**Introducción**

Desde tiempos remotos, el hombre ha venido utilizando distintas herramientas para poder aumentar el alcance de su capacidad de manipulación. En un principio, no se trataba más que de palos utilizados para hacer caer la fruta madura de un árbol. En la actualidad existen dispositivos más complejos, como por ejemplo pértigas que, accionando un enganche o pinza a distancia, permiten alcanzar objetos que se encuentran en estanterías o lugares de almacenamiento de difícil acceso.

En los sistemas de teleoperación de robots la intervención del operador humano muchas veces es imprescindible, especialmente en entornos no estructurados y dinámicos en los cuales los problemas de percepción y planificación automática son muy complejos.

En muchos casos, el operador esta físicamente separado del robot, existiendo un sistema de telecomunicaciones entre los dispositivos que utiliza directamente el operador y el sistema de control local del robot.

La comunicación a través de internet puede ser tanto en una vía como en esquemas bilaterales con realimentación, (ortega2004) para esto se han planteado algunas propuestas.

Estas propuestas han tenido gran acogida al resolver problemas de seguridad en la estabilidad de la comunicación, evitando así desconexiones entre los hosts desconexiones hacia el canal de comunicación, problemas en los servicios iniciados en las terminales, errores del sistema base de las aplicaciones, desconexión o a su vez suspensióón de los servicios, la solución mas aceptada es en la utilización de webservice a través de request http polling y administración por DOM, para mantener la consistencia de los valores se ha aplicado formatos en XML.

Aunque muchos de los problemas se han resulto con este esquema de comunicación, aun se ven grandes inconvenientes con el tiempo que toma un paquete llegar desde su host origen a su host destino, ya que los protocolos y modelo de interacción cliente servidor no fueron creados para conexiones en tiempo real y con opciones de comunicación full duplex en los cuales tanto los clientes como los servidores pueden iniciar una comunicación de forma autónoma.

Para simplificar el proceso de envío y recepción de información manteniendo conexiones consistentes y el manejo de datos en paquetes estructurados simplificados para evitar la pérdida de información o generación de basura en memoria se presenta la implementación de un protocolo de comunicación websocket y el uso de paquetes Json para el manejo de comunicación de información con el diseño de el servicio web de manejo de flujo de datos y los respectivos clientes de manejo de acceso al canal.

**Metodología de Desarrollo.**

Antes de realizar cualquier proyecto de software es necesario realizar una planificación de cómo se van a cumplir correctamente los objetivos planteados.

El autor Goncalves plantea que la experiencia ha demostrado que los proyectos exitosos son aquellos que son administrados siguiendo una serie de procesos que permiten organizar y luego controlar el proyecto, considerando válido destacar que aquellos procesos que no sigan estos lineamientos corren un alto riesgo de fracasar. Es necesario destacar la importancia de los métodos, pero el éxito del proyecto depende más de la comunicación efectiva con los interesados, el manejo de las expectativas y las personas que participan en el proyecto.

Existen diferentes modelos y metodologías que han sido en los últimos años herramientas de apoyo para el desarrollo del software. El autor Someerville, menciona que:

Modelo de desarrollo de software: es una representación simplificada del proceso para el desarrollo de software, presentada desde una perspectiva específica.

Entre las metodologías más utilizadas y las que se analizan estan:

**Programación Extrema o XP (Extreme Programing)**

Es una metodología ágil para el desarrollo de software y consiste básicamente en ajustarse estrictamente a una serie de reglas que se centran en las necesidades del cliente para lograr un producto de buena calidad en poco tiempo. La Programación Extrema es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software. Promueve el trabajo en equipo, preocupándose en todo momento del aprendizaje de los desarrolladores y estableciendo un buen clima de trabajo. Este tipo de método se basa en una realimentación continuada entre el cliente y el equipo de desarrollo con una comunicación fluida entre todos los participantes, también busca simplificar las soluciones implementadas para los múltiples cambios. Este tipo de programación es la adecuada para los proyectos con requisitos imprecisos, muy cambiantes y con un riesgo técnico excesivo.

**Scrum**

Scrum es un proceso en el que se aplican de manera regular un conjunto de mejores prácticas para trabajar en equipo y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. Scrum, más que una metodología de desarrollo software, es una forma de auto-gestión de los equipos de programadores. Un grupo de programadores deciden cómo hacer sus tareas y cuánto van a tardar en ello. Ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro. Además, permite además seguir de forma clara el avance de las tareas a realizar, de forma que los "jefes" puedan ver día a día cómo progresa el trabajo. “También se realizan entregas parciales y regulares del resultado final del proyecto, priorizadas por el beneficio que aportan al receptor del proyecto.

