2.1.6. Desempeño bajo las condiciones climáticas.**

“En condiciones meteorológicas cuando el espacio está congestionado por nubes provoca ruido en los paquetes informáticos, ya que interfiere la con la recepción y envío de los haces de luz, así mismo los fuertes vientos pueden desajustar el ángulo de la antena provocando ruido en la comunicación”. (Maria, 2019) Sin mencionar que las constantes lluvias pueden oxidar los equipos de recepción y transmisión.

## **2.2. Video Conferencia**

### **2.2.1. Generalidades.**

La videoconferencia es un marco genérico por el cual permite la transmisión de información simultánea entre dos puntos dispersos geográficamente (Ordoñez, 2009). En este marco se maneja sesiones de control para el inicio y fin de las sesiones. Basado en un software informático haciendo uso de los protocolos de **streaming** o flujo de datos, posibilita las sesiones de los participantes durante tiempo indefinido o pre establecido.

### **2.2.2. Definición.**

La videoconferencia conforma un conjunto de hardware, software, personas, redes de comunicación y técnicos con el cuál se establecen reuniones, cursos, educación evitando gastos de transporte y tiempo de personas.

Entre la comunicación simultánea tenemos: Tecnología síncrona, Comunicación en tiempo real y comunicación bidireccional. Lo cual será la base para el proyecto de investigación.

Para una observación más detallada de los componentes véase la tabla 1.

| <b>Componentes<br/>síncronos</b> | <b>Componentes asíncronos</b>  |
|----------------------------------|--------------------------------|
| Audio conferencia                | Correo electrónico.            |
| Videoconferencia                 | Directorio de contactos        |
| Editor de textos                 | Calendario                     |
| Mensajería instantánea.          | Base de datos de conocimiento. |
| Juegos síncronos.                | Software social.               |

*Tabla 1 Extraída de Videoconferencia (Ordoñez, 2009).*

### **2.2.3. Antecedentes.**

Partiendo de la comunicación radio frecuencia, telefónica, televisiva, grabaciones de audio y video dirigió al mundo moderno a fundir los pilares de las telecomunicaciones.

“Pero la difusión de solo video o solo audio presenta un gran inconveniente cuando dos o más personas desean comunicarse, y es su poco grado de interacción. El software de videoconferencia suple esa carencia permitiendo una comunicación interactiva, bidireccional y en tiempo real entre dos o más personas mediante la transmisión de video, audio y datos.” (Eduardo Polo

Ortega, 2015). De esta manera se lograría simular la presencia física de los participantes al momento de estar en comunicación instantánea.

#### **2.2.4. Protocolos y clasificación a utilizar videoconferencia**

Entre los protocolos de comunicación tenemos los más representativos: RTSP, RTP, RTCP y RTMP que son reglas informáticas de flujo de datos o **streaming** en tiempo real. Véase más al detalle en el capítulo streaming sección protocolo [RTCP, RTMP](#) y [otros protocolos para el uso de streaming](#).

En la clasificación de videoconferencia varía de acuerdo a los dispositivos que intervienen en la sesión, por ejemplo:

**Punto a punto:** 2 dispositivos en una misma red donde puede intervenir uno o grupo de personas donde comparten contenido comprimido de audio y video. (Eduardo Polo Ortega, 2015)

**Multipunto o multiconferencia:** Se utiliza más de 2 dispositivos que participan en la sesión, compartiéndose contenido comprimido de audio y video. (Eduardo Polo Ortega, 2015)

### **2.2.5. Tecnología síncrona**

Conjunto de teorías y técnicas con de carácter científico que aprovecha los recursos para coincidir en un mismo tiempo. (REAL ACADEMIA ESPAÑOLA, 2020). Dicho de esta manera en el mundo informático, existe software especializado en establecer dicha sincronía para cumplir con una necesidad dada. Entre los más destacados softwares con tecnología síncrona tenemos:

- Software de videoconferencia como: Skype, Google Meeting, Zoom, etc.
- Software de mensajería o chat instantáneo como: Whatsapp, Messenger, Telegram, etc.
- Software de juegos en línea o multijugador como: Minecraft, Fornite, Warcraft, etc.
- Software de llamadas telefónicas como: Skype, True Caller, Clever Dialer, etc.

### **2.2.6. Tecnología asíncrona.**

Al igual que la tecnología síncrona solo que el recurso no coincide en el mismo tiempo, pero está disponible al ser requerido por un cliente. También existe software que corrobora esta tecnología como es el caso de:

- Software de Correo Electrónico: Outlook, Yahoo!, etc.
- Software de Redes Sociales: Facebook, Instagram, etc.
- Software de Noticias: Las Wikis, Blogs, Podcast, etc.

## 2.3. Streaming

### 2.3.1. Generalidades.

“Una forma de proporcionar el servicio de audio y video es el streaming o flujo (Español), el cual se caracteriza por ajustar contenidos a transmitir y adecua el ancho de banda necesario para su funcionamiento. En su transmisión realiza descargas temporales de contenido audio y video, que tras su reproducción se eliminan.” (Eduardo Polo Ortega, 2015)

### 2.3.2. Definición.

Conociendo la naturaleza del streaming o flujo de transmisión de datos multimedia podemos encontrar 2 tipos de flujo de datos.

**Streaming en tiempo real o directo:** “El servidor de multimedia asume el control del audio o video, digitalizando un elemento digital, lo comprime y transmite simultáneamente hacia el cliente. El servidor que realiza el flujo suele necesitar de otro equipo independiente **broadcaster** que capture la información y envíe su transmisión”. (Eduardo Polo Ortega, 2015). Adicional a ello usuarios pueden hacer de oyentes o espectadores del flujo con equipos independientes **subscribers**, suscribiéndose a la fuente de contenidos temporales.

**Streaming bajo demanda:** El contenido multimedia se digitaliza, se comprime y se almacena la grabación en el servidor, tal que los clientes

puedan acceder al recurso cuando ellos decidan siendo ellos que controlan la recepción. (Eduardo Polo Ortega, 2015)

Como dato extra dicha información comprimida haciendo uso de sus protocolos se pueden difundir de las siguientes soluciones:

**Unicast:** Los clientes usan su propio streaming con el cual acceden al contenido multimedia con una conexión directa al servidor. Puede generar tráfico en la red y alto consumo de ancho de banda. (Eduardo Polo Ortega, 2015)

**Broadcast:** El servidor envía el flujo a todos los equipos que estén conectados en la red. Cuando se genera información en altas proporciones puede saturar la red. (Eduardo Polo Ortega, 2015)

**Multicast:** Se hace uso de un solo flujo de transmisión para todos los clientes suscritos, reduciendo el tráfico en la emisión de la información. Suele usarse en streaming de tiempo real. (Eduardo Polo Ortega, 2015)

**Relaying Media:** Combina la solución unicast para la transmisión entre servidor multimedia, broadcaster, la red de destino de los clientes, y multicasts para difundir el contenido multimedia a todos los clientes que estén en la red. (Eduardo Polo Ortega, 2015)

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

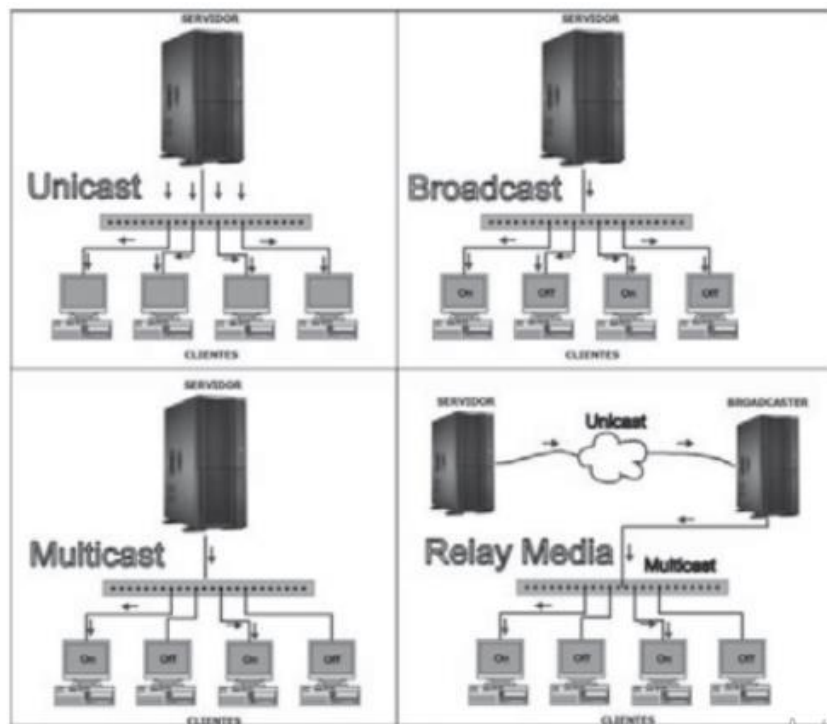


Figura extraída del capítulo 9 sección 3. (Eduardo Polo Ortega, 2015)

### 2.3.3. Antecedentes.

Al tratarse de una transmisión en tiempo real, se ha incorporado protocolos livianos de comunicación que controlan la transmisión, con el fin de evitar demoras al cliente. En la existencia de latencia durante la transmisión del contenido digital los protocolos se encargan de que su existencia sea imperceptible para el cliente. (Eduardo Polo Ortega, 2015)

#### **2.3.4. Protocolo RTMP y RTCP**

**RTMP (Real Time Messaging Protocol):** Protocolo, desarrollado por Adobe Systems, de mensajería streaming de audio, video y datos entre Internet y aplicaciones Flash player y Media Server. (Guniganti, 2013)

**RTCP (Real Time Control Protocol):** Protocolo diseñado para subministrar información que utilizan otros protocolos de capas superiores que moldean la transmisión multimedia según el estado de la red. (Eduardo Polo Ortega, 2015)

#### **2.3.5. Otros protocolos para el uso de Streaming.**

**RTSP (Real Time Streaming Protocol):** Este protocolo se encarga de que tanto cliente y servidor estén en sincronía controlando el flujo de datos de multimedia de ambos. (Eduardo Polo Ortega, 2015)

**RTP (Real Time Transport Protocol):** Protocolo diseñado para el transporte en tiempo real de datos procesados en multimedia y se encarga de la transmisión de la misma. (Eduardo Polo Ortega, 2015)

**UDP (User Datagram Protocol):** Protocolo que puede encargarse de transmisión de contenido multimedia. (Eduardo Polo Ortega, 2015)

## **2.4. Servidor Media**

### **2.4.1. Red5**

“Servidor de media open source para soluciones de streaming en vivo” (Red5, 2020). Desde el 2005 han dedicado a desarrollar soluciones de ingeniería utilizando herramientas de Adobe Flash para la comunicación de streaming (RTMP), y en la actualidad han incluido protocolos de comunicación haciendo uso de WebSockets, HLS y RTSP.

**WebSockets:** Protocolo de comunicación de 2 vías entre el cliente corriendo el código en un ambiente controlado hacia el servidor anfitrión. El objetivo de este protocolo es la comunicación bidireccional con servidores que no se confíe en abrir múltiples conexiones HTTP. (I. Fette, 2011)

**HLS (HTTP Live Streaming):** Desarrollado por Apple, es un protocolo de streaming con el propósito de transmitir flujos enlazados y no enlazados de datos multimedia. (GilJin Yang, 2014)

## 2.5. Herramientas de desarrollo

### 2.5.1. Red5 pro SDK.

**SDK (Software Development Kit):** Es un compendio de librerías y herramientas de software que basados en interfaces de programación ayudan a la creación de nuevas aplicaciones.

El SDK de red 5 consiste en un paquete de librerías en JavaScript que por medio de funciones paramétricas y de configuraciones, permiten el flujo de datos con la comunicación bidireccional entre aplicación del cliente y el servidor multimedia de RED 5.

En las configuraciones al especificar los parámetros se detalla el nombre el protocolo, el host anfitrión, el puerto, el nombre de la app, el nombre de la sesión.

```
var rtcPublisher = new red5prosdk.RTCPublisher();
var rtcSubscriber = new red5prosdk.RTCSubscriber();
var config = {
  protocol: 'ws',
  host: 'localhost',
  port: 5080,
  app: 'live',
  streamName: 'mystream',
  rtcConfiguration: {
    iceServers: [{urls: 'stun:stun2.l.google.com:19302'}],
    iceCandidatePoolSize: 2,
    bundlePolicy: 'max-bundle'
  } // See https://developer.mozilla.org/en-US/docs/Web/AF
};
```

Figura extraída de la instalación del SDK HTML5 (Red5, 2020)

### **2.5.2. JavaScript.**

Es un lenguaje de desarrollo de aplicaciones entre cliente/servidor a través de Internet. Se ejecuta dentro del documento HTML con tag de scripts haciendo las páginas HTML más dinámicas. (Maza, 2012) Se dice que una aplicación es dinámica porque lenguaje modifica el DOM (Document Object Model, estructura del documento de HTML) que a su vez modifica los atributos de los componentes de HTML.

### **2.5.3. PHP**

PHP ó Hypertext Preprocessor es un lenguaje de código abierto generalmente para el desarrollo web. Su código se ejecuta del lado del servidor de forma interpretada. (The PHP Group, 2001-2020 )

### **2.5.4. Laravel.**

Laravel es un framework de PHP que utiliza **Artisan** para maquetar y generar código web. (Taylor Otwell, 2011-2020). El framework nos dota de un patrón de diseño que es Modelo Vista Controlador, con el cual integramos nuestro proyecto web.

**Artisan:** Es una interface de línea de comandos que nos asiste en la creación de nuestra aplicación. (Taylor Otwell, 2011-2020).

**Modelo:** Es una capa del negocio en el cual se alojarán todos los archivos de PHP que tienen la estructura de la base de datos transformados a clases para tratarlos como objetos.

**Vista:** Capa de negocio donde manejamos todo el contenido gráfico e interfaces de usuario GUI.

**Controlador:** Capa de negocio donde se gestiona el procesamiento de los datos obtenidos de la vista con el objeto del modelo. Generalmente en esta capa interactúa con **Eloquent** (ORM de laravel) los procesos de guardado, edición, selección y borrado que llaman a la base de datos.

**Eloquent:** Es el ORM de laravel que simplifica la codificación de las consultas y llamadas a la base de datos. Utiliza los objetos de la capa del Modelo para realizar las relaciones o llamadas de la base de datos. (Taylor Otwell, 2011-2020)

**ORM (Object Relational Mapping):** Es una técnica de programación de conversión de datos entre las bases de datos relacional a los lenguajes de programación orientado a objetos. (C. Xia, 2009)

### 2.5.5. MySQL

Sistema de manejo y administración de base de datos Open Source, desarrollado, distribuido y con soporte de Oracle Corporation. (Oracle Corporation and/or its affiliate, 2020)

### 2.5.6. Angular

Angular es un framework de diseño para el desarrollo de plataformas eficientes y sofisticadas en single-page-app (angular.io, 2020), es decir que se empaqueta toda la aplicación en una sola página web. Basado en el lenguaje de programación **TypeScript** su estructura basada en rutas y componentes, permite la optimización y acoplamiento en soluciones empresariales complejas.

