



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Creación de un sistema asistido de posicionamiento de un balón en un encuentro deportivo de fútbol para un servicio de HbbTV para aplicaciones de accesibilidad a interactividad.

Narvaez Cruz, Andrés Fernando

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica y Telecomunicaciones

Trabajo de titulación, previo a la obtención del título de Ingeniero en Electrónica y Telecomunicaciones

Ing. Larco Bravo, Cesar Julio Msc.

22 de julio del 2022



04_Trabajo_Titulación_Narvaez_Cruz_Andres_Narvaez.pdf

Scanned on: 21:5 August 11, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text

Identical Words	455
Words with Minor Changes	78
Paraphrased Words	308
Omitted Words	0





Departamento de Eléctrica, Electrónica y Telecomunicaciones
Carrera de Ingeniería en Electrónica y Telecomunicaciones

Certificación

Certifico que el trabajo de titulación, “Creación de un sistema asistido de posicionamiento de un balón en un encuentro deportivo de fútbol para un servicio de HbbTV para aplicaciones de accesibilidad a interactividad” fue realizado por el señor **Narvaez Cruz, Andrés Fernando** el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí, 22 de julio del 2022



Firmado digitalmente por:
**JULIO CESAR
LARCO BRAVO**

Ing. Larco Bravo, Cesar Julio
C. C: 1710638808



Departamento de Eléctrica, Electrónica y Telecomunicaciones
Carrera de Ingeniería en Electrónica y Telecomunicaciones

Responsabilidad de Autoría

Yo, **Narvaez Cruz Andrés Fernando**, con cédula de ciudadanía no 1726514431, declaro que el contenido, ideas y criterios del trabajo de titulación: **“Creación de un sistema asistido de posicionamiento de un balón en un encuentro deportivo de fútbol para un servicio de HbbTV para aplicaciones de accesibilidad a interactividad”** es mi autoría y responsabilidad, cumpliendo con requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 22 de julio del 2022



.....
Narvaez Cruz Andrés Fernando

C. C: 1726514431



Departamento de Eléctrica, Electrónica y Telecomunicaciones
Carrera de Ingeniería en Electrónica y Telecomunicaciones

Autorización de Publicación

Yo, **Narvaez Cruz Andrés Fernando**, con cédula de ciudadanía no 1726514431, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **“Creación de un sistema asistido de posicionamiento de un balón en un encuentro deportivo de fútbol para un servicio de HbbTV para aplicaciones de accesibilidad a interactividad”** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad

Sangolquí, 22 de julio del 2022

A handwritten signature in blue ink, which appears to read 'Narvaez Cruz Andrés Fernando', is written over a horizontal dotted line.

Narvaez Cruz Andrés Fernando

C. C: 1726514431

Dedicatorias

Dedico principalmente este trabajo a mis padres Pedro y Sofía, quien con su esfuerzo, dedicación y preocupación lograron que yo pueda finalizar esta etapa tan en mi vida tan importante para verme crecer como profesional y humano. A mi hermana Micaela que siempre estuvo incondicionalmente allí desde el inicio hasta el fin de mi carrera universitaria, dedico también a Andrea Jiménez quien siempre me apoyo en cada metra propuesta. A mi abuelita Ruby y a mi bisabuelita Blanquita quienes siempre estuvieron pendientes de mí y apoyándome en cualquier situación que se presentase. A mi primo David quien desde otro país fue un apoyo mutuo para cumplir los sueños y metas de cada uno. A Dayana, Jonathan y todos mis amigos y compañeros que a lo largo de la carrera fueron un pilar importante para cumplir esta meta. Por último, dedico este trabajo a Dios, por darme las fuerzas y perseverancia para no rendirme en este camino y continua al siguiente.

Narvaez Cruz Andrés Fernando

Agradecimiento

Agradezco principalmente a mi tutor Julio Larco y Diego Villamarín, quienes me fueron guías en este trabajo de titulación apoyándome con su conocimiento y consejos me permitieron alcanzar esta meta.

Agradezco a todos los compañeros y docentes de la Universidad de las Fuerzas Armadas ESPE por formar parte de este desarrollo ético, profesional y social para la implementación de nuevos profesionales en el mundo.

Narvaez Cruz Andrés Fernando

Tabla de contenido

Reporte Antiplagio	2
Certificación	3
Responsabilidad De Autoría	4
Autorización De Publicación	5
Dedicatorias	6
Agradecimiento	7
Índice De Tablas	11
Índice De Figuras	12
Resumen	14
Abstract	15
Capítulo I: Introducción	18
Antecedentes	19
Justificación e importancia del proyecto	20
Alcance del proyecto	21
Objetivos	22
General	22
Específicos	22
Resumen de contenidos	23
Primer Capítulo	23

	9
Segundo Capítulo _____	23
Tercer Capítulo _____	23
Cuarto Capítulo _____	24
Quinto Capítulo _____	24
Capítulo II: Marco Teórico _____	25
Televisión digital terrestre _____	25
Estándares de la Televisión Digital Terrestre _____	27
Beneficios de la TDT _____	30
HBBTV (Hybrid Broadcast Broadband TV) _____	30
Definición _____	30
Arquitectura _____	33
Especificaciones _____	34
Tecnologías utilizadas _____	36
Herramientas _____	40
Python _____	40
Plataforma Digital Ocean _____	40
Raspberry Pi _____	40
Plataforma IoT Adafruit _____	41
FFmpeg _____	42
Capítulo III: Diseño E Implementación _____	43
Herramientas de desarrollo _____	43

	10
Instalación y configuración del software _____	44
Esquema general del proyecto _____	46
Sistema de interpretación de datos _____	49
Sistema de servidor de datos _____	55
Capítulo IV: Resultados _____	65
Funcionamiento del Sistema _____	65
Guante Háptico _____	71
Evaluación con usuarios _____	74
Capítulo V: Conclusiones Y Recomendaciones _____	80
Conclusiones _____	80
Recomendaciones _____	81
Trabajos Futuros _____	82
Referencias Bibliografía _____	83

Índice De Tablas

Tabla 1 <i>Especificaciones estándar HbbTv</i>	35
Tabla 2 Librerías utilizadas en el sistema asistido.	51
Tabla 3 Codificación del vector posición en el plano bidimensional	53
Tabla 4 Librerías utilizadas en el sistema del servidor de datos.....	57
Tabla 5 Características del servidor básico en la plataforma Digital Ocean.	58
Tabla 6 Pruebas de diagnóstico de red de pruebas ICMP	66
Tabla 7 Muestra de datos tomados del archivo generado en el servidor.....	71
Tabla 8 Muestra de preguntas	77

Índice De Figuras

Figura 1 <i>Arquitectura TDT</i>	25
Figura 2 <i>Frecuencia de transmisión MFN-SFN</i>	27
Figura 3 <i>Difusión de sistemas de TDT en el mundo</i>	29
Figura 4 <i>Arquitectura HbbTv</i>	34
Figura 5 <i>Arquitectura del fichero MPD en MPEG-DASH</i>	38
Figura 6 <i>Arquitectura protocolo DVB-CSS</i>	39
Figura 7 <i>Elementos de Raspberry Pi</i>	41
Figura 8 <i>Plataforma IoT Adafruit</i>	42
Figura 9 <i>Esquema general del sistema</i>	48
Figura 10 <i>Diagrama de flujo del sistema de interpretación de datos.</i>	49
Figura 11 <i>Imagen de campo de fútbol implementado</i>	50
Figura 12 <i>Referencia de las subdivisiones del campo de futbol.</i>	52
Figura 13 <i>Diagrama de la interfaz del sistema implementado.</i>	54
Figura 14 <i>Interfaz inicializada</i>	55
Figura 15 <i>Diagrama de flujo del sistema de servidor de datos.</i>	56
Figura 16 <i>Datos tomados del archivo CSV</i>	59
Figura 17 <i>Servidor dedicado en la plataforma Digital Ocean</i>	60
Figura 18 <i>Pruebas de la página web incluido el complemento HbbTv</i>	61
Figura 19 <i>Servidor IoT en la plataforma Adafruit</i>	62
Figura 20 <i>Usuario y llave para la configuración de Adafruit en dispositivos</i>	64
Figura 21 <i>Pruebas iniciales del sistema</i>	66
Figura 22 <i>Presentación del sistema de interpretación de datos</i>	68
Figura 23 <i>Captura del analizador de protocolos Wireshark</i>	69

Figura 24 <i>Paquete UDP del analizador de protocolos Wireshark</i>	70
Figura 25 <i>Haptic Glove Accesible TV</i>	72
Figura 26 <i>Interpretación de ubicación entre el sistema y el guante háptico</i>	73
Figura 27 <i>Adaptación de la sala de pruebas</i>	75
Figura 28 <i>Prueba del sistema y dispositivo de aplicación a la accesibilidad</i>	76
Figura 29 <i>Valores de aceptación del dispositivo</i>	78
Figura 30 <i>Valores de aceptación posterior a la segunda prueba</i>	79

Resumen

En la actualidad, el desarrollo de sistemas de reconocimiento o posicionamiento de objetos es un tema que se encuentra en constante desarrollo, la implementación de estos trae beneficios a diferentes áreas principalmente a los sectores vulnerables. Sistemas de ayuda asistida desarrollados se limitan a ofrecer información de eventos que ya ocurrieron porque es más sencillo procesar información ya almacenada. Una forma de extraer esa información es codificarla mediante el uso de vectores de posicionamiento que luego son transmitidos en un formato adecuado a la persona con capacidades especiales. El objetivo de este proyecto es ayudar a personas con discapacidad visual a que disfruten de un partido de fútbol, lo que se va a realizar mediante un operador que se encuentre observando el evento deportivo de fútbol, sea en vivo o pregrabado, este operador interpretará la posición del balón mediante una tableta gráfica llegando a generar vectores de posicionamiento con la ayuda de una interfaz diseñada en una plataforma de desarrollo. Esta plataforma almacena dichos datos en un servidor web para la tecnología HbbTv accesible para ofrecer una conexión con los eventos futbolísticos y aplicaciones de accesibilidad a la interactividad. Los resultados del desarrollo de este sistema asistido de posicionamiento de un balón en un campo de fútbol pueden abrir las puertas a varias aplicaciones destinadas al uso de datos y a brindar servicios de accesibilidad y más aún si los datos que se generan están disponibles en servidores web siendo sistemas económicos en sentido de implementación y desarrollo.

Palabras clave: sistema asistido, HbbTv, interactividad

Abstract

Currently, the development of object recognition or positioning systems is a topic that is in constant development, the implementation of these systems brings benefits to different areas, mainly to vulnerable sectors. The assisted assistance systems developed are limited to providing information on events that have already occurred because it is easier to process information that has already been stored. One way to extract that information is to encode it by using positioning vectors that are then transmitted in a suitable format to the person with special abilities. The objective of this project is to help visually impaired people to enjoy a soccer match, which will be done by an operator who is watching the soccer sporting event, either live or pre-recorded, this operator will interpret the position of the ball through a graphics tablet to generate positioning vectors with the help of an interface designed on a development platform. This platform stores the data on a web server for HbbTv technology accessible to provide a connection with soccer events and interactivity accessibility applications. The results of the development of this assisted system of positioning a ball on a soccer field can open the doors to several applications aimed at using data and providing accessibility services and even more if the data generated are available on web servers being economic systems in the sense of implementation and development.

Key words: assisted system, Hbbtv, interactivity

Capítulo I: Introducción

En la actualidad, el desarrollo de sistemas de reconocimiento o posicionamiento de objetos es un tema que se encuentra en constante desarrollo, la implementación de estos trae beneficios a diferentes áreas, ya sea para generar aplicaciones para sectores vulnerables o para desarrollar aplicaciones de sistemas estadísticos y análisis de datos. Por lo general, sistemas ya desarrollados se usan con una gran carga computacional, los métodos más conocidos se basan en *Machine Learning* y *Deep Learning*, los dos métodos se basan en registrar una base de datos previa o realizar un aprendizaje mediante las mismas. Otros métodos se limitan a ofrecer información que se extrae de eventos pasados, esto se debe a que los mismos generan sus vectores de forma manual. El desarrollo de un sistema asistido de posicionamiento puede llegar a brindar un método más sencillo y básico de obtener los datos, reemplazando el complejo desarrollo del aprendizaje de máquina por un operador físico. (“Artificial Intelligence & Machine Learning”, 2017)

De igual forma, la televisión digital está empezando a surgir con diferentes aplicaciones para mejorar la calidad visual hacia los usuarios. A más de optimizar mejor el espectro radioeléctrico, mejora la calidad de la imagen, sonido e incluye la posibilidad de interactuar con el usuario.

Proyectos relacionados con la nueva interactividad de la televisión digital fueron creados, por ejemplo, HbbTv (*Hybrid Broadcast Broadband TV*) es un proyecto paneuropeo de televisión híbrida cuyo objetivo es combinar emisiones de televisión con servicios de banda ancha para entregar a los usuarios un servicio de entretenimiento a través de una pantalla de TV. (Gobierno de España, 2020)

HbbTV genera nuevas experiencias hacia la TV interactiva, se pueden acceder a nuevos servicios de entretenimiento, entre los principales que se va a enfocar el proyecto es: Publicidad interactiva e Información personalizada en el televisor ya que, al ser un sistema para aplicaciones de interactividad, debe obtener una forma de conexión de una manera fácil o detallada la cual se visualizará en forma de información personalizada. (Instituto Europeo de Normas de Telecomunicaciones, 2021)

Antecedentes

En la actualidad, existe una gran cantidad de proyectos relacionados al reconocimiento de objetos basados en *Machine Learning* y *Deep Learning*, lo que no se puede decir en relación de sistemas asistidos. Entre la mayoría de los proyectos relacionados a reconocimiento de objetos, se puede mencionar a lo que hoy en día se utiliza más lo que es el sistema de reconocimiento de personas o la detección de objetos para el desarrollo de pilotos automáticos en vehículos.(Foresti, 1999)

En el ámbito de sistemas asistidos, al momento nos podemos fijar más en trabajos a distancia o trabajos remotos, dichos trabajos son basados más al control de equipos o maquinaria. La empresa CHINA MOLIBDENO, debido a la pandemia ocasionada por el Covid-19 ha empezado a utilizar el sistema de redes telefónicas 5G para adaptar la maquinaria de la mina de Sandaozhuang en equipos remotos que pueden ser controlados desde los hogares de los trabajadores, esto debido a sistemas asistidos de control combinados con un sistema de red de baja latencia como lo es el 5G. (GrupoRPP, 2020)

Poniendo en contexto de trabajos relacionados con la ayuda o la asistencia a personas con discapacidad visual durante partidos de futbol, se empiezan a desarrollar

dispositivos que permiten interactuar entre la persona no vidente y el encuentro deportivo por medio de un guante el cual emite vibraciones, pero este solamente permite la asistencia de eventos pasados ya que su configuración es de uso mediante la obtención de los datos de forma manual una vez se haya terminado el partido, lo que con el proyecto presente se pretende mejorar. (Villamarin & Mendez, 2021)

Justificación e importancia del proyecto

El desarrollo social en la actualidad se ha mantenido en constante crecimiento, principalmente la creación de sistemas y aplicaciones para mejorar la vida de ciertas personas, con ello, unirlas más con el entorno a su alrededor y mejorar las relaciones interpersonales. Se puede mencionar que uno de los sectores vulnerables con más dificultad de relacionarse con el entorno son personas con deficiencia visual por dicha razón han generado varios prototipos de manejo para mantenerlos conectados con el mundo. (Villamarin & Narvaez, 2021)

Sistemas de reconocimiento de objetos son una tendencia hoy en día, su uso y desarrollo de aplicaciones se ha visto limitado al alto costo que estos representan ya que en su mayoría son autónomos y necesitan de amplias bases de datos para desarrollar efectivamente el procesamiento de los datos y adicional otras bases de datos de almacenamiento de la información obtenida en caso de que se use para un propósito mayor. De la misma forma, el crecimiento de la televisión digital trae nuevos campos para el desarrollo de lo que se conoce como la TV interactiva. Al unir estos dos proyectos se puede generar varias ideas para impactar en el sector social tanto de personas pertenecientes al sector vulnerable o personas que desean obtener acceso a dichas aplicaciones para mejorar la experiencia. (Eslava Arce, 2014)

Alcance del proyecto

El objetivo principal del proyecto es conformar un sistema capaz de obtener datos de posición en forma de vectores para ser usados por aplicaciones de accesibilidad a la interactividad en tiempo real durante los eventos deportivos. Este se centra en 4 puntos principales, el primero es el desarrollo de una interfaz capaz de transformar la posición del cursor en la pantalla en un vector, este debe ser amigable con el operador, que sea fácil de controlar y a su vez, pueda procesar y enviar los datos de una manera eficiente hacia un servidor. La categorización de los datos se realizará dependiendo la posición del puntero en la pantalla, la misma que se codificará en áreas rectangulares delimitadas por un cierto número de pixeles; para realizar un envío seguro de datos, a cada área se le dará un código numérico el cual servirá para enviar al servidor cada cierto intervalo de tiempo para así, tener los datos en lo que se puede delimitar como tiempo real de la posición del balón en un campo de fútbol. El segundo punto es el desarrollo del programa de recepción y almacenamiento de datos que será ubicado en el servidor, este categorizará el orden de llegada de los paquetes y los almacenará en el archivo de datos. El mismo catalogará los datos recibidos y decodificará la posición en un vector X, Y, Z para determinar el área en la que se encuentra el balón. Adicional tomará el mismo vector posición y lo enviará por un puerto al cual se podrán conectar las aplicaciones de accesibilidad a la interactividad. El tercer punto es la conexión entre el servidor y la aplicación de TV digital. Este debe enviar un mensaje, en este caso en el inicio del encuentro deportivo, el cual reciba el telespectador y pueda conectar su dispositivo de interactividad al servidor. Como cuarto punto se tiene a los aplicativos de interactividad. Estos pueden ayudar al espectador a

tener una experiencia diferente, así como ayudar a personas de algún sector vulnerable, a relacionarse con lo que sucede en el encuentro futbolístico.

Para iniciar con el desarrollo de la interfaz, se tendrá que realizar una investigación sobre los diferentes lenguajes de programación, esto para desarrollarlo de una forma más eficiente y práctico. Al completar cada etapa se realizarán pruebas de funcionamiento y corrección de fallas que se vayan detectando ya sea en la generación de datos o en el envío de estos a las diferentes etapas. Al final se realizará un análisis de calidad de servicio para verificar si los protocolos utilizados son los correctos en el envío de datos en tiempo real

Objetivos

General

- Desarrollar un sistema asistido de posicionamiento de un balón en un encuentro deportivo de fútbol para un servicio de HbbTv para aplicaciones de accesibilidad a interactividad

Específicos

- Realizar un previo estudio sobre los sistemas manuales, asistidos o autónomos de reconocimiento de objetos en un video.
- Determinar el hardware y software óptimo para el reconocimiento de la posición de un balón en un encuentro deportivo de fútbol para el desarrollo del sistema asistido.
- Implementar una interfaz en una plataforma de desarrollo para el control y la conectividad entre el usuario guía y el sistema de posicionamiento.

