



Curso de
**Frameworks y
Librerías de
JavaScript**

JuanDC



ESCUELA DE



www.platzi.com/web

Websites vs. Webapps



LIVE

[Cursos](#) [Blog](#) [Foro](#) [Agenda](#) [TV](#) [Contáctanos](#)[Iniciar mi plan](#) ▾[Iniciar sesión](#)[Buscar en Platzi](#)

Fórmate online como profesional en tecnología



70% de los graduados de Platzi **duplican sus ingresos**

20% crean su propia empresa de tecnología o startup

Acelera tu carrera profesional:

Escribe tu correo electrónico

[ENTRA A PLATZI](#)



Al ingresar aceptas [Términos de Servicio](#) y [Políticas de privacidad](#)



LIVE

[Cursos](#) [Blog](#) [Foro](#) [Agenda](#) [TV](#) [Contáctanos](#)[Iniciar mi plan](#) ▾[Iniciar sesión](#)[Buscar en Platzi](#)

Fórmate online como profesional en **HTML** tecnología

70% de los graduados de Platzi dup

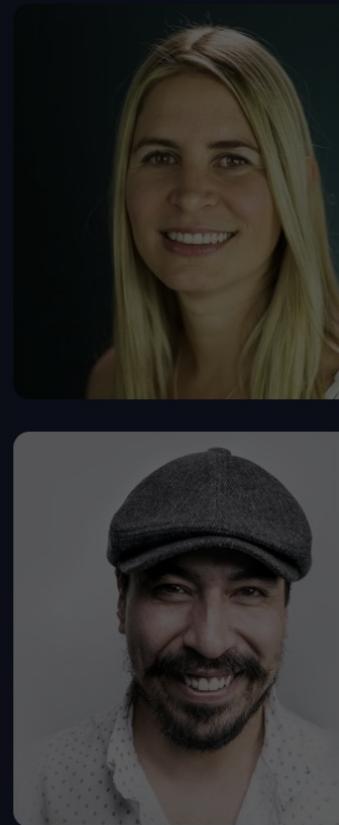
20% crean su propia empresa de tec

Acelera tu carrera profesional:

Escribe tu correo electrónico

[ENTRA A PLATZI](#)

Al ingresar aceptas Términos de Servicio y Políticas de privacidad



[Inicio](#) > [Blog](#) > Search

Entradas de juandc

[Nuevos](#)[+ Escribe en nuestro blog!](#)

[React Server Components es una revolución para el Desarrollo Web](#)

React Server Components nos ayuda a construir interfaces con buena UX de forma rápida, mantenible y por un bajo costo de desarrollo. Revoluciona el desarrollo...



juandc

🕒 hace 10 días

💬 3 comentarios



[Mejores sagas de cursos para aprender DevOps, Seguridad Informática y Criptomonedas](#)

Aprende cloud computing, devops y/o seguridad informática profesionalmente para aumentar tus ingresos en desarrollo de software a los mejores pagados en la...



juandc

🕒 el mes pasado

💬 12 comentarios



[Mejores sagas de cursos para aprender Desarrollo Web, Multiplataforma y Ciencia de Datos](#)

El desarrollo web abarca desde maquetación en HTML y CSS hasta optimización de bases de datos con SQL. Aprendiendo Data Science puedes componer sistemas para...



juandc

🕒 el mes pasado

💬 17 comentarios



[Mejores sagas de cursos para aprender Desarrollo de Software](#)

Las sagas son series de cursos para dominar una tecnología o habilidad en programación profesionalmente. Son impartidos por profesoras expertas en la industria...



juandc

🕒 el mes pasado

💬 15 comentarios

[Inicio](#) > [Blog](#) > Search

Entradas de juandc

Nuevos artículos en tu blog!

HTML

CSS

JS

Buscar

React Server Components es una revolución para el Desarrollo Web

5 artículos

112 likes

112 likes

81 likes

El desarrollo web abarca desde maquetación en HTML y CSS hasta optimización de bases de datos con SQL. Aprendiendo Data Science puedes componer sistemas para...

juandc el mes pasado 17 comentarios

Mejores sagas de cursos para aprender Desarrollo de Software

Las sagas son series de cursos para dominar una tecnología o habilidad en programación profesionalmente. Son impartidos por profesoras expertas en la industria...

