



# Unidad # 3

## Incorporando tecnologías de Servidor Web

Programación Computacional IV

Gisela Yasmín García Espinoza



**UNIVERSIDAD  
GERARDO BARRIOS**  
Líderes en Gestión del Conocimiento





## Competencia de la asignatura

Aplicar los lenguajes de programación y tecnologías para la Web con el propósito de utilizarlos en el desarrollo de aplicaciones basándose en estándares internacionales de desarrollo de software trabajando de forma individual y colaborativa.

# Contenidos a desarrollar

- Introducción
- Socket.IO





Socket.IO



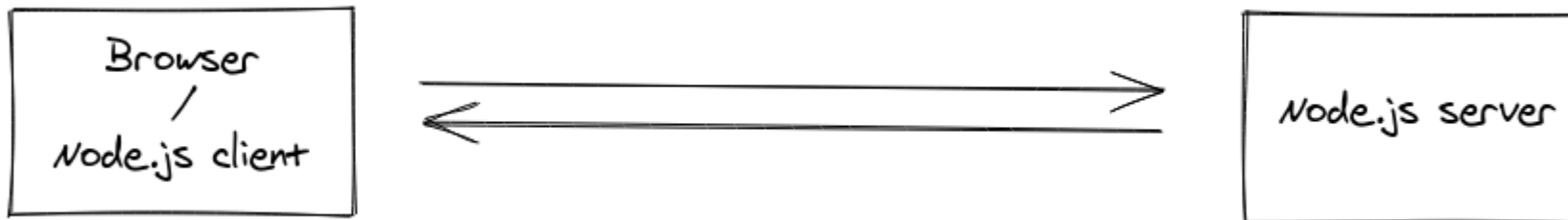
# Introducción

Los websockets son una tecnología que permite una comunicación bidireccional entre cliente y servidor sobre un único socket TCP. En cierta manera es un buen sustituto de AJAX como tecnología para obtener datos del servidor, ya que no tenemos que pedirlos, el servidor nos los enviará cuando haya nuevos.

# Socket.IO

Socket.IO es una biblioteca que permite la comunicación en tiempo real, bidireccional y basada en eventos entre el navegador y el servidor, consiste en:

- Un servidor Node.js
- Una biblioteca cliente de Javascript para el navegador (que también se puede ejecutar desde Node.js)



Fuente: <https://socket.io/images/bidirectional-communication.png>

# Socket.IO

El cliente intentará establecer una WebSocket conexión si es posible y, si no, recurrirá al sondeo largo HTTP.

WebSocket es un protocolo de comunicación que proporciona un canal full-duplex y de baja latencia entre el servidor y el navegador.

*Socket.IO NO es una implementación de WebSocket. Aunque Socket.IO de hecho usa WebSocket como transporte cuando es posible, agrega metadatos adicionales a cada paquete. Es por eso que un cliente WebSocket no podrá conectarse correctamente a un servidor Socket.IO, y un cliente Socket.IO tampoco podrá conectarse a un servidor WebSocket simple.*

# Socket.IO

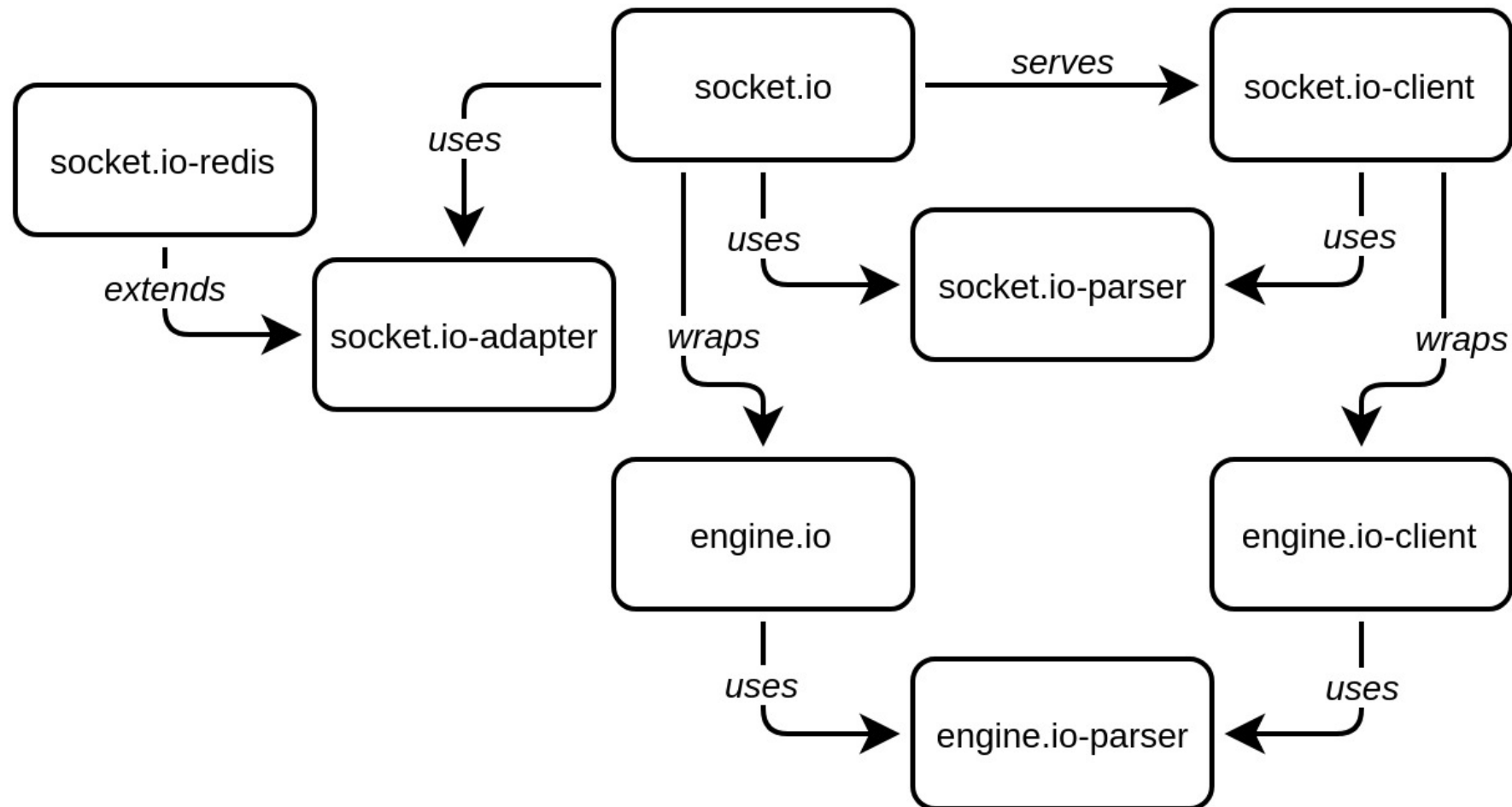
Socket.IO se creó en 2010. Se desarrolló para utilizar conexiones abiertas para facilitar la comunicación en tiempo real, todavía un fenómeno relativamente nuevo en ese momento.