- Establecer conexión entre el software desarrollado y el servidor web para obtener datos con el menor retardo posible.

Resumen de contenidos

Primer Capítulo

En el presente capítulo se detallará una breve introducción sobre la importancia y justificación del presente proyecto, así como los antecedentes y trabajos relacionados con el mismo. Se especificará los puntos o metas a cumplir detallados en los objetivos ya definidos. Por último, se detallará los trabajos u sistemas ya desarrollados con relación al tema propuesto.

Segundo Capítulo

Muestra una recopilación de teoría e información sobre todo lo relacionado al estándar de televisión digital europeo (HbbTv) así como antecedentes en trabajos relacionados con la detección de objetos y el uso de estos en ámbitos específicos de los deportes. Se describirá el uso de lenguajes de programación como Python y diversas librerías necesarias para el correcto funcionamiento del trabajo implementado. Por último, se describirá algunas plataformas utilizadas para la comunicación entre todo el sistema.

Tercer Capítulo

Se describe la implementación y el desarrollo del sistema, las bases en las que se generó los códigos, el esquema en el que se basa el diseño de este y los diagramas

de flujo de la programación desarrollada para cada parte del sistema. Adicional se describe la forma de comunicación de todos los sistemas y las diferentes plataformas que se utilizó con sus características.

Cuarto Capítulo

Abarca la descripción de las pruebas realizadas inicialmente desde los sistemas por separados, la integración de los sistemas en un servidor local y por último la implementación final ya desarrollada en plataformas en la red.

Quinto Capítulo

Se hace una breve discusión sobre los resultados obtenidos, adicional a ello, se definen las conclusiones a partir de los objetivos planteados y se proponen trabajos que pueden mejorar o dar continuidad al presente proyecto, así como mejoras o adaptaciones que pueden llegar a finalizar con un producto más confiable.

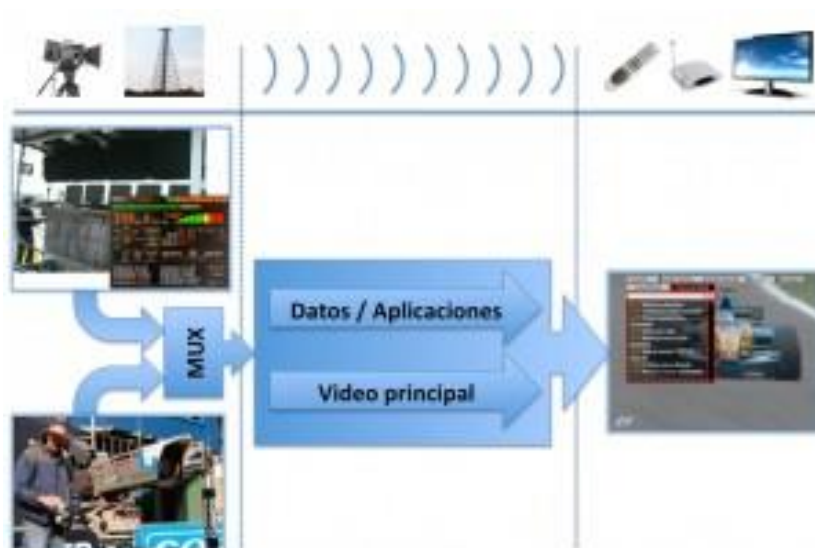
Capítulo II: Marco Teórico

Televisión digital terrestre

La Televisión Digital Terrestre (TDT) se refiere a la transmisión de tanto imágenes como sonido por medio de codificación binaria, este método permite hacer el uso de aplicaciones que permite al usuario interactuar en conjunto con la programación habitual. Es así como a través de las ondas que representa la señal digital se transmite el audio, video y datos, pues, las ondas recorren en la atmosfera, por lo que no se requiere un cable o satélite, pero si de una red de antenas repetidoras. (Muñoz, 2016). La arquitectura se observa en la Figura 1.

Figura 1

Arquitectura TDT



Nota. El gráfico muestra la arquitectura de TDT. Tomado de “*Televisión Digital Terrestre*”, (Ayala, 2014)

Por su parte, el Ministerio de Telecomunicaciones (2019) menciona que la TDT representa un nuevo mecanismo para la transmisión de televisión gratuita, donde se tiene mejor calidad en la imagen, audio y video, pues, se aprovecha el espectro, donde se liberan las frecuencias que se pueden emplear en nuevas tecnologías. ´

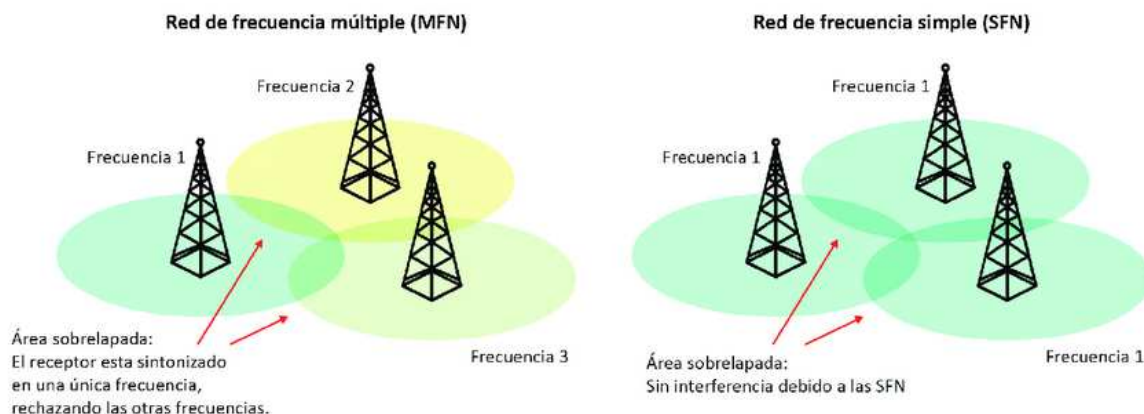
Por lo tanto, la TDT representa un mecanismo para la transmisión de la señal de televisión abierta con una mejor calidad (sonido, video e imagen), pues, utiliza ondas hercianas para transportar señales digitales y para la recepción de la señal se emplea antenas UHF convencionales. Según Millán (2014) los medios para la transmisión de TDT pueden seguir dos tipos de arquitectura como las siguientes:

- **MFN (*Multiple Frequency Network*)**

Es una arquitectura a nivel nacional con múltiples frecuencias en cada transmisor cuenta con radiofrecuencias que no necesita sincronización, incluso se efectúan desconexiones de la programación según la opinión del editor. Cabe mencionar que se requieren mayor cantidad de frecuencias para el funcionamiento de la red.

- **SFN (*Single Frequency Network*)**

Este funciona a nivel provincial o local, puesto que las redes de frecuencia y los transmisores de cobertura están en similar frecuencia al igual que las emisiones se modulan con la misma señal, por lo que se deben sincronizar a los transmisores. Aspectos que ayudan al mayor aprovechamiento del espectro y los ecos generados en la transmisión ocasionan ganancias en la recepción de señal. Esto se aprecia en la Figura 2

Figura 2*Frecuencia de transmisión MFN-SFN*

Nota. El gráfico muestra la frecuencia de transmisión. Tomado de “*Televisión Digital Terrestre*”, (Millán, 2014)

En cuanto a la recepción de la señal se requieren cambiar las antenas y se deben añadir un amplificador para potenciar y recibir la señal de forma adecuada. No obstante, en edificaciones antiguas se deben cambiar la instalación para adaptar a la TDT. Mientras que la recepción en edificaciones unifamiliares debe adquirir un decodificador para la conexión al televisor analógico o que cuente con integración del decodificador para la visualización de canales digitales y analógicos. El decodificador permite la recepción de la señal de Tv, servicios interactivos, entre otros.

Estándares de la Televisión Digital Terrestre

A continuación, según la Organización de Telecomunicaciones de Iberoamérica (2016) se detallan los estándares de la TDT:

- Estándar norteamericano ATSC (*Advanced Television Systems Committee*): Es un estándar que privilegia la alta definición sobre la transmisión de datos,

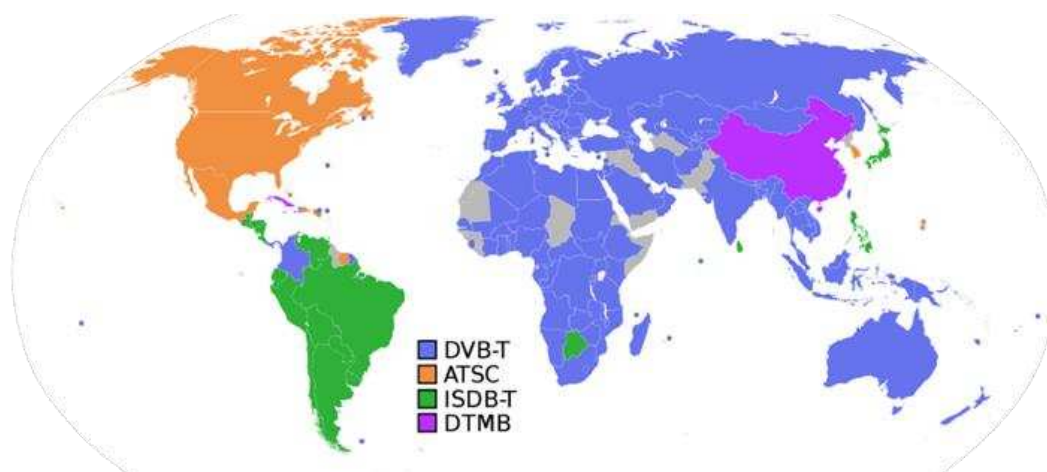
utiliza 6 MHz para la transmisión de la señal, la comprensión del video es bajo MPEG-2, audio Dolby AC-3 y modulación con E8-VSD (Azarcoya, 2019).

- Estándar europeo DVB (*Digital Video Broadcasting*): Permite administrar los estándares digitales para la transmisión de televisión por satélite, cable y terrestre. Aplica 8 MHz para la transmisión de la señal, codificación de audio y video en MPEG-2, opción para video AVC (Jácome, et al., 2018).
- Estándar japonés ISDB (*Integrated Services Digital Broadcasting*): Se denomina Transmisión Digital de Servicios Integrados, este se enfoca en transmitir señales de radio y televisión digital. Entre sus componentes se encuentra ISDB-Tb (Jácome, et al., 2018).
- Estándar ISDB-Tb (*Integrated Services Digital. Broadcasting – Terrestrial Brazilian*): Basado en el ISDB: Para la comprensión del video emplea MPEG-4, lo cual ayuda a cargar una gran cantidad de datos, mejor calidad de imagen e interactividad debido a que emplea Middleware Ginga. Además, transmite múltiples programas a través de un mismo canal de difusión. Por lo que emplea *Transport Stream* para la programación (Aguirre, 2018).
- Estándar de la República popular de China DTMB (*Digital Terrestrial Multimedia Broadcast*): La codificación para audio y video aplica MPEG-2 y MPEG-4, modulación con TDS-OFDM, alta definición E8 – VSB, empleado para redes múltiple (MFN) y la red de frecuencia única (SNF) Ver la Figura 2 (Montesdeoca, 2016).

En la Figura 3 se presenta un mapa de difusión de los sistemas de TDT a nivel mundial:

Figura 3

Difusión de sistemas de TDT en el mundo



Nota. El gráfico muestra la difusión de los sistemas TDT a nivel mundial. Tomado de “*Televisión Digital Terrestre*”, (Organización de Telecomunicaciones de Iberoamérica, 2016)

En el caso de Ecuador y la mayor parte de países Sudamericanos el estándar que se usa es el ISDB-Tb (Integrated Service Digital Broadcasting - Terrestrial) (Intel, 2017). Además, en el Plan Maestro de transición a la televisión digital terrestre se establece estrategias para la adopción del estándar ISDB-Tb con la finalidad de aumentar la cobertura del servicio de televisión a nivel nacional e implementación de soluciones adecuadas. Además, se detalla las condiciones para la implementación como el modo de transmisión (*multicast y broadcast*), asignación de canales, enlaces auxiliares, interactividad (*one seg gap fillers*, canalización, entre otros (Intel, 2018).

Beneficios de la TDT

A continuación, se exponen los beneficios de la TDT:

- Uso óptimo del espectro debido a que aplica mecanismos para codificación.
- Utiliza la red SFN para la transmisión en un solo frecuencia.
- Mayor robustez para evitar interferencias.
- Soporte múltiple para audio con una resolución HD (720 – 1080).
- Se puede visualizar imágenes en tamaño 16:9.
- Permite cubrir varias áreas para la transmisión de la señal de televisión digital.
- La señal puede ser receptada mediante terminales móviles y portátiles.
- Permite el uso de aplicaciones para la interactividad con los usuarios.
- Ofrece variedad de canales de televisión.
- Nivel mínimo de consumo de frecuencias.
- Posibilita la recepción de alertas de emergencia.
- Permite un adecuado aprovechamiento del ancho de banda.
- Señal de calidad.
- Posibilidad de acceso a la información y contenido interactivo (Intel, 2017).

HBBTV (Hybrid Broadcast Broadband TV)

Definición

La transmisión híbrida de televisión de banda ancha (*Hybrid Broadcast Broadband TV - HbbTv*) es una iniciativa global dirigida en la armonización de los estándares para la entrega de contenido al usuario final, utilizando al mismo tiempo la

radiodifusión (*broadcast*) y red de banda ancha (banda ancha) a través de los dispositivos de TV vinculados (TV conectada, Smart TV y Set -Top-Boxes) (Jaksic, et al., 2017).

En realidad, HbbTv es una tecnología que combina los servicios de radiodifusión con los servicios ofrecidos a través de la Internet, todo esto en una sola pantalla. A diferencia de la televisión tradicional, los usuarios pueden usar aplicaciones interactivas para entretenimiento o para obtener información.

Por lo tanto, es una tecnología desarrollada para maximizar el impacto que se dispone de la TV mediante la digitalización de la señal para una experiencia más interactiva con el usuario. Aunque esta tecnología se popularizó en España después de algunos proyectos como “Botón rojo RTVE”, el cual fue desarrollado como una TV híbrida. Fue una iniciativa europea que busca combinar las emisiones en directo de la televisión (Broadcast) con los beneficios y contenido de la web (*Broadband*) (Fallahi, Joki, & Teirikangas, 2019).

Los servicios que se están desarrollando para este estándar son los siguientes:

- Recuperación de programas de televisión bajo demanda
- Publicidad interactiva
- Juegos
- Aplicaciones interactivas
- Navegación WEB
- Información personalizada en el televisor
- Servicios personalizados por cada programación (teletexto, subtítulos, guía electrónica de programación) (Fallahi, Joki, & Teirikangas, 2019).

Cabe mencionar que hasta el año 2018, este estándar se implementó en 27 países, 26 emisoras y operadores, llegando a casi 587 millones de personas en todo el mundo. Incluso casi 60 empresas están desarrollando, comercializando servicios y aplicaciones de HbbTV (Fallahi, Joki, & Teirikangas, 2019).

Según Llorente (2018) las funciones de este estándar se exponen de la siguiente manera:

- Escenarios para la utilización de segundas pantallas.
- Permite sincronizar contenidos entre las segundas pantallas y los terminales del estándar.
- Posibilidad de contar con personalización de contenido (audio, video, accesibilidad y publicidad).
- Se codifica el video HEVC (*High Efficiency Video Coding*) para el contenido UDH que se envía por medio de broadband.
- Se puede realizar streaming adaptativo a través de MPEG-DASH.
- Permite establecer conexiones entre aplicaciones mediante el uso de WebSocket.
- La sincronización de contenido en diversos dispositivos se efectúa mediante MPEG TEMI y DVB-CSS.
- Con el protocolo DIAL se puede identificar los dispositivos de una red doméstica y crear aplicaciones en otros dispositivos.
- Utiliza como lenguaje CSS, JavaScript y HTML5.
- Permite contar con subtítulos en otros idiomas.
- Contiene contador digital para gestión electrónica.

Uso se supertexto con interfaz atractivo con información (noticias, meteorología, tráfico, resultados deportivos, bolsa, etc.).

- Posibilidad de contar con estadísticas o información adicional de los programas de televisión.
- Se puede participar en programas de televisión mediante votos.
- Accesos directos a más de 15.000 canales de todo el mundo.
- Visualización en diferido de los programas de televisión, restauración de programas, video on demand.
- Juegos sencillos.
- Home shopping, cursos y teleeducación.

Arquitectura

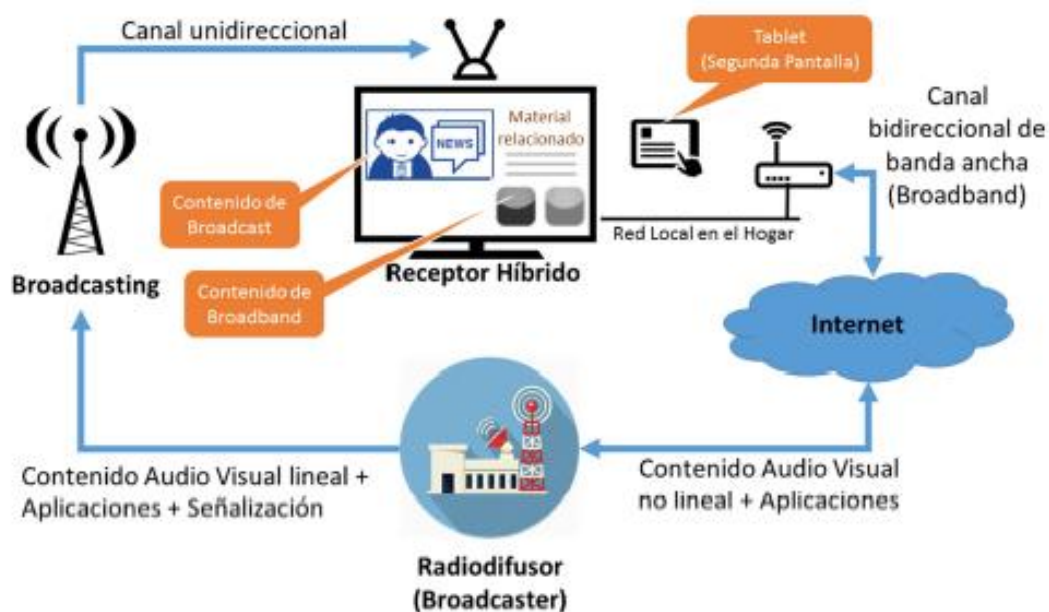
Como se menciona, se trata de un terminal híbrido, el cual tiene la capacidad de conectarse a dos redes paralelamente. La primera red se trata de la transmisión de televisión digital, en este caso el estándar europeo se rige en DVB (*Digital Video Broadcast*) ramificado en terrestre, satélite y cable (Jaksic, et al., 2017).

El estándar HbbTv emplea dos tipos de redes: transmisión y banda ancha, ambos para enviar datos y aplicaciones en dispositivos híbridos (terminales) como televisores – decodificadores conectados a Internet, Smart TV, NetTV, computadores con sintonizadores y dispositivos móviles que tienen receptores de emisión. El terminal híbrido puede conectarse a las dos redes de forma paralela, es decir, el uno mediante difusión de la red y recepta la transmisión sincronizando entre la imagen y el sonido con los datos. La otra conexión se realiza por Internet a través de una interfaz de banda ancha, lo cual permite receptar aplicaciones del proveedor, incluso a través de la

interfaz se puede conectar a otros terminales de HbbTv en la misma red de área local (Fallahi, Joki, & Teirikangas, 2019). La arquitectura del estándar se observa en la Figura 4.

Figura 4

Arquitectura HbbTv



Nota. El gráfico muestra la arquitectura del estándar HBBTV. Tomado de “*Characteristics of hybrid broadcast broadband television (HbbTv)*”, (Jaksic, et al., 2017)

Especificaciones

Este estándar fue desarrollado y publicado por el Instituto Europeo de Normalización de las Telecomunicaciones Instituto Europeo de Normalización de las Telecomunicaciones (ETSI) en febrero del 2009. En junio del 2014, la asociación se vinculó con la organización de ser servicios de televisión por protocolo de Internet (IPTV), quienes colaboraron para establecer las especificaciones de la televisión y

decodificadores que se conectan a la red. A continuación, se detalla las especificaciones de la red

Tabla 1

Especificaciones estándar HbbTv

Especificaciones	Aprobación ETSI	Año
HbbTV 1.0	TS 102 796 v1.1.1	Junio – 2010
HbbTV 1.5	TS 102 796 v1.2.1	Noviembre – 2012
HbbTV 2.0	TS 102 796 v1.3.1	Noviembre – 2015

Nota. La tabla muestra las especificaciones del estándar HBBTV. Tomado de “*Characteristics of hybrid broadcast broadband television (HBBTV)*”, (Jaksic, et al., 2017)

La versión 1.0 fue aprobada en junio del 2010 como ETS TS 796. La versión 1.5 se aprobó en noviembre del 2012, esta ofrece una variedad de servicios de radiodifusión con soporte para streaming adaptativo HTTP asado en la especificación MPEG-DASH que mejora la calidad de vídeo en conexiones de Internet demasiado cargadas o lentas, mejorando la información del programa de televisión proporcionado por los proveedores mediante EPG (Haro, 2019). La versión 2.0 se aprobó en noviembre del 2015, este servicio de televisión interactivo híbrido. Los usuarios pueden disfrutar los siguientes servicios:

- Uso de aplicaciones avanzadas mediante HTML5.
- Posibilidad de ver contenidos mediante diferentes dispositivos (PC, Tablet, celulares inteligentes).
- Información detallada del programa, votaciones, juegos y etc.

- El contenido entregado es Ultra HD a través de la compresión en HEVC.
- Dispone de un soporte para la traducción en varios idiomas.
- Mayor privacidad para el consumidor.
- Se puede acceder a los contenidos grabados en el disco duro del receptor.
- Utiliza tecnologías HTML5, DVB CI Plus 1.4, vídeo HEVC y subtítulos TTML (*Timed Text Markup Language*).
- Mejora la compatibilidad con las tecnologías existentes como MPEG DASH, DSM-CC (*Digital Storage Storage Media Command and Control*) y la sincronización en el televisor.
- La versión 1.0 y 1.5 continúa funcionando en los dispositivos HbbTv 2.0.

Tecnologías utilizadas

A continuación, se describe las tecnologías utilizadas por el estándar HbbTV:

HTML5:

- Contiene etiqueta para integración de contenido multimedia.
- Uso de etiquetas de audio con un controlador de las pistas como flujo alternativo.
- Dispone de elemento de etiqueta para los subtítulos y composición de imágenes (Alvarez, Galeano, & Peña, 2017).

Protocolo DIAL:

- Se utiliza para el reconocimiento mutuo de los dispositivos.

- Ayuda a que las segundas pantallas identifiquen los terminales del estándar.
- Aplica modelo cliente DIAL – servidor DIAL.
- Tiene dos servicios (*DIAL service Discovery / DIAL REST service*).
- *DIAL service Discovery* permite que identificar los dispositivos con servidores en una red y acceder a DIAL REST, utilizando el protocolo SSDP (*Simple Service Discovery Protocol*) y protocolo UDP (*Use Datagram Protocol*).
- DIAL REST service permite realizar consultas y detener aplicaciones a través de HTTP.

WebSocket:

- Permite realizar emparejamiento de las aplicaciones.
- Posibilidad una comunicación bidireccional en el protocolo TCP (*Transmission Control Protocol*).
- Permite receptor y enviar datos de manera simultánea, así como intercambio de información.
- Se implementa en servidores y navegadores web.
- Tiene una API para utilizar el protocolo en apps web (Rodas & Valencia, 2018).

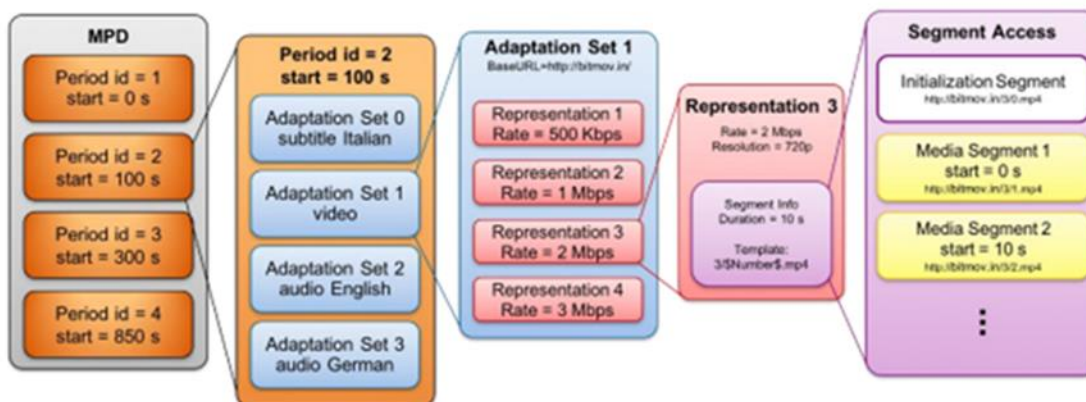
MPEG-DASH:

- Posibilita distribuir el contenido de broadband del estándar HbbTV.
- Tiene reproductor DASH con código abierto, esta es una librería de JavaScript (dash.js) que ayuda a la reproducción de audio y video.

- El cliente DASH posee varios algoritmos para lograr una calidad óptima del contenido.
- Esta tecnología permite crear fragmentos multimedia que se codifican y guardan en un servidor web HTTP.
- Utiliza el fichero MPD para detallar la estructura del contenido audiovisual y establece las direcciones para que el cliente acceda a los segmentos almacenados.
- El fichero MPD está formado por periodos (tiempo inicio – duración del contenido), adaptation (conjunto de flujos para video – audio), representaciones (adaptación con igual contenido codificado y diferente parámetro) y segmentos (partes divididos del contenido multimedia) (Nakachi, 2017).

Figura 5

Arquitectura del fichero MPD en MPEG-DASH



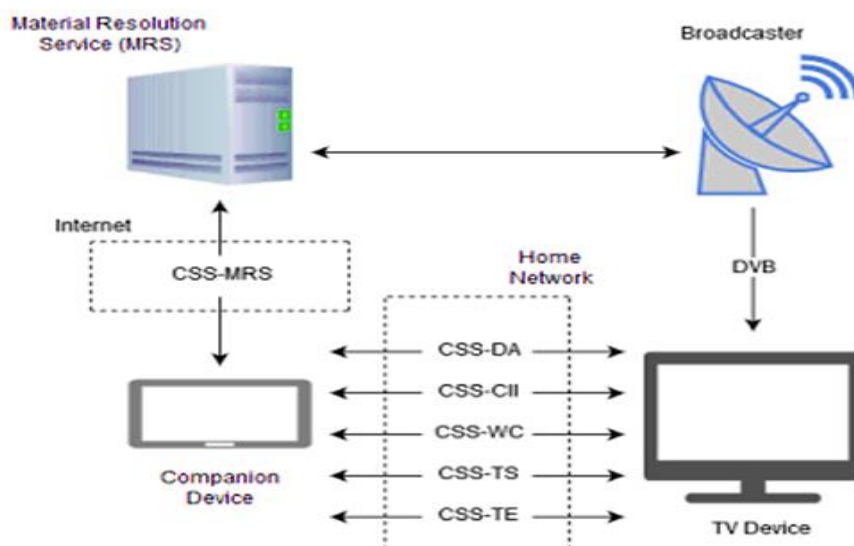
Nota. El gráfico muestra la arquitectura del fichero MPD en MPEG-DASH. Tomado de “MPEG Media Transport technologies and its application to immersive media communication services”, (Nakachi, 2017).

DVB-CSS:

- Permite definir los protocolos para la sincronización de la presentación multimedia entre un televisor y los dispositivos complementarios como celulares, tablets, entre otros.
- Se incluye dentro de la especificación HbbTV para los televisores conectados.
- Se utiliza como parte de las pantallas complementarias.

Figura 6

Arquitectura protocolo DVB-CSS



Nota. El gráfico muestra la arquitectura del protocolo DVB-CSS. Tomado de “*Desarrollo de servicios OTTsíncronos mediante el estándar HBBTV 2.0.1 para TV digital*”, (Llorente, 2018)

Herramientas

Python

En un lenguaje de programación enfocada a objetos para el desarrollo de aplicaciones web y móvil. Se puede aplicar en el desarrollo rápido de aplicaciones debido a que presenta dinámica, disminuye el costo de mantenimiento, soporta módulos y paquetes. Además, el código se puede reutilizar en diversos proyectos, la librería es gratuita y tiene la opción multiplataforma. Los módulos del código se pueden intercambiar y no transforma el código en lenguaje de máquina sino en “código byte), por ende, necesita de un intérprete (Máquina Virtual Python) (West, 2018).

Plataforma Digital Ocean

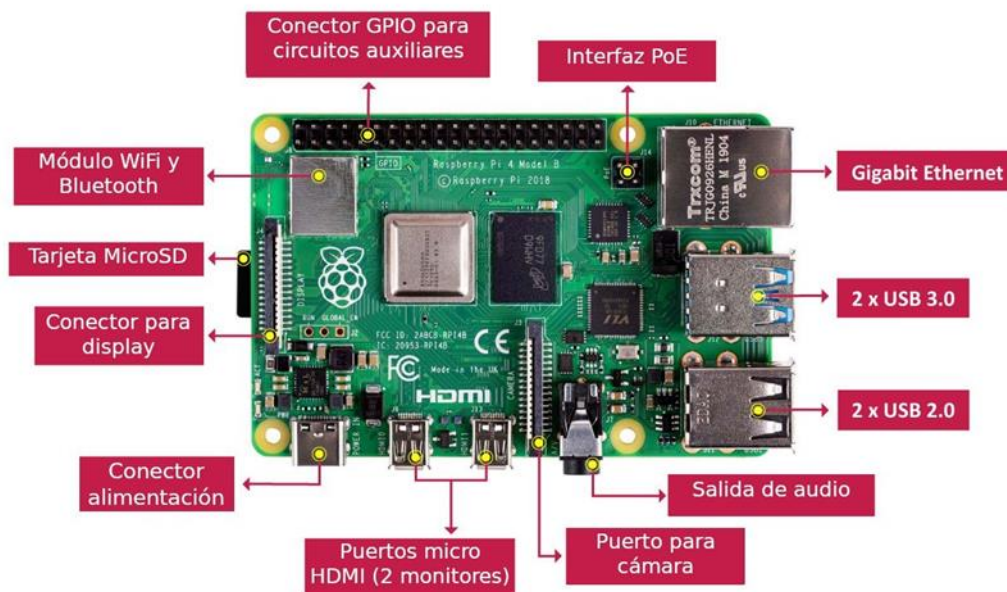
Esta plataforma ofrece servicios de computación en la nube a precios que varían dependiendo el uso de estos, oferta servidores accesibles desde planes básicos hasta avanzados con una IP publica capaz de ser manejados por desarrolladores e implementarse en línea. (DigitalOcean, 2022)

Raspberry Pi

Es un ordenador de un coste bajo utilizado para el desarrollo de prototipos y facilitar acceso a los usuarios; empleado en sistemas operativos como GNU/Linux (Caballero, et al., 2020). Los elementos se muestran a continuación en la Figura 7

Figura 7

Elementos de Raspberry Pi



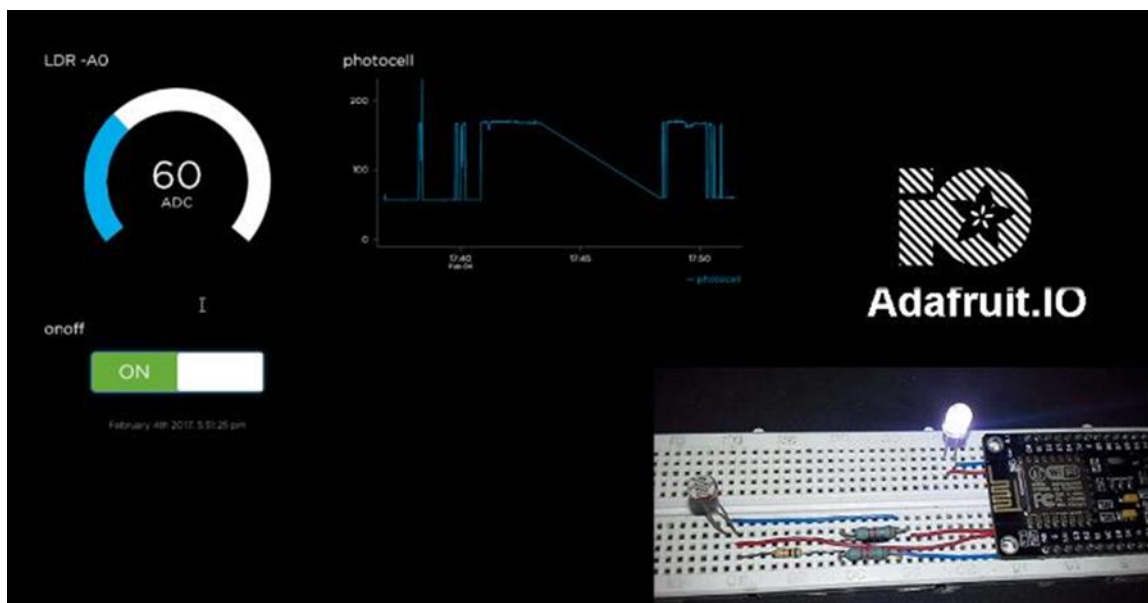
Nota. El gráfico muestra los elementos de Raspberry Pi. Tomado de “*Raspberry Pi, conectividad y programación mediante puertos GPIO*”, (Caballero, et al., 2020).

Plataforma IoT Adafruit

Esta plataforma open - source permite el diseño de aplicaciones relacionadas con Internet de las cosas (IoT), para su creación se considera las plataformas como Raspberry Pi, Arduino, entre otros. La API empleado para la comunicación se relaciona con cliente MQTT y servidores Adafruit.IO. Con esta plataforma se puede diseñar dashboard de calidad, conectar el proyecto a la web, visualización de datos en tiempo real, tratamiento de múltiples datos, programación de eventos de correo, conexión a RSS feeds, Twitter, entre otros (Martínez, 2019). Esto se observa en la Figura 8.

Figura 8

Plataforma IoT Adafruit



Nota. El gráfico muestra la pantalla de la plataforma IoT Adafruit. Tomado de “*Diseño e implementación de un sistema de alarma IoT basada en tecnologías Open Source*”, (Martínez, 2019).

FFmpeg

Se trata de un conjunto de bibliotecas y herramientas utilizadas para el procesamiento de contenido multimedia (audio, video, metadatos y subtítulos). Tiene licencia LGPL bajo GPL. La biblioteca se enfoca en códecs, transmitir protocolos, descompresores, conversión de color, rutinas de mezcla, entre otros. Las herramientas se relacionan con comando para convertir – transmitir contenido multimedia, análisis simple y reproductor (GitHub Inc., 2017).

Capítulo III: Diseño E Implementación

En el presente trabajo pretende realizar un sistema asistido para obtener la posición de un balón durante un encuentro deportivo de fútbol para el uso de estos datos en aplicativos o dispositivos accesibles a la interactividad y la inclusión de personas con discapacidades a la interacción con este deporte como espectadores. Esto se desarrolló mediante programación Python, con el uso de una plataforma de servidores *Digital Ocean* y la plataforma IoT *Adafruit*, así se logra llevar al sistema a un ámbito global y no local para en un futuro se realice el desarrollo de aplicativos que puedan interactuar con este y poder incluir a más personas con discapacidades y simplemente hacer una experiencia mejor para los espectadores del partido. La información viaja a través de estas plataformas por medio de web sockets y llega al usuario por medio de la plataforma IoT y a través de la Tv digital se puede informar o interactuar conjunto con la transmisión del encuentro deportivo y el dispositivo a usarse y al ser implementado esto en la nube, se vuelve versátil para la conexión a distancia.

Como dispositivo de pruebas se usará un guante háptico desarrollado para el uso de personas con discapacidad visual, este proyecto solo se puede usar en repeticiones de partidos puesto que se debe obtener de forma manual los vectores de posición para ser enviados al guante y una vez obtenida la matriz, se puede poner en funcionamiento el dispositivo. Con este proyecto se pretende automatizar dicho dispositivo y crear un camino al desarrollo de más dispositivos que puedan vincularse con este sistema de posicionamiento asistido.