juandc el mes pasado 15 comentarios

Continúa aprendiendo



Escuela de Inglés

6%

Básico

Curso de Estrategias para Aprender Inglés Online ✓

Curso de Inglés para Principiantes

Taller de Inglés para Principiantes

Curso de Inglés Básico: Fundamentos

Curso de Inglés Básico: Descripciones y Comparativos

Curso de Inglés Básico: Gramática

Curso de Inglés Básico: Conversación

Curso de Inglés Básico: Pronunciación

Curso de Inglés para Compras

Intermedio

Curso de Inglés Práctico: Fundamentos

Curso de Inglés Práctico: Gramática

Curso de Inglés Práctico: Conversación

Curso de Inglés Práctico: Pronunciación

Curso de Inglés Intermedio: Gramática

Render tree

Curso de Introducción a la Web: His...

Virtual reality: Transportation and giving directions

Curso de Inglés para Viajes

Platzi **LIVE** Founders By Platzi. [Entra ahora](#)

Aporte Curso Práctico de HTML y CSS / Chrome Dev Tools

Ernesto Briceño hace un año

La extensión en visual Code se llama “Live Server”

7 respuestas más

Kenji_290 hace una hora

Gracias

[ver más](#)

41 · 8 · Comentar

Pregunta Curso Básico de Photoshop / Menú, Paneles y Espacios de Trabajo

Magrey Oscarina Tousaintte Hernandez hace 7 meses

Cual es mejor Photoshop CC o CS6? tienen alguna diferencia?

3 respuestas más

Fernando Remache hace 2 horas

Buscar en Platzi



MARKETING

DISEÑO Y UX

CRECIMIENTO PROFESIONAL

Escuela de Finanzas Personales

15 Cursos

Escuela de Desarrollo Profesional

34 Cursos

Escuela de Inglés

35 Cursos

STARTUPS

DESARROLLO E INGENIERÍA

Desarrollo de Apps con React Native

13 Cursos

Desarrollo Backend con Go

7 Cursos

Desarrollo con WordPress

8 Cursos

Seguridad Informática

22 Cursos



Curso de Inteligencia Emocional

Instruido por: Carlos van Oosterzee

Básico 2 horas de contenido

[Ver la ruta de aprendizaje](#) ▾



Proyecto del curso

Resuelve conflictos en el ámbito laboral

Adquiere conocimientos y herramientas para la elaboración de una estrategia comunicativa que te permita gestionar y resolver de forma satisfactoria situaciones en tu ámbito laboral por medio de la inteligencia emocional.

[+ Agregar módulo](#)

[+ Agregar material](#)

Contenido del Curso

Tutoriales de estudiantes

Progreso del curso: 16/25 contenidos (44%)



[Continuar donde me quedé](#)

[Presenta el examen](#)

Profesionales, emociones e inteligencia



Bienvenida/o al curso



Nuevos

Más Votados

Sin Responder



liz6 Estudiante · [hace 5 días](#)

cómo puedo influir si estoy en remoto y los integrantes del miembro del equipo nunca nos hemos visto, no nos conocemos, sólo hemos hablad...



0



Andres Felipe Beltran Rojas Felipe Beltran Rojas

Estudiante · [hace 5 días](#)

Hola, Esto es lo que considero según mi criterio:
1.Muy de acuerdo 2. Bastante de acuerdo 3. Poco de acuerdo 4. Bastante de acuerdo 5. Mu...



0



liz6 Estudiante · [hace 7 días](#)

M motiva el dinero, bonos económicos eso estará mal?



3



daniela97 Estudiante · [hace 8 días](#)

Chicos como llevan ustedes su “Registro Profesional” ? ¿Qué límites han puesto dentro de su vida profesional para que no intervengan sus ...



3



Gloria Elizabeth Mancilla

· [hace 10](#)



1
2
3
4
5
6
7
8
9
10

Curso de Angular Forms: Creación y Optimización de Formularios Web

Nicolas Molina @nicobytes

Nicolas Molina Monroy

0:03 / 0:48 2x Auto (720p)

Descripción Reportar un problema ► Presentación del proyecto: t...

¿Por qué aprender Angular Forms?



Curso de Angular Forms: Creación y Optimización de Formularios Web

Aportes 10

Preguntas 1

Marcadores 0

Archivos y Enlaces 2

Más Votados

Nuevos

Sin Responder



Escribe aquí tu comentario

+ 2



Eduardo Antonio Vergara Sánchez Estudiante · hace 2 meses



4



Nicolas Molina · hace 2 meses

Con todo el power!



