

# From web platform to decision: a comprehensive approach for CNET in La Maná, Ecuador

Diginomics.

2022; 1:15

DOI: 10.56294/digi202215

ISSN: 3072-8428

## *De la plataforma web a la decisión: un enfoque integral para CNET en La Maná, Ecuador*

Franco Joel Lloacana Llugsa<sup>1</sup>, Steven Ricardo Castillo Calderón<sup>1</sup>

<sup>1</sup>Universidad Técnica de Cotopaxi, Facultad de Ciencias de la Ingeniería y Aplicadas, Carrera de Ingeniería en Sistemas de información. La Maná-Ecuador.

**Citar como:** Lloacana Llugsa FJ, Castillo Calderón SR. From web platform to decision: a comprehensive approach for CNET in La Maná, Ecuador. Diginomics. 2022; 1:15. <https://doi.org/10.56294/digi202215>

### ABSTRACT

The document presented a comprehensive framework for designing and building a web application geared towards sales control and decision-making. It began with the consolidation of the web as infrastructure, interface and business space, and characterised the technological platform (HTML5, CSS3, JavaScript, HTTP/2, HTTPS) together with client-server architectures, PWA and WebAssembly. It analysed application types (static, dynamic, portals, animated) and linked them to contemporary patterns—three layers, microservices, serverless and PWA—evaluating their compromises in terms of modularity, deployment and maintenance. On that basis, he described the proposed stack: front-end with web standards and frameworks (React/Angular/Vue), back-end with PHP/Laravel or Python/Django, and relational persistence with MariaDB for its performance and compatibility with MySQL. He also considered the tactical use of WordPress when speed and extensibility were paramount, defining its limits in relation to transactional and analytical needs. Methodologically, he justified the adoption of an agile approach with Scrum for the CNET case in Cantón La Maná, highlighting incremental delivery, uncertainty management, and early feedback. He linked the development cycle with DevOps practices (CI/CD, automated testing, and infrastructure as code) to reduce time-to-value. At the analytical layer, he articulated a Business Intelligence model that integrated ETL processes, OLTP/OLAP data warehouses, dimensional modelling, and architectural options (traditional, modern, cloud, and distributed), proposing a hybrid deployment: central data governance and data marts by domain. Finally, it outlined a roadmap for security, observability and augmented analytics (forecasting, segmentation, anomaly detection) and recognised common risks — heterogeneous data quality and technical debt—as variables that can be managed through governance and continuous training.

**Keywords:** Web Architecture; Agile Methodologies; Business Intelligence; MariaDB; ETL.

### RESUMEN

El documento presentó un marco integral para diseñar y construir una aplicación web orientada al control de ventas y a la toma de decisiones. Partió de la consolidación de la web como infraestructura, interfaz y espacio de negocio, y caracterizó la plataforma tecnológica (HTML5, CSS3, JavaScript, HTTP/2, HTTPS) junto con arquitecturas cliente-servidor, PWA y WebAssembly. Analizó tipologías de aplicaciones (estáticas, dinámicas, portales, animadas) y las conectó con patrones contemporáneos —tres capas, microservicios, serverless y PWA— evaluando sus compromisos en modularidad, despliegue y mantenimiento. Sobre esa base, describió el stack propuesto: front-end con estándares web y frameworks (React/Angular/Vue), back-end con PHP/Laravel o Python/Django, y persistencia relacional con MariaDB por su rendimiento y compatibilidad con MySQL. Consideró además el uso táctico de WordPress cuando primó la rapidez y extensibilidad, delimitando sus límites frente a necesidades transaccionales y analíticas. Metodológicamente, justificó la adopción de un enfoque ágil con Scrum para el caso CNET del Cantón La Maná, destacando entrega incremental, gestión de incertidumbre y retroalimentación temprana. Vinculó el ciclo de desarrollo con prácticas DevOps (CI/CD, pruebas automatizadas e infraestructura como código) para reducir el time-to-value. En la capa analítica, articuló un modelo de Business Intelligence que integró procesos ETL, almacenes de datos OLTP/OLAP, modelado dimensional y opciones arquitectónicas (tradicional, moderna, nube y distribuida), proponiendo un despliegue híbrido: gobierno central del dato y datamarts por dominio. Finalmente, delineó una hoja de ruta de seguridad, observabilidad y analítica aumentada (pronósticos, segmentación, detección de anomalías) y reconoció riesgos habituales —calidad de datos heterogénea y deuda técnica— como variables gestionables mediante gobernanza y capacitación continua.

**Palabras clave:** Arquitectura Web; Metodologías Ágiles; Business Intelligence; MariaDB; ETL.

## INTRODUCCIÓN

El vertiginoso avance de las tecnologías de la información y la comunicación (TIC) ha redefinido de manera estructural los procesos productivos, las formas de interacción social y los modelos de gestión organizacional. En este escenario, la web —como plataforma tecnológica y como ecosistema de estándares, lenguajes y servicios— opera simultáneamente como infraestructura, como interfaz y como espacio de negocio. La rápida expansión de Internet y sus aplicaciones, sumada a la consolidación de arquitecturas distribuidas, ha desplazado definitivamente a los enfoques monolíticos, impulsando soluciones escalables, seguras y orientadas al dato. En el ámbito empresarial, esta transición tiene una expresión concreta: el control eficiente de ventas y la toma de decisiones basada en evidencia, dos capacidades que, integradas en una aplicación web moderna, permiten mejorar la eficiencia operativa, optimizar la experiencia de usuario y sostener ventajas competitivas en entornos dinámicos.

El presente desarrollo sitúa esa problemática en un marco conceptual y técnico integral. Primero, caracteriza la plataforma web como un conjunto de tecnologías (HTML5, CSS3, JavaScript/ECMAScript, APIs del navegador, HTTP/2, HTTPS) y servicios (alojamiento, bases de datos, escalado en la nube) que habilitan aplicaciones accesibles, seguras y multiplataforma. Luego, revisa los fundamentos de la arquitectura cliente–servidor, subrayando sus implicaciones en rendimiento, interoperabilidad y seguridad (autenticación, cifrado, control de acceso), así como el rol del cliente web (navegadores modernos) en la adopción de estándares (PWA, WebAssembly) que elevan las expectativas de usabilidad y desempeño. En paralelo, se examinan tipologías de aplicación —estáticas, dinámicas, animadas, portales— y sus atributos en términos de tiempos de carga, personalización, escalabilidad y costes, para enlazarlas con patrones arquitectónicos contemporáneos: tres capas, microservicios, serverless y aplicaciones web progresivas, cada uno con sus trade-offs de modularidad, despliegue y mantenimiento.

Sobre esa base se discuten las tecnologías de programación que articulan la solución: lenguajes de presentación e interacción (HTML5, CSS3, JavaScript con marcos como React/Angular/Vue), lenguajes y frameworks de servidor (PHP con Laravel/Symfony, Python con Django/Flask) y componentes de datos (MariaDB como evolución abierta y de alto rendimiento de MySQL). Se incorporan, además, paradigmas de integración (APIs REST y GraphQL) y prácticas de seguridad aplicadas a APIs. El documento también considera plataformas de gestión de contenidos como WordPress cuando los requerimientos privilegian rapidez de puesta en marcha, extensibilidad por plugins y un backoffice maduro, sin perder de vista sus límites frente a necesidades de analítica avanzada o flujos transaccionales específicos.

En términos metodológicos, se propone un análisis comparativo de modelos de desarrollo (cascada, ágil, DevOps, prototipos, espiral), justificando la selección de un enfoque ágil con Scrum para el caso de uso “Desarrollo de una Aplicación Web para el Control de Ventas y la Toma de Decisiones en la Corporación Nacional de Emprendedores y Telecomunicaciones – CNET del Cantón La Maná”. Tal selección se sustenta en la entrega incremental de valor, la gestión de la incertidumbre de requisitos y la incorporación temprana de retroalimentación de los usuarios. Finalmente, se aborda la capa decisional mediante Business Intelligence (BI): procesos ETL, almacenes de datos

(OLTP/OLAP), modelos dimensionales y arquitecturas de BI (tradicional, moderna, en la nube y distribuida), junto con modelos de implementación (centralizado, descentralizado, híbrido, cloud) y un panorama de herramientas (ETL, DBMS, visualización, reportería, minería de datos). El objetivo último es articular una propuesta técnicamente sólida y gobernable que conecte la captura transaccional de ventas con tableros analíticos confiables para la toma de decisiones estratégicas y operativas.

## DESARROLLO

### Antecedentes

Las tecnologías de información y comunicación (TIC) representan uno de los factores determinantes para el desarrollo y crecimiento económico de los países. Un factor que requiere infraestructura y personas con las habilidades que les permitan utilizar eficazmente estas tecnologías globales. La aparición de lo que en su momento llamaron “Nuevas Tecnologías” en las últimas décadas del siglo XX, ha tenido una incidencia creciente en todos los ámbitos de nuestra vida. Las TIC y en particular Internet se están expandiendo e integrando rápidamente en la vida de los ciudadanos.

En menos de diez años, se han demostrado los efectos que internet y sus diversas aplicaciones tienen en la vida de las personas, empresas, instituciones y gobiernos. Además, la forma en que las personas se comunican, se organizan, trabajan o se divierten ha cambiado mucho. En el entorno empresarial actual, el control eficiente de ventas y la toma de decisiones informadas son elementos fundamentales para el éxito y la competitividad de las organizaciones. Se ha identificado la necesidad de mejorar los procesos en estas áreas para optimizar sus operaciones y ofrecer servicios más efectivos. El desarrollo de una aplicación web especializada para el control de ventas y la toma de decisiones se presentan como una solución viable para abordar este desafío.

