



**RASPBERRY PI COMO SERVIDOR WEB: UN ENFOQUE PRÁCTICO
PARA ENTORNOS EDUCATIVOS**

**RASPBERRY PI AS A WEB SERVER: A PRACTICAL APPROACH FOR
EDUCATIONAL ENVIRONMENTS**

RASPBERRY PI COMO SERVIDOR WEB: UN ENFOQUE PRÁCTICO PARA ENTORNOS EDUCATIVOS

RASPBERRY PI AS A WEB SERVER: A PRACTICAL APPROACH FOR EDUCATIONAL ENVIRONMENTS

Montalvo Armijos Renny Geovanny ¹,
Guaño Álvarez Ana Cristina ²,
Luis Alberto Zabala Aguiar ³,
Perkins Santiago Haro Parra ⁴,

¹ Instituto Superior Tecnológico Carlos Cisneros, Ecuador, renny.montalvo@istcarloscisneros.edu.ec

² Instituto Superior Tecnológico Carlos Cisneros, Ecuador, anita.guano@istcarloscisneros.edu.ec

³ Instituto Superior Tecnológico Carlos Cisneros, Ecuador, luis.zabala@istcarloscisneros.edu.ec

⁴ Instituto Superior Tecnológico Carlos Cisneros, Ecuador, perkins.haro@istcarloscisneros.edu.ec

RESUMEN

Este estudio aborda la implementación de un servidor web utilizando una Raspberry Pi como plataforma de alojamiento para páginas informativas. La investigación se centra en la selección del hardware y software adecuados, la configuración del sistema y el desarrollo de una página web optimizada para facilitar el acceso a la información académica.

Para la implementación, se utilizó una Raspberry Pi 4 Model B con Raspberry Pi OS y Apache como servidor web. Se configuró el sistema operativo, se establecieron conexiones de red y se desarrolló una interfaz web basada en HTML y CSS. Se realizaron pruebas de rendimiento evaluando el tiempo de carga, estabilidad y consumo de recursos.

Los resultados mostraron que la Raspberry Pi es una opción viable para servidores web de bajo costo, con un consumo energético reducido y una gestión eficiente del tráfico. Durante las pruebas de estrés, el sistema mantuvo estabilidad sin interrupciones en la carga de las páginas. Además, el análisis del uso de recursos reveló que la CPU alcanzó picos elevados en momentos de alta demanda, pero sin afectar significativamente la experiencia del usuario.

El proyecto demuestra que la Raspberry Pi es una alternativa accesible para alojar información académica en instituciones con recursos limitados. Se recomienda monitorear continuamente el almacenamiento y optimizar la configuración del servidor para garantizar su rendimiento a largo plazo.

Palabras clave: Raspberry Pi, servidor web, Apache, redes, páginas informativas.

ABSTRACT

This study addresses the implementation of a web server using a Raspberry Pi as a hosting platform for informational web pages. The research focuses on the selection of appropriate hardware and software, system configuration, and the development of an optimized web interface to facilitate access to academic information.

For the implementation, a Raspberry Pi 4 Model B with Raspberry Pi OS and Apache as the web server was used. The operating system was configured, network connections were established, and a web interface based



on HTML and CSS was developed. Performance tests were conducted to evaluate loading times, stability, and resource consumption.

The results showed that the Raspberry Pi is a viable option for low-cost web servers, with reduced energy consumption and efficient traffic management. During stress tests, the system-maintained stability without interruptions in page loading. Additionally, resource usage analysis revealed that the CPU reached high peaks during periods of high demand, but without significantly affecting the user experience.

The project demonstrates that the Raspberry Pi is an accessible alternative for hosting academic information in institutions with limited resources. Continuous storage monitoring and server configuration optimization are recommended to ensure long-term performance.

Keywords: Raspberry Pi, web server, Apache, networking, informational web pages.

Recibido: Agosto 2025
Received: August 2025

Aceptado: Diciembre 2025
Accepted: December 202



1. INTRODUCCIÓN

En la actualidad, la web constituye el principal medio de acceso a la información y de interacción entre usuarios y sistemas inteligentes. Los buscadores recogen millones de datos provenientes de todas las páginas web del mundo, generando una gran base de datos interna que alimenta procesos de búsqueda, análisis y gestión del conocimiento [1]. Este contexto ha impulsado el desarrollo de aplicaciones web interactivas capaces de controlar dispositivos físicos y de procesar datos en tiempo real, como se evidencia en sistemas de monitoreo y control remoto [2].

En este marco, la Raspberry Pi ha emergido como una plataforma de bajo costo y alta versatilidad para la implementación de servidores y sistemas web educativos y de investigación. Su uso en proyectos de monitoreo ambiental, control de procesos y enseñanza de sistemas embebidos demuestra su relevancia en distintos entornos [3] [4]. La integración de hardware, software y acceso remoto a través de la web facilita la enseñanza práctica de conceptos de redes, programación y sistemas distribuidos [5]; [6].

