

Sistema de entretenimiento multimedia distribuido por zonas mediante el uso de tecnología WIFI

Camilo Andres Barreto Jimenez y Christian Alfredo Coronado Serrano

Trabajo de Grado para Optar por el Título de Ingenieros Electrónicos

Director

Ariel Yezid Villarreal Solano

Magister en ingeniería de sistemas e informática

Codirector

Homero Ortega Boada

Doctor en Ingeniería

Universidad Industrial de Santander

Facultad de Físico Mecánicas

Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T)

Bucaramanga

2025

Dedicatoria 1

A mi madre, Laudys Jiménez, por ser la fuente inagotable de motivación en mi vida, por inculcarme valores que me hicieron sobresalir y por levantarme en los momentos en que pensé en rendirme. Gracias por creer en mí incluso cuando yo mismo dudaba, por tu voz de aliento en las dificultades y por recordarme que el esfuerzo siempre trae consigo una recompensa. Mamá, gracias por sostener mis sueños y darme la fortaleza necesaria para seguir adelante.

A mi padre, Hugo Barreto, por brindarme un apoyo incondicional y por ser un ejemplo constante de perseverancia y lucha. Gracias por enseñarme que todo es posible con disciplina y determinación, y por mostrarme con tu ejemplo que ningún obstáculo es más grande que la voluntad de superarlo.

A mi hermana, Dayana Rey, por su amor incondicional, por estar siempre a mi lado en los momentos de duda y por convertirse en mi refugio seguro. Gracias por alentarme, por fortalecer mi confianza y por recordarme lo valioso que soy cuando más lo necesitaba.

Este logro no es únicamente mío; detrás de cada paso que di estuvieron ustedes, dándome el impulso que a veces me faltaba. Gracias por ser mi mayor soporte y el motor de este sueño hecho realidad.

Camilo Andres Barreto Jimenez

Dedicatoria 2

Dedico este proyecto con todo mi cariño a mis padres y a mi hermana, quienes han sido mi fuerza, mi refugio y mi mayor inspiración desde siempre. Su amor incondicional, sus palabras de aliento y su fe en mí me han impulsado a continuar aun en los momentos más difíciles. Cada logro alcanzado es también reflejo de su esfuerzo, sacrificio y apoyo inquebrantable.

Extiendo esta dedicatoria a mis familiares, amigos y a todas las personas que han formado parte de mi camino, dejando huellas imborrables con su compañía, su afecto y su sabiduría. A quienes ya no están presentes, pero siguen vivos en mis recuerdos y en las enseñanzas que me dejaron, gracias por haber contribuido a mi crecimiento personal y profesional.

Este trabajo no solo representa un logro académico, sino también el resultado del amor, la paciencia y la fortaleza que he recibido de quienes siempre creyeron en mí.

Christian Alfredo Coronado Serrano

Tabla de Contenido

	Pag
Introducción	15
1. Conceptos previos.....	16
1.1. Sistemas de reproducción multimedia	16
1.1.1. ¿Qué es un sistema de reproducción multimedia?	16
1.1.2. ¿Qué es un reproductor multimedia?	17
1.1.3. ¿Qué hace un reproductor multimedia?	17
1.2. Protocolos de transmisión de video	17
1.3. Raspberry pi	18
1.3.1. ¿Qué es Raspberry pi?	18
1.3.2. Raspberry pi 4	19
1.4. Plataformas	20
1.4.1. Kodi.....	20
1.4.2. Plex	21
1.5. Protocolos de compartición de archivos	22
1.5.1. ¿Qué son y para qué sirven estos protocolos?	22
1.5.2. FTP.....	23
1.5.3. SFT.....	23
1.5.4. SSH	24
2. Desarrollo de la solución	24
2.1. Investigación de dispositivos y herramientas empleadas en el proyecto	24

2.2.	Adquisición de los dispositivos	24
2.3.	Instalación de sistema operativo en la Raspberry pi 4	25
2.4.	Acceso remoto remoto de la Raspberry pi 4	25
2.5.	Instalación de KODI	31
2.5.1.	Estudio de interfaz	32
2.6.	Instalación de <i>addons</i>	32
2.7.	Interfaz Iterativa	33
2.8.	Configuración del control remoto	35
2.9.	Elaboración de la Compartida	36
3.	Resultados	37
3.1.	Las pruebas se organizaron en tres ejes de análisis	38
3.1.1.	Entorno de red	38
3.1.2.	Codec de video	38
3.1.3.	Resolución de pantalla	38
3.2.	Pruebas en el entorno doméstico	38
3.3.	Pruebas en el entorno universitario	39
3.3.1.	Variables registradas en las tablas de datos	41
3.3.2.	Sistematización de resultados	42
4.	Conclusiones	44
5.	Recomendaciones	45
	Referencias	48
	Apéndices	51

Lista de Figuras

	Pág.
Figura 1 <i>Raspberry Pi 4</i>	20
Figura 2 <i>Plataforma Kodi</i>	21
Figura 3 <i>Plataforma Plex</i>	22
Figura 4 <i>Raspberry PI OS</i>	25
Figura 5 <i>VNC y PuTTY</i>	26
Figura 6 <i>Archivo SSH</i>	27
Figura 7 <i>Archivos Micro SD</i>	27
Figura 8 <i>Centro de Configuración PuTTY</i>	28
Figura 9 <i>Terminal Raspberry Pi</i>	28
Figura 10 <i>Sudo raspi-config</i>	29
Figura 11 <i>Herramientas de configuración</i>	29
Figura 12 <i>Herramientas de configuración Opcion3</i>	30
Figura 13 <i>Interface VNC</i>	30
Figura 14 <i>Comprobación de identidad</i>	30
Figura 15 <i>Login</i>	31
Figura 16 <i>Interface Kodi</i>	32
Figura 17 <i>Fondo UIS</i>	34
Figura 18 <i>Fondo Pantalla de inicio</i>	34
Figura 19 <i>Addon Alfa</i>	35
Figura 20 <i>Kore</i>	36

Figura 21 <i>Drive</i>	37
Figura 22 <i>Pruebas hogar</i>	39
Figura 23 <i>Pruebas Universidad (Uis)</i>	41
Figura 24 <i>Grafica FPS Vs Resolución de video Raspberry B Pantalla 1360x768 @ 60 Hz</i>	43
Figura 25 <i>Grafica porcentaje de uso de CPU Vs Resolución de video Raspberry B Pantalla 1360x768 @ 60 Hz</i>	43
Figura 26 <i>Codec Vs Porcentaje de uso CPU Raspberry A & B Pantalla 3840x2160 (4K) @ 30 Hz</i>	44

Lista de Tablas

	Pág.
Tabla 1 <i>Tabla resumen datos resultados pruebas entorno doméstico.</i>	38
Tabla 2 <i>Tabla resumen datos resultados pruebas en universidad.</i>	40
Tabla 3 <i>Tabla datos de rendimiento para las pruebas de resolución de video Raspberry B Pantalla 1360x768 @ 60 Hz.</i>	42
Tabla 4 <i>Tabla datos de rendimiento para las pruebas de Codec MPEG-2, VP8 y Raspberry A & B Pantalla 3840x2160 (4K) @ 30 Hz.</i>	44

Lista de Apéndices

Los apéndices están disponibles en el Repositorio Institucional

Apéndices A. En este anexo encontraras el paso a paso de cómo se puede instalar el sistema operativo de la Raspberry pi 4.

Apéndices B. En este anexo encontraras el paso a paso de cómo se puede instalar Kofi desde la terminal en la Raspberry pi 4.

Apéndices C. En este anexo encontraras el paso a paso de la instalación de la interfaz interactiva.

Apéndices D. En este anexo encontraras el paso a paso de cómo se realizó la elaboración de la carpeta compartida desde la terminal en la Raspberry pi 4 nombrada Drive.

