

# Angular

Introducción



# Angular

---

“Angular, es un **framework JavaScript** de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página (SPA)”



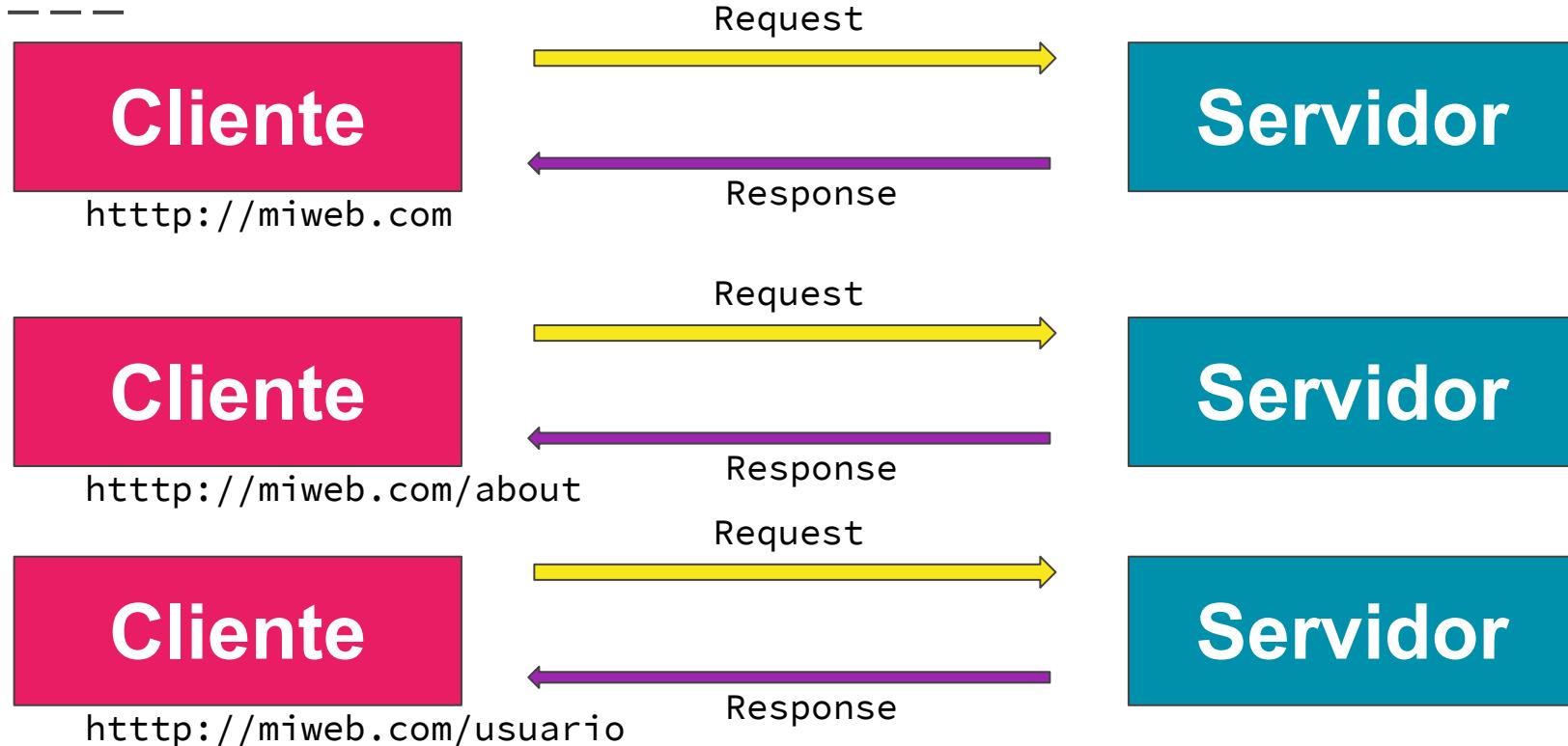
WIKIPEDIA  
The Free Encyclopedia



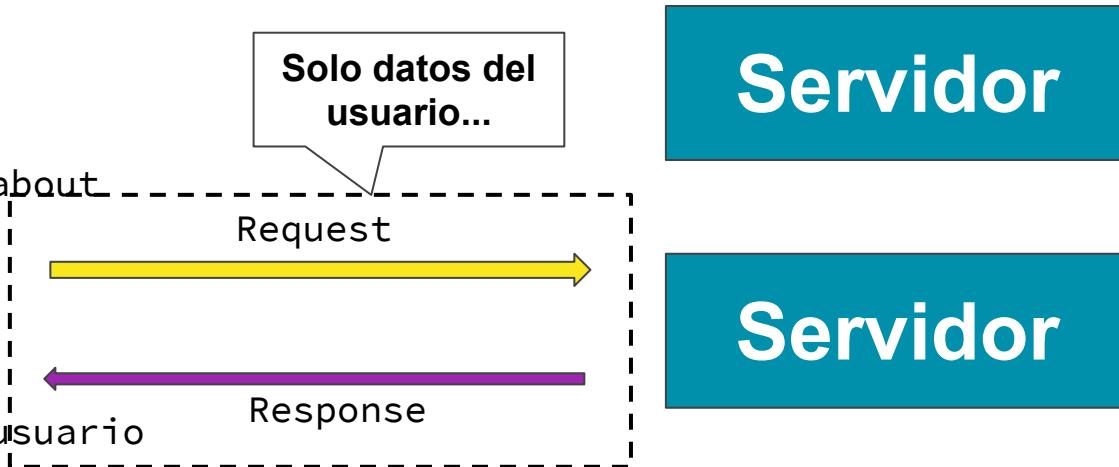
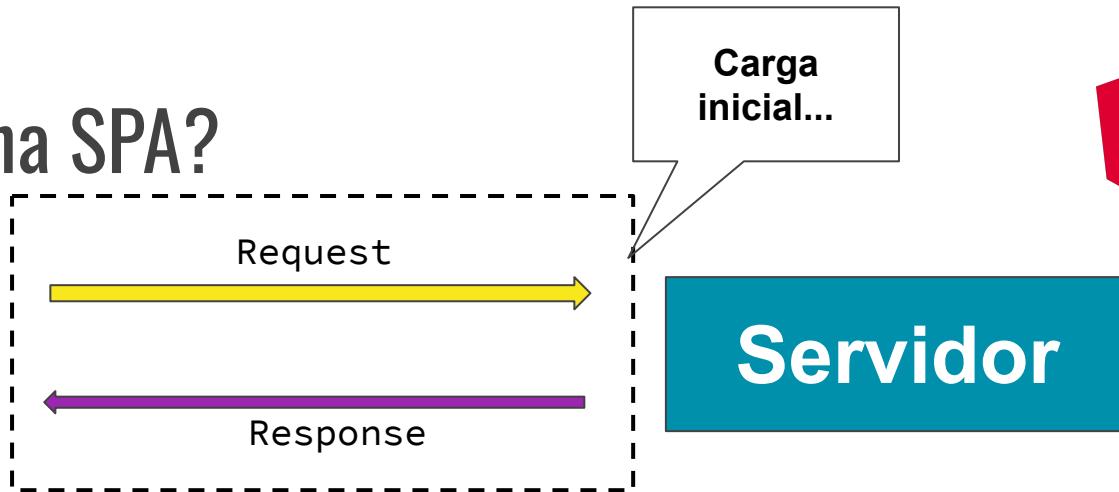
# Qué es una Single Page App (SPA)?



