

# Plataforma Web de Comunicación en Tiempo Real para Tele-rehabilitación

Your Name  
School of Electrical and  
Computer Engineering  
Institute of Technology  
99999 Testcity  
Email: test@test.tes

Second Name  
Ecole Superieure  
Nantes, France  
Email: second@second.fr

Third Name  
and Fourth Name  
Star Academy  
San Francisco, California 99999-9999  
Telephone: (800) 555-5555  
Fax: (888) 555-5555

**Abstract—** En este trabajo se describe el desarrollo de una plataforma de tele-rehabilitación en tiempo real, se propone el uso del protocolo de comunicación bidireccional Websocket para la comunicación entre el sitio local y remoto, JSON para estructuración de los datos, WebRTC para la transmisión de audio y video y finalmente Unity para construir un modelo 3D.

En este contexto el paciente recibe los movimientos que son realizados en el sitio remoto por el especialista a través de un dispositivo de retroalimentación, en este proyecto se utiliza el dispositivo háptico Novint Falcon. En cambio el especialista recibe el stream de video que transmite el paciente y un modelo 3D que refleja los movimientos reales del paciente. Por último, los resultados experimentales son reportados para verificar el funcionamiento del sistema propuesto.

**Index Terms—**Tele-Rehabilitación, 3d, Protocolos, websocket, Json, cliente, servidor

## I. INTRODUCCIÓN

El aumento de la demanda de programas de rehabilitación en la mayoría de países de nuestro entorno se debe tanto al incremento de la longevidad de la población, como al creciente número de individuos que presentan algún grado de discapacidad como resultado de múltiples procesos patológicos. La telerehabilitación se presenta como una alternativa útil y accesible desde el punto de vista tecnológico y económico[1].

La tele-rehabilitación es la aplicación de las tecnologías de la información y las comunicaciones (TICs) para poder realizar acciones de rehabilitación a distancia, desarrollada desde disciplinas como la Telesalud, la Telemedicina y la Teleasistencia[2]. La telerehabilitación permite, entre otras cosas, minimizar los desplazamientos de pacientes y profesionales en rehabilitación, lo que conlleva a una mejora en las condiciones de atención para pacientes que se encuentran en zonas rurales o zonas de difícil acceso, como también la de los pacientes que por su condición no pueden movilizarse en áreas urbanas. La figura 1 muestra el esquema general de la plataforma de tele-rehabilitación.

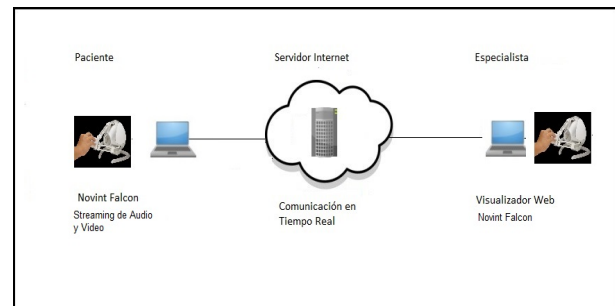


Fig. 1. Esquema de Plataforma Tele-rehabilitación

Existen pocos estudios sobre plataformas de tele-rehabilitación, pero entre las que mas impacto social han generado estan: Biomechanical motion capture methods focused on tele-physiotherapy de los autores M. Callejas Cuervo, A. F. Ruiz Olaya, R. M. Gutiérrez Salamanca, en la cual realizan un esquema en Tele-fisioterapia para personas con trastornos motores del miembro superior[3], en cambio el autor T.Rovira realiza un programa clínico de tele-rehabilitación cognitiva en el traumatismo craneoencefálico.

El trabajo que genera este artículo muestra el uso de las TICs para implementar un modelo de tele-rehabilitación del brazo, para ser mas específicos de los daños que se ocasionan en las muñecas de la mano, permitiendo asegurar la transferencia de información entre los profesionales de la rehabilitación y las personas que necesitan rehabilitación.

Este trabajo se encuentra dividido de la siguiente manera, la sección 2 describe el dispositivo de retroalimentación que se va a utilizar, la sección 3 muestra la información sobre Modelo 3D, en la sección 4 se muestra el desarrollo plataforma de comunicación, la sección 5 muestra la implementación de streaming en tiempo real, Un visualizador de datos web en tiempo real se muestra en la sección 6, en la sección 7 se muestran los resultados obtenidos y finalmente en la sección 8 estan las conclusiones y trabajo futuro.

