



¡*V*uenvenidos a Vue!

El framework Javascript progresivo

¡Hola! ¿Empezamos?



Daniel Moya

Senior Frontend Developer

¡Hola! ¿Empezamos?



Daniel Moya

Senior Frontend Developer

Certificación
Oficial de Vue

+8 años
experiencia

Todo tipo de
proyectos

¡Hola! ¿Empezamos?



Daniel Moya

Senior Frontend Developer

Certificación
Oficial de Vue

+8 años
experiencia

Todo tipo de
proyectos

Europcar
Mobility Group

Nuxt 2 → Nuxt 3
Jest & Playwright
Migración a DDD
SEO & Performance

¡Hola! ¿Empezamos?



Daniel Moya

Senior Frontend Developer

Certificación
Oficial de Vue

+8 años
experiencia

Todo tipo de
proyectos

Europcar
Mobility Group

Nuxt 2 → Nuxt 3
Jest & Playwright
Migración a DDD
SEO & Performance

Background
Técnico

Startups &
side projects

Equipos
grandes



¡AVISO!

**Esta presentación presupone
un conocimiento básico de
JavaScript, HTML y CSS.**

¡Hola! ¿Empezamos?

01. ¿Qué es Vue?

02. Single File Components

03. Reactividad

04. Reutilización

05. Directivas

06. Gestión de Estado

07. ¿Por qué Vue?

08. ¡Workshop!

01. ¿Qué es Vue?



¿framework ... ?

Manera de trabajar

Herramientas

Ecosistema

Flexible

01. ¿Qué es Vue?



¿framework ... ?

¿ ... javascript ... ?

Manera de trabajar

¡o TypeScript!

Herramientas

y para HTML

Ecosistema

¡ ... incluso CSS!

Flexible

01. ¿Qué es Vue?



¿framework ... ?

Manera de trabajar

Herramientas

Ecosistema

Flexible

¿ ... javascript ... ?

¡o TypeScript!

y para HTML

¡ ... incluso CSS!

¿ ... progresivo?

Para HTML estático

Para SPA

Para SSR

Para SSG

Y lo que se le eche

01. ¿Qué es Vue?



Evan You

Ex-Google cansado de las
historias de Google

01. ¿Qué es Vue?



Vue 1

2015

Basado en AngularJS



Evan You

Ex-Google cansado de las
historias de Google

01. ¿Qué es Vue?



Vue 1

Vue 2

2015

2016

Basado en AngularJS

Misma sintaxis

Mejora general

BOOM de popularidad



Evan You

Ex-Google cansado de las
historias de Google

01. ¿Qué es Vue?



Vue 1

2015

Basado en AngularJS

Vue 2

2016

Misma sintaxis

Mejora general

BOOM de popularidad

Vue 3

2020

Breaking changes...

Reactividad refinada

Composition API

Madurez



Evan You

Ex-Google cansado de las
historias de Google

02. Single File Components

```
<script setup>
// Aquí va todo el JavaScript o Typescript
</script>

<template>
// Aquí va el HTML, el Pug, o cualquier otro preprocesador
</template>

<style>
// Aquí va el CSS, SCSS, SASS, etc
</style>
```

02. Single File Components

```
<script setup>
// Aquí va todo el JavaScript o Typescript
</script>

<template>
// Aquí va el HTML, el Pug, o cualquier otro preprocesador
</template>

<style>
// Aquí va el CSS, SCSS, SASS, etc
</style>
```

Tiene extensión .vue

Puede separarse en
diferentes archivos

Es una sección
de lógica

Debe compilarse

Todo está
correlacionado

Unidades mínimas
de código

03. Reactividad (I)

```
import { ref } from 'vue';
```

ref Variable reactiva, la base de todo

03. Reactividad (I)

```
import { ref } from 'vue';
```

ref Variable reactiva, la base de todo

```
const counter = ref(0);  
  
console.log(counter.value)  
// 0
```

03. Reactividad (I)

```
import { computed } from 'vue';
```

`computed` Variable reactiva **dependiente**

03. Reactividad (I)

```
import { computed } from 'vue';
```

computed Variable reactiva **dependiente**

```
const name = ref('Dani');
const lastName = ref('Moya');

const fullName = computed(() => {
  return name.value + ' ' + lastName.value
});

name.value = 'Daniel';
console.log(fullName.value) // 'Daniel Moya'
```

03. Reactividad (I)

```
import { watch } from 'vue';
```

watch Ejecuta cosas cuando cambian cosas

03. Reactividad (I)

```
import { watch } from 'vue';
```

watch Ejecuta cosas cuando cambian cosas

```
const name = ref('Dani');

watch(counter, (newValue, oldValue) => {
  console.log(newValue); // 'Daniel'
  console.log(oldValue); // 'Dani'
  alert('Hola ' + newValue); // Enseñará un modal con 'Hola Daniel'
});

name.value = 'Daniel';
```

03. Reactividad (II)

¿Cómo lo usamos en HTML?

```
<script setup>
const name = ref('Dani');
const fullName = computed(() => name.value + ' Moya');
const divStyles = ref({ background: "red", width: "50px", height: "50px", });

function handleClick() { alert('¡Hola!'); }
</script>

<template>
  <h1>{{ name }} ({{ fullName }})</h1>

  <input v-model="name">

  <div :style="divStyles" @click="handleClick()" />
</template>
```

