



UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

**Centro de
e-Learning**

Desarrollo en React JS

Centro de e-Learning SCEU UTN - BA.

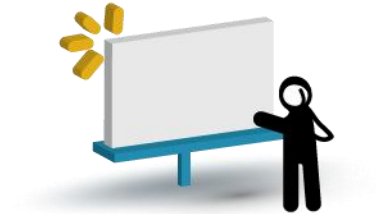
Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning

Módulo I

Nivelación, Javascript y configuración del entorno

Unidad 3: Instalación y Configuración



Presentación:

En esta unidad veremos que es react js, cómo funciona, qué ventajas tiene respecto de otras tecnologías.

Luego comenzaremos a realizar la puesta en marcha de nuestro primer desarrollo, como instalarlo y como configurarlo.



Objetivos:

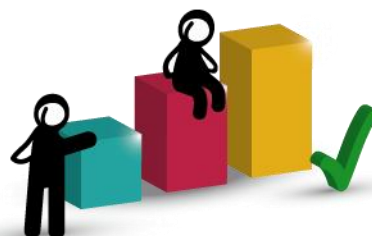
Que los participantes*:

- Comprendan que es react js y que ventajas nos brinda
- Aprendan a instalar y configurar react js
- Realicen una primer aplicación en react js



Bloques temáticos*:

- React JS
- Virtual DOM
- Ventajas de utilizar React JS
- Node JS
- Crear una aplicación utilizando el CLI



Consignas para el aprendizaje colaborativo

En esta Unidad los participantes se encontrarán con diferentes tipos de actividades que, en el marco de los fundamentos del MEC*, los referenciarán a tres comunidades de aprendizaje, que pondremos en funcionamiento en esta instancia de formación, a los efectos de aprovecharlas pedagógicamente:

- Los foros proactivos asociados a cada una de las unidades.
- La Web 2.0.
- Los contextos de desempeño de los participantes.

Es importante que todos los participantes realicen algunas de las actividades sugeridas y compartan en los foros los resultados obtenidos.

Además, también se propondrán reflexiones, notas especiales y vinculaciones a bibliografía y sitios web.

El carácter constructivista y colaborativo del MEC nos exige que todas las actividades realizadas por los participantes sean compartidas en los foros.



Tomen nota*

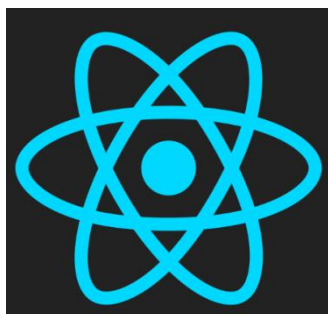
Las actividades son opcionales y pueden realizarse en forma individual, pero siempre es deseable que se las realice en equipo, con la finalidad de estimular y favorecer el trabajo colaborativo y el aprendizaje entre pares. Tenga en cuenta que, si bien las actividades son opcionales, su realización es de vital importancia para el logro de los objetivos de aprendizaje de esta instancia de formación. Si su tiempo no le permite realizar todas las actividades, por lo menos realice alguna, es fundamental que lo haga. Si cada uno de los participantes realiza alguna, el foro, que es una instancia clave en este tipo de cursos, tendrá una actividad muy enriquecedora.

Asimismo, también tengan en cuenta cuando trabajen en la Web, que en ella hay de todo, cosas excelentes, muy buenas, buenas, regulares, malas y muy malas. Por eso, es necesario aplicar filtros críticos para que las investigaciones y búsquedas se encaminen a la excelencia. Si tienen dudas con alguno de los datos recolectados, no dejen de consultar al profesor-tutor. También aprovechen en el foro proactivo las opiniones de sus compañeros de curso y colegas.



React JS

¿Qué es?



ReactJS es una librería Javascript de código abierto, desarrollada por facebook. Esta librería es utilizada en algunas de sus plataformas, por ejemplo Instagram. Esta librería, a pesar de tener detalles algo polémicos, ofrece grandes beneficios en performance, modularidad y promueve un flujo muy claro de datos y eventos, facilitando la planeación y desarrollo de apps complejas.