## II. DISPOSITIVO HÁPTICO NOVINT FALCON

Háptica es un área que estudia e investiga la interacción de la modalidad sensorial del tacto con un mundo virtual. Las interfaces hápticas son dispositivos bidireccionales que

proporcionan sensaciones de fuerzas o tacto al operador a través de la misma interfaz con la que envía consignas al sistema remoto; son básicamente posicionadores de avanzadas prestaciones que permiten simular sensaciones táctiles gracias a la realimentación de fuerzas [4].

De acuerdo a los resultados de selección de una interfaz háptica para aplicaciones de asistencia en tele-rehabilitación se concluye que los joysticks hápticos Phantom Omni® y Novint Falcon™ son dispositivos de bajo costo con una serie de características electromecánicas que permiten una interacción apropiada en sistemas de teleoperación donde se requiere de ciertas habilidades manuales especiales.

Es un dispositivo con realimentación de fuerzas de configuración paralela, fabricado por Novint Technologies Inc., el cual permite la interacción en escenarios tridimensionales 3D. El Novint HDAL SDK es el kit de desarrollo para el dispositivo Novint Falcon y contiene toda la documentación y los archivos de software necesarios para desarrollar aplicaciones con el dispositivo háptico desde la capa de abstracción HDAL (Haptic Device Abstraction Layer)[4].

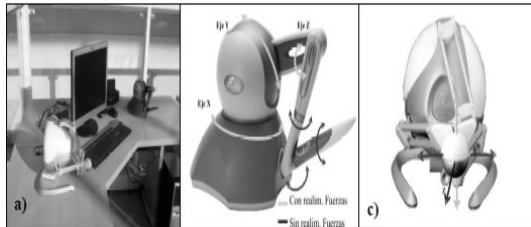


Fig. 2. a) Estación de trabajo de transmisión y recepción. b) Falcon Novint

Se captura los datos que muestre el Novint Falcon en tiempo real, y se los envía en formato json hacia el servidor de internet, utilizando el protocolo de comunicación websocket.

### III. MODELO 3D

En computación, un modelo en 3D es un "mundo conceptual en tres dimensiones". Un modelo 3D puede "verse" de dos formas distintas. Desde un punto de vista técnico, es un grupo de fórmulas matemáticas que describen un "mundo" en tres dimensiones. Desde un punto de vista visual, valga la redundancia, un modelo en 3D es una representación esquemática visible a través de un conjunto de objetos, elementos y propiedades que, una vez procesados, se convertirán en una imagen en 3D o una animación 3d[5].

En la actualidad debido a los avances tecnológicos, la medicina requiere de ciertos adelantos que impliquen programas computacionales en línea, que sean óptimos, rápidos y que manejen datos en tiempo real, para solucionar los problemas que se presentan.

El modelo 3D es desarrollado en Unity con las funciones necesarias para capturar los datos que envía el falcon Novint y realizar el mismo movimiento que hace la mano en tiempo real. Una vez terminado el modelo se exporta en formato WebGL con el fin de poder visualizarlo en una pagina web.

Al Modelo WebGL, se le añade la funcionalidad de un cliente websocket, en nuestro caso similar al visualizador web que se explica mas adelante, esto se hace para poder transmitir los datos en tiempo real desde sitio local al sitio remoto, la figura 3 muestra el modelo 3D que se utiliza para replicar el movimiento del paciente.

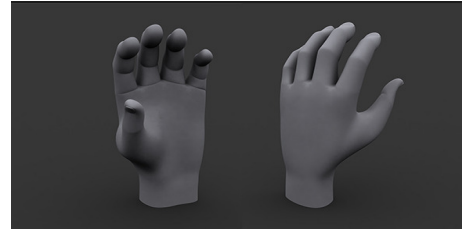


Fig. 3. Modelo 3D

### IV. PLATAFORMA DE COMUNICACIÓN

La plataforma de comunicación está desarrollada en php, utilizando la librería web PHPwebsocket.php, que es una librería que contiene los métodos, funciones y variables que utiliza el protocolo websocket para poder crear el canal de comunicación[6]. La figura 4, muestra cómo se realiza el proceso de funcionamiento del servidor.

