



Antes de empezar...

- Asegúrate de tener listo tu editor (Visual Studio Code, por ejemplo) y de manejar la consola
- Asegúrate de haber bajado las actualizaciones del proyecto desde el github de Taws.
https://github.com/Taws-Espol/front_vue
- Asegúrate de haber bajado los complementos de Vue para tu editor, y el navegador web (opcional)



Vue.js

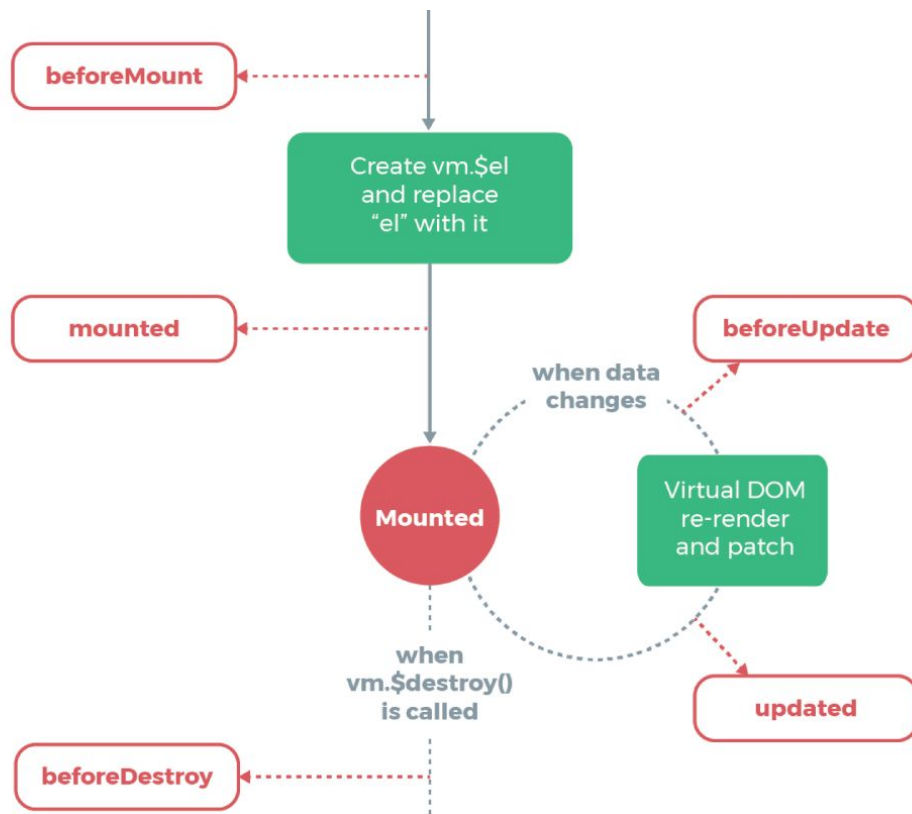
Taller de Vue.js Parte 2

Presentador: Piero Ulloa

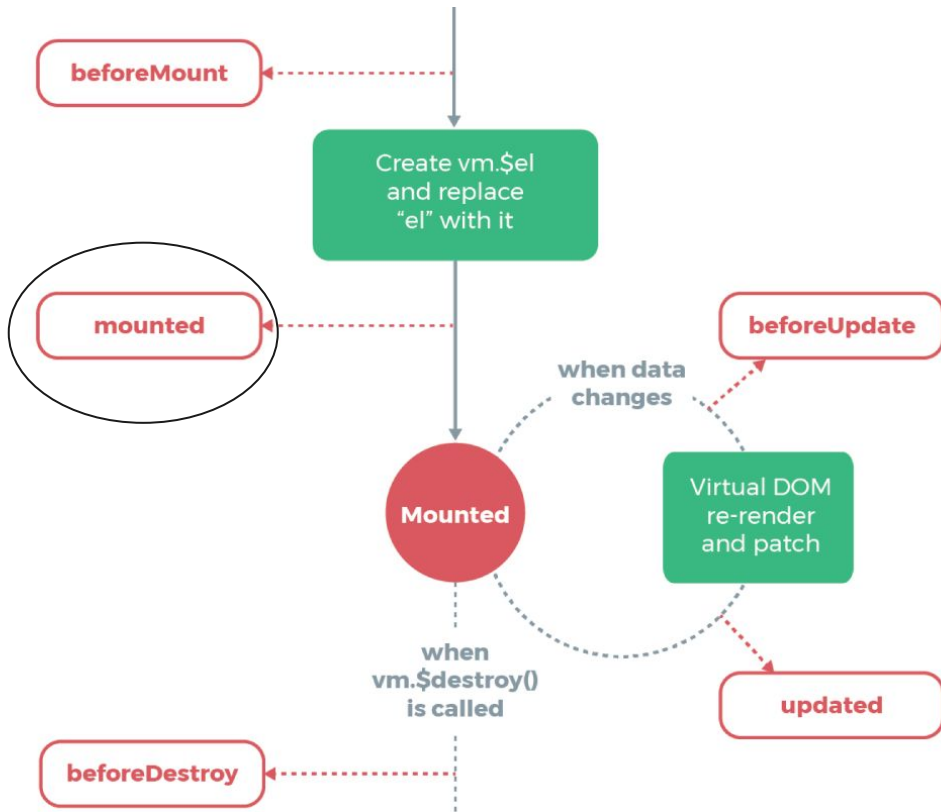
Lifecycle hooks

Hooks de ciclo de vida

- Los componentes tienen un ciclo de vida
- Funciones como `mounted`, `beforeMount`, `beforeUpdated`, `updated`, `beforeDestroy`, son llamadas por Vue en sus componentes.

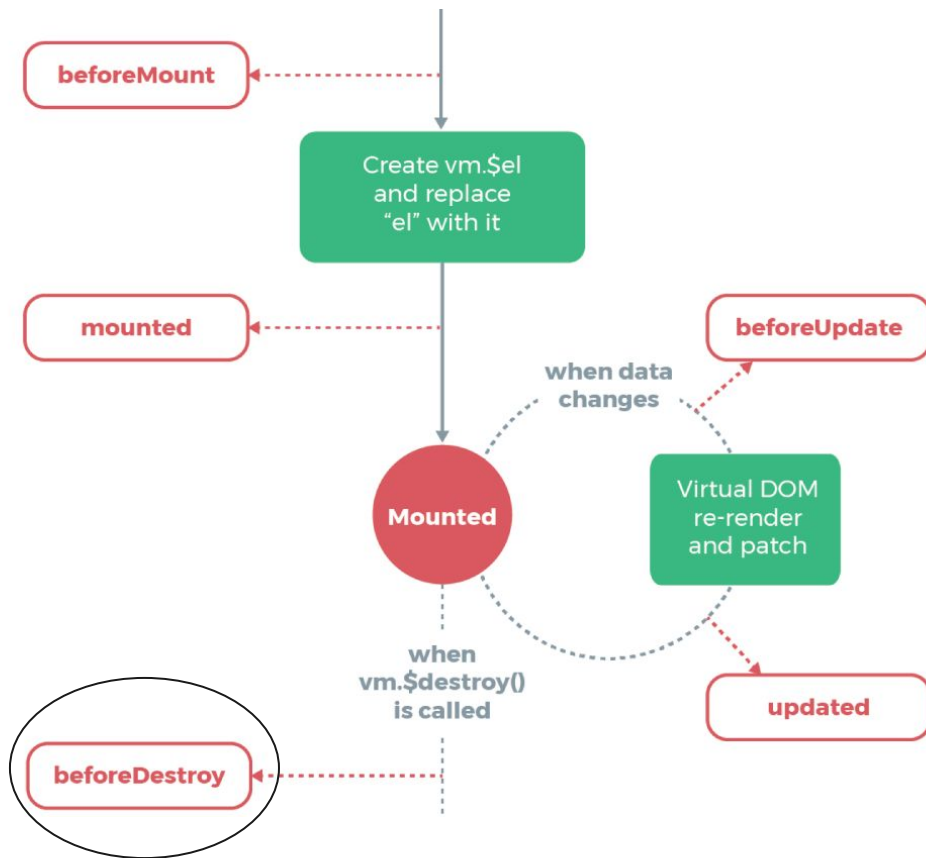


- Esta función es llamada cuando los objetos reactivos ya están disponibles y es posible alterar los valores.
- En este callback se puede usar funciones que hagan algo se haga antes de ser mostrado al usuario.



beforeDestroy()

- Sirve para prepararse para eliminar callbacks.
- Ej. estás usando un API que te permite refrescar la pantalla dinámicamente. En este callback deberías eliminar el callback que haces cuando recibes datos.



Pequeña intro a Bootstrap



+





¿Qué es Bootstrap?

Un framework de reglas CSS que facilitan el trabajo al programador web, evitando tener que estilizar cada componente.

Se utiliza estableciendo el atributo class a una de las clases utilizadas por Bootstrap





¿Cómo descargar bootstrap?