Herramientas de desarrollo

Para el desarrollo del presente trabajo se hizo la utilización de:

- PC o Laptop (Windows 10 – Windows 11)
- Tableta gráfica INTUOS
- Software Visual Studio Code y Python versión 3.10
- Plataforma IoT Adafruit (Versión gratuita)
- Plataforma Digital Ocean (Servidor Virtual Linux)

Instalación y configuración del software

Para la instalación del software se debe verificar en la PC tanto como en el sistema Linux y Windows donde se vaya a trabajar el operador, que se cuente con las características recomendadas para que pueda soportar y trabajar con normalidad.

Para a descarga del software Python se lo puede adquirir fácilmente de su página web www.python.org/downloads/. Allí se puede revisar cuales son las versiones disponibles, lo más recomendable es descargar la última versión ya que cuenta con la mayoría de librerías disponibles y actualizadas que pueden facilitar el desarrollo de este, de igual forma para descargarlo en un sistema basado en Linux simplemente se utilizan los comandos de instalación “`sudo apt-get install python(version)`” y para usarlo se genera el código en un archivo con la programación en cualquier editor de texto (esto para Linux) y se lo guarda al mismo con la extensión de Python (.py) (Python , 2022)

En el sistema operativo de Windows, para un desarrollo más sencillo en caso de ser un programador primerizo, se recomienda descargar los complementos de Python o programadores auxiliares basados en Python, los cuales tienen funciones de corrección y detección de errores en forma automática tales como *Anaconda* con su complemento *Spyder* o *Visual Studio Code*.

La plataforma más usada que adapta estas versiones es *Visual Studio Code*, el mismo es un conjunto de compiladores de varios lenguajes de programación, posee detección de errores en el código de programación y permite abrir varios programas simultáneos en caso de requerirlos. Otra plataforma es *Anaconda*, el cual posee varios subprogramas que se pueden usar dependiendo la complejidad del programa como *Jupyter* que se puede usar para crear archivo desde aprendizaje y pruebas de código hasta programación basada en aprendizaje de máquina con sus respectivas subdivisiones; o de la misma forma *Spyder* el cual posee un IDE de programación similar a Python y posee la detección automática de errores de código. (Microsoft, 2022)

Ya con el software instalado, se debe tener en cuenta que, en el desarrollo del proyecto, fueron requeridas varias librerías específicas que no vienen descargadas por defecto, las librerías a instalar son "*Pillow*" para el uso de la interfaz gráfica y obtener los datos de la posición del ratón en pixeles el cual se usará para la generación de vectores. Como secundaria se requiere la librería "*Adafruit Blinka*" o "*Adafruit-IO*" para la conectividad con el servidor IoT. Y por último para la adaptación de la programación de TV Digital se debe usar la librería "*Flask*" la misma permite interactuar entre los dos lenguajes de programación como son Python y HTML.

Para la instalación del controlador para la tarjeta gráfica es recomendable revisar las especificaciones técnicas necesarias ya que al instalarla en los sistemas basados en Linux puede traer complicaciones con la capacidad de memoria o problemas directamente con el controlador. Para la instalación en los sistemas de Windows se puede adquirir los controladores de dos sitios, por lo general el controlador llega con un CD en el equipo, y en caso de no tenerlo se lo puede descargar de la página oficial de

WACOM donde se lo buscaría por el modelo de la tarjeta gráfica a utilizar. (Wacom, 2022)

Esquema general del proyecto

El proceso que sigue el proyecto se puede subdividir en 3 partes, las dos primeras desarrolladas para el correcto funcionamiento del sistema y la tercera va orientada a las funcionalidades que el usuario quiera darle a la información proporcionada, ya sea para conectar dispositivos de interpretación de datos, almacenamiento de datos o simplemente gadgets ya sea de ayuda para personas con alguna discapacidad, siempre y cuando sean compatibles con la información que proporciona el servidor IoT o puedan obtener una forma de comunicación con el mismo.

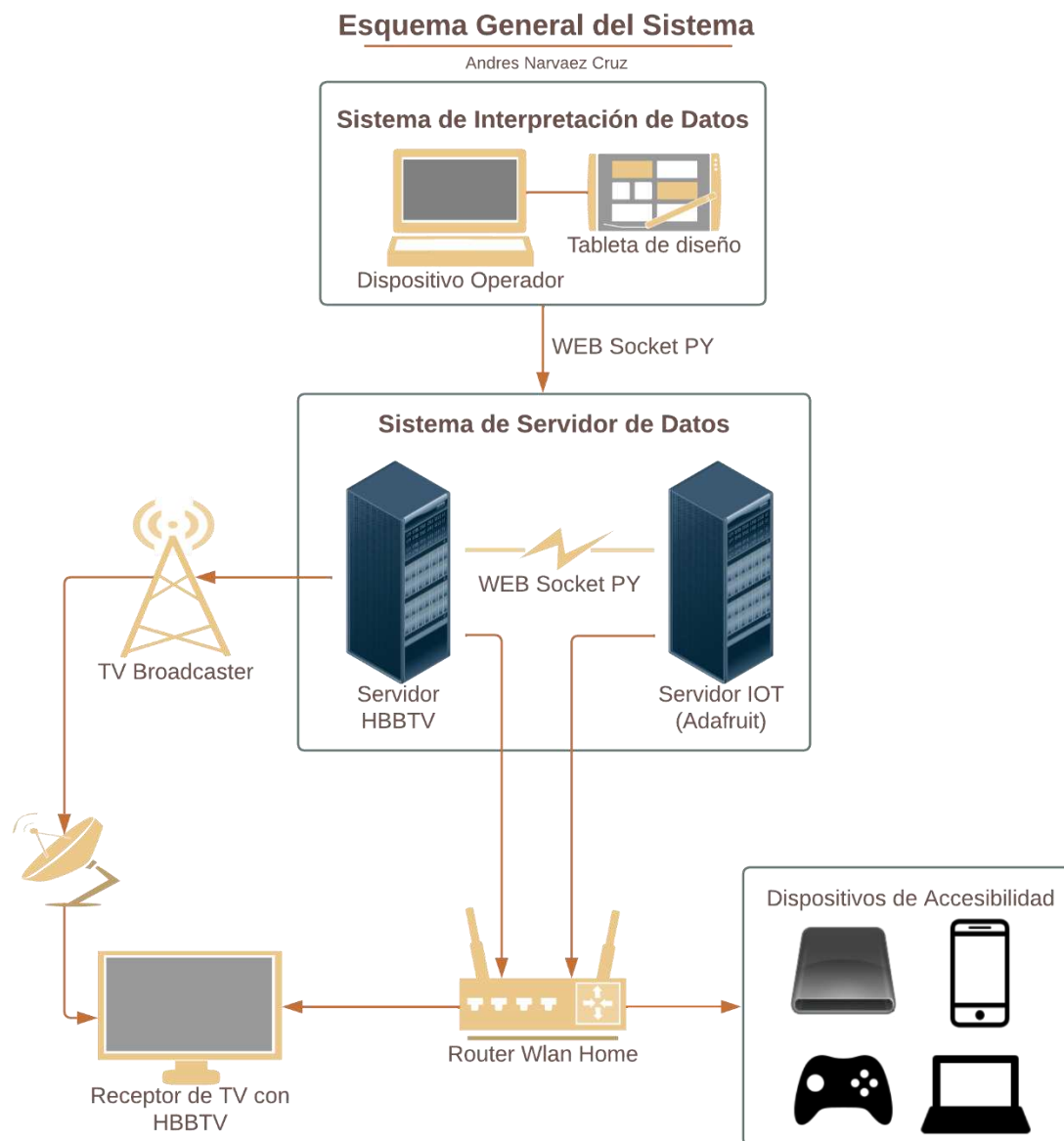
Como primera parte se tiene al sistema de interpretación de datos, este viene a ser manejado por un operador que pueda encontrarse en el estadio como espectador o puede encontrarse frente a una TV observando el encuentro deportivo de fútbol, en los dos casos el operador necesita una conexión estable a internet para que los datos puedan enviarse sin complicaciones y de esta forma evitar pérdidas de datos. Como diagramas dentro de este sistema se tiene el dispositivo del operador, este va a ser expresado como tal por una computadora, el cual tendrá el programa de interpretación de datos que se basa en lenguaje Python. Adicional se complementa con una tabla digitalizadora la cual se utiliza para codificar la posición al balón en la pantalla y transformar esa información a un vector que será enviado como dato. También se experimentó compilando al programa en una tarjeta de desarrollo (Raspberry Pi3) para hacer un modelo más compacto del dispositivo, pero el mismo requiere de una pantalla, teclado y ratón para poder funcionar completamente lo cual limita el modelo compacto y

hace de una forma más complementaria usar como dispositivo a una computadora portátil. Ver Figura 9

Como segunda parte tenemos al Sistema de Servidor de datos, el cual viene dado por dos servidores los cuales se encuentran en la nube, el primero es un servidor dedicado contratado en la plataforma Digital Ocean, este interactúa entre el emulador de TV Digital y al mismo tiempo recibe constantemente los datos generados por el sistema anterior, transforma los mismos y coteja la información, almacena toda la cadena de datos en una base para en caso futuro ser usado, este a su vez analiza el dato enviado por el primer sistema y obtiene el vector codificado de la posición del balón el cual es enviado al servidor IoT por medio de programación bajo Web Sockets. El segundo servidor está sobre una plataforma IoT ya desarrollada e implementada en la nube (Adafruit) que permite interactuar con varios dispositivos o aplicativos remotamente. Utiliza *Feeds*, estos son flujos de contenido por los que la plataforma puede desplazar la información, estos son programados para visualizar en la plataforma los datos y relacionarse con diferentes dispositivos que puedan interpretar o vincularse para el uso de estos datos ya sea solo para almacenarlos o trabajar con actuadores. (A Minority and Woman-owned Business Enterprise (M/WBE), 2022).

Como tercera parte se interpreta como el usuario. Este depende del aplicativo que se use ya que tiene acceso a los datos en tiempo real de las posiciones del balón, para este proyecto será utilizado un guante háptico desarrollado para personas con discapacidad visual.

Figura 9

Esquema general del sistema

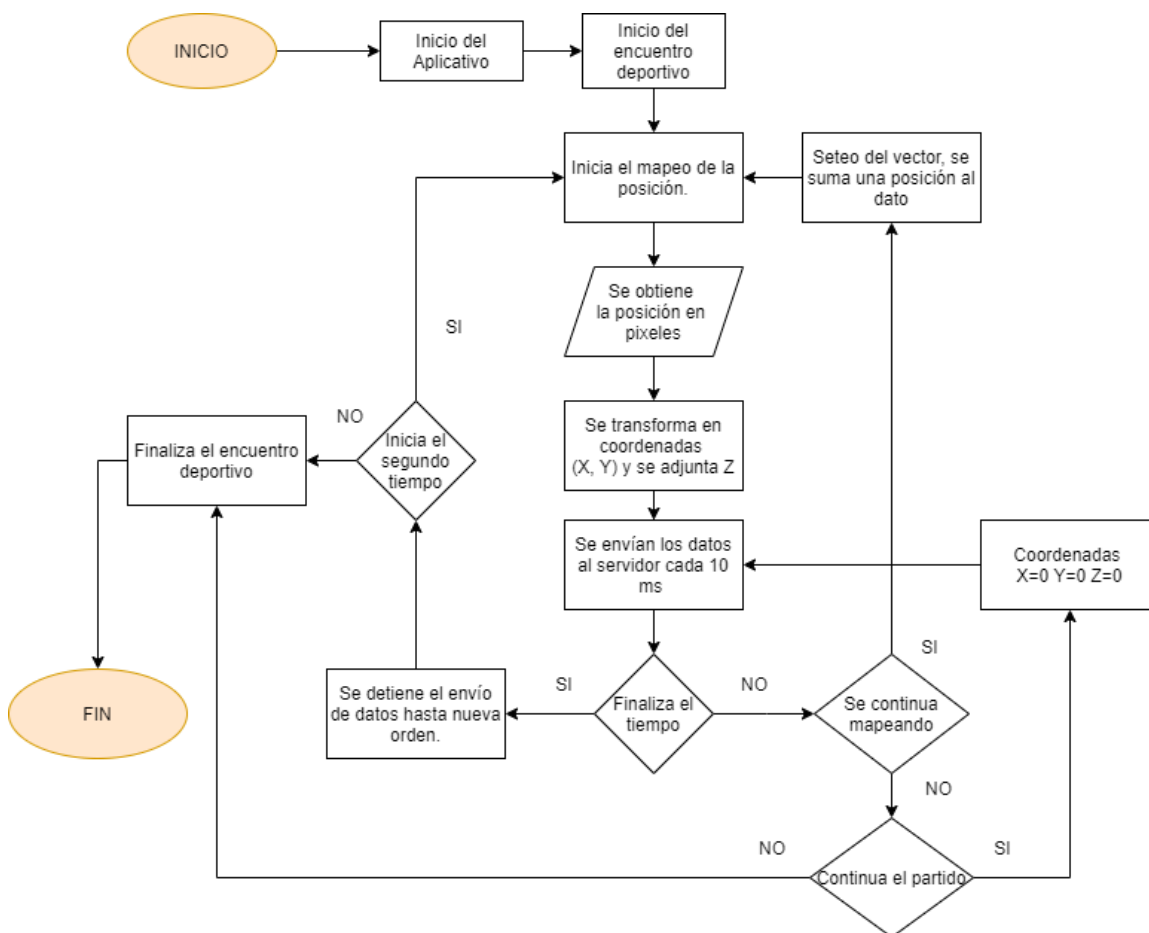
Nota: La figura representa el esquema según el prototipo del sistema asistido.

Sistema de interpretación de datos

Este sistema consiste en interpretar la posición del cursor en la pantalla la cual será interpretada por la Figura 11 donde este simulará ser el balón en la cancha, obtener la posición del píxel en el que se ubica el puntero y de acuerdo con las limitaciones transformarlo a un vector posición el cual va a ser nuestro dato principal para el sistema. La programación viene dada de acuerdo con la Figura 10.

Figura 10

Diagrama de flujo del sistema de interpretación de datos.

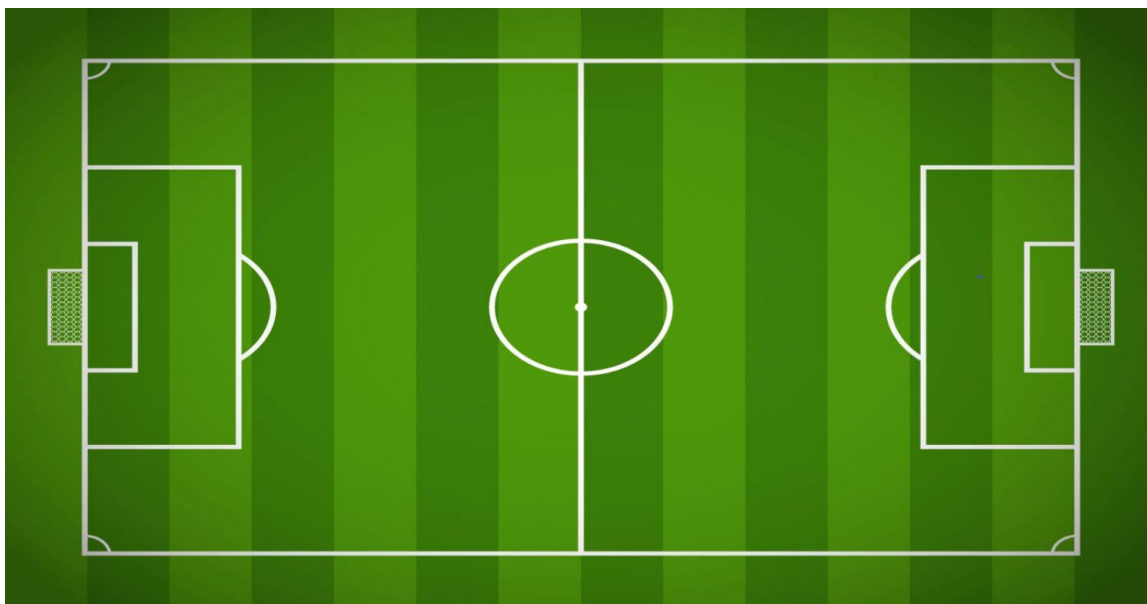


Nota: El diagrama representa el modelo base para el desarrollo de la programación del sistema.

Para el inicio del programa se basó en implementar la interfaz gráfica bajo un modelo amigable o aplicación sencilla de usar para el operador. Para esto se implementó bajo la librería *Pillow* la cual permite realizar procesamiento de imágenes, ya sea implementando en una imagen en un aplicativo o realizar las operaciones bajo los pixeles de la pantalla.

Figura 11

Imagen de campo de fútbol implementado



Nota: La imagen del campo de futbol fue utilizado de fondo para referenciar el campo de futbol y ser más amigable para el operador, Tomado de *Pinterest* (Pargeter, 2019)

En la programación del sistema de interpretación de datos se basó bajo algunas librerías las cuales son de utilidad variada tanto en la presentación del sistema como en la obtención y envío de datos. Los mismos se presentan bajo la Tabla 2

Tabla 2

Librerías utilizadas en el sistema asistido.

Librería	Especificación
Tkinter	Interfaces gráficas de usuario con TK
PIL	Python Imaging Library
Time	Librería para uso de Fecha y Hora
Socket	Interfaz de red de bajo nivel

Nota. La tabla muestra las librerías usadas en el primer sistema. Tomado de “*Python Documentation*”, (Python, 2019)

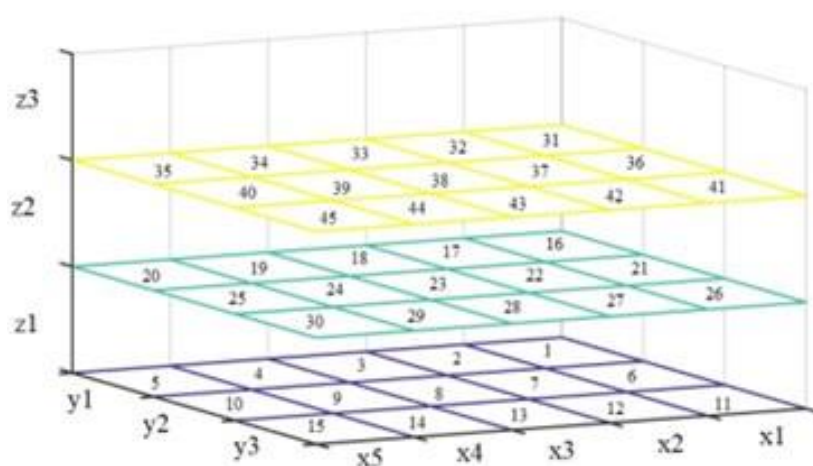
En el desarrollo de la programación se empieza integrando la imagen de fondo la misma representada por la Figura 11, de esta se deben obtener dos valores importantes los cuales son el ancho y largo de la imagen. Estos datos pueden ser obtenidos de las propiedades de la imagen o ya implementando las librerías de procesamiento de imagen para obtener los puntos exactos que delimitan únicamente la cancha de futbol.

