


Capítulo 1: Java Script

Capítulo 2: React JS 16




División de Alta Tecnología

2

React JS 16

React JS Front-End Application Developer

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.




Objetivos

Al finalizar el capítulo, el alumno logrará:

- Entender en funcionamiento de la librería React.
- Crear componentes con JSX.
- Integrar React con otras librerías.
- Crear aplicaciones con el stack React y Redux.
- Conocer la nueva característica de React, los hooks.

2 - 2

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Agenda

- Introducción a React
- Configuración de entorno
- Introducción a JSX
- Componentes y propiedades
- Manejo de estado y ciclo de vida del componente
- Comunicación entre componentes
- Peticiones HTTP
- Manejo de Rutas
- Manejo de Formularios
- Redux
- Hooks

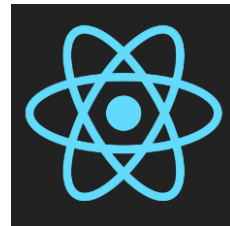
2 - 3

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Introducción a React

- React es una librería open-source de Javascript creada por los desarrolladores de Facebook para construir interfaces de usuario. Su principal característica es su capacidad para hacer cambios en el DOM de la página de una forma muy eficiente.
- React está basada en Componentes, facilitando su reutilización.
- Facebook la utiliza en producción para su red social en determinadas partes, como los comentarios y también en Instagram.



2 - 4

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Introducción a React

- No se trata de un framework JS de cliente como pueden ser Angular, Backbone o Ember, si no de una librería que se encarga del renderizado de las vistas de una aplicación web.
- En el modelo MVC (Model-View-Controller), es sólo la V.
- Puede ser combinado con cualquier otra tecnología rápidamente, puesto que no asume ninguna instalación o estructura previa.
- Es fácil comenzar un proyecto nuevo con React, así como integrarlo a uno existente.
- React 15, 16 soporta todos los navegadores populares, incluyendo Internet Explorer 9 y mayores.

2 - 5

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Introducción a React

- Lo que ha hecho que React se vuelva trending es su velocidad de renderizado de vistas. Eso es posible gracias a un **Virtual DOM**.
- El **Virtual DOM** crea una copia del DOM en memoria, luego utiliza un algoritmo para comparar (diff) las propiedades y actualizar solamente las partes que cambiaron y no tener que revisar y repintar el DOM entero de nuestra página. **React Fiber** es la nueva mejora a este algoritmo.
- Como sabemos, dónde más tiempo se pierde en una aplicación web es en el renderizado y pintado del DOM.

2 - 6

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Configuración de entorno

- Hay dos formas para trabajar con React
 - Usando cdns.

```
<script src="https://unpkg.com/babel-standalone@6.26.0/babel.min.js"></script>
<script crossorigin src="https://unpkg.com/react@16/umd/react.development.js"></script>
<script crossorigin src="https://unpkg.com/react-dom@16/umd/react-dom.development.js"></script>
```

- Creando un proyecto React usando el CLI **create-react-app**

```
npm install -g create-react-app
create-react-app my-app

cd my-app
npm start
```

2 - 7

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Introducción a JSX

- JSX es una extensión de JavaScript inspirada en XML que facilita el repintado de elementos DOM.
- No necesariamente tiene que usar JSX con React, si usted desea puede usar solo javascript, claro está que se recomienda utilizar JSX por que es más legible a la hora de desarrollar.
- Para transpilar JSX puedes usar babel.
- Tiene una sintaxis muy parecida a la de HTML.

```
<div className="chat-box">
  <h1 className="title">Chat con ReactJS</h1>
  <MessageList messageList={messageList} />
  <ChatForm />
</div>
```

2 - 8

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Introducción a JSX

- Condicionales.

```
<div>
  {isLoggedIn && <LogOutButton />}
</div>
```

```
<div>
  {isLoggedIn ? <LogOutButton /> : <LogInButton />}
</div>
```

- Iteraciones.

```
<ul>
  {items.map(item => <li>{item.title}</li>)}
</ul>
```

2 - 9

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Componentes

- Un componente viene a ser cada uno de los elementos de nuestra aplicación, como por ejemplo: el header, footer, botones, entre otros elementos de nuestra UI.
- Un componente React pueden tener un método render, como también opcionalmente otros métodos del ciclo de vida.
- Los componentes son como funciones, reciben parámetros llamados “props” y retornan un elemento React describiendo lo que va a mostrar en la pantalla.

```
function Welcome(props) {
  return <h1>Hello, {props.name}</h1>;
}
```

2 - 10

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Componentes

- También se puede hacer uso de ES6 class para definir un componente.

```
class Welcome extends React.Component {  
  render() {  
    return <h1>Hello, {this.props.name}</h1>;  
  }  
}
```

- Tanto la creación de componentes usando funciones o clases son equivalentes.

