



**Implementación de una arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las Comunicaciones del Agrupamiento de Comunicaciones y Guerra Electrónica (AGRUCOMGE)**

Ramos Vargas, Fabian Daniel

Departamento de Eléctrica y Electrónica

Carrera de Tecnología Superior en redes y telecomunicaciones

Monografía, previo a la obtención del título de Tecnólogo en Redes y Telecomunicaciones

Ing. William Robert, Bastidas Bravo

24 de febrero 2022

Latacunga



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**  
**CARRERA DE TECNOLOGÍA DE REDES Y TELECOMUNICACIONES**

**CERTIFICACIÓN**

Certifico que la monografía, **Implementación de una arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las Comunicaciones del Agrupamiento de Comunicaciones y Guerra Electrónica (AGRUCOMGE)**, fue realizado por el señor **Ramos Vargas, Fabian Daniel**, el mismo que ha sido revisado y analizado en su totalidad por la herramienta de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos y científicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPEL, razón por la cual me permito acreditar y autorizar para que lo sustenten públicamente.

Latacunga, 24 de febrero 2022



Firmado electrónicamente por:  
**WILLIAM ROBERT  
BASTIDAS BRAVO**

.....  
Ing. Bastidas Bravo, William Roberto

C.C.: 0501908636

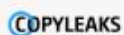
## Reporte de verificación de contenido



Ramos Fabian proyecto Final (Revisión).docx  
Scanned on: 5:44 February 20, 2022 UTC



Identical Words	35
Words with Minor Changes	4
Paraphrased Words	246
Omitted Words	27



Website | Education | Businesses



Firmado electrónicamente por:  
**WILLIAM ROBERT  
BASTIDAS BRAVO**

Ing. Bastidas Bravo, William Roberto  
C.C.: 0501908636



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**  
**CARRERA DE TECNOLOGÍA DE REDES Y TELECOMUNICACIONES**

**Responsabilidad de autoría**

Yo, **Ramos Vargas Fabian Daniel**, declaro que el contenido, ideas y criterios de la monografía: **Implementación de una arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las Comunicaciones del Agrupamiento de Comunicaciones y Guerra Electrónica (AGRUCOMGE)**, es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 24 de febrero 2022



.....  
Ramos Vargas, Fabian Daniel

C.C.: 1805018866



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**  
**CARRERA DE REDES Y TELECOMUNICACIONES**

**Autorización de publicación**

Yo **Ramos Vargas Fabian Daniel** Autorizo a la Universidad de las Fuerzas Armadas Espe publicar la monografía: **Implementación de una arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las comunicaciones del agrupamiento de comunicaciones y guerra electrónica (Agrucomge)**, en el repositorio institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 24 de febrero 2022



Firmado electrónicamente por:  
**FABIAN DANIEL**  
**RAMOS VARGAS**

.....  
Ramos Vargas, Fabian Daniel

C.C.: 1805018866

### **Dedicatoria**

Dedico el presente trabajo de titulación en primer lugar a nuestro Dios por regalarnos la vida y darnos la fuerza que día a día necesitamos para cumplir nuestros objetivos planteados y poder obtener este logro tan anhelado.

A mi familia que ha sido el pilar fundamental en este largo camino y comprendieron mis momentos de ausencia, pero nunca faltaron sus palabras de aliento para seguir adelante y no desmayar, un agradecimiento profundo a mi hermoso hijo Anderson por siempre recibirme con los brazos abiertos y con una sonrisa en su rostro que me motivaban todos los días y de esta manera ser un ejemplo para él.

Ramos Vargas, Fabian Daniel

### **Agradecimiento**

Mi más grande agradecimiento a mi familia y al glorioso Ejército Ecuatoriano por darme la grandiosa oportunidad de continuar con mi preparación profesional e intelectual el cual se verán reflejados en mis actividades dentro y fuera de esta noble institución.

Agradezco a todos mis docentes de la Universidad de las Fuerzas Armadas ESPE, por haber compartidos sus conocimientos sobre los aspectos necesarios para forjar mis conocimientos y de manera especial, al Ing. William Bastidas; tutor de mi proyecto de investigación quien me ha brindado su paciencia, y su gran conocimiento durante la planificación y desarrollo de este trabajo de investigación. Su disposición a dar su tiempo tan generosamente quedo muy agradecido.

Ramos Vargas, Fabian Daniel

<b>Tabla de contenido</b>	
<b>Carátula</b> .....	<b>1</b>
<b>Certificación</b> .....	<b>2</b>
<b>Reporte de verificación de contenido</b> .....	<b>3</b>
<b>Responsabilidad de autoría</b> .....	<b>4</b>
<b>Autorización de publicación</b> .....	<b>5</b>
<b>Dedicatoria</b> .....	<b>6</b>
<b>Agradecimiento</b> .....	<b>7</b>
<b>Tabla de contenido</b> .....	<b>8</b>
<b>Índice de figuras</b> .....	<b>12</b>
<b>Resumen</b> .....	<b>14</b>
<b>Abstract</b> .....	<b>15</b>
<b>Tema</b> .....	<b>16</b>
<b>Antecedentes</b> .....	<b>16</b>
<b>Planteamiento del problema</b> .....	<b>17</b>
<b>Justificación e importancia</b> .....	<b>18</b>
<b>Objetivos</b> .....	<b>19</b>
<i>Objetivo General</i> .....	<b>19</b>
<i>Objetivos Específicos</i> .....	<b>19</b>
<b>Alcance</b> .....	<b>19</b>
<b>Marco teórico</b> .....	<b>20</b>
<b>Historia del protocolo cliente-servidor</b> .....	<b>20</b>



<b>Definición del protocolo cliente-servidor .....</b>	<b>20</b>
<b>Arquitectura cliente-servidor .....</b>	<b>21</b>
<i>Arquitectura del servicio de mensajería .....</i>	<i>22</i>
<b>Características de la arquitectura cliente-servidor .....</b>	<b>23</b>
<b>Protocolos de mensajería instantánea .....</b>	<b>25</b>
<b>Definición de servidores.....</b>	<b>25</b>
<b>Tipos de servidores.....</b>	<b>26</b>
<i>Servidores proxy.....</i>	<i>26</i>
<i>Servidores web .....</i>	<i>27</i>
<i>Servidores de aplicaciones .....</i>	<i>28</i>
<i>Servidores FTP.....</i>	<i>28</i>
<i>Servidores cloud.....</i>	<i>29</i>
<b>Protocolo de mensajería.....</b>	<b>30</b>
<b>Tipos de protocolos de mensajería .....</b>	<b>31</b>
<i>Protocolo Tipo (MSNP) Mobile Status Notification.....</i>	<i>31</i>
<i>Protocolo Oscar .....</i>	<i>31</i>
<i>Protocolo YMSG.....</i>	<i>32</i>
<i>Protocolo XMPP .....</i>	<i>32</i>
<i>Características del protocolo XMPP.....</i>	<i>33</i>
<i>Aplicación del protocolo XMPP .....</i>	<i>34</i>
<i>Arquitectura del protocolo XMPP .....</i>	<i>34</i>
<b>Seguridad en las comunicaciones .....</b>	<b>36</b>
<b>Ingeniería en la seguridad de datos .....</b>	<b>36</b>

Encriptación .....	36
Análisis de vulnerabilidad y control.....	37
Diseño .....	38
Descripción de la arquitectura cliente-servidor.....	38
<i>Partes de la arquitectura cliente-servidor .....</i>	<i>38</i>
Protocolo XMPP.....	39
Descripción del servicio .....	40
<i>Instalación .....</i>	<i>40</i>
<i>Registro.....</i>	<i>40</i>
<i>Comunicaciones .....</i>	<i>40</i>
<i>Agenda .....</i>	<i>41</i>
Funcionamiento.....	41
Servicio de mensajería o Chat .....	41
<i>Diagrama de flujo del servicio de chat.....</i>	<i>42</i>
<i>Diagrama de flujo del servicio de contactos.....</i>	<i>43</i>
<i>Diagrama de flujo del servicio de llamadas a contactos nuevos .....</i>	<i>44</i>
<i>Diagrama de flujos del registro de eventos.....</i>	<i>45</i>
<i>Diagrama de flujos de la transferencia de imágenes .....</i>	<i>46</i>
<i>Diagrama de flujos de la toma de imágenes instantáneas .....</i>	<i>47</i>
<i>Diagrama de flujos de video llamada.....</i>	<i>48</i>
<i>Estándares del sistema.....</i>	<i>49</i>
Propuesta del sistema de comunicación militar .....	49
<i>Lista de componentes.....</i>	<i>50</i>

<b>Fases del proceso .....</b>	<b>51</b>
<i>Establecer direcciones de XMPP.....</i>	<i>51</i>
<i>Adquisición de un servidor con almacenamiento .....</i>	<i>51</i>
<i>Elección de la plataforma del servicio de mensajería.....</i>	<i>52</i>
<i>Determinación de las características del software.....</i>	<i>52</i>
<b>Desarrollo y resultados.....</b>	<b>54</b>
<b>Diagrama de Conexión Cliente AGRUCOMGE.....</b>	<b>54</b>
<b>Conexión a la infraestructura de virtualización - VMWARE .....</b>	<b>55</b>
<b>Descarga del servidor .....</b>	<b>60</b>
<b>Pasos para instalar Openfire en Ubuntu 20.04 LTS .....</b>	<b>60</b>
<b>Instalador Web .....</b>	<b>61</b>
<b>Configuración Openfire en Ubutu 20.04 LTS .....</b>	<b>65</b>
<b>Desarrollo de la Apk para dispositivos Android.....</b>	<b>67</b>
<b>Página web para la difusión del aplicativo.....</b>	<b>68</b>
<b>Conclusiones .....</b>	<b>69</b>
<b>Recomendaciones .....</b>	<b>70</b>
<b>Bibliografía .....</b>	<b>71</b>
<b>Anexos .....</b>	<b>75</b>

## Índice de figuras

<b>Figura 1</b> <i>Diagrama simple de una red de datos LAN.</i> .....	21
<b>Figura 2</b> <i>Arquitectura de un servicio.</i> .....	23
<b>Figura 3</b> <i>Diagrama de un servidor Proxi.</i> .....	27
<b>Figura 4</b> <i>Esquema de un servidor Web.</i> .....	28
<b>Figura 5</b> <i>Esquema de un servidor FTP.</i> .....	29
<b>Figura 6</b> <i>Diagrama de un servidor FTP.</i> .....	30
<b>Figura 7</b> <i>Arquitectura de XMPP.</i> .....	35
<b>Figura 8</b> <i>Comunicación entre clientes XMPP.</i> .....	35
<b>Figura 9</b> <i>Diagrama de flujo del servicio de chat.</i> .....	42
<b>Figura 10</b> <i>Diagrama de flujo del servicio de contactos.</i> .....	43
<b>Figura 11</b> <i>Diagrama de flujo de llamadas a contactos nuevos.</i> .....	44
<b>Figura 12</b> <i>Diagrama de flujo del registro de eventos.</i> .....	45
<b>Figura 13</b> <i>Diagrama de flujo de la transferencia de imágenes.</i> .....	46
<b>Figura 14</b> <i>Diagrama de flujo de la toma de imágenes.</i> .....	47
<b>Figura 15</b> <i>Diagrama de flujo de la toma de imágenes.</i> .....	48
<b>Figura 16</b> <i>Propuesta del sistema de comunicación militar.</i> .....	50
<b>Figura 17</b> <i>Diagrama de Conexión del servidor.</i> .....	54
<b>Figura 18</b> <i>Editor de conexión VPN.</i> .....	56
<b>Figura 19</b> <i>Datos de usuario VPN.</i> .....	56
<b>Figura 20</b> <i>Datos IP asignada.</i> .....	57
<b>Figura 21</b> <i>Instalación de plugin.</i> .....	58
<b>Figura 22</b> <i>Configuración de la VM.</i> .....	58
<b>Figura 23</b> <i>Resumen de la VM.</i> .....	59
<b>Figura 24</b> <i>Descarga servidor.</i> .....	60

<b>Figura 25</b> <i>Comprobación del funcionamiento</i> .....	61
<b>Figura 26</b> <i>Instalador del servidor, selección del idioma</i> .....	62
<b>Figura 27</b> <i>Distribución del origen de datos</i> .....	62
<b>Figura 28</b> <i>Configuración de la fuente de datos- Conexión Estándar</i> .....	63
<b>Figura 29</b> <i>Configuración del perfil</i> .....	63
<b>Figura 30</b> <i>Cuenta del administrador</i> .....	64
<b>Figura 31</b> <i>Finalización del proceso</i> .....	64
<b>Figura 32</b> <i>Instalación de la consola de administración</i> .....	65
<b>Figura 33</b> <i>Configuración del servidor de la consola de administración</i> .....	66
<b>Figura 34</b> <i>Revisión de los usuarios</i> .....	66
<b>Figura 35</b> <i>Código Fuente</i> .....	67
<b>Figura 36</b> <i>Página web Aplicativo Chat Militar</i> .....	68

## Resumen

Se analizó las características de la seguridad de las comunicaciones del Agrupamiento de Comunicaciones y Guerra Electrónica (AGRUCOMGE) y los posibles modelos de arquitecturas de mensajería que son útiles para mejorar el sistema. El problema es la posible pérdida de información en el agrupamiento de comunicaciones, así como la dificultad de los servidores de mensajería utilizados por el personal administrativo por no tener una arquitectura de mensajería instantánea XMPP. El objetivo es implementar una arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las comunicaciones del Agrupamiento de Comunicaciones y Guerra Electrónica (AGRUCOMGE). Se utilizó el método deductivo y la investigación exploratoria para analizar la información de las fuentes de referencia, con temas como la descripción de la arquitectura, las partes, la descripción del servicio, instalación, registro, comunicaciones, agenda y funcionamientos, así como los servicios de mensajería. Resultó un diagrama de conexión cliente AGRUCOMGE que posee varias rutas de comunicación para que varios clientes puedan conectarse en un determinado tiempo. Se concluye que para el funcionamiento se debe tener en cuenta los procedimientos; como requisitos y las condiciones para que la app pueda funcionar correctamente, así como el procedimiento para instalar el aplicativo para la seguridad del chat - Militar.

Palabras clave:

- **ARQUITECTURA CLIENTE - SERVIDOR**
- **AGRUPAMIENTO DE COMUNICACIONES**
- **SERVIDOR XMPP**
- **APLICATIVO DE MENSAJERÍA INSTANTÁNEA**

**Abstract**

The characteristics of the security of the communications of the Communications Group and Electronic Warfare (AGRUCOMGE), and the possible models of messaging architectures that are useful to improve the system were analyzed. The problem is the possible loss of information in the grouping of communications, as well as the difficulty of the messaging servers used by the administrative staff for not having an XMPP instant messaging architecture. The objective is to implement a client architecture – instant messaging server (XMPP), for the security of the communications of the Communications Group and Electronic Warfare (AGRUCOMGE). The deductive method and the exploratory research were obtained to analyze the information from the reference sources, with topics such as the description of the architecture, the parts, the description of the service, installation, registration, communications, agenda and operations, as well as the services. messaging. It resulted in an Agrucomge client connection diagram that has the communication paths between two or more clients so that they can connect at the same time. It is concluded that for the operation the procedures must be taken into account; as requirements and conditions so that the app can work correctly, as well as the procedure to install the application for chat security - Military.

Key words:

- **CLIENT - SERVER ARCHITECTURE**
- **GROUPING OF COMMUNICATIONS**
- **XMPP SERVER**
- **INSTANT MESSAGING APPLICATION**

## Capítulo I

### 1. Tema

Implementación de una arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las comunicaciones del Agrupamiento de Comunicaciones y Guerra Electrónica (Agrucomge).

#### 1.1. Antecedentes

En la actualidad las empresas utilizan protocolos de comunicación de redes y servidores. Es por ello que la seguridad de este proceso es uno de los puntos más importantes dentro de la organización a nivel mundial, muchos investigadores han desarrollado diferentes técnicas de seguridad teniendo en cuenta el avance de la tecnología mejorando la complejidad de cada sistema y encriptación de datos.