Apéndices E. En este anexo se evidenciarán las imágenes de todas las pruebas realizadas en el transcurso de la elaboración del proyecto tanto en la universidad como en el hogar

Apéndices F. En estos anexos encontraras tablas con los datos arrojados en la práctica y graficas que demuestran el rendimiento del sistema.

Glosario

Add-on: extensión o complemento que amplía las funciones del software Kodi, permitiendo integrar servicios de streaming, bibliotecas y herramientas adicionales.

API (Interfaz de Programación de Aplicaciones): conjunto de funciones y protocolos que permiten la comunicación entre diferentes programas o sistemas.

Bitrate: cantidad de bits procesados por unidad de tiempo en una transmisión de video o audio, que determina su calidad y fluidez.

Código abierto: tipo de software cuyo código fuente está disponible para ser modificado, distribuido y mejorado por los usuarios.

Códec: programa o algoritmo que codifica y decodifica datos de audio o video para optimizar su almacenamiento y transmisión.

Control remoto: dispositivo o aplicación que permite gestionar las funciones de un sistema multimedia sin interacción física directa.

Decodificación: proceso mediante el cual se traduce una señal o archivo codificado a un formato que puede ser comprendido por el dispositivo receptor.

Drive (Google Drive): servicio de almacenamiento en la nube que permite compartir y sincronizar archivos entre dispositivos conectados.

FPS (Frames Per Second): medida que indica el número de imágenes por segundo que se muestran en una reproducción de video, afectando su fluidez.

Hardware: conjunto de componentes físicos de un sistema informático, como la Raspberry Pi, cables o dispositivos de salida.

Interfaz gráfica: entorno visual mediante el cual el usuario interactúa con un sistema o aplicación, representado por menús, íconos y botones.

Kodi: plataforma de código abierto utilizada para la reproducción y gestión de contenido multimedia, adaptable mediante complementos.

LAN (Local Area Network): red local que conecta dispositivos dentro de un área geográfica limitada, como una casa o una universidad.

Multimedia: integración de varios tipos de contenido digital, como audio, video e imágenes, dentro de un mismo sistema de reproducción.

Protocolos de comunicación: conjunto de normas que permiten el intercambio de datos entre dispositivos en una red.

RAM (Random Access Memory): memoria temporal de acceso rápido utilizada por los sistemas para ejecutar procesos y aplicaciones.

Raspberry Pi 4: microcomputadora de bajo costo y tamaño reducido, utilizada para proyectos de automatización, domótica y sistemas multimedia.

Red Wi-Fi: tecnología inalámbrica que permite la conexión entre dispositivos mediante ondas de radio, sin necesidad de cables físicos.

Skin: diseño visual o tema que modifica la apariencia de la interfaz de una aplicación, en este caso la de Kodi.

Streaming: método de transmisión de contenido multimedia a través de internet, sin necesidad de descargar los archivos completos antes de su reproducción.

Resumen

Título: Sistema de entretenimiento multimedia distribuido por zonas mediante el uso de tecnología WIFI*

Autor: Camilo Andres Barreto Jimenez y Christian Alfredo Coronado Serrano**

Palabras Clave: Entretenimiento multimedia, Wi-Fi, Raspberry Pi 4, Kodi, Distribución por zonas.

Descripción: El presente proyecto propone el diseño e implementación de un sistema de entretenimiento multimedia distribuido por zonas, utilizando conectividad Wi-Fi junto con la plataforma Raspberry Pi 4 configurada con la aplicación Kodi. La iniciativa busca ofrecer una alternativa accesible y flexible para la gestión de contenido en diferentes espacios, aprovechando las ventajas del software libre y del hardware de bajo costo.

El problema identificado radica en las limitaciones de los televisores y reproductores tradicionales, que restringen el acceso a contenido, presentan incompatibilidades y elevan los costos al depender de soluciones comerciales cerradas. Para atender esta necesidad, se planteó un sistema que integra componentes de red y reproductores digitales, con una interfaz personalizable y de fácil control desde dispositivos móviles.

Las pruebas realizadas en entornos domésticos y universitarios evaluaron el desempeño a través de la tasa de cuadros por segundo (FPS), considerando códecs, resoluciones y condiciones de red. Los resultados evidenciaron un mejor desempeño con el códec H.264 y un funcionamiento estable hasta 1080p.

En conclusión, la propuesta demostró ser técnica y económicamente viable, aportando una solución funcional aplicable tanto en el ámbito doméstico como educativo. Este proyecto reafirma el potencial del hardware libre como herramienta de innovación accesible para la distribución multimedia en redes locales.

* Trabajo de Grado

** Facultad de Físico Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Director: Ariel Yezid Villarreal Solano, Magíster en ingeniería de sistemas e informática. Codirector: Homero Ortega Boada, Doctor en Ingeniería

Abstract

Title: Multimedia entertainment system distributed by zones using Wi-Fi technology*

Author(s): Camilo Andrés Barreto Jiménez and Christian Alfredo Coronado Serrano**

Keywords: Multimedia entertainment, Wi-Fi, Raspberry Pi 4, Kodi, zonal distribution.

Description: This project proposes the design and implementation of a multimedia entertainment system distributed by zones, using Wi-Fi connectivity together with the Raspberry Pi 4 platform configured with the Kodi application. The initiative aims to offer an accessible and flexible alternative for content management in different spaces, taking advantage of the benefits of open-source software and low-cost hardware.

The identified problem lies in the limitations of traditional televisions and media players, which restrict access to content, present compatibility issues, and increase costs by depending on closed commercial solutions. To address this need, a system was developed that integrates network components and digital players with a customizable interface that can be easily controlled from mobile devices.

Tests conducted in domestic and university environments evaluated performance through the frame rate (FPS), considering codecs, resolutions, and network conditions. The results showed better performance with the H.264 codec and stable operation up to 1080p resolution.

In conclusion, the proposal proved to be both technically and economically viable, providing a functional solution applicable to domestic and educational settings. This project reaffirms the potential of open hardware as an accessible innovation tool for multimedia distribution in local networks.

* Bachelor's Thesis

** Faculty of Physical and Mechanical Engineering. School of Electrical, Electronic, and Telecommunications Engineering. Director: Ariel Yezid Villarreal Solano, Master in Systems and Computer Engineering. Co-director: Homero Ortega Boada, Doctor in Engineering.

Introducción

Los avances tecnológicos en hardware de bajo costo y software libre han impulsado el desarrollo de soluciones versátiles para la gestión y reproducción de contenido multimedia. En este contexto, la Raspberry Pi 4 se ha consolidado como una herramienta accesible que permite implementar sistemas personalizados con amplias posibilidades de configuración. La combinación de esta plataforma con Kodi, un centro multimedia de código abierto, facilita la creación de entornos domésticos o institucionales capaces de organizar, distribuir y reproducir contenido digital de forma eficiente.

Uno de los principales desafíos actuales en el consumo de medios digitales es la distribución estable de contenido multimedia en múltiples zonas sin depender de soluciones comerciales costosas o cerradas. Las limitaciones en las redes locales, la incompatibilidad de códecs y el consumo elevado de recursos dificultan el acceso fluido al contenido audiovisual, especialmente en contextos educativos o de conectividad limitada.

De acuerdo con Deshpande, Sannakki y Madi (2021), el uso de dispositivos embebidos como la Raspberry Pi permite construir sistemas domésticos automatizados y de bajo costo, mientras que Feng (2025) plantea la importancia de optimizar la cooperación entre la nube y los dispositivos de borde para mejorar la distribución de video en tiempo real. Por su parte, Xiao et al. (2023) destacan el papel del ecosistema Kodi como una plataforma abierta que favorece la personalización, la integración de códecs y la expansión de funcionalidades mediante complementos (add-ons).

Asimismo, los aportes de Dow (2024) y Carvalho (2023) profundizan en la aplicación del Internet de las Cosas (IoT) y la automatización doméstica, demostrando que la combinación de hardware económico con software libre ofrece una base sólida para el desarrollo de soluciones sostenibles, escalables y orientadas al usuario.