2 - 11

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Componentes

- Componente sin JSX.

```
class Hello extends React.Component {  
  render() {  
    return React.createElement('div', null, 'Hello ${this.props.toWhat}');  
  }  
}  
  
ReactDOM.render(  
  React.createElement(Hello, {toWhat: 'World'}, null),  
  document.getElementById('root')  
);
```

- Componente con JSX.

```
class Hello extends React.Component {  
  render() {  
    return <div>Hello {this.props.toWhat}</div>;  
  }  
}  
  
ReactDOM.render(  
  <Hello toWhat="World" />,  
  document.getElementById('root')  
);
```

2 - 12

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Manejando el estado y las propiedades

- Props:
 - Haciendo analogía con el HTML tradicional, las propiedades son los atributos de las etiquetas.
 - Son datos inmutables que pasan de padres a hijos hasta el final de la cadena.
 - Un componente no puede cambiar sus props, pero es responsable de pasarlo de forma correcta a sus hijos.
 - Se pueden definir el tipo de dato para cada prop y a la vez asignarle un valor por defecto usando la librería **prop-types**.
 - Desde el componente se accede a los props usando **"this.props"** en clases o **"props"** en funciones.
 - Todos los componentes tienen props.

2 - 13

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.

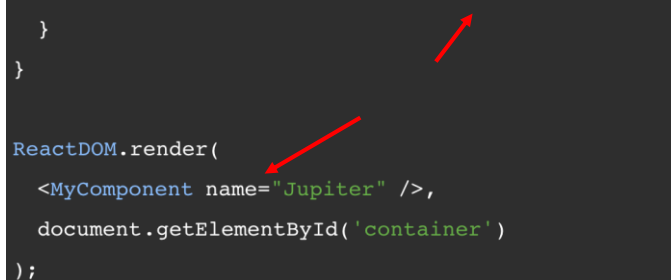


Manejando el estado y las propiedades

- Props.

```
class MyComponent extends React.Component{
  render(){
    return <div>MyComponent {this.props.name}</div>
  }
}

ReactDOM.render(
  <MyComponent name="Jupiter" />,
  document.getElementById('container')
);
```



2 - 14

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Manejando el estado y las propiedades

- Props.

```
function Welcome(props) {
  return <h1>Hello, {props.name}</h1>;
}

function App() {
  return (
    <div>
      <Welcome name="Sara" />
      <Welcome name="Cahal" />
      <Welcome name="Edite" />
    </div>
  );
}

ReactDOM.render(
  <App />,
  document.getElementById('root')
);
```

2 - 15

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Manejando el estado y las propiedades

- State:

- Los componentes no solo tienen data estática, pueden mantener un estado interno y mutable que influye en el comportamiento y aspecto del componente.
- Este estado puede ser cambiado en cualquier momento, interna o externamente, y el componente se adaptará instantáneamente a su nuevo estado.
- Es opcional si deseas que un componente tenga o no un state.
- Su uso es indispensable para hacer tu aplicación interactiva.
- Se les llama Stateful Components a los componentes que tienen un state.

2 - 16

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Manejando el estado y las propiedades

- State.

```
class MyComponent extends React.Component{
  constructor(props){
    super(props);
    this.state = {name: 'Ganimedes'};
  }
  render(){
    return <div>My Component {this.state.name}</div>;
  }
};

ReactDOM.render(
  <MyComponent />,
  document.getElementById('container')
);
```

2 - 17

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Manejando el estado y las propiedades

- Se les llama Stateful Components a los componentes que tienen un state. Se usa clases para ello.
- Los Stateless Components simplemente se encargan de renderizar una pequeña parte del código y pasar las props a sus hijos. Se usa funciones para ello.

2 - 18

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ejercicio Nº 2.1: Crear una aplicación usando create-react-app

Crear una aplicación usando create-react-app.

Al finalizar el laboratorio, el alumno logrará:

- Crear una aplicación react usando el CLI create-react-app.

2 - 19

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ejercicio Nº 2.2: Creación de componentes usando JSX

Creación de componentes usando JSX.

Al finalizar el laboratorio, el alumno logrará:

- Crear componentes usando JSX.

2 - 20

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ejercicio N° 2.3: Manejar la información entre componentes

Manejar la información entre componentes y uso de formulario.

Al finalizar el laboratorio, el alumno logrará:

- Comunicar información entre componentes.

2 - 21

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Rutas con React router

- React Router está en su versión 4 y a diferencia de sus versiones anteriores, las rutas funcionan como componentes reales de React, lo cual es un enorme logro ya que simplifica mucho el manejo de rutas en nuestras aplicaciones React.
- React Router sirve para cambiar de rutas dentro de tu aplicación sin recargar el navegador.
- Para instalar hacer lo siguiente:

```
npm install --save react-router-dom
```

2 - 22

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Rutas con React router

- Route

```
<Route path="/" exact component={Home} />
<Route component={NotFoundPage} />
```

- HashRouter y Switch

```
<HashRouter>
  <Switch>
    <Route path="/about-us" component={AboutUsContainer} />
    <Route path="/terms" component={TermsAndConditionsContainer} />
  </Switch>
</HashRouter>
```

2 - 23

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ciclo de vida de los componentes

```
class Clock extends React.Component {
  constructor(props) {
    super(props);
    this.state = {date: new Date()};
  }

  componentDidMount() {
  }

  componentWillUnmount() {
  }

  render() {
    return (
      <div>
        <h1>Hello, world!</h1>
        <h2>It is {this.state.date.toLocaleTimeString()}</h2>
      </div>
    );
  }
}
```

2 - 24

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ejercicio Nº 2.4: Navegación entre vistas

Usar react router para crear rutas.