Hernán Mendoza, en su proyecto de investigación desarrollado en el año 2021 cuyo tema es “Análisis del desempeño del protocolo de comunicación XMPP en una Red IOT” ha realizado la implementación de un Protocolo de mensajería donde se obtuvo que la comunicación funciona de manera totalmente funcional, obtiene una media del tiempo que responde en 1,506s, de tal manera que la comunicación solo un segundo y medio en dar el cumplimiento de la actividad (Mendoza, 2021).

Elizabeth Becerra, en su proyecto de investigación desarrollado en el año 2016 cuyo tema es “Implementación de monitoreo de red utilizando los protocolos ICMP y SNMP” creo e implementó un Cacti en Linux con lo cual se monitorea los dispositivos y la lista de datos a través del interruptor cisco. Como la parte probatoria del proyecto se utilizó una conexión portátil a un puerto libre del interruptor, con lo cual se formó todo el flujo de figuras y gráficos de Cacti (Becerra, 2016).



La asistencia de la comunicación XMPP se lo debe realizar en base de los nodos Raspberry y en los servicios de Windows y Android, con lo cual se utiliza las librerías con lo que se desarrolla cada actividad de manera independiente.

## **1.2. Planteamiento del problema**

En el momento que se fundó la Universidad de Fuerzas Armadas ESPE con sede en Latacunga y teniendo en cuenta el avance tecnológico se ha establecido plataformas con el que se comparten los datos XML que se utiliza en los servicios de mensajería debido a su fácil acceso y sencillez del XML que parten de la extracción del XMPP. Sin embargo, no se cuenta con una arquitectura cliente servidor para la seguridad de las comunicaciones, por lo cual genera un alto índice de inseguridad y vulnerabilidad de las comunicaciones y transmisión de datos en el Agrupamiento de Comunicaciones y en la Guerra Electrónica.

La consecuencia de no tener una arquitectura cliente-servidor de mensajería instantánea (XMPP) propia y administrada por personal militar, genera vulnerabilidades de información dentro Agrupamiento de Comunicaciones y en la Guerra Electrónica. Además, que se pierde tiempo en el envío y recepción de órdenes y disposiciones en tiempo real.

De no solucionarse el problema se dificultaría la privacidad de la información delicada, que se genera en el agrupamiento tanto como disposiciones y órdenes del mando superior. Además, que no se contara con sistema de mensajería privado y controlado por militares en servicio activo.

### **1.3. Justificación e importancia**

Debido a los avances tecnológicos dado en los últimos años las redes XMPP han ido evolucionando, es por ello que existen múltiples implementaciones de los estándares para clientes, servidores, componentes y bibliotecas y estos pueden estar aislados a la red pública es por ello que se debe establecer un sistema de seguridad para proteger su información.

Para la Universidad de Fuerzas Armadas es muy importante la implementación de una arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las comunicaciones del agrupamiento de comunicaciones y guerra electrónica (Agrupomge) que permitan mejorar el sistema seguro de la información.

Con la arquitectura cliente-servidor de mensajería XMPP, se puede hacer diferentes funcionalidades sobre este servidor, para mantener la interoperabilidad, ya que las extensiones más comunes son las más gestionadas.

La implementación de este sistema de control de acceso y seguridad tiene como fin precautelar la seguridad e integridad de los servidores de mensajerías XMPP además de minimizar los problemas de redundancia en los servicios de utilización. Los beneficiarios de este tema de investigación es el personal militar del Agrupamiento de Comunicaciones y en la Guerra Electrónica, precautelando los sistemas de seguridad de la comunicación entre servidor- cliente además que el proyecto encaminará al aprendizaje de la telecomunicación mediante la práctica con la manipulación los dispositivos tecnológicos.

## **1.4. Objetivos**

### **1.4.1. *Objetivo General***

Implementar una arquitectura cliente-servidor de mensajería instantánea.

### **1.4.2. *Objetivos Específicos***

- Determinar la parte teórica de la arquitectura cliente-servidor basado en el protocolo XMPP.
- Investigar las metodologías de la arquitectura cliente-servidor.
- Desarrollar la aplicación móvil que permita el envío y recepción del mensaje (multimedia y voip) al lado del cliente.

## **1.5. Alcance**

El presente proyecto abarcará la implementación de una arquitectura cliente – servidor de mensajería instantánea (XMPP), la cual permita tener un alto índice de seguridad de las comunicaciones del agrupamiento de comunicaciones y guerra electrónica (Agrucomge). Además, con la estructura cliente servidor se podrá ver la utilización de los sistemas de mayor escala y su seguridad frente a los diferentes escenarios de inseguridad electrónicos, descentralizados, extensible y accesible para los clientes.

## Capítulo II

### 2. Marco teórico

#### 2.1. Historia del protocolo cliente-servidor

A mediados de los 60 es decir en 1964 con el IBM OS/ 360 se remonta la arquitectura cliente-servidor. A este servidor se le enviaba las solicitudes para ejecutar las posibles labores y el sistema le retroalimentaba la acción.

Al comienzo de la tecnología en computación, la arquitectura era unas grandes computadas que ocupara gran espacio y tenía un alto valor económico. Aún no había las computadoras personales o microcomputadores, por lo cual el uso de estos equipos era compartido por varios usuarios. En base a estas acciones se minimiza y se optimiza el nivel de rendimiento, con lo cual se consigue utilizar las horas perdidas, los horarios nocturnos para realizar las diferentes labores, con esto se utilizaba estos horarios y los momentos libres de los usuarios y se manejaba de mejor manera el trabajo de la máquina (Alsina, 2019).

Debido a que en esta época las empresas contienen servidores que sólo son computadoras que muchos de ellos no están en el alcance del manejo de algún usuario, por lo cual no se puede verificar de mejor manera eficiente los programas y servicios de operación (como ficheros, impresoras, almacenamiento,) utilizados desde los puestos de trabajo.

#### 2.2. Definición del protocolo cliente-servidor

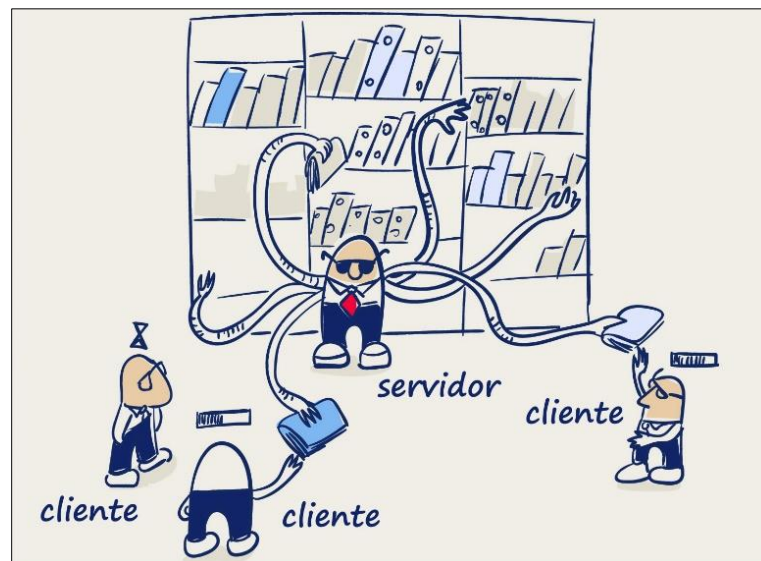
El protocolo cliente-servidor permiten crear un modelo de comunicación entre diferentes puntos que interactúan entre sí: en un primer punto se tiene una Pc o dispositivo electrónico y del otro una arquitectura informática (conocido de manera

puntual como servidor, cuyo principio es el diseño de una computadora activa para dar solución a los servicios de solicitudes y mensajerías) (Alsina, 2019).

En este protocolo el servidor tiene que poseer la potencia suficiente para contestar a la demanda. Es por ello que se toma como ejemplo Amazon que contiene una red o serie de servidores que trabajan de forma continua y simultánea.

### Figura 1

*Diagrama simple de una red de datos LAN.*



*Nota.* En la figura se visualiza el diagrama un proceso cliente-servidor. Obtenida de:  
(Alsina, 2019)

### 2.3. Arquitectura cliente-servidor

Cuando se cuenta con dos aplicaciones en una red y que exista una comunicación, uno de los componentes de los programas debe esperar el requerimiento del otro medio de comunicación como los clientes. En una estructura cliente servidor un programa o aplicación suele esperar de manera pasiva hasta que el inicio se dé por el otro cliente, este método es conocido como un paradigma del cliente servidor. El

programa que está esperando de forma pasiva se le conoce como servidor y el que da inicio a sistema se le denomina cliente.

Características de los clientes y servidores.

**Cliente:**

1. Es la interacción que hace el usuario y su duración depende de la necesidad de la actividad que se está realizando.
2. Es encaminado de forma local entre el usuario y el sistema del computador.
3. Da un modelo activo entre un contacto y un servidor adquirido y seleccionado.

**Servidor:**

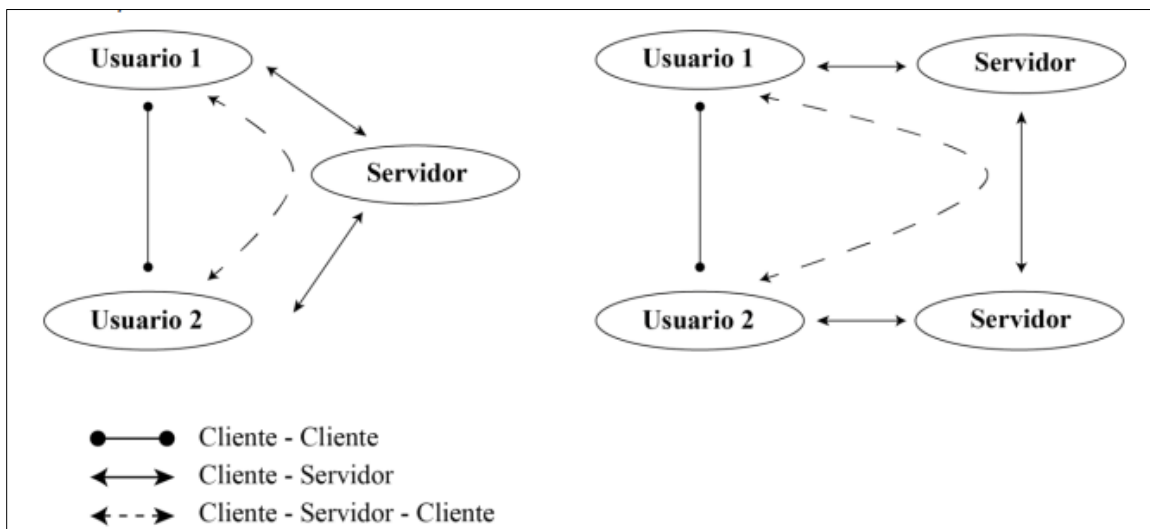
1. Es encaminado en un computador de uso compartido.
2. Es necesario que se conecten otros clientes y dar paso al requerimiento.
3. Acepta su inicialización por clientes los diversos clientes, y a los que es contactado les ofrece un servicio definido (García, 2016).

**2.3.1. Arquitectura del servicio de mensajería**

Conocer la historia de las arquitecturas de servicios de mensajería es importante para comprender las arquitecturas utilizadas en los protocolos de mensajería instantánea, cuando los dispositivos que se quieren conectar entre sí no son compatibles porque utilizan diferentes parámetros, fue necesario crear estándares para que todos pudieran seguirlos. y se podía lograr una buena comunicación de los dispositivos como en el año 2000, cuando la Internet Engineering Task Force (IETF) realizó una publicación de dos Request for Comments (RFC) acerca de la arquitectura del servicio de mensajería. Con el tiempo esto se conoció IMPPWG (Instant Messagon and Precence Protocol Working Group), por lo cual en al año 2001 nació el primer protocolo XMPP (Mendoza, 2021).

**Figura 2**

*Arquitectura de un servicio.*



*Nota.* En la figura se visualiza el diagrama de la arquitectura usuario-servidor. Obtenida de: (Mendoza, 2021)

En la figura se observa que el usuario 1 tiene una conexión con el servidor, el cual le hará llegar las notificaciones que se encuentren disponibles. Después el usuario realiza una petición dirigida al servidor para que se empiece la comunicación, este mensaje es recibido por el servidor y lo reenvía al usuario 2.

#### **2.4. Características de la arquitectura cliente-servidor**

La arquitectura cliente servidor está relacionado con las siguientes características:

- La composición o combinación que se da entre el cliente, el servidor y el usuario que crea la interacción de los recursos adquiridos y proporcionados. El método de interacción del cliente da la interfaz del usuario con respecto a los demás componentes del sistema.

- Al momento de referirnos a las tareas dirigidas entre el cliente y el servidor se hace énfasis a la gran variedad de requerimientos, esto con respecto a ciertos aparatos de cómputo y los datos o rapidez de respuesta del procesador, su nivel de memoria y las capacidades con las que cuenta el disco.
- Se da una gran variedad de distinción de las funciones que cumple el servicio y que papel desempeñan dentro del proceso cliente servidor.
- El Establecimiento de las relaciones son diferentes, en este caso el servidor poseerá la opción de dar un sistema de servicio dirigido a cierto número de clientes, de manera regular y con el respectivo control del acceso y el compartimiento de los recursos.
- Los procesos activos están conformados por los clientes, por la razón que estos son los encargados de realizar las diferentes solicitudes del servicio al servidor. Por otra parte, los servidores poseen un carácter pasivo, ya que deben estar esperando el acceso de los clientes.
- Entre el cliente y los servidores solo existe una relación que es el intercambio de los mensajes entre los dos. Esta forma de compartición de información se la conoce como mensaje y es el encargado de pedir y entregar las diferentes solicitudes para el servicio.
- La escalabilidad vertical, así como horizontal puede ser utilizado en cualquier modelo cliente servidor. Con la escalabilidad de tipo horizontal se puede tener más facilidad en las estaciones de trabajo con una forma activa, sin la necesidad de que se afecte el rendimiento. En cambio, mediante el uso de la escalabilidad vertical se logra una mejor utilización y características de los servidores (García, 2016).



## **2.5. Protocolos de mensajería instantánea**

Dado por las personas que requieren una forma directa y de manera sencilla de comunicarse nace a mensajería instantánea que se basó en el uso de los nuevos y avances tecnológicos de la comunicación, ya que este problema no se podía resolver por el uso del correo electrónico. La solución es el chat, el envío y la recepción de mensajes de texto basados en software en tiempo real, siempre y cuando el dispositivo que intervenga esté activo. Adicionalmente, dependiendo del protocolo que se utilice para este fin, se puede requerir un servidor para controlar el proceso. (Velásquez, 2021).

Dado por los avances de la tecnología y el uso de la mensajería se logra que se realice video llamadas mediante el uso de los mismos protocolos, por lo cual se hace énfasis las siguientes funciones:

- Notificación de exterior: notifica a los usuarios cuándo conectarse ahora.
- Buzón: Almacena mensajes enviados a usuarios que no están en línea.
- Transferencia de archivos: Realiza el envío y recepción de los archivos.
- Comunicación de voz: la comunicación se la hace a través de un micrófono y mediante el uso de las llamadas telefónicas.
- Chat de video: puede agregar el uso compartido de la cámara durante una llamada.

## **2.6. Definición de servidores**

Este servidor es el encargado de proporcionar y dar los servicios especiales llamados clientes que se pueden usar localmente o en una red. Esto servicios hacen dependencia del modelo del software que posee el servidor y del sistema de comunicación del cliente-servidor.

Un servidor es un software que realiza tareas específicas en nombre del usuario. El término servicio también se usa para referirse a una computadora física que ejecuta software diseñado para hacer que los datos estén disponibles para que otras máquinas puedan usarlos. El servidor entrega información a las computadoras conectadas a él. Cuando los usuarios se conectan al servidor, pueden acceder a programas, archivos y otra información en el servidor. En Internet, el modelo de servidor web se le conoce como la computadora que usa el protocolo de http para hacer el envío de las diferentes páginas web hacia la computadora del usuario cuando el usuario lo solicita. La mayoría de las personas que usan Internet probablemente usan servidores web, servidores de correo y servidores de bases de datos. (García, 2016).