Una vez obtenidos los puntos específicos, se utiliza la librería PIL con la cual se va a trazar las delimitaciones para dividir a todo el campo de futbol en menos secciones, en este caso se delimitó todo el campo por 15 secciones en un plano de dos dimensiones, ver la Figura 12, tomando en cuenta la referencia en plano cartesiano, se tiene 3 delimitaciones en el eje de las ordenadas y 5 delimitaciones en el eje de las abscisas, esto ya que se procede a trabajar con un dispositivo para personas con discapacidad visual tipo guante que simula un cuadrante en el eje de las abscisas en cada dedo de la mano derecha y viene acompañado con tres tipos de vibraciones diferentes para simular cada cuadrante en el eje de las ordenadas, ver la Figura 26. Adicional se implementó un tercer vector para aplicarlo de una forma tridimensional,

este viene a ser representado por 3 divisiones más dando un total de 45 subdivisiones. Esto se encuentra representado en la Figura 12. De la misma forma se toma como referencia a la primera posición empezando desde el recuadro superior derecho.

Figura 12

Referencia de las subdivisiones del campo de futbol.



Nota: El esquema representa las subdivisiones tomadas como referencia para cada segmento en el campo de futbol, Tomado de “*Sistema asistido de posicionamiento de un balón en partidos de fútbol con un servidor HbbTv integrado a un guante háptico de TV accesible para personas con deficiencia visual.*” (Villamarin & Mendez, Haptic Glove TV Device for People with Visual Impairment, 2021)

Para realizar la conversión de los valores se toma como referencia la posición del puntero en la pantalla, este viene dado por una posición en píxeles a lo ancho y largo. Con las librerías establecidas se puede obtener ese dato, para realizar el programa óptimo se toma como referencia el tamaño de la figura de fondo y se realiza por medio de operaciones y comparaciones. Transforma el dato píxel en una referencia $[X, Y]$ de acuerdo con la Tabla 3. Para la implementación del punto Z ya que no se

puede representar en un plano bidimensional, se optó por utilizar la pulsación de teclas las cuales viene a ser: tecla Z: $Z = 0$, tecla X $Z = 1$, tecla C $Z = 2$. Representando estos valores en el vector Z se puede decir que el valor 0 viene a ser una posición del balón al ras del piso, el valor 1 viene a ser una posición de altura media del balón sobre el nivel de la cancha de fútbol y el valor 2 viene a ser una posición de altura elevada del balón sobre el nivel de la cancha de fútbol.

Tabla 3

Codificación del vector posición en el plano bidimensional

Vector X	Vector Y	Código representativo
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
1	2	6
2	2	7
3	2	8
4	2	9
5	2	10
1	3	11
2	3	12
3	3	13
4	3	14
5	3	15

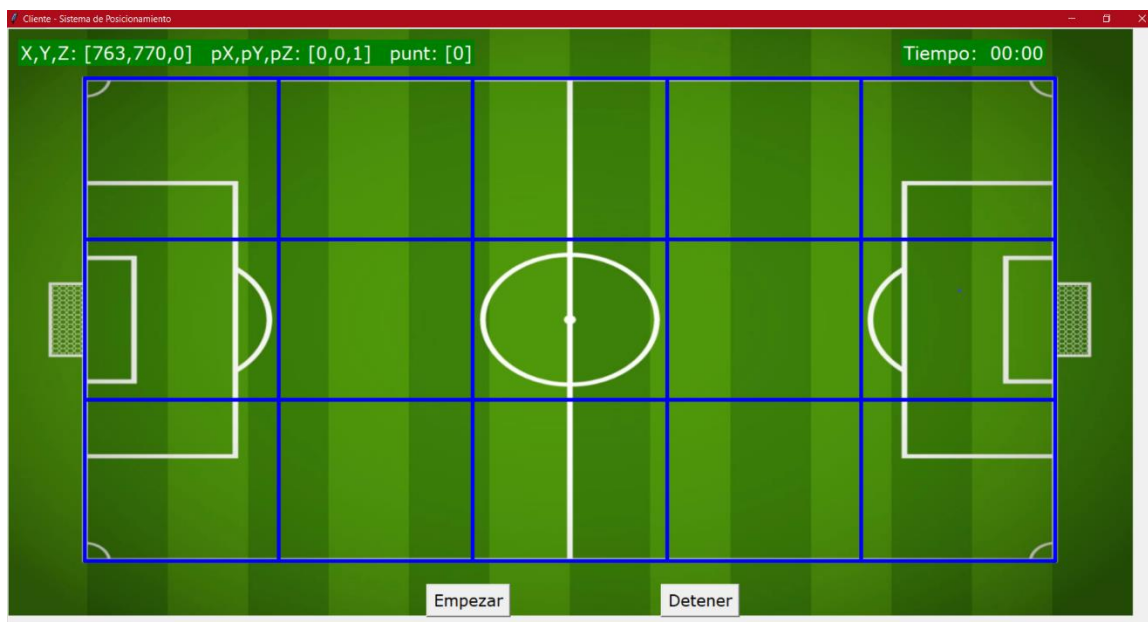
Nota: La tabla muestra el valor numérico asignada a la posición del vector en el plano bidimensional tomando en cuenta que el valor de $Z = 0$.

Ya con las delimitaciones, se agregó a la interfaz dos cuadros informativos, en la parte superior izquierda, ver la Figura 13, se tiene la información exacta del punto en pixeles en el que se encuentra el puntero, contiguo se tiene la información de los vectores a los que se referencia ese punto dentro de las delimitaciones establecidas y

por último se informa el código numérico asignado al segmento el cual viene dado según la Figura 12. En la parte superior derecha se tiene la información del tiempo transcurrido del partido, este funciona a la par con los botones que se encuentran en la parte inferior, una vez el operador de clic en el botón empezar, el programa empezará a correr el tiempo y a su vez empieza a enviar los datos del vector codificado por medio de datagramas usando web socket para así tener una transferencia en tiempo real.

Figura 13

Diagrama de la interfaz del sistema implementado.



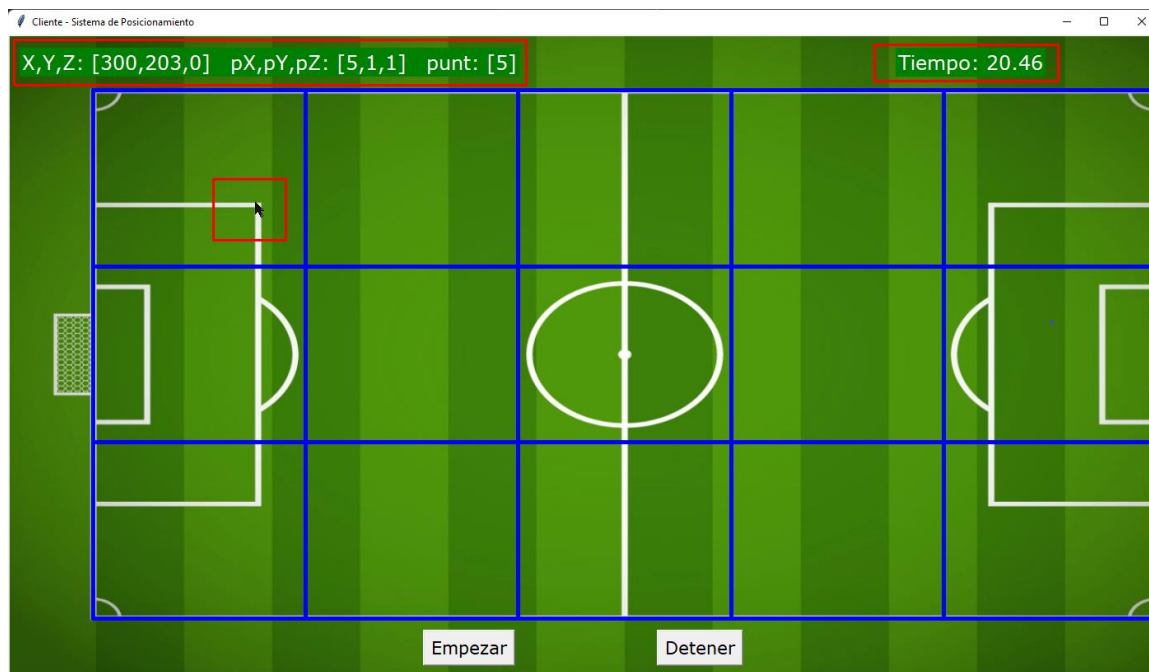
Nota: La captura representa el diseño final de la interfaz de usuario para el sistema de interpretación de datos.

Al iniciar el aplicativo, este empezará inicializando una interfaz representada por la Figura 13, pero la misma se mantiene a la espera de que el operador pulse el botón empezar. Al pulsar el mismo, el programa empieza a correr el tiempo, el cual se visualizará en el reloj ubicado en la esquina superior derecha y de la misma forma toma

de la programación la dirección IP y puerto donde se tiene asignado el Servidor de Datos para activar el envío de estos a través de Web Socket. Ver la Figura 14

Figura 14

Interfaz inicializada.



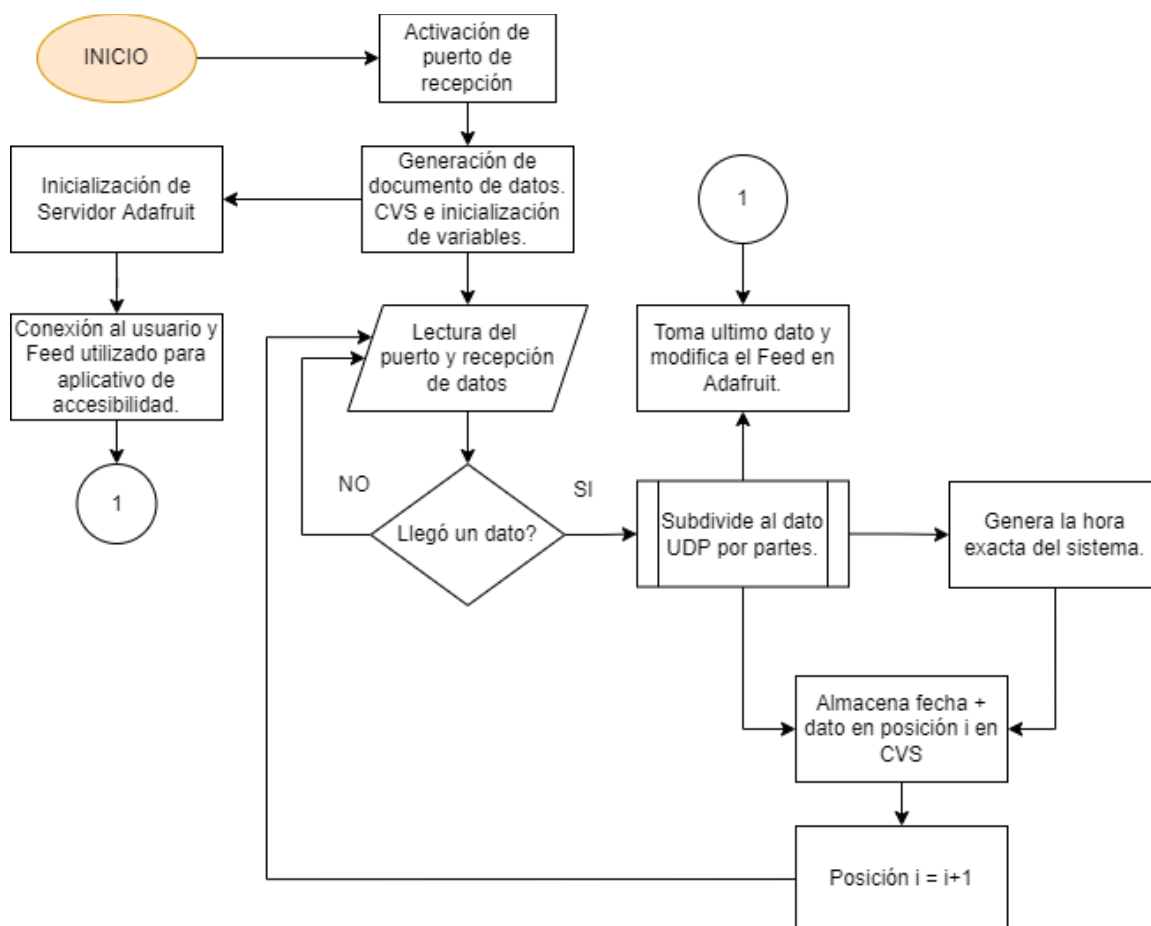
Nota: La captura representa la interfaz de usuario trabajando según la posición del cursor en la pantalla y codificándolos en vectores posición según el cuadrante.

Sistema de servidor de datos

El sistema de servidor de datos tiene el propósito de interactuar entre el sistema presentado en el apartado 0 y el servidor IoT para poder correlacionar los datos desde un sistema central establecido en la nube. Este se implementó en dos modelos, el primero bajo un servidor local el cual sirvió para realizar pruebas iniciales y corrección de errores y el segundo ya se lo desarrollo en un servidor dedicado en la red el cual tendrá la programación descrita en la Figura 15.

Figura 15

Diagrama de flujo del sistema de servidor de datos.



Nota: El diagrama representa el modelo base para el desarrollo del programa que será implementado en el servidor.

Una segunda parte de este sistema viene a ser una plataforma IoT la cual viene a ser el medio de compartición de datos con dispositivos accesibles a la interactividad, esta se trabajó sobre Adafruit, al ser una plataforma de acceso gratuito y fácil de implementar en distintas tarjetas de desarrollo y aplicativos, tiene tiempos de retardo menores que otras plataformas y permite visualizar estos datos desde su portal web para detección de fallas.

El sistema del almacenamiento de datos se realizó usando programación en Python y el uso de varias librerías mostradas en la Tabla 4 para el manejo de los datos, así como recibir, decodificar, usar y almacenar los mismos.

Tabla 4

Librerías utilizadas en el sistema del servidor de datos.

Librería	Especificación
Adafruit_IO	Interfaz para manejo de datos lot Adafruit
Datetime	Interfaz para manejo de fechas y funciones
OS	Interfaces misceláneas del sistema operativo
Flask	Complemento para utilizar páginas web con Python
Socket	Interfaz de red de bajo nivel

Nota. La tabla muestra las librerías usadas en el segundo sistema. Tomado de “Python Documentation”, (Python, 2019)

Tomando en referencia al programa desarrollado para el sistema de interpretación de datos, este genera un datagrama el cual contiene en orden los siguientes datos: fecha y hora, posición del píxel en coordenadas, los vectores codificados (Z , X , Y) y codificación numérica del cuadrante al que pertenecen según la Figura 12. A este sistema se le asigna una IP dependiendo del lugar donde sea implementado, para las pruebas iniciales se desarrolló bajo una IP local y un puerto abierto para el uso de transferencia de datos UDP, para las pruebas finales se almacenó este programa en la plataforma de servidores virtuales Digital Ocean la cual ofrece una variedad de recursos dependiendo el uso que se le va a dar al servidor dedicado. (DigitalOcean, 2022)

El servidor en la plataforma Digital Ocean se generó bajo las características representadas en la Tabla 5

Tabla 5

Características del servidor básico en la plataforma Digital Ocean.

Características	Especificación
Sistema Operativo	Ubuntu 20.04 (LTS) x64
Memoria RAM	1 GB
Memoria ROM	25 GB SSD
Transferencia Datos	1000 GB
Región Servidor	New York

Nota. La tabla muestra las características del plan Básico que se puede contratar en la plataforma Digital Ocean para un servidor dedicado del cual se basó para el utilizado en el proyecto. Tomado de “*Create Droplets – Digital Ocean*” (DigitalOcean, 2022)

En esta plataforma se almacenó tanto el programa de recepción de datos y a su vez posee una data que almacena todos los datos recibidos para un uso futuro de requerirlo. Una vez implementado se puede realizar un monitoreo en la misma plataforma tanto del uso de ancho de banda, memoria y recursos los cuales se muestran en una forma gráfica como se puede apreciar en la Figura 17 y de la misma forma la configuración y uso de la misma es bajo un terminal que ofrece la plataforma o se puede realizar bajo los programas de conectividad SSH como por ejemplo *Putty* del cual se conecta con la IP pública que se genera automáticamente al crear el servidor.

Figura 16

Datos tomados del archivo CSV.