ReactJS es una librería enfocada en la visualización. Si estamos iniciando un proyecto podemos basarnos en la arquitectura Flux, pero si ya tenemos un proyecto usando un Framework MVC como AngularJS podemos dejar AngularJS como Controlador y que ReactJS se encargue de las vistas. Esto tiene sentido pues ReactJS tiene un performance superior al momento para manipular el DOM, y esto tiene un gran impacto cuando se trata con listas largas que cambian constantemente en nuestra visualización.

¿Cómo Funciona?

React.js está construido en torno a hacer funciones, que toman las actualizaciones de estado de la página y que se traduzcan en una representación virtual de la página resultante. Siempre que React es informado de un cambio de estado, vuelve a ejecutar esas funciones para determinar una nueva representación virtual de la página, a continuación, se traduce automáticamente ese resultado en los cambios del DOM necesarios para reflejar la nueva presentación de la página.

A primera vista, esto suena como que fuera más lento que el enfoque JavaScript habitual de actualización de cada elemento, según sea necesario. Detrás de escena, sin embargo, React.js hace justamente eso: tiene un algoritmo muy eficiente para determinar las diferencias entre la representación virtual de la página actual y la nueva. A partir de esas diferencias, hace el conjunto mínimo de cambios necesarios en el DOM.

Pues utiliza un concepto llamado el DOM virtual que hace selectivamente sub-árboles de los nodos sobre la base de cambios de estado, desarrollando esto, con la menor cantidad de manipulación DOM posible, con el fin de mantener los componentes actualizados, estructurando sus datos.



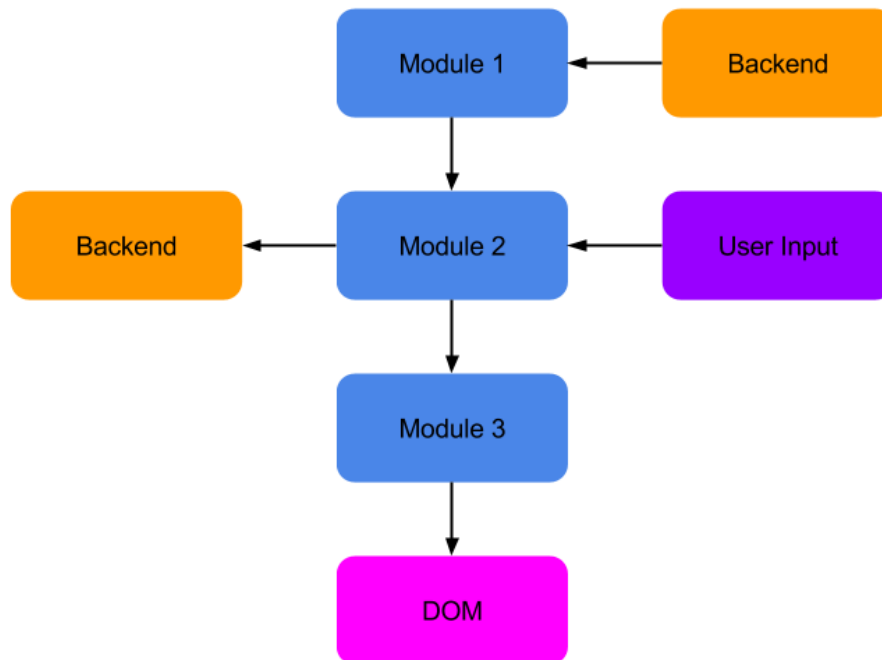
Características

- **JSX:** Es una extensión sintáctica de javascript, si bien su uso no es obligatorio con react js es recomendable
- **Components:** En react js debemos pensar todo como un componente, de esta manera podremos lograr tener aplicaciones escalables y de fácil mantenimiento.
- **Flujo de datos unidireccional:** A diferencia de los frameworks modernos, el flujo de datos en react js es unidireccional. Esto lo hace más fácil para debugear y mantener
- **Licencia:** Esta licenciado bajo "the Facebook Inc. Documentation is licensed under CC BY 4.0."