En la presente investigación se utiliza Scrum, que al ser una metodología versátil permite adecuarse perfectamente en el trabajo que se esta desarrollando ya que permite trabajar colaborativamente, en equipo y asi obtener el mejor resultado posible de un proyecto.

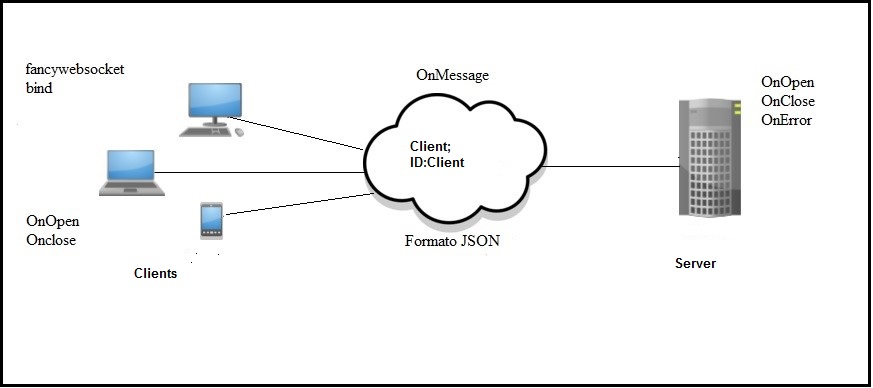
**Proceso de Tele Operación**

**1. Identificación de Clientes y Re direccionamiento**

En el proceso de tele operación es necesario definir el origen y destino de los datos que se envian entre los clientes y el servidor. Para esto es necesario que cada cliente se identifique en el servidor utilizando un formato de datos especifico que comprenda el servidor, esto es necesario para que el servidor pueda enviar correctamente los datos correspondientes a cada cliente.

En la identificación de los clientes, el formato de datos utilizado para que sea interpretado tanto por el servidor como por el cliente es Json, que es un formato ligero de intercambio de datos, está basado en el lenguaje de programación JavaScript, Json es un formato de texto que es completamente independiente del lenguaje de programación que se esté utilizando.

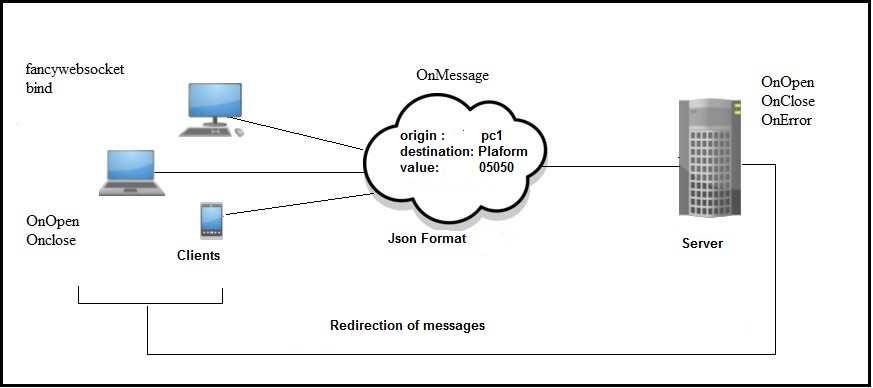
El cliente se identifica en el servidor de la siguiente manera: {“cliente”: “NombreCliente”} de esta manera el servidor conoce que cliente se acaba de conectar y queda almacenado en memoria.



En el Re direccionamiento es necesario que se identifique que cliente es el origen y que cliente es el destino de dicho paquete, ya que en el servidor se envía los datos en base a la identificación del cliente y al origen y destino del paquete.

El paquete es enviado por parte de los clientes al servidor de la siguiente forma:

{“origen”: “NombreOrigen”, “destino”: “NombreOrigen”, “mensaje”: “Mensaje”}



Esto permite tener el control de los datos que envian y reciben los clientes que esten conectados con el servidor.