### *¿Qué es una Plataforma Web?*

Una plataforma web se refiere a un conjunto de tecnologías, herramientas y servicios que permiten el desarrollo, despliegue y gestión de aplicaciones y servicios en la web. Una plataforma web proporciona un entorno en línea donde los desarrolladores pueden crear y ejecutar aplicaciones web sin tener que preocuparse por aspectos como la infraestructura, la seguridad o el escalado.<sup>(1)</sup>

Estas plataformas ofrecen servicios como alojamiento web, bases de datos, herramientas de desarrollo y API (interfaz de programación de aplicaciones) que facilitan la construcción y gestión de aplicaciones web. Las plataformas web han evolucionado para ser más flexibles y escalables. Con la adopción de arquitecturas basadas en la nube, las plataformas web pueden aprovechar recursos computacionales distribuidos y almacenamiento escalable para satisfacer las demandas de aplicaciones con alto tráfico y grandes volúmenes de datos.<sup>(1)</sup>

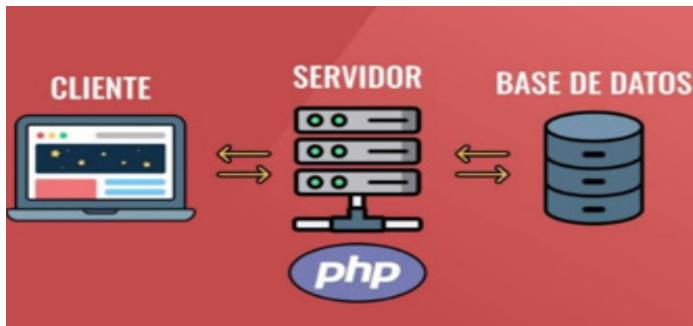
### *Características de las Aplicaciones Web*

El usuario puede acceder fácilmente a estas aplicaciones mediante el uso de un navegador web cliente o una aplicación similar, y si es a través de Internet, el usuario puede conectarse desde cualquier parte del mundo donde tenga acceso a Internet, y puede haber miles de usuarios, pero una sola aplicación instalada en un servidor, lo que permite actualizar y mantener una sola aplicación mientras todos sus usuarios ven los resultados

al instante. Utilizan tecnologías que les permiten moverse fácilmente entre plataformas. Una aplicación flash en línea, por ejemplo, puede funcionar en un Smartphone, un ordenador con Windows, Linux u otro sistema operativo.<sup>(2)</sup>

### Modelo Cliente y Servidor

El modelo Cliente y Servidor es un paradigma fundamental en el diseño de sistemas distribuidos y aplicaciones de red. Este modelo se basa en la interacción entre dos componentes principales: el cliente, que solicita servicios o recursos, y el servidor, que proporciona dichos servicios o recursos.<sup>(3)</sup>



Fuente: [https://ikastaroak.ulhi.net/edu/es/DAW/DWEC/DWEC02\\_es\\_DAW\\_DWEC02\\_Contenidos/website\\_3\\_arquitectura\\_en\\_aplicaciones\\_web.html](https://ikastaroak.ulhi.net/edu/es/DAW/DWEC/DWEC02_es_DAW_DWEC02_Contenidos/website_3_arquitectura_en_aplicaciones_web.html)

Figura 1. Arquitectura en aplicaciones Web

En el modelo Cliente y Servidor, la comunicación se realiza a través de un protocolo de red, como HTTP o TCP/IP. El cliente envía solicitudes al servidor, especificando la acción o recurso deseado, y el servidor responde con la información correspondiente. Esta arquitectura permite una distribución eficiente de tareas y recursos, ya que múltiples clientes pueden conectarse a un servidor centralizado o a varios servidores en una configuración distribuida.<sup>(3)</sup>

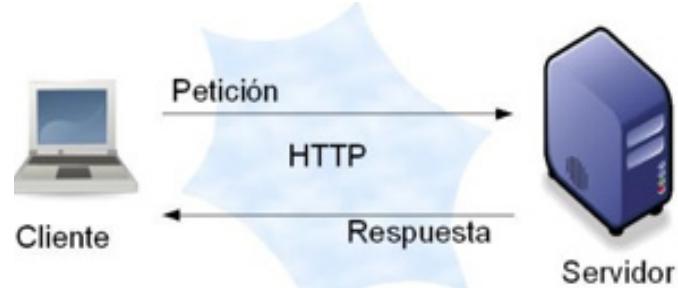
Además, la seguridad es un aspecto crítico en el modelo Cliente y Servidor. Se implementan mecanismos de autenticación y cifrado para garantizar la confidencialidad y la integridad de la comunicación entre el cliente y el servidor. Estos mecanismos contribuyen a proteger la información sensible y prevenir ataques maliciosos.<sup>(2)</sup>

### El Cliente Web

Es un componente fundamental en la arquitectura de las aplicaciones web y su importancia ha seguido creciendo. El Cliente Web se refiere al software o dispositivo utilizado por los usuarios para acceder y utilizar aplicaciones web. En el contexto de la web, los navegadores web son los clientes más comunes utilizados para acceder a sitios y aplicaciones web, los navegadores web populares, como Google Chrome, Mozilla Firefox y Safari, han continuado evolucionando para mejorar la experiencia del usuario y ofrecer un soporte más avanzado para las tecnologías web.<sup>(3)</sup>

Uno de los aspectos clave en el desarrollo del Cliente Web es la compatibilidad con los estándares y tecnologías web actuales, los navegadores web han estado trabajando en la implementación de los estándares HTML5, CSS3 y JavaScript, así como en el soporte para tecnologías emergentes como WebAssembly y Progressive Web Apps (PWA), los Clientes Web han experimentado avances en términos de rendimiento

y seguridad. Por ejemplo, la implementación de tecnologías como HTTP/2 y HTTPS ha mejorado la velocidad de carga de las aplicaciones web y la seguridad de la comunicación entre el cliente y el servidor.<sup>(3)</sup>



Fuente: <https://htascolaborativas.wordpress.com/2012/08/25/rol-del-cliente-rol-del-servidor/>

Figura 2. Rol del cliente y servidor

### Servidor Web

El Servidor Web desempeña un papel crucial en la entrega de contenido y servicios a través de Internet, y su importancia para el desarrollo de las aplicaciones web. Un Servidor Web es un software o hardware que responde a las solicitudes de los clientes web y envía las respuestas correspondientes. En los últimos años, los Servidores Web han experimentado avances significativos en términos de rendimiento, escalabilidad y seguridad, se han desarrollado y adoptado nuevas tecnologías para mejorar la eficiencia y la capacidad de respuesta de los Servidores Web.

### Interfaz de Usuario

La Interfaz de Usuario (UI, por sus siglas en inglés) es un elemento fundamental en el diseño de aplicaciones web y móviles, y su importancia ha crecido significativamente. La UI se refiere a la forma en que los usuarios interactúan con una aplicación y cómo se presenta la información en la pantalla.

Una buena interfaz de usuario debe ser visualmente atractiva, fácil de usar y fácil de entender. Debe permitir a los usuarios realizar tareas de manera eficiente y sin confusiones. Además, la interfaz debe adaptarse a diferentes dispositivos y tamaños de pantalla para proporcionar una experiencia consistente.<sup>(4)</sup>

El diseño centrado en el usuario es un enfoque clave en el desarrollo de interfaces de usuario efectivas. Esto implica comprender las necesidades, expectativas y habilidades de los usuarios para diseñar una interfaz que sea accesible y satisfactoria.

El uso de técnicas como pruebas de usabilidad y prototipos interactivos ayuda a evaluar y mejorar la calidad de la interfaz. En cuanto a las tendencias en el diseño de interfaces de usuario, el diseño minimalista y limpio ha ganado popularidad. Se enfoca en la simplicidad, la legibilidad y el uso de espacios en blanco para resaltar la información clave. Además, el diseño responsive se ha vuelto crucial, ya que las aplicaciones web y móviles deben adaptarse a diferentes dispositivos, como teléfonos inteligentes, tabletas y computadoras de escritorio.<sup>(2)</sup>

### Tipos de Aplicaciones

#### Aplicaciones Web Multiplataforma

Aplicaciones web multiplataforma se refiere a un conjunto de tecnologías y herramientas que permiten el desarrollo de

aplicaciones web que pueden ejecutarse en diferentes dispositivos y sistemas operativos, la demanda de aplicaciones web que sean compatibles con múltiples plataformas ha aumentado debido a la variedad de dispositivos y sistemas operativos utilizados por los usuarios. Una plataforma web multiplataforma utiliza tecnologías basadas en estándares web, para garantizar la portabilidad de las aplicaciones. Estas tecnologías permiten que las aplicaciones web se ejecuten en diferentes navegadores web y se adapten automáticamente a diferentes tamaños de pantalla y resoluciones. La ventaja clave de las aplicaciones web multiplataforma es su capacidad para alcanzar a una audiencia más amplia, ya que no se limitan a un sistema operativo o dispositivo específico. Además, el desarrollo y el mantenimiento de una única base de código para múltiples plataformas pueden ser más eficientes y rentables para las empresas y los desarrolladores.<sup>(5)</sup>

El estándar sobre tecnologías web que se manejan en el desarrollo de aplicaciones web multiplataforma se basa en una combinación de tecnologías y estándares ampliamente adoptados. A continuación, se detallan algunos de los elementos clave:

- HTML (HyperText Markup Language): HTML es un estándar ampliamente utilizado para estructurar y presentar el contenido de una página web. HTML5, la última versión del estándar, introdujo nuevas características y elementos, como formularios mejorados, etiquetas semánticas y soporte para multimedia.
- CSS (Cascading Style Sheets): CSS es un estándar que define cómo se presenta el contenido HTML en términos de diseño, colores, fuentes y efectos visuales. CSS3, la versión más reciente, agregó nuevas características, como transiciones, animaciones y sombras.
- JavaScript: JavaScript es un lenguaje de programación que permite agregar interactividad y funcionalidad a las aplicaciones web. ECMAScript, el estándar subyacente de JavaScript, se actualiza regularmente para introducir nuevas características y mejoras. ECMAScript (también conocido como ES6) fue una actualización importante que agregó sintaxis mejorada y características como clases y módulos.
- Web APIs: las Web APIs proporcionan una interfaz para interactuar con diferentes capacidades del navegador y del dispositivo. Algunas de las API más comunes incluyen la API de Geolocalización, la API de almacenamiento web (como localStorage y sessionStorage) y la API de notificaciones.

Estas tecnologías y estándares web forman la base para el desarrollo de aplicaciones web multiplataforma. Estos estándares son respaldados y evolucionados por organizaciones como el World Wide Web Consortium (W3C) y la Ecma International.

Además, se promueve el uso de estándares abiertos y compatibles entre diferentes navegadores para garantizar la interoperabilidad y la portabilidad de las aplicaciones web multiplataforma.<sup>(5)</sup>

#### *Aplicación Web Estática*

Las aplicaciones web estáticas se han vuelto cada vez más populares debido a su simplicidad y eficiencia en el desarrollo y despliegue de sitios web. Estas aplicaciones son aquellas en las que el contenido y la estructura de las páginas web se generan

durante el tiempo de desarrollo y no cambian dinámicamente en tiempo de ejecución.

A continuación, se presentan algunas características y ventajas de las aplicaciones web estáticas:

- Eficiencia de carga: las aplicaciones web estáticas se componen de archivos HTML, CSS y JavaScript que se sirven directamente al cliente desde un servidor web.
- Seguridad: debido a su naturaleza estática, las aplicaciones web estáticas tienen menos puntos de entrada para ataques maliciosos. Al no permitir la ejecución de código en el servidor, se reduce el riesgo de vulnerabilidades de seguridad.
- Escalabilidad: las aplicaciones web estáticas son altamente escalables, ya que se pueden servir a través de una red de entrega de contenido (CDN, por sus siglas en inglés) que distribuye el contenido en servidores ubicados en diferentes regiones geográficas.
- Bajo costo: el desarrollo y mantenimiento de aplicaciones web estáticas tienden a ser más económicos en comparación con las aplicaciones dinámicas. Al no requerir una infraestructura de base de datos compleja ni recursos de servidor dedicados para el procesamiento en tiempo real, se reducen los costos de infraestructura y de desarrollo.