## **2.7. Tipos de servidores**

El servidor realiza tareas como proteger servicios, transmitir video a su intranet y proteger páginas web. Teniendo en cuenta las tareas y las actividades a realizar se deducen la clasificación de los servidores:

### **2.7.1. Servidores proxy**

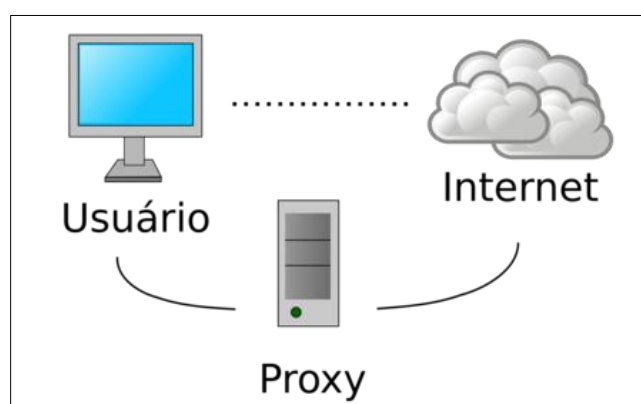
Entre el navegador del programa cliente se realiza la colocación del servidor proxy dirigido hacia un servidor externo (generalmente se refiere a otro servidor web) estos son utilizados para dar solicitudes, mejorar el rendimiento y compartir conexiones. El servidor se encuentra entre la aplicación cliente (como un navegador web) y el servidor real. Intercepta todas las solicitudes al servidor real para determinar si puede ejecutar la solicitud por sí mismo. Si no, envía tu solicitud al servidor real. (García, 2016).

Los servidores proxy tienen dos propósitos principales:

- Mejorar el rendimiento: un servidor proxy puede mejorar considerablemente el rendimiento de un grupo de usuarios al guardar los resultados de todas las solicitudes durante un período de tiempo específico.
- Filtro solicitudes: Los proxis también se pueden usar para filtrar solicitudes.

### Figura 3

*Diagrama de un servidor Proxi.*



*Nota.* En la figura se visualiza el diagrama del diagrama de un servidor proxy. Obtenida de: (Mendoza, 2021)

#### **2.7.2. Servidores web**

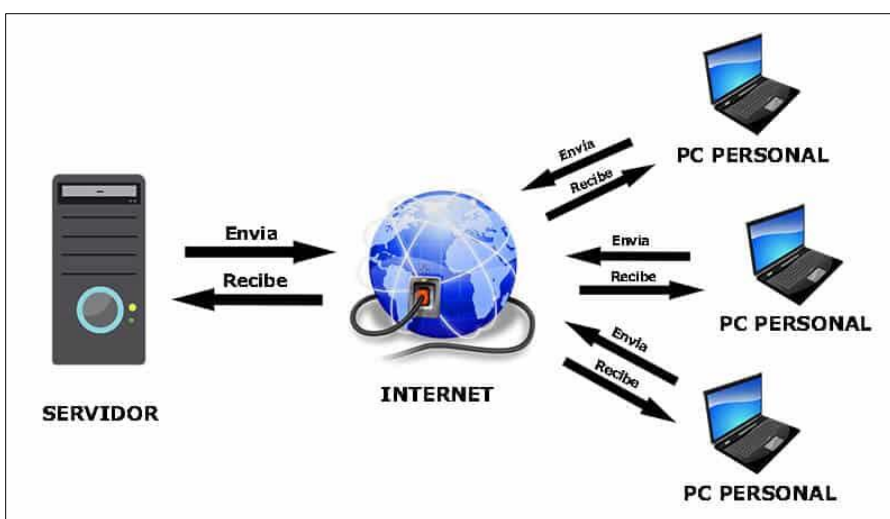
Estos servidores son los que ejecutan sitios web. Tiene una dirección IP y posiblemente un nombre de dominio. Luego, el servidor descarga una página llamada index.html y la envía a su navegador.

Cualquier computadora puede convertirse en un servidor web instalando el software del servidor y conectando el dispositivo a Internet. Hay muchas aplicaciones de software de servidor web, incluido el software gratuito NCSA y Apache, así como paquetes de software comercial de Microsoft, Netscape y otros. Básicamente, un

servidor web alimenta el navegador con contenido estático, transfiere archivos y lo pone a disposición del navegador del usuario a través de la red. Este intercambio se realiza a través del navegador y el servidor comunicándose entre sí a través del protocolo HTTP (García, 2016).

#### Figura 4

*Esquema de un servidor Web.*



*Nota.* En la figura se visualiza el esquema del diagrama de un servidor Web. Obtenida de: (Mendoza, 2021)

#### **2.7.3. Servidores de aplicaciones**

A veces llamado un tipo de middleware (software que conecta dos aplicaciones), un servidor de aplicaciones ocupa la mayor parte del espacio entre el servidor de la base de datos y el usuario, y generalmente conecta los datos del servidor de la base de datos con los usuarios (García, 2016).

#### **2.7.4. Servidores FTP**

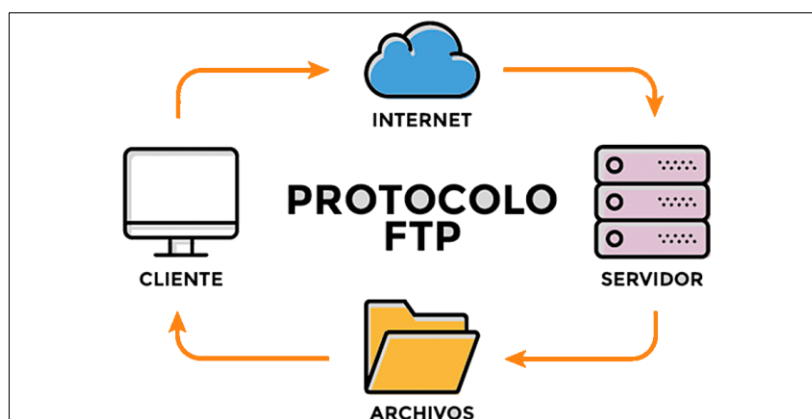
Uno de los servicios más antiguos de Internet, el Protocolo de transferencia de archivos, le permite transferir de forma segura uno o más archivos entre diferentes

computadoras, manteniendo sus archivos seguros, organizados y bajo control. La seguridad se volvió un tema de gran interés. A lo largo de los años, los servidores ftp se han comunicado "públicamente" con los clientes, lo que significa que los inicios de sesión y las contraseñas pueden interceptarse fácilmente. Los servidores FTP como BulletProof FTP, SecureFTP, SurgeFTP, TitanFTP y WS\_FTP ahora son compatibles con SSL/TLS y usan el mismo cifrado que en los sitios web seguros (García, 2016).

FTP es un protocolo cliente-servidor. Aquí el cliente solicita archivos y el servidor los procesa. Establecer una conexión FTP requiere dos canales básicos: un comando de inicio, un canal de comando para transferir información básica y un canal de datos para transferir datos de archivos entre dos dispositivos.

### Figura 5

*Esquema de un servidor FTP.*



*Nota.* En la figura se visualiza el diagrama del esquema de un servidor FTP. Obtenida de:  
(Mendoza, 2021)

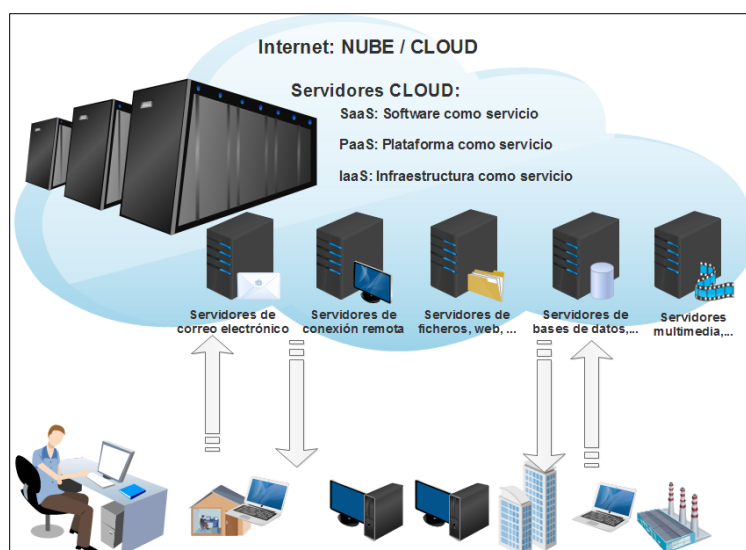
#### **2.7.5. Servidores cloud**

Son utilizados por empresas que solo alquilan espacio en sus servidores a personas o empresas para almacenar información de forma remota. Se utilizan para

almacenar grandes cantidades de datos, protegiendo así la información de una organización o individuo. Suele utilizarse para "asegurar" la información porque el proveedor garantiza que los datos no se perderán ni se filtrarán, además de garantizar el acceso inmediato. (DocuSign, 2021).

## Figura 6

*Diagrama de un servidor FTP.*



*Nota.* En la figura se visualiza el diagrama del esquema de un servidor FTP. Obtenida de: (Mendoza, 2021)

## 2.8. Protocolo de mensajería

El XMPP, Extensible Messaging Presence Protocol, se refiere a una regla de la mensajería instantánea y que puede hasta ser usado de manera accidental. XMPP es conocido como un protocolo de tipo abierto. Esta fue lanzada en 1999 bajo el nombre de jabber. Tiene interfaz de programación en lenguajes como C, Python y Java, multiplataforma y lo puede buscar en Linux, Windows, Android y Mac. Lo utilizaron en Facebook y Google, ya que, en el año 2013, Google para cubrir la mensajería implemento un servicio gtalk ampliamente utilizado, usando XMPP. Con la llegada de

Hangouts, fue reemplazado por su propio protocolo. Esta es una gran opción para enviar mensajes con algunas garantías de seguridad y anonimato (García, 2016) .

Para comprender los pros y los contras del protocolo XMPP, es necesario comprender sus capacidades, y los protocolos de mensajería instantánea más utilizados son MSNP, OSCAR, YMSG y Skype.

## **2.9. Tipos de protocolos de mensajería**

Entre los tipos de protocolo se tiene: el Mobile Status XMPP, protocolo Oscar, protocolo YMSG y el protocolo XMPP.

### **2.9.1. Protocolo Tipo (MSNP) Mobile Status Notification**

MSNP fue creado por Microsoft en 1999 y utilizado por su famoso cliente Windows Live Messenger, es muy popular debido a que se instala con el sistema operativo Windows y su última versión es MSNP24. Tiene su propio código y los tipos de servidores que usan son Message Server (NS), Dispatch Server (DS), y Control Panel Server (SS). La comunicación se inicia con una conexión al servidor de despacho (messenger.hotmail.com puerto 1863), el cual redirige al servidor de mensajes, donde comienza el siguiente proceso (García, 2016).

Para este tipo de protocolo se debe tener en cuenta lo siguiente:

- Que su mensajería sea Simple Object Access Protocol (SOAP) y que posea las credenciales.
- Debe tener una respuesta de tipo RST.srf
- Durante el proceso de autenticación, se proporciona una lista de contactos y su estado.

### **2.9.2. Protocolo Oscar**

Open System for Real-Time Communication (OSCAR) es un protocolo

patentado de AOL que utilizan los clientes de AIM e ICQ. Al igual que MSNP, para realizar diferentes operaciones utiliza gran cantidad de servidores, estos se refieren al acceso (AS, login.oscar.aol.com) y (BOSS) que es el servidor principal (García, 2016).

El proceso de este protocolo debe seguir los siguientes pasos:

- Inicio de sesión en el servidor AS.
- Tasa de cookie
- Inicio de sesión del servidor BOSS
- Confirmación la información de inicio de sesión
- Transferir los contactos y lista de servicios

### **2.9.3. Protocolo YMSG**

Este pertenece a Yahoo! y muy similar a Windows Live Messenger, pero con adaptabilidad a la red. Su arquitectura también es estilo cliente-servidor, pero la diferencia es el tipo de servidor que el cliente accede es de carácter aleatorio y desconocido, esto se logra con un dominio como csNN.msg.dcn.yahoo.com donde NN es cualquier número. (García, 2016).

Este protocolo no utiliza encriptación de comunicaciones, por lo que se excluyen los clientes que utilizan encriptación SSL.

### **2.9.4. Protocolo XMPP**

XMPP, como todos los demás protocolos, define el formato de transmisión de datos entre dos o más entidades que se comunican entre sí, en este caso el cliente y el servidor. Existen varios modelos de XMPP que se los puede encontrar en la Web y este pertenece a una red de información conectada entre sí. Este protocolo usa XML por lo que se mejora el aprovechamiento el nivel de conocimiento y se mejora la calidad del lenguaje y su compatibilidad con un lenguaje dado en una cierta



comunidad (García, 2016).

### **2.9.5. Características del protocolo XMPP**

Las características de XMPP se pueden dividir en dos temas principales, lo que permite una comprensión más detallada del protocolo, por lo que se explican por separado a continuación:

#### **Servicios**

- **Cifrado de canal:** el cifrado crea una conexión entre dos servidores, o entre un cliente y un servidor.
- **Autenticación:** con este se logra que el servidor identifique los dispositivos que están en la red e interactúan e intentan comunicarse.
- **Presencia:** Responsable de notificar a un dispositivo si el otro está activo y si el dispositivo ya está en otro chat, si el dispositivo está activo.
- **Lista de Contactos:** Esta es una colección seleccionada de contactos de amigos, conocidos o contactos cercanos con los que cree que tiene la intención de comunicarse en un momento dado.
- **Comunicación entre pares:** permite que dos dispositivos se comuniquen y realicen el proceso de flujo de mensajes; entre estos dispositivos se tiene dos usuarios y sus respectivos clientes uno o dos nodos y sus servidores.
- **Mensajería multi-personas:** se da en una sala en la que varios usuarios ingresan de forma libre y comparten información entre sí.
- **Notificaciones:** se da las alertas de acuerdo al evento o imprevisto que lo acciona.
- **Detección de servicios:** mediante esta detección se puede verificar si existe nuevos ingresos o hay usuarios utilizando el sistema.

- Capacidades de publicidad: da una abreviada forma de notación con la que se detecta los datos y servicios que almacena el cache de acuerdo a las funciones que fueron atendidas por la red.
- Gestión del flujo de trabajo: Esta función permite que el flujo o modelo de trabajo permite que una entidad interactúe con otra en función de los eventos. (García, 2016).

#### **2.9.6. Aplicación del protocolo XMPP**

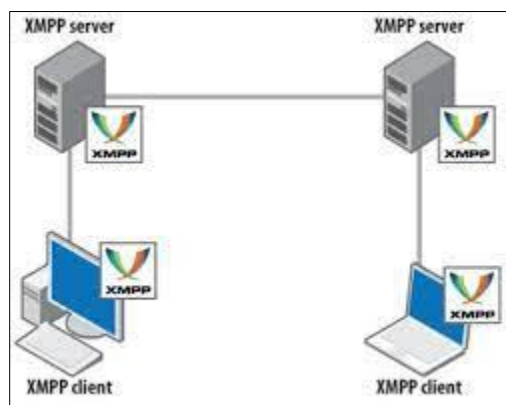
- Mensajería instantánea: esta aplicación utiliza tres servicios principales: Presencia, Contactos y Notificaciones uno a uno.
- Chat grupal: cree un grupo de usuarios con objetivos de comunicación comunes.
- Videojuegos: el uso de mensajes uno a uno y chat grupal aumenta la necesidad de poder jugar en línea.
- Métodos de control: se debe tener en cuenta la gestión de redes, parte científica y el manejo de robots para incluirlas en las diferentes áreas de aplicación.
- Geolocalización: permite que se dé una aplicación de acuerdo a la ubicación, como el seguimiento de vehículos (García, 2016).