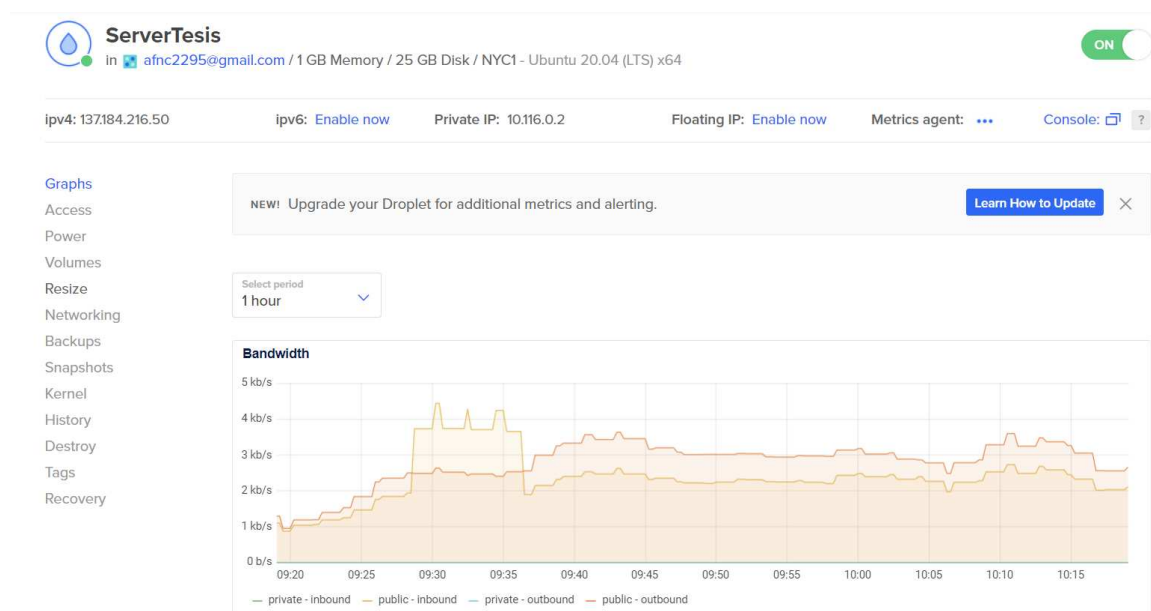
	A	B	C	D	E	F	G
1	date	x	y	z	pX	pY	punt
2	2022/05/31-20:38:08.518832	63	14	0	0	0	0
3	2022/05/31-20:38:08.765124	63	14	0	0	0	0
4	2022/05/31-20:38:09.015627	542	611	0	4	3	14
5	2022/05/31-20:38:09.267435	536	598	0	4	3	14
6	2022/05/31-20:38:09.521871	536	598	0	4	3	14
7	2022/05/31-20:38:09.772977	536	598	0	4	3	14
8	2022/05/31-20:38:10.033003	536	598	0	4	3	14
9	2022/05/31-20:38:10.284455	541	567	0	4	3	14
10	2022/05/31-20:38:10.537615	523	460	0	4	2	9
11	2022/05/31-20:38:10.788216	518	390	0	4	2	9
12	2022/05/31-20:38:11.049465	518	390	0	4	2	9
13	2022/05/31-20:38:11.314458	518	390	0	4	2	9
14	2022/05/31-20:38:11.573228	518	390	0	4	2	9
15	2022/05/31-20:38:11.826084	525	349	0	4	2	9
16	2022/05/31-20:38:12.076494	541	235	0	4	1	4
17	2022/05/31-20:38:12.337487	550	187	0	4	1	4
18	2022/05/31-20:38:12.595627	550	187	0	4	1	4
19	2022/05/31-20:38:12.852171	550	187	0	4	1	4
20	2022/05/31-20:38:13.111004	550	187	0	4	1	4
21	2022/05/31-20:38:13.373232	550	187	0	4	1	4
22	2022/05/31-20:38:13.641166	550	187	0	4	1	4
23	2022/05/31-20:38:13.894687	703	180	0	3	1	3
24	2022/05/31-20:38:14.143627	765	188	0	3	1	3
25	2022/05/31-20:38:14.406524	811	192	0	3	1	3
26	2022/05/31-20:38:14.667135	811	192	0	3	1	3
27	2022/05/31-20:38:14.923915	811	192	0	3	1	3

Nota: La captura muestra los datos almacenados por el sistema de datos en el archivo CSV

En esta plataforma de igual forma se tiene una programación en HTML el cual se complementa con el emulador web para el aplicativo HbbTv, este se tiene configurado de tal forma que el usuario puede empezar con el uso de datos siempre y cuando sea presionado el botón rojo en el control de la TV digital o en este caso en el emulador Web como se muestra en la Figura 18.

Figura 17

Servidor dedicado en la plataforma Digital Ocean.



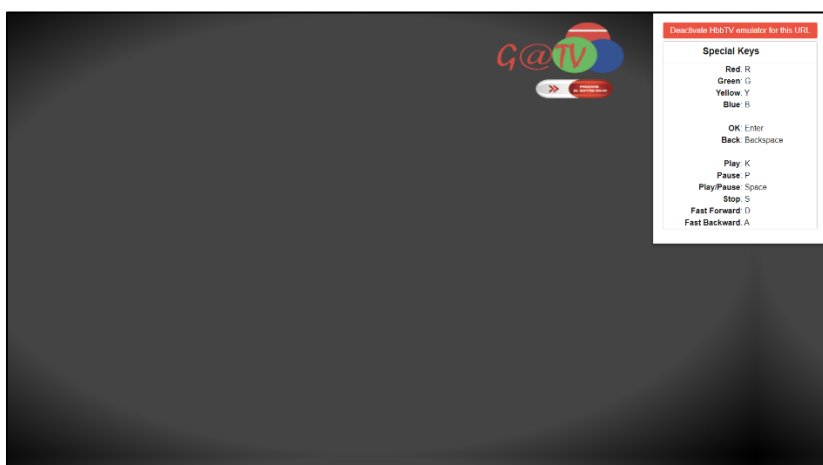
Nota: La figura muestra las características con las que se generó el servidor dedicado en la plataforma Digital Ocean que almacena el sistema del servidor de datos. Tomado de “Digital Ocean” (DigitalOcean, 2022)

Los aplicativos de HbbTv se basan en una programación entre HTML y Java que a su vez pueden ser visualizados a través de una página Web con el uso de complementos o extensiones que poseen los navegadores. En este caso se implementó en forma de un anuncio, este presentaría el mensaje mientras es transmitido el inicio o más conocido como la previa del encuentro deportivo el cual indica al usuario que la transmisión del partido posee un complemento en segundo plano de accesibilidad y para activarlo, el usuario debe aplastar el botón rojo en su control siempre y cuando disponga de un dispositivo o aplicativo compatible con la información. (Yáñez Villamarín, 2015)

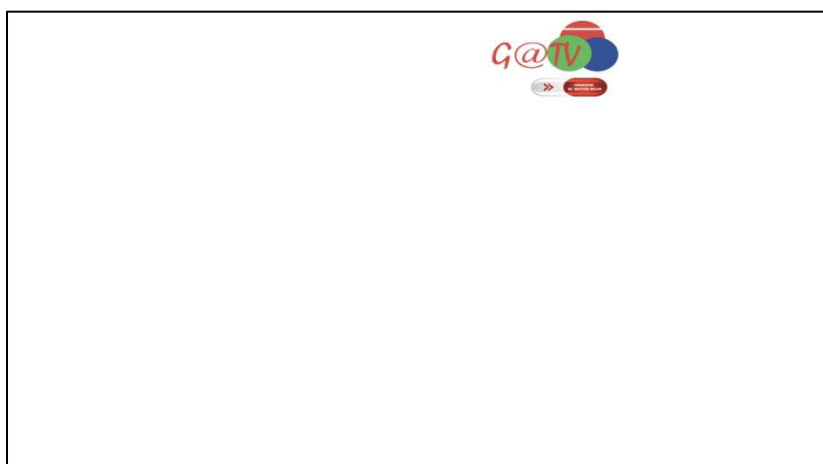
Para poder visualizar este contenido especial, se basó bajo un complemento que posee el navegador Google Chrome y este se lo puede obtener desde sus extensiones con el nombre de *RedOrbit HbbTV Emulator*. En la Figura 18 se muestra dos capturas las cuales muestran las diferencias en el uso de la extensión en una página de prueba. (Redorbit.hbbtv, 2019)

Figura 18

Pruebas de la página web incluido el complemento HbbTv.



(a)

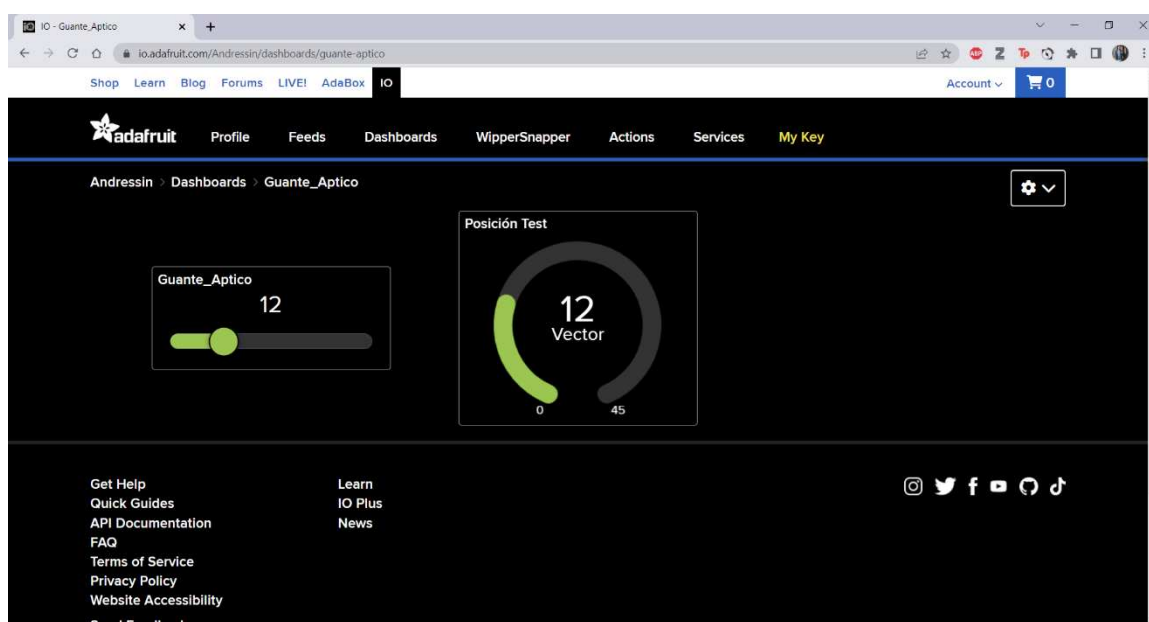


(b)

Nota: La figura muestran las pruebas con el aplicativo web usando la extensión para el navegador Google Chrome RedOrbit HbbTV Emulator (a) Complemento Encendido (b) Complemento Apagado. (Villamarin, 2014)

Figura 19

Servidor IoT en la plataforma Adafruit.



Nota: La figura muestra el entorno de la plataforma IoT de la cual se pueden hacer el uso de varios implementos sean modificables en la misma plataforma como el recuadro de la izquierda o modificables únicamente desde el aplicativo integrado como el recuadro central. (A Minority and Woman-owned Business Enterprise (M/WBE), 2022)

Una vez el usuario acepte el recibimiento de datos, este activará el complemento con Flask que se encuentra en el servidor de datos dando inicio al programa desarrollado (ver la Figura 15) el cual obtiene el datagrama, lo descomprime y almacena

la cadena de caracteres en un archivo de datos CSV que se muestran en la Figura 16, a su vez separa el último dato que viene a ser el carácter asignado al segmento y utiliza el complemento de Adafruit para enviar este dato y almacenarlo en una variable o también llamado *Feed* en la plataforma IoT de Adafruit.

La plataforma IoT es funcional para diferentes áreas de trabajo, ya sean para procesamiento de datos en la nube o intercambio de datos entre aplicativos y tarjetas de desarrollo sin necesidad que las mismas se encuentren en la misma red y pueda obtener un alcance más global y no local. En si el sistema es independiente de la red en la que se encuentren, siempre y cuando la misma tenga conectividad total con internet, esto permite el uso a distancia de cualquier aplicativo o dispositivo de accesibilidad. En esta plataforma se pueden programar varios entornos de desarrollo y aplicar variables o conocidos como *Feed* en la plataforma los cuales van a ser utilizados para almacenar los datos y los cambios realizados. En el entorno como se puede apreciar en la Figura 19, se pueden generar varios bloques visibles ya sean modificables desde la plataforma o solo visibles para el usuario dependiendo la aplicación que se le vaya a dar al mismo. En el proyecto se está usando el valor codificado del vector posición dado en la Figura 12 que fue previamente analizado por el sistema de datos, se envía a la plataforma Adafruit por enlace en sus librerías (ver en la Tabla 4), la plataforma introduce ese dato en la variable asignada o también llamado *Feed* y modifica los bloques que se hayan configurado con esa variable. Para el uso de los dispositivos, estos se programan con la configuración de la cuenta con la que se está trabajando. Esto se realiza dependiendo el dispositivo o lenguaje de programación que se use y la plataforma facilita su forma de implementar dando las líneas de programación como se muestra en la Figura 20 (A Minority and Woman-owned Business Enterprise (M/WBE), 2022)

Figura 20

Usuario y llave para la configuración de Adafruit en dispositivos.

YOUR ADAFRUIT IO KEY ✕

Your Adafruit IO Key should be kept in a safe place and treated with the same care as your Adafruit username and password. People who have access to your Adafruit IO Key can view all of your data, create new feeds for your account, and manipulate your active feeds.

If you need to regenerate a new Adafruit IO Key, all of your existing programs and scripts will need to be manually changed to the new key.

Username

Active Key

[Hide Code Samples](#)

Arduino


```
#define IO_USERNAME "Andressin"
#define IO_KEY      "aio_bGxG93aRhwJM3LiIKx0AYiqKmneC"
```

Linux Shell

```
export IO_USERNAME="Andressin"
export IO_KEY="aio_bGxG93aRhwJM3LiIKx0AYiqKmneC"
```

Scripting

```
ADAFRUIT_IO_USERNAME = "Andressin"
ADAFRUIT_IO_KEY = "aio_bGxG93aRhwJM3LiIKx0AYiqKmneC"
```



Nota: La figura muestra tanto el usuario como la llave que debe ser configurado en los dispositivos receptores a usarse. En este caso se realizó la configuración de este en el Guante Háptico usando la conectividad con Arduino. (A Minority and Woman-owned Business Enterprise (M/WBE), 2022)

De igual forma los dispositivos reciben estos datos simultáneamente se modifiquen en la plataforma siempre y cuando se encuentren activos y dependiendo su programación pueden llegar a realizar actividades varias, como en el caso de este proyecto controla micromotores y los modifica para que vibren en intensidades distintas de acuerdo con la posición del balón en los cuadrantes.

Capítulo IV: Resultados

Para el análisis completo se desarrollan diversas pruebas, consistentes en el uso de los programas por separado y en conjunto en forma local, con esto determinar los errores existentes entre ambos para así corregirlos de una manera óptima; una segunda prueba se desarrolla entre el sistema de interpretación de datos y el servidor implementado en la nube, con eso se determina los retardos perceptibles, análisis de tráfico y tiempo de retardo que tiene el sistema al ser implementado en la nube. Por último, se realizarán pruebas el dispositivo de accesibilidad (Guante Háptico) siendo utilizado por una persona con discapacidad visual durante un encuentro deportivo para determinar la relación, aceptabilidad y manejo de los datos ante un dispositivo de accesibilidad a la interactividad.

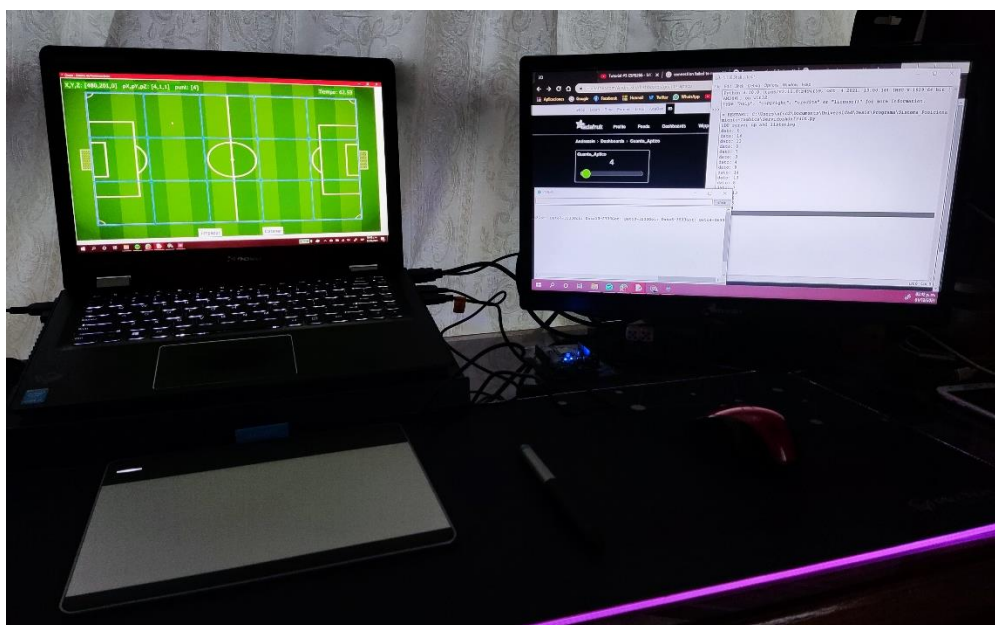
Funcionamiento del Sistema

Para las pruebas del sistema se implementó como primera instancia como un servidor local, el cual, bajo la misma red no poseía ningún tipo de retardo en la transferencia de datos entre el sistema de interpretación de datos y el servidor, puesto que los mismos se encontraban en la misma red local, dichas pruebas son representadas en la Figura 21; al tener un servidor local, la transferencia de datos se realiza internamente en el dispositivo usado por el operador y este mismo sistema almacena un archivo CSV con todos los datos transmitidos durante las pruebas. Para poder percibir el retardo que se tiene entre el dispositivo del operador y la plataforma IoT, se procedió a realizar pruebas de paquetes ICMP obteniendo los resultados de los valores medios descritos en la Tabla 6 tanto del sistema implementado en forma local, y ya con el mismo sistema implementado en la nube.

Tabla 6*Pruebas de diagnóstico de red de pruebas ICMP*

Dispositivo Inicial	Dispositivo final	Implementación	Tiempo respuesta medio (ms)
PC Local (45.173.228.72)	Servidor Adafruit (Adafruit.io)	Servidor local	39
PC Local (45.173.228.72)	Servidor Datos (137.184.216.50)	Servidor en línea	112
Servidor Datos (137.184.216.50)	Servidor Adafruit (Adafruit.io)	Servidor en línea	1.175

Nota: La tabla muestra el valor del tiempo de respuesta medio de un análisis realizado en pruebas de ICMP a un promedio de 100 a 200 paquetes.

Figura 21*Pruebas iniciales del sistema*

Nota: La figura muestra todos los sistemas en funcionamiento, en las primeras pruebas de escritorio realizadas.

