**UNIDAD 4: ReactJS**

1. [UNIDAD 4: Introducción](https://learn.nextu.com/mod/page/view.php?id=3608&pid=P_WEBDEV_V2)

**Bienvenido a Herramientas, Build Systems y tecnologías emergentes**

ReactJS es conocida por ser una de las librerías JavaScript con mayor rendimiento gracias a todas las herramientas y/o características que ésta posee, especialmente por su DOM virtual. En esta Unidad veremos cómo es el funcionamiento de ReactJS y por qué su rendimiento es tan alto.

**Puntos de aprendizaje**

Unidad 4: ReactJS

**Los objetivos del Programa que corresponden a esta Unidad son:**

* Crear ambientes de programación y a gestionar recursos.
* Implementar la construcción de sitios web a partir de componentes.

**Los objetivos del Curso 8 que corresponden a esta Unidad son:**

* Identificar y apropiar tendencias web.
* Aplicar el concepto ambiente web.
* Implementar web components.

1. Lección 1: ¿Qué es ReactJS?
   1. [Introducción a ReactJS](https://learn.nextu.com/mod/lesson/view.php?id=3609&pid=P_WEBDEV_V2)

Standard de desarrollo – Web components / Componentes Web, que cuentan con:



Esto con el fin de poder reutilizarlo cuantas veces se desee y dentro de otra aplicación si es necesario.

ReactJS:

* + Es bastante eficiente
  + Es bastante funcional

Librería escrita en JavaScript y desarrollada por Facebook.



Características de ReactJS:

* + Rendimiento y eficiencia
  + Flujo de datos unidireccional – se dice que es una de la forma más eficiente de datos. Más sencillo para encontrar errores.
  + DOM virtual – principal característica, y este se encarga de guardar en memoria un estado inicial de cada uno de los componentes que conforma la app, y de hacerle un seguimiento exhaustivo de cada uno de ellos con el fin de identificar cuales presentan cambios. Una vez lo detecta se encarga de renderizar únicamente los componentes que presentan modificaciones sin la necesidad de renderizar todos los componentes.
  + Compatibilidad: compatibilidad con los diferentes navegadores y compatibilidad con otras librerías y frameworks javascript
    - Compatibilidad con otros navegadores es bastante buena, incluso en versiones más antiguas como en el caso de internet explorer.
    - Compatibilidad con otros Frameworks y/o librerías JavaScript: es muy buena siempre y cuando, estas no tengan un control sobre el DOM ya que React es quien se encarga de controlarlo.
  + Fácil de aprender
  + Orientada a componentes lo que facilita: crear, actualizar, modificar y unir componentes en cualquier momento durante el proceso de desarrollo y soporte

Aplicaciones con funcionalidades desarrolladas con React:

Facebook e Instagram

Netflix

HipChat (Atlassian) comunicaciones en equipo de trabajo

* 1. Actividad 1

Tiempo de pensar

Hemos visto la introducción de la librería de JavaScript llamada ReactJS, sus principales características y algunas de las aplicaciones más usadas en la actualidad en las que se usó esta librería para la creación de ciertos componentes y/o funcionalidades durante su proceso de desarrollo, así que te invitamos a que realices esta actividad para que afiances tus conocimientos.  
  
Instrucciones:  
  
Lee el enunciado con atención y realiza la asociación que consideres correcta.  
  
Enunciado:  
  
ReactJS es una librería escrita en código JavaScript y desarrollada por la reconocida empresa Facebook, con el fin de facilitar el proceso de creación de componentes Web que puedan ser reutilizados. Esta herramienta tiene unas características que la hacen especial, a continuación, verás algunas de ellas listadas y tu misión será asociales la definición que les corresponde. ¡Adelante!