El presente proyecto propone el diseño e implementación de un sistema de entretenimiento multimedia distribuido por zonas mediante el uso de tecnología Wi-Fi, empleando la Raspberry Pi 4 configurada con Kodi y complementada con una interfaz gráfica personalizable. La solución busca ofrecer una alternativa funcional, económica y flexible que permita acceder a contenido audiovisual desde distintos espacios, manteniendo una reproducción fluida y estable incluso bajo condiciones de red variables. Su desarrollo integra principios de accesibilidad tecnológica y apropiación del conocimiento, orientados a demostrar que es posible implementar sistemas multimedia de calidad utilizando recursos abiertos y asequibles.

Finalmente, este documento presenta la metodología aplicada, los resultados obtenidos y la evaluación del desempeño del sistema bajo diferentes condiciones de red y configuraciones de códecs. Con ello, se busca aportar una base experimental que sirva de referencia para futuros proyectos orientados a la optimización de sistemas multimedia y su adaptación a contextos educativos, domésticos y profesionales.

1. Conceptos previos

1.1. Sistemas de reproducción multimedia

1.1.1. ¿Qué es un sistema de reproducción multimedia?

Un reproductor multimedia es fundamental para el entretenimiento digital, la radiodifusión y la vigilancia , ya que permite una reproducción multimedia fluida en diversas plataformas.

1.1.2. ¿Qué es un reproductor multimedia?

Un reproductor multimedia es un software o hardware diseñado para reproducir contenido multimedia, como audio, vídeo o transmisiones en directo. Ya sea que esté viendo una película, transmitiendo un evento deportivo en directo o revisando grabaciones de vigilancia, los reproductores multimedia son herramientas esenciales en múltiples sectores, desde el entretenimiento para el consumidor hasta la radiodifusión profesional.

1.1.3. ¿Qué hace un reproductor multimedia?

Los reproductores multimedia decodifican y renderizan archivos o transmisiones multimedia, convirtiendo el vídeo y el audio comprimidos en una experiencia de reproducción agradable. Realizan diversas tareas, entre ellas:

Reproducción de archivos multimedia locales (MP4, MKV, AVI).

Transmisión de contenido en vivo o bajo demanda.

Admite protocolos como HLS y MPEG-DASH para una transmisión fluida (Holmdahl, 2025).

1.2. Protocolos de transmisión de video

Cada vez que pulsas "Reproducir" en un video, ya sea de Netflix, YouTube, un partido deportivo en vivo o incluso una videollamada, algo ocurre silenciosamente en segundo plano que garantiza que el video te llegue. Ese "algo" es un protocolo de transmisión de video.

Un protocolo de streaming es básicamente un conjunto de reglas que define cómo se transmiten los datos de video y audio por internet. Piénsalo como el método de entrega de video. Decide cómo se fragmenta el contenido, cómo se envía y cómo se reensambla sin problemas en tu dispositivo. Sin el protocolo adecuado, los videos se almacenarían en búfer sin parar, perderían calidad o, peor aún, no se reproducirían.

Cada estándar tiene sus propias fortalezas y debilidades: algunos están diseñados para contenido a pedido de alta calidad, otros para transmisión en vivo de baja latencia y algunos son formatos heredados que dieron forma a la historia del video en línea.

Comprender los protocolos de transmisión es importante porque afectan directamente la experiencia:

Qué tan rápido comienza a reproducirse un vídeo, que tan suave se ve en Internet lento, que bien protegido está contra la piratería, e incluso cuánto cuesta entregarlo a gran escala. (VdoCipher, 2025)

1.3. Raspberry pi

1.3.1. ¿Qué es Raspberry pi?

La Raspberry Pi es una computadora de bajo costo y con un tamaño compacto, del porte de una tarjeta de crédito, puede ser conectada a un monitor de computador o un TV, y usarse con un mouse y teclado estándar. Es un pequeño computador que corre un sistema operativo Linux capaz de permitirle a las personas de todas las edades explorar la computación y aprender a programar lenguajes como Scratch y Python. Es capaz de hacer la mayoría de las tareas típicas de un computador de escritorio, desde navegar en internet, reproducir videos en alta resolución, manipular documentos de ofimática, hasta reproducir juegos.

Además la Raspberry Pi tiene la habilidad de interactuar con el mundo exterior, puede ser usada en una amplia variedad de proyectos digitales, desde reproductores de música y video, detectores de padres, estaciones meteorológicas hasta cajas de aves con cámaras infrarrojas.

1.3.2. Raspberry pi 4

La Raspberry Pi 4 Modelo B cuenta con un procesador de cuatro núcleos de 64 bits de alto rendimiento, compatibilidad con dos pantallas con resoluciones de hasta 4K a través de dos puertos micro HDMI, decodificación de vídeo por hardware de hasta 4Kp60, hasta 8 GB de RAM, LAN inalámbrica de doble banda de 2,4/5,0 GHz, Bluetooth 5.0, Gigabit Ethernet, USB 3.0 y capacidad PoE (mediante un complemento PoE HAT independiente). Para el usuario final, la Raspberry Pi 4 Modelo B ofrece un rendimiento de escritorio comparable al de los sistemas PC x86 básicos.

Este producto mantiene la retrocompatibilidad con la generación anterior de la Raspberry Pi 3 Modelo B+ y tiene un consumo de energía similar, a la vez que ofrece mejoras sustanciales en la velocidad del procesador, el rendimiento multimedia, la memoria y la conectividad.

La LAN inalámbrica de doble banda y el Bluetooth cuentan con certificación de conformidad modular, lo que permite integrar la placa en productos finales con pruebas de conformidad significativamente reducidas, lo que reduce el coste y el plazo de comercialización.

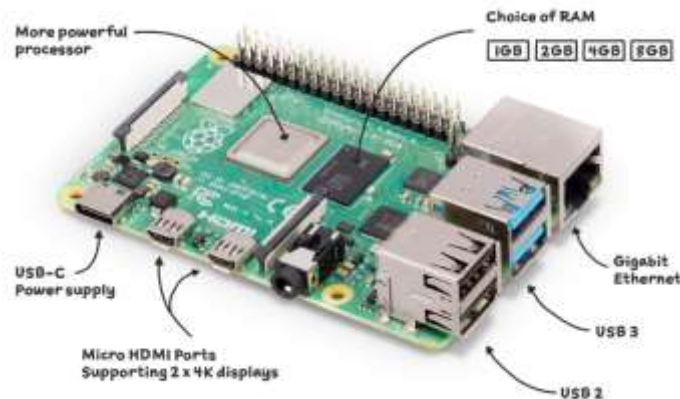
Ventajas:

- La Raspberry Pi, que no utiliza ventilador y es energéticamente eficiente, funciona silenciosamente y consume mucha menos energía que otras computadoras.
- Tiene capacidad USB mejorada: junto con dos puertos USB 2 encontrarás dos puertos USB 3, que pueden transferir datos hasta diez veces más rápido.

- Cuenta con Gigabit Ethernet, junto con redes inalámbricas integradas y Bluetooth.
- Existen varias versiones de Raspberry Pi 4, según la cantidad de RAM: 1 GB, 2 GB, 4 GB u 8 GB (RaspberryPi, s.f.).

Figura 1

Raspberry Pi 4



Nota. Adaptado de Raspberry Pi. (s.f.). Raspberry Pi 4 [Imagen]. Raspberry Pi

(<https://assets.raspberrypi.com/static/raspberry-pi-4-labelled@2x-1c8c2d74ade597b9c9c7e9e2fff16dd4.png>)

1.4. Plataformas

1.4.1. Kodi

Kodi nació de la pasión por los medios. Es un centro multimedia y de entretenimiento que reúne todos tus recursos digitales en un paquete atractivo e intuitivo. Es 100 % gratuito y de código abierto, muy personalizable y compatible con una amplia variedad de dispositivos. Cuenta con el apoyo de un equipo dedicado de voluntarios y una enorme comunidad.