**TypeScript:** Lenguaje de programación de código abierto que simplifica el manejo de JavaScript en un lenguaje estructurado.

### 2.5.7. Guías de diseño.

Para el diseño de la interfaz gráfica y experiencia del usuario se ha seleccionado Bootstrap 3.5

**Bootstrap:** Es un conjunto de librerías en javascript y estilos (CSS) Open Source, escalados a plantillas de diseño establecidos para una presentación responsiva. (Bootstrap Community, 2020)

Para el diseño de este proyecto se ha seleccionado los patrones de diseño de software que vienen dado por los frameworks de desarrollo.

Del lado del cliente Angular como del servidor Laravel nos provee del patrón MVC.

## **Capítulo 3: Metodología de Desarrollo**

### **3. Método y Metodología ágil para el desarrollo.**

#### **3.1. Modelo Evolutivo-Espiral.**

Dentro del desarrollo de software existen modelos (representaciones abstractas del proceso de desarrollo) en donde se conceptualizan las etapas del estado del software. Para el presente trabajo de disertación el modelo de desarrollo evolutivo nos facilitará en implementación y documentación durante el proceso de desarrollo de software.

Según Barry W. Bohem “el modelo en espiral nace de la experiencia y el refinamiento del modelo en cascada” (Barry Boehm, 1988) en donde cada actividad se conecta con la otra y de ser el caso se realiza otro recorrido por todas las fases hasta lograr el software desea por el cliente. En este modelo se puede visualizar el costo del desarrollo en el radio del modelo.



### **3.1.1. Comunicación con el cliente**

En esta etapa se abstrae las necesidades del cliente-usuario final del producto de software, partiendo de los antecedentes que posee. Cuando es segunda etapa en adelante, se evalúa nuevas necesidades del cliente.

En el caso del proyecto se realizó un primer acercamiento con el Director Encargado de la Estación Científica Yasuní, para aclarar las necesidades reales existentes que presentan los investigadores y estudiantes frente a una comunicación con ancho de banda limitado.

### **3.1.2. Planificación**

En esta fase se ordena y se procesa las necesidades en requerimientos formales del usuario a ser automatizados durante un periodo planificado.

Para el caso práctico se definió los requerimientos funcionales de la aplicación de acuerdo al orden impacto:

- Gestión de Usuarios
- Creación de Conferencias
- Publicación de Conferencias
- Envío y Recepción de mensajería para chat virtual
- Suscripción de Conferencias

Posterior a eso se seleccionó las herramientas de desarrollo para backend y frontend. De esto obtenemos los requerimientos no funcionales de la aplicación.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

|                               |                        |
|-------------------------------|------------------------|
| <b>Backend</b>                | PHP 7.3 -> Laravel 5.2 |
| <b>Servidor Servicios</b>     | Apache 2.4             |
| <b>Motor de Base de Datos</b> | MySQL 8.0              |
| <b>Servidor Multimedia</b>    | Red 5 pro              |
| <b>Frontend</b>               | Angular CLI 5          |

### **3.1.3. Análisis de riesgo**

En esta etapa se evalúa los posibles riesgos antes del desarrollo de los módulos como, por ejemplo, el nivel de conocimientos del lenguaje de programación que satisfagan la construcción del módulo, la infraestructura si puede soportar la aplicación montada y la comunicación entre servidores de aplicación como de servicios.

### **3.1.4. Ingeniería**

La fase de ingeniería es donde se centra en el proceso de desarrollo y construcción de los módulos en el framework previamente mencionado hasta completar un entregable.

### **3.1.5. Evaluación del cliente**

Se realiza pruebas de funcionamiento, y se muestra un entregable al cliente. En esta etapa replantea nuevas necesidades que pueden incorporarse al proyecto. Al ser aceptados se acuerda una nueva iteración en el ciclo de vida.

### **3.1.6. Construcción y entrega**

Se empaqueta la solución y se monta en un servidor de aplicaciones.

## **3.2. Scrum.**

Otro recurso, para el desarrollo de la aplicación, existe el marco de trabajo de desarrollo como es el caso de SCRUM

El proceso Scrum fue creado por Jeff Sutherland en 1993 adoptando el nombre scrum de una analogía donde se comparaba el alto rendimiento, interfuncionalidad en equipo hacia la formación scrum utilizada en los equipos de Rugby publicada en 1986 en "Harvard Business Review" por Takeuchi y Nonaka

"Scrum provee un marco de trabajo ágil para el cumplimiento de proyectos complejos. Fue formalizado para el desarrollo de proyectos de software. Trabaja bien para cualquier ámbito complejo e innovador" (Scrumalliance, 2017).

### **3.2.1. Planeación o Scrum Planning.**

En planificación o Sprint Planning se compila todas las actividades generadoras de valor propuestas por un cliente y se las ordena según el orden de prioridad. Se establece las tareas con una estimación temporal de cada iteración.

### **3.2.2. Sprint**

Son segmentos de tiempo cortos en los que promete un resultado completo para una nueva iteración. También es conocido como una iteración como se menciona anteriormente en el modelo de espiral.

### **3.2.3. Diario Scrum**

Son reuniones al inicio del Sprint, que expone información los miembros del equipo y se analiza los impedimentos o facilidades para el cumplimiento del Sprint.

### **3.2.4. Sprint retrospective**

Es un control sobre el Sprint en el cuál se analiza de manera continua el cumplimiento de los objetivos del Sprint.

### **3.2.5. Roles del equipo**

**3.2.5.1. Scrum Master:** Persona que administra las tareas y organiza al equipo.

**3.2.5.2. Product Owner:** Son los interesados en el producto o servicio desarrollado, estos indican que se debe o no hacer en el desarrollo.

**3.2.5.3. Scrum Team:** Equipo de trabajo en cumplir las tareas. El equipo no debe ser mayor a 10 personas.

**3.2.5.4. Usuarios:** Son los que darán uso al producto o servicio.

### **3.2.6. Product backlog.**

Lista de objetivos o requisitos priorizadas en la visión y expectativas del cliente.

### **3.2.7. Scrum Taskboard.**

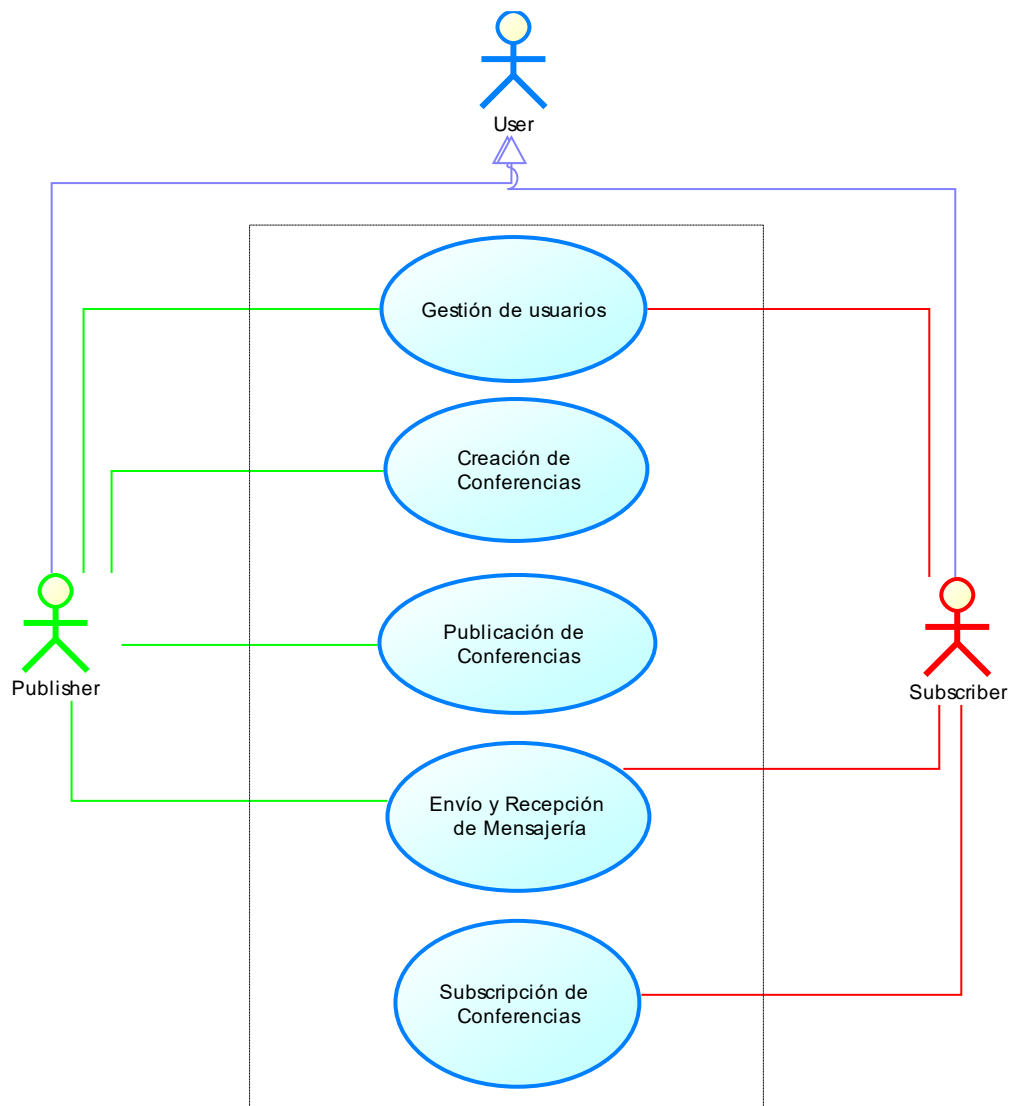
Tabla de tareas con los objetivos del sprint que nos muestra gráficamente el cómo va realizándose en el transcurso del tiempo. Se suele utilizar diagramas Kanban que muestran las actividades con estados de: pendiente, iniciado, en progreso y hechos

### **3.2.8. Burn down chart.**

Es un gráfico de trabajo pendiente que muestra la velocidad con la que se completa los objetivos.

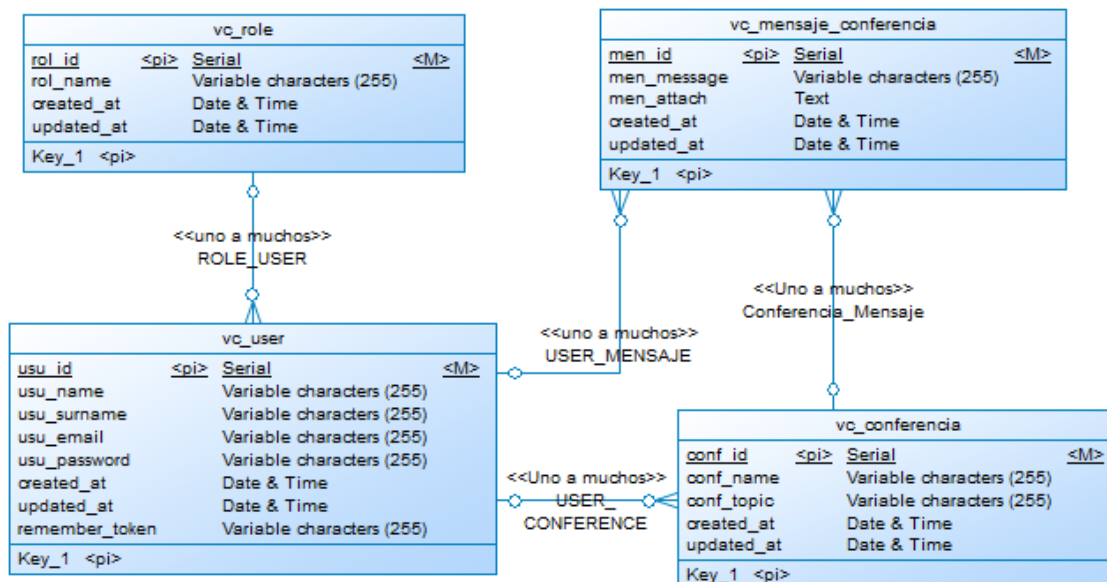
### 3.3. Diseño

#### 3.3.1. Diseño Casos de Uso

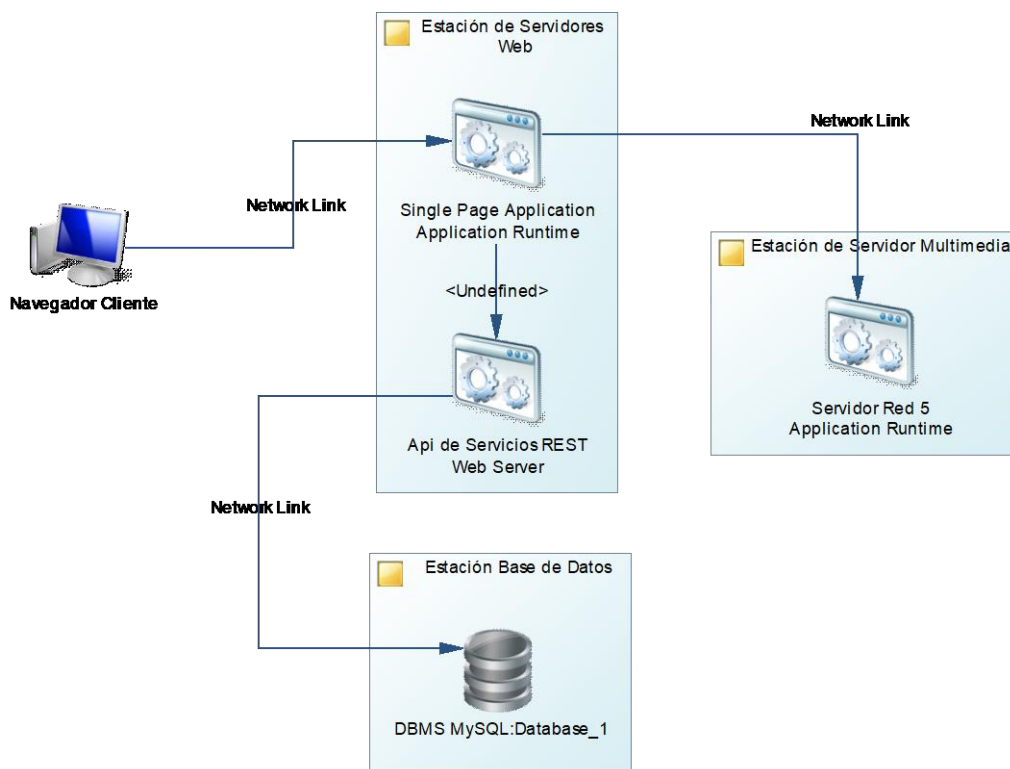


Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
 Caso de estudio: Estación Científica Yasuní De La PUCE