@ → v-on

Entre las comillas solo pueden ir expresiones JS

04. Reutilización

ParentComponent.vue

```
<script setup>
import { ChildComponent } from 'child-component.vue'
function handleGreet(name) {
  alert('¡Hola ' + name + '!')
}
</script>

<template>
  <h1>Saludador</h1>

  <ChildComponent
    name="Dani"
    @greet="handleGreet('Dani')">
    >
      Esto aparecerá dentro del slot
  </ChildComponent>
</template>
```


04. Reutilización

ParentComponent.vue

```
<script setup>
import { ChildComponent } from 'child-component.vue'
function handleGreet(name) {
  alert('¡Hola ' + name + '!')
}
</script>

<template>
  <h1>Saludador</h1>

  <ChildComponent
    name="Dani"
    @greet="handleGreet('Dani')">
    >
    Esto aparecerá dentro del slot
  </ChildComponent>
</template>
```

ChildComponent.vue

```
<script setup>
defineProps({ name: String });

defineEmits(['greet']);
</script>

<template>
  <div>
    <h2>{{ name }}</h2>

    <slot />

    <button @click="$emit('greet')">
      Greet
    </button>
  </div>
</template>
```

04. Reutilización

ParentComponent.vue

```
<script setup>
import { ChildComponent } from 'child-component.vue'
function handleGreet(name) {
  alert('¡Hola ' + name + '!')
}
</script>

<template>
  <h1>Saludador</h1>

  <ChildComponent
    name="Dani"
    @greet="handleGreet('Dani')">
    >
      Esto aparecerá dentro del slot
  </ChildComponent>
</template>
```

ChildComponent.vue

```
<script setup>
defineProps({ name: String });

defineEmits(['greet']);
</script>

<template>
  <div>
    <h2>{{ name }}</h2>

    <slot />

    <button @click="$emit('greet')">
      Greet
    </button>
  </div>
</template>
```

```
<h1>Saludador</h1>
<div>
  <h2>Dani</h2>
  Esto aparecerá dentro del slot
  <button>Greet</button>
</div>
```

05. Directivas (I)

v-if y v-else

```
const showElement1 = ref(true);  
const showElement2 = ref(false);
```

```
<div v-if="showElement1">Element 1</div>  
  
<div v-if="showElement2">Element 2</div>  
<div v-else>Element 3</div>
```

05. Directivas (I)

v-if y v-else

```
const showElement1 = ref(true);  
const showElement2 = ref(false);
```

```
<div v-if="showElement1">Element 1</div>  
  
<div v-if="showElement2">Element 2</div>  
<div v-else>Element 3</div>
```

```
<div>Element 1</div>
```

```
<div>Element 3</div>
```

05. Directivas (II)

v-for

```
const elements = ref(['Dani', 'Ana', 'Juan']);
```

```
<div  
  v-for="(element, index) in elements"  
  :key="index"  
>  
  {{ index }}. {{ element }}  
</div>
```

05. Directivas (II)

v-for

```
const elements = ref(['Dani', 'Ana', 'Juan']);
```

```
<div  
  v-for="(element, index) in elements"  
  :key="index"  
>  
  {{ index }}. {{ element }}  
</div>
```

<div> 1. Dani </div>

<div> 2. Ana </div>

<div> 3. Juan </div>

06. Gestión de Estado



Pinia

Gestión global de “stores”

Sucesora de “Vuex”

Sencilla, ligera, y ... sustituible

07. ¿Por qué Vue?

Curva de aprendizaje más suave.

Vue es más intuitivo y fácil de aprender.

Su sintaxis es HTML, CSS y JS estándar en su mayoría.

No necesitas aprender TypeScript (como Angular) o JSX (como React).

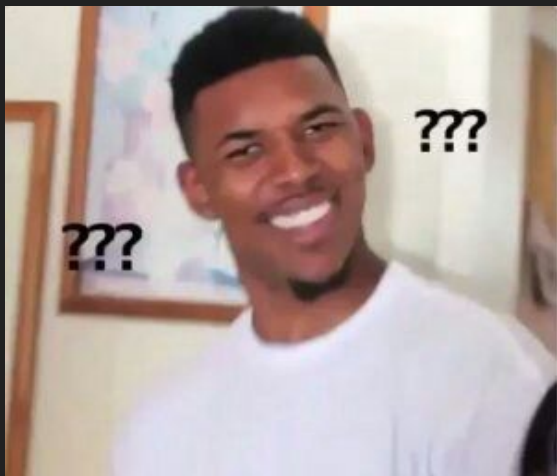
07. ¿Por qué Vue?

Documentación excelente y comunidad accesible.

La documentación oficial es clara, extensa y bien organizada.

Una comunidad muy activa y enfocada en hacer el framework accesible.

Centrados en mejorar el ecosistema, no solo Vue.



¿Dudas?

08. ¡Workshop!










Gastos

Buscar...

Saldo: **863€**

Concepto	Importe	Categoría	
Concepto...	Importe...		Enviar

Se está buscando:

 #1 Nómina de septiembre	1000€	 	1/9/2024
 #2 Gasolina	-50€	 	10/9/2024
 #3 Compra en Mercadona	-87€	 	20/9/2024

<https://play.vuejs.org/>