#### *Aplicación Web Dinámica*

Las aplicaciones web dinámicas han experimentado un crecimiento por su capacidad de proporcionar interacción en tiempo real y contenido personalizado a los usuarios. A diferencia de las aplicaciones web estáticas, las aplicaciones web dinámicas generan contenido de manera dinámica en tiempo de ejecución, permitiendo una mayor flexibilidad y funcionalidad.

Se presentan algunas características y ventajas de las aplicaciones web dinámicas:

- Interacción en tiempo real: las aplicaciones web dinámicas permiten la interacción en tiempo real con los usuarios. Esto se logra a través de tecnologías como JavaScript y AJAX, que permiten la actualización de contenido sin necesidad de recargar la página completa. Esto facilita la implementación de características como chat en vivo, notificaciones en tiempo real y actualización automática de datos.
- Personalización del contenido: las aplicaciones web dinámicas tienen la capacidad de personalizar el contenido para cada usuario. A través de técnicas como la gestión de sesiones y la utilización de bases de datos, se pueden mostrar datos y elementos específicos para cada usuario, adaptándose a sus preferencias y necesidades.
- Gestión de bases de datos: las aplicaciones web dinámicas suelen utilizar sistemas de gestión de bases de datos para almacenar y administrar datos. Esto permite la manipulación y recuperación de información en tiempo real, lo que resulta en una mayor capacidad para realizar consultas, filtrar datos y generar informes personalizados.
- Escalabilidad y flexibilidad: las aplicaciones web dinámicas están diseñadas para adaptarse a cambios en la demanda y escalar de manera eficiente. Al utilizar tecnologías como la computación en la nube y arquitecturas escalables, como la arquitectura de microservicios, estas aplicaciones pueden manejar grandes volúmenes de usuarios y crecer según las

necesidades del negocio.

### *Portales Web*

Los portales web se han convertido en una herramienta clave para proporcionar contenidos y servicios personalizados a los usuarios. Un portal web es un sitio web centralizado que proporciona información, aplicaciones y servicios de manera integrada a través de una única interfaz de usuario.<sup>(2)</sup>

Algunas características y beneficios de los portales en línea:

- Centralización de la información: los portales de Internet actúan como un punto de contacto centralizado para obtener información importante para una organización o comunidad. Permiten a los usuarios acceder a diversos recursos como noticias, documentos, servicios, aplicaciones, etc. desde una única plataforma, facilitando la búsqueda y el acceso a la información deseada.
- Personalización de contenidos: los portales de Internet ofrecen la posibilidad de personalizar los contenidos según las preferencias y necesidades del usuario. Las técnicas de segmentación y recomendación se pueden utilizar para proporcionar y adaptar contenido relevante para cada usuario, mejorando la experiencia y satisfacción del usuario.
- Integración de aplicaciones y servicios: los portales web le permiten integrar diferentes aplicaciones y servicios en una sola plataforma, brindando una experiencia unificada para los usuarios. Esto puede incluir, por ejemplo, la integración con sistemas de gestión de contenido, sistemas de gestión de relaciones con clientes (CRM) y sistemas de gestión de aprendizaje (LMS), lo que facilita el uso de diferentes herramientas y servicios desde un único punto de vista.
- Colaboración y participación: los portales en línea fomentan la colaboración y participación de los usuarios al proporcionar funciones como foros de discusión, blogs, comentarios y funciones de redes sociales. Fomenta la comunicación entre los usuarios y crea comunidades en línea donde se puede compartir información, ideas y experiencias.

### *Aplicación Web Animada*

Las aplicaciones web animadas han experimentado un crecimiento significativo en los últimos años, ofreciendo una experiencia más interactiva y dinámica para los usuarios. Estas aplicaciones utilizan tecnologías como HTML5, CSS3 y JavaScript para crear animaciones y efectos visuales en tiempo real.

Una de las principales ventajas de las aplicaciones web animadas es su capacidad para brindar interactividad. Los usuarios pueden interactuar con elementos de la página de forma dinámica, como hacer scroll para activar animaciones o utilizar elementos interactivos como botones y formularios.<sup>(6)</sup>

Esto crea una experiencia más atractiva y agradable para el usuario. Además del aspecto interactivo, las aplicaciones web animadas también ofrecen un atractivo visual. Las animaciones y efectos visuales pueden hacer que el diseño sea más atractivo y llamativo. Mediante el uso de técnicas como transformaciones 2D y 3D, transiciones de color y efectos de paralaje, se puede crear una interfaz visualmente impactante que capte la atención del usuario. Otra ventaja de las aplicaciones web animadas es

su capacidad para contar historias de manera visual. A través de animaciones secuenciales y efectos visuales, se puede presentar información de forma más dinámica y atractiva, lo que ayuda a transmitir mensajes y narrativas de manera efectiva.

Por último, las animaciones en las aplicaciones web pueden mejorar la experiencia del usuario al hacer que la navegación sea más fluida y los tiempos de carga sean más agradables. Además, las animaciones pueden proporcionar retroalimentación visual, indicando acciones realizadas por el usuario o cambios en el estado de la aplicación.

### **Clasificación y Usos**

Las aplicaciones web se pueden clasificar en diferentes categorías según su funcionalidad y características.

A continuación, se presentan algunas de las clasificaciones más comunes y los usos correspondientes de las aplicaciones web:

- Aplicaciones web empresariales: estas aplicaciones están diseñadas para satisfacer las necesidades específicas de las empresas, como la gestión de recursos humanos, el seguimiento de proyectos, la contabilidad y la gestión de inventario. Proporcionan funcionalidades empresariales clave y suelen ser personalizables para adaptarse a los procesos y requisitos de la organización.
- Aplicaciones web de comercio electrónico: estas aplicaciones permiten realizar transacciones de compra y venta en línea. Los usuarios pueden buscar productos, agregarlos al carrito de compras, realizar pagos y realizar un seguimiento de los pedidos. Las aplicaciones web de comercio electrónico son utilizadas por minoristas en línea y brindan una experiencia de compra conveniente y segura para los usuarios.
- Aplicaciones web de redes sociales: estas aplicaciones permiten a los usuarios conectarse y comunicarse con otros usuarios en línea. Proporcionan funcionalidades como perfiles de usuario, publicación de contenido, interacción social (como comentarios y me gusta) y la posibilidad de seguir a otras personas. Las aplicaciones web de redes sociales son ampliamente utilizadas para mantenerse en contacto con amigos, compartir contenido y participar en comunidades en línea.
- Aplicaciones web educativas: estas aplicaciones están diseñadas para apoyar el aprendizaje en línea. Pueden proporcionar contenido educativo interactivo, pruebas y evaluaciones, seguimiento del progreso del estudiante y herramientas de colaboración. Las aplicaciones web educativas son utilizadas por instituciones educativas, profesores y estudiantes para facilitar el aprendizaje en línea.
- Aplicaciones web de entretenimiento: estas aplicaciones ofrecen contenido y servicios relacionados con el entretenimiento, como transmisión de video, música, juegos en línea y plataformas de contenido multimedia. Proporcionan una experiencia de entretenimiento accesible a través de navegadores web, sin necesidad de descargar aplicaciones adicionales.

### **Arquitectura de Aplicativo Web**

El desarrollo web ha experimentado avances significativos en términos de arquitecturas utilizadas para construir aplicaciones

web. Estas arquitecturas proporcionan estructuras y patrones que permiten el desarrollo eficiente, escalable y mantenible de aplicaciones web.<sup>(7)</sup>

A continuación, se presentan algunos enfoques y tendencias relevantes en la arquitectura del aplicativo web.

- Arquitectura de tres capas: la arquitectura de tres capas sigue siendo ampliamente utilizada en el desarrollo de aplicativos webs. Esta arquitectura separa la aplicación en tres capas principales: la capa de presentación, la capa de lógica de negocio y la capa de acceso a datos. Esta separación permite un mayor modularidad y facilita la mantenibilidad del aplicativo.

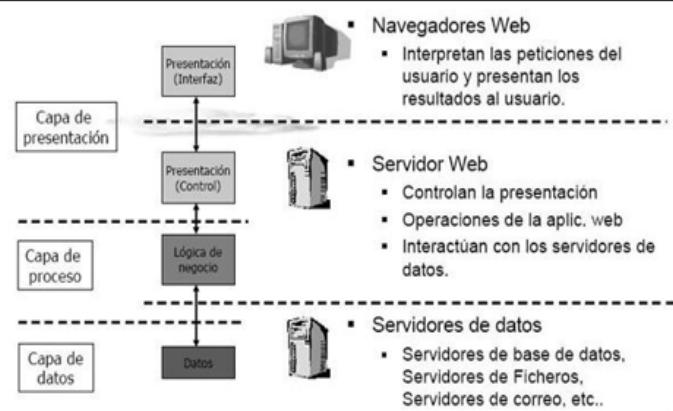
- Arquitectura basada en microservicios: la arquitectura basada en microservicios ha ganado popularidad en los últimos años. En esta arquitectura, la aplicación se divide en servicios independientes y autónomos, cada uno con su propio contexto y funcionalidad específica. Esta arquitectura ofrece flexibilidad, escalabilidad y facilita la implementación ágil de nuevas funcionalidades.

- Arquitectura sin servidor (Serverless): la arquitectura sin servidor es una tendencia emergente en el desarrollo de aplicativos webs. En esta arquitectura, la infraestructura subyacente es administrada por un proveedor de servicios en la nube, eliminando la necesidad de administrar servidores. Los desarrolladores se centran en escribir y desplegar funciones independientes que se ejecutan en respuesta a eventos. Esta arquitectura proporciona escalabilidad automática y reduce la carga operativa.

- Arquitectura de aplicativos webs progresivos (Progressive Web Applications, PWA): las PWA son aplicativos web que pueden funcionar tanto en navegadores web como en dispositivos móviles. Estas aplicaciones combinan las características de una aplicación web y una aplicación móvil, ofreciendo una experiencia de usuario similar a la de una aplicación nativa. Las PWA utilizan tecnologías como Service Workers y Web App Manifest para proporcionar funcionalidades offline, notificaciones push y acceso a hardware del dispositivo.