4



david bryan alcocer cañete Estudiante · hace 2 meses

Justo lo que necesitaba!



3



Nicolas Molina · hace 2 meses

Genial dale con toda vamos a aprender mucho sobre Angular Forms



3



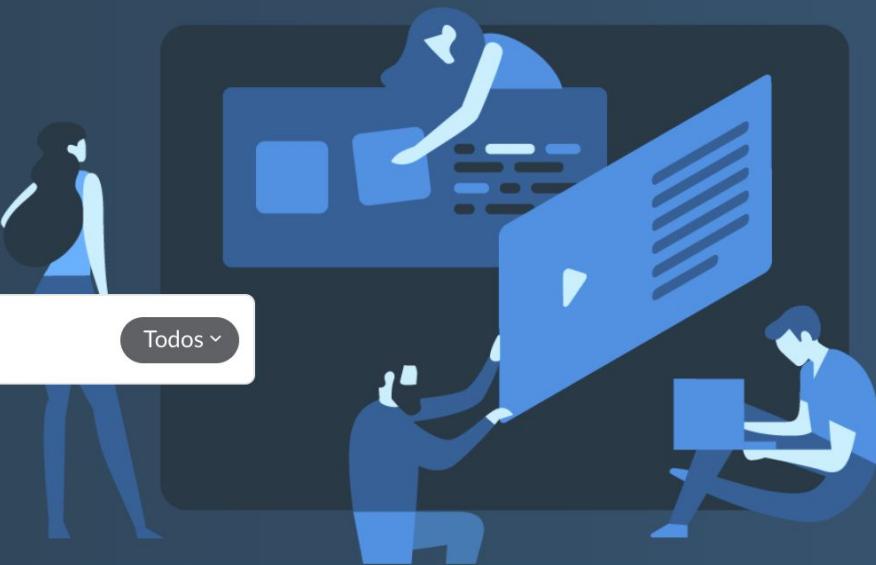
felixjesusmedina Estudiante · hace 2 meses

Foros de Platzi

Comparte y discute sobre temas de programación, diseño, marketing, video, startups y negocios.

Buscar una discusión

Todos ▾



+Iniciar una discusión + 5 R

Discusiones ordenadas por: Mejores ▾



¿Cómo es que seleccionan a las empresas para ser parte de la w3c y así determinar los nuevos estándares de la web?

alanya 50

0 hace 8 horas

Desarrollo e ingeniería



al tratar de trabajar con vim me ha salido esto sudo apt install vim sudo apt install vim-gtk3 sudo apt install vim-tiny sudo apt install...

eeg968 886

Todos

Marketing

Diseño y UX

Crecimiento Profesional

Startups

Desarrollo e Ingeniería

Negocios y Emprendimiento

Contenido Digital

Comunidad

+

+

+

+

+

+

+

+



Juan David Castro

The Eternal Student / Course Director in Platzi /
Computer Science Faculty / I love academic writing and
programming / #NuncaParesDeAprender



<https://platzi.com/blog/autores/juandc/>

101.920

Puntos

60

Preguntas

5.807

Respuestas

[Mis Cursos](#)[Mi Portafolio](#)

Tus cursos

Los materiales de cada curso equivalen a un 70% del total, mientras que el examen comprende el otro 30%. Puedes tomar el examen en cualquier momento cuando estés listo.



Curso de Javascript y JQuery 2014 93%

12/12 Materiales

aprobado

[Continua el curso](#)

Curso de Diseño de Circuitos Electrónicos 71%

Internet of Things

aprobado

[Continua el curso](#)

Aportes



1

[Writing E2E tests for a Next.js app with Cypress](#)

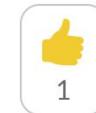
Curso de Next.js hace 4 meses



1

[ZEIT is now Vercel – Vercel](#)

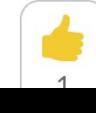
Curso de Deploy con Now.sh hace 7 meses



1

[Off-Page SEO: The Definitive Guide](#)

Curso de Buenas Prácticas de SEO hace 7 meses



1

[How to Create the Perfect Ending for Your Story - The Writing Connection](#)











