MANUAL PROYECTO TYPESCRIPT

Contenido

[**0.** **Instalación de las herramientas necesarias** 2](#_Toc84362438)

[**1.** **Preparación del entorno de trabajo** 5](#_Toc84362439)

[**2.** **Transpilar TS 🡪 JS** 6](#_Toc84362440)

[**3.** **Subida a GitHub** 8](#_Toc84362441)

1. **Instalación de las herramientas necesarias**

Para realizar un proyecto con **TypeScript** es aconsejable trabajar desde un entorno de desarrollo de código, dada las facilidades y ayudas que este brinda a los desarrolladores. En este caso vamos a usar **Visual Studio Code**. Sin embargo, este no es requisito fundamental ya que puede hacerse todos los procesos desde **PowerShell** de Windows, pero es menos intuitivo.

Typescript es un lenguaje de programación fuertemente tipado construido sobre JavaScript, por lo que necesitamos ciertas herramientas para que funcione en nuestra máquina.

El primer software que vamos a necesitar es **Node.js**. Node es un entorno de ejecución para **JavaScript**. Para descargarnos el programa, nos dirigimos a <https://nodejs.org/es/>

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente

Después de elegir la opción que queramos se descargará automáticamente un ejecutable para su instalación. Seguimos los pasos que nos facilita el asistente de instalación hasta el fin de esta.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Con la descarga de **Node.js**, podemos avanzar al siguiente paso: Descarga de paquetes para node.

Para descargar **Typescript**, debemos de dirigirnos a la pagina de ***node package manager***, <https://www.npmjs.com/>

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Si es la primera vez que accedemos a este portal debemos de registrarnos, sin embargo, si ya tenemos cuenta, solo tenemos que identificarnos.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En el buscador ponemos **Typescript** y se nos abrirá una ventana, en la que se explica como instalar el lenguaje en el sistema. La opción -g nos dice que se instalará a nivel global.

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Estas sentencias deben ser ejecutadas en la consola de **PowerShell** de Windows.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Como se puede comprobar, **Node** se descargó correctamente al igual que **TypeScript**.

1. **Preparación del entorno de trabajo**

Para iniciar un proyecto, debemos de crear un directorio donde irá alojado. Dentro de este directorio crearemos las carpetas de **doc** y **src.**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Desde VSC abrimos nuestro directorio y nos dirigimos a la terminal. Desde la ruta donde está alojado nuestro proyecto, ejecutamos los siguientes comandos:

**Npm install 🡪 node\_modules**

**npm init -y 🡪 package.json**

**tsc --init 🡪 tsconfig.json**

Estos comandos nos van a generar una serie de archivos con la configuración por defecto de nuestro proyecto.

Texto

Descripción generada automáticamenteDentro del archivo **tsconfig.json** debemos cambiar el “**target**” a “es6” y en el campo “**outDir**”, pondremos la carpeta en la que se generará el código transpilado, por defecto se elige “**./dist**”

En este momento, tendremos nuestro proyecto con estas carpetas.

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Como no queremos que **./dist** ni **./node\_modules** se suban a GitHub cuando el proyecto este finalizado, crearemos un **.gitignore** en el que escribiremos el nombre de ambas carpetas.

Dentro de la carpeta **src** se suele crear una carpeta **view** que contiene contenido auxiliar que necesitaremos a lo largo del proyecto, separándose del grueso del proyecto.

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

1. **Transpilar TS 🡪 JS**

Una vez que tengamos nuestro código en Typescript listo, nos dirigiremos a la terminal e introduciremos el comando **tsc -w**

Texto

Descripción generada automáticamente

Una vez realizado este proceso, comprobaremos que la carpeta **dist** tiene los mismos archivos que la carpeta **src**, con la diferencia de que estarán en .js

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Texto

Descripción generada automáticamentePara evitar que algún archivo de la carpeta node\_modules pueda interferir, en el archivo **tsconfig.json** podemos introducir lo siguiente al final de este.

1. **Subida a GitHub**

Creamos un repositorio para alojar nuestro nuevo proyecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Introducimos los siguientes comandos, y nuestro repositorio quedará alojado en GitHub posibilitando al resto de la comunidad a acceder a ellos.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente