

Tipo : Guía de Enunciado
Capítulo : Single Page Application (SPA) con React
Duración : 60 minutos

I. OBJETIVO

Configuración de proyecto web con ReactJs.

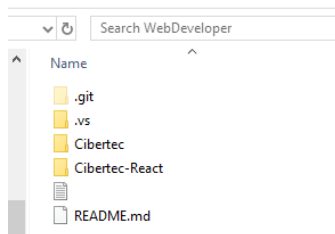
II. REQUISITOS

Los siguientes elementos de software son necesarios para la realización del laboratorio:

- Windows 10 (como mínimo Windows 8)
- NodeJs
- Visual Studio Code

III. EJECUCIÓN DEL LABORATORIO

1. En la raíz del repositorio crear la carpeta “**Cibertec-React**” (Ver Imagen)



2. En la carpeta recién creada copia los archivos de la carpeta “**07 Fuentes**”, que son los aquí listados:

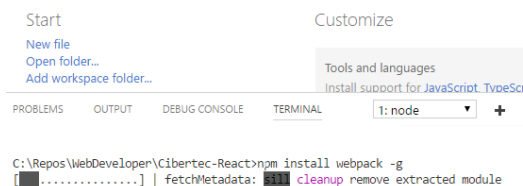
- package.json
- tsconfig.json
- webpack.config.js

3. Abrir Visual Studio Code y abrir la carpeta “**Cibertec-React**”

4. Abrir el terminal integrado presionando “**Ctrl + ñ**”:

5. En el terminal digitar el siguiente comando:

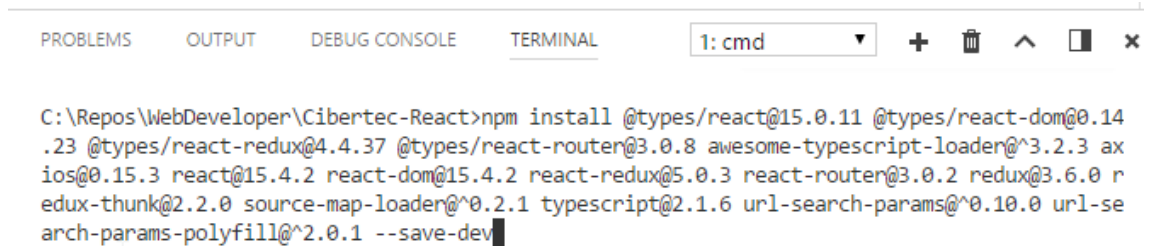
- npm install webpack -g



Dado que vamos a trabajar con transpilación de código Typescript, en el encargado de hacer ese proceso será webpack y por tal motivo es necesario instalarlo de manera global (-g) en nuestro ordenador.

6. Ejecutar la siguiente línea de comandos:

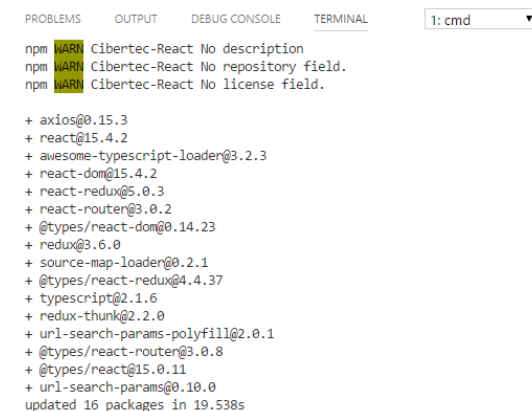
```
npm install @types/react@15.0.11 @types/react-dom@0.14.23 @types/react-  
redux@4.4.37 @types/react-router@3.0.8 awesome-typescript-loader@^3.2.3 ax  
ios@0.15.3 react@15.4.2 react-dom@15.4.2 react-redux@5.0.3 react-router@3.0.2  
redux@3.6.0 redux-thunk@2.2.0 source-map-loader@^0.2.1 typescript@2.1.6 url-  
search-params@^0.10.0 url-search-params-polyfill@^2.0.1 --save-dev
```