Kodi es un galardonado reproductor multimedia y centro de entretenimiento de código abierto (GPL), compatible con Linux, OSX, Windows, iOS, tvOS y Android. Está diseñado con una interfaz de usuario intuitiva para televisores y mandos a distancia.

Los usuarios pueden reproducir y ver la mayoría de los vídeos, música, podcasts, juegos y otros archivos multimedia digitales desde dispositivos de almacenamiento locales y de red, así como desde internet. Nuestros foros y wiki están repletos de información y ayuda, desde el nuevo usuario hasta el desarrollador de aplicaciones (KODI, s.f.).

Figura 2

Plataforma Kodi



Nota. Adaptado de Kodi Foundation, (s.f.), About – devices [Imagen], Kodi (<https://kodi.tv/images/about--devices.webp>).

1.4.2. Plex

Plex te ofrece un único lugar para encontrar y acceder a todo el contenido multimedia que te interesa. Desde contenido personal en tu servidor, hasta películas y series gratuitas y a la carta, TV en directo y música en streaming, puedes disfrutarlo todo en una sola app, en cualquier dispositivo (Plex, s.f.).

Figura 3

Plataforma Plex



Nota. Adaptado de Plex: 200 canales gratis [Imagen]. Por ADSLZone

(<https://www.adslzone.net/app/uploads-adslzone.net/2021/11/plex-200-canales-gratis.jpg>).

1.5. Protocolos de compartición de archivos

En Internet, la transferencia de archivos juega un papel crucial al permitir el intercambio fluido de información entre dispositivos. Desde la carga de una foto en una red social hasta la sincronización de archivos en la nube, los protocolos de transferencia de archivos son los motores invisibles que hacen posible este flujo de datos.

1.5.1. ¿Qué son y para qué sirven estos protocolos?

Los protocolos de transferencia de archivos son conjuntos de normas que definen cómo se envían archivos de un sistema a otro a través de una red. Son esenciales para que la comunicación entre dispositivos sea coherente, segura y libre de errores.

No solo permiten enviar o recibir archivos de cualquier tamaño, sino que también aportan eficiencia y seguridad. Y en un mundo donde la seguridad digital es cada vez más importante, entender estos protocolos y saber cuál elegir puede marcar la diferencia.

Además, estos protocolos no se limitan al entorno personal.

1.5.2. FTP

El FTP (File Transfer Protocol) es uno de los métodos más antiguos para transferir archivos. Durante años fue el estándar para subir y descargar contenido entre clientes y servidores por su simplicidad.

Pero su punto débil es claro: transmite los datos en texto plano, lo que lo hace vulnerable si no se usa en entornos seguros. Para evitar esto, es recomendable aplicar medidas adicionales como TLS, lo que da lugar al protocolo FTPS.

FTP sigue siendo útil para transferencias básicas, especialmente en redes internas o cuando se manejan archivos no sensibles. Si quieres aprender a conectarte fácilmente mediante este protocolo, consulta esta guía para conectar por FTP con FileZilla.

1.5.3. SFT

SFTP (SSH File Transfer Protocol) se basa en el protocolo SSH y ofrece un entorno cifrado para transferencias seguras.

No solo cifra los datos, sino que también verifica la identidad del usuario y del servidor, lo que lo convierte en una opción robusta y confiable para empresas y profesionales que manejan información crítica.

SFTP es ideal si necesitas garantizar la confidencialidad desde el primer momento. Además, es compatible con múltiples plataformas y fácil de integrar, lo que lo convierte en una opción sólida para entornos corporativos.

Y si usas FTP, recuerda que también puedes proteger tu conexión FTP para reforzar la seguridad de tus datos.

1.5.4. SSH

SSH no es un protocolo de transferencia de archivos como tal, sino un protocolo para acceder de forma segura a dispositivos remotos. Sin embargo, muchas herramientas de transferencia como SFTP o SCP (Secure Copy Protocol) se basan en él.

Es muy utilizado por administradores de sistemas para conectarse a servidores y ejecutar comandos de forma segura. Si necesitas mantener una conexión (Cdmon, 2025).

2. Desarrollo de la solución

2.1. Investigación de dispositivos y herramientas empleadas en el proyecto

En esta etapa se llevó a cabo una revisión exhaustiva de los dispositivos y plataformas disponibles que pudieran contribuir al desarrollo del proyecto. Inicialmente, se evaluaron diferentes alternativas de hardware, concluyendo que la Raspberry Pi 4 resultaba la opción más adecuada, debido a que es la versión más actualizada disponible en el mercado y ofrece un rendimiento óptimo para los requerimientos planteados.

De manera complementaria, se analizaron diversas plataformas destinadas a la gestión y visualización de contenido multimedia. Entre las alternativas consideradas se encontraban Kodi y Plex; sin embargo, tras comparar sus características, se optó por el uso de Kodi. Esta decisión se fundamentó en que Kodi, al ser un software de código abierto, presenta mayor flexibilidad para su personalización y carece de muchas de las restricciones que limitan a otras plataformas, lo que lo convierte en una herramienta idónea para los objetivos del proyecto.

2.2. Adquisición de los dispositivos

Se realizó la cotización y compra de los dispositivos

2.3. Instalación de sistema operativo en la Raspberry pi 4

En esta fase se procedió a la instalación del sistema operativo en la Raspberry Pi 4, seleccionando Raspberry Pi OS en su versión de 64 bits, por ser la más adecuada para aprovechar al máximo las capacidades de este dispositivo. Una vez instalado el sistema, se llevó a cabo el proceso de configuración inicial, que incluyó la creación de un usuario con sus respectivas credenciales de acceso, la asignación de una contraseña segura y la definición de parámetros básicos del entorno.

De igual manera, se establecieron las opciones de red necesarias para garantizar la conectividad del dispositivo dentro de la infraestructura prevista para el proyecto. Finalmente, se habilitó el servicio SSH (Secure Shell), con el fin de permitir el acceso remoto y la administración del sistema de forma segura, lo cual facilita el control y la gestión sin necesidad de conexión directa mediante periféricos.

Figura 4

Raspberry PI OS



Nota. Para un análisis detallado, consulte el Apéndice A.

2.4. Acceso remoto de la Raspberry pi 4

Durante el proceso de instalación y configuración del sistema operativo se presentó la limitación de no disponer de periféricos básicos como monitor, teclado y mouse para la interacción directa con la Raspberry Pi 4. Frente a esta situación, se optó por implementar un esquema de acceso remoto que permitiera la administración del dispositivo desde un computador personal.

Para ello, se utilizaron herramientas ampliamente reconocidas en este tipo de entornos: PuTTY, que permitió establecer conexiones seguras mediante el protocolo SSH, y VNC (Virtual Network Computing), que facilitó la visualización y el control del entorno gráfico del sistema operativo. Esta estrategia no solo resolvió la restricción inicial de hardware, sino que también brindó mayor flexibilidad en la gestión del dispositivo, al posibilitar su configuración y operación sin necesidad de una conexión física directa.

Con el sistema operativo descargado de la Raspberry pi 4, procedemos a instalar PuTTY y VNC para acceder a la Raspberry pi de forma remota

Figura 5

VNC y PuTTY



Creamos un archivo SSH en la memoria de la Raspberry pi, para que así desde PuTTY con la IP de la Raspberry pudiéramos acceder al dispositivo por protocolo SSH.