### 3.3.2. Diseño conceptual.

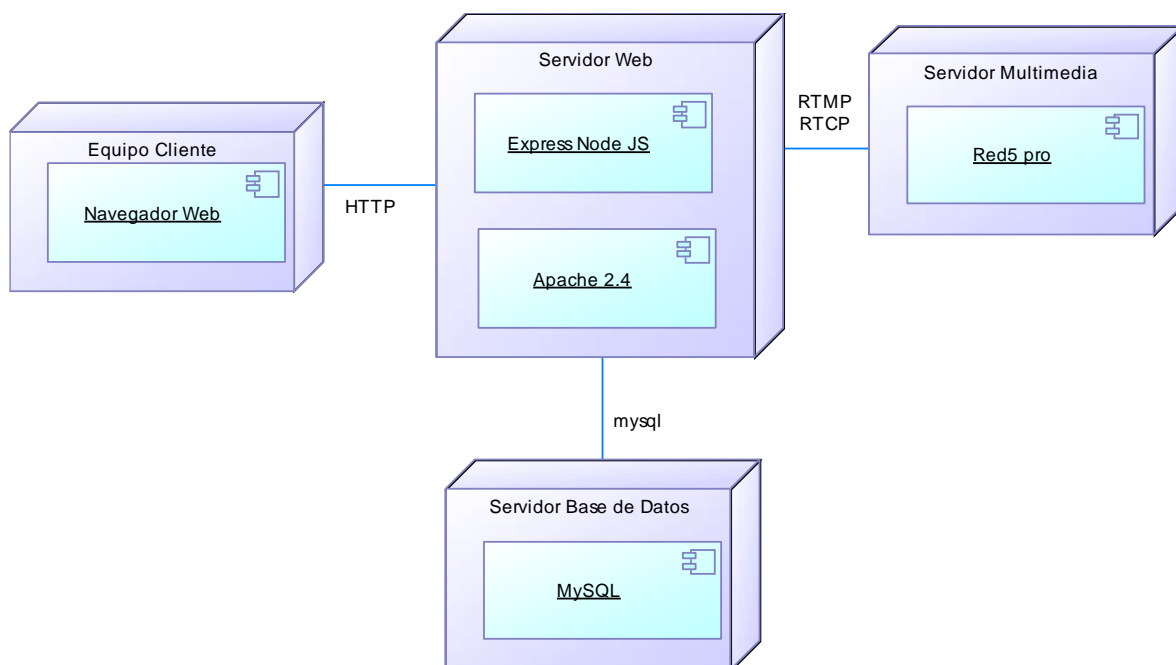


### 3.3.3. Diseño de arquitectura infraestructura

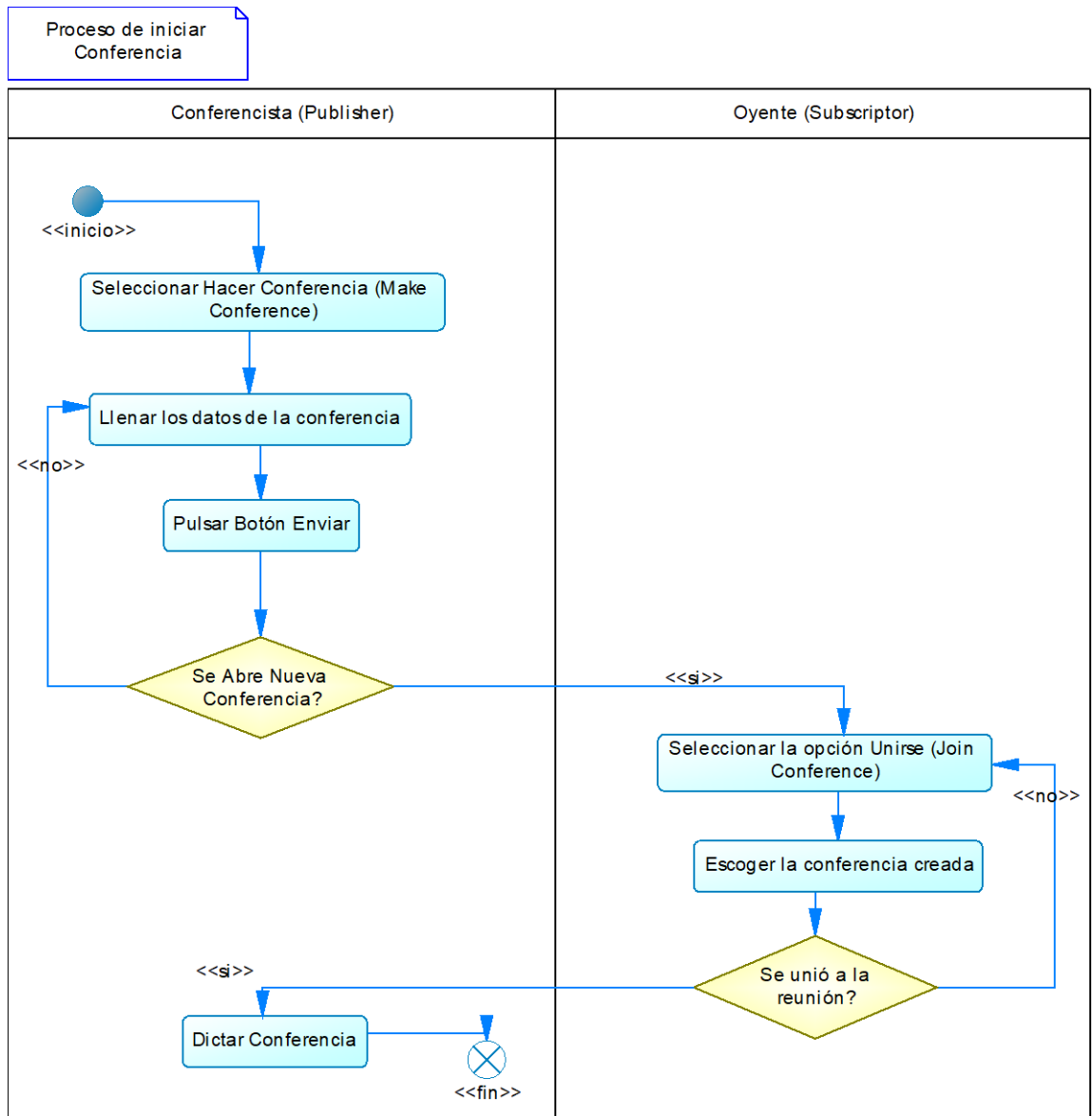


Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

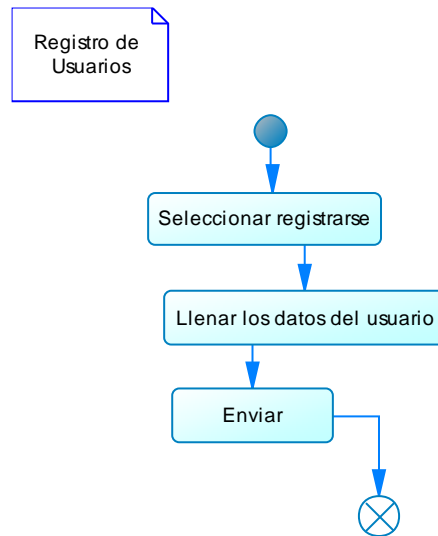
### 3.3.4. Diseño de implementación



### 3.3.5. Diagrama de Actividades.



Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

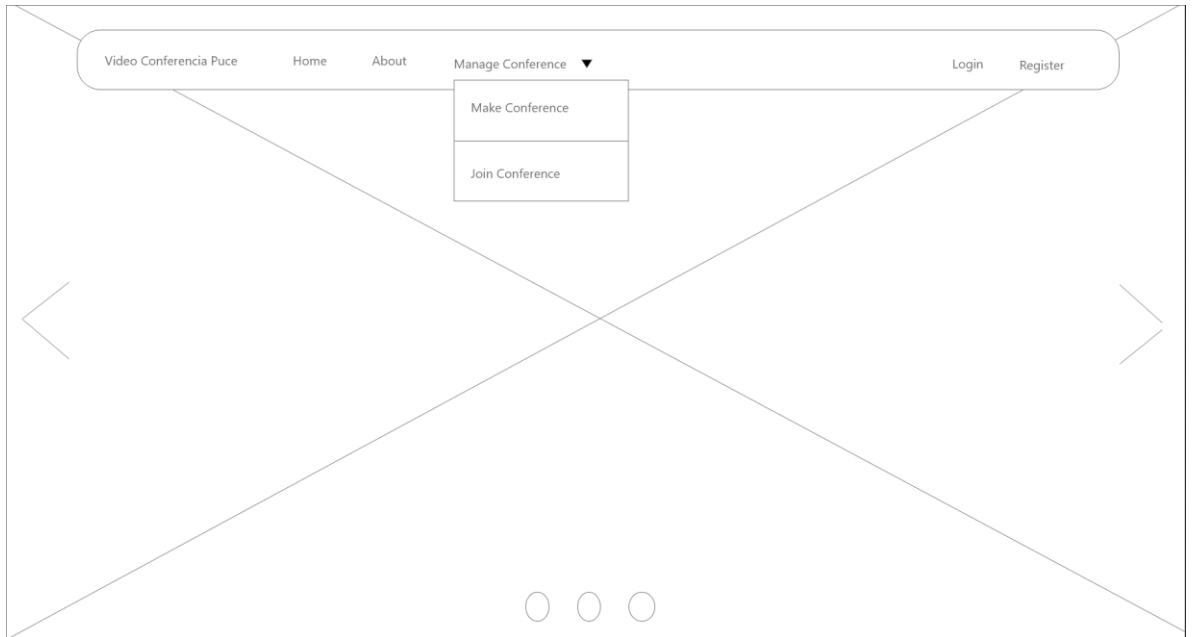


### 3.3.6. Prototipo.

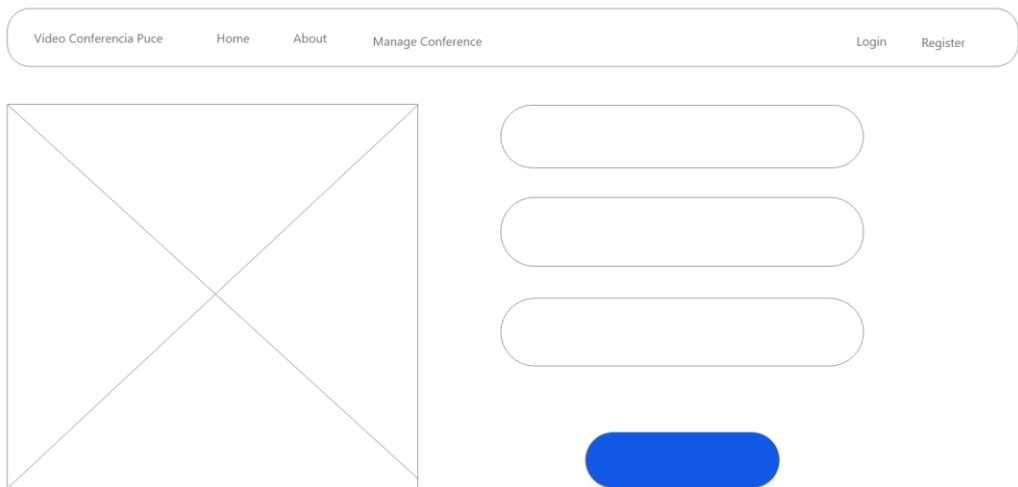
En el diseño de prototipo se hizo una maqueta con **Adobe Xd** (Experience Design), herramienta con el cual podemos realizar interfaces gráficas de usuario para soluciones informáticas previos a su desarrollo. En este modelo se diseñó el Frontend del cliente.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

## Home



## Register User - Make Conference



## Join Conference

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

Video Conferencia Puce   Home   About   Manage Conference   Login   Register


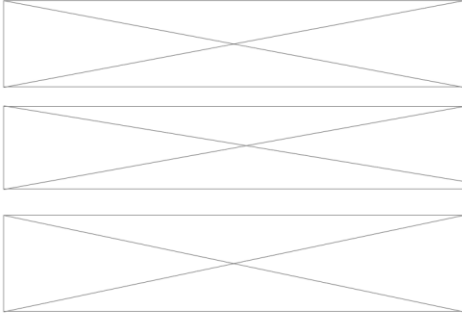
| # | CONFERENCE NAME | TOPIC NAME | ACTIONS                             |
|---|-----------------|------------|-------------------------------------|
|   |                 |            | <input type="button" value="JOIN"/> |
|   |                 |            | <input type="button" value="JOIN"/> |
|   |                 |            | <input type="button" value="JOIN"/> |
|   |                 |            | <input type="button" value="JOIN"/> |

## Conference

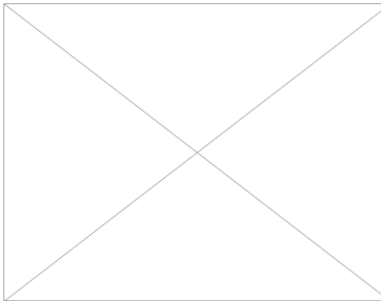
Video Conferencia Puce   Home   About   Manage Conference   Login   Register

GOOGLE SEARCH

SEARCH...



CHATROOM



Typing...