Diversos estudios han demostrado la utilidad de esta plataforma en la educación y la industria. Por ejemplo, [7] describen cómo los estudiantes analizan dinámicas de sistemas físicos mediante simulaciones controladas por Raspberry Pi, fortaleciendo su comprensión teórica. En el ámbito agrícola, el uso de esta tecnología ha permitido automatizar el riego por goteo mediante aplicaciones web y Android, posibilitando el monitoreo remoto desde teléfonos móviles [8]. De forma similar, [9] proponen un método de alta precisión en la adquisición y procesamiento de datos, mientras que [10] utilizan la Raspberry Pi como nodo central en sistemas de monitoreo estructural (SHM) para infraestructuras de transporte.

Asimismo, se han explorado aplicaciones orientadas a la seguridad y la automatización, como la detección de incendios mediante sensores ambientales [11] o la gestión visual de flujos de datos con herramientas como Node-RED, que simplifican la programación mediante entornos web intuitivos [12]. La eficiencia de estos sistemas depende, en gran medida, del almacenamiento y la gestión

distribuida de la información, lo que ha motivado investigaciones sobre el almacenamiento en caché y la arquitectura multinivel de servicios web [13].

En el ámbito educativo, [14] demuestran que la Raspberry Pi puede configurarse como un servidor LAMP (Linux, Apache, MySQL, PHP), ejecutando aplicaciones como Moodle y proporcionando un entorno de aprendizaje local y portátil. Este enfoque abre la posibilidad de crear laboratorios prácticos accesibles en instituciones con recursos limitados. En ese sentido, el presente estudio se diferencia de los anteriores al proponer el uso de una Raspberry Pi como servidor web educativo, configurado mediante conexión Ethernet y orientado a la enseñanza práctica de administración de servicios web, sin necesidad de infraestructura compleja ni de redes inalámbricas externas.

El sistema desarrollado permite la gestión y control de servicios web en tiempo real, con procesamiento instantáneo de datos y una interfaz gráfica accesible para estudiantes sin experiencia previa en programación [15]. Además, la flexibilidad del hardware facilita la expansión hacia sistemas de control avanzados, como aquellos que emplean placas AD/DA para procesamiento analógico y digital de señales [16].

Por tanto, el problema central que aborda este trabajo radica en la falta de herramientas prácticas, económicas y accesibles que permitan a los estudiantes comprender de forma aplicada la implementación de servidores web. El objetivo principal es demostrar la viabilidad técnica y educativa del uso de una Raspberry Pi como servidor web funcional dentro de un entorno académico, promoviendo el aprendizaje experimental y la integración de conocimientos en redes, sistemas operativos y servicios web.

En este contexto, la justificación del estudio se sustenta en la necesidad de disponer de plataformas reales que repliquen escenarios productivos a bajo costo, fortaleciendo las competencias tecnológicas de los estudiantes en instituciones de educación técnica y superior.



2. MATERIALES Y MÉTODOS

El proceso de implementación se estructuró en cuatro etapas principales: selección de hardware y software, instalación y configuración, desarrollo del entorno web y evaluación del rendimiento del sistema.

- Selección de hardware y software:**
 Se utilizó una Raspberry Pi 4 Model B con 4 GB de memoria RAM, procesador Broadcom BCM2711 de cuatro núcleos a 1.5 GHz y almacenamiento mediante tarjeta microSD de 32 GB. El sistema operativo instalado fue Raspberry Pi OS (64 bits), por su compatibilidad con servidores LAMP. Como software principal se empleó el servidor Apache, junto con PHP y MariaDB, para la gestión de contenidos y bases de datos. Para la administración remota del sistema se usaron las herramientas PuTTY (para acceso SSH) y WinSCP (para la transferencia segura de archivos).
- Instalación y configuración:**
 Se realizó la instalación limpia del sistema operativo mediante la herramienta *Raspberry Pi Imager*. Posteriormente, se configuró la red en modo Ethernet con dirección IP estática y se habilitaron los servicios SSH y HTTP. Apache, PHP y MariaDB se instalaron a través del gestor de paquetes apt. Finalmente, se verificó la correcta ejecución del servicio web mediante acceso local y remoto desde distintos dispositivos de prueba.
- Desarrollo de la página web:**
 Se diseñaron interfaces básicas con HTML y CSS, priorizando la estructura modular y la navegación intuitiva. Las pruebas de visualización se realizaron en navegadores web (Firefox y Chromium) para asegurar la compatibilidad multiplataforma.
- Evaluación del rendimiento:**
 Esta etapa tuvo como objetivo medir la capacidad de respuesta y estabilidad del servidor bajo diferentes condiciones de carga.