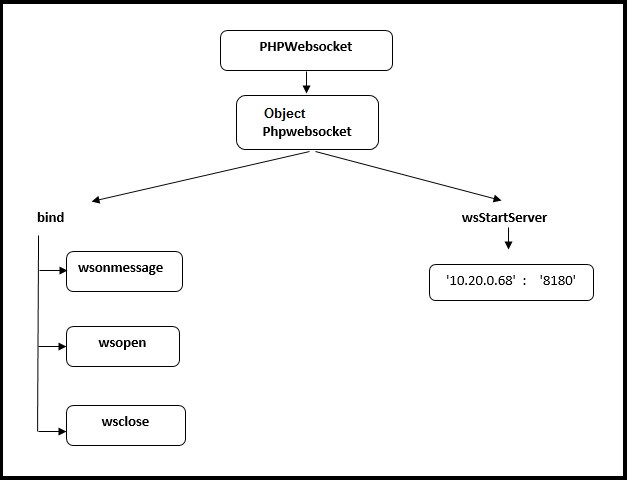
**2 Implementación del Servidor**

Como se trata de comunicación en tiempo real, para implementar un servidor es necesario utilizar un protocolo de comunicación que cumpla con los requerimientos de la investigación.

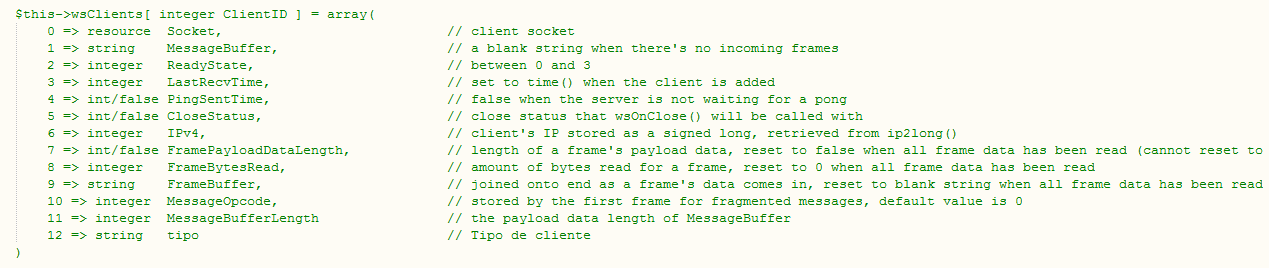
Los protocolos de comunicación son una serie de normas que usan los equipos informáticos para gestionar sus diálogos en los intercambios de información. Dos o más equipos diferentes de marcas diferentes se pueden comunicar sin problemas en el caso en que usen el mismo protocolo de comunicaciones.

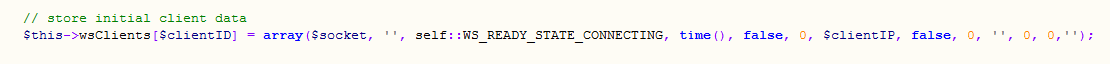
El protocolo de comunicación que se utiliza para implementar el servidor es el protocolo Websocket, que es una tecnología que proporciona un canal de comunicación bidireccional y full-duplex. Está diseñada para ser implementada en navegadores y servidores web, pero puede utilizarse por cualquier aplicación cliente/servidor.

El servidor está desarrollado en PHP, utilizando la librería PHPwebsocket que es una librería en software libre que contiene los métodos necesarios para la comunicación fullduplex en tiempo real.



Al tratarse de una librería libre se puede editar su contenido, para el re direccionamiento se añadió un campo al vector predefinido que viene en la librería, este campo permite almacenar el tipo de cliente que se conecta, almacenando el nombre, origen, destino y en base a esta información poder realizar el re direccionamiento de los paquetes entre los clientes que estén conectados en el servidor.





En el servidor se crean los métodos wsOnMessage, wsOnOpen, wsOnClose que son los que administran los mensajes que llegan al servidor, permiten abrir y cerrar la comunicación.

La función wsonmessage es en donde se realiza el redireccionamiento de paquetes, primero interpreta el paquete que esta en formato json y se realiza el re direccionamiento en base al origen y destino de los paquetes que llegan.

Una vez se tienen las funciones anteriormente mencionadas se debe especificar la dirección Ip y el puerto al que deben conectarse los clientes, al tratarse de un servidor perteneciente a Cedia, se trabaja con:

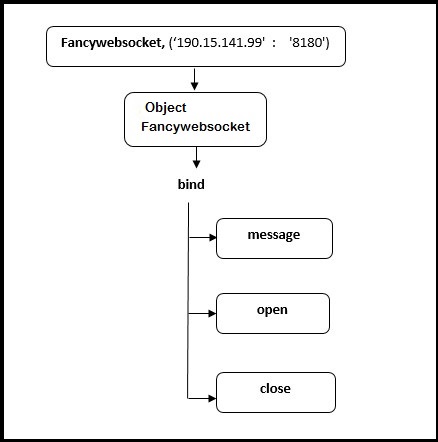
Ip interna 10.20.0.68, Ip Externa: 190.15.141.99, puerto: 8180.