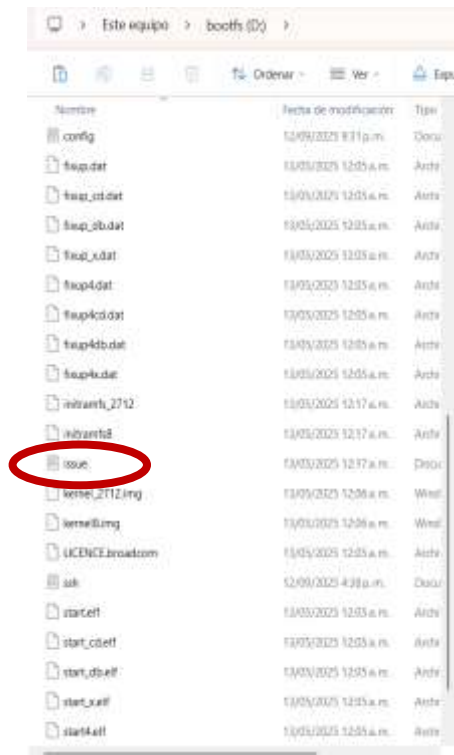
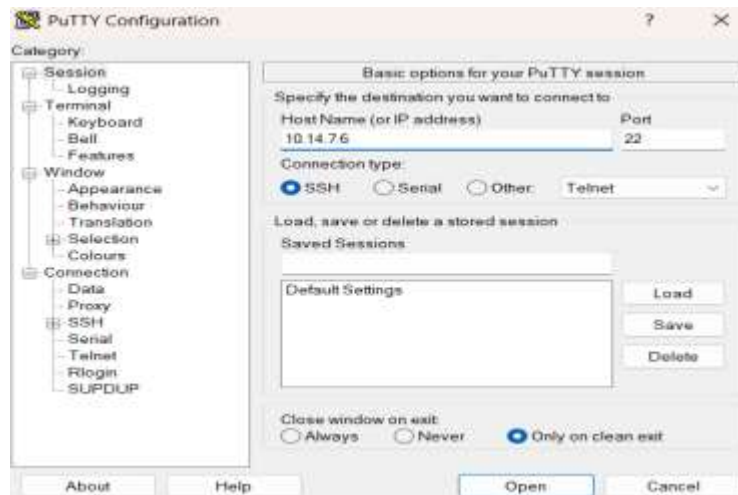
Figura 6*Archivo SSH***Figura 7***Archivos Micro SD*

Figura 8

Centro de Configuración PuTTY



Ingresamos el Usuario y contraseña de nuestro dispositivo

Figura 9

Terminal Raspberry Pi

```
proyecto1@raspberrypi ~  
login as: proyecto1  
proyecto1@10.14.7.6's password:  
Linux raspberrypi 6.12.25+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.12.25-1+rpt1 (2025-04-30) aarch64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Fri Sep 12 16:37:47 2025  
proyecto1@raspberrypi:~$
```

Ejecutamos el comando `raspi-config`, para acceder a la configuración de herramientas

Figura 10

Sudo raspi-config

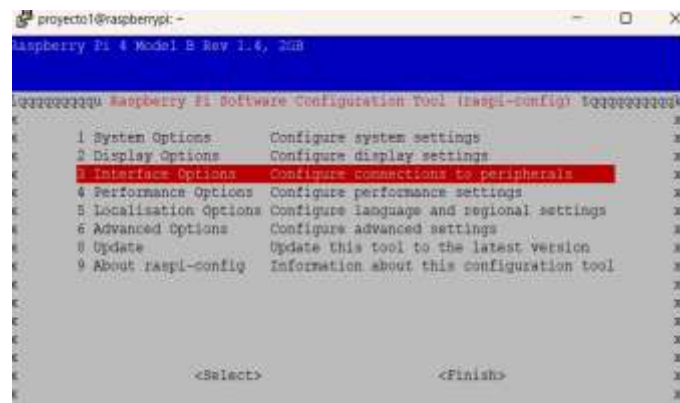
```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Sep 12 16:37:47 2025
proyecto1@raspberrypi:~$ sudo raspi-config
```

Accedemos a la configuración de herramientas de la raspberry pi, seleccionamos la opción

3

Figura 11

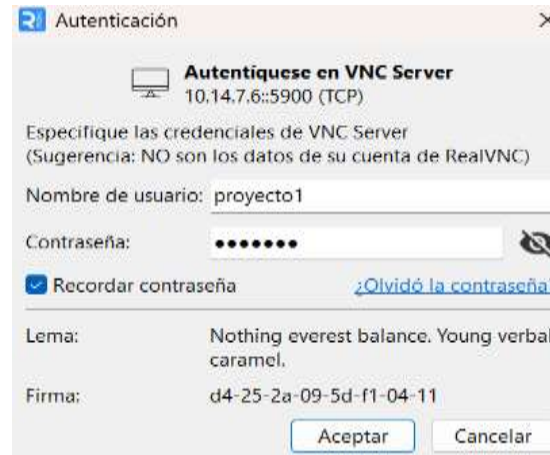
Herramientas de configuración



```
project1@raspberrypi: -
Raspberrry Pi 4 Model B Rev 1.4, 2GB
##### Raspberry Pi Software Configuration Tool (raspi-config): #####
1 System Options      Configure system settings
2 Display Options     Configure display settings
3 Interface Options   Configure connections to peripherals
4 Performance Options Configure performance settings
5 Localisation Options Configure language and regional settings
6 Advanced Options    Configure advanced settings
8 Update              Update this tool to the latest version
9 About raspi-config  Information about this configuration tool

<Select>              <Finish>
```

Seleccionamos la opción I3 para darle acceso remoto desde VNC

Figura 15*Login*

Después de realizar todos estos pasos, tenemos acceso remoto desde el Pc a las Raspberry pi para iniciar con la configuración del sistema.

2.5. Instalación de KODI

Una vez configurado el sistema operativo en la Raspberry Pi 4, se procedió con la instalación de Kodi, la plataforma seleccionada para la gestión y reproducción de contenido multimedia. Dado que Raspberry Pi OS está basado en Linux, la instalación se realizó mediante la terminal, ejecutando una serie de comandos específicos que permitieron descargar e integrar la aplicación de manera correcta en el sistema.

Este procedimiento no solo aseguró la incorporación de Kodi al entorno operativo, sino que también facilitó la posibilidad de actualizarlo y mantenerlo a través de los repositorios oficiales, garantizando estabilidad, seguridad y compatibilidad con los demás componentes del proyecto.

Figura 16*Interface Kodi*

Nota. Para un análisis detallado, consulte el Apéndice B.

2.5.1. Estudio de interfaz

Posterior a la instalación, se procedió a la apertura de la aplicación Kodi con el fin de verificar que el proceso de integración al sistema se hubiera realizado de manera satisfactoria. Una vez confirmado su correcto funcionamiento, se llevó a cabo un recorrido exploratorio por la interfaz, identificando su estructura visual, menús principales y opciones de navegación.

De igual forma, se revisaron los ajustes disponibles dentro de la aplicación, analizando las posibilidades de configuración y personalización que ofrece. Este proceso permitió reconocer los beneficios de la plataforma, entre los que destacan su diseño intuitivo, la facilidad de acceso a los recursos multimedia y la flexibilidad para adaptarse a las necesidades específicas del proyecto.

2.6. Instalación de *add-ons*

Con el objetivo de ampliar las funcionalidades de Kodi y optimizar su desempeño en la reproducción de contenido multimedia, se procedió a la instalación de diversos addons. Durante este proceso se evaluaron diferentes opciones disponibles en la plataforma, seleccionando aquellas

que mejor se ajustaban a los requerimientos del proyecto, en particular las relacionadas con la gestión y visualización de archivos multimedia.

La elección se realizó considerando criterios como la facilidad de uso, la compatibilidad con el sistema operativo instalado y la experiencia de usuario. En este sentido, se descartaron addons que presentaban configuraciones excesivamente complejas o que no aportaban beneficios significativos al propósito planteado. De esta manera, se priorizaron extensiones que ofrecieran un manejo sencillo, práctico y accesible, asegurando que los usuarios pudieran interactuar con el sistema de forma intuitiva.

