

## CONFIGURACIÓN PARA DESARROLLO LOCAL

*Instala la semilla para Inicio rápido(QuickStart seed) para un rápido y mas eficiente desarrollo en tu computadora.*

## CONFIGURANDO EL ENTORNO DE DESARROLLO LOCAL

El ejemplo QuickStart live-coding es un playground de angular. No es donde deberías desarrollar una aplicación real. Tu deberías desarrollar localmente en tu propia computadora... y así también es como pensamos deberías aprender Angular,

Configurar un proyecto en tu pc es rápido y fácil con QuickStart seed(La semilla de Inicio rápido), mantenida en github.

Asegurate de que tienes nodeJS y NPM instalado en tus sistema. Siendo así..

1. Crea una carpeta para el proyecto(puedes llamarla *quickstart* y renombrarla después).

2. Clona desde github o descarga la Quickstart seed dentro de la carpeta de tu proyecto.

3. Instala los paquetes npm.

4. Corre **npm start** para arrancar la aplicación de ejemplo.

### Clona desde github.

Agiliza los pasos clonar-y-arrancar con estos comandos de terminal.

```
git clone https://github.com/angular/quickstart.git quickstart
cd quickstart
npm install
npm start
```

## Descarga

Descarga la Quickstart seed y descomprime el zip dentro de la carpeta del proyecto. Entonces agiliza los siguientes pasos con los siguientes comandos.

```
cd quickstart
npm install
npm start
```

## Borra los archivos no esenciales(opcional).

Puedes borrar los archivos no esenciales que conciernen al repositorio de mantenimiento del quickStart y testeo.(Incluyendo todo lo relacionado con git así como la carpeta **.git** y **.gitignore!**).

*Haz esto solo al principio para evitar el borrado de tus propios test y configuracion de .git!*

Abre la ventana de la terminal en la carpeta del proyecto e ingresa los siguientes comandos de acuerdo a tu sistema.

## OS/X bash

```
xargs rm -rf < non-essential-files.osx.txt
rm src/app/*.spec*.ts
rm non-essential-files.osx.txt
```

## Windows

```
for /f %i in (non-essential-files.txt) do del %i /F /S /Q
rd .git /s /q
rd e2e /s /q
```

## Que hay en la QuickStart seed?

La quickstart seed contiene la misma aplicación que la Quickstart seed playground (el ejemplo en línea), pero su verdadero propósito es proveer sólidos fundamentos para un desarrollo local. Consecutivamente hay muchos otros archivos dentro de la carpeta de tu proyecto, muchos sobre los cuales puedes aprender después.

Pongamos atención en el siguiente árbol de archivos typescript en la carpeta **/src**

src

```
|--- app
|   |--- app.component.ts
|   |--- app.module.ts
|--- main.ts
```

src/app/app.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'my-app',
  template: `<h1>Hello {{name}}</h1>`
})
export class AppComponent { name = 'Angular'; }
```

src/app/app.module.ts

```
import { NgModule }      from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { AppComponent }  from './app.component';

@NgModule({
  imports:      [ BrowserModule ],
  declarations: [ AppComponent ],
  bootstrap:   [ AppComponent ]
})
export class AppModule { }
```

src/main.ts

```
import { platformBrowserDynamic } from '@angular/platform-browser-dynamic';
import { AppModule } from './app/app.module';

platformBrowserDynamic().bootstrapModule(AppModule);
```

Todas las guías y libros tienen al menos estos archivos principales. Cada archivo tiene un propósito distinto y evoluciona independientemente a medida que la aplicación crece.

Los archivos que están fuera de la carpeta **src/** conciernen a la construcción, despliegue y teste de tu aplicación. En ellos se incluyen los archivos de configuración y las dependencias externas.

Los archivos que están dentro de la carpeta **src/** “pertenecen” a tu aplicación. Añade archivos Typescript, css y html nuevos dentro del directorio **src/**, la mayoría de ellos dentro de **src/app** a menos que se le indique lo contrario.

Los siguientes están dentro de **src/**

**app/app.component.ts :**

Define el mismo **AppComponent** que el de quickstart playground. Es el componente raíz de lo que se convertirá en un árbol de componentes anidados a medida que la aplicación evoluciona.

**app/app.module.ts :**

Define **AppModule** el modulo raíz que le dice a angular como ensamblar la aplicación. En este momento solo declara el AppComponent. Pronto habrá mas componentes por declarar.

**main.ts :**

Compila la aplicación con el compilador JIT y arranca el modulo principal de la aplicación(AppModule) para correr en el navegador. El compilador JIT es una elección razonable durante el desarrollo de la mayoría de los proyectos y es la única elección viable para un ejemplo corriendo en un entorno de programación en línea(live-coding) como Plunker. Aprenderás sobre opciones alternativas para la compilación y el despliegue después en la documentación.

**SIGUIENTE PASO**

Si eres nuevo(a) en Angular, te recomendamos continuar con la ruta de aprendizaje.*(Que se explica en la sección "Aprendiendo angular" del archivo Angular2Español.pdf).*

**INSTALACIÓN DE NODEJS Y NPM**

Node.js y npm son esenciales para el desarrollo web moderno con Angular y otras plataformas.

Node potencia el desarrollo en cliente y de herramientas para el mismo. El administrador de paquetes NPM es una aplicación de node, e instala librerías JavaScript.

Consiguelos ahora a traves de este enlace:

<https://docs.npmjs.com/getting-started/installing-node>

**Verifica que tienes corriendo node v4.x.x o superior y npm 3.x.x o superior ingresando los comandos `node -v` y `npm -v` en la ventana de la terminal/consola.** Versiones mas antiguas producen errores.

Nosotros recomendamos NVM para el manejo de multiples versiones de node y npm. Quizas necesites NVM si ya tinens proyectos corriendo en tu maquina que use otras versiones de node y npm.

**Fuente:**<https://angular.io/docs/ts/latest/glossary.html>

**Daniel GA:** *Este documento así como todos los que traduzca son hechos en mis ratos libres no se cada cuanto saque el siguiente capítulo. Esto lo hago por apoyar a la comunidad JavaScript Hispanohablante que a veces se siente limitada por la falta de buena documentación gratuita y en nuestro idioma, en su momento fui uno de ellos, y también es para practicar el idioma inglés.*

*Si quieres aportar traduciendo puedes ponerte en contacto conmigo para sincronizar nuestro trabajo.*

Pretendo mientras Dios y el tiempo me dejen, completar esta documentación y continuar con otra tal vez React o Vue no se . Si quieres estar al tanto de los documentos traducidos que vaya sacando puedes agregarme o seguirme en Facebook <https://www.facebook.com/DanielGAcom> y si quieres publicar este material adelante solo dame créditos :).