Virtual DOM

La clave de react js

El secreto de ReactJS para tener un performance muy alto, es que implementa algo llamado Virtual DOM y en vez de renderizar todo el DOM en cada cambio, que es lo que normalmente se hace, este hace los cambios en una copia en memoria y después usa un algoritmo para comparar las propiedades de la copia en memoria con las de la versión del DOM y así aplicar cambios exclusivamente en las partes que varían. Esto puede sonar como mucho trabajo, pero en la práctica es mucho más eficiente que el método tradicional pues si tenemos una lista de dos mil elementos en la interfaz y ocurren diez cambios, es más eficiente aplicar diez cambios, ubicar los componentes que tuvieron un cambio en sus propiedades y renderizar estos diez elementos, que aplicar diez cambios y renderizar dos mil elementos. Son más pasos a planear y programar, pero ofrece una mejor experiencia de usuario y una planeación muy lineal. Una característica importante de ReactJS es que promueve el flujo de datos en un solo sentido, en lugar del flujo bidireccional típico en Frameworks modernos, esto hace más fácil la planeación y detección de errores en aplicaciones complejas, en las que el flujo de información puede llegar a ser muy complejo, dando lugar a errores difíciles de ubicar.



¿Cómo funciona el virtual DOM?

Imagina que tienes un objeto que es un modelo en torno a una persona. Tienes todas las propiedades relevantes de una persona que podría tener, y refleja el estado actual de la persona. Esto es básicamente lo que React hace con el DOM.

Ahora piensa, si tomamos ese objeto y le hacemos algunos cambios. Se ha añadido un bigote, unos bíceps y otros cambios. En React, cuando aplicamos estos cambios, dos cosas ocurren:

- En primer lugar, React ejecuta un algoritmo de “diffing”, que identifica lo que ha cambiado.
- El segundo paso es la reconciliación, donde se actualiza el DOM con los resultados de diff.

Lo que hace React, ante estos cambios, en lugar de tomar a la persona real y reconstruirla desde cero, sólo cambiaría la cara y los brazos. Esto significa que si usted tenía el texto en una entrada y una actualización se llevó a cabo, siempre y cuando el nodo padre de la entrada no estaba programado para la actualización, el texto se quedaría sin ser cambiado.



Ventajas de utilizar React JS

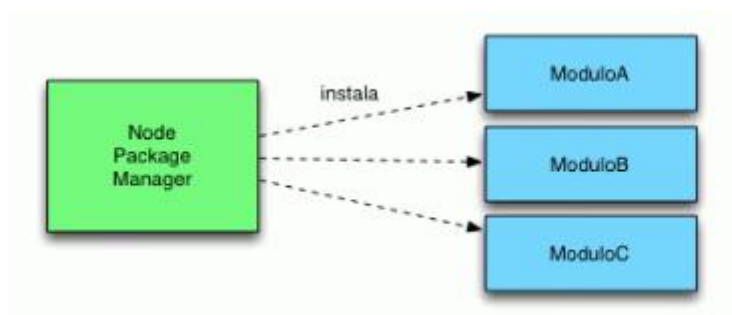
- Performance con DOM Virtual
- Flujo unidireccional de datos y eventos a través de una jerarquía de componentes modulares con punto único de entrada (programación reactiva)
- Binding unidireccional entre el view (vista) y modelo, se evita complejidad y explosión de eventos difíciles para debugging.
- Se usa realmente por sus creadores
- Se integra directamente con implementaciones del paradigma también reactivo de gestión de datos Flux (como Redux) que maneja los datos de la aplicación en una división también jerárquica de componentes contenedores vinculados al estado de los datos en el store, y que pasan estos datos y funciones gestores de esos datos a componentes puros que demuestran los datos en forma predecible sin efectos secundarios y ofrecen puntos de interacción al usuario.
- Es fácil incluir rendering (proyección del componente en el DOM virtual como nodos de elementos con estilo y comportamiento, o sea HTML + CSS + JavaScript) en el lado del servidor con funciones sencillas, para lograr las llamadas aplicaciones universales (antes se llamaban isomórficas).



Node JS

¿Qué es NPM?

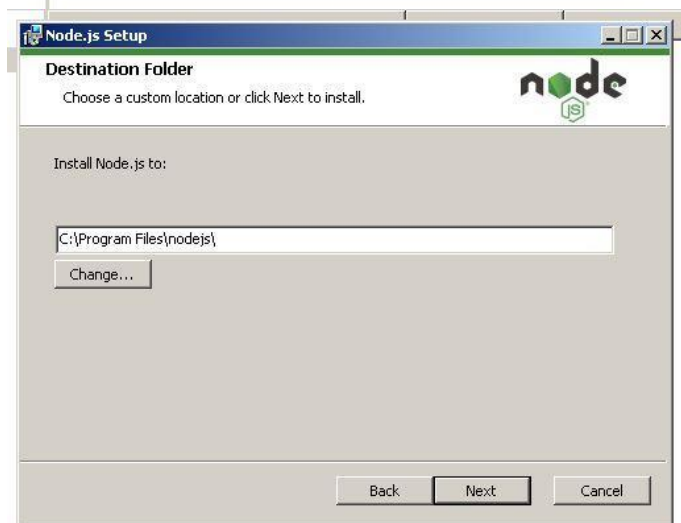
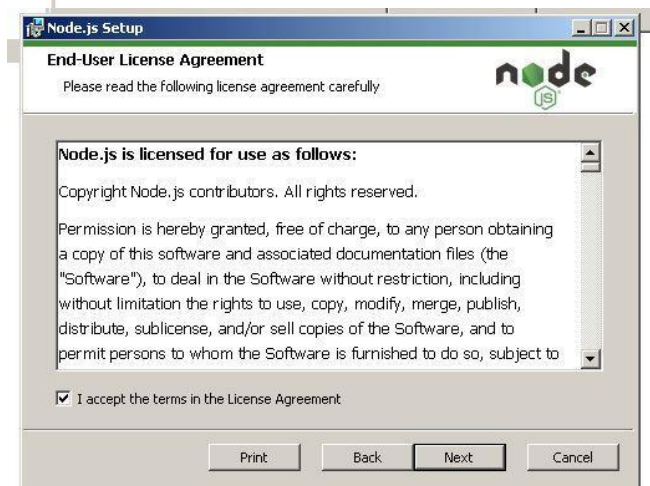
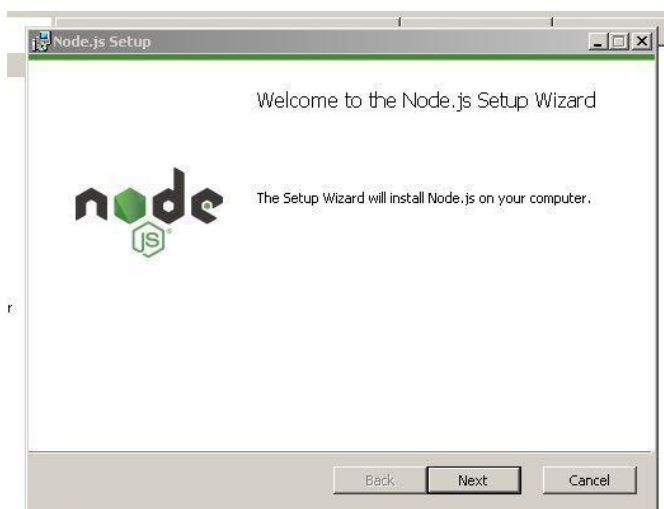
Cuando usamos Node.js rápidamente tenemos que instalar módulos nuevos (librerías) ya que Node al ser un sistema fuertemente modular viene prácticamente vacío. Así que para la mayoría de las operaciones deberemos instalar módulos adicionales. Esta operación se realiza de forma muy sencilla con la herramienta npm (Node Package Manager).