2.7. Interfaz Iterativa

En las etapas iniciales del proyecto se consideró trabajar directamente sobre el código abierto de Kodi, con el propósito de desarrollar una interfaz interactiva totalmente personalizada. Sin embargo, a medida que avanzó la investigación, se identificó que la complejidad técnica de esta tarea era considerable, debido a la gran cantidad de repositorios y líneas de código que conforman la aplicación. El nivel de dificultad, sumado al tiempo requerido para comprender y modificar el código de manera funcional, hacía inviable esta alternativa dentro de los plazos establecidos para el proyecto.

Ante esta situación, se optó por una solución más accesible y viable que permitiera cumplir con los objetivos planteados: la personalización de la interfaz mediante la instalación de una skin. Para este fin, se seleccionó la skin “Arctic Horizon 2”, la cual cuenta con una licencia abierta que posibilita su rediseño y adaptación visual. Esta alternativa ofreció la flexibilidad necesaria para modificar parámetros de diseño y facilitar la interacción del usuario con la plataforma, manteniendo al mismo tiempo un aspecto estético atractivo.

El proceso de instalación de la skin se realizó siguiendo las instrucciones publicadas por el desarrollador en su repositorio oficial de GitHub. Una vez implementada, se llevaron a cabo una serie de modificaciones con el fin de otorgar a la interfaz una identidad propia de la Universidad Industrial de Santander (UIS). Entre los principales cambios destacan:

Inclusión de una imagen de fondo con el logo institucional de la UIS. (**Figura 17**).

Figura 17

Fondo UIS



Ajuste de la gama cromática en menús y teclado, utilizando tonalidades verdes y blancas.

Incorporación del logo institucional en la pantalla de inicio de la aplicación. (**Figura 18**).

Figura 18

Fondo Pantalla de inicio



Reorganización de los menús principales para simplificar la navegación del usuario.

La interfaz final resultante integró ventanas específicas para la búsqueda de contenido, la reproducción de videos y música, así como una lista de sitios web disponibles a través del *Add-on Alfa*, lo cual fortaleció la experiencia interactiva y la usabilidad del sistema. (**Figura 19**).

Figura 19

Add-on Alfa



Nota. Para un análisis detallado, consulte el Apéndice C.

2.8. Configuración del control remoto

Para la implementación del control remoto de la plataforma se evaluaron diferentes alternativas disponibles. En primera instancia, se consideraron las aplicaciones recomendadas por Kodi para este fin: Kore, aplicación oficial de la plataforma, y Yatse, una versión modificada por la comunidad que ofrece una interfaz más intuitiva, aunque con limitaciones en su versión gratuita. Ambas aplicaciones se encuentran disponibles en la tienda Google Play, lo que facilita su instalación y acceso para los usuarios.

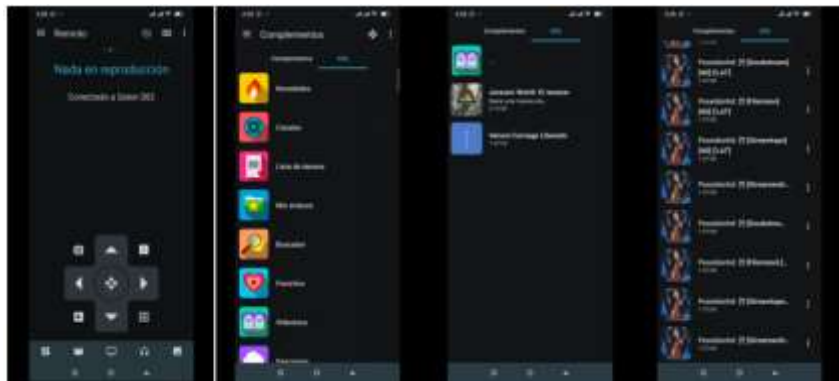
Adicionalmente, se valoró la posibilidad de diseñar desde cero una aplicación personalizada que cumpliera con las mismas funciones de las soluciones previamente mencionadas. Sin embargo, esta opción fue descartada debido al tiempo limitado y a la falta de

experiencia en el desarrollo y prueba de aplicaciones móviles, lo cual hubiera representado una dificultad considerable dentro del marco del proyecto.

En consecuencia, se optó por la utilización de Kore, dado que constituye la solución más práctica y eficiente. Esta aplicación, al ser oficial, cuenta con soporte directo por parte de los desarrolladores de Kodi, es completamente gratuita y presenta un diseño que se ajusta a las necesidades del proyecto. Su configuración permite controlar cada dispositivo conectado a la red mediante la asociación de una dirección IP, junto con un usuario y contraseña específicos para cada Raspberry Pi, garantizando así un manejo seguro y centralizado de la plataforma. (**Figura 20**).

Figura 20

Kore



2.9. Elaboración de la Compartida

Con el propósito de facilitar el acceso y la gestión del contenido multimedia, se implementó una carpeta compartida a través de la plataforma Google Drive. Para ello, inicialmente se creó la carpeta desde la cuenta principal, configurando los permisos necesarios para que pudiera ser utilizada como repositorio común dentro del proyecto.

Posteriormente, en la Raspberry Pi se accedió a la terminal y, mediante la ejecución de comandos específicos, se procedió a la instalación de la herramienta que permite la integración de

Google Drive con el sistema operativo. Una vez completada esta configuración, se estableció un vínculo entre la carpeta en la nube y el dispositivo, de manera que quedara sincronizada automáticamente cada vez que la Raspberry Pi iniciara.

Gracias a este procedimiento, el sistema garantizó la disponibilidad constante de los archivos almacenados, permitiendo que los diferentes dispositivos asociados pudieran acceder a ellos de forma eficiente y organizada.

Figura 21

Drive



Nota. Para un análisis detallado, consulte el Apéndice D.

3. Resultados

En esta sección se presentan los resultados obtenidos a partir de las diferentes pruebas realizadas, acompañados de sus respectivas tablas de datos y representaciones gráficas. El objetivo principal fue evaluar el comportamiento del sistema bajo condiciones variables, con el fin de identificar los límites de rendimiento de la Raspberry en términos de reproducción multimedia.

Para la evaluación del desempeño se empleó como métrica principal los fotogramas por segundo (FPS), ya que este parámetro permite medir de manera directa la fidelidad y la fluidez de los videos transmitidos desde el dispositivo de control hacia la Raspberry.

3.1. Las pruebas se organizaron en tres ejes de análisis

3.1.1. Entorno de red

Se modificó la ubicación de las pruebas con el propósito de observar la reacción del dispositivo frente a diferentes velocidades y niveles de estabilidad de conexión.

3.1.2. Códec de video

Se seleccionaron tres códecs con diferentes exigencias de procesamiento —H.264, MPEG-2 y VP8— para determinar el impacto que cada uno genera en el rendimiento de la Raspberry.

3.1.3. Resolución de pantalla

Se experimentó con resoluciones de 720p, 1080p y 4K, con el fin de establecer los límites de capacidad del hardware.

3.2. Pruebas en el entorno doméstico

La primera serie de pruebas se llevó a cabo en el hogar, utilizando dos dispositivos Raspberry configurados en paralelo. Cada uno se conectó a una pantalla distinta: una con resolución máxima de 720p y otra con resolución de 1080p. En este escenario se analizaron los tres códecs mencionados y, adicionalmente, se realizaron comparaciones de desempeño entre las diferentes resoluciones probadas. (**Tabla 1** y **Figura 22**).

Tabla 1

Tabla resumen datos resultados pruebas entorno doméstico.

Raspberry	Códec	Resolución	FPS Renderizado	Memoria (%)	CPU Total
A	H264	720p	38.09	38%	37%
A	MPEG-2	1080p	37.7	37%	47%
A	VP8	1080p	37.11	39%	45%
A	H264	720p	35.16	31%	40%

A	H264	4K	32.23	37%	94%
B	MPEG-2	1080p	59.52	27%	21%
B	VP8	1080p	59.52	26%	22%
B	H264	720p	59.52	27%	14%
B	H264	4K	54.56	33%	94%

Nota. Los Datos se obtuvieron con resoluciones de pantalla de 1920x1080 @ 60 Hz y 1360x768 @ 60 Hz, para más información ver Apéndices F.