* 1. Lectura: Ventajas y desventajas de ReactJS

Ventajas & desventajas de ReactJS



Como ya lo sabes, ReactJS es una de las librerías Front-end de JavaScript más potentes que existe actualmente y que cuenta con una comunidad de desarrolladores que hacen que cada vez mejore más y tenga nuevas características para ofrecerle a quienes lo usan en la creación de sus aplicaciones Web. ReactJS, como cualquier otra librería JavaScript, tiene sus ventajas y desventajas, así que veamos cuáles son:

##### **Ventajas de ReactJS:**

Es una herramienta fácil de aprender, ya que cuenta con una documentación y una comunidad de desarrolladores muy amplia que siempre está abierta a resolver dudas o inquietudes. Adicionalmente, a nivel de sintaxis es simple, ya que si comprendemos apropiadamente el concepto de Web components y cómo es esta tendencia de trabajo para la creación de aplicaciones Web, será bastante fácil aprender ReactJS.



Al trabajar con componentes independientes, la creación y el mantenimiento de nuestras aplicaciones se convierte en una tarea muy sencilla y rápida. Detectar los errores no toma mucho tiempo y agregar nuevas funcionalidades a la aplicación se convierte en algo bastante simple.



Su capacidad para renderizar rápidamente cada uno de los componentes de la aplicación permite brindar una mejor experiencia de usuario.



Es posible combinar React con otros frameworks o bibliotecas para lograr funcionalidades necesarias con las que este no cuenta pero con los que se está combinando y/o integrando.

##### REACT + JS LIBRARIES

Contar con una comunidad de desarrollo tan amplia y con un grupo de desarrolladores tan expertos y reconocidos detrás de su creación hace pensar que es una librería que estará un buen periodo de tiempo en el mundo del desarrollo Front-end.



ReactJS es de las pocas librerías JavaScript que se integra y soporta versiones antiguas de Internet Explorer, de forma que, si dentro de los requisitos de desarrollo de tu aplicación Web se encuentra el soporte de versiones antiguas de Internet Explorer, ReactJS es la librería que necesitas.

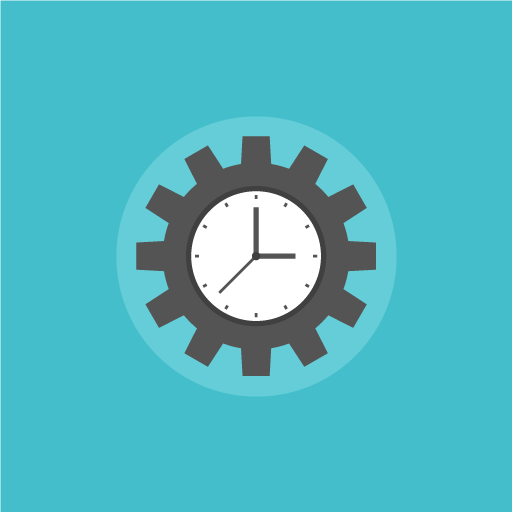


##### **Desventajas de ReactJS:**

React, al tener un DOM virtual que se encarga de actualizar el estado y el contenido de cada componente de manera individual, no se lleva bien con librerías como jQuery que buscan tener una manipulación total del DOM, lo que en ocasiones, al juntarlas, causa que no se muestre o no se renderice la aplicación por la compatibilidad de éstos.



Si se tiene un volumen de datos complejo y bastante amplio, en algunas ocasiones el tiempo de renderizado puede tomar más tiempo de lo esperado.



Aunque mencionamos que React es compatible con otras librerías y frameworks que agregan ciertas funcionalidades a nuestras aplicaciones, al momento de compilarlas y desplegarlas puede tomar un poco más de tiempo, ya que debe instalar dependencia de dichas librerías que se usaron para agregar funciones requeridas y que, como mencionamos, React no posee, por eso lo conocen como el “Frankenstein” de las librerías JavaScript.