Para ello, se emplearon las siguientes métricas y herramientas:

- Tiempo de respuesta promedio (ms):**
 medido con la herramienta ab (Apache Benchmark), ejecutando 500 y 1000 peticiones concurrentes desde un equipo cliente conectado a la misma red local.
- Rendimiento en solicitudes por segundo (req/s):**
 obtenido del mismo análisis con ab, determinando la tasa máxima sostenida que el servidor pudo manejar sin errores.
- Uso de CPU y memoria:**
 registrado con el comando top durante la ejecución de las pruebas de carga, para observar el comportamiento de los recursos del sistema.
- Latencia de red y disponibilidad:**
 evaluadas mediante ping y uptime, con el fin de identificar tiempos de respuesta irregulares o caídas en el servicio.

Los resultados de estas pruebas se registraron en una hoja de cálculo, donde se calcularon promedios y desviaciones estándar para cada métrica. De este modo, se garantizó una comparación objetiva entre los diferentes escenarios de carga.

Complementariamente, se aplicó una encuesta de usabilidad a un grupo de 10 estudiantes del área de redes y sistemas, quienes interactuaron con la interfaz web implementada. Se evaluaron aspectos como facilidad de acceso, velocidad percibida, claridad del contenido y utilidad educativa, utilizando una escala tipo Likert de 1 a 5.

Esta combinación de pruebas cuantitativas y cualitativas permitió obtener una visión integral del desempeño del servidor y su aplicabilidad en entornos académicos, la figura 1 muestra el inicio de la instalación del SO.

Figura 1: Imagen de instalación del SO Raspberry Pi





Fuente: El autor

3. RESULTADOS

El primer paso para instalar un servicio web en la Raspberry Pi estuvo marcado por la elección del sistema operativo (SO). Se utilizó Raspberry Pi OS (64 bits) con entorno gráfico (GUI), lo que facilitó el proceso de instalación y la posterior configuración de servicios. Durante las pruebas iniciales se verificó la correcta detección del hardware, incluyendo conectividad Ethernet y estabilidad térmica (temperatura promedio: 46 °C en reposo, 52 °C en carga).

El servidor Apache fue instalado y configurado satisfactoriamente, permitiendo la gestión estable de peticiones HTTP. La Figura 2 muestra el código utilizado durante la instalación en el entorno Raspberry Pi.

Figura 2. Instalación de Apache en Raspberry Pi

Fuente: El autor

Posteriormente, se establecieron los enlaces simbólicos necesarios para el arranque automático del servicio y se configuró la página inicial en el directorio /var/www/html. Para el desarrollo del sitio se adaptó una plantilla HTML y CSS, modificada para incluir una barra de navegación, secciones de información académica y enlaces a recursos institucionales. A continuación, se presenta un ejemplo del código HTML utilizado para la creación de la barra de navegación.

Código HTML Ejemplo

```
<!-- Navbar Start -->
<div class="container-fluid position-relative
nav-bar p-0">
<div class="container-lg position-relative p-0
px-lg-3" style="z-index: 9;">
<nav class="navbar navbar-expand-lg bg-light
navbar-light shadow-lg py-3 py-lg-0 pl-3 pl-lg-
5">
<a href="" class="navbar-brand">
<!-- <h1 class="m-0 text-primary"><span
class="text-dark">TRAVEL</span>ER</h1> -->

</a>
<button type="button" class="navbar-toggler"
data-toggle="collapse" data-
target="#navbarCollapse">
<span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse justify-
content-between px-3" id="navbarCollapse">
<div class="navbar-nav ml-auto py-0">
<a href="index.html" class="nav-item nav-link
active">Inicio</a>
<a href="#nosotros-area" class="nav-item nav-
link">Nosotros</a>
<a href="#servicios_area" class="nav-item nav-
link">Servicios</a>
<a href="#galeria_area" class="nav-item nav-
link">Galería</a>
<a href="#nosotros-area" class="nav-item nav-
link">Contáctenos</a>
</div>
</div>
```

```
pi@raspberrypi:~$ sudo apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbg
Suggested packages:
  apache2-doc apache2-ssl-cert apache2-ssl-cert-bin apache2-ssl-cert-bin
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
0 upgraded, 6 newly installed, 0 to remove and 7 not upgraded.
Need to get 2,069 kB of archives.
After this operation, 13.6 MB of additional disk space will be used.
```

</nav>

</div>

</div>

La interfaz resultante, mostrada en la Figura 4, fue diseñada con un enfoque funcional, priorizando la claridad visual y el acceso rápido a los contenidos.

Figura 4. Landing page de Diseño y Mantenimiento de Redes





Fuente: El autor

Evaluación del rendimiento del servidor

Para validar la capacidad operativa del sistema se realizaron pruebas de desempeño con la herramienta ab (Apache Benchmark). Las pruebas se ejecutaron desde un cliente conectado a la misma red local, utilizando 500 y 1000 solicitudes HTTP concurrentes. Los resultados obtenidos se resumen en la Tabla 1.