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

## Capítulo 4: Implementación

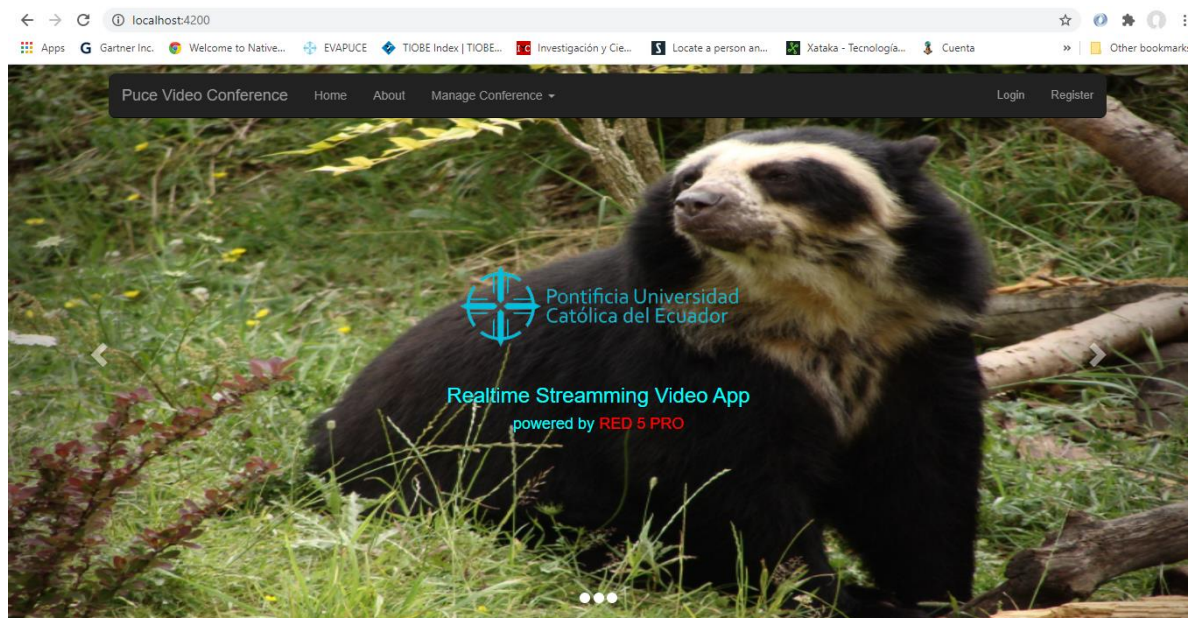
### 4. Implementación de la aplicación.

#### 4.1. Implementación de interface.

Para la implementación de la interface Web se utilizó una hoja de estilos CSS personalizada y con la biblioteca de estilos que nos presenta Bootstrap en su versión 3. De tal forma que el diseño es responsivo.

Con el Node Js y el framework Angular CLI en su versión 5 realizamos una Single Page Application para el cliente (SPA) con programación de componentes en TypeScript, de tal forma que la experiencia de usuario fluya sin demora de carga.

### Home

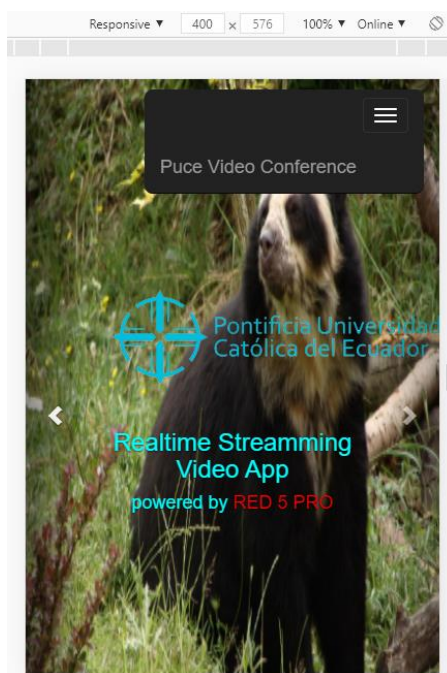


El componente de inicio o Home se carga la navegación y los menús que nos re direccionaran al componente requerido.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

Está conformado por un componente app que carga la navegación y un componente default que carga las imágenes con el contenido en el carrusel.

## Home – Responsive



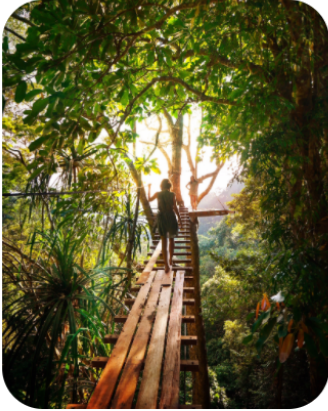
Con Bootstrap manejando su hoja de estilos podemos hacer que la interface web sea adaptativa a cualquier tamaño de cualquier dispositivo.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

## Register User

Puce Video Conference Home About Manage Conference ▾ Login Register

New User



Name  
  
Name is required

Surname  
  
Surname is required

Email  
  
Email is required

Password  
  
Password is required

Register

Al finalizar los parámetros se registra en base de datos un usuario.

**Nota:** Todas las contraseñas y tokens están cifrados con un método de cifrado (SHA#), como manejo de seguridad.

### 4.2. Implementación de manejo de sesiones de usuario.

En el manejo de sesiones de usuario, es requerido que exista el usuario registrado en el sistema, y se podrá hacerse el inicio de sesión.

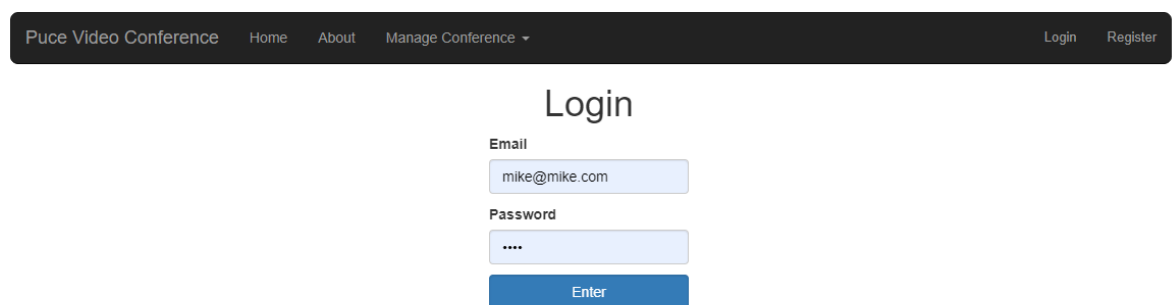
Las sesiones y las peticiones al api de servicios en Laravel apartir de esta etapa se los realiza a través de un cliente HTTP y con Tokens de Autorizacion.

Los tokens son gestionados con **JWT**.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

**JWT:** Es un estándar abierto que se compacta y se auto contiene para transmitir información de forma segura los objetos JSON. (Auth0® Inc, 2013 - 2020).

## Login



The image shows a screenshot of a web application's login page. At the top, there is a dark navigation bar with the text "Puce Video Conference" on the left and "Login Register" on the right. Below the navigation bar, the word "Login" is centered in a large font. Underneath, there are two input fields: "Email" with the value "mike@mike.com" and "Password" with four dots. A blue "Enter" button is positioned below the password field.

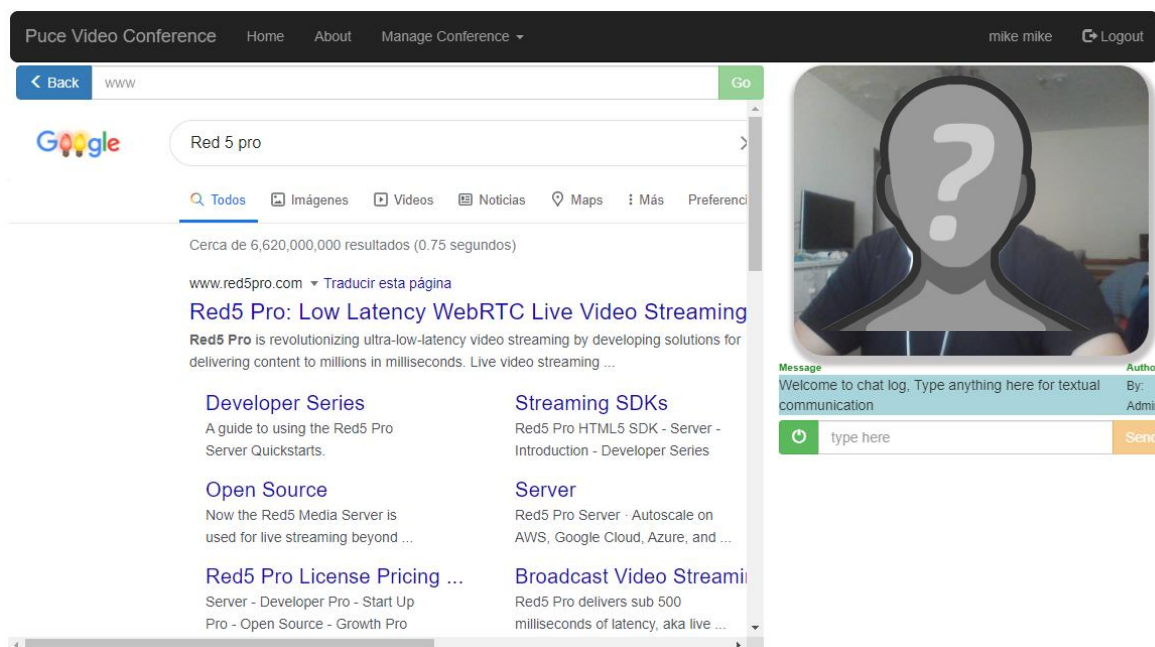
Al iniciar a la autenticación del usuario, se genera un token de autorización que irán en la cabecera de todas las peticiones Rest al api de servicios Laravel.

Se comprueba que el token no esté expirado para proceder con el resto de peticiones, caso contrario, todas las peticiones a los servicios serán rechazados.

Para persistir las variables de sesión como tokens o usuarios en el navegador se utilizó el Local Storage del mismo navegador. Es decir, en una base de datos temporal en el navegador del cliente.



Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE



Esta es una sala de reuniones que está dividido por 3 secciones.






- La primera sección situado a la izquierda es la navegación haciendo uso del motor de búsqueda de google.
- La segunda sección es el video streaming que transmitirá audio y video.
- La tercera sección es el chat que estará escuchando los mensajes emitidos por los integrantes de la reunión.

Cómo dato adicional si se desea retomar una reunión pasada, se puede ir al perfil del usuario haciendo clic en el nombre y lo re direccionará a todas las reuniones creadas por el conferencista.



Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE


## Profile | Conferences made


| # | Conference Name         | Topic               | Actions   |
|---|-------------------------|---------------------|---|
| 1 | VideoconferencewithRed5 | Red 5 pro           |  |
| 2 | Pandemonic              | Pandemonic          |  |
| 3 | GlobalWarming           | Global Warming 2020 |  |
| 4 | Social Media            | Community Manager   |  |
| 5 | Software Enhancement    | Red 5 pro           |  |

## Subscripción de Conferencias

Puce Video Conference Home About Manage Conference ▾ jeff jeff Logout

### Join Conference



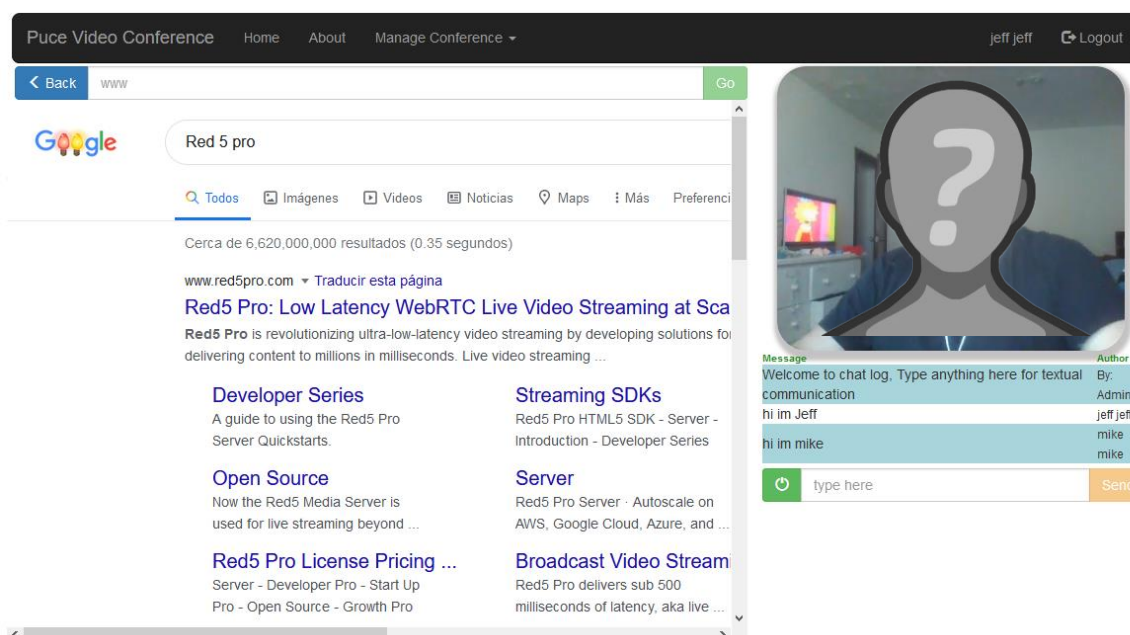
| # | Conference Name         | Topic     | Active Members | Actions   |
|---|-------------------------|-----------|----------------|---|
| 1 | VideoconferencewithRed5 | Red 5 pro | 0              |  |

Al seleccionar unirse a una conferencia el cliente de Angular hace las siguientes peticiones por HTTP:

- Primero se hace una petición al servidor multimedia para conocer los id(s), nombres y número de participantes activos, en las transmisiones que se están ejecutando en vivo.
- Con el nombre de las transmisiones se realiza una petición HTTP al api de servicios en Laravel para recuperar los datos de la reunión como por ejemplo el tema de la reunión.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

En este componente nos desplegará todas las videoconferencias que están siendo transmitidos en Vivo. Al hacer clic nos unimos al video conferencia como subscriptores.



### 4.3. Implementación de Streaming usando el protocolo RTMP, RTCP y WS con Red5.

Primero se necesitará crearse una cuenta en la página de <https://www.red5pro.com/>. Seguido a eso se necesitará descargarse el servidor multimedia y el SDK en JS para las peticiones por Javascript.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.

Caso de estudio: Estación Científica Yasuní De La PUCE

|                          |                  |                       |       |
|--------------------------|------------------|-----------------------|-------|
| conf                     | 29/11/2020 5:16  | Carpeta de archivos   |       |
| lib                      | 29/11/2020 5:16  | Carpeta de archivos   |       |
| log                      | 07/12/2020 16:32 | Carpeta de archivos   |       |
| plugins                  | 29/11/2020 5:16  | Carpeta de archivos   |       |
| webapps                  | 29/11/2020 5:16  | Carpeta de archivos   |       |
| work                     | 29/11/2020 5:16  | Carpeta de archivos   |       |
| build_dependencies.txt   | 25/08/2020 14:48 | Documento de tex...   | 2 KB  |
| commons-daemon-1.1.0.jar | 29/05/2019 18:21 | Executable Jar File   | 25 KB |
| install-service.bat      | 19/05/2016 0:10  | Archivo por lotes ... | 4 KB  |
| LICENSE.KEY              | 07/12/2020 17:11 | Entradas de registro  | 1 KB  |
| license.txt              | 25/08/2020 14:43 | Documento de tex...   | 37 KB |
| red5                     | 25/08/2020 14:43 | Archivo               | 4 KB  |
| red5.bat                 | 25/08/2020 14:43 | Archivo por lotes ... | 2 KB  |
| red5.service             | 19/05/2016 16:39 | Archivo SERVICE       | 1 KB  |
| red5.sh                  | 25/08/2020 14:43 | Shell Script          | 3 KB  |
| red5-debug.bat           | 11/01/2016 14:22 | Archivo por lotes ... | 1 KB  |
| red5-debug.sh            | 19/05/2016 21:33 | Shell Script          | 1 KB  |

Una vez descargado se procede a iniciar red5.bat encender el servidor multimedia.