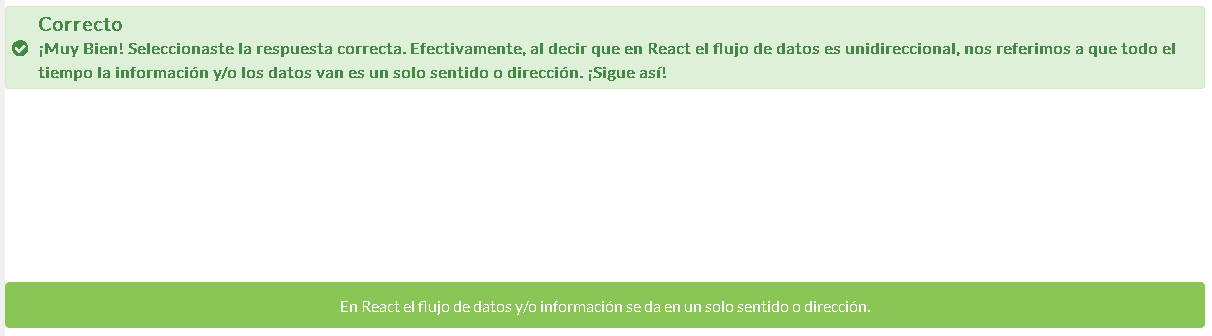


Como podemos ver, ReactJS tiene más ventajas que desventajas. A decir verdad, es una librería muy potente y sencilla de implementar para trabajar la parte visual de nuestra aplicación y con el pasar del tiempo continuará creciendo y mejorando para que todas estas desventajas empiecen a desaparecer.

* 1. Actividad 2

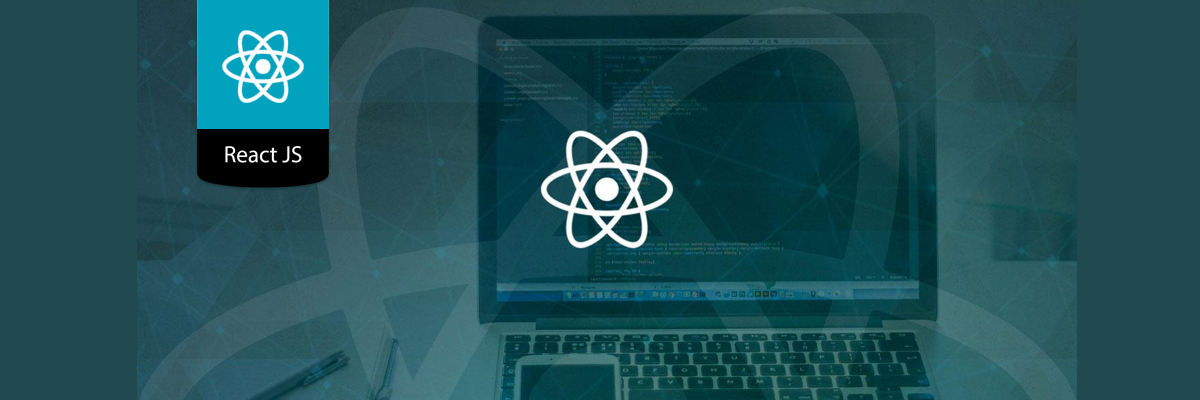
Tiempo de pensar

Hemos estudiado las diferentes ventajas y desventajas que nos ofrece ReactJS al momento de utilizarla para construir las vistas de nuestras aplicaciones Web, así que es hora de afianzar tus conocimientos a través de la siguiente actividad.  
  
Instrucciones:  
  
Lee el siguiente enunciado y selecciona la respuesta correcta.  
  
Enunciado:  
  
Al referirnos a React como una librería cuyo flujo de datos es unidireccional, estamos afirmando que:



* 1. Lectura: ¿Cómo funciona ReactJS?

