



UNIVERSIDAD AUTÓNOMA
DE AGUASCALIENTES

Ciclo de vida de un componente

Tecnologías Web

ALUMNO:

Melgoza de la Torre Abraham

PROFESORA

Dra. Georgina Salazar Partida

6-C

ISC

constructor

ngOnChanges

ngOnInit

ngDoCheck

ngAfterContentInit

ngAfterContentChecked

ngAfterViewInit

ngAfterViewChecked

ngOnDestroy

“En Angular, cada componente tiene un ciclo de vida, una cantidad de etapas diferentes que atraviesa. Hay 8 etapas diferentes en el ciclo de vida de los componentes. Cada etapa se denomina *lifecycle hook event* o en ‘evento de enlace de ciclo de vida’. ” [1]

Constructor: Es lo primero que se ejecuta antes de cualquier otro *life hook*. Aquí se inyectan dependencias mayormente.

ngOnChanges: Este evento se ejecuta cada vez que se cambia un valor de un *input control*. Siempre recibe un *change data map* el cual contiene el valor actual anterior de la propiedad vinculada en un *SimpleChange*.

ngOnInit: Se ejecuta una vez que Angular despliega los *data-bound properties* o cuando el componente se ha iniciado y *ngOnChanges* se haya ejecutado.

ngDoCheck: Se activa cada vez que se verifica las propiedades de entrada de un componente. Nos permite implementar un algoritmo de detección de cambios personalizados.

ngAfterContentInit: Se ejecuta cuando angular realiza cualquier muestra de contenido dentro de las vistas de componentes y justo después de *ngDoCheck*. Está vinculado con las inicializaciones del componente hijo.

ngAfterContentChecked: Se ejecuta cuando el componente se ha verificado, se llama después del método *ngAfterContentInit*. También se invoca después de cada *ngDoCheck* y se relaciona principalmente con las inicializaciones del componente hijo.

ngAfterViewInit: Se ejecuta cuando la vista se ha inicializado por completo. Se llama después de *ngAfterContentChecked*. Solo se aplica a componentes.

ngAfterViewChecked: Se ejecuta después del método *ngAfterViewInit* y cada vez que la vista del componente verifique cambios y cuando se ha modificado cualquier enlace en directivas secundarias. Es muy útil cuando el componente espera algún valor que proviene de sus componentes secundarios.

ngOnDestroy: Este método se ejecuta justo antes de que Angular destruya los componentes. Permite dar de baja los observables y desconectar *event handlers* para poder evitar *memory leaks* o fugas de memoria.

Angular: Componentes y sus ciclos de vida - Angular Chile - Medium

By Tatiana Molina Container: Medium Publisher: Angular Chile Year: 2018 URL:
<https://medium.com/angular-chile/angular-componentes-y-sus-ciclos-de-vida-aa639e13a688>