



everis

an NTT DATA Company

Manual De Instalación Proyecto

Tótems - Porvenir

HISTÓRICO DEL DOCUMENTO

Versión	Fecha	Autor	Empresa
1.0	22/10/2019	Equipo Front	everis

OBJETIVO

Este documento tiene como finalidad explicar la instalación del proyecto en un ambiente Windows, instalando todas las dependencias necesarias para su correcto funcionamiento.

INSTALACIÓN DEL PROYECTO

1. Para descargar, compilar el proyecto **se requiere tener instalado Node.js y Maven:**

Los enlaces para descargar son los siguientes:

Node Js:

La instalación de Node.js en versión estable v 12.16.0 o superior.

<https://nodejs.org/es/download/releases/>

Una vez descargado el instalador seguir la guía de instalación hasta finalizar, una vez terminada la instalación para verificar la versión instalada abra una terminal de comandos (cmd.exe) y ejecute el siguiente comando **node -v** o **node --version**.

Maven:

La instalación de Maven en versión estable v 3.6.3

En primer lugar, es necesario que tengamos instalado el JDK 1.8. en nuestro equipo. Para ello:

Accedemos a la web de Oracle y buscamos la versión que queramos. Seguimos el asistente hasta el final: siguiente, siguiente, siguiente.

Y configuramos las variables de entorno de Windows. Para ello hacemos clic con el botón derecho del ratón sobre Equipo > Propiedades > Configuración avanzada del sistema > pestaña Opciones avanzadas > botón Variables de entorno. Tenemos que añadir las siguientes:

CLASSPATH: C:\Program Files\Java\jdk1.8.0_79\jre

JAVA_HOME: C:\Program Files\Java\jdk1.8.0_79

PATH: ... ;C:\Program Files\Java\jdk1.8.0_79\bin; ... (en este caso, la variable PATH ya existirá, así que añadimos esa cadena a continuación de lo que haya, separada por ;

Si todo está bien, al abrir un terminal (cmd.exe) y escribir «java -version» debería mostrar los datos de la versión instalada.

Accedemos a la **página web de Maven** y descargamos la última versión.

<http://maven.apache.org/download.cgi#Installation>

Descomprimos el archivo en C:/maven, de modo que todos los archivos de Maven se encontrarán en C:/maven/apache-maven-3.6.3

Configuramos las variables de entorno al igual que lo hicimos en el caso anterior. En este caso añadiríamos:

M2: %M2_HOME%\bin

M2_HOME: C:\maven\apache-maven-3.6.3

PATH: ...;C:\maven\apache-maven-3.6.3\bin; ...

Accedemos al terminal y escribimos «mvn -version». Asegúrate de que el terminal no estaba abierto cuando cambiaste las variables de sesión, porque entonces no reconocerá estas variables. Debes cerrarlo y volverlo a abrir. Si todo está bien te mostrará la versión de Maven instalada.

- Después de tener instalados Node y Maven procedemos a la descarga de la carpeta del proyecto, el proyecto se encuentra en la siguiente ruta [\\$/Service Center/Dm totem/Prue.Dm totem sc/Web Totem](#) , como sugerencia el proyecto se debe dejar en c:\proyecto\, después abrimos una terminal de comandos y abrimos la carpeta del proyecto, una vez dentro escribimos los siguientes comandos:

Instalar proyecto y dependencias

npm install

Este paso puede tardar varios minutos dependiendo de la conexión a internet, aquí se instalan todas las dependencias para que el proyecto funcione correctamente.

Compilar

nvm package

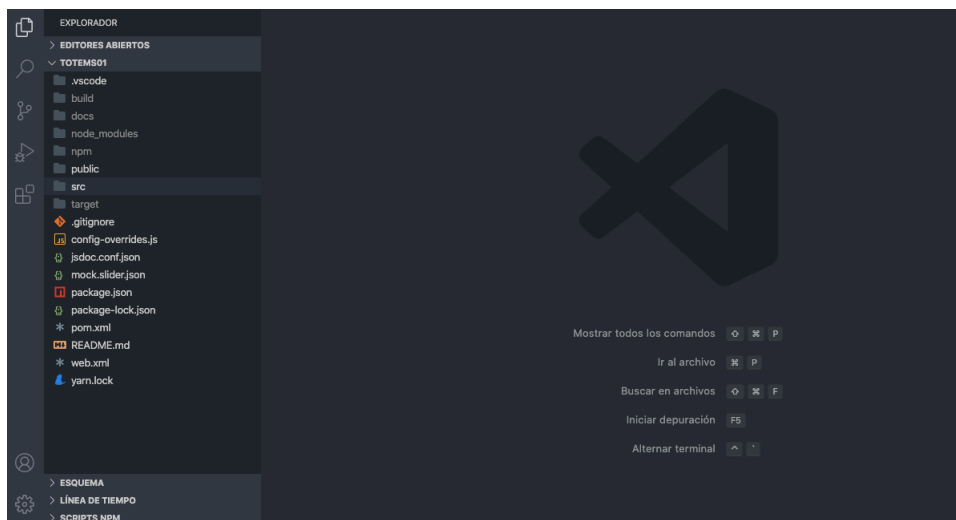
3. A terminar estos pasos se generará una carpeta llamada **target**, donde se encontrará el archivo .war