Tabla 1. Resultados de las pruebas de rendimiento del servidor Apache en Raspberry Pi

Prueba	Nº de solicitudes	Tiempo medio de respuesta (ms)	Solicitudes por segundo (req/s)	Uso promedio de CPU (%)	Uso promedio de RAM (%)
1	500	118	84.6	32	41
2	1000	137	72.9	45	49

Fuente: Elaboración propia con datos obtenidos mediante Apache Benchmark y top.

Los resultados muestran que el servidor mantuvo una respuesta estable y un uso moderado de recursos incluso bajo condiciones de carga simultánea. El tiempo de respuesta promedio fue inferior a 150 ms y el uso máximo de CPU no superó el 50 %, lo que evidencia una gestión eficiente de procesos en el entorno ARM de la Raspberry Pi.

El monitoreo del rendimiento mediante los comandos top y htop confirmó que el servicio

Apache no generó sobrecarga significativa y que los procesos de red se mantuvieron estables durante todas las pruebas. Además, no se registraron errores de conexión ni caídas del servicio, con una disponibilidad del 100 % durante las 8 horas de observación.

En cuanto al consumo energético, la Raspberry Pi 4 mostró una demanda promedio de 3.4 W en reposo y 6.2 W en carga, lo que representa un ahorro aproximado del 92 % respecto a un servidor convencional de escritorio (70–80 W). Estos valores refuerzan su idoneidad para entornos educativos con limitaciones energéticas o presupuestarias.

Finalmente, se aplicó una encuesta de usabilidad a 10 estudiantes del área de redes, con el fin de evaluar la facilidad de uso y la percepción del desempeño del sistema. Los resultados se presentan en la Tabla 2.

Tabla 2. Resultados de la encuesta de usabilidad (escala 1–5)

Criterio evaluado	Promedio
Facilidad de acceso	4.7
Velocidad percibida	4.5
Claridad del contenido	4.8
Utilidad educativa	4.9

Los usuarios destacaron la fluidez de navegación y la rapidez de respuesta del servidor, coincidiendo con los resultados técnicos obtenidos.

En conjunto, las pruebas cuantitativas y cualitativas demostraron que la Raspberry Pi 4 es una alternativa viable y eficiente como servidor web educativo, ofreciendo un desempeño adecuado, bajo consumo energético y facilidad de implementación en aulas de aprendizaje práctico.

4. DISCUSIÓN

Los resultados obtenidos en la implementación del servidor web sobre Raspberry Pi confirman que esta



plataforma constituye una alternativa viable y eficiente para el alojamiento de páginas web en entornos educativos, tal como lo han señalado otros estudios que destacan su bajo consumo energético y su estabilidad operativa [17][3]. La instalación del sistema operativo Raspberry Pi OS y la configuración del servidor Apache se realizaron sin inconvenientes, manteniendo un desempeño estable incluso bajo cargas moderadas, lo cual coincide con lo reportado por [17], quienes demostraron la eficacia de esta arquitectura para ejecutar entornos LAMP destinados a aplicaciones educativas.

Un aspecto técnico relevante observado en este trabajo es el uso eficiente de los recursos del sistema, con un consumo de CPU que rara vez superó el 50% y tiempos de respuesta inferiores a 150 ms en las pruebas de carga. Este comportamiento es coherente con los hallazgos de [18], quienes señalaron que las herramientas de análisis y optimización web no afectan de forma significativa el tiempo de carga o el First Contentful Paint (FCP), lo que sugiere que las limitaciones del hardware de la Raspberry Pi no impactan de manera drástica la experiencia del usuario cuando el servidor está correctamente configurado.

En comparación con sistemas tradicionales de alojamiento, la Raspberry Pi presenta ventajas económicas y energéticas sustanciales. Su consumo promedio entre 3 y 6 W representa una fracción mínima frente al de servidores convencionales, lo que respalda su utilización en instituciones educativas o proyectos comunitarios donde los recursos financieros y energéticos son limitados. Esta observación coincide con lo descrito por [19], quienes resaltaron la eficiencia de su arquitectura compacta y su capacidad de operación continua con requerimientos mínimos de energía.

Sin embargo, uno de los principales factores limitantes identificados fue el almacenamiento, dado que el sistema depende de una tarjeta microSD, lo cual restringe la capacidad de expansión y la durabilidad del medio de almacenamiento. Esta limitación es consistente con lo discutido por [14], quienes evidenciaron que el almacenamiento limitado influye directamente en la estabilidad de los

sistemas de registro y análisis prolongado de datos. En este sentido, una opción viable para superar esta restricción sería la integración de discos externos mediante USB 3.0 o unidades SSD compactas, que permitirían ampliar el espacio de alojamiento y mejorar la fiabilidad a largo plazo.