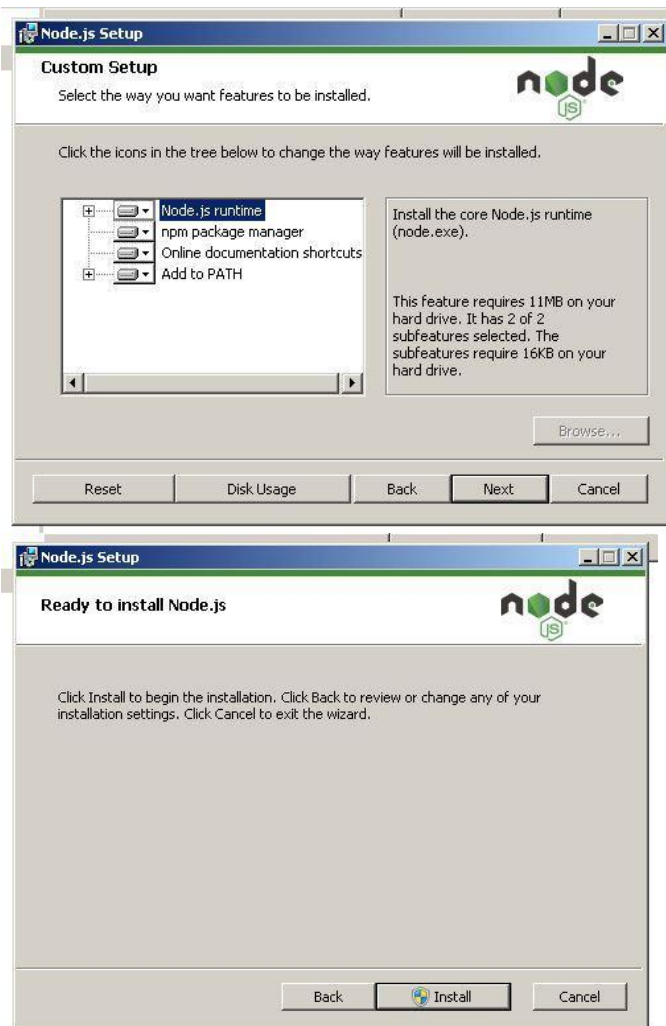


Instalación

Para poder instalar una aplicación de react js desde el CLI, debemos previamente instalar node js.

1. Ingresar a: <https://nodejs.org/en/>
2. Descargar la última versión de node Js
3. Ejecutar el archivo descargado y seguir los siguientes pasos:



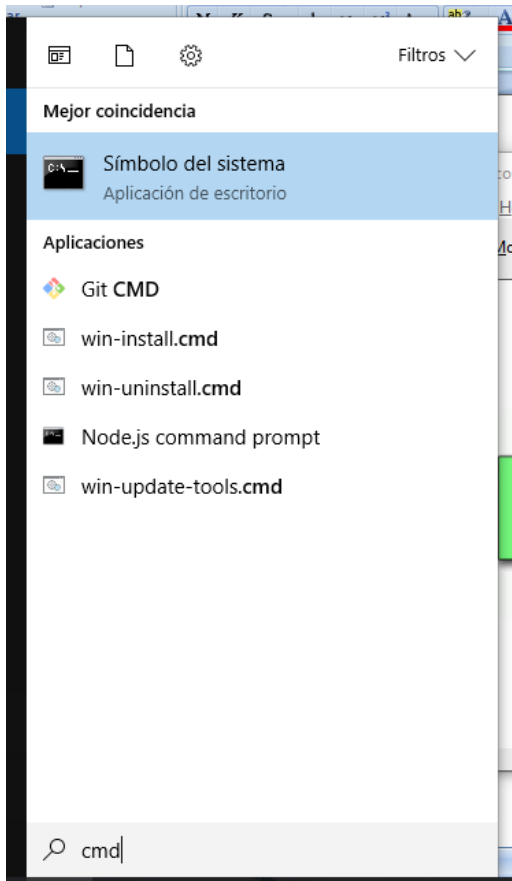




Crear una aplicación utilizando el CLI

Trabajar con la consola de windows

Abrir la consola





Ubicarnos en un directorio específico

Con el comando **cd** podemos ingresar al directorio sobre el cual vamos a crear nuestra aplicación en react. Con **cd..** Volvemos al directorio anterior.

```
G:\sites>cd react
```