#### **2.9.7. Arquitectura del protocolo XMPP**

XMPP utiliza una arquitectura servidor-cliente descentralizada, lo que significa que cada usuario puede crear su propio servidor y cliente para que el equipo también pueda dedicarse por completo a crear la mejor interfaz de usuario por XMPP. Varios equipos están trabajando para optimizar partes del servidor hasta el más mínimo detalle (García, 2016).

## Figura 7

### Arquitectura de XMPP



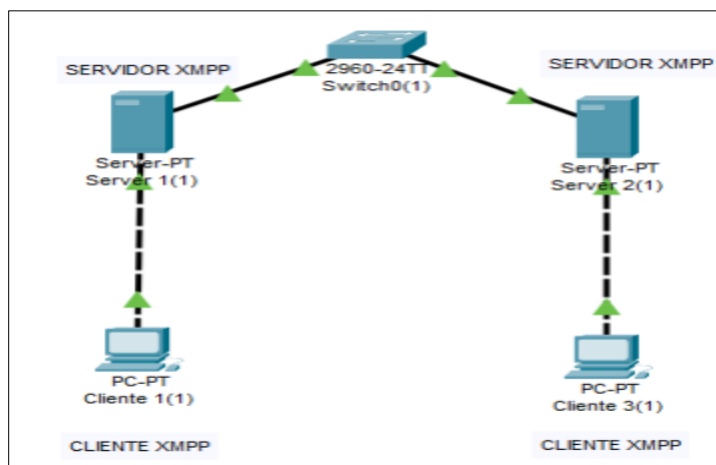
*Nota.* En la figura se visualiza el diagrama de la arquitectura de un protocolo XMPP.

Obtenida de: (Mendoza, 2021)

Para la comunicación entre clientes de diferentes servidores, considere la siguiente representación.

## Figura 8

### Comunicación entre clientes XMPP



*Nota.* En la figura se visualiza el esquema de la arquitectura de un protocolo de comunicación entre clientes XMPP. Obtenida de: (García, 2016)

## **2.10. Seguridad en las comunicaciones**

La confidencialidad de sus comunicaciones incluye protección. Durante esta protección. Consta de tres pilares básicos, a saber, las personas, los procesos y la tecnología.

Esta parte de la seguridad de las comunicaciones incluye el cifrado, el cifrado y otros métodos de administración de claves. Gracias a ellos, los datos están protegidos en cualquier aplicación o plataforma corporativa. Las comunicaciones seguras son una parte importante de la inversión actual. La ciberdefensa está diseñada para proteger activos críticos. Es la marca en sí, su capital intelectual y todo lo relacionado con sus clientes. (MPM, 2019).

## **2.11. Ingeniería en la seguridad de datos**

Este tipo de seguridad se trata de crear seguridad para proteger su red de amenazas. Estas protecciones se crean mediante el diseño de sistemas, diseños o arquitecturas específicos que contienen la información requerida. Se trata de incorporar medidas de seguridad para proteger su red de posibles amenazas.

Si bien esta técnica nos permite realizar ciertas actividades, las actividades relacionadas con la seguridad de las comunicaciones aseguran que no se produzcan otras actividades. En pocas palabras, es el diseño de sistemas seguros con una arquitectura y diseño específico para bloquear la red. Protegen datos, ordenadores y servidores. (MPM, 2019).

## **2.12. Encriptación**

El cifrado es responsable de proteger todos los datos o archivos almacenados o distribuidos en Internet. Esto hace que sea difícil acceder a ellos y solo se pueden leer si tiene la clave correcta. Esta herramienta debe ser parte de la propia red y de su

funcionamiento si queremos garantizar una comunicación segura (MPM, 2019).

### **2.13. Análisis de vulnerabilidad y control**

Los sistemas NIDS o de detección de intrusos se activan cuando se sospecha alguna acción. Esto se marca para una revisión posterior y la información de tráfico inusual se recopila automáticamente y se informa al administrador. Cuando se trata de escanear vulnerabilidades, los piratas informáticos buscan agujeros en el sistema a los que puedan acceder. El analista de seguridad de las comunicaciones se encarga de destapar estos espacios y cerrarlos (MPM, 2019).

## Capítulo III

### 3. Diseño

#### 3.1. Descripción de la arquitectura cliente-servidor

Básicamente, una arquitectura cliente-servidor consiste en un cliente que envía una solicitud a otro programa, llamado servidor de respuesta. Este servicio contendrá una idea adicional que se puede aplicar a los programas que se ejecutan en una sola computadora, con la ventaja de que un sistema operativo multiusuario se distribuye en una red informática. Dado que la interacción cliente-servidor es la base de la mayoría de las comunicaciones, es útil comprender la base sobre la que se construyen los algoritmos distribuidos.

##### 3.1.1. Partes de la arquitectura cliente-servidor

###### **Cliente**

En el caso de la arquitectura cliente-servidor, los clientes serán los dispositivos (teléfonos móviles, ordenadores, etc.) que podrán visualizar los datos obtenidos del colector o de los nodos IoT una vez ya procesados por el servidor y podrán cambiar remotamente la configuración de estos nodos. Con el fin de lograr que el software de incluya en la comunicación del nodo con el servidor, se debe proceder a la instalación del XMPP.

Estos clientes debe estar debidamente indentificados y no cualquier persona puede ingresar a esta aplicación.

###### **Servidor XMPP**

Este servidor tendrá una función principal, como es la de permitir el inicio de sesión en diferentes dispositivos y tener en cuenta sus credenciales, para lo cual se

deberá instalar una aplicación u programa que contenga el servidor tipo XMPP. La base de datos es la que esta formado por las capacidades que cuenta cada nodo de acuerdo a la red. Almacenamiento y gestión de la información recopilada por las personas.

### **Nodo IoT**

Tiene como función principal el recopilar y enviar la información, para luego actuar como clientes que pidan los datos de diferentes servidores u nodos para elaborar la configuración. Esto incluye la instalación de un cliente XMPP para comunicarse con los sensores de recopilación de datos.

### **Nodo IoT simple**

A diferencia de los nodos anteriores, los nodos simples podrán recopilar datos utilizando sensores instalados y deberán procesar la información y enviarla a través de XMPP. Son capaces de responder a las solicitudes y reconfigurar los sensores instalados en función de cómo estén programados.

### **Servidor de análisis**

El servidor debería poder procesar información de la base de datos, por lo que debería poder enviar advertencias sobre la muerte o si algo está fuera de servicio para que funcione correctamente. Necesita instalar un administrador de base de datos y un cliente XMPP para almacenar la información procesada. En este punto, se agregará el programa necesario para el análisis de datos.

## **3.2. Protocolo XMPP**

XMPP será una de esas funciones que hará que la mensajería instantánea (IM) entre usuarios sea lo más accesible y sencilla posible. Por lo tanto, tiene una

arquitectura Servidor-Cliente descentralizada, lo que significa que la comunicación entre clientes no es a través de un servidor central, sino que cada cliente puede crear su propio servidor XMPP y participar en la red, ayudando a que la conexión no se sature. y no hay punto de falla.

### **3.3. Descripción del servicio**

El servicio tendrá acceso a todos los usuarios cercanos, que necesiten utilizar el servicio se integrará con la plataforma, es decir, los usuarios que deseen utilizar las funciones de la plataforma deberán registrarse previamente en la aplicación.

#### **3.3.1. Instalación**

La aplicación contará con un servicio de descargar gratuita desde la plataforma, mismo que sera gratuito y para su ingreso se debera contar con las credenciales brindadas por servicio brindado por la plataforma militar.

#### **3.3.2. Registro**

El cliente será enviado a la persona registrada oficialmente para usar el servidor, es decir, el usuario asignado a la aplicación. Después de descargar la aplicación, el usuario debe ingresar a la aplicación e ingresar el usuario provisto y el número de identificación para ingresar. Una vez completado el registro, el usuario puede acceder a la aplicación y así se le mostrará una gama de herramientas, incluyendo acceso a la plataforma a través de la cual el usuario podrá acceder e interactuar con las funcionalidades de dicha plataforma.

#### **3.3.3. Comunicaciones**

Durante este proceso el usuario puede seleccionar la aplicación, ingresar y continuar usándola, se observará el mensaje de la pantalla de bienvenida, mientras que durante la carga se observarán automáticamente todos los componentes y



registros necesarios que estén conectados. Mensajería de voz y servicios. Después de un breve período de tiempo, la plataforma estará completamente cargada y lista para el trabajo del usuario. El servicio estará disponible y gratuito después de que el usuario haya pasado la seguridad.

#### **3.3.4. Agenda**

La aplicación deberá poseer una opción de agenda o llamada Bloc de notas para que los usuarios sigan creando contactos y agregando nombres para identificarlos. Además, los usuarios registrados en la aplicación podrán usarla como libreta para agregar texto o ver posibles eventos en la aplicación, categorizados por diferentes servicios como fecha o nombre.

#### **3.4. Funcionamiento**

La actividad de la aplicación se muestra en el estado de 3 servicios, llamada de servicio de chat o mensaje y su diseño le permite ejecutar desde un caso simple con pocos botones hasta un caso complejo con filas de cientos de nodos y docenas de servidores. Esto admite una cantidad específica de nodos, mantiene relaciones de capacidad para cada servidor y controla cómo se comparte la capacidad de memoria del servidor de análisis con otros contactos o registradores.

#### **3.5. Servicio de mensajería o Chat**

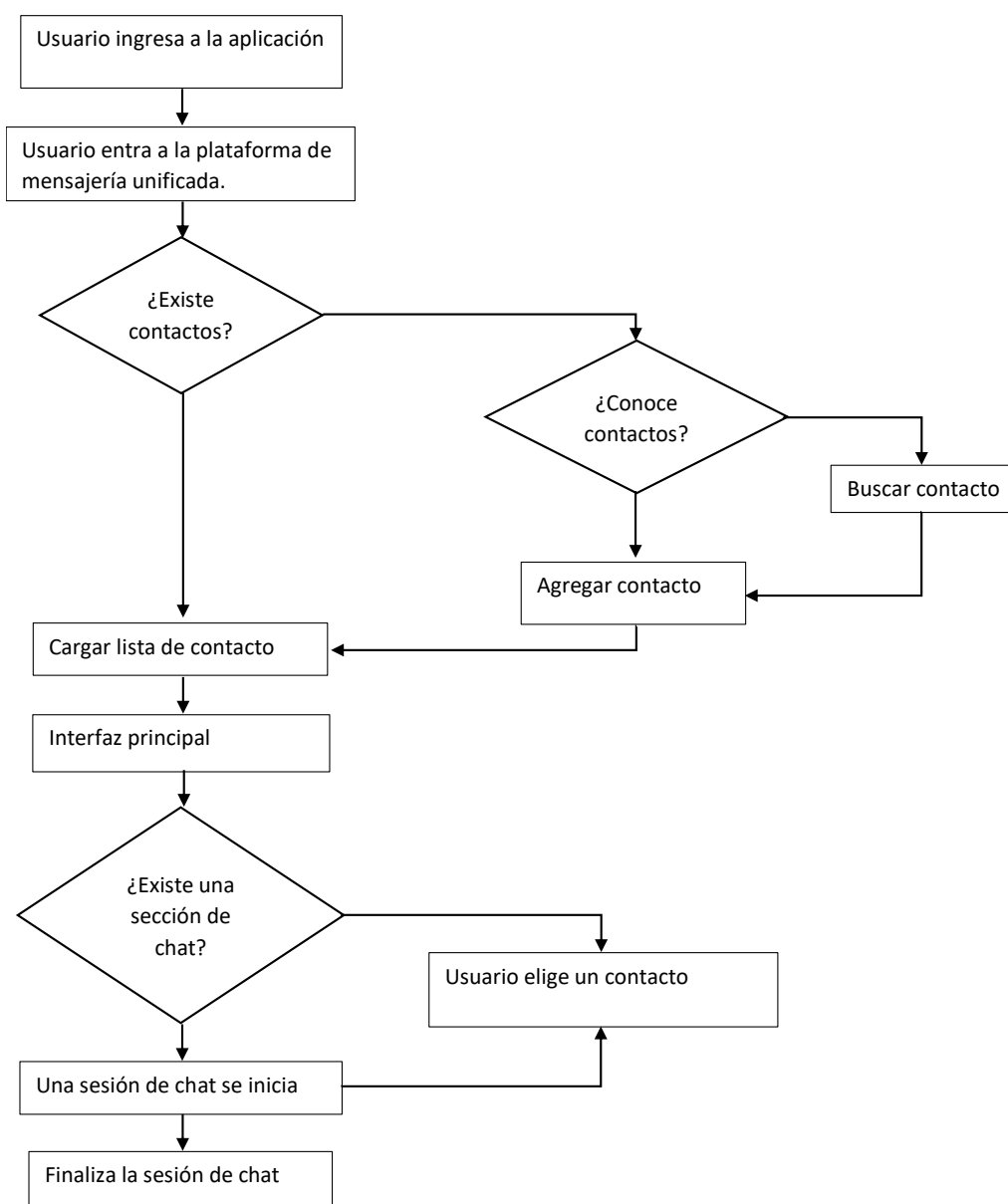
Dentro de la aplicación un servicio importante es el de mensajería o Chat, en la cual la aplicación cuenta con un módulo que carga los contactos de un usuario específico junto con información adicional de los mismos y se podrá interactuar con los demás contactos registrados. Este servicio permite interactuar con un servidor privado que a diferencia de las demás aplicaciones solo funciona para los usuarios que tengan la información o credenciales brindadas por las plataformas militares.

### 3.5.1. Diagrama de flujo del servicio de chat

En la figura 9 se observa un diagrama de flujo de mensajes que detalla las funciones básicas de esta herramienta: dónde poner aplicaciones primero y agregar contactos para iniciar una sesión de chat con alguien.

**Figura 9**

*Diagrama de flujo del servicio de chat*

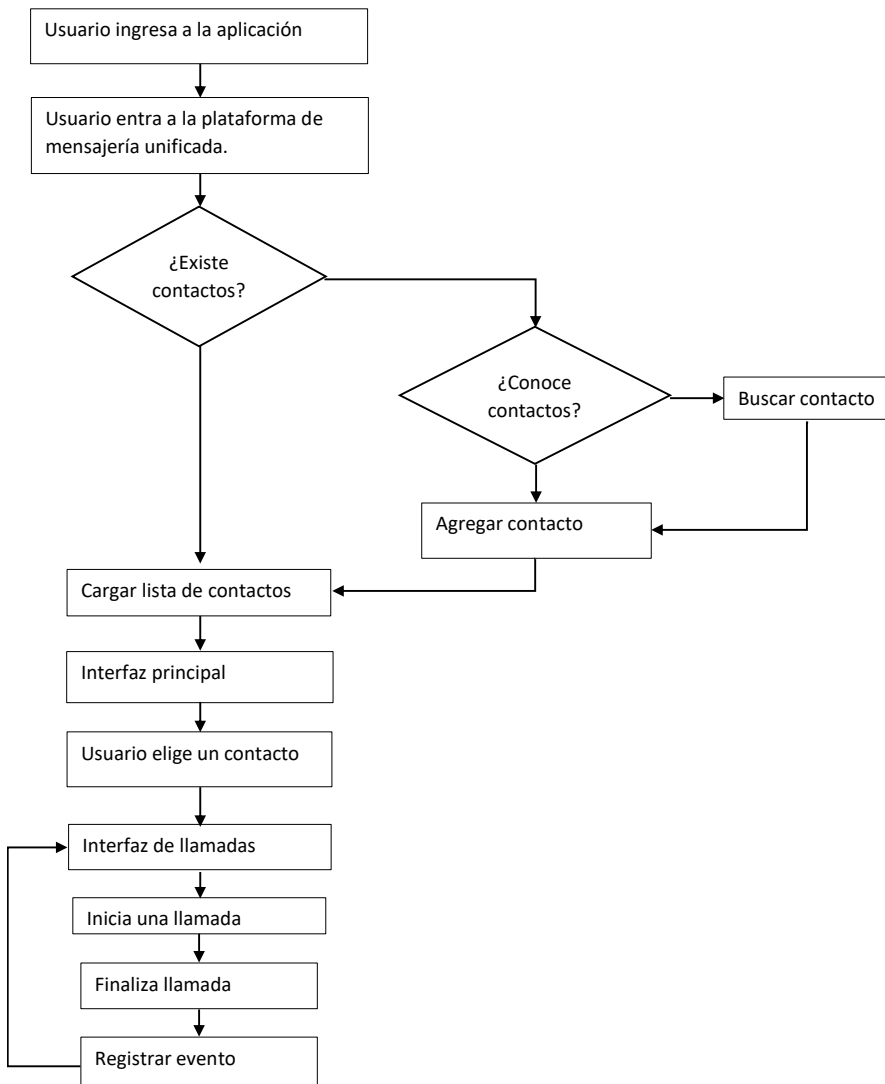


### 3.5.2. Diagrama de flujo del servicio de contactos

El diagrama de flujo muestra las funciones básicas de la sección Contactos de la aplicación: primero, de la lista de contactos disponibles, debe elegir una persona para llamar o enviar un mensaje de texto, y luego aparece una interfaz para la implementación. La llamada, cuando se complete, continuará registrando el evento y guardando el registro.