La arquitectura del aplicativo web es un aspecto fundamental en el diseño y desarrollo de aplicaciones web modernas. La arquitectura define la estructura, componentes y relaciones de un sistema de software, y en el caso de los aplicativos webs, se centra en la estructura y flujo de información entre el cliente y el servidor.<sup>(4)</sup>

Uno de los enfoques comunes en la arquitectura de aplicativos webs es el modelo cliente-servidor. Este modelo se basa en la separación de responsabilidades entre el cliente, que es la interfaz con la que interactúa el usuario, y el servidor, que gestiona los datos y la lógica del negocio. La comunicación entre el cliente y el servidor se realiza generalmente a través de protocolos de comunicación como HTTP. Además del modelo cliente-servidor, han surgido diferentes enfoques arquitectónicos en el desarrollo de aplicativos webs. Una arquitectura popular es la arquitectura de microservicios, que se basa en dividir una aplicación en pequeños servicios independientes y altamente cohesivos. Cada microservicio se desarrolla y despliega de forma independiente, lo que permite una mayor escalabilidad y flexibilidad en el desarrollo y mantenimiento de la aplicación web.<sup>(7)</sup>



Fuente: <https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/>

Figura 3. Arquitectura de las aplicaciones Web

## Tecnologías de Programación APIs

Las APIs (Application Programming Interfaces) son conjuntos de reglas y protocolos que permiten que diferentes aplicaciones se comuniquen entre sí. Las APIs han desempeñado un papel fundamental en el desarrollo de aplicaciones web y móviles. Su importancia radica en su capacidad para facilitar la integración de sistemas y servicios, lo que permite crear aplicaciones más poderosas y escalables, además el uso de APIs está en constante evolución con la aparición de nuevas tecnologías y estándares.

Se presenta algunas de las APIs más usadas:

- API REST: el estilo de arquitectura de la API Representational State Transfer (REST) se ha vuelto muy popular. Las APIs RESTful utilizan los métodos HTTP (GET, POST, PUT, DELETE, etc.) para acceder y manipular recursos. Proporcionan una interfaz uniforme y escalable para la comunicación entre aplicaciones.

- API GraphQL: GraphQL es un lenguaje de consulta para APIs y una alternativa a las APIs REST. Proporciona a los clientes la capacidad de solicitar solo los datos necesarios y reduce la sobrecarga de tráfico de red. Con GraphQL, los clientes pueden especificar los datos que necesitan y obtenerlos en una sola llamada.

- APIs de servicios en la nube: los proveedores de servicios en la nube, como Amazon Web Services (AWS), Google Cloud Platform (GCP) y Microsoft Azure, ofrecen APIs para acceder y administrar los recursos en sus plataformas. Estas APIs permiten a los desarrolladores crear y escalar aplicaciones en la nube de manera eficiente.

- APIs de redes sociales: las redes sociales, como Facebook, Twitter y LinkedIn, ofrecen

APIs que permiten a los desarrolladores acceder a sus datos y funcionalidades. Estas APIs permiten la integración de aplicaciones con las redes sociales, como compartir contenido, iniciar sesión con cuentas sociales y obtener información de perfiles de usuarios.

- Seguridad de las APIs: la seguridad de las APIs es un aspecto crítico. Se utilizan técnicas como autenticación, autorización y cifrado para proteger las APIs y los datos que se intercambian a través de ellas. También se utilizan prácticas de gestión de API para

controlar el acceso, monitorear el uso y garantizar la integridad de las transacciones.

### Lenguaje de Programación Web

Estos son solo algunos de los lenguajes de programación web relevantes.

Es importante destacar que el desarrollo web es un campo en constante evolución, y pueden surgir nuevos lenguajes y tecnologías en el futuro aquí algunos de los más usados:

#### *JavaScript*

JavaScript es un lenguaje de programación ampliamente utilizado en el desarrollo web, este ha sido el lenguaje principal para programar interacciones y comportamientos en el lado del cliente en las páginas web. Permite crear experiencias interactivas y dinámicas, manipular elementos HTML, realizar peticiones al servidor y construir aplicaciones web de una sola página (SPA). JavaScript se ejecuta en el navegador del usuario y ha evolucionado con el tiempo, con el surgimiento de frameworks populares como React.js, Angular y Vue.js que facilitan el desarrollo de aplicaciones web modernas y escalables.<sup>(6)</sup>

#### *HTML5*

HTML5 es la última versión del lenguaje de marcado utilizado para estructurar el contenido de las páginas web. HTML5 ha permitido a los desarrolladores web crear sitios más ricos en contenido multimedia y más accesible. Introduce nuevos elementos semánticos, etiquetas de video y audio, gráficos vectoriales, almacenamiento local y capacidades de geolocalización. HTML5 proporciona una base sólida para la creación de sitios web modernos y adaptables a diferentes dispositivos y pantallas.<sup>(6)</sup>

#### *CSS3*

CSS3 es el lenguaje utilizado para dar estilo y presentación a los elementos HTML en una página web. CSS3 ha mejorado significativamente la capacidad de los desarrolladores para diseñar sitios web atractivos y responsivos. CSS3 ofrece características avanzadas como transiciones, animaciones, sombras, efectos visuales y diseño flexible. Permite controlar el diseño, la tipografía, los colores y otros aspectos visuales de una página web.<sup>(6)</sup>

#### *PHP*

PHP es un lenguaje de programación ampliamente utilizado en el desarrollo web del lado del servidor. A pesar de las críticas, sigue siendo una opción popular debido a su facilidad de uso y la amplia disponibilidad de recursos y documentación, PHP ha evolucionado con mejoras en rendimiento, seguridad y capacidad de desarrollo. Frameworks como Laravel y Symfony han impulsado el desarrollo de aplicaciones web robustas y escalables en PHP.<sup>(6)</sup>

#### *Python*

Python es un lenguaje de programación versátil que ha ganado popularidad en el desarrollo web, se ha utilizado ampliamente para el desarrollo web del lado del servidor. Python es conocido por su sintaxis legible y su amplia biblioteca estándar, lo que facilita el desarrollo rápido y eficiente. Frameworks como Django y Flask han sido ampliamente adoptados para crear aplicaciones web en Python.<sup>(6)</sup>

### Modelos de Desarrollo app WEB

Los modelos de desarrollo de aplicaciones web son enfoques utilizados para organizar y estructurar el proceso de desarrollo de una aplicación web.<sup>(8)</sup> Aquí tienes un resumen de los principales modelos:

- **Modelo de desarrollo tradicional:** también conocido como modelo de cascada, este enfoque de desarrollo sigue una secuencia lineal de pasos, donde cada etapa se completa antes de pasar a la siguiente. Comienza con el análisis de requisitos, seguido del diseño, implementación, pruebas y finalmente el despliegue. Aunque ha sido ampliamente utilizado, este modelo puede resultar inflexible en proyectos web complejos y con cambios frecuentes.

- **Modelo de desarrollo ágil:** el enfoque ágil se basa en la flexibilidad y la colaboración continua entre los miembros del equipo de desarrollo. Los métodos ágiles, como Scrum y Kanban, se centran en la entrega iterativa y incremental de funcionalidades. El equipo trabaja en ciclos cortos, llamados sprints, y realiza revisiones y adaptaciones constantes para satisfacer las necesidades cambiantes del proyecto y los clientes.

- **Modelo DevOps:** el modelo DevOps promueve la colaboración y la integración continua entre los equipos de desarrollo y operaciones. Combina prácticas ágiles con enfoques automatizados para el desarrollo, pruebas, implementación y operación de aplicaciones web. El objetivo es lograr una entrega más rápida y confiable de software, mejorando la comunicación y la eficiencia entre los equipos involucrados.

- **Modelo de desarrollo basado en prototipos:** este enfoque implica la creación de prototipos rápidos y iterativos de la aplicación web. El objetivo es obtener una retroalimentación temprana de los usuarios y validar conceptos y funcionalidades clave antes de la implementación completa. El desarrollo basado en prototipos permite ajustes rápidos y reducir el riesgo de desarrollar una aplicación web que no cumpla con los requisitos del usuario.

- **Modelo de desarrollo en espiral:** el modelo en espiral combina elementos del modelo de cascada y la metodología ágil. Se enfoca en la gestión de riesgos y el desarrollo iterativo. Cada ciclo comienza con la identificación y evaluación de riesgos, seguido del desarrollo, pruebas y evaluación del progreso. El modelo en espiral permite la adaptación y la mitigación de riesgos a medida que se avanza en el desarrollo de la aplicación web.

### Análisis Comparativo

#### *Modelo de Desarrollo Tradicional*

El modelo de desarrollo tradicional, también conocido como enfoque en cascada, se caracteriza por tener una estructura secuencial y lineal. Se divide en etapas bien definidas, como análisis de requisitos, diseño, implementación, pruebas y mantenimiento.<sup>(9)</sup>

Algunas características clave de este modelo son:

- **Enfoque predictivo:** se establecen todos los requisitos y se planifica el proyecto en su totalidad antes de comenzar el desarrollo.

- **Entregables finales:** el desarrollo se enfoca en la

entrega de un producto final completo y funcional.

- Poca flexibilidad: los cambios en los requisitos son difíciles de incorporar una vez que el proceso está en marcha.
- Mayor énfasis en la documentación: se generan documentos detallados en cada etapa del proceso.

### *Modelo de Desarrollo Ágil*

El modelo de desarrollo ágil se basa en la adaptabilidad y la colaboración cercana con los stakeholders. Se enfoca en entregar incrementos de software funcionales en ciclos cortos y frecuentes.<sup>(8)</sup>

Algunas características clave de este modelo son:

- Enfoque iterativo e incremental: el desarrollo se divide en iteraciones cortas llamadas “sprints” que permiten la entrega temprana de valor.
- Colaboración y retroalimentación: se fomenta la colaboración constante entre el equipo de desarrollo y los stakeholders, con retroalimentación continua para mejorar el producto.
- Flexibilidad ante el cambio: los requisitos pueden adaptarse y ajustarse a medida que se obtiene una mejor comprensión del proyecto.
- Mayor énfasis en la comunicación y la interacción: se prioriza la comunicación directa y el trabajo en equipo.

### *Modelo DevOps*

El modelo DevOps se centra en la integración continua y la entrega continua de software. Combina los aspectos del desarrollo de software y las operaciones de TI en un enfoque colaborativo y automatizado.

Algunas características clave de este modelo son:

- Automatización: se automatizan los procesos de desarrollo, pruebas, implementación y gestión de infraestructura.
- Integración continua: se busca la integración frecuente y continua de cambios en el código fuente.
- Entrega continua: se enfoca en entregar software funcional de alta calidad de manera rápida y frecuente.
- Mayor énfasis en la colaboración entre equipos: los equipos de desarrollo y operaciones trabajan juntos de manera colaborativa para agilizar el ciclo de entrega.

### *Modelo de Desarrollo Basado en Prototipos*

El modelo de desarrollo basado en prototipos se centra en la construcción rápida de prototipos iterativos para validar y refinar los requisitos del proyecto.<sup>(8)</sup>

Algunas características clave de este modelo son:

- Construcción rápida de prototipos: se desarrollan versiones preliminares del software para obtener retroalimentación temprana de los stakeholders.
- Iterativo y evolutivo: los prototipos se van refinando y mejorando en cada iteración hasta llegar al producto final.
- Flexibilidad en los requisitos: los cambios en los requisitos pueden incorporarse rápidamente en función de los comentarios obtenidos de los prototipos.
- Mayor énfasis en la interacción con los usuarios: los prototipos se utilizan como herramientas de comunicación y colaboración con los usuarios finales.