En términos de conectividad, la elección del puerto Ethernet como medio principal de comunicación resultó adecuada para garantizar una transferencia de datos estable y con baja latencia, evitando los problemas asociados a la variabilidad de las redes inalámbricas. Este enfoque coincide con las recomendaciones de [20], quienes destacaron que la encapsulación de tramas Ethernet bajo el protocolo DTLS mejora la seguridad y la eficiencia de la comunicación entre dispositivos IoT. La experiencia práctica de este trabajo demostró que el uso de conexión cableada simplifica la configuración, reduce el retardo en las peticiones web y optimiza la estabilidad del servicio.

Por otro lado, la observación de picos de uso de CPU en ciertos intervalos de carga, como se evidenció en los gráficos de monitoreo [14], indica la necesidad de aplicar técnicas de balanceo o ajuste de servicios para maximizar el rendimiento del servidor cuando se incrementa el número de peticiones concurrentes. Aun así, estos picos no comprometieron la estabilidad general del sistema ni la accesibilidad del servicio, lo que confirma la resiliencia de la Raspberry Pi ante escenarios de estrés moderado.

Finalmente, los resultados obtenidos validan la pertinencia pedagógica y técnica de la propuesta. La plataforma no solo cumplió con los objetivos de funcionalidad y estabilidad, sino que además promueve el aprendizaje experimental de tecnologías web, servidores y redes, en concordancia con lo planteado por [7] y [5], quienes destacan la importancia de integrar herramientas de bajo costo en los procesos de enseñanza de sistemas y control.

En conclusión, este estudio demuestra que la Raspberry Pi puede desempeñar un rol central en la formación tecnológica aplicada, ofreciendo un



equilibrio adecuado entre costo, rendimiento y consumo energético. No obstante, se recomienda continuar con investigaciones futuras que incluyan pruebas de escalabilidad, comparaciones con otras arquitecturas de microservidores y la integración de sistemas de almacenamiento externo para superar las limitaciones detectadas en esta experiencia.

5. CONCLUSIÓN

La implementación de un servidor web en una Raspberry Pi resultó en una solución eficiente y accesible para el alojamiento de páginas informativas. La configuración del servidor y la optimización de la interfaz web permitieron una experiencia fluida para los usuarios. [1]

El análisis del rendimiento evidenció tiempos de carga adecuados y estabilidad operativa. La Raspberry Pi demostró ser una opción viable para entornos educativos, con un consumo energético reducido y un costo accesible. Sin embargo, se recomienda considerar el almacenamiento disponible y realizar mantenimiento periódico para evitar limitaciones a futuro. [2]

En conclusión, la Raspberry Pi ofrece una plataforma adaptable y funcional para la gestión de contenido web, representando una alternativa efectiva para instituciones con presupuestos limitados que buscan digitalizar su infraestructura informativa. [3]

6. AGRADECIMIENTO

Expresamos nuestro más sincero agradecimiento al Instituto Superior Tecnológico “Carlos Cisneros” por su invaluable apoyo en la ejecución de este proyecto. Su compromiso con la innovación y el desarrollo tecnológico ha sido de gran ayuda para llevar a cabo esta investigación, proporcionando un entorno propicio para la experimentación y el aprendizaje. [4]

De manera especial, queremos reconocer la dedicación y el esfuerzo de los docentes y estudiantes que participaron activamente en las pruebas y evaluaciones de este proyecto. Su entusiasmo, disposición y valiosa retroalimentación han sido un fuerte bastión para identificar áreas de mejora y optimizar el funcionamiento del sistema implementado. La colaboración de todos los involucrados ha enriquecido este estudio, [5] [6] permitiendo alcanzar resultados más robustos y confiables.

Asimismo, agradecemos a todos aquellos que, de manera directa o indirecta, contribuyeron con sus conocimientos, sugerencias y apoyo técnico. Este proyecto no habría sido posible sin el trabajo en equipo y la sinergia entre todos los participantes. Esperamos que los frutos de esta investigación sirvan como un gran aporte para la comunidad educativa y tecnológica, y que inspire futuros proyectos innovadores.