**Figura 10**

*Diagrama de flujo del servicio de contactos.*

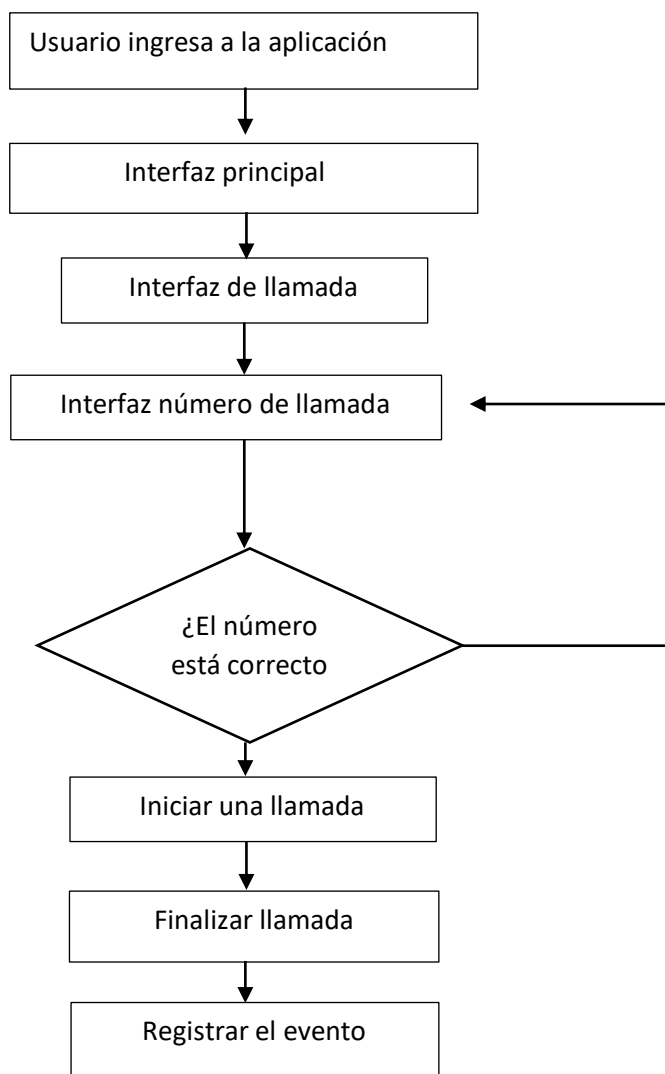


### 3.5.3. Diagrama de flujo del servicio de llamadas a contactos nuevos

En esta aplicación se debe manejar el sistema de llamadas al igual que el diagrama del módulo anterior, pero adicionalmente es necesario que el usuario ingrese un número o el contacto que contenga, el cual debe ser validado para que no se ingresen caracteres no numéricos, así se minimiza el riesgo de error en el ingreso del número y con esto pueda realizar el servicio de llamadas.

**Figura 11**

*Diagrama de flujo de llamadas a contactos nuevos.*



### 3.5.4. Diagrama de flujos del registro de eventos

Dentro de la aplicación se debe realizar un registro de los acontecimientos a los que llamaremos eventos: para estos se debe buscar la manera de aprovechar alguna característica de los servidores elegidos que pueden ser algún tipo de llamadas, mismos que nos permita efectuar el registro de eventos, detallando la hora de inicio, hora fin, la duración, el cliente encargado en iniciar el evento, el que lo recibió.

El servidor que se encuentra en la aplicación ya instalada debe registrar eventos en una base de datos, estos eventos son conocidos como registro detallado de llamadas. Por lo tanto, todo el flujo de llamadas que pase por este servidor será registrado en una base de datos para saber el estado de ese servicio.

#### Figura 12

*Diagrama de flujo del registro de eventos.*

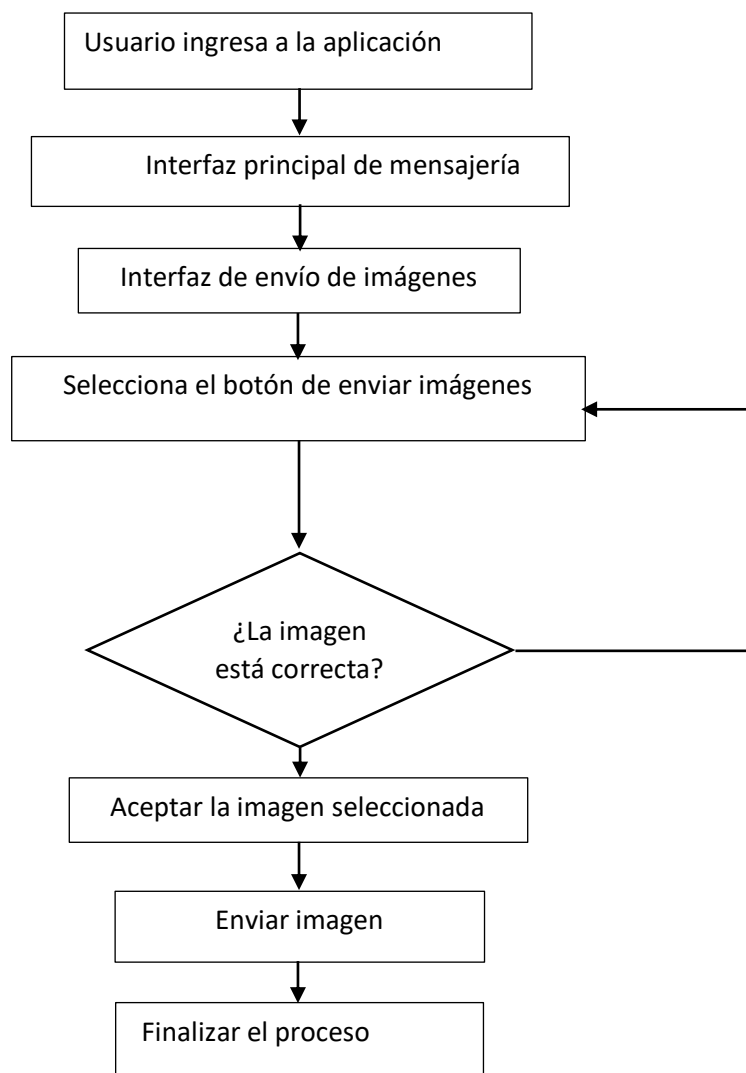


### 3.5.5. Diagrama de flujos de la transferencia de imágenes

Dentro de la aplicación se debe crear la facilidad y posibilidad de enviar y recibir imágenes por medio del servidor de mensajería implementado, esto con el fin de garantizar el sistema de mensajería y su correcto funcionamiento. Este procedimiento deberá llevarse a cabo en base a los códigos de programación que contendrá la aplicación.

#### Figura 13

Diagrama de flujo de la transferencia de imágenes.

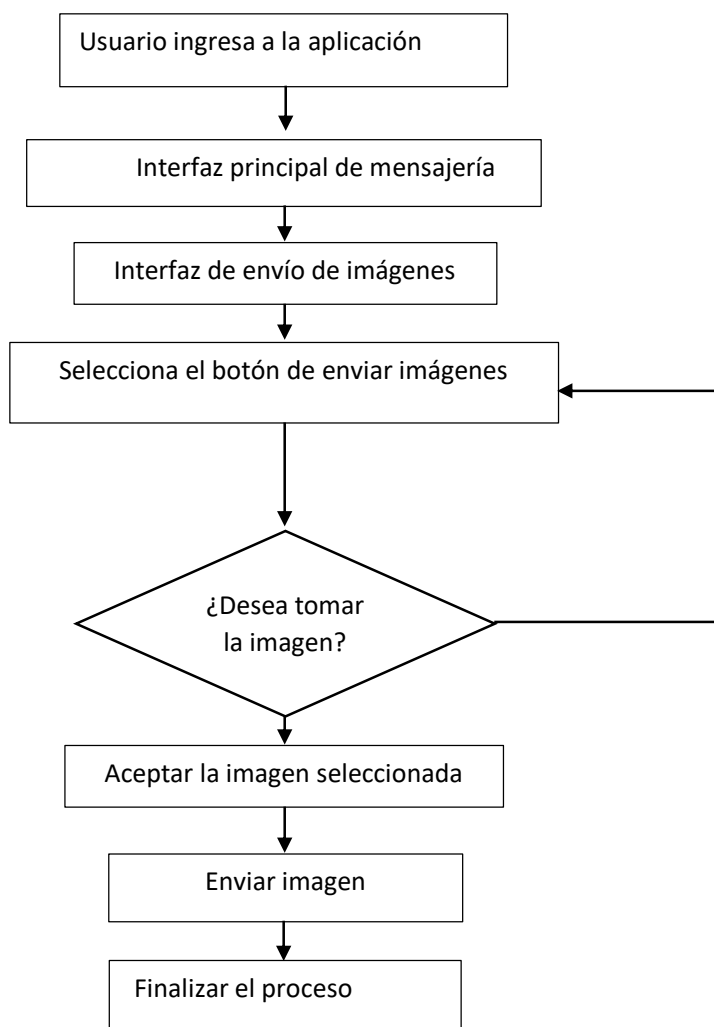


### 3.5.6. Diagrama de flujos de la toma de imágenes instantáneas

Dentro de la aplicación se debe crear la facilidad y posibilidad de enviar imágenes en tiempo real mediante el proceso de toma de fotografía por medio del servidor de mensajería implementado conjuntamente con la cámara del teléfono inteligente, esto con el fin de garantizar el sistema de mensajería y su correcto funcionamiento. Este procedimiento deberá llevarse a cabo en base a los códigos de programación que contendrá la aplicación.

**Figura 14**

*Diagrama de flujo de la toma de imágenes.*

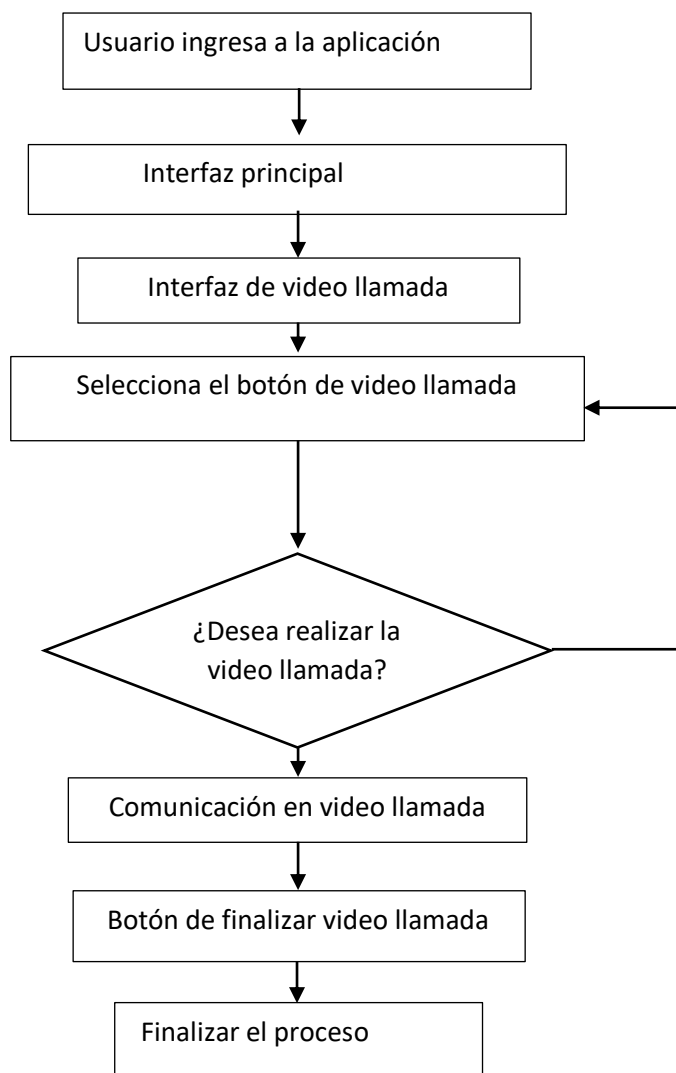


### 3.5.7. Diagrama de flujos de video llamada

En la aplicación se debe establecer la habilidad y capacidad para realizar una videollamada entre el usuario y los contactos registrados, y esta se debe realizar en tiempo real, a través del servidor de videollamadas y mediante la cámara del smartphone, con el fin de garantizar el sistema y su funcionamiento normal. Esta acción debe realizarse de acuerdo a los códigos de programación que contendrá la aplicación.

**Figura 15**

*Diagrama de flujo de la toma de imágenes.*





### **3.5.8. Estándares del sistema**

Dentro del servidor que se pretende crear se deberá ser objetivo y en lo posible seguir ciertos estándares para facilitar la comprensión del código desarrollado y así facilitar futuras implementaciones. Las interfaces deberán disponer de botones con el mismo tamaño de letras y un fondo que permita la visualización del servidor o la acción que se pretende realizar.

Crear los diferentes estándares:

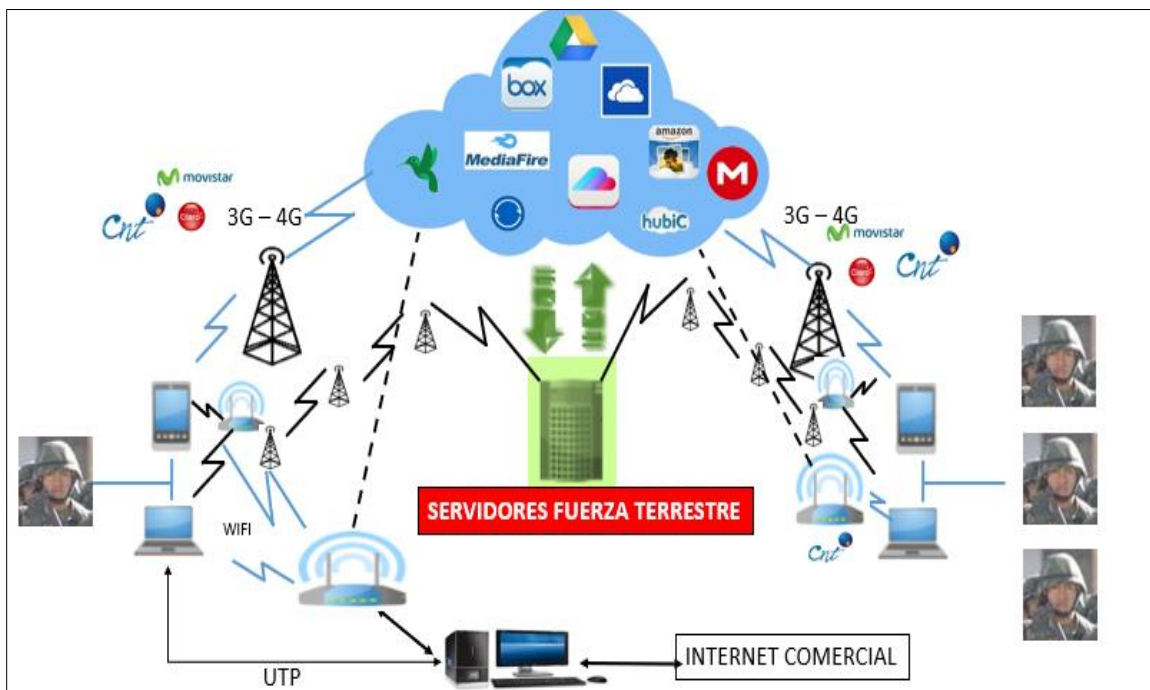
- a) Estándares de diseño de interfaces: En donde consten botones de interfaz digital, letras, fondos de pantallas y demás palabras que faciliten su utilización.
- b) Estándares de programación: Deberá contener las clases, métodos y variables con nombres acordes con las funcionalidades y valores que representan para así hacer fácil su utilización.
- c) Estándares usados en base de datos: Debe crearse una interfaz y programación de datos que permita los servicios de voz y chat. Este proceso dependerá de los lineamientos y códigos de fuente para la programación.