### **Selección de Modelo WEB**

De modelos web basado en el tema de tesis “Desarrollo de una Aplicación Web para el Control de Ventas y la Toma de Decisiones en la Corporación Nacional de Emprendedores y Telecomunicaciones - CNET del Cantón La Maná”, se seleccionará el Modelo de Desarrollo Ágil. El Modelo de Desarrollo Ágil es un enfoque iterativo e incremental que se centra en la colaboración cercana entre el equipo de desarrollo y los stakeholders para entregar rápidamente un software funcional y de alta calidad. Este modelo se basa en los principios del Manifiesto Ágil y promueve la flexibilidad, adaptabilidad y respuesta rápida a los cambios. Se presenta un análisis de modelos web basado en este enfoque:

- Modelo de casos de uso ágiles: en el desarrollo ágil, se utiliza el concepto de historias de usuario en lugar de casos de uso tradicionales. Las historias de usuario describen los requisitos funcionales desde la perspectiva del usuario y se utilizan como base para la planificación y el seguimiento del progreso del proyecto. Estas historias se pueden representar en forma de tarjetas o documentos sencillos.
- Modelo de base de datos ágil: en el desarrollo ágil, el diseño de la base de datos evoluciona a medida que se desarrolla la aplicación. Se utiliza un enfoque incremental para el modelado de la base de datos, donde las tablas y relaciones se van agregando o modificando a medida que se identifican nuevos requisitos. Esto permite una mayor flexibilidad y adaptabilidad a medida que se descubren nuevas necesidades durante el desarrollo.
- Modelo de navegación ágil: en el desarrollo ágil, se fomenta la iteración y la retroalimentación continua. El modelo de navegación se va refinando y ajustando en cada iteración del desarrollo, a medida que se incorporan los comentarios y las necesidades de los usuarios finales. Esto garantiza que la interfaz de usuario sea intuitiva y se adapte a las necesidades cambiantes de los usuarios.
- Modelo de proceso de ventas ágil: en el desarrollo ágil, se priorizan las funcionalidades y se desarrollan en ciclos cortos llamados “sprints”. El modelo de proceso de ventas se divide en incrementos entregables, lo que permite que la aplicación web sea funcional y utilizable en cada iteración. Esto brinda la oportunidad de obtener retroalimentación temprana de los usuarios y ajustar el proceso de ventas según sus necesidades.
- Modelo de informes ágil: los informes también se desarrollan de forma iterativa en el modelo ágil. Se identifican los requisitos clave para los informes iniciales y se van refinando y ampliando en cada iteración. Esto permite que los informes se ajusten a las necesidades cambiantes de la organización y se entreguen de manera oportuna.

### **Desarrollo de una Web Basada en Wordpress**

“WordPress es uno de los sistemas de gestión de contenido más populares y ampliamente utilizados en el desarrollo web. Ofrece una interfaz intuitiva y amigable que permite a los usuarios sin conocimientos técnicos crear y gestionar su propio contenido. Además, cuenta con una amplia gama de temas y plugins que facilitan la personalización y extensión de las funcionalidades del sitio web. La comunidad activa de desarrolladores y usuarios de WordPress proporciona un valioso

soporte y recursos para aprovechar al máximo esta plataforma".<sup>(10)</sup>

#### *¿Qué es WordPress?*

WordPress es un sistema de gestión de contenido (CMS, por sus siglas en inglés) de código abierto utilizado para crear y administrar sitios web de manera fácil y efectiva. Fue lanzado por primera vez en 2003 y desde entonces se ha convertido en una de las plataformas más populares para la creación de sitios web.

WordPress se destaca por su interfaz intuitiva y amigable, lo que permite a personas sin conocimientos técnicos desarrollar y gestionar su propio contenido en línea. Su estructura modular y flexible se basa en un núcleo central al que se le pueden añadir temas y plugins para personalizar y ampliar las funcionalidades del sitio web.<sup>(10)</sup>

Los temas en WordPress son plantillas que definen la apariencia visual y la estructura del sitio. Hay miles de temas disponibles, desde diseños simples y elegantes hasta opciones más complejas y personalizables. Los usuarios pueden elegir un tema que se ajuste a sus necesidades y personalizarlo según sus preferencias. Por otro lado, los plugins en WordPress son extensiones de software que se pueden agregar al sitio para agregar funcionalidades adicionales.



## WORDPRESS

Fuente: <https://es.localo.com/marketing-dictionary/what-is-wordpress>

Figura 4. Word Press

Existen miles de plugins disponibles para casi cualquier necesidad, como formularios de contacto, galerías de imágenes, tiendas en línea, optimización de motores de búsqueda (SEO) y mucho más.<sup>(10)</sup>

Los plugins permiten a los usuarios personalizar su sitio web y agregar características específicas sin tener que programar desde cero. WordPress también ofrece una interfaz de administración que permite a los usuarios crear, editar y publicar contenido fácilmente. Los usuarios pueden crear y gestionar páginas, entradas de blog, categorías, etiquetas y otros elementos del sitio web de manera eficiente.

#### *Características de WordPress*

Algunas de las Características de utilizar WordPress incluyen:

- Facilidad de uso: WordPress ofrece una interfaz intuitiva y amigable que permite a los usuarios sin conocimientos técnicos crear y gestionar su propio contenido.
- Amplia variedad de temas y plugins: WordPress cuenta con una amplia gama de temas y plugins que permiten personalizar y extender las funcionalidades del sitio web.
- Comunidad activa: existe una gran comunidad de desarrolladores y usuarios que brindan soporte, comparten recursos y contribuyen a la mejora continua de WordPress.
- Personalización del diseño: los temas determinan la apariencia visual y la estructura del sitio. Pueden ser

seleccionados de un repositorio de temas o personalizados según las necesidades específicas del proyecto.

- Desarrollo de plugins: los plugins son complementos que agregan funcionalidades adicionales al sitio web. Pueden ser utilizados para agregar características específicas, como formularios de contacto, galerías de imágenes, tiendas en línea, entre otros.

- Optimización para motores de búsqueda (SEO): WordPress proporciona herramientas y plugins que facilitan la optimización del sitio web para los motores de búsqueda. Estas herramientas permiten gestionar aspectos como las metaetiquetas, las estructuras de URL amigables y la optimización del contenido, lo que contribuye a mejorar el posicionamiento en los resultados de búsqueda.

- Gestión de contenido: WordPress ofrece una interfaz de administración que permite gestionar y organizar el contenido del sitio web de manera eficiente. Los usuarios pueden crear y publicar contenido, administrar páginas, categorías, etiquetas, entre otros elementos.

- Seguridad y actualizaciones: es importante mantener el sitio web seguro y actualizado. WordPress ofrece actualizaciones periódicas que corrigen vulnerabilidades y mejoran la seguridad. Además, existen plugins y buenas prácticas de seguridad que se pueden implementar para proteger el sitio web.

#### *Lenguajes Que Utiliza Wordpress*

WordPress utiliza principalmente dos lenguajes en su estructura y desarrollo: PHP (Hypertext Preprocessor) y MySQL.

- PHP es un lenguaje de programación de código abierto ampliamente utilizado en el desarrollo web. Es el lenguaje principal utilizado por WordPress para la creación de temas, plugins y el manejo de la lógica del sitio web. PHP permite la interacción con la base de datos y el procesamiento de datos dinámicos en WordPress.

- MySQL es un sistema de gestión de bases de datos relacional utilizado por WordPress para almacenar y recuperar datos. WordPress utiliza MySQL para administrar y almacenar información sobre páginas, entradas, comentarios y configuraciones del sitio web. Proporciona una forma eficiente de organizar y acceder a los datos necesarios para el funcionamiento de WordPress.

Además de PHP y MySQL, WordPress también utiliza otros lenguajes y tecnologías como HTML (Hypertext Markup Language) para la estructura y el contenido de las páginas, CSS (Cascading Style Sheets) para el diseño y la presentación visual, y JavaScript para la interactividad y la mejora de la experiencia del usuario en el sitio web.<sup>(10)</sup> Estos lenguajes y tecnologías trabajan en conjunto para permitir la funcionalidad y la personalización de WordPress. La combinación de PHP y MySQL, junto con otros lenguajes y tecnologías, brinda a los desarrolladores la capacidad de crear sitios web dinámicos, interactivos y personalizados utilizando la plataforma de WordPress.<sup>(10)</sup>

#### *MariaDB*

MariaDB Server destaca como uno de los servidores de bases de datos más usados en el ámbito global. Este sistema, creado por

los desarrolladores originales de MySQL, se distingue por ser completamente de código abierto. Su función principal consiste en transformar datos en información estructurada, siendo utilizado en diversas aplicaciones que abarcan desde entornos bancarios hasta sitios web. Se presenta como una mejora directa y avanzada con respecto a MySQL.<sup>(11)</sup>

La elección de MariaDB se fundamenta en su rapidez, escalabilidad y robustez, respaldadas por un amplio ecosistema de motores de almacenamiento, complementos y diversas herramientas que le otorgan versatilidad para adaptarse a una amplia variedad de escenarios de uso. Desarrollado como software de código abierto y orientado como una base de datos relacional, MariaDB proporciona una interfaz SQL para la gestión de datos, incorporando además características GIS y JSON en sus versiones más recientes.<sup>(11)</sup>



**Fuente:** [https://www.mariadbTutorial.com/getting-started/what-is-mariadb/](https://www.mariadbtutorial.com/getting-started/what-is-mariadb/)

**Figura 5.** ¿Qué es MariaDB?

#### Ventajas Maria Db

Se han incorporado nuevos motores de almacenamiento, destacando Aria como un reemplazo de MyISAM y XtraDB como sustituto de InnoDB, dirigidos principalmente a la mayoría de los usuarios. Aria se presenta como un motor a prueba de fallos basado en MyISAM, mientras que XtraDB se define como la alternativa que reemplaza al motor InnoDB, basándose en su plug-in correspondiente.

Además de estos, se suman otros motores de almacenamiento notables, como PBXT, reconocido por sus características transaccionales, y FederatedX, que ocupa el lugar del motor Federated.

- Las mejoras en velocidad se concentran, sobre todo, en consultas complejas cuando se emplea el motor de almacenamiento Aria. Este motor optimiza el rendimiento al almacenar en caché los datos de tablas temporales en la memoria, reduciendo así la dependencia del disco duro.
- Se introdujeron también nuevas tablas de sistema (INFORMATION\_SCHEMA) diseñadas para almacenar estadísticas que facilitan la optimización de las bases de datos. En términos generales, se han implementado diversas modificaciones para potenciar el rendimiento, la velocidad e incluso incorporar nuevas características.
- El sistema de manejo de conexiones ha experimentado mejoras notables al implementar el sistema pool-of-threads de MySQL 6.0, permitiendo gestionar más de 200 000 conexiones en MariaDB. Este avance representa una mejora significativa en la capacidad del sistema para manejar múltiples conexiones de manera eficiente.