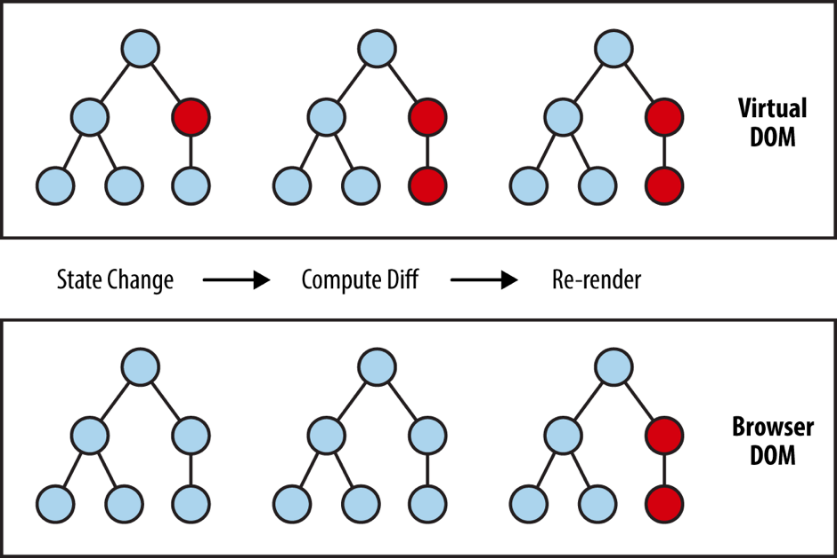
**¿Cómo funciona ReactJS?**



ReactJS es conocida por ser una de las librerías JavaScript con mayor rendimiento gracias a todas la herramientas y/o características que ésta posee, especialmente por su DOM virtual, pero veamos cómo es el funcionamiento de ReactJS que hace que su rendimiento sea tan alto.

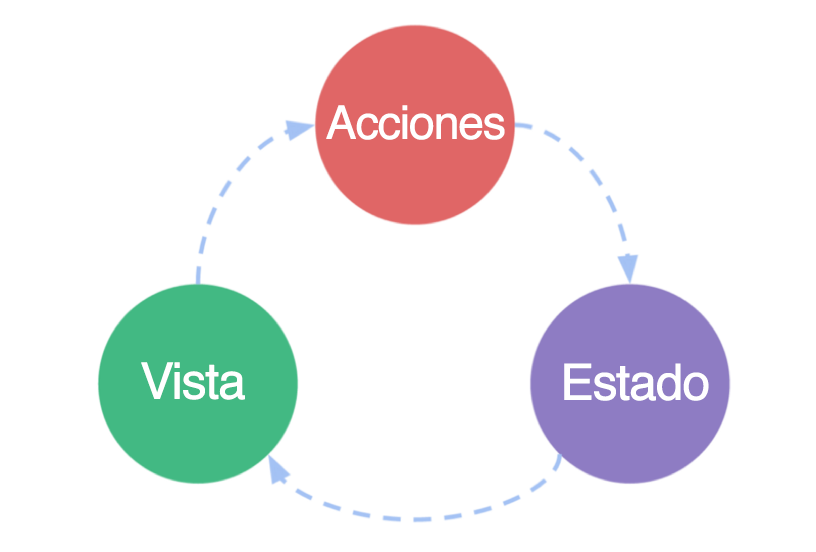
**Funcionamiento de react**

ReactJS está creado con el fin de de manejar funciones que permitan tener un control muy preciso sobre los diferentes componentes que construimos y que conforman nuestra página Web, dicho control se logra a través de un elemento fundamental conocido como el estado o state. La primera vez que nuestra aplicación es renderizada, el DOM virtual de React guarda en memoria el estado inicial de todos lo componentes de nuestra página y los mantiene en constante vigilancia. Una vez ocurre algún cambio en alguno de los estados de los componentes de nuestra aplicación, el ya mencionado DOM virtual se encarga de identificar dichos componentes que sufrieron modificaciones de estado comparándolos con el estado inicial que tiene almacenado en memoria y se da el proceso por el cual se considera que el rendimiento de React es tan alto. El DOM virtual se encarga de renderizar nuevamente los componentes que presentan cambios, guardando este nuevo estado en memoria y sin tener la necesidad de cargar y renderizar nuevamente nuestra aplicación para notar los cambios que presentan nuestros componentes.