### **3.6. Propuesta del sistema de comunicación militar**

Como se hace renombre de forma anterior este diseño usará una red de baja complejidad porque es fácil de usar, pero tiene un sistema de protección de privacidad por lo que solo los usuarios definidos pueden usarlos, por lo que la topología seleccionada tendrá dos nodos de red, un cliente y un nodo libre y servidor XMPP. Los botones serán tabletas y los móviles de los clientes serán smartphones con sistema operativo Android.

**Figura 16**

*Propuesta del sistema de comunicación militar.*



Como se observa el servidor de mensajería contará con un internet comercial, computadores celulares y servidores de la fuerza terrestre misma que mediante el usuario y la asignación de seguridad permitirá garantizar la mensajería y la seguridad de este proceso.

### **3.6.1. Lista de componentes**

- ✓ Computadores
- ✓ Servicios de internet
- ✓ Teléfono inteligente Android
- ✓ Servicio de red
- ✓ Servidor XMPP

### **3.7. Fases del proceso**

Dentro de las fases del proceso se debe considera todos los servidores, software y operadores que se utilizarán en el diseño de esta arquitectura cliente – servidor de mensajería instantánea (XMPP), para la seguridad de las comunicaciones del agrupamiento de comunicaciones y guerra electrónica (Agrucomge).

#### **3.7.1. Establecer direcciones de XMPP**

Dado que XMPP se basa normalmente en el sistema de Servidor de nombres de dominio (DNS) para proporcionar la dirección a un usuario, en lugar de usar una dirección IP. Se debera utilizar un puerto del destino estándar para el proceso de recibir los mensajes XMPP, Este servidor debera estar compuesto por un nombre de usuario, un dominio y un recurso.

#### **3.7.2. Adquisición de un servidor con almacenamiento**

Dentro del proceso de adquisición de un servidor se deberá realizar un análisis de las características de los servidores y establecer un proceso de fase de adquisición:

##### **FASE I**

- a) Adquisición de un servidor con almacenamiento
- b) Uso de software libre para el servidor
- c) Uso de aplicativo libre para los teléfonos
- d) Configuración personalizada
- e) Uso de operadoras disponibles
- f) Uso de Intranet la F.T.
- g) Uso de Intranet al interior de las unidades

## **FASE II**

a) Desarrollo de un aplicativo propio de la F.T.

### **3.7.3. Elección de la plataforma del servicio de mensajería**

Para el análisis del desarrollo del aplicativo de la plataforma de mensajería que se pretende crear se debe tomar la decisión mediante el estudio de dos variables: la cantidad de dispositivos móviles que soportan la plataforma y el precio de adquisición de la programación para el servicio y el tipo de servidor.

Se debe contrastar el sistema Android con otros sistemas operativos para ver cuál es el índice de beneficio que esta se presenta con la implementación del servicio de mensajería propuesto.

### **3.7.4. Determinación de las características del software**

Dado que la aplicación pretende ser un servicio de mensajería y con el sistema de seguridad adecuada debe ser amigable con el usuario, para lo cual el software deberá enfocarse con las siguientes funcionalidades dentro de los servidores.

#### **Configurar un servidor de colaboración en tiempo real**

Este permitirá que se el sistema de mensajería y llamadas permita la colaboración en el tiempo real, donde al utilizar internet y la tecnología presentada por la aplicación para comunicarse con los demás usuarios permitan establecer los códigos adecuados para que se actualice a cada momento y permita enviar mensajes instantáneos y no se pierda el tiempo en el proceso de carga de la información.

#### **Utilizar protocolos XMPP**

El utilizar los protocolos XMPP (Extensible Message Protocol and Presence Communication) permite que la plataforma cree un sistema de intercambio de datos

XML que se puede utilizar en aplicaciones de mensajería instantánea, para garantizar que la información se transmita a los usuarios registrados. El protocolo también debe permitir detallar las características de adaptabilidad y sencillez.

### **Seguridad y rendimiento**

Dentro de este proceso de seguridad se debe enfocar en el soporte de mensajería del computador, la seguridad en los servicios de llamadas en tiempo real, el envío de imágenes, la recepción de imágenes, los proceso de video llamadas: este sistema debe estar programado y diseñado para soportar varios sistemas operativos tales como: Windows, Unix, Linux, Mac Os.

Además, este sistema debe garantizar el sistema de mensajería militar y el paso de la información como transferencia de imágenes, llamadas e información adicional que resulta de gran interés para el personal, en base a ello se detalla características que debe tener en cuenta este sistema.

- Cifrado de Datos (AES y BLOWFISH)
- Panel de administración web
- SSL/TLS (Protocolo de seguridad)
- Se puede ajustar según sea necesario
- Reunión
- Interactuar con MSN, Google Talk, Yahoo Messenger, AIM, ICQ, Jingle
- Estadísticas del servidor, paquetes y más.
- Clústeres con diferentes servidores
- Entrega de archivos
- Compresión de la información de los datos
- Tarjeta de avatar personal

## Capitulo IV

### 4. Desarrollo y resultados

#### 4.1. Diagrama de Conexión Cliente AGRUCOMGE

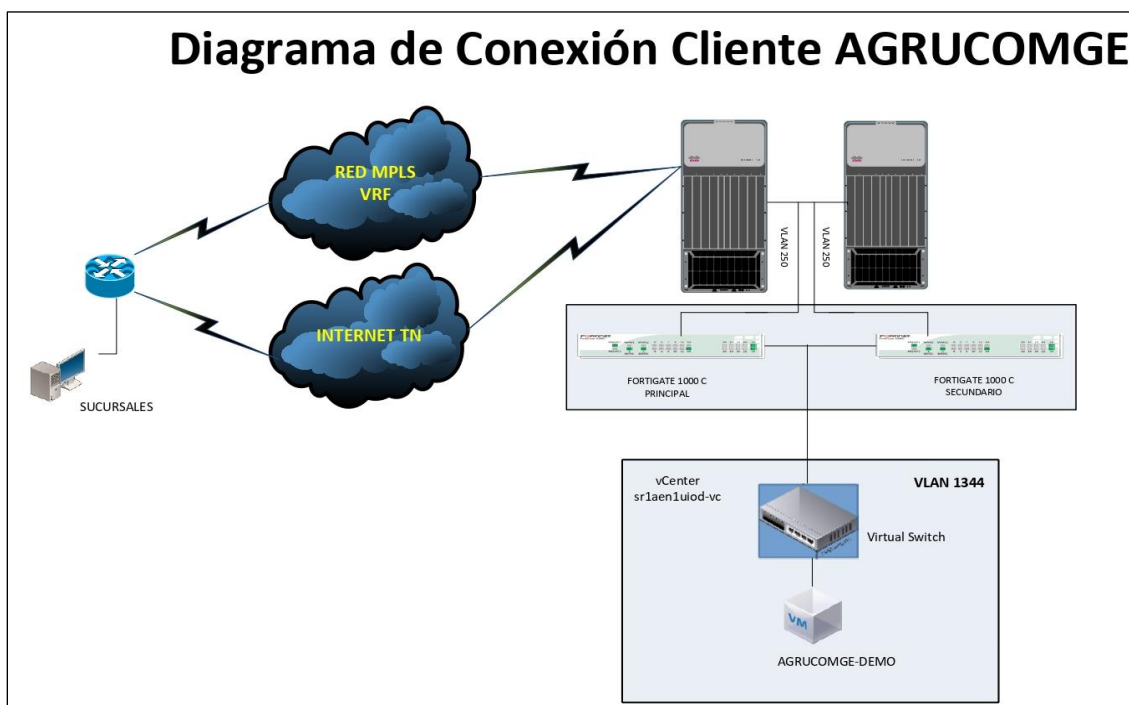
Se realiza el diagrama de conexión cliente AGRUCOMGE, cuenta con las rutas que tienen la comunicación entre los dispositivos, en donde se observa la arquitectura cliente servidor, con una comunicación con dos o más clientes que pueden estar activos al mismo tiempo utilizando una misma cuenta. (Anexo I)

Se tiene el extracto técnico con las siguientes características:

- ✓ **Login:** Agrucomge-datacenteruio
- ✓ **VLAN LAN:** 1334
- ✓ **Servidores contratados:** CLOUD IAAS, Internet

**Figura 17**

*Diagrama de Conexión del servidor*



### **Características de la VM**

- AGRUCOMGE-DEMO
- IP:192.168.1.2/24
- 4 Vcore
- 16 GB RAM
- Ubuntu 18.04
- Storage 200 GB

#### **4.2. Conexión a la infraestructura de virtualización - VMWARE**

Se accede a la infraestructura virtual, utilizando el enlace <https://181.198.26.10> , con las siguientes credenciales:

**Usuario:** agrucomge

**Contraseña:** \*\*\*\*\*

Para los usuarios que ingresan por primera vez se les solicita la descarga de un plugin de Fortinet, luego de instalarlos, les aparecen 3 botones, se selecciona Conectar, y de ahí se conecta a la VPN, que permite el acceso a la plataforma.

El plugin Forticlient puede descargarse del sitio oficial: <http://www.forticlient.com/>, una vez instalado se edita la conexión VPN.

**Figura 18**

*Editor de conexión VPN.*

**Editar Conexión VPN**

VPN VPN SSL VPN IPsec

Nombre de Conexión: agrucomge

Descripción:

Gateway Remoto: 181.198.26.10 ✕  
+Adicionar Gateway Remoto

Personalizar puerto: 443

Certificado de Cliente: Ninguno

Autenticación:  Preguntar en el login  Guardar login  
 No advertir de Certificado de Servidor Inválido

**Cancelar** **Guardar**

Una vez ingresado los datos y continuando con el proceso se presenta una pantalla con el ingreso del VPN, nombre de usuario y contraseña, se asigna los datos y se da clic en conectar.

**Figura 19**

*Datos de usuario VPN.*

Nombre de VPN: agrucomge

Nombre de Usuario: agrucomge

Contraseña: .....

**Conectar**



Luego de conectarse se muestra la IP asignada para la conexión a la infraestructura virtual.

### Figura 20

*Datos IP asignada.*

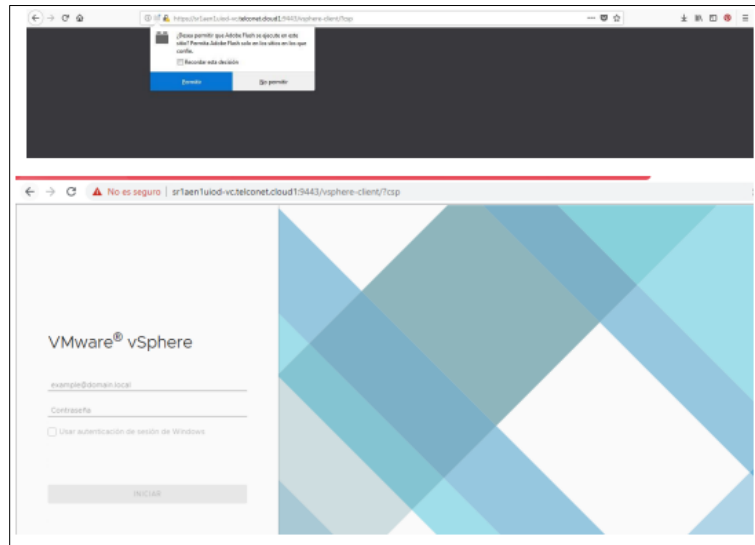


Una vez listo la conexión con la VPN, se abre el ambiente virtual, para conectarse de manera virtual se accede al siguiente enlace: <https://sr1aen1uiod-vc.telconet.cloud1:9443>

Antes de ingresar se instala el plugin que permite el ingreso a las consolas de las máquinas virtuales, se da clic en permitir y se abre la página VMware.

**Figura 21**

*Instalación de plugin.*



El usuario y la contraseña es la misma que para la VPN.

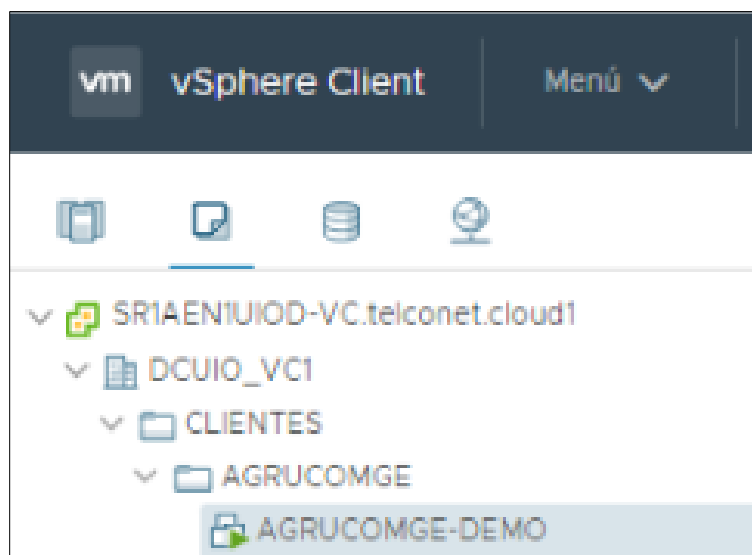
**Usuario:** agrucomge

**Contraseña:** \*\*\*\*\*

Una vez que se ingresa ya se puede navegar por el menú:

**Figura 22**

*Configuración de la VM.*



Si se desea ver un resumen de la VM se puede navegar en la pestaña

“Resumen”

### Figura 23

Resumen de la VM.