Websites vs. Webapps

Componentes en la historia y evolución del desarrollo web

Fundamentos del desarrollo web
profesional

Componentes en la historia y evolución del desarrollo web

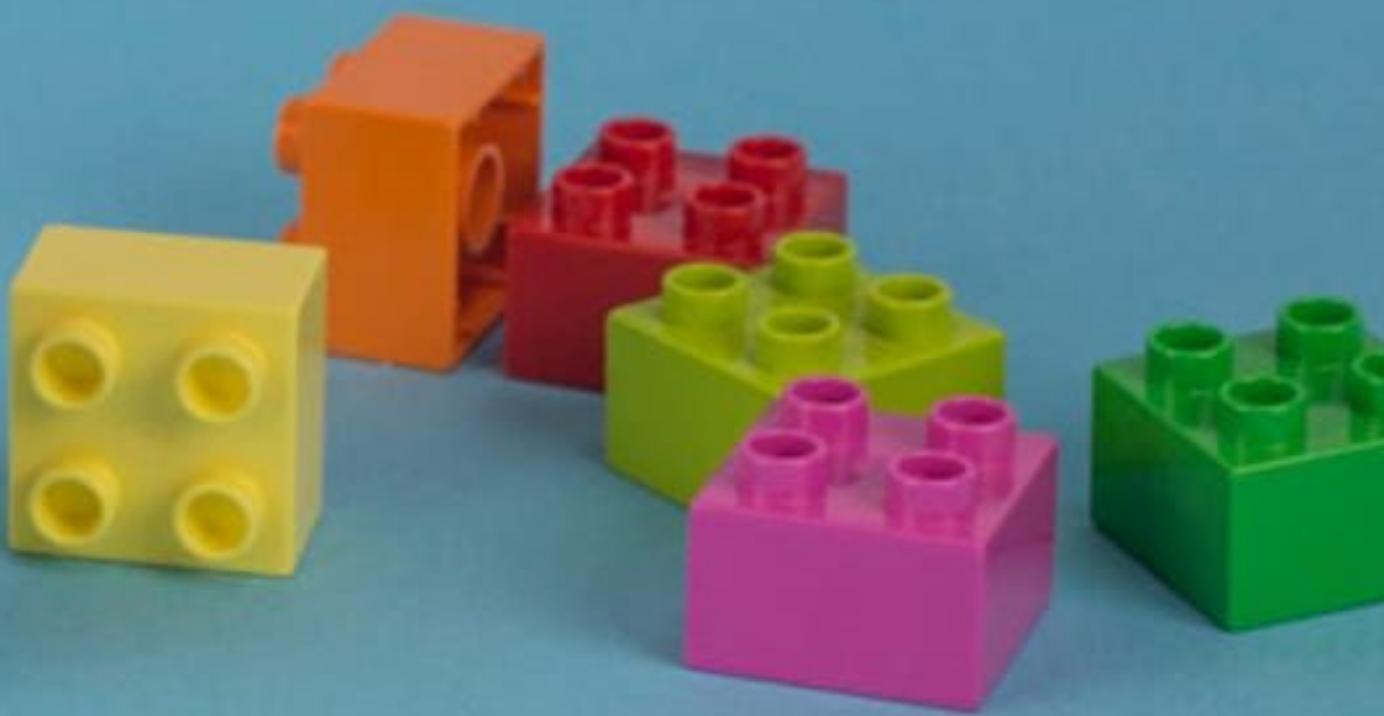
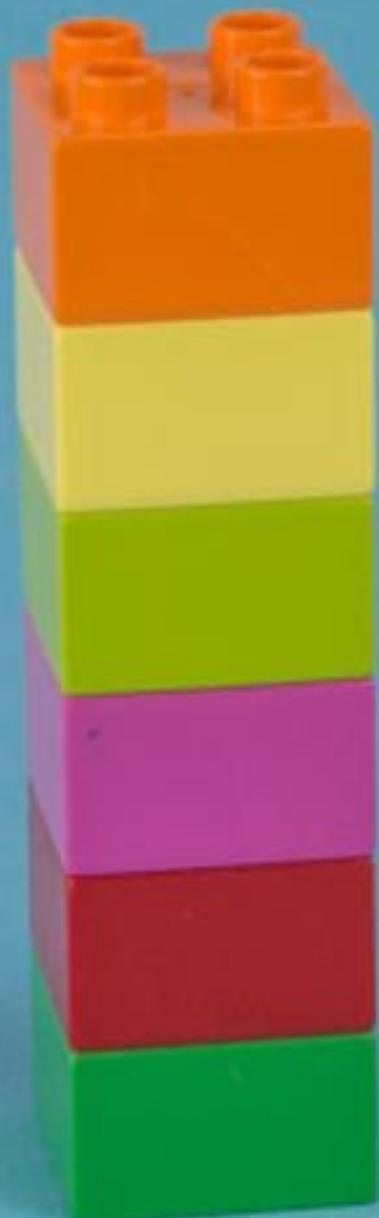
Fundamentos del desarrollo web
profesional