**Flujo de datos:**

Dentro del funcionamiento de React, hay una característica y/o punto muy importante que debemos mencionar: el flujo de datos unidireccional, también conocido como flujo de datos en un solo sentido. Éste permite planear la forma de construir las aplicaciones más complejas e identificar los diferentes errores que se pueden presentar en éstas. A diferencia de las demás librerías que manejan un flujo de datos bidireccional, el flujo en un solo sentido permite hacer un seguimiento muy detenido al funcionamiento de nuestra aplicación y transporte de información, y así obtener una respuesta mucho más rápida. Es aquí donde podemos respaldar lo que hablábamos inicialmente cuando dijimos que React es una de las librerías con un alto rendimiento a comparación de las demás.

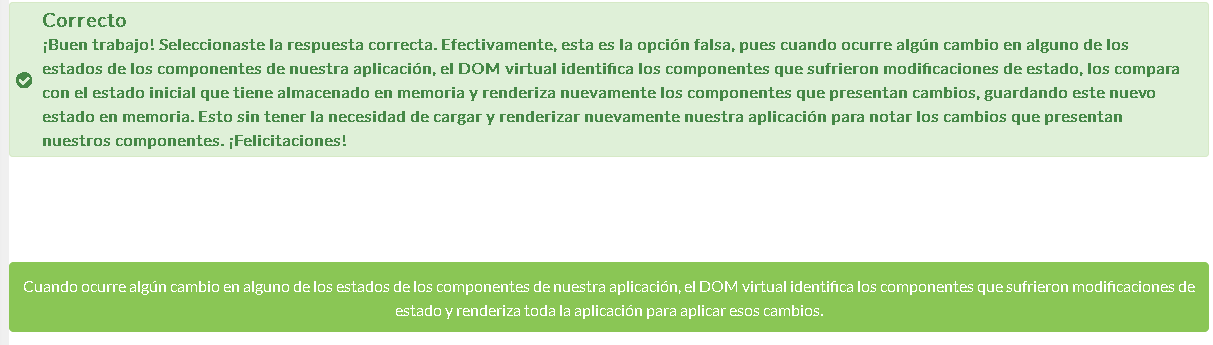


Esta es la razón por la que el funcionamiento de ReactJS es tan alto, pues cada una de las características o herramientas que la componen tiene una tarea en específico que va cumpliendo a medida que los componentes van cambiando y se le van mostrando al usuario, reduciendo los tiempos de carga y permitiendo reutilizar dichos componentes para desarrollar futuras aplicaciones.

* 1. Actividad 3

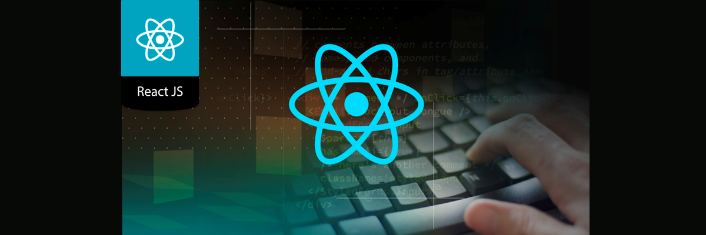
Tiempo de pensar

*Ahora que sabes cómo es el funcionamiento de ReactJS, te invitamos a reforzar tus conocimientos con la siguiente actividad.  
  
Instrucciones:  
  
Lee el siguiente enunciado y selecciona la respuesta correcta.  
  