Al finalizar el laboratorio, el alumno logrará:

- Agregar rutas a una aplicación react.

2 - 25

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Consumir APIs usando axios

- Axios es un cliente HTTP basado en Promesas para Javascript, el cual puede ser utilizado tanto en tu aplicación Front-end, como en el Back-end con NodeJS. Utilizando Axios, es muy sencillo enviar peticiones a *endpoints* REST y realizar operaciones CRUD.
- Instalación:

```
npm install axios
```

2 - 26

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ejercicio Nº 2.5: Consumir datos desde servicios externos

Hacer consultas HTTP para obtener y modificar datos externos

Al finalizar el laboratorio, el alumno logrará:

- Comunicar una aplicación react con una API REST.

2 - 27

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Formularios

- La etiqueta `<form>` tiene una propiedad llamada **onSubmit** a la cual se le debe pasar una función que maneje la data que se envía en el formulario.
- Los elementos de formulario como `<input>`, `<textarea>` y `<select>` deben tener su propio estado y ello se actualiza según cada cambio que se realice sobre dichos elementos.
- Estos elementos de formulario tienen una propiedad llamada **onChange** a la cual se le debe pasar una función que actualize el estado del elemento.

2 - 28

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Formularios

```
class NameForm extends React.Component {
  constructor(props) {
    super(props);
    this.state = { name: '' };
  }
  handleNameChange = (event) => {
    this.setState({name: event.target.value});
  }
  handleSubmit = (event) => {
    event.preventDefault();
    // Validate this.state.name
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <label>
          Name:
          <input type="text" value={this.state.name} onChange={this.handleNameChange} />
        </label>
        <input type="submit" value="Submit" />
      </form>
    );
  }
}
```

2 - 29

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ejercicio Nº 2.6: Crear Formularios y manejo de validaciones

Crear un formulario y manejar las validaciones.

Al finalizar el laboratorio, el alumno logrará:

- Crear formularios con la librería Formik.
- Manejo de validaciones con la librería Yup.

2 - 30

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Avanzada manera de gestionar el estado usando Redux

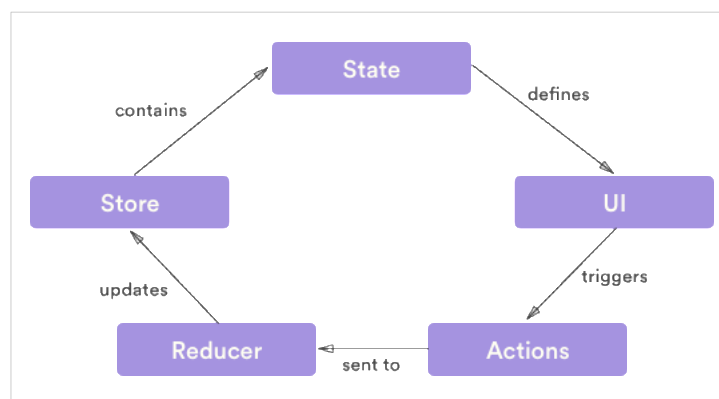
- Redux es una librería para controlar el estado de nuestras aplicaciones web fácilmente, de una forma consistente entre cliente y servidor, testeable y con una gran experiencia de desarrollo.
- Redux está en gran parte influenciado por la arquitectura Flux propuesta por Facebook para las aplicaciones de ReactJS.
- Instalación:

2 - 31

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Avanzada manera de gestionar el estado usando Redux



2 - 32

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Ejercicio Nº 2.6: Utilizar Redux para manejar la información entre componentes

Al finalizar el laboratorio, el alumno logrará:

- Usar redux para ordenar y hacer escalable una aplicación react.

2 - 33

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Lecturas adicionales

Para obtener información adicional, puede consultar:

- React JS
 - <https://reactjs.org/>
 - <https://reacttraining.com/react-router/web/guides/quick-start>
 - <https://jaredpalmer.com/formik>
 - <https://github.com/axios/axios>
 - <https://redux.js.org/>
 - <https://facebook.github.io/jest/docs/en/getting-started.html>
 - <https://bulma.io/>

2 - 34

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Resumen

En este capítulo, usted aprendió:

- El uso de la librería React para la capa de presentación de sus proyectos web. Agregándole un manejador de rutas, de estados a nivel de aplicación con Redux, formularios, consumir servicios externos..
- A desarrollar un administrador de productos.

2 - 35

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.



Tarea Nº 2: Aplicar los conceptos sobre React

Implementar un mantenimiento de un carrito de compras.

Al finalizar el laboratorio, el alumno logrará:

- Crear un proyecto react integrado con redux.
- Integrar un proyecto react con el backend mediante una API.

2 - 36

Copyright © Todos los Derechos Reservados – IES Cibertec SAC.