**Plataforma Web**

**1 Cliente Web**

Una vez se tiene el proceso de tele operación se procede a realizar un cliente web que muestra los datos en tiempo real que estan siendo enviados por los clientes, y un modelo 3D que simula el movimiento del cliente que esta siendo tele operado.

En la parte del cliente web también es necesario utilizar el protocolo de comunicación websocket, esto se consigue utilizando la librería fancywebsocket que es una librería libre escrita en javascript, que contiene las funciones open, close, para la abrir y cerrar la conexión, message, para administrar los paquetes de datos, además se deben especificar la dirección ip y el puerto que estan abiertos en el servidor para conocer a donde debe conectarse el cliente web.



Para la visualización de los datos se utiliza javascript que es un lenguaje de programación interpretado orientado al lado del cliente implementado como parte de un navegador web y jquery que es una biblioteca javascript que permite manipular ordenadamente los elementos de una página web.

Desde el servidor se envian todos los paquetes que le llegan, a un cliente web que de la misma manera se identificará y esperará que le lleguen paquetes, este cliente contiene los métodos para desempaquetar los mensajes que llegan en formato Json, permitiendo manipular lo que se va a mostrar en tiempo real en el cliente web.

**2 Streaming de Audio y Video**

La tecnología de streaming se utiliza para optimizar la descarga y reproducción de archivos de audio y video que suelen tener un cierto peso, es decir es la forma con la que se puede compartir contenido multimedia desde una página web, sin necesidad de descargarlo al ordenador.

Una transmisión de streaming nunca queda almacenada en el equipo del usuario, razón por la cual la hace un poco mas segura para evitar plagio o que capturen el archivo de audio/video.

Se utilizó la Api WebRTC para implementar streaming de audio y video, Las API de WebRTC están diseñadas para permitir que las aplicaciones Javascript puedan crear conexiones en tiempo real con los canales de audio, vídeo y/o datos directamente entre usuarios a través de sus navegadores, o para servidores que soportan los protocolos de WebRTC.

2.1 Cliente Streaming

El cliente streaming se encarga de realizar la codificación de los datos de tal forma de que sean lo suficientemente “livianos” para tranmistirse rapidamente sin perder calidad. Codecs como AAC plus y mp3 para el Audio y H264 o VP6 en video son los mas comunes.

2.2 Servidor Streaming

Es este quien se encarga de recibir la señal desde el emisor, codificarla o decodificarla según sea el caso, y redistribuir dichos datos a tantos usuarios como se soliciten desde la pagina web del usuario.

Técnicamente cualquier PC podría ser servidor de streaming, sin embargo dado que son estos equipos quienes procesan todos los datos de audio y video requieren gran capacidad de procesamiento. Generalmente son servidores XEON QuadCore con hasta 8 núcleos. Por otra parte también son los encargados de la distribución hacia todo Internet. por tanto deben tener un gran ancho de banda y transferencia para poder recibir cientos o miles de solicitudes simultaneas.

**3 Modelo 3D**

El modelo generalmente se obtiene a partir de un diseño CAD en SolidWorks, el cual asegura el funcionamiento cinemático sin problemas. Es necesario guardarlo como un archivo de ensamblaje (\*.ASM).

Se importa el modelo en formato \*.asm en el programa 3ds Max, con el objetivo de establecer jerarquías de cada uno de los elementos que conforman el modelo 3D. La organización de los componentes mediante jerarquías asegura la correcta compatibilidad de movimiento cinemático que tendrá el modelo, dependiendo del funcionamiento del robot.

Después de este proceso, se debe exportar el modelo en formato \*.fbx, para posteriormente importarlo en Unity, El proceso de agregar texturas al modelo se lo realiza dentro del entorno de Unity con el fin de observar en tiempo real el estado del modelo y que los recursos empleados sean compatibles con Unity.

Para darle movimiento al modelo 3D, tal y como lo haría un modelo en la vida real, se analizará su modelo cinemático. En la mayoría de modelos mecánicos y robóticos existen dos tipos de movimiento predominantes el movimiento por punto de rotación y el movimiento por rigging.

Una vez se tiene listo el modelo 3m se lo exporta en formato WebGl, para que sea compatible con web, es decir reprodicir el video 3d desde una pagina Web.

El formato WebGl contiene los archivos Javascript, css, html necesarios para poder utilizar las funciones creadas desde unity para dar movimiento al modelo 3D. Al modelo 3d se le añade la funcionalidad de Websocket para que pueda reflejar los mismos moviemientos que realiza el robot real, esto se logra al enviar los mismos datos al robot real y al modelo 3d creado.