Socket.IO permite la comunicación bidireccional entre cliente y servidor. Las comunicaciones bidireccionales se habilitan cuando un cliente tiene Socket.IO en el navegador y un servidor también ha integrado el paquete Socket.IO. Si bien los datos se pueden enviar de varias formas, JSON es el más simple.

Para establecer la conexión e intercambiar datos entre el cliente y el servidor, Socket.IO usa Engine.IO . Esta es una implementación de nivel inferior que se usa bajo el capó. Engine.IO se utiliza para la implementación del servidor y Engine.IO-client se utiliza para el cliente.



# Socket.IO



Fuente: <https://images.ctfassets.net/ee3ypdtck0rk/1Lj7lbqX54WCiHI2uVVL3x/e145c3b8194f84b5c1219e6dd3dcb9da/introduction-d4b7a9009c7a7a0306c050af52a4f8be8ddebbce4882ad8128c752da505f8b9.png>

# Socket.IO | Características

- Confiabilidad (respaldo al sondeo largo HTTP en caso de que no se pueda establecer la conexión WebSocket)
- Reconexión automática
- Almacenamiento en búfer de paquetes
- Expresiones de gratitud
- Transmitiendo a todos los clientes o a un subconjunto de clientes (lo que se llama "Sala")
- Multiplexación (lo que es llamado "espacio de nombres")

## Tabla de actividades

Nombre de la actividad	Portafolio del estudiante
Tipo de actividad	Tarea
Tipo de participación	Individual
Competencias específica de la asignatura	Integrar tecnologías y arquitecturas web, paradigmas de programación y gestión de datos con DBMS para desarrollar aplicaciones web mediante el lenguaje de programación PHP trabajando con un enfoque lógico y aplicando las buenas prácticas de programación de manera individual y colaborativa.
Instrucciones	<p>Después de dar lectura al contenido de la semana y participar activamente en la video conferencia deberá: elaborar un sitio web (herramienta a utilizar es elección del estudiante) que contenga el portafolio del estudiante el cual contendrá lo siguiente:</p> <ul style="list-style-type: none"> <li>• Portada</li> <li>• Introducción</li> <li>• Contenidos de la asignatura</li> <li>• Breve resumen de contenidos por computo <ul style="list-style-type: none"> <li>• Incluir evidencias de evaluaciones</li> <li>• Tareas</li> <li>• Apuntes</li> <li>• Investigaciones</li> <li>• Guías resueltas</li> </ul> </li> <li>• Conclusión</li> </ul>
Fecha de entrega	La fecha límite de participación será el domingo al final de la semana, Hora limite: 23:59 p.m
Instrumento de evaluación	Rúbrica
Ponderación	30% Laboratorio II

## Recursos Complementarios

Recurso	Título	Cita Referencial
Sitio Web	Chat con Socket.io y node.js	<a href="https://medium.com/@noufel.gouirhate/build-a-simple-chat-app-with-node-js-and-socket-io-ea716c093088">https://medium.com/@noufel.gouirhate/build-a-simple-chat-app-with-node-js-and-socket-io-ea716c093088</a>
Sitio Web	Full socket.io client and server example	<a href="https://gist.github.com/luciopaiva/e6f60bd6e156714f0c5505c2be8e06d8">https://gist.github.com/luciopaiva/e6f60bd6e156714f0c5505c2be8e06d8</a>

¿Preguntas?

# ¡Muchas gracias!