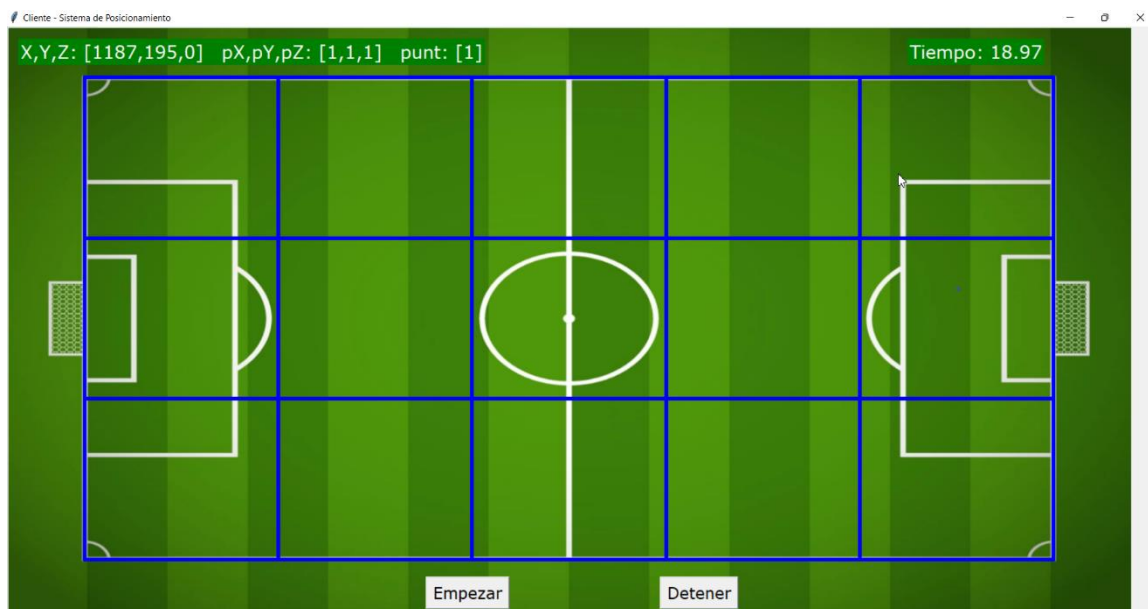
Para la segunda comunicación con el servidor IoT de Adafruit se pudo observar que no se tiene retardo perceptible desde el punto de vista del usuario con el dispositivo de aplicación de interactividad, pero se puede llegar a medir mediante el retardo que existe realizando una prueba de diagnóstico de red.

De igual forma se procedió a verificar el correcto funcionamiento de los sistemas tanto por separado, el sistema completo en un servidor local y ya por último el sistema completo con el servidor en la red. En la Figura 22 se aprecia la interfaz gráfica final para el sistema de interpretación de datos el cual será manejado por el operador, se puede visualizar al cursor en el recuadro superior derecho y con ellos los vectores posicionados con la información de este y en la parte derecha el tiempo transcurrido del encuentro.

Para las pruebas finales se utiliza la tableta de diseño gráfico con sus respectivas configuraciones, la misma se puede visualizar en la Figura 21. La tableta de diseño gráfico, desde sus controladores tiene la factibilidad de la configuración de pulsaciones de lápiz tanto de los dos botones del costado como de los botones internos de la tableta y la pulsación del lápiz sobre la tableta. Entre las configuraciones se colocaron como ayuda los botones que posee el lápiz para utilizarlos como saltos en el cuadrante Z y la pulsación del lápiz sobre la tableta simulando el balón sobre el piso. En la pantalla de la derecha se puede observar tanto la página que se enlaza al servidor IoT y un panel de comandos en el cual se puede visualizar el estado del servidor de datos el cual se encuentre recibiendo los mismos sin interrupciones.

Figura 22

Presentación del sistema de interpretación de datos.



Nota: La figura muestra el diseño final del programa del sistema de interpretación de datos desarrollado para el operador en funcionamiento.

Para realizar una análisis de datos se usó el complemento Wireshark para el análisis de los paquetes enviados en el proceso desde la aceptación del envío de datos por parte del usuario hasta las muestras de paquetes UDP enviados en el transcurso del programa como se muestra en la Figura 23, donde se muestra en el primer recuadro dos respuestas ACK correspondientes al puerto 8080 en el servidor, el cual se encuentra configurado la página la cual funciona con el complemento HbbTv. Posterior a estos, se visualizan los paquetes UDP enviados al puerto 20001, los cuales al tomar un paquete se pueden observar los datos como se muestra en la Figura 24.

Figura 23

Captura del analizador de protocolos Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
35	8.943615	192.168.100.14	137.184.216.50	TCP	55	59468 → 8080 [ACK] Seq=1 Ack=1 Win=512 Len=1
36	8.943834	192.168.100.14	137.184.216.50	TCP	55	59477 → 8080 [ACK] Seq=1 Ack=1 Win=512 Len=1
77	15.299229	192.168.100.14	137.184.216.50	UDP	55	62197 → 20001 Len=13
79	15.546193	192.168.100.14	137.184.216.50	UDP	55	62197 → 20001 Len=13
81	15.796622	192.168.100.14	137.184.216.50	UDP	58	62197 → 20001 Len=16
89	16.048478	192.168.100.14	137.184.216.50	UDP	58	62197 → 20001 Len=16
100	16.303010	192.168.100.14	137.184.216.50	UDP	58	62197 → 20001 Len=16
108	16.553998	192.168.100.14	137.184.216.50	UDP	58	62197 → 20001 Len=16
109	16.813981	192.168.100.14	137.184.216.50	UDP	58	62197 → 20001 Len=16
111	17.065541	192.168.100.14	137.184.216.50	UDP	58	62197 → 20001 Len=16
112	17.318870	192.168.100.14	137.184.216.50	UDP	57	62197 → 20001 Len=15
113	17.569557	192.168.100.14	137.184.216.50	UDP	57	62197 → 20001 Len=15
115	17.830523	192.168.100.14	137.184.216.50	UDP	57	62197 → 20001 Len=15
117	18.095630	192.168.100.14	137.184.216.50	UDP	57	62197 → 20001 Len=15
119	18.354229	192.168.100.14	137.184.216.50	UDP	57	62197 → 20001 Len=15
123	18.606526	192.168.100.14	137.184.216.50	UDP	57	62197 → 20001 Len=15
127	18.857586	192.168.100.14	137.184.216.50	UDP	57	62197 → 20001 Len=15

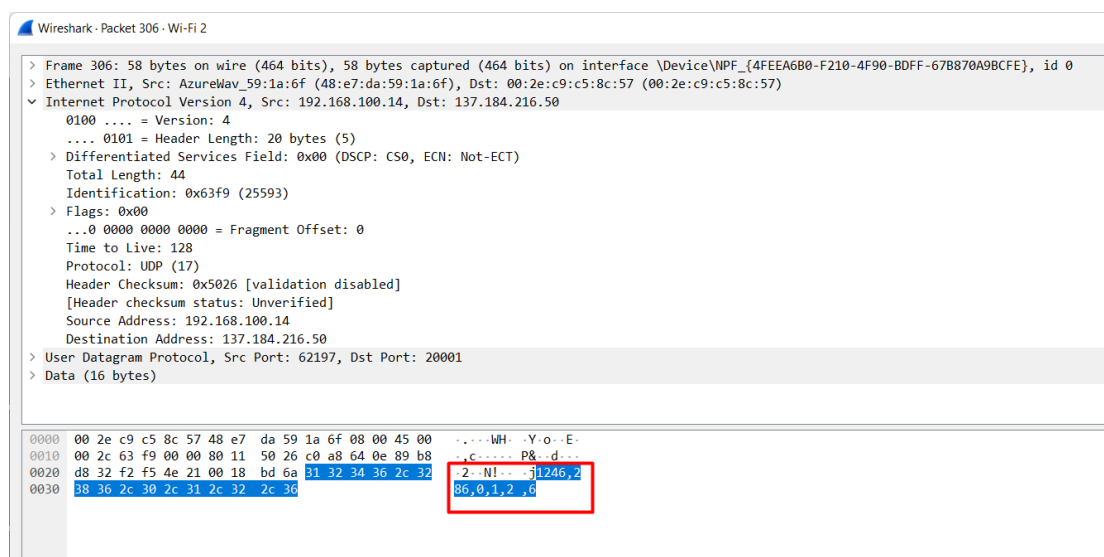
> Frame 77: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface \Device\NPF_{4FEEA6B0-F210-4F90-BDFF-67B870A9BCFE}, id 0
 > Ethernet II, Src: AzureWav_59:1a:6f (48:e7:da:59:1a:6f), Dst: 00:2e:c9:c5:8c:57 (00:2e:c9:c5:8c:57)
 > Internet Protocol Version 4, Src: 192.168.100.14, Dst: 137.184.216.50
 > User Datagram Protocol, Src Port: 62197, Dst Port: 20001
 > Data (13 bytes)

Nota: La figura muestra la captura de protocolos enviados al iniciar la respuesta de aceptación del servidor HbbTv (ACK) y continua con el envío de datos al servidor dedicado (UDP).

Describiendo la Figura 24, se detalla la trama UDP y los datos son visualizados en la parte final delimitados por comas, estos al no estar encriptados pueden ser visualizados fácilmente por un analizador de protocolos, este dato contempla una cadena de 6 caracteres numéricos delimitados por coma en el siguiente orden: Posición del píxel en coordenadas, los vectores codificados (Z, X, Y) y codificación numérica del cuadrante al que pertenecen. Una vez añadida la fecha y hora son almacenados como se muestra en la Tabla 7.

Figura 24

Paquete UDP del analizador de protocolos Wireshark



Nota: La figura muestra los datos contenidos dentro de un paquete UDP enviado desde el sistema de interpretación de datos hacia el servidor dedicado.

Como se explica en la Figura 15 en el apartado 0 el servidor se encuentra operando plenamente en la plataforma web a menos que se tenga una interrupción por comandos. El programa al mismo tiempo que recibe los datos asigna la fecha y hora exacta en la que fueron recibidos los mismos y los almacena en un archivo CSV en el servidor para su posterior uso o análisis de ser el caso.

La Tabla 7 es una muestra tomada de 8 datos en el momento de inicio y cambios de posición realizado. Se puede apreciar que la fecha y hora se almacena con una precisión de microsegundos, adicional se tiene el valor exacto del punto en la pantalla a más del vector de transformación obtenido. Con esto se puede trasladar el programa de conversión de datos hacia el servidor sin tener complicaciones con el operador y reducir la carga computacional en el equipo del operador.

Tabla 7

Muestra de datos tomados del archivo generado en el servidor.

Fecha	Píxel X	Píxel Y	Punto Z	Punto X	Punto Y	Codificación
2022/05/31- 20:38:08.765124	63	14	0	0	0	0
2022/05/31- 20:38:09.015627	542	611	0	4	3	14
2022/05/31- 20:38:10.537615	523	460	0	4	2	9
2022/05/31- 20:38:12.076494	541	235	0	4	1	4
2022/05/31- 20:38:23.917134	1253	488	1	1	2	21
2022/05/31- 20:38:24.174064	1253	488	0	1	2	6
2022/05/31- 20:38:25.713941	1182	656	0	1	3	11
2022/05/31- 20:38:25.964030	1001	757	0	0	0	0

Nota: La tabla visualiza una pequeña muestra de datos tomados del archivo generado en el servidor.

Guante Háptico

Como dispositivo de aplicativo a la interactividad se tomó un Guante Háptico desarrollado para personas no videntes y ayudarlas a experimentar una sensación de ubicación del balón mientras escuchan la repetición de un partido de fútbol. El desarrollo inicial de este dispositivo fue realizado para recibir datos ya mapeados de una interfaz análoga que una vez concluido el encuentro, se realizaba un mapeo, se obtenían los datos y posterior se los cargaba para ser usados y mostrar ya sea una repetición del encuentro. (Villamarin & Mendez, Haptic Glove TV Device for People with Visual Impairment, 2021)

Figura 25

Haptic Glove Accesible TV



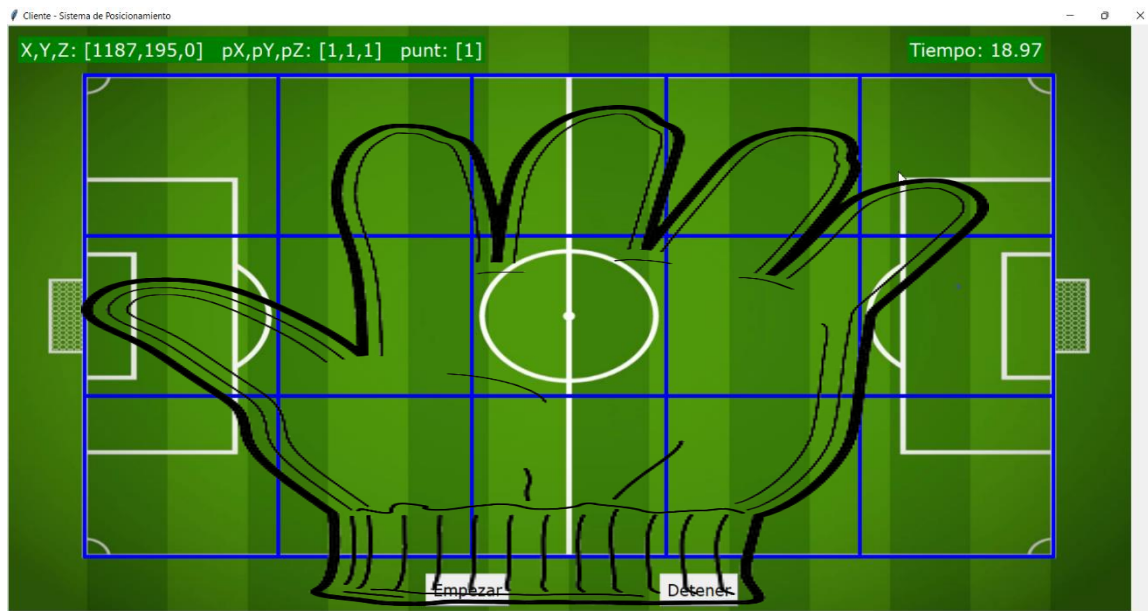
Nota: La figura representa el dispositivo del guante háptico para la ayuda y uso de personas no videntes y las pueda ayudar a ubicar la posición del balón en un encuentro deportivo. Tomado de *“Haptic Glove TV Device for People with Visual Impairment”* (Villamarin & Mendez, 2021)

A este dispositivo se le tuvo que realizar varios cambios, entre ellos el cambio de la recepción de los datos; se le indexó a la plataforma Adafruit con el usuario y llave que se visualiza en la Figura 20. Esto enlazó el *Feed* programado para el intercambio de datos como se muestra en la Figura 19. Del este modo se programó el dispositivo para ser usado en una red Wlan ya que se puede considerar como Gadget de uso limitado. El guante interactúa con la persona no vidente por medio de micromotores ubicados en sobre cada dedo, al recibir un dato la plataforma Adafruit, este modifica instantáneamente a todos los dispositivos vinculados. El guante toma el dato y realiza la

caracterización para encender cada motor y de igual forma con intensidades diferentes para así crear una ilusión de ubicación sensorial como se muestra en la Figura 26.

Figura 26

Interpretación de ubicación entre el sistema y el guante háptico.



Nota: La imagen muestra como el guante háptico muestra una ilusión de posición del balón según las secciones creadas en el sistema de interpretación de datos.

En las pruebas realizadas con este dispositivo se puede apreciar la versatilidad y el fácil uso entre las personas con discapacidad visual, al usar una red Wireless y el uso de baterías recargables, librea al usuario de cables que puedan interferir con la movilidad y entre sus desventajas, se limita en gran parte al alcance de la red ya que su forma compacta limita la distancia de separación que puede llegar a tener el dispositivo y el equipo router. De igual forma, si se mantiene una conexión estable, se tiene una respuesta rápida ante los cambios de datos que envía la plataforma Adafruit.

Evaluación con usuarios

Debido a la pandemia, se tuvo varias complicaciones para las pruebas con usuarios, puesto que las personas con discapacidades fueron gran parte de los sectores vulnerables, varias fundaciones y personas se aislaron para prevenir los contagios. Varios meses después se logró ubicar a una persona con discapacidad visual para realizar estas pruebas y de igual forma al ser su tiempo algo limitado, se desarrollaron solo dos pruebas. Se puede describir que la persona no tuvo esa discapacidad toda su vida, su déficit fue paulatino con forme al tiempo que pasaba por lo que tiene una imagen de lo que es el área de una cancha de un estadio de futbol.

Para estas pruebas, se tomó una transmisión de un partido de futbol de la Serie A de Ecuador y también se realizaron pruebas con una repetición de los goles de un partido por clasificación mundialista entre Colombia y Ecuador. Para la evaluación del sistema a través del usuario, se toma como dato principal que la persona no vidente cuenta con una discapacidad visual del 100%, el cual hizo uso del guante mientras se transmitía el encuentro deportivo (Liga Deportiva Universitaria vs Sociedad Deportiva Aucas). Para detectar la reacción de la persona, se le colocó en la mano derecha el guante mientras que en la izquierda se ubicó sobre una hoja representada por el modelo de la cancha para que él mostrara su ubicación según su sensación con el uso del dispositivo, ver la Figura 28, se colocó con superficies diferentes cada cuadrante y se le explicó los niveles de vibraciones que iba a experimentar y cada posición a la que tenía que relacionar estas, esto con el fin de brindarnos una ayuda a visualizar como él interpreta las vibraciones en el guante y va relacionando la posición según su percepción.

Figura 27

Adaptación de la sala de pruebas.



Nota: La imagen muestra una toma antes del inicio de las pruebas preliminares del sistema completo integrando un dispositivo de accesibilidad a la interactividad.

Como se muestra en la Figura 27, para adaptar la sala de pruebas se colocaron dos ordenadores, en el ordenador del lado izquierdo se encuentra el programa del operador con la tableta gráfica el cual va a ser usado para ingresar la ubicación del balón mientras transcurre el encuentro deportivo el cual es reproducido en la parte de atrás. En la parte derecha se encuentra un segundo ordenador que realizará la interacción con emulador del complemento web el aplicativo de HbbTv el cual debe aceptar para empezar la transmisión de los datos hacia el guante háptico.

Durante el transcurso de la prueba se pudo observar que el usuario se orientaba de forma correcta mientras escuchaba la narración del locutor con respecto al partido, pero también se pudo determinar que, al momento de presentarse cambios bruscos de posición, perdía un poco la ubicación hasta que fueran más paulatinos los cambios. Cabe destacar que el usuario mencionó durante el encuentro que le era un poco complicado mostrarnos la ubicación en el lado izquierdo puesto que en ratos tenía que recorrer toda el área para interpretar la posición, pero en su mano derecha ubicaba correctamente en su mentalidad la posición mientras escuchaba la locución en su parte de atrás.

Figura 28

Prueba del sistema y dispositivo de aplicación a la accesibilidad.



Nota: La imagen muestra una toma durante las pruebas preliminares del sistema.

(Villamarin & Narvaez, 2021)

Adicional se observaba que la reacción de ubicación en cada cuadrante era rápida siempre y cuando tenga mentalizado el mapa de su mano izquierda por parte del usuario a los cambios realizados en el dispositivo del operador y concordando con la ubicación del balón en el encuentro deportivo.

