

Desarrollo Web en Entorno Cliente

# UD 14. Introducción a Vue 2

---

Actualizado Enero 2020

## Licencia



**Reconocimiento – NoComercial - CompartirIgual (BY-NC-SA):** No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

## Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

 **Importante**

 **Atención**

 **Interesante**

## ÍNDICE DE CONTENIDO

<b>1. Introducción</b>	<b>4</b>
1.1. ¿Qué es programación reactiva?	4
1.2. Acceso a los ejemplos presentados	4
<b>2. ¿Preparar el sistema de desarrollo Vue a mano o utilizar Vue-CLI?</b>	<b>4</b>
<b>3. Depurando Vue 2</b>	<b>5</b>
<b>4. Ejemplos prácticos con Vue 2</b>	<b>5</b>
4.1. Ejemplo 01: Hola-mundo	5
4.2. Ejemplo 02: componente-tablas-de-multiplicar	6
4.3. Ejemplo 03: consumir-api-axios	6
4.4. Ejemplo 04: interpolación-atributos	7
4.5. Ejemplo 05: directiva-v-html	7
4.6. Ejemplo 06: lista-de-tareas	7
4.7. Ejemplo 07: simulación-chat-dos-usuarios	7
4.8. Ejemplo 08: carrito-compra	7
4.9. Ejemplo 09: datos-artistas	8
<b>5. Enlaces interesantes sobre Vue 2</b>	<b>8</b>
<b>6. Bibliografía</b>	<b>8</b>
<b>7. Autores (en orden alfabético)</b>	<b>8</b>

## UD14. INTRODUCCIÓN A VUE 2

### 1. INTRODUCCIÓN

Vue 2 JS <https://vuejs.org/> es un framework de Javascript orientado a facilitar el desarrollo de aplicaciones web, aplicaciones SPA (Single Page Application), etc. Una de las principales ventajas de usar Vue es que implementa lo que se llama un programación reactiva.

#### 1.1 ¿Qué es programación reactiva?

De una forma muy simple, podemos decir que la programación reactiva es un paradigma de programación donde los cambios son propagados al resto del sistema.

Un ejemplo muy sencillo sería una etiqueta HTML que muestra el texto de una variable llamada “miVariable”. Al hacer un cambio en dicha variable, ese cambio se propaga automáticamente a todas las zonas de la aplicación

Para más información [https://en.wikipedia.org/wiki/Reactive\\_programming](https://en.wikipedia.org/wiki/Reactive_programming)

**Esta unidad está basada en la presentación de ejemplos comentados**, así que este documento es más bien una guía de qué ejemplos visitar y en qué orden.

En algunos de los ejemplos se usa Bootstrap para el aspecto de la página <https://getbootstrap.com/> que habéis visto en el módulo “Desarrollo de interfaces Web”. Si no lo estáis cursando, no os preocupéis, no es necesario para realizar la unidad.

Aquí tenéis un tutorial interesante de Bootstrap <https://www.w3schools.com/bootstrap/>

#### 1.2 Acceso a los ejemplos presentados

Los ejemplos que han sido comentados para esta unidad están disponibles en el aula virtual e incluidos en mi cuenta de GitHub <https://github.com/sergarb1/Vuejs2-Examples>.

Como extra, durante la unidad, os recomiendo que visitéis estos Screencast donde explican de manera visual cómo poner en marcha los ejemplos de la documentación oficial de Vue 2 <https://scrimba.com/g/glearnvue>

### 2. ¿PREPARAR EL SISTEMA DE DESARROLLO VUE A MANO O UTILIZAR VUE-CLI?

En los distintos ejemplos presentados, gestionaremos a mano los paquetes necesarios a instalar mediante NPM, la configuración de Webpack, la inclusión de la biblioteca Vue mediante CDN, la inclusión de distintas bibliotecas, componentes, etc.

El motivo de hacerlo “a mano” es que visitar los ejemplos os ayude a comprender cómo está montado todo por debajo, que está pasando, que hace cada biblioteca, etc.

No obstante, a efectos prácticos, existe la herramienta “Vue CLI” (VUE Command Line Interface) que nos permite crear y configurar proyectos Vue de forma sencilla, con todo lo necesario para funcionar desde ya <https://cli.vuejs.org/>.

Aunque en esta unidad no la trataremos, una vez estéis familiarizados con proyectos, recomendamos su uso. Además podéis utilizarla para las prácticas propuestas si lo estimáis conveniente.

