# Curso de Angular

COMANDOS VERSIONES

* Node JS: node -v
* TypeScript: tsc --version
* Angular: ng v
* Ionic: ionic -v
* NPM: npm --version

# Comandos NPM

npm install -> instala todas las dependencias que el proyecto necesita para funcionar en NodeJS.

npm start -> levanta nuestro webpack-server -> servidor para ejecutar NodeJS.

CTRL + C -> Baja nuestro webpack-server -> servidor para ejecutar NodeJS.

# TypeScript

Documentación oficial: <https://www.typescriptlang.org/docs/handbook/2/everyday-types.html>

# Angular

Sudo npm uninstall -g @angular/cli: uninstall angular

Sudo npm install -g @angular/cli@latest: installer angular

## Crear un proyecto paso a paso

* Creamos una carpeta del proyecto
* Abrimos el terminal y nos ubicamos en la carpeta
* Ejecutamos el comando -ng new “nombre del proyecto”
* Le decimos que si -y
* Le decimos que no -n > si lo queremos crear en manual
* Elegimos el tipo de styles a manejar.

## Listo..

# Comandos

ng serve -o : Tomamos la aplicación > transpira todo nuestro código a JS > monta un servidor mediante web-pack >? -o que lo abra en cuanto este disponible.

ng g m ‘nombre del modulo’ : Genera un modulo en la carpeta app

ng g c ‘ruta / nombre del componente’ ‘path’ : Generar un componente en la ruta especifica

ng g s ‘ruta / nombre del servicio‘path’ : Generar un servicio en la ruta especifica

PATHS

* --skip-tests: Decimos que no genere el archivo de testing
* -is: decimos que no genere el archivo de styles

PRODUCCION

ng build --prod : Nos genera la versión de producion para subir al hosting

NODE COMANDOS

npm init: Inicializamos un proyecto de nodeJS – enter a todo

node index.js: corremos el backend de la aplicación

podemos instalar “nodemon” para que corra sola la aplicación:

npm install -g nodemon –Global

npm install --save-dev nodemon –proyecto

nodemon index.js para que corra la aplicación con nodemon y no toque estar ejecutando el comando.

Podemos crear dos scrip para ejecutar la aplicación por defecto.

"scripts": {

"dev": "nodemon index.js", -- desarrollo

"start": "node index.js" producion

},

npm run dev: colocamos a correr la aplicación en desarrollo

npm start: colocamos a correr la aplicación en producción

/////--///

npm i “packete” decirle a node que instale ese paquete

bcryptjs – instalar packete para incriptar contraseña

cors – instalar packete Para hacer peticiones cross domain

dotenv – instalar packete paraconfigurar variables de entorno

express – framework para montar rápidamente un servidor con servicios rest por defecto.

Express-validator – hacer validaciones en express

Jsonwebtoken – para creae jsonwebtoken

Mongoose: - para realizar la comunicación, conexión y trabajo con la base de datos mongoDb

Npm i “nombre”@”version”: instala version en especifico.