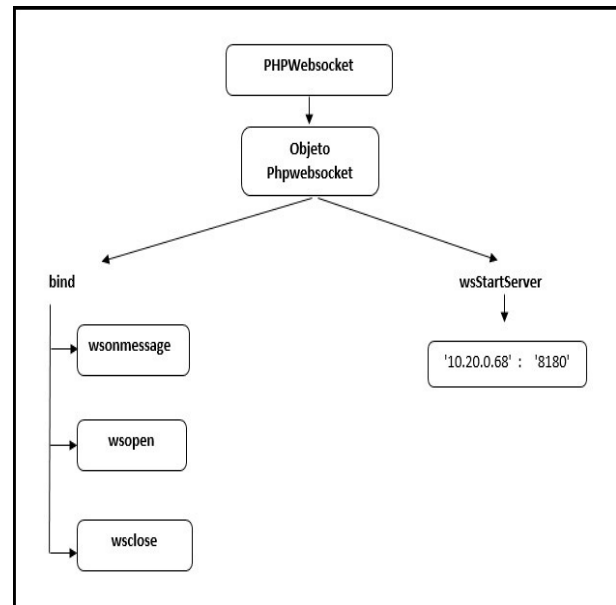


Fig. 4. Esquema del Servidor

El funcionamiento de forma detallada es el siguiente:

- Crear una instancia de la librería PHPWebSocket, para poder utilizar los métodos que contiene esta librería.
  - `$this->socket= new PHPWebSocket();`
- Enlaza la función wsOnMessage con la librería PHP-Websocket mediante "message", permitiendo utilizar las variables y funciones referentes a los mensajes que se están enviando.

- `$this->socket->bind('message', 'wsOnMessage');`
- Enlaza `wsOnOpen` con la librería `PHPWebsocket` mediante “open”. Permitiendo utilizar las funciones y variables que permiten abrir la comunicación.
  - `$this->socket->bind('open', 'wsOnOpen');`
- Enlaza `wsOnClose` con la librería `PHPWebsocket` mediante “close”. Permitiendo utilizar las funciones y variables que permiten cerrar la comunicación.
  - `$this->socket->bind('close', 'wsOnClose');`
- Inicializa el socket en la ip y puerto especificados.
  - `return $this->socket->wsStartServer('192.168.1.102',9300);`

## V. STREAM DE AUDIO Y VIDEO

El streaming se refiere a la entrega de flujos de datos, generalmente con contenidos multimedia. Ello trae ventajas que no son posibles utilizando procedimientos comunes de descarga de ficheros. Así como se realiza el streaming de contenidos estáticos, puede realizarse el Streaming de contenido en vivo[7].

En el presente proyecto para transmisión y recepción de video y audio se utiliza en protocolo estandar WebRTC.

WebRTC, también conocido como Web Real-Time Communications, es un proyecto de código abierto – promovido por Google, Mozilla y otros – que permite comunicaciones en tiempo real sin plug-ins a través de una API Javascript. Facilita las aplicaciones de llamadas de voz, chat de video y compartimiento de archivos entre navegadores[8], la arquitectura de funcionamiento se muestra en la figura 5.

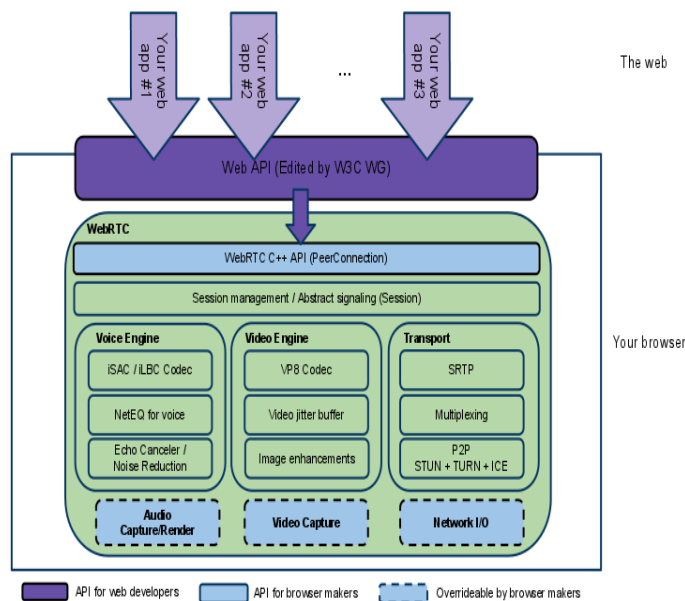


Fig. 5. Arquitectura WebRTC de: <https://webrtc.org/>

## VI. VISUALIZADOR WEB

Se creó una página web que mostrará los datos enviados por los clientes, esta página web también es un cliente más con

la diferencia que es un cliente de visualización de datos, se utilizó html y css para la creación de la misma. La aplicación cliente, es decir el websocket cliente está escrito en javascript y jquery, en esta parte también se utiliza una librería que contiene los métodos para conectarse a un websocket server [9]. Esta librería se llama fancywebsocket, de la misma manera tiene los métodos open, message, close, el esquema se muestra en la figura 6.

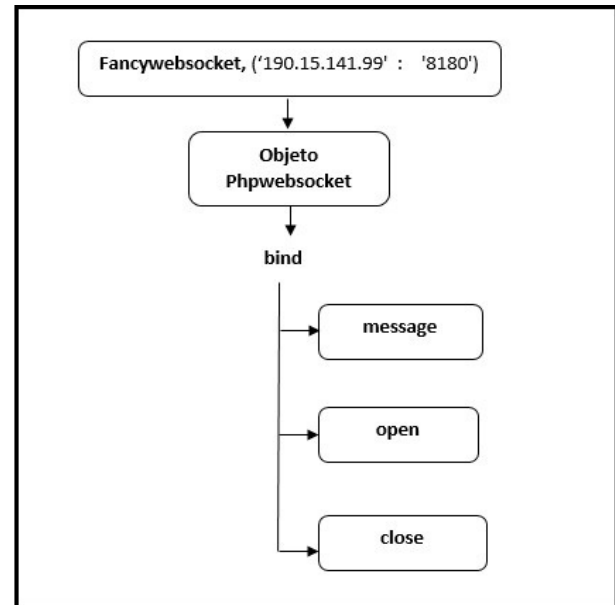


Fig. 6. Esquema del Cliente

El funcionamiento es el siguiente:

- Crear una instancia de la librería fancywebsocket, especificando la dirección Ip y el puerto al que se debe conectar.
  - `Server = new FancyWebSocket('ws://192.168.1.102:9300');`
- Abrir la conexión para poder conectarse al servidor.
  - `Server.bind('open', function(){..... });`
- Envía el mensaje hacia el servidor enpaquetado en formato json.
  - `Server.send('message', JSON.stringify(mensaje) );`
- Cierra la conexión con el servidor.
  - `Server.bind('close', function( data ) -`
- Obtiene mensajes de los clientes para poder mostrarlos en pantalla.
  - `Server.bind('message', function( payload ); var res = jQuery.parseJSON(payload); console.log(res);`

## VII. RESULTADOS

### A. Servicio de Comunicación

Se tiene un servidor apache 2.0 en el cual se debe iniciar servicio websocket esta en espera de los clientes que se van a conectar y brinda la comunicación full-duplex. Se inicia a través de línea de comandos en terminal linux en nuestro caso, para esto se hace lo siguiente:

- Ubicarse en la dirección donde estén los archivos y escribir : /var/www/html/Teleoperacion-Bilateral/.
- Escribir el comando para iniciar el servicio websocket, la figura 7 detalla los servicios que se inician. php index.php Administrador/iniciar\_socket.

```

Actividades Terminal mar 19:34
renato@renato: ~
renato@renato:~$ su
Contraseña:
root@renato:/home/renato# cd /var/www/html/Teleoperacion-Bilateral/
root@renato:/var/www/html/Teleoperacion-Bilateral# php index.php Administrador/iniciar_socket

```

Fig. 7. Inicar Websocket

### B. Visualizador Web

En cuanto al visualizador web, Conectar el cliente abriendo un navegador web especificando la dirección del servidor, en este caso se trabaja en un servidor local, como se muestra en la figura 8 <http://190.15.141.9/Teleoperacion-Bilateral/Administrador/>

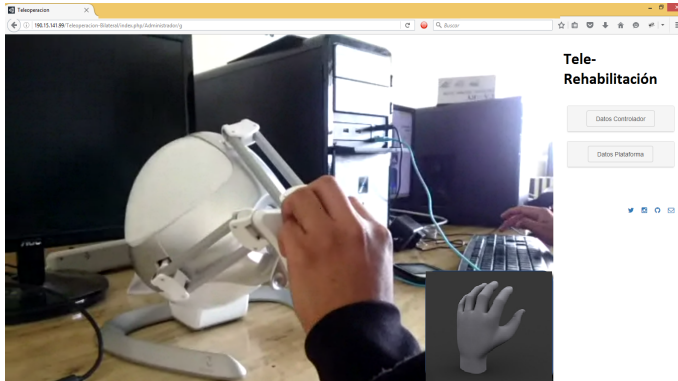


Fig. 8. Cliente Web

### C. Comunicación Bidireccional

- La primera conexión permite identificar al servidor que cliente se conecta, como se puede ver en la figura 9.
- El cliente envía el mensaje de la siguiente forma: {"cadena":"c"}
- Desde el cliente se envían los datos en formato JSON {"cadena":"cadena","cadena":"cadena"}

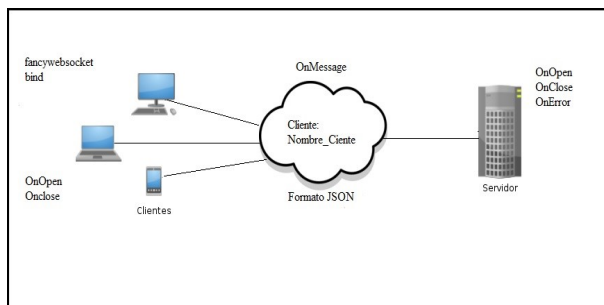


Fig. 9. Identificación de clientes

- Los mensajes llegan al servidor identificando {"cadena":"cadena","cadena":"cadena"}
- El re-direccionamiento de los mensajes va a variar dependiendo de los datos que llegue en la cadena json en el campo origen y destino, como se observa en la figura 10.

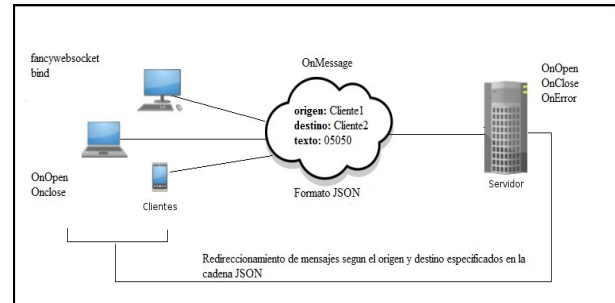


Fig. 10. Redireccionamiento

## VIII. CONCLUSIONES Y TRABAJO FUTURO

Se utilizó el protocolo de comunicación websocket, debido a que brinda una comunicación estable, full duplex y con un tiempo de respuesta mínimo.

Es necesario empaquetar los datos en formato Json para poder utilizar cada campo de la cadena que se crea, permitiendo tener el control de los datos que entran y salen del servidor.

La animación en 3D en tiempo real de sistemas redundantes permite tener un conocimiento de las diferentes posiciones de las articulaciones de la mano, ayudando a realizar las tareas de mejor manera.

Una vez se tiene funcionando correctamente la plataforma de tele-rehabilitación, se puede realizar funciones para almacenar los datos que envía el paciente, con el fin de ser analizados y verificar que la plataforma funciona de manera eficiente o si necesita algún cambio en su estructura o programación.

## ACKNOWLEDGMENT

bla bla

## REFERENCES

- [1] M. McCue, A. Fairman, and M. Pramuka, "Enhancing quality of life through telerehabilitation," *Physical medicine and rehabilitation clinics of North America*, vol. 21, no. 1, pp. 195–205, 2010.
- [2] M. Pramuka and L. van Roosmalen, "Telerehabilitation technologies: Accessibility and usability," *International Journal of Telerehabilitation*, vol. 1, no. 1, pp. 85–98, 2009. [Online]. Available: <http://telerehab.pitt.edu/ojs/index.php/Telerehab/article/view/6016>
- [3] M. C. Cuervo, A. R. Olaya, and R. G. Salamanca, "Biomechanical motion capture methods focused on tele-physiotherapy," in *Health Care Exchanges (PAHCE), 2013 Pan American*. IEEE, 2013, pp. 1–6.
- [4] J. M. S. E. Pinto Salamanca, Mara Luisa; Sabater Navarro, "Anlisis de la deteccin de colisiones en un entorno virtual para aplicaciones hapticas de asistencia quirrgica," vol. 31, pp. 204–212, 2011.
- [5] J. Trivino. (2012). [Online]. Available: <https://prezi.com/ykbfoua8ukra/modelado-2d-3d/>
- [6] M. Casario, P. Elst, C. Brown, N. Wormser, and C. Hanquez, *HTML5 Solutions: Essential Techniques for HTML5 Developers*. Springer, 2011.

- [7] E. Z. Huavel, L. S. Nieto, L. M. Ulloa, and H. V. Urco, "Streaming de video en vivo por internet," *Electrónica-UNMSM*, no. 27, pp. 36–43, 2011.
- [8] J. K. Nurminen, A. J. R. Meyn, E. Jalonen, Y. Raivio, and R. G. Marrero, "P2p media streaming with html5 and webrtc," in *Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on*, April 2013, pp. 63–64.
- [9] M. Banchoff Tzancoff, "Websocket: comparación de performance e implementación de aplicaciones web," Ph.D. dissertation, Facultad de Informática, 2011.