The screenshot shows a terminal window with the following command and output:

```
C:\Repos\WebDeveloper\Cibertec-React>npm install @types/react@15.0.11 @types/react-dom@0.14.23 @types/react-redux@4.4.37 @types/react-router@3.0.8 awesome-typescript-loader@^3.2.3 axios@0.15.3 react@15.4.2 react-dom@15.4.2 react-redux@5.0.3 react-router@3.0.2 redux@3.6.0 redux-thunk@2.2.0 source-map-loader@^0.2.1 typescript@2.1.6 url-search-params@^0.10.0 url-search-params-polyfill@^2.0.1 --save-dev
```

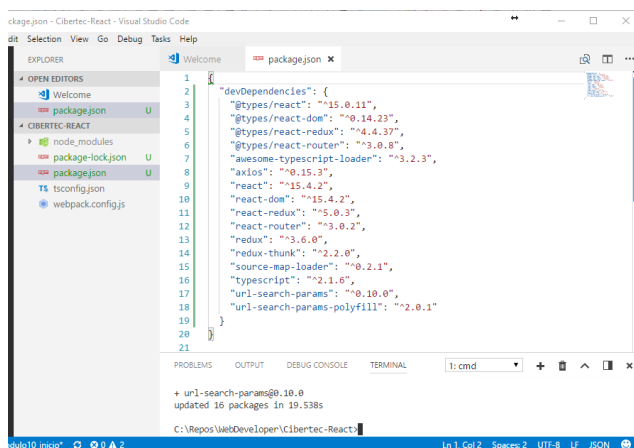
Finalizada la instalacion visualizaremos:



The screenshot shows a terminal window with the following output:

```
npm WARN Cibertec-React No description  
npm WARN Cibertec-React No repository field.  
npm WARN Cibertec-React No license field.  
  
+ axios@0.15.3  
+ react@15.4.2  
+ awesome-typescript-loader@3.2.3  
+ react-dom@15.4.2  
+ react-redux@5.0.3  
+ react-router@3.0.2  
+ @types/react-dom@0.14.23  
+ redux@3.6.0  
+ source-map-loader@0.2.1  
+ @types/react-redux@4.4.37  
+ typescript@2.1.6  
+ redux-thunk@2.2.0  
+ url-search-params-polyfill@2.0.1  
+ @types/react-router@3.0.8  
+ @types/react@15.0.11  
+ url-search-params@0.10.0  
updated 16 packages in 19.538s
```

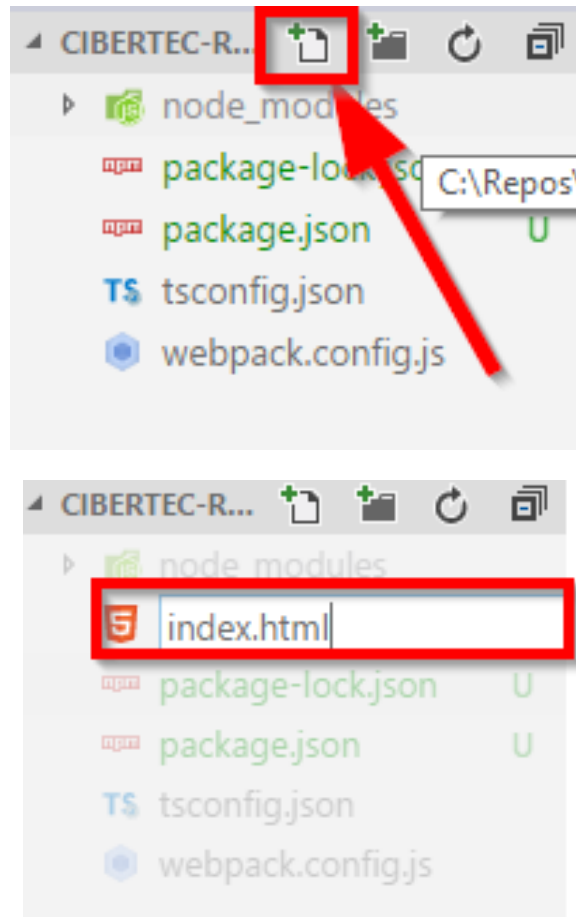
También debemos de validar el fichero “package.json”