**Nota:** En el directorio conf se puede configurar los niveles de seguridad del servidor como por ejemplo: usuario, contraseña o peticiones por Token como en JWT.

```
configuration: {
  protocol: 'ws',
  port: 5080,
  host: 'localhost',
  app: 'live',
  streamName: '',
  rtcConfiguration: {
    iceServers: [{ urls: 'stun:stun2.1.google.com:19302' }],
    iceCandidatePoolSize: 2,
    bundlePolicy: 'max-bundle'
  },
  streamMode: 'live',
  mediaElementId: 'red5pro-publisher',
  bandwidth: {
    audio: 56,
    video: 512
  },
}
```

El **sdk** de red5 nos permite configurar el host, el puerto y el protocolo al que se va a comunicarse y también se puede parametrizar el ancho de banda en

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

Bytes. Para este caso hemos utilizado valores mínimos para la transmisión cuando la velocidad y latencia de red es constante.

```
mediaConstraints: {
  audio: true,
  video: {
    width: {
      exact: 480
    },
    height: {
      exact: 480
    },
    frameRate: {
      min: 8,
      max: 24
    }
  }
}
```

Como últimos parámetros se puede configurar las dimensiones de video y decir con cuantos cuadros por segundo será la transmisión.

```
[INFO] [QuePush@audio:564232249] com.red5pro.jmfext.renderer.rtmp.RTMPAudioRenderer - Start first audio timestamp: 2143
Codec init
[aac @ 00000000008F73C0] Warning: not compiled with thread support, using thread emulation
codec opened
AAC encoder codec opened
malloc 4096 init
[INFO] [Connection Checker] com.red5pro.server.stream.Red5ProConnManager - Pro connections; Total count: 1, WebRTC ports
allocated: 1, edge-proxy: 0, re-streamers: 0, sm-pulses:0
```

Cuando creamos o iniciamos una videoconferencia el servidor de aplicaciones escucha y abre una conexión para que los miembros se puedan unir.

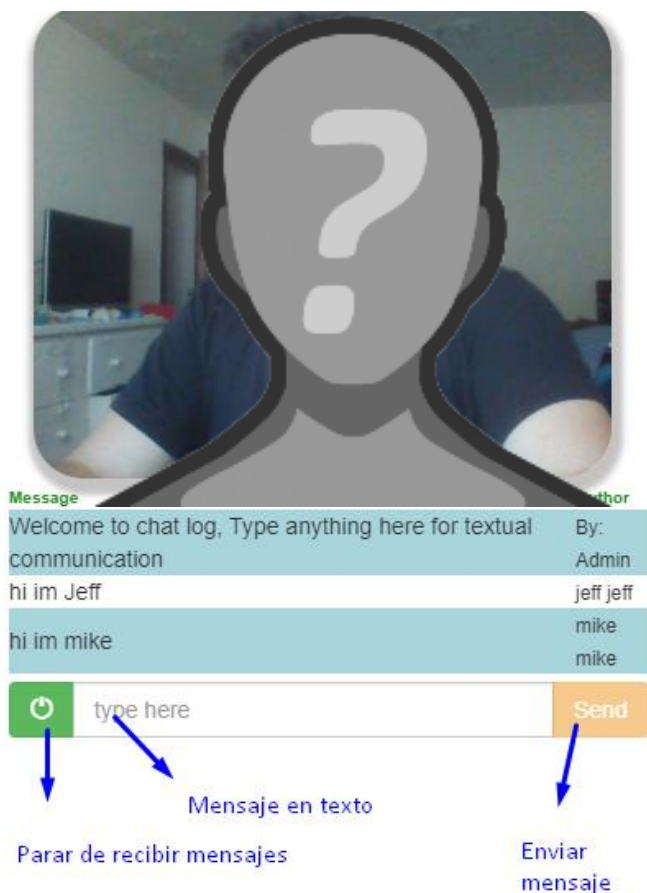
Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

```
C:\WINDOWS\system32\cmd.exe
allocated: 1, edge-proxy: 0, re-streamers: 0, sm-pulses:0
[INFO] [Connection Checker] com.red5pro.server.stream.Red5ProConnManager - Pro connections; Total count: 1, WebRTC ports
allocated: 1, edge-proxy: 0, re-streamers: 0, sm-pulses:0
[INFO] [Connection Checker] com.red5pro.server.stream.Red5ProConnManager - Pro connections; Total count: 1, WebRTC ports
allocated: 1, edge-proxy: 0, re-streamers: 0, sm-pulses:0
[INFO] [http-nio2-0.0.0.0-5080-exec-1] com.red5pro.server.stream.Red5ProConnManager - LicenseManager max connections: 10
limit counter: 2 true
[INFO] [http-nio2-0.0.0.0-5080-exec-1] com.red5pro.server.stream.Red5ProConnManager - Adding session id: TGCEBH1JQOPZX
RTCCConnection
[INFO] [http-nio2-0.0.0.0-5080-exec-1] com.infrared5.red5pro.live.Red5ProLive - W3C x-category:session x-event:connect c
ip:127.0.0.1 c-client-id:2-subscriber-3f8c
[INFO] [http-nio2-0.0.0.0-5080-exec-5] com.infrared5.red5pro.live.Red5ProLive - isPlaybackAllowed /live Videoconferencw
ithRed5
[INFO] [http-nio2-0.0.0.0-5080-exec-5] com.red5pro.server.util.NetworkManager - Public address: 190.152.83.251
[INFO] [http-nio2-0.0.0.0-5080-exec-5] com.red5pro.server.util.NetworkManager - Local address: 192.168.1.23
[WARN] [Terminator: j80t1eov] org.ice4j.ice.HostCandidate - Wrapper from lookup: null current: IceUdpSocketWrapper [tran
sportAddress=192.168.1.23:49154/udp, session=null]
[INFO] [http-nio2-0.0.0.0-5080-exec-10] com.red5pro.io.rtp.RTPConnectorOutputStream - Transformer: com.red5pro.io.rtp.tr
ansform.dtls.DtlsPacketTransformer@f96421c ready? false
[INFO] [BundleStart@VideoconferencewithRed5:subscriber-3f8c] com.red5pro.jmfext.codec.audio.opus.CauldronOpusEncoder - S
ettings - bandwidth: auto bitrate: 128000 DTX: false FEC: false
Opus encoder - channels in: 2 out: 2 sample rate: 48000 bitrate: 128000
Opus encoder opened
Opus encoder codec opened
[INFO] [http-nio2-0.0.0.0-5080-exec-10] com.infrared5.red5pro.live.Red5ProLive - streamSubscriberStart
[INFO] [http-nio2-0.0.0.0-5080-exec-10] com.infrared5.red5pro.live.Red5ProLive - W3C x-category:stream x-event:play c-ip
:127.0.0.1 x-sname:null
[INFO] [Connection Checker] com.red5pro.server.stream.Red5ProConnManager - Pro connections; Total count: 2, WebRTC ports
allocated: 2, edge-proxy: 0, re-streamers: 0, sm-pulses:0
```

Durante la reunión de la sesión, el log del servidor multimedia nos indica cuantos integrantes están activos en transmisión.

#### 4.4. Implementación del manejo de mensajería.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

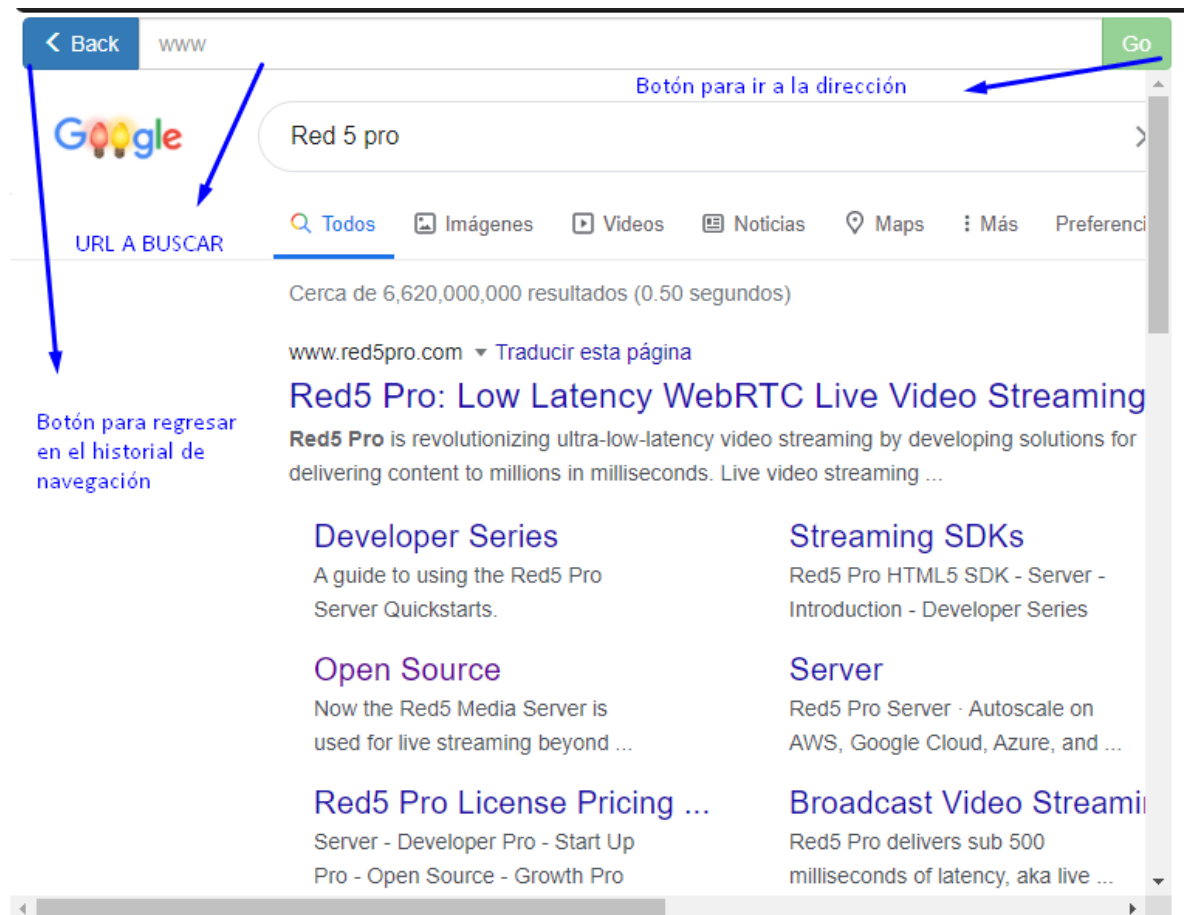


La funcionalidad de mensajería consta de peticiones constantes en un hilo de ejecución donde se está recuperando todos los mensajes de la conferencia activa. La interface consta de un botón que activa o para las peticiones de mensajes, un cuadro donde muestra el log de los mensajes, un cuadro de texto de lo que se desea comunicar y un botón para enviar al servicio Post en el servidor de servicios en Laravel.

#### **4.5. Utilización del servicio de Google para búsquedas de artículos científicos o de carácter técnicos.**

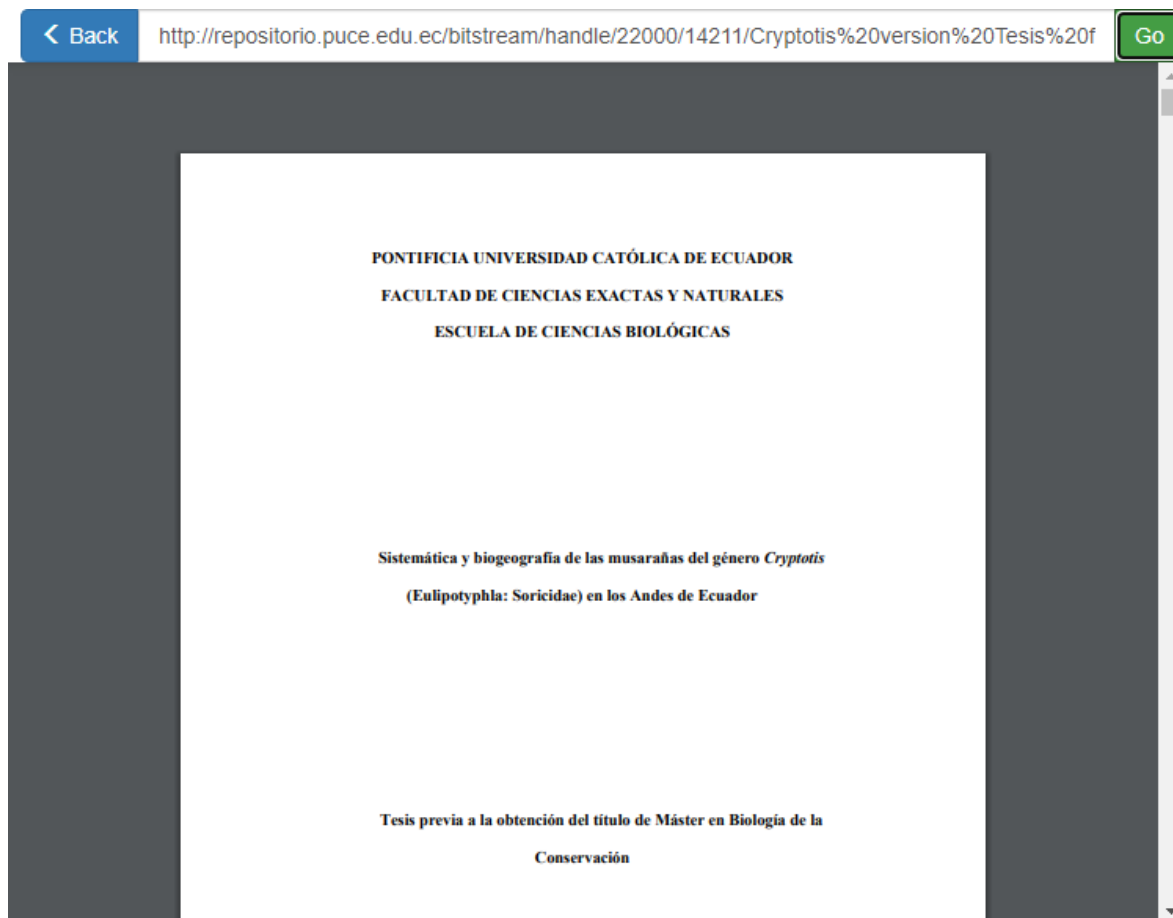
Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

La navegación de google consta de un iframe, un cuadro de texto para las direcciones urls, un botón para regresar por el historial navegado y un botón para ir a la dirección url.