#### Desventajas MariaDB

La única posible desventaja que podría surgir se relaciona

con casos aislados de incompatibilidad, sin embargo, dado que la mayoría de servicios existentes son muy similares, es poco probable encontrarse con situaciones de este tipo. Si tu aplicación opera de manera efectiva en MySQL, puedes tener una confianza cercana al 99,99 % de que funcionará igualmente en MariaDB.

#### Por qué usar MariaDB en lugar de MySQL

La justificación para migrar de MySQL a MariaDB se fundamenta en todas las mejoras significativas de rendimiento. Además, es importante destacar el notable respaldo que la comunidad brinda a MariaDB, convirtiéndose en una plataforma de gran relevancia.

Tanto es así que algunas distribuciones Linux, como Fedora, han optado por reemplazar MySQL con MariaDB. Es crucial recordar que MariaDB opera bajo la licencia GPL, que es menos restrictiva y completamente gratuita.

#### Qué diferencias hay entre MariaDB vs MySQL

- Motores: en contraste con MySQL, que hace uso de los motores MyISAM e InnoDB, MariaDB introduce dos sustitutos. El primero de ellos, Aria, reemplaza a MyISAM, mientras que el segundo, XtraDB, sustituye a InnoDB. Cabe destacar que Aria se desarrolló a partir de MyISAM, y XtraDB funciona como un plugin de InnoDB.

Adicionalmente, MariaDB incorpora dos motores adicionales: PBXT y FederatedX.

- Velocidad: el desempeño de MariaDB supera al de MySQL, aunque la diferencia no sea astronómica, es perceptiblemente mejor. Específicamente, el motor Aria destaca por su rendimiento sobresaliente en comparación con MyISAM, gracias a su caché en la memoria RAM, en contraste con el caché de disco utilizado por MyISAM.
- Tablas: MariaDB introduce nuevas tablas a nivel del sistema, contribuyendo significativamente a las tareas de optimización de bases de datos al almacenar estadísticas del servicio.

#### Scrum

Scrum representa una metodología ágil y adaptable diseñada para gestionar el desarrollo de software, con el objetivo central de maximizar el retorno de la inversión (ROI) de la empresa. Se fundamenta en la premisa de construir primero la funcionalidad de mayor valor para el cliente y se rige por principios clave como la inspección continua, la adaptación, la autogestión y la innovación.<sup>(12)</sup>

Esta metodología se configura como un enfoque colaborativo en el que se busca lograr resultados funcionales de manera eficiente, especialmente orientado a proyectos de desarrollo de aplicaciones móviles o programación web.

#### Beneficios Scrum

- Adaptabilidad a cambios: la metodología exhibe una notable capacidad de reacción frente a los cambios en los requisitos, originados por las necesidades emergentes o evoluciones del mercado. Su diseño está específicamente concebido para ajustarse de manera efectiva a tales cambios.
- Mejora en la calidad del software: la sistemática de trabajo, junto con la necesidad de obtener una

versión funcional después de cada iteración, contribuye significativamente a lograr un software de calidad superior.

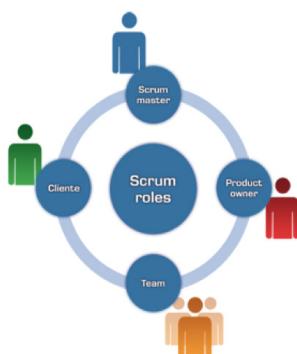
- Incremento de la productividad: este aumento se logra, entre otras razones, mediante la eliminación de la burocracia y la motivación del equipo, que se beneficia de la autonomía para organizarse.
- Reducción del Time to Market: la posibilidad de comenzar a utilizar las funcionalidades clave del proyecto antes de su conclusión total constituye un beneficio, permitiendo un acceso temprano a las características más importantes.
- Maximización del retorno de la inversión (ROI): la metodología se orienta a la producción de software que incorpora exclusivamente las funcionalidades que aportan un mayor valor de negocio, gracias a la priorización basada en el retorno de inversión.

#### *Proceso de Scrum*

El desarrollo sigue un enfoque iterativo e incremental, dividiéndose en iteraciones llamadas Sprints, con una duración predefinida de entre 2 y 4 semanas. Al finalizar cada Sprint, se obtiene una versión de software con nuevas funcionalidades listas para su uso.

- Reserva: conjunto de requisitos denominados historias, descritos en un lenguaje no técnico.
- Planificación de Iteración: el equipo determina la cantidad de historias que puede comprometerse a completar en esa iteración y luego organiza la forma de lograrlo.
- Reunión diaria de Iteración: breve reunión diaria de máximo 15 minutos, donde el equipo se sincroniza para trabajar de manera coordinada.
- Reserva de Iteración: lista de tareas necesarias para llevar a cabo las historias de la iteración.
- Iteración: periodo de tiempo predefinido durante el cual el equipo trabaja para convertir las historias de la reserva, a las que se ha comprometido, en una nueva versión del software.
- Demo y retrospectiva: reunión que tiene lugar al final de la iteración, en la cual el equipo presenta las historias completadas mediante una demostración de su trabajo.

#### *Roles Scrum*



Fuente: <https://steemit.com/spanish/@jarote27/scrum-metodologia-agil>

Figura 6. Los roles Metodología SCRUM

En Scrum, el enfoque principal del equipo es la construcción de software de alta calidad. La gestión de un proyecto Scrum se concentra en definir las características esenciales del producto a construir (qué construir, qué no construir y en qué orden hacerlo), así como en superar cualquier obstáculo que pueda dificultar el progreso del equipo de desarrollo.<sup>(12)</sup>

El equipo Scrum se compone de los siguientes roles:

- Facilitador o Scrum Master: es la persona que lidera al equipo, guiándolo para cumplir con las reglas y procesos de la metodología. Además, gestiona la resolución de impedimentos del proyecto y colabora estrechamente con el propietario del producto.
- Propietario del Producto (PO): representa a los accionistas y clientes que utilizan el software. Su enfoque se centra en la parte de negocio, siendo responsable del proyecto y asegurándose de entregar un valor superior al dinero invertido.
- Equipo: este grupo está conformado por profesionales con los conocimientos técnicos necesarios, quienes trabajan de manera colaborativa para desarrollar el proyecto.

#### **Toma de decisiones**

La toma de decisiones es un proceso fundamental en todas las áreas de la vida y en el ámbito empresarial en particular. Se han realizado investigaciones significativas en el campo de la toma de decisiones para comprender mejor cómo los individuos y las organizaciones toman decisiones y cómo mejorar este proceso.

La teoría de la toma de decisiones ha evolucionado en los últimos años, y se han desarrollado nuevos enfoques y modelos para analizar y mejorar la toma de decisiones. Por ejemplo, se ha prestado atención a la toma de decisiones basada en datos y evidencias, utilizando técnicas como el análisis cuantitativo y el análisis de datos para respaldar las decisiones.

Se ha reconocido la importancia de los aspectos emocionales y cognitivos en la toma de decisiones. Los estudios han destacado la influencia de los sesgos cognitivos y las emociones en el proceso de toma de decisiones, lo que ha llevado a un mayor enfoque en la educación y el entrenamiento para mejorar la toma de decisiones.<sup>(13)</sup>

Otro avance importante en la toma de decisiones ha sido la integración de la inteligencia artificial y el aprendizaje automático en el proceso. Estas tecnologías han demostrado ser útiles para analizar grandes cantidades de datos, identificar patrones y tendencias, y brindar recomendaciones para la toma de decisiones.

En el ámbito empresarial, la toma de decisiones estratégicas ha sido un tema de investigación clave. Se han desarrollado enfoques y marcos para apoyar la toma de decisiones estratégicas, considerando factores como el entorno empresarial, la competencia y los objetivos organizacionales.

#### **Business Intelligence**

Business Intelligence (BI) se ha convertido en un área clave en el campo de la gestión empresarial. BI se refiere al conjunto de tecnologías, herramientas y prácticas que permiten a las organizaciones recopilar, analizar y presentar información relevante para la toma de decisiones.

A lo largo de los años, el campo de BI ha experimentado avances significativos, impulsados por el crecimiento explosivo de datos y el desarrollo de nuevas tecnologías, el enfoque

en BI se ha centrado en la capacidad de procesar grandes volúmenes de datos, extraer información valiosa y presentarla de manera comprensible y accesible. La adopción de tecnologías como el big data y el análisis avanzado ha sido fundamental para el desarrollo de BI. Estas tecnologías permiten a las organizaciones procesar datos de diferentes fuentes, identificar patrones y tendencias, y obtener conocimientos accionables. La visualización de datos también ha sido un área de enfoque en el campo de BI, se han desarrollado herramientas y técnicas para presentar la información de manera visualmente atractiva y comprensible, lo que facilita la interpretación de los datos por parte de los usuarios.<sup>(14)</sup>

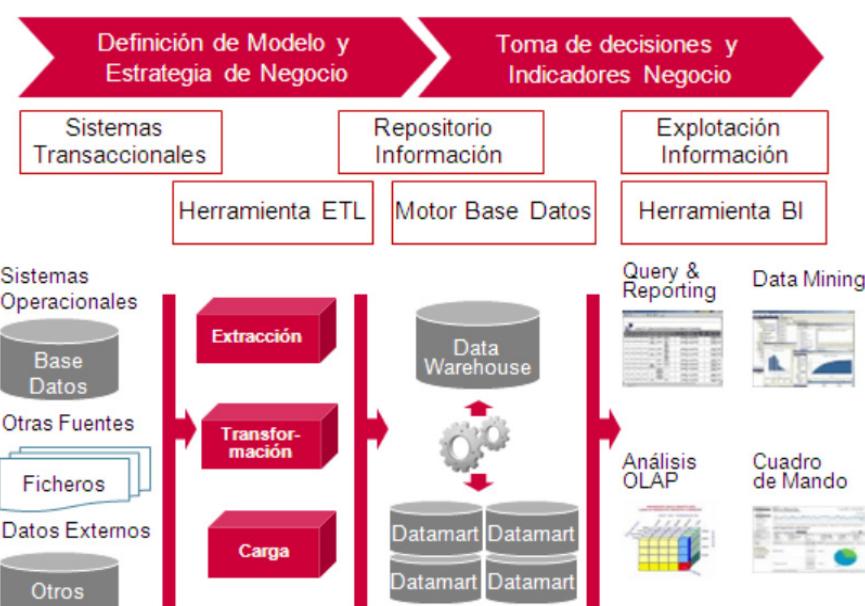
En el ámbito empresarial, BI se ha utilizado para diversos propósitos, como el análisis de mercado, la monitorización del rendimiento empresarial y la toma de decisiones estratégicas. Las organizaciones han reconocido el valor de utilizar BI como una herramienta para obtener una ventaja competitiva y mejorar la eficiencia operativa. Cada solución de inteligencia empresarial comienza con los sistemas fuente de la empresa. Lograr esto requiere extraer, transformar y cargar datos (ETL). Este paso suele estar respaldado por un repositorio intermedio llamado ODS, que actúa como intermediario entre los sistemas de origen y los sistemas de destino (generalmente un almacén de datos). La información recibida se almacena en el almacén de datos de la empresa, que puede ser la base para crear marchas de

datos específicas del departamento.<sup>(11)</sup>

Se caracterizan por tener una estructura óptima para lograr el análisis de datos de cada área de la empresa ya sea a través de bases de datos transaccionales (OLTP) o analíticas (OLAP). Los datos almacenados en el almacén de datos o en cada almacén de datos se utilizan, entre otras cosas, junto con herramientas de informes y análisis de negocios.