# Cómo funciona una página web tradicional?



# Cómo funciona una SPA?





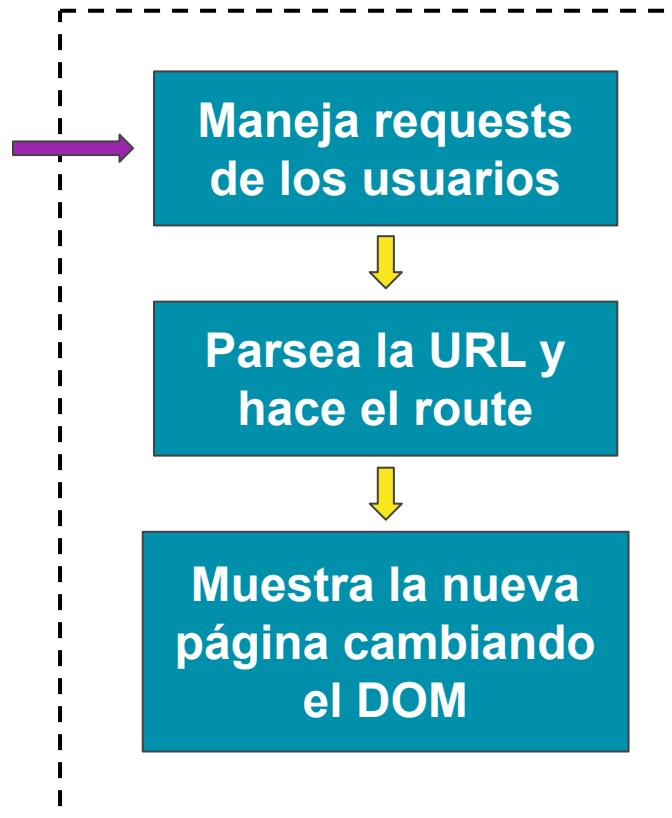
ORIGINAL  
COMEDY



# Cómo funciona?

## Angular

Cambia la URL/  
Modifica el DOM



Angular Route

Directives and  
Component  
Databinding

# Un poco de historia...

---



- Creado y mantenido por Google + Comunidad
- AngularJS:
  - Primera versión Octubre 2010
  - Versión estable 1.7.x
- Angular (2+):
  - La versión 2.x reescribe completamente el código
  - Está escrito en **TypeScript**.
  - Versión en actual 10.x
- Angular es mucho más performante que AngularJS (Angular 1).

# Vamos a aprender

Angular



-----

# Qué es TypeScript?

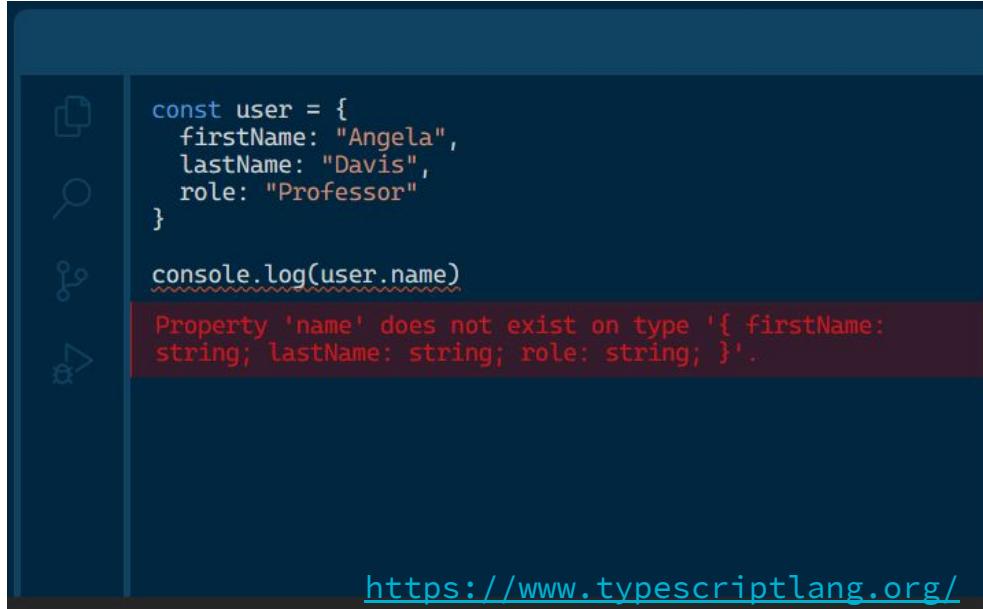
---

---

Angular está enteramente escrito en TypeScript (TS), y además propone este lenguaje como principal.

TypeScript extiende  
JavaScript agregando:

- **Tipos** (string, number, boolean, etc)
- **Interfaces**



The screenshot shows a code editor interface with a dark theme. On the left, there's a vertical toolbar with icons for file operations, search, and navigation. The main area contains the following TypeScript code:

```
const user = {  
  firstName: "Angela",  
  lastName: "Davis",  
  role: "Professor"  
}  
  
console.log(user.name)
```

A red error message is displayed in a tooltip below the line `console.log(user.name)`:

Property 'name' does not exist on type '{ firstName: string; lastName: string; role: string; }'.