**Nota:** No todas las páginas web externas permiten la navegación por iframe, ya que otras direcciones de página poseen un alto nivel de seguridad.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE



Este es un ejemplo que se puede revisar documentación de artículos o documentos de disertación a través del uso del iframe, mientras se produce la videoconferencia.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

## Capítulo 5: Pruebas

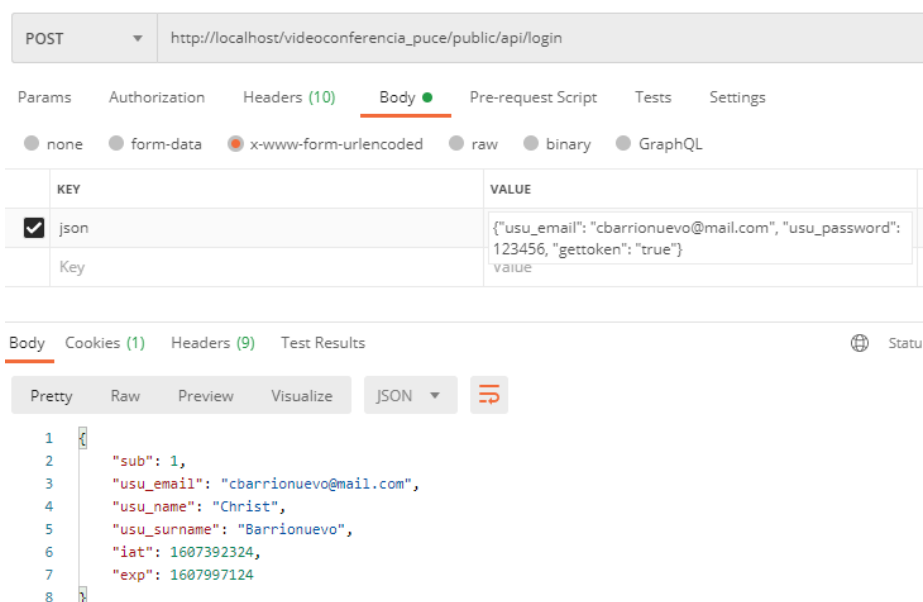
### 5. Pruebas

#### 5.1. Pruebas funcionales.

Para las pruebas funcionales se realizó una malla de pruebas.

#### Prueba de api de servicios

En las pruebas de api se utilizó POSTMAN, que es una herramienta que nos permite mandar peticiones HTTP con cualquier método, por ejemplo, el login de usuario.



The screenshot shows a Postman interface for a POST request to `http://localhost/videoconferencia_puce/public/api/login`. The request body is set to JSON with the following content:

| KEY  | VALUE   |
|------|---|
| json | <pre>{ "usu_email": "cbarrionuevo@mail.com", "usu_password": "123456", "gettoken": "true" }</pre> |
| Key  | value   |

The response body is displayed in JSON format:

```
1 {
2   "sub": 1,
3   "usu_email": "cbarrionuevo@mail.com",
4   "usu_name": "Christ",
5   "usu_surname": "Barrionuevo",
6   "iat": 1607392324,
7   "exp": 1607997124
8 }
```

Para comprobar que todos los servicios están funcionando correctamente se realizó una malla de pruebas. Véase en el [Anexo 6.7.1](#)

## Prueba de aplicación cliente web

Se realizó la carga de cada componente de Angular con sus rutas ya establecidas. Y se hizo pruebas de integración entre las peticiones hacia Laravel y hacia el servidor multimedia.

### 5.2. Pruebas no funcionales.

En las pruebas no funcionales monitorearemos el rendimiento de los servidores de la aplicación ya en el desenvolvimiento con los requerimientos del usuario. Para esto se utilizó una herramienta de Apache llamada Jmeter.

**Jmeter:** Es una aplicación de código abierto, escrito en JAVA para realizar las pruebas de comportamientos de las aplicaciones.

#### 5.2.1. Pruebas de carga.

Para esta prueba trataremos de simular la concurrencia de usuarios en la aplicación web para ver cómo responde y el rendimiento de la app.

- Pruebas de carga en el api Login

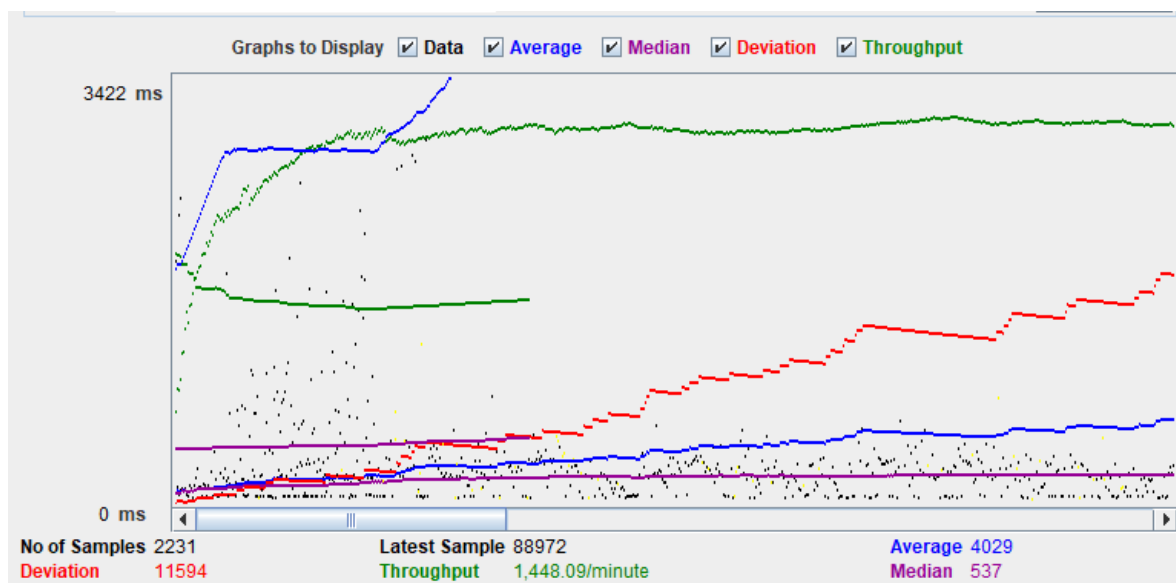
En el grupo de usuarios se tomo a 100 usuarios concurrentes en un periodo de 5 segundos haciendo peticiones por POST para realizar el login de usuarios y generando JWT TOKENS.



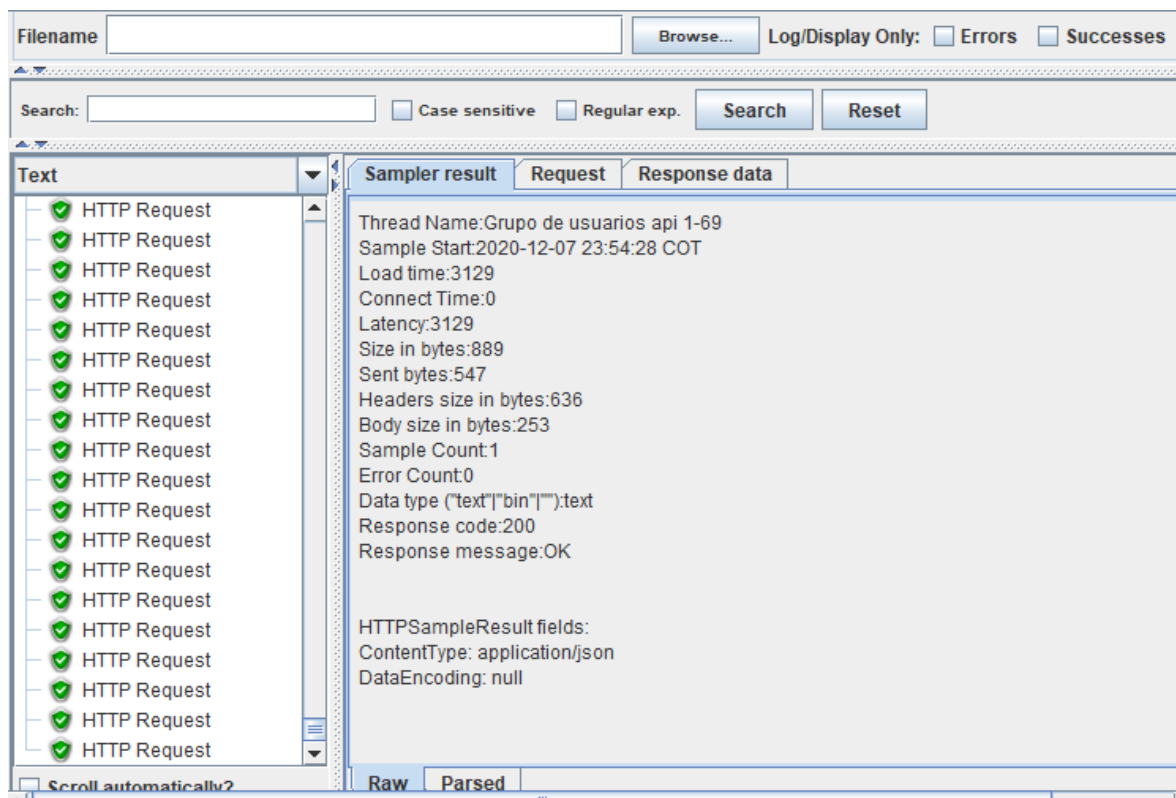
The image shows a screenshot of the 'Thread Properties' dialog box in JMeter. It contains three input fields: 'Number of Threads (users):' with the value '100', 'Ramp-up period (seconds):' with the value '5', and 'Loop Count:' with a checked 'Infinite' checkbox and an empty text field.

Y los resultados fueron los siguientes:

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

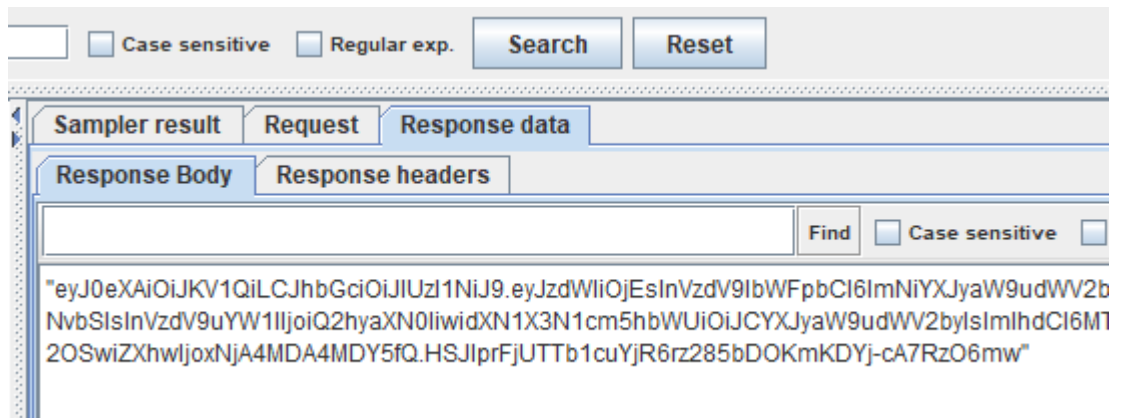


Se tomó 2231 muestras tenemos que en nuestro rendimiento o throughput, podemos realizar alrededor 1448 peticiones por minuto. La desviación nos indica el tiempo en el que varía la respuesta de la petición que en nuestro caso es de 11 segundos.



Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

Todas las muestras que se realizaron tuvieron de status 200 cada uno con la respuesta del token al momento de hacer login



De las 2231 muestras que se contabilizaron 151 peticiones con falla y 2080 peticiones correctas.

Haciendo un cálculo de 3 tenemos los siguiente:

Número de peticiones buenas => b

Número de peticiones totales => t

$$\frac{b \times 100}{t}$$

Con el cuál se concluye que nuestra aplicación de servicios tiene una disponibilidad del **93%**. Con una latencia promedio de 4307 milisegundos.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
 Caso de estudio: Estación Científica Yasuní De La PUCE

**View Results in Table**

Name:

Comments:

Write results to file / Read from file

Filename   Log/Display Only:  Errors  Successes

| Sample # | Start Time   | Thread Name               | Label        | Sample Time(...) | Status ^ | Bytes | Sent Bytes | Latency |
|----------|--------------|---------------------------|--------------|------------------|----------|-------|------------|---------|
| 1        | 23:53:34.827 | Grupo de usuarios api 1-1 | HTTP Request | 155              | ✓        | 890   | 544        | 155     |
| 2        | 23:53:34.888 | Grupo de usuarios api 1-2 | HTTP Request | 94               | ✓        | 892   | 535        | 94      |
| 3        | 23:53:34.938 | Grupo de usuarios api 1-3 | HTTP Request | 94               | ✓        | 890   | 541        | 94      |
| 4        | 23:53:34.982 | Grupo de usuarios api 1-2 | HTTP Request | 70               | ✓        | 891   | 541        | 70      |
| 5        | 23:53:34.986 | Grupo de usuarios api 1-4 | HTTP Request | 82               | ✓        | 890   | 550        | 82      |
| 6        | 23:53:34.982 | Grupo de usuarios api 1-1 | HTTP Request | 101              | ✓        | 889   | 547        | 101     |
| 8        | 23:53:35.068 | Grupo de usuarios api 1-4 | HTTP Request | 77               | ✓        | 891   | 529        | 77      |
| 9        | 23:53:35.052 | Grupo de usuarios api 1-2 | HTTP Request | 125              | ✓        | 887   | 556        | 125     |
| 10       | 23:53:35.083 | Grupo de usuarios api 1-1 | HTTP Request | 108              | ✓        | 887   | 544        | 108     |
| 11       | 23:53:35.036 | Grupo de usuarios api 1-5 | HTTP Request | 179              | ✓        | 888   | 559        | 179     |
| 12       | 23:53:35.177 | Grupo de usuarios api 1-2 | HTTP Request | 66               | ✓        | 889   | 544        | 66      |
| 13       | 23:53:35.086 | Grupo de usuarios api 1-6 | HTTP Request | 169              | ✓        | 890   | 547        | 169     |
| 14       | 23:53:35.112 | Grupo de usuarios api 1-3 | HTTP Request | 144              | ✓        | 888   | 529        | 144     |