The screenshot shows a Visual Studio Code editor with the following content in the package.json file:

```
{  
  "devDependencies": {  
    "@types/react": "^15.0.11",  
    "@types/react-dom": "^0.14.23",  
    "@types/react-redux": "^4.4.37",  
    "@types/react-router": "^3.0.8",  
    "awesome-typescript-loader": "^3.2.3",  
    "axios": "^0.15.3",  
    "react": "^15.4.2",  
    "react-dom": "^15.4.2",  
    "react-redux": "^5.0.3",  
    "react-router": "^3.0.2",  
    "redux": "^3.6.0",  
    "redux-thunk": "^2.2.0",  
    "source-map-loader": "^0.2.1",  
    "typescript": "^2.1.6",  
    "url-search-params": "^0.10.0",  
    "url-search-params-polyfill": "^2.0.1"  
  }  
}
```

7. Ahora que ya tenemos todas nuestras librerías instaladas procedemos a crear el html inicial que será llamado "index.html":



Agregamos el siguiente código:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <title> Cibertec React</title>
    <link rel="stylesheet"

href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap
.min.css"
    integrity="sha384-
BVYiiSIFeK1dGmJRAkycuHAHRg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
    crossorigin="anonymous">
    <link rel="stylesheet" href="assets/styles.css">
  </head>
  <body>
    <div id="example"></div>
    <script src="./node_modules/react/dist/react.js"></script>
```

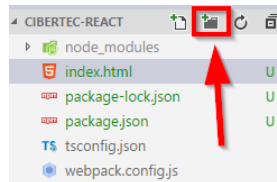
```

        <script src="./node_modules/react-dom/dist/react-dom.js"></script>
        <script src="./dist/bundle.js"></script>
    </body>
</html>

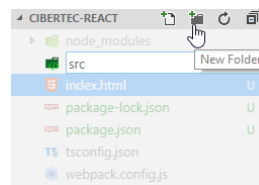
```

8. Creamos una nueva carpeta con el nombre **"src"**.

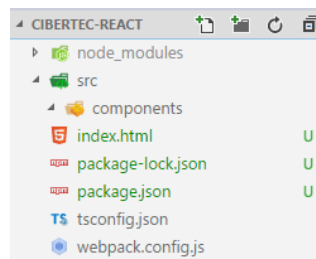
(1) Hacer clic en el icono de nueva carpeta



(2) Ingresar el nombre de la carpeta



9. En la carpeta **"src"** creamos una nueva carpeta con el nombre **components**.



10. En la carpeta **"components"** creamos el fichero **"Hello.tsx"**, con el siguiente código:

```

import * as React from "react";

export interface HelloProps {
    compiler: string;
    framework: string;
}

export class Hello extends React.Component<HelloProps, any> {
    render() {
        return <h1>Hello from {this.props.compiler} and
{this.props.framework}!</h1>
    }
}

```

11. En la carpeta **"src"** creamos el archivo **"index.tsx"** con el siguiente código:

```

import * as React from "react";

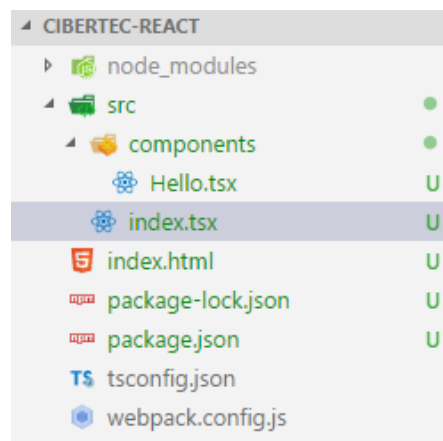
```

```
import * as ReactDOM from "react-dom";
import 'url-search-params-polyfill';

import { Hello } from "../components/Hello";

ReactDOM.render(
  <Hello compiler="Webpack" framework="React" />,
  document.getElementById("example")
);
```

Al finalizar la estructura de nuestro proyecto será:



12. Desde el terminal integrado (Ctrl + ñ) ejecutamos el comando:

```
> webpack

PROBLEMS  OUTPUT  TERMINAL  ...
1: cmd

C:\Repos\WebDeveloper\Cibertec-React>webpack

[at-loader] Using typescript@2.1.6 from typescript and "tsconfig.json" from C:\Repos\WebDeveloper\Cibertec-React\tsconfig.json.