At the bottom right of the slide, there is a URL:

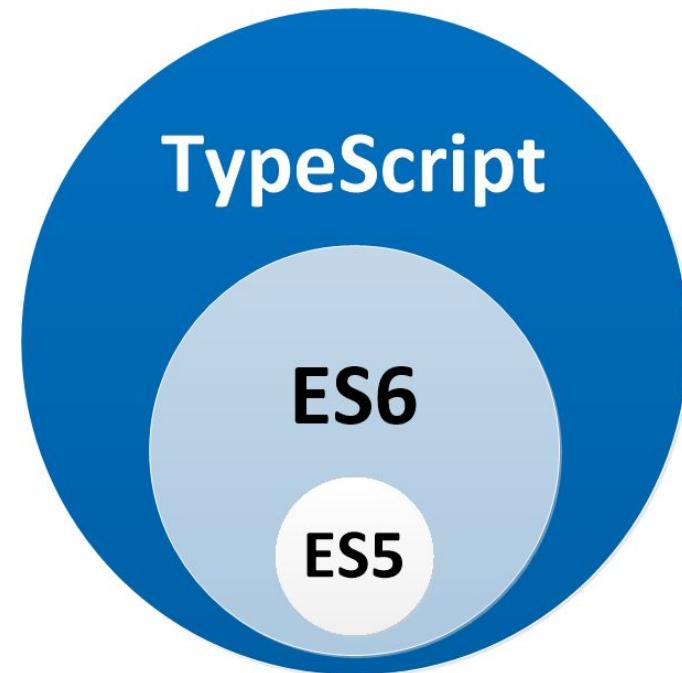
<https://www.typescriptlang.org/>

# Cómo funciona?

---

TypeScript es un super-conjunto de Javascript.

- Escribo el código en TypeScript.
- Se **transpila** a Javascript (ES5 o ES6).
- Se ejecuta en el browser como Javascript.





# Por qué lo eligieron para escribir Angular?

---

- Enteramente Open Source.
- Código más fácil de leer y de entender.
- Ahorra tiempo encontrando y evitando errores antes de ejecutar el código.
- Tiene muchas y buenas herramientas que un IDE puede aprovechar (autocompletado, refactoring, navegación entre métodos, etc.)



# Qué va a cubrir este curso?

---

---

- Components & Templates
- Databinding
- Directives & Pipes
- Services & Dependency Injection
- Routing
- HttpClient (consumir servicios web)
- Ademas de aprender **TypeScript!**



# Setup del ambiente

---

- Que vamos a necesitar?
  - [Node.js](#) (v10.13+)
  - npm (se instala con node)
  - [Git](#)
  - IDE: [VSCode](#)/[Atom](#)/Sublime/Vim?
  - Browser: [Chrome](#)/[Firefox](#)/Safari/IE?



**Para aprender....**

**Vamos a construir  
algo**



-----



# Proyecto “The Brewery”

---

Nos contactó el dueño de una nueva cervecería en Tandil y quiere armar una aplicación para vender sus cervezas por internet.





# Proyecto “The Brewery” - Requerimientos

---

- Como usuario quiero ver una lista de cervezas, con nombre, descripción, precio, stock e imagen.
- Como usuario quiero poder agregar cervezas a mi carrito de compras.
- Como usuario quiero poder indicar la cantidad de cervezas que quiero en mi carrito de compras.
- Como usuario quiero poder eliminar cervezas de mi carrito de compras.



# Formato y aprobación



# Cómo se aprueba?

---

- Nosotros seguimos un ejemplo conductor
- Ustedes hacen una app web de lo que quieran (ej, la misma de Web 1)
- Individual

# Qué evaluamos?

---

- Que hayan aplicado lo mismo que vimos.
- No se evalúa la UI, nos enfocamos en la parte de Angular y TS.
- Entregado -> 10 / No entregado -> Desaprueba
- <https://stackblitz.com/>

# Recuerden

----

Ustedes sigan el ejemplo, pero haciendo un sitio del tema que prefieran.

Algo similar a lo de Web 1 les va a andar bien.

Este “proyecto” es lo que tienen que entregar para aprobar.

# Ready, Go!

Creemos nuestra primer app  
con Angular



-----



# Angular CLI

---

Angular CLI (Command Line Interface) nos facilita el trabajo a la hora de crear y mantener una aplicación Angular.

Para instalarlo usando *npm*:

- Abrir una consola
- Ejecutar **npm install -g @angular/cli**
  - -g lo instala global en nuestra máquina
- Nos habilita el comando **ng**
  - Para probar ejecutar **ng --version**

# Behind the wall...

---



Si no les anda npm en Facultad, tienen que configurar el proxy.

```
npm config set proxy http://proxy.exa.unicen.edu.ar:8080  
npm config set https-proxy http://proxy.exa.unicen.edu.ar:8080
```

No se olviden de desconfigurarlo cuando están en sus casas

```
npm config rm proxy  
npm config rm https-proxy
```



# Angular CLI

---

## ng new

Crea una app Angular

- Ejecutar: **ng new [nombre-de-la-app]**

--routing=true  
--style=scss

### Ejemplo

```
ng new brewery --routing=true --style=scss
```

A screenshot of a terminal window titled '1.bash'. The window has a dark background and light-colored text. It shows the command 'ng new brewery --routing=true --style=scss' being typed at the prompt. The terminal is located on a Mac OS X desktop, as indicated by the window title bar and the Dock at the bottom.