Aquí enlace un video tutorial de Vue CLI” [https://www.youtube.com/watch?v=6V9AZ93I\\_rg](https://www.youtube.com/watch?v=6V9AZ93I_rg)

### 3. DEPURANDO VUE 2

Para facilitar la depuración de aplicaciones que usen Vue, se recomienda el uso de la extensión del navegador “Vue Devtools”. La tenéis disponible en <https://github.com/vuejs/vue-devtools>

Más información sobre su uso en este video <https://www.youtube.com/watch?v=wBtNNomNGVc>

### 4. EJEMPLOS PRÁCTICOS CON VUE 2

Antes de empezar, es de recomendada lectura la guía de inicio de la propia documentación de Vue 2 <https://vuejs.org/v2/guide/> (inglés) o <https://es-vuejs.github.io/vuejs.org/v2/guide/> (castellano). Vuelvo a recordar, que como extra os recomiendo que visitéis estos Screencast donde explican de manera visual cómo poner en marcha los ejemplos de la documentación oficial de Vue 2 <https://scrimba.com/g/glearnvue>

A continuación comentamos los ejemplos que he preparado en el orden de visita recomendado. Cada vez que se trata un elemento nuevo de Vue, se incluye un enlace a la ayuda donde explica dicho elemento.

#### 4.1 Ejemplo 01: Hola-mundo

Ejemplo de aplicación sencilla hola mundo. En esta aplicación se muestran aspectos básicos como la creación de una aplicación Vue, su enlace a un contenedor donde será válida, la inclusión de bibliotecas y la asociación de un evento clic a un botón.

Ahí vemos uno de los principios de Vue: modificar el valor de una variable hace que automáticamente se actualice el nuevo valor en todos los lugares que aparece dicha variable.

A la hora de configurarlo, hemos seguido la tónica que seguiremos en posteriores ejemplos:

- Hemos incluido “Vue” mediante un CDN. Esto se ha hecho porque el uso de CDN (en producción) aumenta su velocidad, al estar habitualmente cacheados por los ISP.
  - Por defecto, se enlaza a Vue en versión desarrollo, para poder realizar depuración utilizando “Vue Devtools”.
    - `<script src="https://vuejs.org/js/vue.js"></script>`
  - Para ponerlo en producción, quitar el script anterior y enlazar al CDN minimizado.
    - `<script src="https://cdn.jsdelivr.net/npm/vue"></script>`
  - El resto de bibliotecas si están en local (Webpack, Babel, ESLint, etc.)
  - Los fuentes están en la carpeta “src” y los ficheros generados se guardan en la carpeta “dist”
  - Al construir la aplicación para su distribución, usamos comando “npm run build”.
  - Para hacer pruebas, utilizaremos el comando “npm run server” que lanzará nuestro código en un pequeño servidor web para realizar pruebas y se actualizará automáticamente con cada cambio que hagamos.

#### 4.1.1 Enlaces a novedades del ejemplo

- Guía de inicio: <https://vuejs.org/v2/guide/>
- Manejo de eventos: <https://vuejs.org/v2/guide/events.html>

#### 4.2 Ejemplo 02: componente-tablas-de-multiplicar

En esta aplicación, se introducen conceptos importantes como la creación de componentes re-utilizables en Vue, el uso de v-model para enlazar el valor de un elemento HTML con una variable (los cambios en el elemento modifican el valor de la variable y viceversa), el uso de v-for para repetir código y el uso de v-show para decidir si algún elemento HTML se muestra o no.

En este ejemplo, introducimos el formato de componentes **SFC** (Single File Component). Este formato nos permite definir en un único fichero todo lo necesario para incluir un componente Vue y que sea utilizado por la aplicación. En el ejemplo el componente está en el fichero **“ComponenteTabla.vue”**.

Para facilitar la carga de estos componentes utilizando Webpack, hemos utilizado la biblioteca “vue-loader” <https://vue-loader.vuejs.org/guide/>

Si queréis instalar “Vue Loader” en algún proyecto, podemos hacerlo con la orden "npm install vue vue-loader vue-template-compiler css-loader"

En el ejemplo presentado la instalación y configuración ha sido manual. Los ficheros “package.json” y “webpack.config.js” se han adaptado para el uso de “Vue Loader” siguiendo el ejemplo disponible en el sitio web <https://vue-loader.vuejs.org/guide/#manual-setup>

Otra opción para poder realizar de forma sencilla la configuración de “Vue Loader” sería utilizar “Vue CLI” que facilita mucho esta y otras tareas tal como hemos comentado en el punto 3.

#### 4.2.1 Enlaces a novedades del ejemplo

- v-model: <https://vuejs.org/v2/guide/forms.html>
- v-for: <https://vuejs.org/v2/guide/list.html#v-for-with-a-Range>
- v-show: <https://vuejs.org/v2/guide/conditional.html#v-show>
- Componentes Vue: <https://vuejs.org/v2/guide/components.html>

#### 4.3 Ejemplo 03: consumir-api-axios

Ejemplo donde mediante la biblioteca Axios, se consulta mediante una petición asíncrona (AJAX) una API que devuelve información en formato JSON y se muestra por pantalla. El funcionamiento es análogo a otras bibliotecas que hemos visto para usar AJAX.

La petición AJAX se realiza cuando se lanza el evento **“mounted”** que indica que la aplicación Vue se ha montado. Para más información

#### 4.3.1 Enlaces a novedades del ejemplo

- Eventos existentes al ser creada una aplicación Vue:

- <https://vuejs.org/v2/guide/instance.html#Instance-Lifecycle-Hooks>
- Biblioteca Axios: <https://github.com/axios/axios>
- Ejemplos Axios con Vue:
  - <https://es.vuejs.org/v2/cookbook/using-axios-to-consume-apis.html>

#### 4.4 Ejemplo 04: interpolación-atributos

Ejemplo donde valiéndose entre el enlace de atributos a variables usando **v-bind**.