The screenshot shows the vSphere Client interface with the 'Resumen' tab selected for the VM 'AGRUCOMGE-DEMO'. The interface includes a search bar at the top, a navigation tree on the left, and a main content area with the following sections:

- Acciones:** Resumen, Supervisar, Configurar, Permisos, Almacenes de datos, Redes.
- Sistema operativo invitado:** Ubuntu Linux (64-bit)
- Compatibilidad:** ESXi 6.5 y posterior (máquina virtual versión 13)
- VMware Tools:** En ejecución, versión: 10304 (Administrado por invitado)
- Nombre DNS:** ubuntu
- Direcciones IP:** 192.168.12
- Host:** Ver las 2 direcciones IP
- Uso de recursos:**
  - USO DE CPU: 21 MHz
  - USO DE MEMORIA: 163 MB
  - USO DE ALMACENAMIENTO: 21,06 GB
- Hardware de máquina virtual:**
  - CPU: 4 CPU
  - Memoria: 16 GB, 0,16 GB memoria activa
  - Disco duro 1: 200 GB
  - Adaptador de red 1: (conectado)
  - Unidad de CD/DVD 1: Desconectado
- Notas:** Usuario: ubuntu, Editar notas...
- Atributos personalizados:**

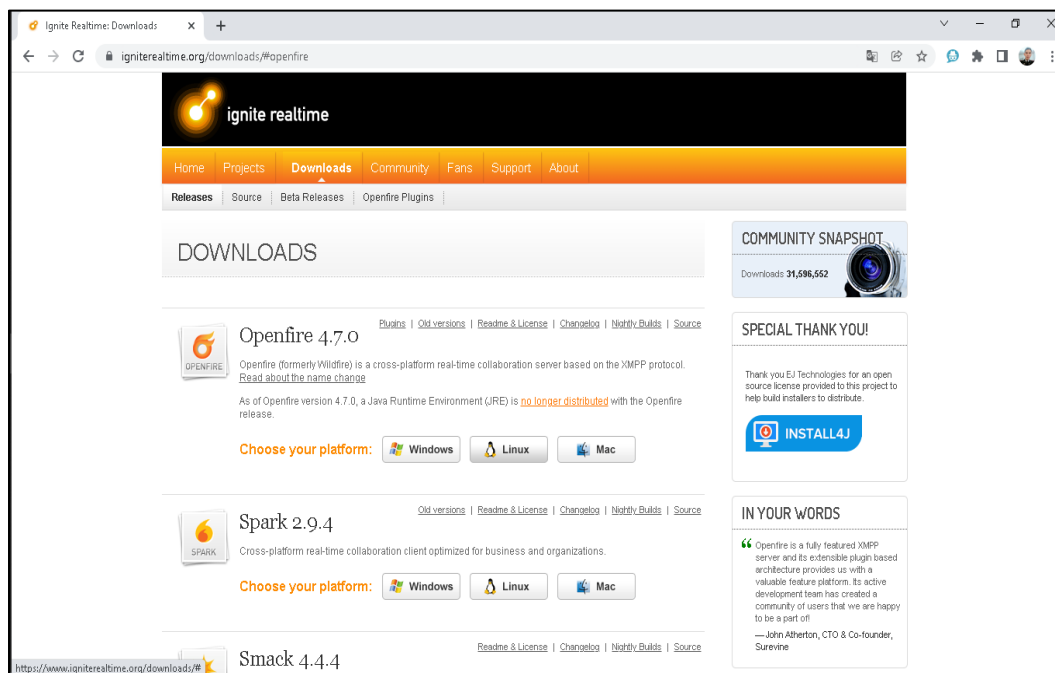
Atributo	Valor
FA.GosAgent	

### 4.3. Descarga del servidor

El software a utilizar es Openfire en Ubuntu 20.04 LTS, para su descarga ingresar al sitio <https://www.igniterealtime.org/downloads/>

**Figura 24**

*Descarga servidor.*



Se descarga el paquete que cuyo nombre es openfire 4.7.0.tar.gz, el cual es amigable con cualquier versión de Linux.

### 4.4. Pasos para instalar Openfire en Ubuntu 20.04 LTS

Una vez descargado el archivo se debe descomprimir (ver Anexo II).

a) Se extrae el contenido desde la ubicación.

```
~$ sudo tar xf openfire_4_6_2.tar.gz -C /opt/
```

b) Se configura el servidor del sistema mediante el enlace simbólico al binario principal.

```
~$ sudo ln -s /opt/openfire/bin/openfire /etc/init.d/
```

c) Se crea el scripts para el servidor

```
~$ sudo update-rc.d openfire defaults
```

d) Se lanza el servicio de manera manual.

```
~$ sudo systemctl start openfire
```

e) Se comprueba el servicio mediante el comando `systemctl status openfire`

## Figura 25

Comprobación del funcionamiento.

```

chat@chat: ~
File Edit View Search Terminal Help
chat@chat:~$ service openfire status
● openfire.service - LSB: Start/stop openfire jabber server
   Loaded: loaded (/etc/init.d/openfire; generated)
   Active: active (running) since Wed 2022-02-09 13:28:14 -05; 1min 53s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 1677 ExecStart=/etc/init.d/openfire start (code=exited, status=0/SUCCESS)
    Tasks: 264 (limit: 4915)
   CGroup: /system.slice/openfire.service
           └─1887 /usr/lib/jvm/java-11-openjdk-amd64/bin/java -server -Dopenfire
             └─2633 /usr/lib/jvm/java-11-openjdk-amd64/bin/java -Xmx1024m -XX:+Hea
               └─2652 /usr/lib/jvm/java-11-openjdk-amd64/bin/java -Xmx1024m -XX:+Hea

Warning: Journal has been rotated since unit was started. Log output is incomplete.
lines 1-12/12 (END)

```

Una vez que se comprueba que Openfire funciona de manera correcta, se procede a habilitarlo con lo cual arranca de forma automática conjuntamente con Ubuntu 20.04 LTS.

### 4.5. Instalador Web

De acuerdo al servidor utilizado, se emplea el siguiente enlace:

<http://ubuntu2004.local.lan:9090>

Se elige el idioma y se da clic en continuar.

**Figura 26**

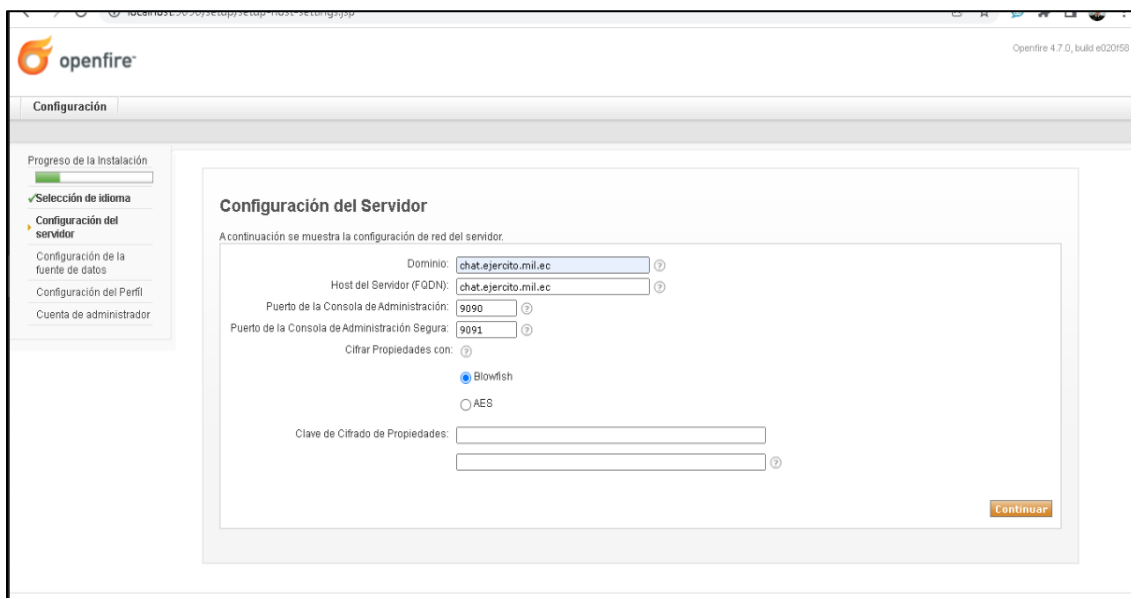
*Instalador del servidor, selección del idioma.*



Se configura la fuente de datos, en este caso se utiliza la conexión Estándar, debida a que ya se tiene una base de datos anterior y se continúa con el proceso.

**Figura 27**

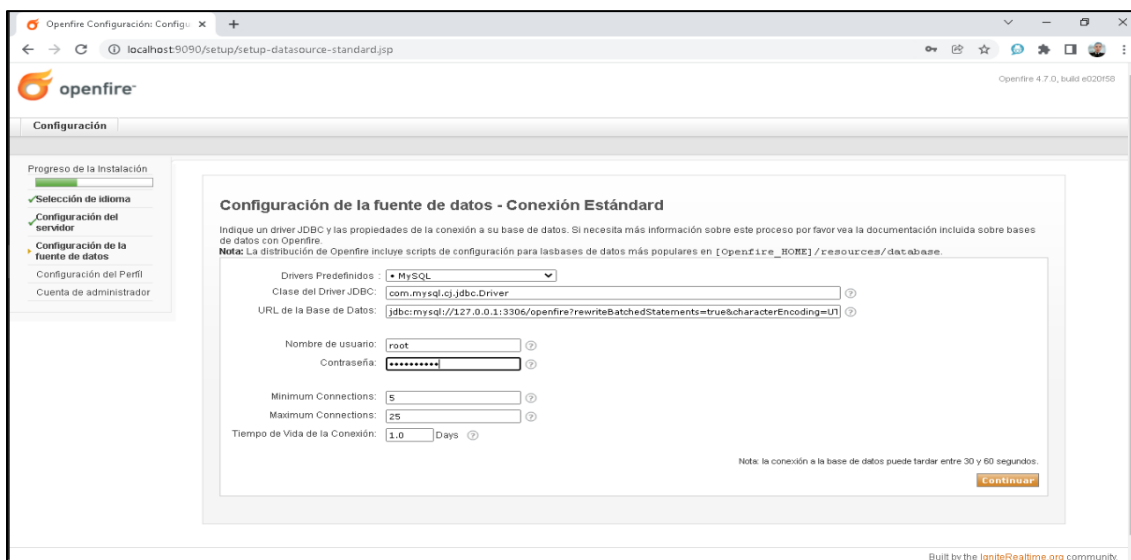
*Distribución del origen de datos.*



Se configura la fuente de datos de acuerdo a la conexión estándar, y se genera el usuario y contraseña.

## Figura 28

*Configuración de la fuente de datos- Conexión Estándar.*



Se continúa con la configuración del perfil

## Figura 29

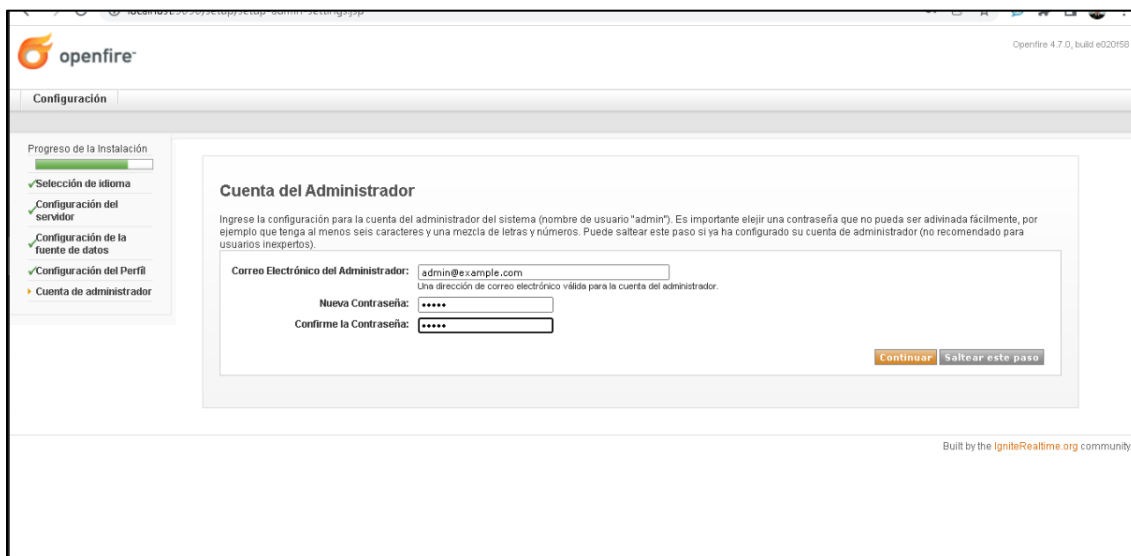
*Configuración del perfil.*



Se configura la cuenta del administrador.

**Figura 30**

*Cuenta del administrador.*

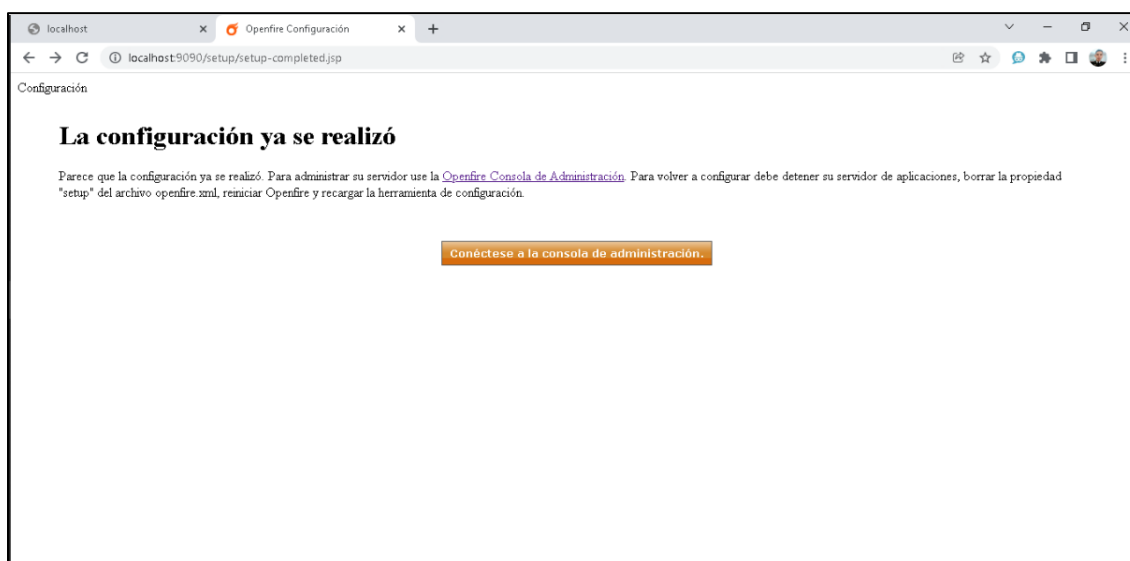


The screenshot shows the Openfire web interface during the installation configuration phase. The page title is "Cuenta del Administrador". On the left, a progress bar indicates the installation steps: "Selección de idioma", "Configuración del servidor", "Configuración de la fuente de datos", "Configuración del Perfil", and "Cuenta de administrador" (which is the current step). The main content area contains the following text: "Ingrese la configuración para la cuenta del administrador del sistema (nombre de usuario 'admin'). Es importante elegir una contraseña que no pueda ser adivinada fácilmente, por ejemplo que tenga al menos seis caracteres y una mezcla de letras y números. Puede saltar este paso si ya ha configurado su cuenta de administrador (no recomendado para usuarios inexpertos).". Below this text are three input fields: "Correo Electrónico del Administrador" (with the value "admin@example.com"), "Nueva Contraseña:" (with masked characters "\*\*\*\*\*"), and "Confirme la Contraseña:" (with masked characters "\*\*\*\*\*"). At the bottom right of the form area are two buttons: "Continuar" and "Saltar este paso". The footer of the page reads "Built by the IgniteRealtime.org community".

El proceso está finalizado.

**Figura 31**

*Finalización del proceso.*



The screenshot shows the Openfire web interface after the configuration process is complete. The page title is "Configuración". The main heading is "La configuración ya se realizó". Below this heading, the text reads: "Parece que la configuración ya se realizó. Para administrar su servidor use la [Openfire Consola de Administración](#). Para volver a configurar debe detener su servidor de aplicaciones, borrar la propiedad 'setup' del archivo openfire.xml, reiniciar Openfire y recargar la herramienta de configuración." At the bottom center of the page is a button labeled "Conéctese a la consola de administración." The browser's address bar shows the URL "localhost:9090/setup/setup-completed.jsp".