**View Results in Table**

Name:

Comments:

Write results to file / Read from file

Filename   Log/Display Only:  Errors  Successes

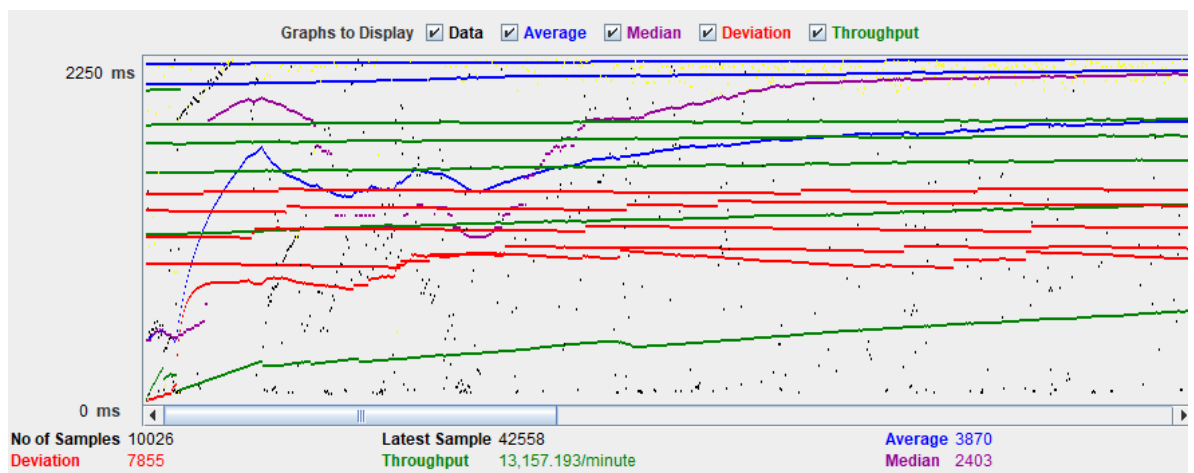
| Sample # | Start Time   | Thread Name                | Label        | Sample Time(...) | Status v | Bytes | Sent Bytes | Latency | Connect Ti... |
|----------|--------------|----------------------------|--------------|------------------|----------|-------|------------|---------|---------------|
| 7        | 23:53:35.032 | Grupo de usuarios api 1-3  | HTTP Request | 80               | ✗        | 9247  | 544        | 75      | 0             |
| 43       | 23:53:35.584 | Grupo de usuarios api 1-6  | HTTP Request | 381              | ✗        | 9247  | 553        | 289     | 0             |
| 53       | 23:53:35.915 | Grupo de usuarios api 1-2  | HTTP Request | 229              | ✗        | 9247  | 550        | 165     | 0             |
| 57       | 23:53:36.089 | Grupo de usuarios api 1-5  | HTTP Request | 139              | ✗        | 9247  | 532        | 138     | 0             |
| 61       | 23:53:36.129 | Grupo de usuarios api 1-9  | HTTP Request | 168              | ✗        | 31331 | 556        | 164     | 0             |
| 63       | 23:53:36.026 | Grupo de usuarios api 1-3  | HTTP Request | 313              | ✗        | 31329 | 538        | 309     | 0             |
| 81       | 23:53:36.423 | Grupo de usuarios api 1-8  | HTTP Request | 348              | ✗        | 9247  | 535        | 253     | 0             |
| 87       | 23:53:36.737 | Grupo de usuarios api 1-12 | HTTP Request | 185              | ✗        | 9247  | 544        | 183     | 0             |
| 135      | 23:53:37.615 | Grupo de usuarios api 1-16 | HTTP Request | 364              | ✗        | 9247  | 535        | 230     | 0             |
| 136      | 23:53:37.918 | Grupo de usuarios api 1-14 | HTTP Request | 88               | ✗        | 9247  | 547        | 87      | 0             |
| 141      | 23:53:37.916 | Grupo de usuarios api 1-11 | HTTP Request | 162              | ✗        | 9247  | 535        | 161     | 0             |
| 151      | 23:53:35.792 | Grupo de usuarios api 1-20 | HTTP Request | 2547             | ✗        | 9247  | 550        | 2416    | 0             |
| 165      | 23:53:38.450 | Grupo de usuarios api 1-15 | HTTP Request | 384              | ✗        | 9247  | 547        | 309     | 0             |
| 202      | 23:53:39.532 | Grupo de usuarios api 1-18 | HTTP Request | 203              | ✗        | 9247  | 547        | 202     | 0             |
| 208      | 23:53:39.758 | Grupo de usuarios api 1-10 | HTTP Request | 69               | ✗        | 5122  | 532        | 68      | 0             |
| 231      | 23:53:40.325 | Grupo de usuarios api 1-24 | HTTP Request | 52               | ✗        | 9247  | 538        | 51      | 0             |
| 247      | 23:53:40.510 | Grupo de usuarios api 1-6  | HTTP Request | 205              | ✗        | 9247  | 547        | 138     | 0             |
| 257      | 23:53:40.730 | Grupo de usuarios api 1-17 | HTTP Request | 197              | ✗        | 9247  | 535        | 162     | 0             |
| 2370     | 23:53:35.145 | Grupo de usuarios api 1-4  | HTTP Request | 122331           | ✗        | 21685 | 544        | 122000  | 0             |
| 2408     | 23:53:38.383 | Grupo de usuarios api 1-19 | HTTP Request | 122531           | ✗        | 21297 | 535        | 122325  | 0             |

### 5.2.2. Pruebas de stress.

Las pruebas de stress es para hacer una carga masiva y conocer el momento y con que peticiones se cae el sistema por completo.

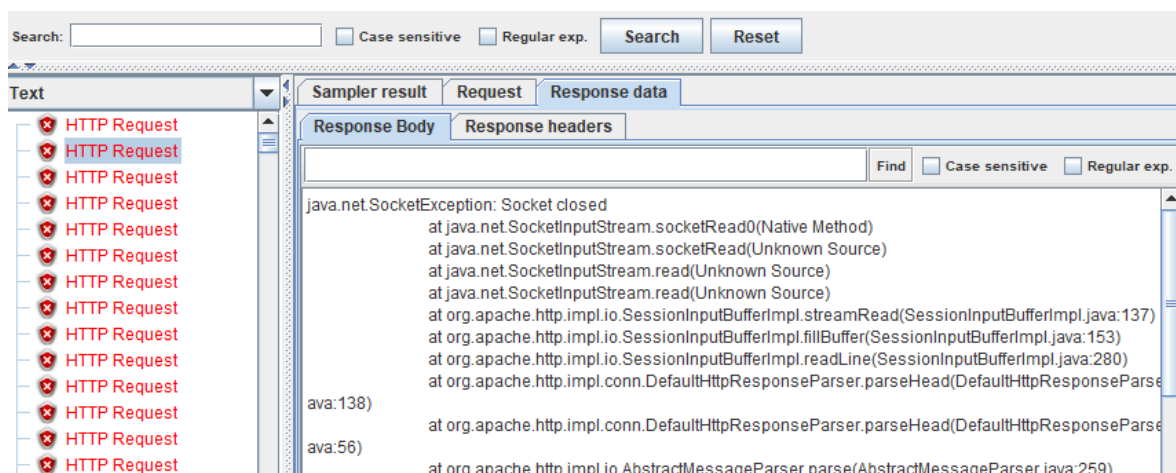
Para esta prueba realizaremos con 1000 usuarios que harán peticiones en un periodo de 5 segundos.

Los resultados de la prueba fueron los siguientes. A los 500 usuarios concurrentes hubo caídas en las peticiones.



Y después de 30 segundos en el bucle de las peticiones. Hubo solo fallas en los request de los usuarios.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE



Lo que nos indica que alrededor de 500 usuarios concurrentes en un periodo de 15 segundos es nuestro punto de inflexión a la caída del servidor local.

## Capítulo 6: Pruebas de campo

### 6. Pruebas en la localidad del caso de estudio.

En el año 2018 se realizó una visita a la Estación Científica Yasuní con fines de voluntariado y vinculación (servicio a la comunidad). En donde se evaluó la infraestructura de TICS y se tomaron los datos de ancho de banda de la estación.

Con la muestra de datos de Internet de la Estación se simulará las pruebas funcionales de la aplicación.

Para ello se limita la velocidad de internet sobre el navegador que se probaron las anteriores pruebas.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

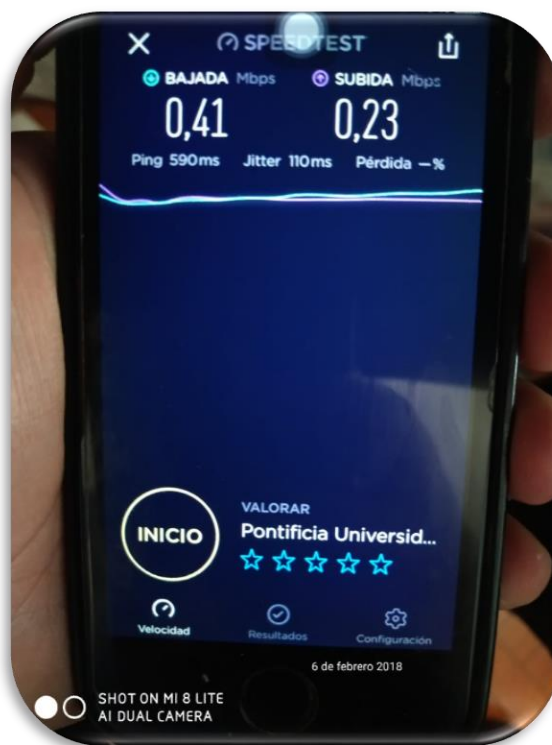


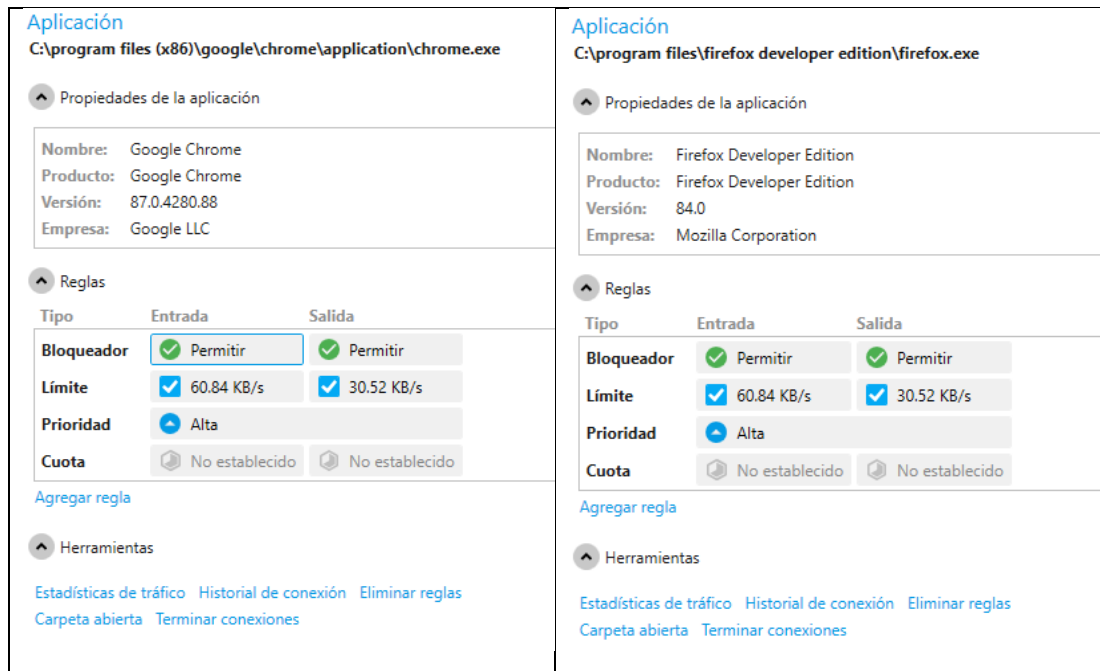
Figura de la muestra de Velocidad de Internet tomada el 6 febrero del 2018

La herramienta que pondrá las restricciones de Internet será el **NetLimiter** en su versión 4.

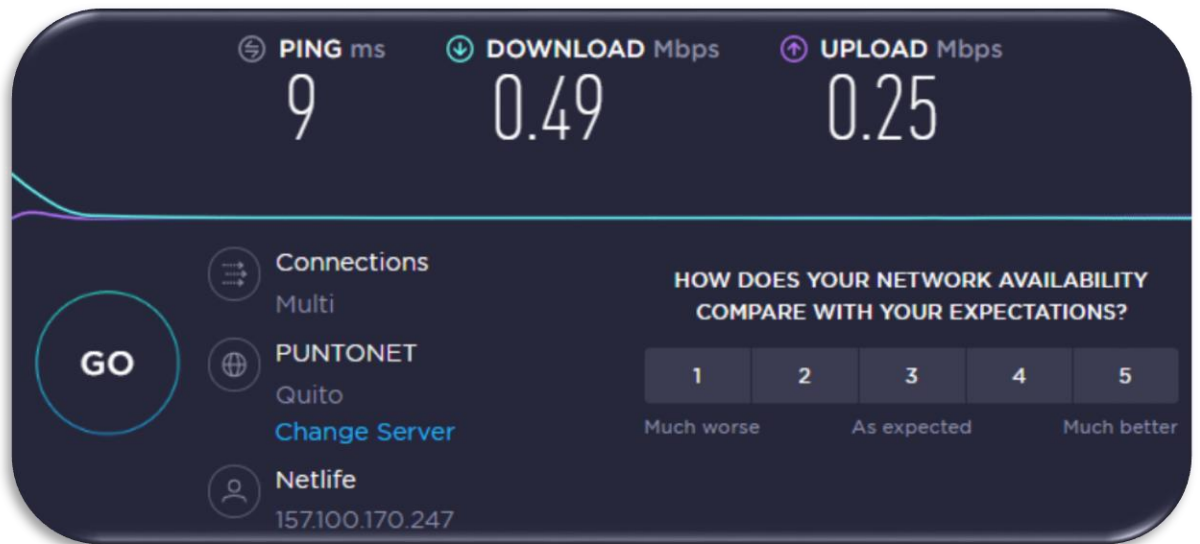
### 6.1. Pruebas funcionales.

Para la prueba funcional limitamos el ancho del navegador Chrome (Usuario que hará la conferencia (Publisher)) y el navegador Firefox Developer (Usuario que se unirá a la conferencia).

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
 Caso de estudio: Estación Científica Yasuní De La PUCE



Con el limitador tenemos el siguiente resultado haciendo el test de velocidad.