Luego estamos dentro del directorio react

```
G:\sites\react>
```

Instalar react js

Debemos ejecutar el comando **npm install -g create-react-app**

```
G:\sites\react>npm install -g create-react-app
```




Crear una aplicación

Para crear nuestra aplicación debemos ejecutar **create-react-app my-app**

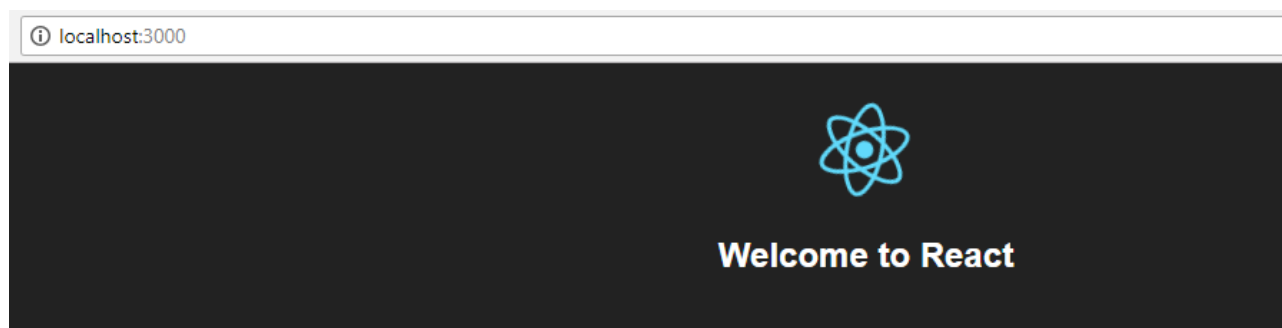
```
G:\sites\react>create-react-app my-app  
Creating a new React app in G:\sites\react\my-app.
```

Ejecutar aplicación en el navegador

Para ejecutar una aplicación y poder acceder desde el navegador debemos ejecutar (dentro del directorio de la aplicación creada) **npm start**

```
G:\sites\react\utn>npm start  
  
> utn@0.1.0 start G:\sites\react\utn  
> react-scripts start  
Starting the development server...  
Compiled successfully!  
  
You can now view utn in the browser.  
  
Local:            http://localhost:3000/  
On Your Network:  http://192.168.0.11:3000/  
  
Note that the development build is not optimized.  
To create a production build, use npm run build.
```

El resultado obtenido será el siguiente:



To get started, edit `src/App.js` and save to reload.

Centro de e-Learning SCEU UTN - BA.

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning



Bibliografía utilizada y sugerida

Fedosejev, A. (2015). React.js Essentials (1 ed.). EEUU, Packt.

Amler, . (2016). ReactJS by Example (1 ed.). EEUU, Packt.

Stein, J. (2016). ReactJS Cookbook (1 ed.). EEUU, Packt.

<https://reactjs.org/tutorial/tutorial.html>

<https://reactjs.org/docs/hello-world.html>

<https://carlosvillu.com/introduccion-a-reactjs/>

<https://platzi.com/blog/react-js-de-javascript/>

<http://code.ezakto.com/react/introduccion-a-react.html>



Lo que vimos:

En esta unidad aprendimos que es react js, como funciona, que es el virtual dom de react y como instalar nuestra primer aplicación desde el CLI

Lo que viene:

En la próxima unidad aprendemos más sobre react js, que es jsx, como es su sintaxis y cómo funciona el webpack.

