

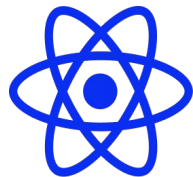
# ReactJS Developer

Glosario de términos

- **Aplicación de página única:** se emplea para sumar una única página HTML y componentes necesarios (JavaScript y CSS) para realizar la aplicación.
- **Bundlers:** su función es tomar el código JavaScript, más CSS y los junta en archivos mejorados para los navegadores.
- **CDN** (siglas en inglés de *Content Delivery Network*): trata de la Red de Entrega de Contenido fijo en caché desde una red de servidores alrededor del mundo.
- **Compiladores:** al usar el código JavaScript, éste lo modifica y lo reintegra en un formato diferente.
- **Componentes:** son fragmentos de código que retornan un elemento de React para ser renderizado en una página específica.
- **CSS:** Lenguaje de reglas de estilo (colores de fondo, tipografía y columnas) para adaptar contenido HTML.
- **Elementos:** son agrupaciones de elaboración de una aplicación de React.
- **ES6, ES2015, ES2016:** son versiones del estándar de Especificación de Lenguaje ECMAScript, en el que JavaScript es usada para ponerse en marcha.

- **Estado:** se traduce al cambio de los datos, a través del tiempo, de un componente.
- **Eventos:** son piezas de React que tienen diferencias en su concordancia y/o jerarquía.
- **Framework:** plantilla que sintetiza la elaboración de una tarea o proyecto.
- **HTML:** lenguaje empleado para sistematiza el contenido web: imágenes y vídeos, párrafos, encabezados y tablas de datos.
- **JavaScript:** lenguaje de programación para ejecutar funciones específicas en páginas webs, actualizaciones interactivas como animaciones, reproducción de vídeos, etc.
- **JSX:** es una prolongación de construcción para JavaScript. Una especie de guía del lenguaje con JavaScript.
- **Keys:** se trata de una propiedad especial (cadena de texto) para hacer un ajuste de elementos. Las Keys permiten al React fijar qué elementos se han modificado, sumado o eliminados. Responden a una identificación única de los datos.
- **Métodos de ciclo de vida:** Es una operatividad particular que se ejecuta en las etapas de un componente.

- **Package managers:** herramientas para organizar las dependencias de los proyectos.
- **Props:** son accesos de un componente de React. Por lo general, su información es enviada desde un componente creador a un componente resultado o descendiente. Los Props no pueden ser cambiados.
- **Referencias:** son atributos especiales que se pueden incluir en un componente.



**Ahora sí,  
¡comencemos!**