Figura 22

Pruebas hogar



3.3. Pruebas en el entorno universitario

Posteriormente, se replicaron las pruebas en la Universidad, aplicando nuevamente las variaciones de códec y resolución. En este caso se introdujo un nuevo factor de análisis: la franja horaria en la que se realizaron las pruebas. Se eligieron dos momentos específicos del día: el mediodía, cuando la red suele estar más saturada por el alto número de usuarios conectados, y la tarde, donde la demanda de ancho de banda es menor. De esta manera fue posible observar el rendimiento del sistema bajo condiciones de red tanto adversas como favorables.

Es válido aclarar que, dentro de este escenario, también se incorporó un análisis de distancia. El propósito fue demostrar que, mientras el dispositivo esté conectado a la red, la señal del control puede transmitirse a la Raspberry sin importar el nivel de separación física entre ambos. Para ello se ubicó el control en el cuarto piso, una de las Raspberry en el tercer piso y la otra en el primer piso del edificio. Todo este proceso fue registrado en video, lo que permitió garantizar la validez de las pruebas y facilitar la posterior toma de datos. (**Tabla 2** y **Figura 23**).

Tabla 2

Tabla resumen datos resultados pruebas en universidad.

Raspberry	Hora	Códec	Resolución Video	FPS Renderizado	Memoria (%)	CPU Total
A	Medio día	MPEG-2	1080p	10.75	35%	20%
A	Medio día	VP8	1080p	10.42	40%	25%
A	Medio día	H264	720p	9.47	40%	18%
A	Medio día	H264	4K	8.8	49%	92%
B	Medio día	MPEG-2	1080p	13.69	44%	42%
B	Medio día	VP8	1080p	14.66	42%	18%
B	Medio día	H264	720p	13.77	44%	11%
B	Medio día	H264	4K	13.26	49%	92%
A	Tarde-Noche	H264	720p	9.18	36%	11%
A	Tarde-Noche	MPEG-2	1080p	9.47	35%	24%
A	Tarde-Noche	VP8	1080p	9.47	38%	29%
A	Tarde-Noche	H264	4K	9.18	42%	63%
B	Tarde-Noche	H264	720p	13.77	39%	10%
B	Tarde-Noche	MPEG-2	1080p	14.66	40%	22%
B	Tarde-Noche	VP8	1080p	14.66	40%	17%
B	Tarde-Noche	H264	720p	13.26	40%	9%
B	Tarde-Noche	H264	4K	13.69	45%	60%

Nota. Las Datos se obtuvieron con una resolución de pantalla de 3840x2160 @ 30 Hz, para más información revisar el Apéndice F.

Figura 23

Pruebas Universidad (UIS)



Nota. ver Apéndice E.

3.3.1. Variables registradas en las tablas de datos

Los resultados se recopilaban de manera sistemática en tablas, que incluyen no solo las métricas generales de desempeño, sino también parámetros técnicos que aportan mayor precisión al análisis. Entre las variables registradas se encuentran:

- Bitrate de video y Bitrate de audio, como indicadores de la calidad y la demanda de procesamiento en la transmisión.
- Tipo de decodificador (hardware o software), ya que el desempeño de la Raspberry difiere significativamente dependiendo del método utilizado.
- Formato de pixelación, donde se compararon configuraciones como drm prime y yuv420p, observando que influyen directamente en la eficiencia de reproducción.

- FPS de codificación del video y FPS de renderizado, los cuales permiten diferenciar la velocidad de procesamiento interno frente a la visualización final.
- Uso de los diferentes núcleos de la CPU, para identificar el nivel de carga y distribución de tareas en el dispositivo.
- Franja horaria, factor considerado en las pruebas universitarias para evaluar la influencia de la saturación de la red en distintos momentos del día.

3.3.2. Sistematización de resultados

Todos los datos obtenidos fueron procesados y representados mediante diferentes tipos de gráficas. Este tratamiento visual permitió interpretar con mayor claridad las tendencias y variaciones del desempeño de la Raspberry, facilitando al lector la comprensión de los resultados y evidenciando las condiciones bajo las cuales el sistema alcanza su mayor y menor eficiencia. (Tabla 3 y Figura 24, Figura 25, Tabla 4 y Figura 26).

Tabla 3

Tabla datos de rendimiento para las pruebas de resolución de video Raspberry B Pantalla 1360x768 @ 60 Hz.

Raspberry B Pantalla 1360x768 @ 60 Hz							
Resolución	FPS Renderizado	Memoria (%)	CPU 0	CPU 1	CPU 2	CPU 3	CPU TOTAL
720p	59.52	27%	16%	14%	33%	20%	21%
1080p	59.52	27%	18%	18%	29%	22%	22%
4k	54.56	33%	98%	98%	96%	84%	94%

Nota. Ver Apéndices F.

Figura 24

Grafica FPS Vs Resolución de video Raspberry B Pantalla 1360x768 @ 60 Hz.



Figura 25

Grafica porcentaje de uso de CPU Vs Resolución de video Raspberry B Pantalla 1360x768 @ 60 Hz.

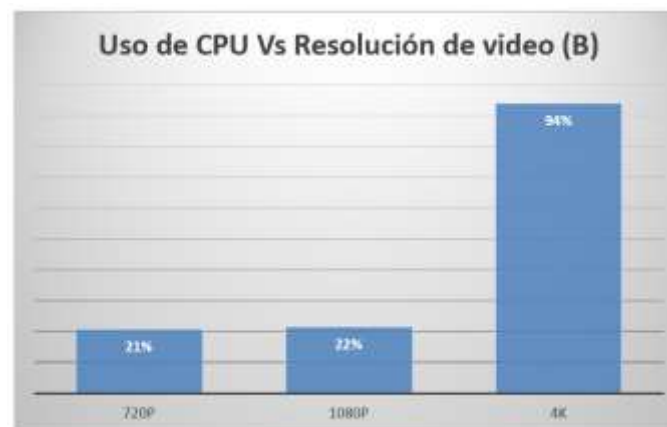


Tabla 4

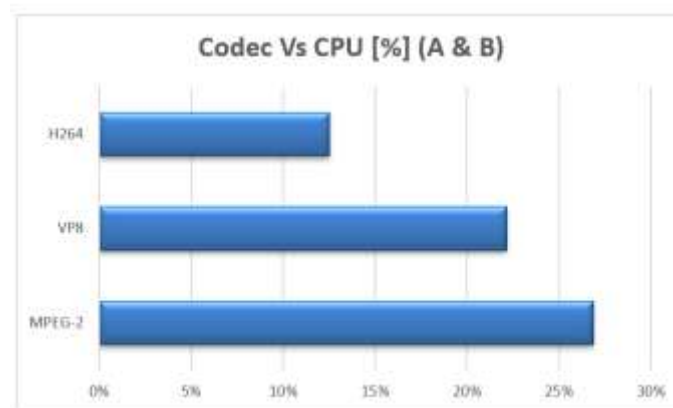
Tabla datos de rendimiento para las pruebas de Códec MPEG-2, VP8 y Raspberry A & B

Pantalla 3840x2160 (4K) @ 30 Hz

Raspberry A & B Pantalla 3840x2160 @ 30 Hz							
Códec	FPS Renderizado	Memoria (%)	CPU 0	CPU 1	CPU 2	CPU 3	CPU TOTAL
MPEG-2	12.1425	38.50%	24.50%	20.50%	41.25%	21.25%	27%
VP8	12.3025	40.00%	22.35%	22.50%	23.75%	20.00%	22%
H264	11.65	40.25%	13.20%	14.58%	10.00%	12.25%	13%

Figura 26

Códec Vs Porcentaje de uso CPU Raspberry A & B Pantalla 3840x2160 (4K) @ 30 Hz.