7. REFERENCIAS BIBLIOGRÁFICAS

- M. J. López, X. Continente, E. Sánchez, and M. Bartoli, “Nota metodológica Intervenciones que incluyen webs y redes sociales: herramientas e indicadores para su evaluación Activities using websites and social networks: tools and indicators for evaluation,” *Gac Sanit*, vol. 31, no. 4, pp. 346–348, 2017, doi: 10.1016/j.gaceta.2016.12.006.
- S. Wisnusenna, M. S. Yonatan, A. Wibisurya, Fanny, and T. Yuwono, “Model of Web Based Application to Control Bridge Traveler Using Raspberry Pi,” *Procedia Comput Sci*, vol. 135, pp. 518–525, Jan. 2018, doi: 10.1016/J.PROCS.2018.08.204.
- V. Vujović and M. Maksimović, “Raspberry Pi as a Sensor Web node for home automation,” *Computers & Electrical Engineering*, vol. 44, pp. 153–171, May 2015, doi: 10.1016/J.COMPELECENG.2015.01.019.
- F. Gómez-Estern, M. López-Martínez, and D. Muñoz de la Peña, “Sistema de Evaluación Automática VíaWeb en Asignaturas Prácticas de Ingeniería,” *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 7, no. 3, pp. 111–119, Jul. 2010, doi: 10.1016/S1697-7912(10)70047-9.
- P. Basanta-Val, M. García-Valls, and P. López-Anastasio, “Herramienta Web Ligera para La Programación en C-Concurrente,” *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 10, no. 4, pp. 465–476, Oct. 2013, doi: 10.1016/J.RIAI.2013.05.010.
- A. Valero-Gómez, P. de la Puente, D. Rodriguez-Losada, M. Hernando, and P. S. Segundo, “Arquitectura de integración basada en Servicios



- Web para sistemas heterogéneos y distribuidos: aplicación a robots móviles interactivos," *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 10, no. 1, pp. 85–95, Jan. 2013, doi: 10.1016/J.RIAI.2012.11.008.
- [7] M. Goubej and L. Bláha, "Raspberry Pi-based Motion Control Testbed for Mechatronics Education," *IFAC-PapersOnLine*, vol. 55, no. 17, pp. 285–290, Jan. 2022, doi: 10.1016/J.IFACOL.2022.09.293.
- [8] R. K. Jain, "Experimental performance of smart IoT-enabled drip irrigation system using and controlled through web-based applications," *Smart Agricultural Technology*, vol. 4, p. 100215, Aug. 2023, doi: 10.1016/J.ATECH.2023.100215.
- [9] M. M. Islam *et al.*, "A deep learning model for cotton disease prediction using fine-tuning with smart web application in agriculture," *Intelligent Systems with Applications*, vol. 20, p. 200278, Nov. 2023, doi: 10.1016/J.ISWA.2023.200278.
- [10] Z. Peng, J. Li, and H. Hao, "Development and experimental verification of an IoT sensing system for drive-by bridge health monitoring," *Eng Struct*, vol. 293, p. 116705, Oct. 2023, doi: 10.1016/J.ENGSTRUCT.2023.116705.
- [11] A. Morchid *et al.*, "IoT-enabled fire detection for sustainable agriculture: A real-time system using flask and embedded technologies," 2024, doi: 10.1016/j.rineng.2024.102705.
- [12] M. A. A. Radia, M. K. E. Nimr, and A. S. Atlam, "IoT-based wireless data acquisition and control system for photovoltaic module performance analysis," *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, vol. 6, p. 100348, Dec. 2023, doi: 10.1016/J.PRIME.2023.100348.
- [13] N. Kortas, H. Azzabi, A. Ben Arbia, and J. B. Tahar, "Communication within Cloudlet using the Raspberry," *Procedia Comput Sci*, vol. 73, pp. 193–198, Jan. 2015, doi: 10.1016/J.PROCS.2015.12.012.
- [14] R. Dhuny, A. A. I. Peer, N. A. Mohamudally, and N. Nissanke, "Performance evaluation of a portable single-board computer as a 3-tiered LAMP stack under 32-bit and 64-bit Operating Systems," *Array*, vol. 15, p. 100196, Sep. 2022, doi: 10.1016/J.ARRAY.2022.100196.
- [15] P. Casado, J. M. Blanes, C. Torres, C. Orts, D. Marroquí, and A. Garrigós, "Raspberry Pi based photovoltaic I-V curve tracer," *HardwareX*, vol. 11, p. e00262, Apr. 2022, doi: 10.1016/J.OHX.2022.E00262.
- [16] A. Fathy, A. Ben Atitallah, D. Yousri, H. Rezk, and M. Al-Dhaifallah, "A new implementation of the MPPT based raspberry Pi embedded board for partially shaded photovoltaic system," *Energy Reports*, vol. 8, pp. 5603–5619, Nov. 2022, doi: 10.1016/J.EGYR.2022.04.035.
- [17] R. Dhuny and N. A. Mohamudally, "RPI64Box: A portable 3-tiered LAMP stack in a 64-bit Operating System environment," *Software Impacts*, vol. 14, p. 100390, Dec. 2022, doi: 10.1016/J.SIMPA.2022.100390.
- [18] C. Petalotis, L. Krumpak, M. S. Floroiu, L. F. Ahmad, S. Athreya, and I. Malavolta, "An empirical study on the performance and energy costs of ads and analytics in mobile web apps," *Inf Softw Technol*, vol. 166, p. 107370, Feb. 2024, doi: 10.1016/J.INFSOF.2023.107370.
- [19] A. Valera, A. Soriano, and M. Vallés, "Plataformas de Bajo Coste para la Realización de Trabajos Prácticos de Mecatrónica y Robótica," *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 11, no. 4, pp. 363–376, Oct. 2014, doi: 10.1016/J.RIAI.2014.09.002.
- [20] K. Rzepka, P. Szary, K. Cabaj, and W. Mazurczyk, "Performance evaluation of Raspberry Pi 4 and STM32 Nucleo boards for security-related operations in IoT environments," 2024, doi: 10.1016/j.comnet.2024.110252.