Sólo tienen que escribir en el terminal dentro de la carpeta de su proyecto:

- `npm install bootstrap bootstrap-vue`

Y ya lo tienen listo para usar.



¿Cómo utilizo bootstrap?

Importálo en tu main.js, para que esté disponible de manera global.

```
import 'bootstrap/dist/css/bootstrap.css';
```



Cómo usar bootstrap.

La parte principal de bootstrap es su sistema grid (cuadrícula).

La cuadrícula tiene 12 "espacios" por defecto

Poseemos las siguientes clases CSS:

.container: Contenedor que centra el contenido en tamaños grandes

.row: Ocupa una línea entera de contenido.

.col-X: Ocupa X celdas de la cuadrícula

col	col	col	col
col-8			col-4

```
1  <div class="bd-example">
2    <div class="row">
3      <div class="col">col</div>
4      <div class="col">col</div>
5      <div class="col">col</div>
6      <div class="col">col</div>
7    </div>
8    <div class="row">
9      <div class="col-8">col-8</div>
10     <div class="col-4">col-4</div>
11   </div>
12 </div>
```

Columnas en acción

iManos a la obra!



¡Vamos a crear una app con Vue!

- Crear un mini-sitio de compras
- Tener carrito de compras
- Hacer checkout!





Comandos para crear APP desde cero

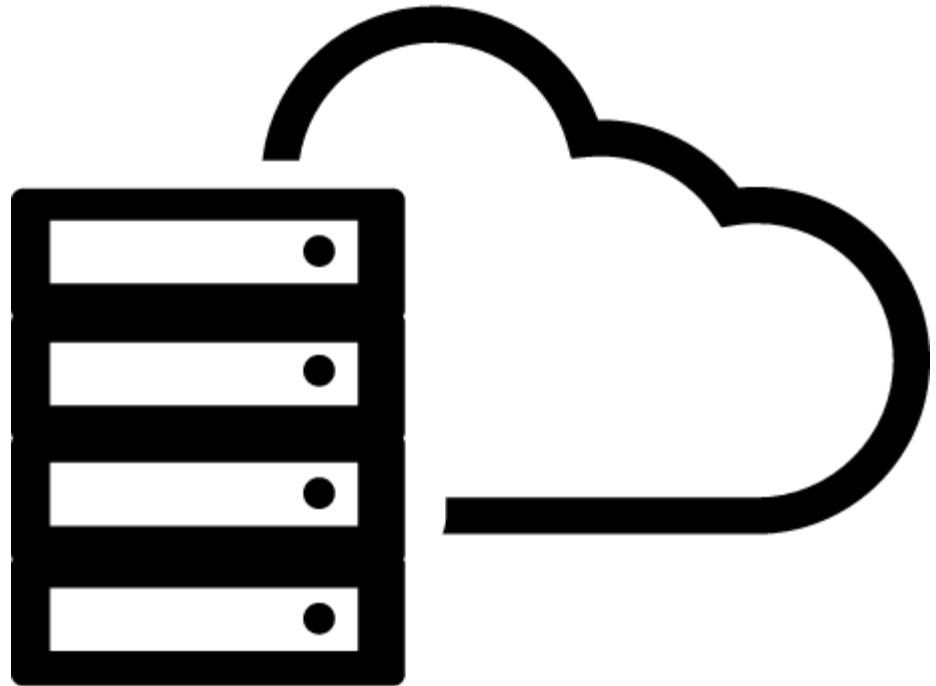
- `vue create <nombre del proyecto> --default`




Paso #1

El modelo

- Tenemos que definir qué modelos del producto queremos mostrar!
- Precio, nombre, foto, stock, etc..
- Precio, nombre, foto, stock son los que usaremos.





Paso #2

Componente del producto.


- Tenemos que transformar la información recibida del producto en un widget para mostrar!
- Creemos un componente en Vue!



Paso #3 Componente del listado de productos


- Tenemos que poder visualizar la lista de productos!
- Cada widget debe tener el CSS apropiado para que tengamos una lista!





Paso #4 Leer los productos desde otra parte.

- Tenemos que suplir el modelo dentro de nuestro componente para que los productos se empiecen a listar.
- Por ahora, vamos a servirlos desde nuestro dev server.



Paso #6: Crear el componente del carrito

- Hay que crear un componente en donde guardan los ítems que se han seleccionado del carrito



Paso #6 Cobrar por el trabajo (Profit)

- ¡Son todos excelentes! ¡Déense una palmadita en la espalda, por que han hecho un buen trabajo!