**Importante:** al contrario que en v-model, que el enlace es bidireccional, usando **v-bind** se enlaza el valor de la variable al atributo, pero no al revés. Con ello se puede conseguir ciertos efectos muy útiles.

##### 4.4.1 Enlaces a novedades del ejemplo

- v-bind: Ejemplo que enlaza class con v-bind
  - <https://vuejs.org/v2/guide/class-and-style.html>

#### 4.5 Ejemplo 05: directiva-v-html

Ejemplo que sencillamente usa **v-html** para mostrar contenido de variables como HTML.

##### 4.5.1 Enlaces a novedades del ejemplo

- v-html: <https://vuejs.org/v2/guide/syntax.html#Raw-HTML>

#### 4.6 Ejemplo 06: lista-de-tareas

Ejemplo de aplicación para añadir tareas, marcar si se han realizado, etc.

Introduce en el ejemplo el uso de **v-if**, similar a v-show. La diferencia entre ambas directivas es la siguiente:

- Si **v-if** no se cumple, el elemento no está en el DOM
- Si **v-show** no se cumple, el elemento si está en el DOM pero no es visible.

Para que haya persistencia de la información, se usa localStorage de Javascript. Para cargarlo al inicio, se carga con el evento **mounted** Para saber cuándo guardar los cambios en localStorage, se introduce un **watcher** a la variable “tareas”. Este watcher que detecta que se modifica esta variable, lanza una acción guardando su contenido en localStorage.

##### 4.6.1 Enlaces a novedades del ejemplo

- v-if: <https://v1.vuejs.org/guide/conditional.html>
- watchers: <https://vuejs.org/v2/guide/computed.html#Watchers>

#### 4.7 Ejemplo 07: simulación-chat-dos-usuarios

Aplicación que usando conceptos vistos anteriormente, simula el chat entre dos usuarios.

#### 4.8 Ejemplo 08: carrito-compra

Aplicación que usando conceptos anteriores, pide al usuario que haga “login”. El usuario por

defecto que se espera es “admin” y su contraseña es “admin”. Una vez realizado el login, el ejemplo nos permite elegir productos para nuestro carrito de la compra.

#### 4.9 Ejemplo 09: datos-artistas

Aplicación que permite consultar datos de artistas y sus discografías, así como añadir nuevos artistas. Como principal novedad, usa los componentes router-link y router-view para con ello recibir información a partir de la ruta del navegador. También hemos utilizado Vuex para compartir datos de forma reactiva entre componentes (se comparten datos de artistas entre componente “Inicio.vue” y “Artista.vue”).

##### 4.9.1 Enlaces a novedades del ejemplo

- Guía para realizar routing: <https://vuejs.org/v2/guide/routing.html>
- Componente router: router-link y router-view: <https://router.vuejs.org/api/>
- Guía para usar Vuex <https://vuex.vuejs.org/guide/>
- Mutaciones en Vuex <https://vuex.vuejs.org/guide/mutations.html>

#### 5. ENLACES INTERESANTES SOBRE VUE 2

- Curso iniciación a Vue 2 <https://scrimba.com/g/glearnvue>
- Guía de Vue 2 en inglés <https://vuejs.org/v2/guide/>
- Guía de Vue 2 en castellano <https://es-vuejs.github.io/vuejs.org/v2/guide/>
- Videos de Vue Alfredo Profesor (Dos playlist)
  - <https://www.youtube.com/playlist?list=PL7VXpoQP1k27nOnbERp6hc3cT1Yc4HFvB>
  - <https://www.youtube.com/playlist?list=PL7VXpoQP1k247QUTTxHcYBNyCg1j2ALfD>
- How to build a SPA with Vue 2
  - <https://scotch.io/tutorials/how-to-build-a-simple-single-page-application-using-vue-2-part-1>
- Learn Vue 2 Step by Step <https://laracasts.com/series/learn-vue-2-step-by-step>
- Vue JS Fundamentals
  - [https://www.youtube.com/playlist?list=PLwAKR305CRO\\_1yAao-8aZiQnBqJeyng40](https://www.youtube.com/playlist?list=PLwAKR305CRO_1yAao-8aZiQnBqJeyng40)

#### 6. BIBLIOGRAFÍA

- [1] Javascript Mozilla Developer <https://developer.mozilla.org/es/docs/Web/JavaScript>  
[2] Javascript ES6 W3C [https://www.w3schools.com/js/js\\_es6.asp](https://www.w3schools.com/js/js_es6.asp)  
[3] Vue <https://es.vuejs.org/v2/guide/>

#### 7. AUTORES (EN ORDEN ALFABÉTICO)

A continuación ofrecemos en orden alfabético el listado de autores que han hecho aportaciones a este documento:

- García Barea, Sergi