ANEXO

Evaluación del rendimiento del servidor

Para validar la capacidad operativa del sistema implementado en la Raspberry Pi, se realizaron pruebas de desempeño utilizando la herramienta Apache Benchmark (ab), incluida en el paquete apache2-utils. Esta herramienta permite medir la capacidad de respuesta de un servidor web frente a múltiples solicitudes concurrentes, obteniendo métricas como el tiempo medio de respuesta, el número de solicitudes procesadas por segundo, y el comportamiento general del hardware bajo carga.

Especificación del servidor Raspberry Pi

- Se utilizó una Raspberry Pi 4 Model B con 4 GB de memoria RAM, procesador Broadcom BCM2711 de cuatro núcleos a 1.5 GHz y almacenamiento mediante tarjeta microSD de 32 GB.
- Sistema operativo Raspberry Pi OS (64 bits), por su compatibilidad con servidores LAMP. Como software principal se empleó el servidor Apache, junto con PHP y MariaDB.
- Configuración de red detallada
- Velocidad de conexión (por ejemplo, LAN 100 Mbps).
- Dirección IP del servidor 192.168.1.10
- Indicar si se usó conexión directa o a través de switch/router.
- Latencia promedio de la red local (resultado del comando ping).
- Comandos exactos de prueba (con versión de Apache Benchmark)
- Ejemplo:
ab -n 500 -c 50 http://192.168.1.10/index.html
ab -n 1000 -c 100 http://192.168.1.10/index.html
- Indicar versión: Apache Benchmark 2.3 <r1879490>.
- Aclarar número de repeticiones y promedio final reportado.
- Procedimientos de monitoreo de recursos (top/htop)
- Parámetros monitoreados: %CPU, %MEM, load average, uptime.

- Frecuencia de registro: cada 10 segundos durante la ejecución de ab.
- Herramienta usada para registro automático (top -b -d 10 -n 30 > monitoreo.txt).
- Tiempo total de monitoreo.

Las pruebas se ejecutaron desde un equipo cliente conectado a la misma red local, con el fin de eliminar la influencia de la latencia externa y centrarse exclusivamente en el rendimiento del hardware y del servicio HTTP. Se configuraron dos escenarios de prueba, con 500 y 1000 solicitudes totales, y concurrencias de 50 y 100 respectivamente. Los comandos utilizados fueron los siguientes:

ab -n 500 -c 50 http://192.168.1.10/

ab -n 1000 -c 100 http://192.168.1.10/

Cada escenario se ejecutó tres veces consecutivas para garantizar la consistencia de los resultados, promediándose los valores obtenidos y descartando posibles valores atípicos. Los resultados se resumen en la Tabla 1.

Tabla 1. Resultados de las pruebas de rendimiento del servidor Apache en Raspberry Pi

Pru eba	Nº de solicit udes	Tiem po medi o de respu esta (ms)	Solicit udes por segun do (req/s)	Uso prom edio de CPU (%)	Uso prom edio de RAM (%)
1	500	118	84.6	32	41
2	1000	137	72.9	45	49

Fuente: Elaboración propia con datos obtenidos mediante Apache Benchmark y top.

Monitoreo de recursos del sistema

Durante la ejecución de las pruebas de carga, se implementó un procedimiento de monitoreo de recursos para registrar el comportamiento del sistema en tiempo real. Se emplearon las herramientas top y htop, las cuales permiten obtener datos dinámicos sobre el uso del procesador, memoria RAM, procesos activos y carga promedio del sistema.

El monitoreo se realizó de la siguiente manera:



- Parámetros observados:
- Porcentaje de uso de CPU (%CPU).
- Porcentaje de uso de memoria RAM (%MEM).
- Carga promedio del sistema (load average).
- Procesos activos relacionados con Apache (PID, estado y consumo de recursos).
- Frecuencia de registro:
- Los valores fueron capturados cada 5 segundos durante la ejecución de las pruebas con ab, utilizando el siguiente comando:
top -b -d 5 -n 60 >> monitoreo.log
- En pruebas complementarias se empleó htop para una visualización más detallada del consumo de recursos y detección de posibles cuellos de botella.

Los datos registrados fueron analizados posteriormente para calcular los promedios de uso de CPU y memoria mostrados en la Tabla 1. Este procedimiento permitió correlacionar el rendimiento del servicio web con el consumo de recursos del sistema, garantizando una evaluación integral del desempeño.

Instrumento empleado

Para la recolección de datos se utilizó un cuestionario estructurado diseñado específicamente para evaluar la percepción y nivel de satisfacción de los estudiantes respecto al uso de la plataforma Moodle. El instrumento fue elaborado tomando como referencia investigaciones previas sobre adopción de entornos virtuales de aprendizaje y satisfacción estudiantil en educación superior.

El cuestionario estuvo conformado por un total de 24 ítems distribuidos en cuatro dimensiones principales:

1. Usabilidad de la plataforma, que evalúa la facilidad de uso, navegación y acceso a los recursos.
2. Funcionalidad académica, que analiza la efectividad de las herramientas disponibles (tareas, cuestionarios, foros, etc.) en los procesos de enseñanza-aprendizaje.
3. Interacción docente-estudiante, que mide la comunicación y acompañamiento pedagógico dentro del entorno virtual.

4. Satisfacción general, que valora la percepción global sobre la utilidad de Moodle en el desarrollo académico.

Escala utilizada

Los ítems fueron medidos mediante una escala tipo Likert de cinco puntos, con los siguientes valores asignados:

- 1 = Totalmente en desacuerdo,
- 2 = En desacuerdo,
- 3 = Ni de acuerdo ni en desacuerdo,
- 4 = De acuerdo,
- 5 = Totalmente de acuerdo.

Esta escala permitió cuantificar las percepciones de los estudiantes de manera uniforme y facilitar el análisis estadístico descriptivo de las respuestas.

Validación del instrumento

Previo a su aplicación, el cuestionario fue sometido a un proceso de validación de contenido por parte de tres expertos en pedagogía y tecnologías educativas, quienes revisaron la redacción, pertinencia y coherencia de los ítems respecto a los objetivos de la investigación. Posteriormente, se realizó una prueba piloto con 25 estudiantes del ISU Carlos Cisneros, pertenecientes a distintas carreras técnicas, con el propósito de verificar la claridad de las preguntas y el tiempo promedio de respuesta.

Los resultados de la prueba piloto se sometieron a un análisis de confiabilidad interna mediante el coeficiente Alfa de Cronbach, obteniéndose un valor de $\alpha = 0.89$, lo que indica una alta consistencia interna del instrumento.

Población y Muestra

La población total estuvo conformada por 2.000 estudiantes matriculados en el Instituto Superior Universitario Carlos Cisneros (ISUCC) durante el período académico abril-agosto 2024.

De esta población, 825 estudiantes respondieron la encuesta, representando el 41,2% del total. El muestreo fue no probabilístico por conveniencia, considerando la accesibilidad y disposición de los estudiantes para participar voluntariamente.

Procedimiento de Aplicación



La encuesta se aplicó en línea durante el mes de agosto de 2024, garantizando el anonimato y la confidencialidad de los participantes.

La duración promedio de respuesta fue de 10 a 12 minutos, y los datos fueron exportados a formato .csv para su posterior análisis estadístico.

Análisis de Datos

Los datos recolectados fueron procesados mediante IBM SPSS Statistics v25 y Microsoft Excel 2024, utilizando estadística descriptiva (frecuencias, medias, desviación estándar) para analizar las tendencias de respuesta.

Se verificó la distribución normal de los puntajes mediante la prueba de Kolmogórov–Smirnov, y se identificaron correlaciones entre las dimensiones “Usabilidad” y “Satisfacción General”.

Análisis e interpretación de resultados

Los resultados evidencian que la Raspberry Pi mantiene un rendimiento estable bajo cargas de hasta 1000 solicitudes concurrentes, con un incremento controlado en el uso de CPU y memoria. El tiempo medio de respuesta se mantiene por debajo de los 150 ms, lo que indica una adecuada capacidad de procesamiento para entornos de uso académico o institucional.

En consecuencia, se demuestra la viabilidad del uso de la Raspberry Pi como servidor web de bajo costo en entornos educativos, siempre que se apliquen estrategias de optimización y control de carga adecuadas.

Análisis descriptivo

Los resultados evidencian una tendencia positiva en la percepción del estudiantado, destacando especialmente los ítems relacionados con la facilidad de uso, la contribución al aprendizaje y la satisfacción general, cuyos valores promedio superan los 4 puntos sobre 5.

Por otro lado, los ítems vinculados con la estabilidad de la plataforma y la comunicación con los docentes presentan valores medios ligeramente menores, indicando áreas de mejora relacionadas con la infraestructura tecnológica y la interacción pedagógica en entornos virtuales.

