INTRODUCCIÓN A PHP

C.E.A.I SENA

MARIA JOSÉ MARIN RIOS

Contenido

[3.2 Actividades de Contextualización e Identificación de conocimientos necesarios para el aprendizaje. 2](#_Toc179366096)

[1 ¿Qué es un lenguaje de programación? 2](#_Toc179366097)

[2 ¿Cuáles son las diferencias entre programación web y programación de escritorio? 2](#_Toc179366098)

[3 ¿Qué es PHP? 2](#_Toc179366100)

[4 ¿Cuáles son las principales características de PHP?. 2](#_Toc179366101)

# Actividad 3.2

# ¿Qué es un lenguaje de programación?

Un lenguaje de programación es un conjunto de reglas, símbolos y palabras que permiten a los desarrolladores crear programas que las computadoras pueden entender y ejecutar. A través de un lenguaje de programación, los humanos pueden dar instrucciones a una máquina para realizar tareas específicas, como cálculos, manipulación de datos, control de hardware, entre otras.

Los lenguajes de programación son esenciales para el desarrollo de software y varían en su nivel de abstracción, funcionalidad y aplicaciones. Existen diferentes tipos de lenguajes de programación, algunos más cercanos al lenguaje humano y otros más orientados a las operaciones internas del hardware.

**Características de un lenguaje de programación:**

Sintaxis: Es la estructura o reglas que deben seguirse al escribir el código. Cada lenguaje tiene su propia sintaxis que define cómo deben estructurarse las instrucciones para ser comprendidas y ejecutadas por la computadora.

Semántica: Se refiere al significado de las instrucciones y cómo son interpretadas por la computadora. Aunque dos lenguajes pueden tener una sintaxis diferente, pueden tener una semántica similar si las instrucciones realizan las mismas operaciones.

Compilación o Interpretación: Algunos lenguajes requieren que el código fuente sea compilado antes de ser ejecutado, como C o Java. Otros son interpretados, lo que significa que el código se traduce a medida que se ejecuta, como Python o JavaScript.

Fuentes:

Libro: "The Pragmatic Programmer" por Andrew Hunt y David Thomas, que proporciona una guía general sobre lenguajes y paradigmas de programación.

"Introduction to Programming" (Python & Java) en Coursera, cursos de instituciones como Stanford y Harvard.

Documentación oficial de lenguajes como Python, Java, C, y JavaScript.

# ¿Cuáles son las diferencias entre programación web y programación de escritorio?

La programación web y la programación de escritorio son dos enfoques distintos en el desarrollo de software, cada uno con sus características, herramientas y usos específicos. Aunque ambos tienen como objetivo crear aplicaciones que faciliten tareas para los usuarios, las plataformas en las que se ejecutan, las tecnologías que utilizan y las habilidades requeridas para cada uno varían considerablemente.

1. Plataforma de Ejecución

Programación web:

Las aplicaciones web se ejecutan en un navegador web y son accesibles a través de Internet o una red local.

Funcionan en múltiples plataformas (Windows, macOS, Linux, dispositivos móviles) siempre que el dispositivo tenga un navegador compatible y conexión a la red.

Las aplicaciones web no necesitan instalación, ya que el usuario solo accede a ellas mediante una URL.

Programación de escritorio:

Las aplicaciones de escritorio se instalan y ejecutan directamente en el sistema operativo del usuario, como Windows, macOS o Linux.

Generalmente son más específicas de la plataforma y pueden necesitar ser adaptadas o reescritas para funcionar en diferentes sistemas operativos.

Requieren instalación, actualizaciones y mantenimiento a nivel local en el dispositivo del usuario.

2. Tecnología Utilizada

Programación web:

Los desarrolladores web utilizan principalmente tecnologías de frontend y backend.

Frontend: Se refiere a la parte de la aplicación que interactúa directamente con el usuario (navegador). Utiliza lenguajes como HTML, CSS y JavaScript para diseñar y hacer que la interfaz sea interactiva.

Backend: Se ocupa del procesamiento del servidor, bases de datos, lógica de negocio, etc. Usa lenguajes y frameworks como PHP, Python, Ruby, Node.js, Java, entre otros.

Las aplicaciones web a menudo se comunican con servidores remotos a través de APIs y protocolos HTTP/HTTPS.

Se pueden integrar con bases de datos externas y sistemas en la nube.

Programación de escritorio:

Los lenguajes y frameworks más comunes para el desarrollo de aplicaciones de escritorio son C++, C#, Java, Python, Electron (JavaScript para apps de escritorio multiplataforma) y Swift (para macOS).

Las aplicaciones de escritorio suelen acceder a recursos locales como el sistema de archivos, hardware (impresoras, cámaras, etc.) y bases de datos instaladas localmente.

Fuentes:

MDN Web Docs: Desarrollo web

Microsoft Docs: Desktop Application Development

# ¿Qué es PHP?

PHP (acrónimo de "Hypertext Preprocessor") es un lenguaje de programación de propósito general, de código abierto, que se utiliza principalmente para el desarrollo web. PHP se ejecuta en el servidor y es responsable de generar contenido dinámico en las páginas web. Aunque originalmente PHP significaba "Personal Home Page", su propósito y funcionalidad han evolucionado significativamente, convirtiéndose en uno de los lenguajes más populares para el desarrollo web del lado del servidor.

Ventajas de usar PHP:

Facilidad de uso: PHP es conocido por su curva de aprendizaje relativamente suave, lo que lo hace accesible tanto para principiantes como para desarrolladores experimentados.

Gratuito y de código abierto: PHP es software de código abierto, lo que significa que es gratuito para su uso y modificación. Esto ha ayudado a su rápida adopción.

Compatibilidad multiplataforma: Funciona en casi todos los servidores web y sistemas operativos, lo que lo hace muy flexible.

## 4 ¿Cuáles son las principales características de PHP?

Características clave de PHP:

Lenguaje del lado del servidor: PHP se ejecuta en un servidor web, y su salida se envía al navegador web en forma de HTML. Los usuarios no pueden ver el código PHP porque el servidor lo procesa antes de que se entregue la página web al usuario final.

Interacción con bases de datos: PHP facilita la interacción con bases de datos como MySQL, PostgreSQL, Oracle, y más, lo que permite crear aplicaciones web dinámicas que pueden almacenar y recuperar información. Es muy común en sistemas como blogs, foros, tiendas en línea, y sistemas de gestión de contenido (CMS).

Integración sencilla con HTML, CSS y JavaScript: PHP se puede incrustar fácilmente en código HTML, lo que lo hace ideal para el desarrollo de páginas web dinámicas. También puede trabajar junto con JavaScript (del lado del cliente) y CSS (para el estilo), formando la base de las aplicaciones web modernas.

Flexibilidad: PHP es flexible y compatible con una amplia gama de plataformas, servidores y bases de datos. Además, puede ejecutarse en casi cualquier sistema operativo, como Windows, macOS y varias distribuciones de Linux.

Comunidad y recursos: PHP tiene una gran comunidad de desarrolladores, lo que significa que hay abundantes recursos, bibliotecas, marcos y documentación disponible. Frameworks como Laravel, Symfony, y CodeIgniter ayudan a simplificar y estandarizar el desarrollo con PHP.

Conclusión:

PHP es una opción sólida para el desarrollo web debido a su simplicidad, flexibilidad y amplio soporte. Su uso ha crecido enormemente en países como Colombia, donde las empresas buscan soluciones rápidas, eficientes y accesibles para el desarrollo de plataformas en línea.

Fuentes:

PHP.net

W3Schools

MDN Web Docs