Enunciado:  
  
A continuación te presentamos algunas afirmaciones sobre el funcionamiento de ReacJS, tu misión será encontrar la opción que contiene una afirmación que no corresponde con lo que has visto hasta ahora. ¡Adelante!*



* 1. Lectura: Por qué trabajar con ReactJS

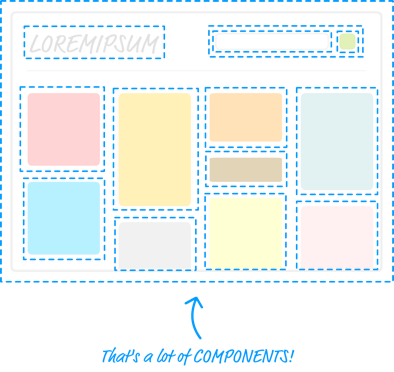
¿Por qué trabajar con ReactJS?



Existen diferentes librerías y frameworks Front-end de JavaScript que nos permiten crear las vistas de nuestras aplicaciones Web de forma rápida y eficiente, lo que dificulta la decisión cuando debemos seleccionar una de estas librerías para implementar en la creación de nuestros proyectos. ReactJS es una de ellas, de la cual hemos venido hablando, ésta nos ofrece muchas características para trabajar, así que te invitamos a explorar algunas de las razones por las cuales deberíamos seleccionarla para crear la interfaz gráfica de nuestras aplicaciones Web:

**WebComponents o Componentes Web**

En la actualidad, el desarrollo de componentes es una tendencia o filosofía de trabajo que se está implementado fuertemente en el proceso de creación de aplicaciones Web y móviles. ReactJs y React Native nos ofrecen la posibilidad de trabajar y crear nuestros propios componentes, los cuales podemos reusar y modificar para construir diferentes aplicaciones en el momento en que sea necesario, estos componentes tienen un ciclo de vida dentro de la aplicación.



**Eficiencia**

El hecho de que React tiene su propio DOM virtual, donde cada uno de los componentes de nuestra aplicación está activo, hace que el rendimiento de la aplicación sea más flexible y amplio, ya que React puede saber qué componente cambió y en qué momento lo hizo sin actualizar toda la aplicación sino solo esa parte que sufrió cambio, lo que hace que el usuario no tenga tiempo de espera mientras se recarga nuevamente la aplicación sino que con ésta mostrándose normalmente en pantalla podrá notar que los componentes se van actualizando uno a uno sin afectar la carga o visualización de los demás.



**Mejoramiento y crecimiento continuo**

Los ingenieros de desarrollo de Facebook fueron los encargados de crear esta librería y una vez fue publicada, la comunidad de desarrolladores empezó a aportar. Esto lo vienen haciendo hasta la fecha para que la librería se mantenga y tenga un constante crecimiento y mejoramiento.



**Compatibilidad con los navegadores**

ReactJS es de las pocas librerías JavaScript que se integra y soporta versiones antiguas de Internet Explorer de manera que, si dentro de los requisitos de desarrollo de tu aplicación Web se encuentra el soporte de versiones antiguas de Internet Explorer, ReactJS será la librería que necesitas. Adicionalmente, se integra a la perfección con los demás navegadores como Safari, Google Chrome, Opera, Firefox, etc.