De igual forma en la percepción visual en el instante de las pruebas, se pudo apreciar una ligera confusión al momento de realizar cambios rápidos de posición a lo que se trató de realizar mediciones subjetivas para analizar más a profundidad el funcionamiento del sistema complementado con el dispositivo. Luego de realizado la simulación y pruebas durante un encuentro deportivo, se consultó al usuario una serie de preguntas descritas en la Tabla 8

Tabla 8

Muestra de preguntas

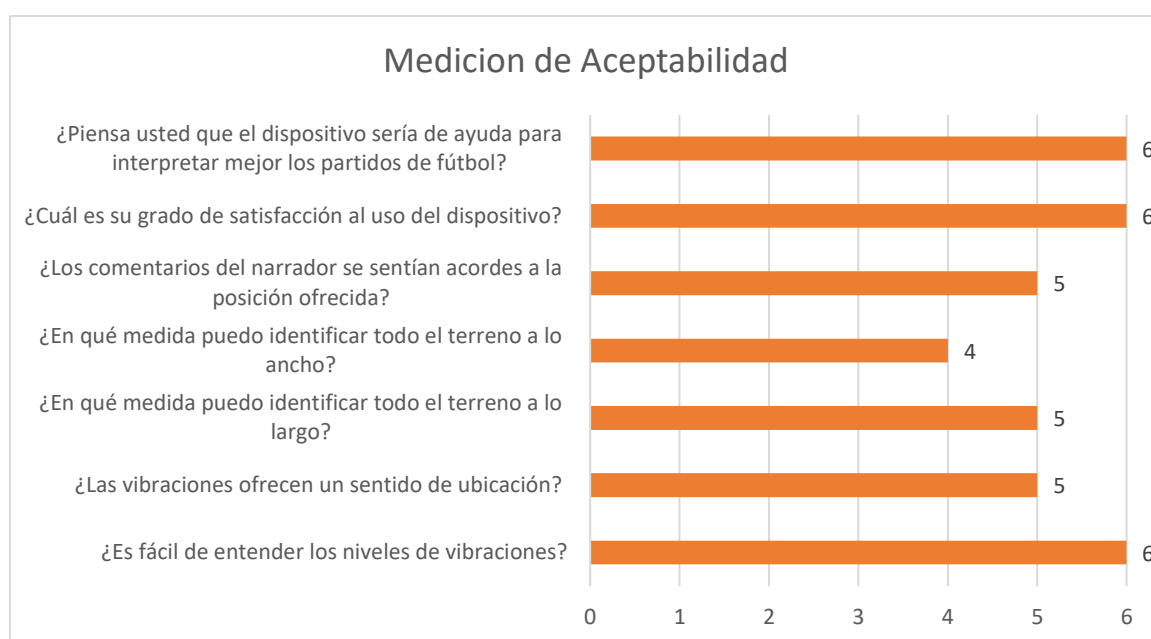
Preguntas	Opciones
¿Es fácil de entender los niveles de vibraciones?	6 5 4 3 2 1
¿Las vibraciones ofrecen un sentido de ubicación?	6 5 4 3 2 1
¿En qué medida puedo identificar todo el terreno a lo largo?	6 5 4 3 2 1
¿En qué medida puedo identificar todo el terreno a lo ancho?	6 5 4 3 2 1
¿Los comentarios del narrador se sentían acordes a la posición ofrecida?	6 5 4 3 2 1
¿Cuál es su grado de satisfacción al uso del dispositivo?	6 5 4 3 2 1
¿Piensa usted que el dispositivo sería de ayuda para interpretar mejor los partidos de fútbol?	6 5 4 3 2 1

Nota: La tabla muestra las preguntas planteadas al finalizar las pruebas del sistema.

Luego de realizar las pruebas con el usuario posterior al encuentro deportivo, se obtuvo como resultado una aceptación considerable hacia el sistema, el mismo se midió de forma subjetiva tomando en cuenta las preguntas de la Tabla 8 y se lo representa en la Figura 29

Figura 29

Valores de aceptación del dispositivo.



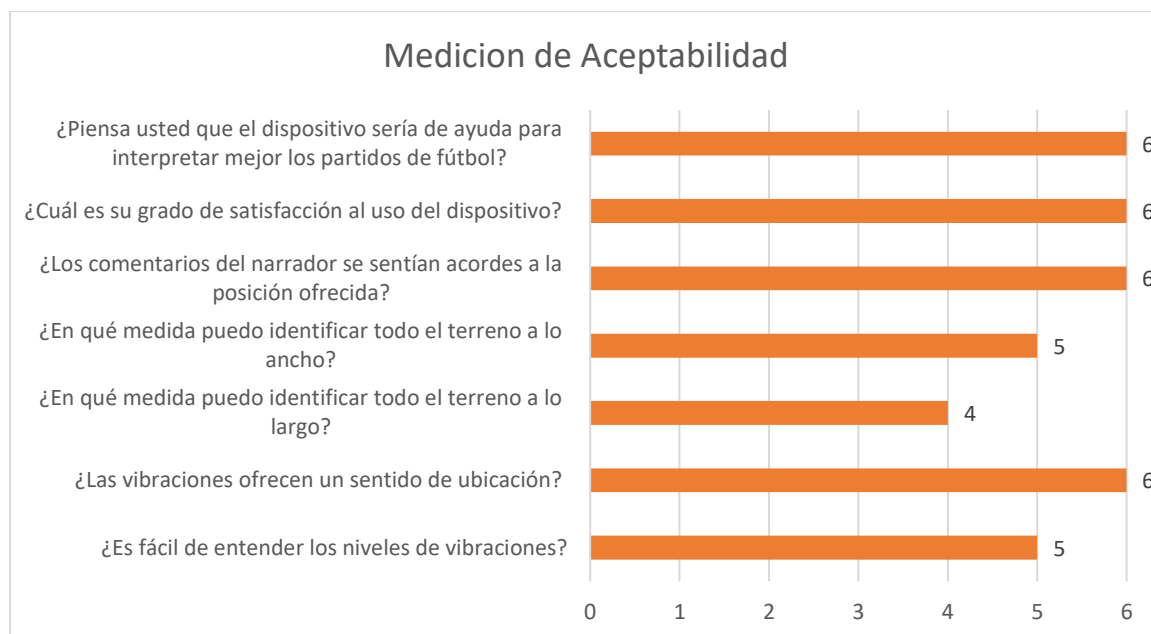
Nota: La gráfica muestra el nivel de aceptación del dispositivo vinculado al sistema asistido y usado por una persona no vidente.

Para una segunda prueba se tomó una repetición de goles del encuentro por la clasificación mundialista entre los equipos de la selección de Ecuador y la selección de Colombia, con esto probar que el uso del sistema no se limita solo a partidos completos sino a ayuda de interpretaciones de varios partidos o repeticiones para incluir de cierta forma a las personas con discapacidad a que formen parte de la sociedad de formas

distintas. Posterior a la segunda prueba se realizó la misma encuesta del análisis de aceptabilidad de la Tabla 8 y se representa en la Figura 30

Figura 30

Valores de aceptación posterior a la segunda prueba.



Nota: La gráfica muestra el nivel de aceptación del dispositivo posterior a una prueba con la repetición de goles del partido entre la selección de Ecuador y la selección de Colombia.

Se puede concluir que el sistema es de uso versátil para la ubicación y logre ayudar a relacionar a personas con discapacidad visual durante partidos de fútbol ya sea desde sus hogares o interactuando en los estadios con este tipo de dispositivos. Cabe recalcar que los comentarios por la persona que fue de ayuda en las pruebas hacen énfasis en que está a gusto con el desarrollo de este tipo de dispositivos, porque ayudan en gran parte a incluirlos en la sociedad ya sea de distintas formas.

Capítulo V: Conclusiones Y Recomendaciones

Conclusiones

- Durante el estudio previo se pudo concluir que los proyectos relacionados con el desarrollo de sistemas asistidos son limitados y mucho más para la ayuda de personas con discapacidades, con el proyecto presente se pretende en dar un soporte al desarrollo de este tipo de sistemas.
- Se analizó los diferentes tipos de lenguajes de programación óptimos para el desarrollo de este sistema llevando como resultado al uso del lenguaje Python ya que ofrece varias librerías y accesibilidad que ayuden a realizar implementaciones bajo distintos sistemas operativos lo cual ayuda al uso de las aplicaciones de un partid de futbol.
- Se desarrolló un algoritmo integrado con una interfaz gráfica capaz de transformar la posición del puntero del mouse en un dato numérico y el mismo en un vector aplicable en un plano bidimensional para ser usado en la identificación de la posición de un balón.
- Se generó un servidor en la red capaz de recibir los datos de posicionamiento de un balón, clasificarlos y enviarlos a una plataforma IoT aplicable con diferentes gadgets o dispositivos para usar los mismos en aplicaciones destinadas a inclusión digital.
- Se generó una base de datos almacenada en el servidor con toda la cadena de datos, clasificados por la fecha y hora de recepción de estos, para ser usados a futuro en un análisis ya sea deportivo o estadístico de los encuentros deportivos.

- Se estableció pruebas con un emulador web a través de librerías para la correlación del sistema con el uso de aplicaciones de TV digital adaptado al estándar europeo HbbTV y así volverlo accesible con los encuentros deportivos mientras son transmitidos en vivo.
- Los resultados obtenidos a través de la experiencia del usuario al utilizar los datos procesados por este trabajo en conjunto con un dispositivo de accesibilidad a la interactividad dan a conocer el potencial de este sistema el cual ayuda a reducir las limitaciones de personas con discapacidades y colaborar con una mayor integración en la sociedad.
- El desarrollo de este sistema es una solución alternativa y práctica de bajo costo, capaz de hacer accesible el contenido visual de un encuentro deportivo transformando en datos aplicables a dispositivos de varios usos, ya sea para la ayuda de percepción de ubicación para personas con discapacidades, o el uso de análisis de estos durante el encuentro.
- Este trabajo crea un diferente uso para la Tv digital en la creación de aplicativos relacionados con la programación y aumentando el potencial que esta ofrece para la interacción entre la programación y el usuario, más aún en la inclusión de personas con capacidades especiales.

Recomendaciones

- La selección de los lenguajes de programación debe analizarse dependiendo el uso o plataformas en las que se vayan a implementar, se debe tomar énfasis principal en esto puesto que varias plataformas interactúan específicamente con ciertos lenguajes y la mayoría no son compatibles por si solos y se debe implementar la ayuda de varios complementos o librerías para el uso de estos.

- El operador que realiza la digitalización de la posición debe estar entrenado en el uso de la interfaz gráfica a fin de evitar que ocurra retardos por su inexperiencia.
- Realizar la familiarización de los escenarios con los dispositivos de ayuda asistida antes de realizar las pruebas.
- Se puede adaptar al servidor para almacenar cada encuentro en una base de datos diferente o archivos diferentes para usarlos o clasificarlos de mejor forma.

Trabajos Futuros

Como trabajos futuros se propone la adecuación de los dos sistemas tanto para las mejoras en las interfaces gráficas y el almacenamiento en el servidor de datos, así como la implementación de una *Machine Learning* o *Deep Learning* o reconocimiento autónomo de objetos por medio de cámaras para obtener la posición exacta del balón en un encuentro deportivo, con esto se puede automatizar el sistema por completo y evitar el uso de un operador.

Se puede implementar una base de datos abierta para el uso de los vectores en futuros eventos o durante repeticiones con el fin de optimizar y hacer uso de estos datos almacenados por varios dispositivos o aplicativos compatibles.

Un principal trabajo a futuro es el desarrollo de aplicaciones o dispositivo para la accesibilidad a la interactividad. Con esto hacer el uso del sistema en indistintamente varias ramas ya sea para personas con distintas discapacidades principalmente visuales, o para crear una experiencia diferente con el usuario durante los encuentros deportivos.

Referencias Bibliografía

- A Minority and Woman-owned Business Enterprise (M/WBE). (2022). *The internet of things for everyone*. Obtenido de Adafruit: <https://io.adafruit.com/>
- Aguirre, D. (2018). Red de Frecuencia Única en el estándar ISDB-Tb. *Revista Espacios*, 39(32), 1-10.
- Alvarez, M., Galeano, D., & Peña, J. (2017). *Manual del lenguaje de marcación de HTML5*. Sevilla : Saga Suite.
- Ayala, A. (25 de Agosto de 2014). *Televisión Digital Terrestre*. Recuperado el 15 de Septiembre de 2020, de <http://alejandroayala.solmedia.ec/?p=491>
- Azarcoya, W. (2019). *La televisión de la nueva generación ATSC 3.0*. Washington D.C: ATSC Organization.
- Benavides Cifuentes, N. B. (2015). *Desfragmentador del flujo de transporte (TS) y analizador de tablas para el sistema de televisión digital terrestre ISDB-T*. Sangolquí.
- Caballero, F., Morales, M., Silva, E., & Caballero, D. (2020). Raspberry Pi, conectividad y programación mediante puertos GPIO. *Revista de Ingeniería Innovativa*, 4(14), 1-13.
- DigitalOcean. (2022). *Simpler cloud*. Obtenido de DigitalOcean: <https://www.digitalocean.com/>

- Fallahi, F., Joki, J., & Teirikangas. (2019). *HbbTV learning for beginners*. Geneva: Afsaneh Fathollahi.
- GitHub Inc;. (12 de mayo de 2017). *FFmpeg*. Recuperado el 17 de noviembre de 2021, de <https://github.com/FFmpeg/FFmpeg>
- Haro, P. (2019). *Herramienta online para la generación de plantillas interactivas multiplataforma para la televisión digital* . Quito: ESPE.
- Instituto Europeo de Normas de Telecomunicaciones. (2021). *Hybrid Broadcast Broadband TV*. Sophia Antipolis - Francia: European Broadcasting Union 2021.
- Jácome, J., Pazmiño, A., Gavilanes, J., Martínez, O., & Ribadeneira, J. (2018). Análisis de modelos de propagación para televisión digital terrestre (TDT), bajo el estándar ISDB-Tb en las ciudades de Ambato y Latacunga, Ecuador. *Journal of Science and Research*, 1(3), 84-90.
- Jaksic, B., Milosevic, I., Petrovic, M., Bojanic, S., & Ilic, S. (2017). Characteristics of hybrid broadcast broadband television (HBBTV). *Publication in Natural Sciences*, 7(1), 41-45.
- Llorente, Á. (2018). *Desarrollo de servicios OTTsíncronos mediante el estándar HBBTV 2.0.1 para TV digital* . Madrid : UPM.
- Martínez, F. (2019). *Diseño e implementación de un sistema de alarma IoT basada en tecnologías Open Source*. Cartagena: Universidad Politécnica de Cartagena .
- Microsoft. (2022). *Code editing Redefined*. Obtenido de Visual Studio Code: <https://code.visualstudio.com/>
- Millán, R. (2014). Televisión Digital Terrestre (TDT). *ACTA*, 1(2), 111-115.

- Ministerio de Asuntos Económicos y Transformación Social. (2020). *HbbTV*. Obtenido de Gobierno de España:
<https://www.televisiondigital.gob.es/TelevisionDigital/tecnologias/Interactividad/Paginas/hbb-tv.aspx>
- Ministerio de Telecomunicaciones. (17 de Agosto de 2019). *TDT Ecuador*. Recuperado el 15 de Septiembre de 2020, de <https://tdtecuador.mintel.gob.ec/#>
- Mintel. (3 de mayo de 2017). *TDT Ecuador*. Recuperado el 5 de febrero de 2022, de <https://tdtecuador.mintel.gob.ec/que-es-la-tdt/>
- Mintel. (5 de mayo de 2017). *TDT Ecuador*. Recuperado el 5 de febrero de 2022, de <https://tdtecuador.mintel.gob.ec/beneficios/>
- Mintel. (2018). *Plan Maestro de transición a la televisión digital terrestre*. Quito: Viceministerio de Tecnologías de la Información y Comunicación.
- Montesdeoca, E. (2016). *Simulación de la capa física del estándar DVB-T2 utilizado en la televisión digital terrestre*. Ambato : Universidad Técnica de Ambato .
- Muñoz, J. (2016). *Estudio de factibilidad para implementación de un laboratorio de Televisión Digital Terrestre (TDT) para el laboratorio de networking en la facultad de Ingeniería Industrial de la Universidad de Guayaquil*. Guayaquil : Universidad de Guayaquil.
- Nakachi, T. (2017). *MPEG Media Transport technologies and its application to immersive media communication services*. Toledo: 7th International Conference on Integrated Circuits, Design, and Verification (ICDV).

Organización de Telecomunicaciones de Iberoamérica. (2016). *Mejores prácticas en la transición a la televisión digital terrestre (TDT)*. CDMX: OTI.

Pargeter, K. (2019). *Fondo de campo de fútbol*. Obtenido de Vecteezy:
<https://es.vecteezy.com/artes-vectoriales/234128-fondo-del-campo-de-futbol>

Python . (2022). *Download the latest version for Windows*. Obtenido de Python:
<https://www.python.org/>

Python. (2019). *Python 3.10.4 documentation*. Obtenido de Python:
<https://docs.python.org/3/>

Redorbit.hbbtv. (11 de 2019). *RedOrbit HbbTV Emulator*. Obtenido de Chrome Web Store: <https://chrome.google.com/webstore/detail/redorbit-hbbtv-emulator/mmgfahampkahlmoahbjcjcgmkmppab?hl=en>

Rodas, A., & Valencia, A. (2018). Desarrollo e implementación de un prototipo para una plataforma tecnológica para la transmisión de texto y video (streaming) en tiempo real empleando tecnología websocket. *Ingenierías USBMed, IX(2)*, 2-10.

Villamarin , D., & Mendez, J. M. (2021). Haptic Glove TV Device for People with Visual Impairment. *Sensors* , 21(7). Obtenido de Universidad Politécnica de Madrid:
<https://doi.org/10.3390/s21072325>

Villamarin , D., & Narvaez, A. (2021). X Iberoamerican Conference on Applications and Usability of Interactive TV JAUTI2021. *Sistema asistido de posicionamiento de un balón en partidos de fútbol con un servidor HBBTV integrado a un guante háptico de TV accesible para personas con deficiencia visual*. Quito: Springer.

Villamarin, D. (2014). *Estudio comparativo y de integración para las plataformas de televisión interactiva europea HbbTv y latinoamericana Ginga*. Madrid: Universidad Politécnica de Madrid. Obtenido de Universidad Politécnica de Madrid.

Wacom. (2022). *Wacom for a creative world*. Obtenido de Wacom:
<https://www.wacom.com/>

West, A. (2018). *The Python Book: The Ultimate Guide to Coding with Python*. Nueva York : Future Publishing.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., & Wesslén, A. (2012). *Experimentation in Software Engineering*. Springer-Verlag Berlin Heidelberg.

Yáñez Villamarín, D. C. (2015). Diseño y desarrollo de guías de laboratorio para generar y manejar el flujo de transporte de televisión digital .