node_modules	19/02/2020 2:22 p. m.	Carpeta de archivos
npm	19/02/2020 2:07 p. m.	Carpeta de archivos
public	18/02/2020 8:00 a. m.	Carpeta de archivos
src	18/02/2020 8:00 a. m.	Carpeta de archivos
target	19/02/2020 2:25 p. m.	Carpeta de archivos
.DS_Store	17/02/2020 9:46 a. m.	Archivo DS_STORE
.gitignore	17/02/2020 9:46 a. m.	Documento de te...
config-overrides	17/02/2020 9:46 a. m.	Archivo JavaScript
jsdoc.conf	17/02/2020 9:46 a. m.	Archivo de origen ...
...	17/02/2020 9:46 a. m.	...

4. Con este archivo ya se tendrá que hacer la respectiva catalogación.

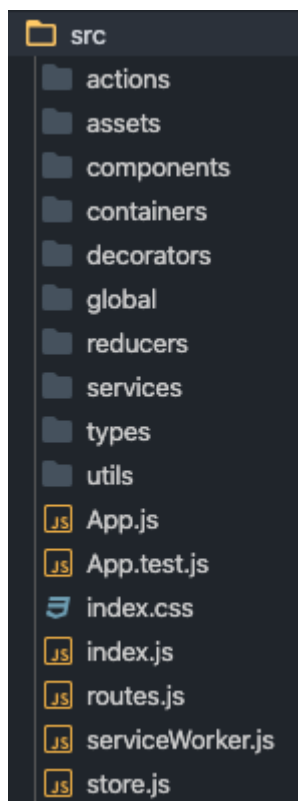
Nombre	Fecha de modificación	Tipo	Tamaño
maven-archiver	19/02/2020 2:25 p. m.	Carpeta de archivos	
Web_Totem	19/02/2020 2:25 p. m.	Carpeta de archivos	
Web_Totem.war	19/02/2020 2:25 p. m.	Archivo WAR	46.321 KB

5. Para editar el código o escribir nuevas funcionalidades, puede elegir el editor de su preferencia (sublime text, VS Code, Atom), cuando tenga el editor abierto arrastre la carpeta del proyecto al editor y suelte, debe ver algo así:



6. La estructura de la aplicación es la siguiente:
- **Build:** esta carpeta el compilado de todo el proyecto.

- **Docs:** esta carpeta se crea cuando se ejecuta el comando para generar la documentación del proyecto, para más información revisar el readme del proyecto.
- **Node_modules:** esta carpeta se crea cuando se ejecuta el comando npm install y aquí dentro están todas las dependencias para que el proyecto funcione sin ningún problema.
- **Public:** en esta carpeta están todos los recursos estáticos de la aplicación (imágenes, audios, videos etc), al igual también está la configuración del service worker que se encarga de manejar el cache de la aplicación.
- **Target:** esta carpeta se genera cuando se ejecuta el comando npm package y allí dentro se crea el war con la aplicación compilada para funcionar en maven.
- **Src:** aquí dentro se encuentra todo el código de la aplicación el cual esta dividido así:



- **Actions:** en esta carpeta están todos los actions que disparan un cambio en el estado global de la aplicación.
- **Assets:** en esta carpeta se encuentran algunas imágenes estáticas y las fuentes que maneja la aplicación.
- **Components:** en esta carpeta están todos los componentes que se reutilizan en varias pantallas de la aplicación, se escriben una sola vez y se utilizan en muchas pantallas.
- **Containers:** en esta carpeta se encuentran todas las páginas de la aplicación por cada pantalla se genera una carpeta que almacena cada una de ellas.

- **Global:** en esta carpeta se encuentran las constantes globales de la aplicación, (variables que se utilizan en toda la aplicación se definen aquí y se llaman en varios sitios en toda la aplicación), los mocks para hacer simulaciones de las respuestas de los servicios y un archivo config con todos los endpoints de la aplicación.
- **Reducers:** en esta carpeta se encuentran los reducers que disparan un action para modificar el estado global de la aplicación.
- **Services:** en esta carpeta esta todo el consumo hacia los endpoints que la aplicación consume.
- **Utils:** archivos con funciones útiles para toda la aplicación.
- **App.js:** es el contenedor principal de la aplicación.
- **Routes.js:** archivo que contiene todas las rutas de la aplicación.
- **Store.js:** archivo con el estado inicial de la aplicación.