[at-loader] Checking started in a separate process...

[at-loader] Ok, 0.148 sec.
Hash: 5e2da9a436b6cea058f2
Version: webpack 3.8.1
Time: 1254ms

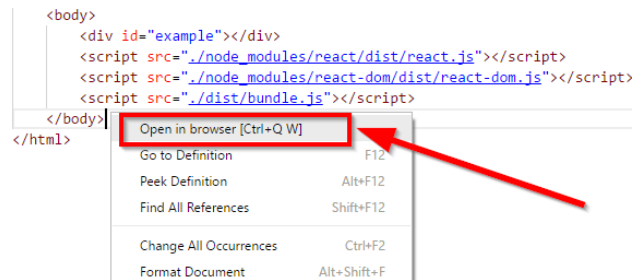
  Asset      Size  Chunks             Chunk Names
  bundle.js  12.8 kB          0  [emitted]  main
  bundle.js.map  15.2 kB          0  [emitted]  main
    [1] ./src/index.tsx 305 bytes {0} [built]
    [4] (webpack)/buildin/global.js 488 bytes {0} [built]
    + 4 hidden modules

C:\Repos\WebDeveloper\Cibertec-React>
```

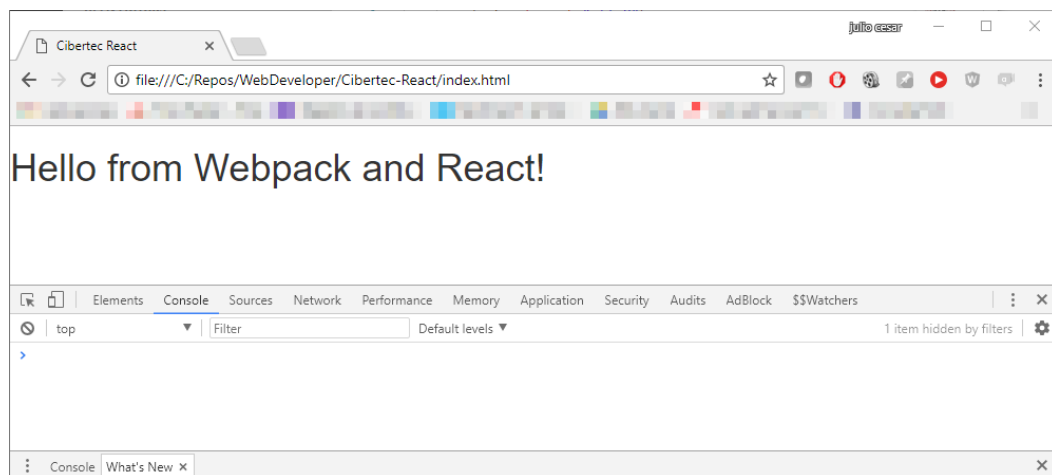
13. Una vez finalizado visualizaremos que en la estructura del proyecto aprecio un nuevo folder llamado “**dist**” con los siguientes ficheros:

- bundle.js
- bundle.js.map

14. Abrimos el archivo “**index.html**” en Visual Studio Code, hacemos clic derecho en el código y seleccionamos “**View in browser**”.



15. El resultado esperado es:



IV. EVALUACIÓN

1. ¿Por qué es importante el uso de Nodejs para el desarrollo frontend?
2. ¿Qué es npm?
3. ¿Qué es webpack?
4. ¿Qué significan para npm el -g y --save-dev?