Se asemeja a la muestra extraída de la Estación.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
 Caso de estudio: Estación Científica Yasuní De La PUCE

## Flujo de pruebas de conferencia simulada.

### Con Velocidad limitada a lo mínimo.

| Ancho de Banda Mbps | Acción                             | Resultados  | Tiempo de Respuesta                    |
|---------------------|------------------------------------|---|--|
| 0.49 – 0.25         | Creación de conferencia.           | Se abre la videoconferencia del publicador            | 6 Segundos                             |
| 0.49 – 0.25         | Selección de conferencias activas. | El suscriptor selecciona la video conferencia         | 10 Segundos                            |
| 0.49 – 0.25         | Enlazarse a la conferencia.        | El usuario abre el sitio de la suscripción del video. | 15 Segundos y caída de la conferencia. |

| Ancho de Banda Mbps | Acción                             | Resultados  | Tiempo de Respuesta |
|---------------------|------------------------------------|---|---------------------|
| 5.49 – 5.25         | Creación de conferencia.           | Se abre la videoconferencia del publicador            | 3 Segundos          |
| 5.49 – 5.25         | Selección de conferencias activas. | El suscriptor selecciona la video conferencia         | 5 Segundos          |
| 5.49 – 5.25         | Enlazarse a la conferencia.        | El usuario abre el sitio de la suscripción del video. | 8 Segundos.         |

| Ancho de Banda Mbps | Acción                             | Resultados                                    | Tiempo de Respuesta |
|---------------------|------------------------------------|---|---------------------|
| Sin restricción     | Creación de conferencia.           | Se abre la videoconferencia del publicador    | 2 Segundos          |
| Sin restricción     | Selección de conferencias activas. | El suscriptor selecciona la video conferencia | 3 Segundos          |

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

|                 |                             |   |             |
|-----------------|-----------------------------|---|-------------|
| Sin restricción | Enlazarse a la conferencia. | El usuario abre el sitio de la suscripción del video. | 1 Segundos. |
|-----------------|-----------------------------|---|-------------|

## 6.2. Rendimiento de la aplicación.

El rendimiento de la aplicación bajo condiciones de un ancho de banda limitada, de acuerdo a los datos tabulados, es **regular**, ya que con 2 subscriptores se cae el servicio y se pierde la conexión con el servidor multimedia.

Cuando tenemos un ancho de banda limitada a una velocidad estándar, como se vio en los datos tabulados, es **aceptable** y no se pierde la comunicación con el servidor.

Finalmente, cuando no hay restricciones del ancho de banda la comunicación entre aplicación y servidor multimedia es **óptima**.

## 6.3. Conclusiones

**6.3.1.** Se documentó los beneficios e inconvenientes de utilizar Internet satelital en la región amazónica.

**6.3.2.** Se analizó herramientas que faciliten la transmisión de datos multimedia con protocolos y servicios para la implementación adecuada de la aplicación.

**6.3.3.** Se aplicó una metodología de desarrollo adecuada a la implementación de la aplicación.

**6.3.4.** Se implementó un sistema basado en metodologías de desarrollo que gestione streaming utilizando el protocolo

RTCP y RTMP, que permita hacer uso de las herramientas de Google para la búsqueda de artículos científicos.

**6.3.5.** Se validó las pruebas del sistema con enfoques prácticos y simulados en el área de interés.

#### **6.4. Recomendaciones.**

**6.4.1.** El servidor de multimedia solo reconoce nombres sin espacio, ya que las peticiones al servicio por Http con el método GET van sin espacio. Por lo tanto, al momento de crear la videoconferencia, internamente se quitan los espacios para que el servidor multimedia lo reconozca.

**6.4.2.** En la navegación por iframe no todas las paginas permiten el acceso.

**6.4.3.** Para que nuestra aplicación supere la carga de usuarios en las pruebas realizadas se recomienda agregar más capacidad de procesamiento a los servidores en donde se montará la aplicación.

**6.4.4.** Se recomienda tener habilitado una extensión CORS para la petición de servicios o desde ambiente Angular tener un archivo que funcionará como proxy.

## 6.5. Bibliografía.

- AXESSNET. (2020, 12 06). *AXESSNET*. Retrieved from <https://axessnet.com/como-funciona-el-internet-via-satelite-enlace-satelital/>
- Barry Boehm, T. D. (1988). A Spiral Model of Software Development and Enhancement. *IEEE*, 12.
- G., E. C. (2011, 08 18). *slideshare.net*. Retrieved from <https://es.slideshare.net/edisoncoimbra/71-redes-por-satlite-sh>
- Grasa, J. M. (2017, 17 10). Acceso a Internet vía satélite. In J. Mora, *Guías de Tecnología fácil* (p. 24). Madrid: Asociación española ingenieros de telecomunicación. Retrieved from [http://www.coitaoc.org/files/estudios/tecnologia\\_facil\\_7aba8393.pdf](http://www.coitaoc.org/files/estudios/tecnologia_facil_7aba8393.pdf)
- Graydon, M. &. (2019, August 7). *'Connecting the unconnected': a critical assessment of US satellite Internet services*. Retrieved from SAGE JOURNALS: <https://doi.org/10.1177/0163443719861835>
- Grijalva, N. (2012, 10 15). *blogspot*. Retrieved from <http://software1nathalgrijalva.blogspot.com/2012/10/modelo-esprial.html>
- Lagatree, K. (2006). Keep it Together. In K. Lagatree, *Keep It Together: 200+ Tips, Tricks, Lists, and Solution for EverydayLife* (p. 432). Random House Reference.
- Ley De Comercio Electrónico, Ley 67 (Congreso Nacional 05 17, 2002).
- Ley Orgánica De Comunicación, 22 (Legislativo 06 25, 2013).
- López Presmanes, J. L. (2007). Internet: servicios básicos. In *Internet: servicios básicos*. La Habana: CU: Editorial Universitaria.
- Ltd, A. S. (2017, 05 21). *Adobe Flash Builder*. Retrieved from <http://www.adobe.com/es/products/flash-builder.html>
- Muñoz, J. (2006, 07 25). *maestrosdelweb*. Retrieved from [www.maestrosdelweb.com/intersatelite/](http://www.maestrosdelweb.com/intersatelite/)
- Ordoñez, J. L. (2009). *Videoconferencia*. Mexico: Alfaomega Grupo Editor.
- PUCE. (2017, 07 10). *LA ESTACIÓN CIENTÍFICA YASUNÍ*. Retrieved from <http://www.yasuni.ec>
- REAL ACADEMIA ESPAÑOLA. (2020). REAL ACADEMIA ESPAÑOLA.
- Red5. (2017, 05 21). *Red5.org*. Retrieved from Red5.org: <http://red5.org/>

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

Richard, A. (2017). Can Weather Affect Satellite Internet? *Hearst Newspapers*, 1. Retrieved from Can Weather Affect Satellite Internet?: <http://smallbusiness.chron.com/can-weather-affect-satellite-internet-26822.html>

Scrumalliance. (2017, 07 11). *scrumalliance.org*. Retrieved from <https://www.scrumalliance.org/why-scrum>

Yolanda Martínez, S. d. (2015, 06). *Triplemente marcadas: Desconexiones comunicativas en la Amazonia sur ecuatoriana*. Cuenca: Universidad de Cuenca. Retrieved from Triplemente marcadas: Desconexiones comunicativas en la Amazonia sur ecuatoriana: [http://www.scielo.cl/scielo.php?script=sci\\_arttext&pid=S0719-367X2015000100007](http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0719-367X2015000100007)

## 6.6. Glosario de términos.

**BROADCAST:** Forma de difusión de información desde un emisor a múltiples subscriptores.

**BROADCASTER:** Usuario que realiza la difusión de la información a múltiples subscriptores

**PUBLISHER:** Usuario que publica n transmisiones a n subscriptores.

**SUBSCRIBE:** Entrar o participar en una publicación de medios que está siendo transmitida.

**SUBSCRIBER:** Usuario que se enlaza con la transmisión publicada.

**OPEN SOURCE:** Modelo de desarrollo que tiene una colaboración abierta a la contribución del software.

**HTML:** Lenguaje de marcado para la estructuración y desarrollo de páginas web.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

**DOM:** Interfaz de plataforma que permite la modificación de los atributos del HTML

**FRAMEWORK:** Marco de trabajo basado en estándares de programación.

**ORM:** Modelo que permite relacionar los objetos con las entidades de base de datos.

**KANBAN:** Tablero publicitario que muestra de forma gráfica los estados de realización de las actividades planificadas.

**DISEÑO RESPONSIVO:** Diseño que se adapta las apariencias graficas a diferentes tipos de dispositivos.

**NODE JS:** Entorno de ejecución de código JavaScript

**REQUEST:** Petición que hace un cliente web por HTTP a un servidor.

**REST:** Más conocido como servicio Rest que se definen como una forma simple de comunicación entre cliente servidor.

**JSON:** Formato de texto que representa datos a grapados en una estructura.

**TOKEN:** Recurso digital que nos maneja la información cifrada.

**IFRAME:** Elemento de HTML que nos permite incrustar documentos HTML.

## 6.7. Anexos.

### 6.7.1. Malla de pruebas

#### Pruebas Funcionales de API en Laravel

| Id | Caso de Prueba                             | Descripción   | Funcionalidad / Característica  | Datos / Acciones de Entrada  | Resultado Esperado   | Nivel Cumplimiento |
|----|--|---|---|--|--|--------------------|
| 1  | Registro de Usuario                        | Se crea un registro de nuevo usuario                | Se envía por método post a la siguiente endpoint: videoconferencia_puce/public/api/register | Se manda los datos en formato json de id de rol, nombre de usuario, apellido, email y password | Se espera un registro en bdd y un json con el objeto del usuario                                       | 100%               |
| 2  | Login Usuario                              | Se verifica el email y password                     | Se envía por método post a la siguiente dirección: videoconferencia_puce/public/api/login   | los datos que van en formato json son el email, el password y un booleano si desea el token    | Cuando el email y la password es correcto, se espera el objeto del usuario o el token generado por jwt | 100%               |
| 3  | Obtener todas las conferencias registradas | Se busca obtener todas las conferencias registradas | Se envía por método Get la siguiente url: videoconferencia_puce/public/api/conferencia      | Los datos que viajan por cabecera la autorizacion de token                                     | Lista en formato json de todas las conferencias  | 100%               |
| 4  | Obtener una conferencia                    | Se busca obtener una conferencia                    | Se envía por método Get la siguiente url:   | Los datos que viajan por cabecera la autorizacion de token y por url                           | Se espera un objeto en formato json de   | 100%               |

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

| Id | Caso de Prueba          | Descripción  | Funcionalidad / Característica  | Datos / Acciones de Entrada  | Resultado Esperado   | Nivel Cumplimiento |
|----|-------------------------|--|---|--|--|--------------------|
|    |                         | encia por el id de conferencia                                   | videoconferencia_puce/public/api/conferencia/{id}   | el id de la conferencia  | una conferencia  |                    |
| 5  | Obtener una conferencia | Se busca obtener una conferencia por el nombre de la conferencia | Se envía por método Get la siguiente url: videoconferencia_puce/public/api/conferencia/{nombre}     | Los datos que viajan por cabecera la autorizacion de token y por url el nombre de la conferencia   | Se espera un objeto en formato json de una conferencia     | 100%               |
| 6  | Registrar conferencia   | Se crea un registro de una nueva conferencia                     | Se envía por método Post a la siguiente dirección: videoconferencia_puce/public/api/conferencia     | Los datos que viaja por cabecera el token de la sesión del usuario y un objeto de conferencia: nombre de conferencia y el topic                | Se espera el objeto de conferencia en formato json         | 100%               |
| 7  | Registro de mensaje     | Se registra un mensaje de una conferencia                        | Se envía por método Post con la siguiente url: videoconferencia_puce/public/api/mensaje_conferencia | Se manda en el request el token de sesión en la cabecera y el objeto de mensaje con: el id de la conferencia, el mensaje, y un mensaje adjunto | Se espera el objeto del mensaje registrado en formato json | 100%               |
| 8  | Obtener mensajes        | Se obtiene todos los   | Por petición http se envía por método Get a la  | Se manda en la cabecera el token y por url el id de la conferencia   | Nos devuelve una lista de mensajes                         | 100%               |

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.

Caso de estudio: Estación Científica Yasuní De La PUCE

| Id | Caso de Prueba | Descripción                              | Funcionalidad / Característica   | Datos / Acciones de Entrada | Resultado Esperado | Nivel Cumplimiento |
|----|----------------|--|--|-----------------------------|--------------------|--------------------|
|    |                | mensajes registrados por una conferencia | siguiente url: videoconferencia_puce/public/api/mensaje_conferencia/{id_conferencia} |                             | de la conferencia  |                    |

### 6.7.2. Marco legal

#### 6.7.2.1. Ley Orgánica De Comunicación (Ley Orgánica De Comunicación, 2013).

Según la ley orgánica de comunicación en el **Art. 5 Medios de comunicación social**, nos dice que los medios de contenido gráfico, audio y video que se envíen y recíben por Internet son considerados medios de comunicación social.

Esta aplicación de video conferencia de la PUCE se ampara bajo el **Art. 8 Prevalencia en la difusión de contenidos**. Ya que los propósitos de difusión son de carácter informativo, educativo y cultural.

### **6.7.2.2. Ley Orgánica De Comercio Electrónico (Ley De Comercio Electrónico, 2002).**

**Art. 2.- Reconocimiento jurídico de los mensajes de datos**, nos indica que todo mensaje tiene un valor jurídico.

**Art 8. Conservación de los mensajes de texto**, nos indica que todo mensaje de texto debe ser conservada un repositorio físico o digital. Por ello en el proyecto se conserva todos los datos del usuario en el motor de base de datos.

**Art 9. Protección de los datos:** Este articulo nos muestra que información obtenida directa o indirectamente se necesitará el consentimiento del titular de la información, el mismo que podrá compartir a terceros. Para el cumplimiento de este articulo toda contraseña de usuarios con fin de mantener la privacidad solo para el titular, son cifrados en métodos ajenos a terceros.

### **6.7.3. Marco Espacial**

#### **6.7.3.1. Infraestructura de la Estación Científica Yasuní.**

De acuerdo a la experiencia de usuario se muestra la infraestructura espacial de la Estación Científica Yasuní.

Donde se hizo un análisis de los lugares donde concentran altas, regulares y bajas intensidades de señal.

Implementación de un sistema que gestione streaming basado en el protocolo RTMP que permita una solución de comunicación satelital entre la Estación Científica Yasuní de la PUCE y otras entidades.  
Caso de estudio: Estación Científica Yasuní De La PUCE