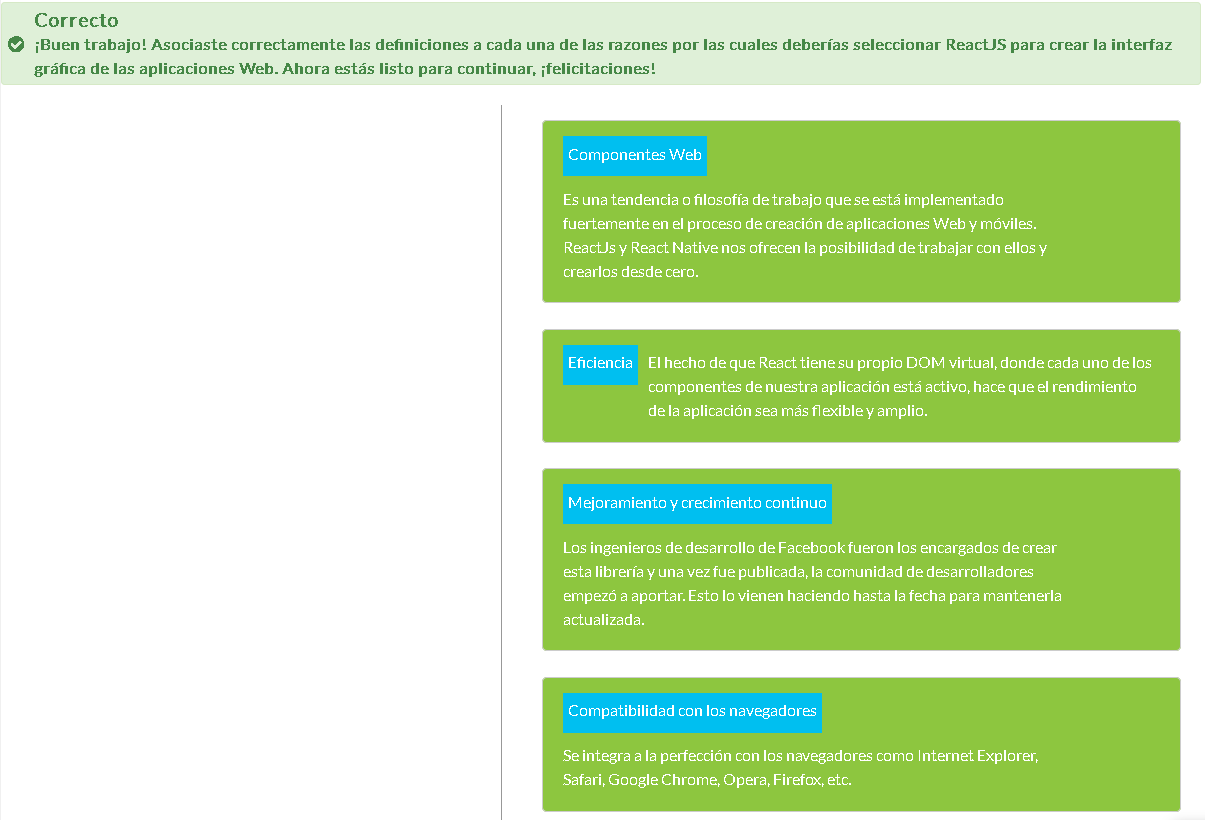
Otras de las razones para elegir ReactJS como la librería Front-end para crear nuestras aplicaciones Web son:

* Navegadores como Google Chrome tienen herramientas y extensiones en la Chrome Store que nos permiten hacerle seguimiento a nuestro código a través del navegador y hacer seguimiento de cómo va funcionando el DOM virtual de ReactJS.
* Cuando se crea una aplicación Web con ReactJS, si se desea crear una aplicación móvil en la parte del Front-end existe React Native, la cual tiene una estructura similar y la transición no es es compleja.
  1. Actividad 4

Tiempo de pensar

Ahora que sabes por qué trabajar con ReactJS, te invitamos a que valides tus conocimientos con la siguiente actividad.  
  
Instrucciones:  
  
Lee atentamente el enunciado y realiza la asociación que consideres correcta.

Enunciado:  
  
ReactJS nos ofrece varias razones por las cuales deberíamos seleccionarla para crear la interfaz gráfica de nuestras aplicaciones Web. A continuación, verás esas razones, las cuales deberás asociar con sus respectivas definiciones. ¡Adelante!



1. Lección 2: Configuración ReactJS
   1. [Lectura: Babel.JS](https://learn.nextu.com/mod/lesson/view.php?id=3610&pid=P_WEBDEV_V2)
   2. 3.2. Actividad 1
   3. 3.3. Configuración del ambiente
   4. 3.4. Actividad 2
   5. 3.5. Lectura: WebPack
   6. 3.6. Actividad 3
2. Lección 3: Web Components
3. Lección 4: Ciclo de vida del componente
4. Lección 5: Routers & Complementos
5. UNIDAD 4: Prueba