4. Conclusiones

- Se identificó que el códec H.264 presentó el mejor desempeño en términos de eficiencia y estabilidad, mientras que otros códecs como MPEG-2 y VP8 demandaron mayores recursos de procesamiento, lo que limitó la fluidez en ciertas condiciones.

- El análisis de distancia desarrollado en la universidad permitió comprobar que la señal de control puede transmitirse de manera efectiva sin importar la separación física entre dispositivos siempre que se mantenga la conexión a la red, ampliando así las posibilidades de uso en infraestructuras distribuidas.
- El sistema presentó un rendimiento óptimo hasta resoluciones de 1080p. En resoluciones superiores como 4K, se observaron caídas significativas en la tasa de cuadros por segundo (FPS), lo que refleja las limitaciones propias del hardware de la Raspberry Pi frente a cargas más exigentes.
- Las pruebas realizadas en la universidad con una pantalla de resolución 4K evidenciaron que el proceso de escalado gráfico, sumado a la limitación de la frecuencia de refresco, redujo de manera significativa la tasa de FPS registrada. No obstante, esta condición no comprometió la fluidez percibida en la reproducción del contenido multimedia.
- La implementación de un sistema de entretenimiento multimedia distribuido por zonas, utilizando la plataforma Raspberry Pi 4 y el software Kodi, demostró ser una alternativa viable, flexible y de bajo costo frente a soluciones comerciales más restrictivas, logrando cumplir con los objetivos planteados en el proyecto.

5. Recomendaciones

A partir de las pruebas realizadas y del análisis de resultados, se plantean las siguientes recomendaciones para el uso adecuado del sistema

1. **Estabilidad de la red local:** Es fundamental contar con una red inalámbrica de buena calidad y con el ancho de banda suficiente, ya que de esta condición depende en gran medida el rendimiento del dispositivo y la estabilidad de la transmisión.
2. **Resolución de video:** Se sugiere utilizar resoluciones de reproducción no superiores a 1080p, dado que a partir de este límite se evidencian caídas en el rendimiento, lo cual afecta la fluidez de los contenidos transmitidos.
3. **Selección de códec:** Se recomienda priorizar el uso de códecs de video bien optimizados, como H.264, que en todas las pruebas realizadas demostró un desempeño estable y sin inconvenientes de reproducción.
4. **Condiciones físicas del dispositivo:** Es aconsejable mantener la Raspberry Pi en un entorno fresco y ventilado, con el fin de evitar el sobrecalentamiento y posibles fallos en el procesamiento.
5. **Tiempo de uso:** Debe evitarse un funcionamiento continuo superior a 24 horas, ya que el fabricante no recomienda un uso prolongado y podría comprometer la vida útil del equipo.
6. **Gestión energética:** Se recomienda el uso de fuentes de alimentación certificadas y con el voltaje adecuado, evitando adaptadores genéricos que puedan generar inestabilidad en el sistema.
7. **Mantenimiento preventivo:** Realizar revisiones periódicas del sistema, tanto de hardware como de software, permite garantizar un funcionamiento óptimo y anticipar posibles fallas.

8. **Optimización del almacenamiento:** Es conveniente emplear memorias microSD de buena calidad y capacidad suficiente, ya que el rendimiento del sistema puede verse afectado si el medio de almacenamiento presenta limitaciones o errores.

Referencias

- Alzalam, I., Reddy, R., Sanon, S. P., Lipps, C., & Schotten, H. D. (2023). Demonstration of real-time traffic forecast on a live 5G testbed [Demostración de pronóstico de tráfico en tiempo real en un entorno de prueba 5G]. En 2023 IEEE Future Networks World Forum (FNWF) (pp. 1–6). IEEE. <https://doi.org/10.1109/FNWF58287.2023.10520371>
- Carvalho, M. (2023). Building smart home automation solutions with Home Assistant: Configure, integrate, and manage hardware and software systems to automate your home [Creación de soluciones de automatización del hogar inteligente con Home Assistant: configurar, integrar y administrar sistemas de hardware y software para automatizar el hogar] (1st ed.).
- Cdmon. (2025, julio 29). Protocolos de transferencia de archivos FTP, SFTP y SSH. Cdmon. <https://www.cdmon.com/es/blog/protocolos-de-transferencia-de-archivos-ftp-sftp-y-ssh>
- Deshpande, G. R., Sannakki, S., & Madi, S. (2021). Advanced home automation by using Raspberry Pi [Automatización avanzada del hogar mediante Raspberry Pi]. *Turkish Journal of Computer and Mathematics Education*, 12(11), 3031–3038.
- Dow, C. (2024). Internet of things programming projects: Build exciting IoT projects using Raspberry Pi 5, Raspberry Pi Pico, and Python [Proyectos de programación de Internet de las cosas: desarrolla proyectos IoT emocionantes usando Raspberry Pi 5, Raspberry Pi Pico y Python] (2nd ed.).

- Feng, J. (2025). Cloud-edge cooperation mechanism for fast live sports video distribution [Mecanismo de cooperación entre la nube y el borde para la distribución rápida de video deportivo en vivo]. *Internet Technology Letters*, 8(4). <https://doi.org/10.1002/itl2.70041>
- González-Huitrón, V. (2021, febrero). Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4 [Detección de enfermedades en hojas de tomate mediante CNN con arquitecturas ligeras implementadas en Raspberry Pi 4]. *Computers and Electronics in Agriculture*. Recuperado en septiembre de 2025 de <https://www.sciencedirect.com/science/article/abs/pii/S0168169920331562>
- Holmdahl, S. (2025, abril 4). What is a media player? [¿Qué es un reproductor multimedia?]. Wowza. <https://www.wowza.com/blog/what-is-a-media-player>
- KODI. (s.f.). About KODI [Acerca de KODI]. <https://kodi.tv/about/>
- Min, J. G., & Lee, Y. (2020). High-quality HTTP live streaming system for limited communication bandwidth [Sistema de transmisión en vivo HTTP de alta calidad para ancho de banda de comunicación limitado]. En *2020 International SoC Design Conference (ISOCC)* (pp. 113–114). IEEE. <https://doi.org/10.1109/ISOCC50952.2020.9333086>
- Plex. (s.f.). What is Plex? [¿Qué es Plex?]. Plex. <https://support.plex.tv/articles/200288286-what-is-plex/>

Raspberry Pi. (s.f.). Raspberry Pi 4 Model B [Modelo B de Raspberry Pi 4]. Raspberry Pi Foundation. <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>

Salahuddin, M. A., Sahoo, J., Glitho, R., Elbiaze, H., & Ajib, W. (2018). A survey on content placement algorithms for cloud-based content delivery networks [Un estudio sobre algoritmos de colocación de contenido para redes de entrega basadas en la nube]. *IEEE Access*, 6, 91–114. <https://doi.org/10.1109/ACCESS.2017.2754419>

VdoCipher. (2025, agosto 20). Video streaming protocols: How they work and which one to choose [Protocolos de transmisión de video: cómo funcionan y cuál elegir]. VdoCipher. <https://www.vdocipher.com/blog/video-streaming-protocols/>

Wamser, F., Pries, R., Staehle, D., Heck, K., & Tran-Gia, P. (2011). Traffic characterization of residential wireless Internet access [Caracterización del tráfico en accesos residenciales de Internet inalámbrico]. *Telecommunication Systems*, 48(1), 5–17. <https://doi.org/10.1007/s11235-010-9324-0>

Xiao, Y., Varvello, M., Warrior, M., & Kuzmanovic, A. (2023). Decoding the Kodi ecosystem [Descifrando el ecosistema de Kodi]. *ACM Transactions on the Web*, 17(1), Artículo 2. <https://doi.org/10.1145/3563700>

Apéndices