Los almacenes de datos están diseñados para facilitar el procesamiento y análisis eficiente de grandes conjuntos de datos. Utilizan esquemas específicos, como el esquema en estrella o copo de nieve, para organizar los datos de manera que sea fácil realizar consultas y análisis. Esté almacén integra datos de bases de datos transaccionales y analíticas, permitiendo su uso con herramientas de informes y análisis de negocios. Facilita la toma de decisiones al proporcionar una visión consolidada y eficiente de la información empresarial.<sup>(11)</sup>

El modelo de Business Intelligence consiste en instrumentos técnicas que se enfocan en la extracción de datos (E), transformación (T) y carga de datos (L). Con fines analíticos (ETL). Se puede obtener información interesante de las bases de datos. Debe obtenerse de los sistemas de información otras fuentes internas, externas y luego convertidas de acuerdo a las necesidades de datos instalación que finalmente se carga en un espacio privado para el próximo uso.<sup>(13)</sup>



Fuente: <https://diarium.usal.es/id00710310/2016/03/16/business-intelligence/>

Figura 7. Business Intelligence

Los procesos ETL luchan con la realidad, lo que demuestra buena parte de los sistemas de información institucional y que explican gran parte de dicha falla iniciativas tales como: falta de arquitectura de base de datos institucional, superposición de profesiones similares en diferentes sistemas, diferentes tecnologías, alta parte de los sistemas de información, baja interoperabilidad de los sistemas, mala calidad de los datos registrados, falta de codificación y bajo uso de estándares criterios institucionales, nacionales y globales.

Si estos puntos no son analizados y resueltos, existe la oportunidad de impedir el desarrollo del plan de inteligencia

desde empresas hasta nivel universitario. (E) información extraída de bases de datos u otras fuentes de interés para la agencia y después de la transformación (T) con fines de análisis, se guardan como borrador BI clásico en almacenes de datos llamados bases de datos (Data Warehouse DW), Data Marts DM o Data Lakes; Forman de factores de la arquitectura y perfilan BI y es el núcleo de este tipo de iniciativas. Se almacena la información necesaria para el análisis y la toma de decisiones institucional. Los elementos de una arquitectura típica de contribución inteligente Business Intelligence (BI) consiste en instrumentos y aplicaciones tecnológicas.<sup>(11)</sup>

### *Arquitecturas de BI*

Las arquitecturas de Business Intelligence (BI) son estructuras que organizan y gestionan el flujo de datos y el procesamiento analítico para permitir la toma de decisiones basada en información.

A continuación, se presentan algunas de las arquitecturas de BI:

- Arquitectura de BI Tradicional: la arquitectura de BI tradicional sigue un enfoque centralizado, en el cual los datos se extraen, transforman y cargan (ETL) desde diversas fuentes en un almacén de datos central. Luego, los datos se transforman y se organizan en un modelo dimensional para facilitar el análisis y la generación de informes. Los usuarios acceden a los datos mediante herramientas de consulta y generación de informes.
- Arquitectura de BI Moderna: la arquitectura de BI moderna se basa en tecnologías más recientes y enfoques ágiles para el análisis de datos. Se caracteriza por la utilización de herramientas de visualización de datos interactivas y auto-servicio. Los datos pueden provenir de diferentes fuentes, como bases de datos relacionales, sistemas en la nube, redes sociales, entre otros. Los datos se integran y preparan para su análisis utilizando técnicas de preparación de datos y modelos de datos más flexibles, como el modelo en estrella o en copo de nieve.
- Arquitectura de BI basada en la nube: la arquitectura de BI basada en la nube utiliza servicios y plataformas en la nube para almacenar, procesar y analizar datos. Los datos se almacenan en la nube, lo que permite un acceso más fácil y escalable desde diferentes ubicaciones y dispositivos. Las herramientas de BI se ejecutan en la nube, lo que reduce la necesidad de infraestructura local y permite un despliegue más rápido.
- Arquitectura de BI Distribuida: la arquitectura de BI distribuida se basa en la descentralización de las funciones de BI en diferentes unidades o departamentos de una organización. Cada unidad puede tener su propio almacén de datos y herramientas de análisis, lo que les brinda autonomía para tomar decisiones basadas en la información relevante para sus áreas específicas. Sin embargo, es importante garantizar la consistencia y la integridad de los datos en toda la organización.

### *Modelos de Implementación de B.I.*

Los modelos de implementación de Business Intelligence (BI) son enfoques utilizados para implementar soluciones de BI en una organización. Estos modelos pueden variar según las necesidades y los recursos de la organización, así como el alcance y la complejidad del proyecto.<sup>(13)</sup>

A continuación, se presentan algunos modelos de implementación de BI comunes:

- Modelo de implementación centralizado: en este modelo, se establece un centro de excelencia o un equipo central encargado de implementar y administrar la solución de BI. Este equipo es responsable de la recopilación, limpieza y procesamiento de los datos, así como del diseño y mantenimiento del almacén de datos. Los usuarios acceden a la solución de BI a través de informes y paneles predefinidos generados por el equipo central.
- Modelo de implementación descentralizado:

en este modelo, cada unidad o departamento de la organización es responsable de su propia implementación de BI. Cada unidad puede tener su propio equipo de BI y su propio almacén de datos, lo que les brinda autonomía para desarrollar y administrar su propia solución de BI. Este modelo permite una mayor adaptación a las necesidades específicas de cada unidad, pero puede requerir una mayor coordinación y gestión de los datos a nivel organizativo.

• Modelo de implementación híbrido: este modelo combina elementos del modelo centralizado y descentralizado. Se establece un equipo central encargado de las actividades de BI a nivel organizativo, como la gestión de los datos y el diseño del almacén de datos. Sin embargo, se permite a las unidades o departamentos tener cierto grado de autonomía para el desarrollo y el análisis de los datos dentro de sus propias áreas.

• Modelo de implementación en la nube: con el avance de la tecnología en la nube, cada vez más organizaciones optan por implementar soluciones de BI en la nube. En este modelo, los datos se almacenan y procesan en la nube, lo que permite un acceso más rápido y escalable a la información. Los proveedores de servicios en la nube ofrecen plataformas y herramientas específicas para el desarrollo y la implementación de soluciones de BI.

Es importante tener en cuenta que la elección del modelo de implementación de BI dependerá de los recursos disponibles, la estructura organizativa, los requisitos del proyecto y las necesidades específicas de la organización. Es recomendable evaluar cuidadosamente cada modelo y adaptarlo según las circunstancias y objetivos de la organización.

### *Análisis Comparativo BI*

Este análisis tiene como objetivo destacar las características y ventajas de cada modelo.

#### *Modelo de implementación centralizado:*

**Características:** en este modelo, se establece un centro de excelencia o un equipo central encargado de implementar y administrar la solución de BI. El equipo central es responsable de la recopilación, limpieza y procesamiento de los datos, así como del diseño y mantenimiento del almacén de datos.<sup>(14)</sup>

#### *Ventajas:*

- Centralización del control y la gestión de la solución de BI.
- Mayor consistencia y calidad de los datos.
- Mejor capacidad para garantizar la seguridad y la integridad de los datos.
- Mayor capacidad para establecer estándares y mejores prácticas en toda la organización.

#### *Modelo de implementación descentralizado:*

**Características:** en este modelo, cada unidad o departamento de la organización es responsable de su propia implementación de BI. Cada unidad puede tener su propio equipo de BI y su propio almacén de datos, lo que les brinda autonomía para desarrollar y administrar su propia solución de BI.

#### *Ventajas:*

- Mayor adaptación a las necesidades específicas de cada unidad o departamento.

- Mayor capacidad para el análisis y la toma de decisiones a nivel local.
- Mayor agilidad y capacidad de respuesta a los cambios en los requisitos.
- Promoción de la colaboración y la responsabilidad en cada unidad.

#### Modelo de implementación híbrido:

Características: este modelo combina elementos del modelo centralizado y descentralizado. Se establece un equipo central encargado de las actividades de BI a nivel organizativo, como la gestión de los datos y el diseño del almacén de datos. Sin embargo, se permite a las unidades o departamentos tener cierto grado de autonomía para el desarrollo y el análisis de los datos dentro de sus propias áreas.<sup>(14,15)</sup>

#### Ventajas:

- Balance entre el control central y la adaptabilidad local.
- Promoción de la estandarización y las mejores prácticas a nivel central.
- Capacidad para atender las necesidades específicas de cada unidad o departamento.

#### Modelo de implementación en la nube:

Características: en este modelo, la solución de BI se implementa en la nube utilizando servicios y plataformas en la nube. Los datos se almacenan y procesan en la nube, lo que permite un acceso más rápido y escalable a la información.

#### Ventajas:

- Mayor flexibilidad y escalabilidad en la infraestructura de TI.
- Acceso a recursos de alta capacidad de procesamiento y almacenamiento en la nube.
- Reducción de costos en infraestructura y mantenimiento.
- Facilidad de acceso remoto a la información desde diferentes ubicaciones y dispositivos.

#### Análisis de modelos BI

El análisis de modelos de BI implica varios aspectos clave:<sup>(16)</sup>

- Análisis de requisitos: en esta etapa, se identifican y comprenden los requisitos y necesidades de información de los usuarios finales. Esto implica entrevistar a los usuarios, analizar los procesos de negocio y definir los indicadores clave de rendimiento (KPI) relevantes para la toma de decisiones.
- Análisis de fuentes de datos: se evalúan las fuentes de datos utilizadas en el modelo de BI, asegurándose de que sean confiables, completas y actualizadas. Se examina la calidad de los datos y se realizan actividades de limpieza y transformación para garantizar su integridad.
- Análisis de modelo dimensional: se revisa la estructura del modelo dimensional utilizado en el almacén de datos. Se evalúa si el esquema de estrella o copo de nieve es apropiado y si se han aplicado las mejores prácticas de modelado dimensional.
- Análisis de informes y paneles: se examinan los informes y paneles utilizados para la presentación de información. Se evalúa la claridad, relevancia y usabilidad de los mismos, así como la alineación con los requisitos y necesidades de los usuarios finales.