# Behind the wall...

---



Si les tira un error de que no se pueden instalar los typings.

- Entrar a la carpeta del proyecto
- Crear un archivo que se llame **.typingsrc**
- Completarlo con:

```
proxy=http://proxy.exa.unicen.edu.ar:8080  
rejectUnauthorized=false
```

- Ejecutar: **typings install**

# Behind the wall...

---

---



1. bash

```
~/Projects/university/angular-app (25|master)
$ vim .typingsr[]
```



# Angular CLI

----

## **ng serve**

Corre la app Angular en un servidor local

- Entramos a la carpeta que nos creó: `cd [Nombre-de-la-App]`
- Ejecutar: **ng serve**
- Abrir el browser en: **http://localhost:4200**

Puerto

A yellow rectangular callout box contains the word "Puerto". An arrow originates from the bottom right corner of this box and points towards the port number "4200" in the URL "http://localhost:4200" listed in the bullet points above.

Más Info: <https://cli.angular.io/>



test-app app is running!

## Resources

Here are some links to help you get started:

Learn Angular >

< > CLI Documentation >

Angular Blog >

## Next Steps

What do you want to do next with your app?

+ New Component

+ Angular Material

+ Add PWA Support

+ Add Dependency

+ Run and Watch Tests



ng generate component xyz



Love Angular? [Give our](#)

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
1: bash
Compiling @angular/forms : es2015 as esm2015
Compiling @angular/platform-browser/testing : es2015 as esm2015
Compiling @angular/platform-browser-dynamic : es2015 as esm2015
Compiling @angular/platform-browser/animations : es2015 as esm2015
Compiling @angular/common/http/testing : es2015 as esm2015
Compiling @angular/router/testing : es2015 as esm2015
Compiling @angular/platform-browser-dynamic/testing : es2015 as esm2015

chunk {main} main.js, main.js.map (main) 60.6 kB [initial] [rendered]
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 141 kB [initial] [rendered]
chunk {runtime} runtime.js, runtime.js.map (runtime) 6.15 kB [entry] [rendered]
chunk {styles} styles.js, styles.js.map (styles) 12.9 kB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 2.66 MB [initial] [rendered]
Date: 2020-08-24T23:02:09.367Z - Hash: eff915c5ad8f829f4e4c - Time: 16573ms
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **
: Compiled successfully.
```

# ¡Nuestra Primera App Angular andando!



— — —



# Qué tiene nuestro proyecto?

---

## Top Level

El nivel superior del workspace contiene archivos de configuración y subcarpetas para los archivos fuente y de testing. Los más importantes son:

- **angular.json** configuración de angular-cli default para el proyecto
- **package.json** define las dependencias del proyecto
- **src/** contiene el código fuente del proyecto
- **node\_modules/** provee paquetes npm globales a todo el proyecto
- **e2e/** contiene el código para testing end-to-end



# Qué tiene nuestro proyecto?

---

## src folder

Contiene todo el código fuente de nuestra aplicación. Los principales son:

- **index.html** es la página HTML principal. Angular CLI agrega automáticamente todos los archivos JavaScript y CSS.
- **app/** contiene la lógica y los datos del proyecto.
- **assets/** contiene las imágenes y otros “assets” que se usan en la app
- **environments/** contiene diferentes configuraciones de compilación
- **style.scss** estilos css principales de la aplicación
- **main.ts** punto de entrada principal de la aplicación

# Actividad: Objetivo a entregar de esta clase

---

1. Definir de qué es el sitio
2. Crear la aplicación local
3. Crear un proyecto en <https://stackblitz.com/>
4. Importarlo ([doc](#))
  - a. Subirla a Github y relacionarlo a Stackblitz (opcional)
  - b. Arrastrar archivos

# Referencias

---

- [Angular.io - Getting Started](#) - Documentación Oficial
- [Angular - The Force Awakens](#) - John Papa (YouTube)