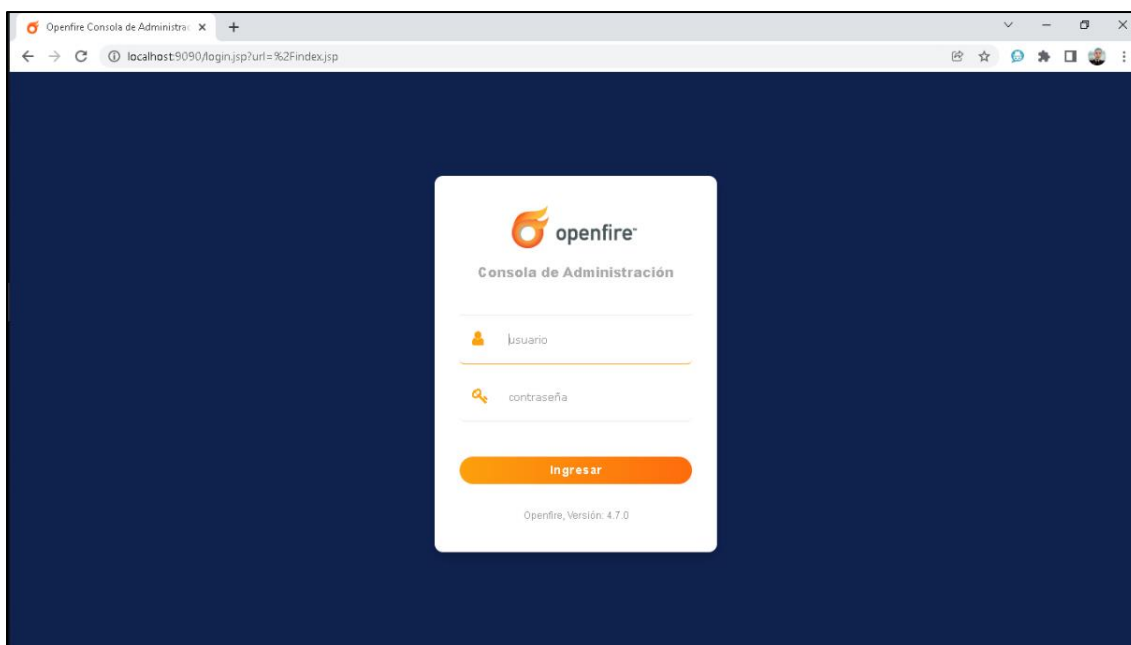


#### 4.6. Configuración Openfire en Ubuntu 20.04 LTS

Al finalizar el proceso de instalación, se conecta a la Consola de administración, en la que se hace el ingreso del usuario con la respectiva contraseña y de esta manera se inicia la sesión.

#### Figura 32

*Instalación de la consola de administración.*



Una vez ingresado, se observa los usuarios del servicio y el estado actual en la que se encuentra.

Figura 33

Configuración del servidor de la consola de administración.

Configuración del Servidor

A continuación están las propiedades de este servidor. Presione en el botón "Editar Propiedades" para cambiar algunas de las propiedades del servidor. Algunas configuraciones no pueden ser cambiadas.

**Propiedades del Servidor**

Tiempo de Actividad del Servidor: 3 minutos -- started 9 feb. 2022 13:28:32  
 Versión: Openfire 4.7.0  
 Ruta al servidor: /usr/share/openfire  
 Nombre del Servidor: chat.ejercito.mil.ec

**Ambiente**

Versión de Java: 11.0.13 Ubuntu -- OpenJDK 64-Bit Server VM  
 Servidor de Aplicaciones: jetty9 4.43.v20210629  
 Nombre del Host: chat.ejercito.mil.ec  
 SO / Hardware: Linux / amd64  
 Idioma / Huso Horario: es / hora de Ecuador (-5 GMT)  
 Dueño del proceso del SO: openfire  
 Memoria de Java 130,67 MB of 4012,00 MB (3,3%) used

**Puertos del Servidor**

Interfaz	Puerto	Tipo	Descripción
Todas direcciones	5222	Cliente-Servidor	El puerto estandar utilizado por clientes para conectarse al servidor. En este puerto se establecen conexiones de texto plano las cuales, dependiendo de los <b>parámetros de seguridad</b> configurables, pueden (o deben) ser mejoradas a conexiones cifradas.

**Noticias de Ignite Realtime**

- Openfire 4.7.0 has been released!, 19 ene. 2022
- Openfire 4.5.6 is released!, 5 ene. 2022
- Restored Openfire nightly builds, 5 ene. 2022
- Openfire 4.6.7 released (Log4j 2.17.1 only change), 3 ene. 2022
- Openfire 4.6.6 and 4.5.5 releases (Log4j-only changes), 16 dic. 2021
- Openfire 4.6.5 released, 10 dic. 2021
- Openfire 4.7.0 beta & Hazelcast plugin 2.6.0 releases!, 6 dic. 2021

Se observa la lista de usuarios, y los usuarios ya pueden registrarse desde clientes instalados en otros dispositivos.

Figura 34

Revisión de los usuarios.

Lista de Usuarios

Total de Usuarios: 4.527 -- Ordenados por Nombre de Usuario -- Usuarios por página: 25

Conectado	Usuario	Nombre	Grupos	Creado	Última Salida	Editar	Borrar
1	aaaltamiranop	Subt. Altamirano Anthony	None	20 may. 2021	Nunca se conectó antes.		
2	aaarellanao	Tnte. Arellano Adner	None	13 may. 2021	Nunca se conectó antes.		
3	aaarobof	Tnte. Arrobo Andrea	None	13 may. 2021	Nunca se conectó antes.		
4	aaastidasp		None	13 may. 2021	Nunca se conectó antes.		
5	aacalderong	Subt. Calderon Andres	None	13 may. 2021	Nunca se conectó antes.		
6	aacamachoz		None	13 may. 2021	Nunca se conectó antes.		
7	aacareral	Mayo. Carrera Alberto	None	13 may. 2021	Nunca se conectó antes.		
8	aacofreb	Subt. Cofe Anthony	None	13 may. 2021	Nunca se conectó antes.		
9	aafeirep	Tnte. Freire Andreina	None	13 may. 2021	Nunca se conectó antes.		
10	aaaguerreros	Subt. Guerrero Andres	None	13 may. 2021	Nunca se conectó antes.		
11	aaajimboj	Tnte. Jimbo Andrea	None	13 may. 2021	Nunca se conectó antes.		
12	aaalandetav	Mayo. Landeta Alex	None	13 may. 2021	Nunca se conectó antes.		
13	aalopezr	Capt. López Angel	None	13 may. 2021	Nunca se conectó antes.		
14	aanavasc	Tcrn. Navas Angel	None	13 may. 2021	Nunca se conectó antes.		

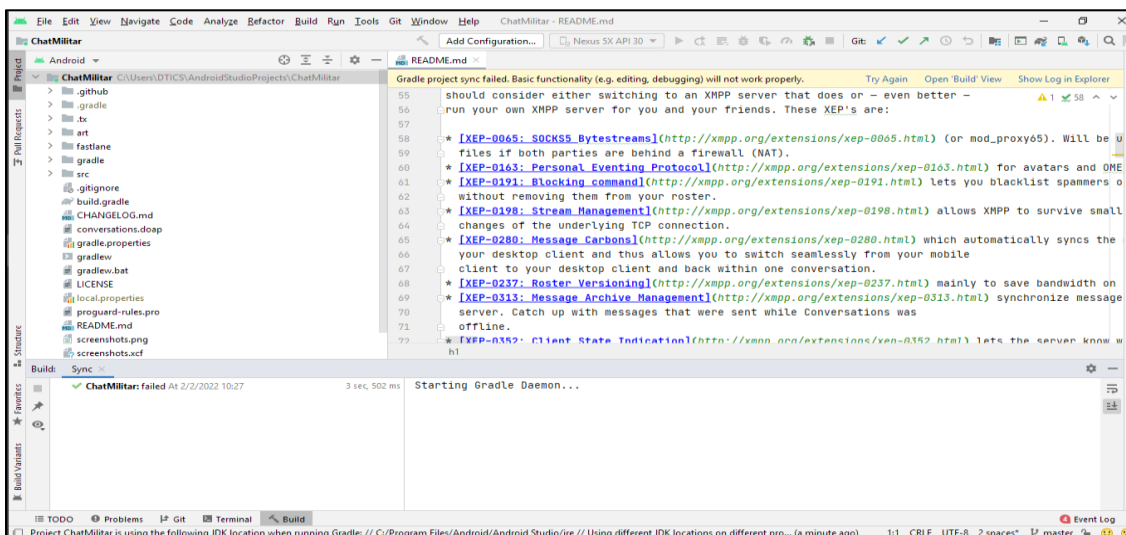
#### 4.7. Desarrollo de la Apk para dispositivos Android

El desarrollo del APK se realizó como repositorio principal <https://github.com/> y entorno de desarrollo Android Studio, el cual permitió generar el APK y su posterior difusión al personal militar del AGRUCOMGE mediante la página web [www.chat.ejercito.mil.ec](http://www.chat.ejercito.mil.ec).

Posteriormente se hace énfasis en el código fuente generada:

#### Figura 35

*Código Fuente.*



Para generar y compilar el APK denominado ChatMilitar, es necesario usar los siguientes comandos:

- Descargue e instale el SDK de Android
- Instale el Repositorio de Google y las herramientas SDK más recientes con Android SDK Manager
- Verificar el código fuente utilizado  
git clone <https://github.com/dannyrms1992/ChatMilitar>
- Compilar con: `./gradlew assemble ChatMilitar FreeSystemDebug`

- La Apk generada se encontrará en el directorio: build/outputs/apk/ ChatMilitar FreeSystem/debug/ ChatMilitar -\$version- ChatMilitar -free-system-debug.apk

Una vez que tengamos el APK debemos realizar la instalación como se muestra en el manual mostrado en el Anexo V.

#### 4.8. Página web para la difusión del aplicativo

Como parte adicional se vio la necesidad de desarrollar una página web alojada en el mismo servidor, basado en nginx para la difusión del aplicativo al personal militar, en donde se encontrará los instaladores, manuales de instalación y más información relevante y de interés.

#### Figura 36

*Página web Aplicativo Chat Militar.*



## Conclusiones

Se estableció que la arquitectura cliente-servidor permite dar un puente de diálogo entre una Pc que interactúa con un Smartphone u otro dispositivo, y entre las características de esta arquitectura se tiene la combinación entre el usuario, cliente, y el servidor, los requerimientos de las tareas como la velocidad del procesador, la memoria del disco, y los servidores proxy cuyo objetivos principales radican en mejorar el rendimiento del resultado en menor tiempo y los filtros de las solicitudes del nuevo usuario.

Se utilizó el método deductivo y la investigación exploratoria para analizar la información de las fuentes de referencia con respecto a la arquitectura de comunicación, en la que se cuenta como partes los clientes y el servidor XMPP, la descripción del servicio, instalación, registro, comunicaciones, agenda, funcionamiento, y los servicios de mensajería.

Se concluye que el funcionamiento de la arquitectura cliente servidor permite tener un alto porcentaje seguridad de la información del Agrupamiento de Comunicaciones y Guerra Electrónica AGRUCOMGE, esto debido a que posee un sistema de gran escala para los distintos escenarios y seguridad frente a los inconvenientes electrónicos descentralizado y accesible para los clientes.

## **Recomendaciones**

Para el desarrollo de la arquitectura primero se debe conocer los elementos que conforman dicho sistema y las necesidades cliente servidor para mejorar sus rendimientos y la seguridad de la información.

Tomar en cuenta los requerimientos para la instalación del Openfire en Ubuntu 20.04 LTS, esto con el fin de que el sistema funcione de mejor manera.

Considerar los requisitos mínimos para la instalación y configuración del aplicativo Chat – Militar en usuarios, como el Sistema operativo Android 4.0.3, el espacio del almacenamiento mayor a 15 MB y la conexión estable a internet si restricciones.

## Bibliografía

- Alsina, G. (Noviembre de 2019). *Significado de cliente servidor*. Obtenido de <https://significado.com/cliente-servidor/>
- Araujo, E. (2018). *Implementación de un sistema de videovigilancia para los exteriores de la UPS mediante minicomputadoras y cámaras Raspberry Pi*. Guayaquil: Primera.
- Becerra, E. (2016). *Implementación de monitoreo de red utilizando los protocolos ICMP y SNMP*. La libertad: Primera.
- Carate, B., & Pozo, D. (2019). *DISEÑO DE UN SISTEMA DE DETECCIÓN DE INTRUSOS (NIDS) PARA UNA RED SIMULADA PYMES EN GNS3, IMPLEMENTADA EN UN MÓDULO RASPBERRY PI PORTÁTIL*. UNIVERSIDAD POLITÉCNICA SALESIANA. Obtenido de <https://dspace.ups.edu.ec/bitstream/123456789/17546/1/UPS%20-%20ST004141.pdf>
- Chang, D. (2018). *Desarrollo e implementación de un sistema para el control e inventario continuo, utilizando tecnología RFID, para la biblioteca de la UPS sede Guayaquil*. Guayaquil: Primera.
- Chávez, G. (2016). *Propuesta de red de datos para la gestión de los servicios de red en el campus politécnico de la ESPAM MFL*. Calceta: Primera.
- DocuSign. (2021). Tipos de servidores que debes conocer. *DocuSign*, 4-10.
- García, G. (2016). *Sistema Computacional de apoyo al docente para cumplir con*. Juárez: Primera.

López, X. (2018). *Rediseño de la red con calidad de servicios para datos y tecnología de voz sobre ip en el ilustre municipio de ambato*. Ambato: Primera.

Machado, A. (8 de Febrero de 2016). *Naps Tecnología y educación*. Obtenido de <https://naps.com.mx/blog/funciones-de-un-administrador-de-redes/>

Manuel, S. O. (16 de Septiembre de 2016). *ITD*. Obtenido de <https://sites.google.com/a/itdurango.edu.mx/10040372/system/app/pages/sistema-p/hierarchy>

Mendoza, H. (2021). *Análisis del desempeño del protocolo de comunicación XMPP en una Red IOT*. Sangolqui: Primera.

MPM. (25 de Junio de 2019). *La seguridad en las comunicaciones*. Obtenido de <https://www.mpmsoftware.com/es/blog/seguridad-en-las-comunicaciones/>

Ortega, A. (2019). *Diseño de un sistema de control de acceso y video vigilancia para la unidad educativa porvenir con la utilización de dispositivos IP*. Cuenca: Primera.

Robles, F. J. (2018). *Planificación y Administración de Redes (GRADO SUP.)*. España: RA-MA.

rosa, I. L. (22 de Octubre de 2021). *pandorafms*. Obtenido de <https://pandorafms.com/blog/es/protocolos-de-administracion-de-redes/>

Salas, J. S. (2020). *DISEÑO DE UNA HERRAMIENTA DE MONITOREO Y CONTROL DE SERVIDORES UTILIZANDO COMO EJE PRINCIPAL CACTI. APLICADO A UNA PYME MEDIANA*. Bogota: UNIVERSIDAD COOPERATIVA DE COLOMBIA.



- SANTIAGO, A. C. (2016). *DESARROLLO DE UN SISTEMA INALAMBRICO BASADO EEG PARA EL MONITOREO DEL SUEÑO EN UN CONDUCTOR*. CUENCA.
- Spec, G. (2017). Tipos de control de acceso. 5-60.
- Sri, Y. (2018). *capterra*. Obtenido de <https://www.capterra.ec/software/135902/zabbix-monitoring-solution>
- UNAN. (2017). *UNAN*. Obtenido de [https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/931/mod\\_resource/content/4/contenido/index.html](https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/931/mod_resource/content/4/contenido/index.html)
- Valarezo Saldarriaga, G. G., & Simisterra Huila, J. C. (2018). *Implementación de un sistema de gestión y administración de redes basados en el protocolo simple de monitoreo de redes SNMP en la red ESPOL-FIEC*. Guayaquil: ESPOL.
- Valdés, B. (Enero de 2021). *Administración de redes*. Obtenido de <https://www.administracionderedes.com>
- Vega, G. (2018). *IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO PARA EL ANÁLISIS DE LA DISPONIBILIDAD, CAPACIDAD, CALIDAD Y LATENCIA DE ENLACES CORPORATIVOS DE ÚLTIMA MILLA*. Obtenido de <http://repositorio.ucsg.edu.ec/bitstream/3317/11890/1/T-UCSG-POS-MTEL-118.pdf>
- Velásquez, M. (2021). *Análisis del desempeño del Protocolo de Comunicación XMPP en una Red IOT*. Ambato: <http://repositorio.espe.edu.ec/jsui/bitstream/21000/23416/1/T-ESPE-044197.pdf>.

Vinicio, P. L. (08 de 2018). *repositorio.uta.edu.ec*. Obtenido de

[https://repositorio.uta.edu.ec/bitstream/123456789/28577/1/Tesis\\_%20t1465ec.p](https://repositorio.uta.edu.ec/bitstream/123456789/28577/1/Tesis_%20t1465ec.pdf)

[df](https://repositorio.uta.edu.ec/bitstream/123456789/28577/1/Tesis_%20t1465ec.pdf)

**Anexos**