- Análisis de rendimiento: se realiza un análisis del rendimiento de la solución de BI, evaluando los tiempos de respuesta, la capacidad de escalabilidad y el rendimiento general del sistema.

- Análisis de seguridad: se verifica la seguridad implementada en la solución de BI para garantizar la protección de los datos sensibles y restringir el acceso no autorizado a la información.

#### Herramientas de Business Intelligence

Las herramientas de Business Intelligence (BI) son aplicaciones y software diseñados para ayudar a las organizaciones a recopilar, analizar y visualizar datos con el fin de tomar decisiones informadas y estratégicas. Estas herramientas ofrecen funcionalidades que permiten el procesamiento y la presentación de información de manera más eficiente y efectiva.<sup>(11)</sup>

A continuación, se presenta una descripción general de las herramientas de BI:

#### 1. Herramientas de extracción, transformación y carga (ETL):

- Ejemplos: Microsoft SQL Server Integration Services (SSIS), Informatica PowerCenter, Talend.
- Funcionalidad: estas herramientas permiten la extracción, transformación y carga de datos desde múltiples fuentes de datos hacia un almacén de datos o data warehouse. Ayudan a limpiar, organizar y estructurar los datos para su posterior análisis.

#### 2. Herramientas de almacenamiento y gestión de datos:

- Ejemplos: Oracle Database, Microsoft SQL Server, PostgreSQL.
- Funcionalidad: estas herramientas proporcionan sistemas de gestión de bases de datos (DBMS) para el almacenamiento y la gestión eficiente de grandes volúmenes de datos. Permiten la creación de modelos de datos, consultas, seguridad y administración de bases de datos.

#### 3. Herramientas de visualización y análisis de datos:

- Ejemplos: Tableau, Microsoft Power BI, QlikView.
- Funcionalidad: estas herramientas permiten la creación de paneles interactivos, gráficos, informes y tableros de control para visualizar y analizar datos. Ofrecen capacidades de arrastrar y soltar, filtrado, perforación y exploración de datos para facilitar la comprensión de la información.

#### 4. Herramientas de generación de informes:

- Ejemplos: SAP Crystal Reports, IBM Cognos Report Studio, Oracle Business Intelligence Publisher.
- Funcionalidad: estas herramientas permiten la creación y distribución de informes formateados y estructurados. Ofrecen opciones de diseño, formatos de salida personalizados y programación de informes automatizados.

### 5. Herramientas de minería de datos:

- Ejemplos: IBM SPSS Modeler, RapidMiner, Microsoft Azure Machine Learning.
- Funcionalidad: estas herramientas permiten el descubrimiento de patrones, tendencias y relaciones ocultas en grandes conjuntos de datos. Ofrecen algoritmos y técnicas de minería de datos, como clasificación, agrupamiento, regresión y análisis de asociación.

## CONCLUSIONES

El recorrido conceptual y técnico realizado permite sostener que una solución contemporánea para control de ventas y apoyo a la decisión debe integrar, de forma coherente, cuatro ejes: (i) una arquitectura web modular, segura y escalable; una experiencia de usuario centrada en tareas y evidencias de uso; un ciclo de desarrollo iterativo que reduzca el tiempo de entrega y el riesgo; y una capa analítica basada en datos de calidad y procesos reproducibles. En el plano arquitectónico, la separación de responsabilidades (presentación, lógica de negocio y datos) combinada con prácticas de microservicios o funciones serverless donde aporten ventajas claras, facilita el escalado selectivo, la resiliencia y el despliegue continuo. El cliente web —apoyado en estándares como PWA— habilita experiencias responsivas, rápidas y seguras, mientras que el servidor expone contratos estables vía REST o GraphQL, con políticas de autenticación y autorización consistentes y telemetría embebida.

Desde la perspectiva tecnológica, el stack propuesto —HTML5/CSS3/JavaScript en el front-end; PHP/Laravel o Python/Django en el back-end; MariaDB en la gestión relacional— ofrece madurez, comunidad, rendimiento y costos controlados. MariaDB, en particular, añade motores y optimizaciones relevantes frente a MySQL, manteniendo la compatibilidad necesaria para minimizar fricciones de migración. La decisión de incorporar WordPress puede ser táctica para módulos de contenido o portales informativos, siempre que se delimiten con precisión los límites entre capa transaccional y CMS, evitando acoplamientos que perjudiquen la analítica o la seguridad.

En metodología, la adopción de Scrum resulta pertinente para el caso CNET-La Maná porque prioriza funcionalidades con mayor retorno, facilita la inspección y adaptación continua, y promueve una cultura de calidad mediante incrementos potencialmente desplegables al final de cada sprint. Ello se ve reforzado por prácticas DevOps (integración y entrega continua, infraestructura como código, pruebas automatizadas) que acortan el “time to value” y elevan la confiabilidad del sistema.

En paralelo, la gobernanza del dato —condición de posibilidad del BI— exige un pipeline ETL robusto, catálogos y diccionarios de datos, control de calidad (validaciones, deduplicación, estandarización), y un modelo dimensional alineado a los KPIs de ventas. La elección entre una arquitectura de BI tradicional, moderna o en la nube debe responder a restricciones reales de presupuesto, competencias internas y requisitos de latencia, siendo razonable un modelo híbrido: datos transaccionales gobernados centralmente, con datamarts por dominio que preserven autonomía analítica sin sacrificar la consistencia.

Mirando a futuro, la hoja de ruta debería contemplar: consolidación de métricas y definiciones maestras; endurecimiento de la seguridad (cifrado en tránsito y en reposo, gestión de secretos, hardening de APIs); observabilidad end-to-end (logs, métricas, trazas); pruebas de rendimiento y escalabilidad; y, en la capa analítica, evolución hacia analítica aumentada con aprendizaje automático para pronósticos de demanda, segmentación de clientes y detección de anomalías en ventas. Entre las limitaciones previsibles se cuentan la calidad heterogénea de fuentes de datos legadas, la posible deuda técnica por integraciones previas y la necesidad de formación continua de los usuarios en lectura crítica de tableros. No obstante, la convergencia entre una arquitectura web bien diseñada, una experiencia de usuario enfocada, una disciplina ágil de construcción y una plataforma de BI gobernada sienta las bases para transformar datos operativos en decisiones oportunas y accionables, materializando mejoras sostenibles en eficiencia, transparencia y competitividad organizacional.

## FINANCIACIÓN

Ninguna.

## CONFLICTO DE INTERESES

Ninguno.

## CONTRIBUCIÓN DE AUTORÍA

*Conceptualización:* Franco Joel Lloacana Llugsa, Steven Ricardo Castillo Calderón.

*Curación de datos:* Franco Joel Lloacana Llugsa, Steven Ricardo Castillo Calderón.

*Análisis formal:* Franco Joel Lloacana Llugsa, Steven Ricardo Castillo Calderón.

*Redacción – borrador original:* Franco Joel Lloacana Llugsa, Steven Ricardo Castillo Calderón.

*Redacción – revisión y edición:* Franco Joel Lloacana Llugsa, Steven Ricardo Castillo Calderón.

## REFERENCIAS BIBLIOGRÁFICAS

1. Flores A. Qué son las aplicaciones web. gcfglobal. 2020. [https://edu\(gcfglobal.org/es/informaticabasica/que-son-las-aplicaciones-web/1/](https://edu(gcfglobal.org/es/informaticabasica/que-son-las-aplicaciones-web/1/)
2. Herraiz M. Diseño y desarrollo de una aplicación web como red social para aficionados a la automovilística. Valencia: Universidad Politécnica de Valencia; 2019. <https://riunet.upv.es/bitstream/handle/10251/173828/Herraiz%20-2019.pdf>
3. Brown A, Johnson R. The Client-Server Model in Network Applications. Int J Comput Netw Commun. 2021.
4. Ridge BV. Diferencias y consideraciones entre UI y UX en el desarrollo web. MEDIUM Multimedia Agencia de Marketing Digital. 2023. <https://www.mediummultimedia.com/web/cual-es-mejor-ui-o-ux/>
5. Carpio Thalia; Saltos Nelly. Desarrollo de un sistema web para el proceso de matrícula en línea de la unidad educativa “Carlos Lozada Quintana” en el cantón La Maná. 2023. <http://repositorio.utc.edu.ec/handle/27000/10184>
6. IONOS EE. Lenguajes de programación web: los más usados en Internet. IONOS Digital Guide. 2019 ago 7. <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/lenguajes-de-programacion-web/>
7. Juanacio Lalanguí B. Desarrollo de un prototipo de página web con un recorrido virtual interactivo del grupo Étnico Shuar. 2021. <https://www.dspace.uce.edu.ec/entities/publication/20e07a5f-dafc-4a46-95d9-4612ca1679b1>
8. Ágiles P. Qué es SCRUM. proyectosagiles.org. 2021. <https://proyectosagiles.org/que-es-scrum/>
9. Perea M. Métodos empíricos de la investigación. Instituto de Ciencias

- Económico. 2020. [https://www.uaeh.edu.mx/docencia/P\\_Presentaciones/icea/asignatura/mercadotecnia/2](https://www.uaeh.edu.mx/docencia/P_Presentaciones/icea/asignatura/mercadotecnia/2)
10. Pelissier C. Programación con PHP. Valparaíso: Universidad Técnica Federico Santa María; 2002. <https://iestpcabana.edu.pe/wp-content/uploads/2021/09/Programacion-con-PHP.pdf>
11. Rodríguez Rodríguez J. Investigación y desarrollo de un sistema de inteligencia de negocios empresarial, mediante una base de datos Hana. 2019. <https://www.dspace.uce.edu.ec/entities/publication/f57d51d3-ccb6-451a-9af4-68d2a09678fe>
12. Valtx. Metodologías para el desarrollo de software: ¿Qué son y para qué sirven? valtx.pe. 2022. <https://www.valtx.pe/blog/metodologias-para-el-desarrollo-de-software-que-son-y-para-que-sirven>
13. Ramón RA. Metodología para procesos de inteligencia de negocios con mejoras en la extracción y transformación de fuentes de datos, orientado a la toma de decisiones. 2019. <https://dialnet.unirioja.es/servlet/tesis?codigo=221922>
14. Subía García JL. Implementación de una solución Business Intelligence para el análisis de la gestión del Ministerio de Desarrollo Urbano y Vivienda. 2013. <https://www.dspace.uce.edu.ec/entities/publication/aee7450e-c897-46c1-9702-88139689d683>
15. Vélez E. Implementación de una aplicación web ecommerce para el salón de eventos. 2021. <http://repositorio.unesum.edu.ec/bitstream/53000/3393/1/RIVERA%20VELEZ%20E>.
16. Kohn PS. Métodos de investigación: qué son y cómo elegirlos. QuestionPro. 2023. <https://www.questionpro.com/blog/es/metodos-de-investigacion/>