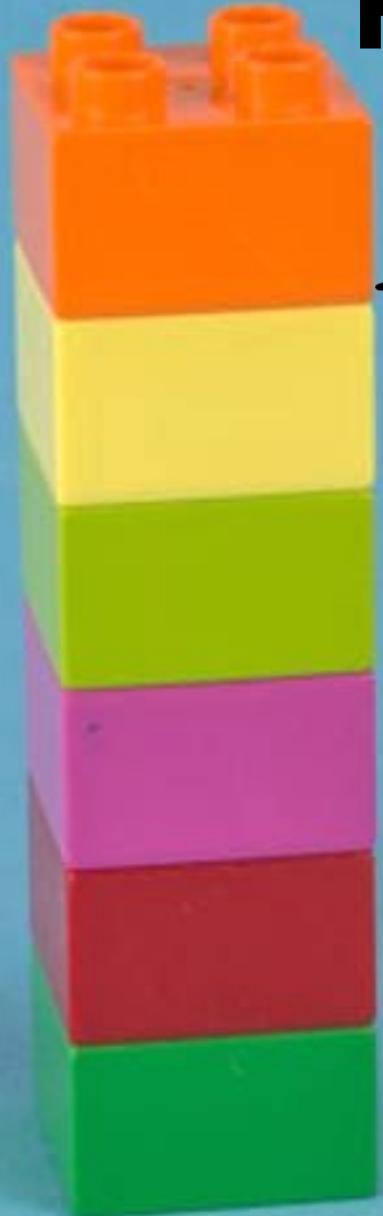








Formulario



Button



Label



Input



Formulario



Button

Composición



Label



Input



1

2

3

4

5

6











Componentes en la historia y evolución del desarrollo web

Fundamentos del desarrollo web
profesional



LIVE

Cursos Blog Foro Agenda TV Contáctanos

Iniciar mi plan ▾

Iniciar sesión

Buscar en Platzi





LIVE

Cursos

Blog

Foro

Agenda

TV

Contáctanos

Iniciar mi plan ▾

Iniciar sesión

Buscar en Platzi





LIVE

Cursos

Blog

Foro

Agenda

TV

Contáctanos

Iniciar mi plan ▾

Iniciar sesión

Buscar en Platzi





LIVE

Cursos

Blog

Foro

Agenda

TV

Contáctanos

Iniciar mi plan ▾

Iniciar sesión

Buscar en Platzi



<Button />

Iniciar sesión

```
<Button type="small" />
```

Iniciar sesión

```
<Button type="big" />
```

Comienza ahora

Regístrate a un curso nuevo

Comienza ahora

Toma tu primera clase

Iniciar sesión

Iniciar mi plan ▾

Reportar un problema

SalesButton

Type: primary o secondary

Size: small, big o ultrabig

Regístrate a un curso nuevo

Toma tu primera clase

Comienza ahora

Iniciar sesión

StudentButton

Icon: componente con el ícono que queremos



Continuar donde me quedé



Presenta el examen

Reportar un problema

Dropdown:

ClickArea: siempre visible

HiddenArea: lo que aparece y desaparece

Iniciar mi plan ▾



102.512 pts ▾





Fórmate online como profesional en tecnología

70% de los graduados de Platzi **duplican sus ingresos**

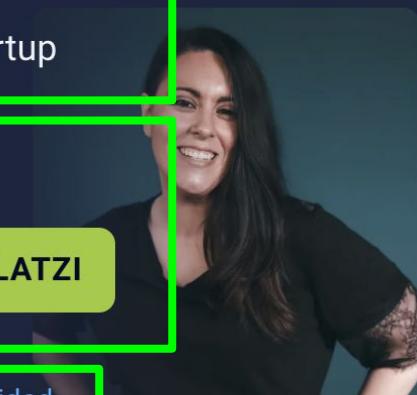
20% crean su propia empresa de tecnología o startup

Acelera tu carrera profesional:

Escribe tu correo electrónico

ENTRA A PLATZI

Al ingresar aceptas [Términos de Servicio](#) y [Políticas de privacidad](#)



The image shows the homepage of the Platzi website. At the top, there's a navigation bar with links for 'LIVE', 'Cursos', 'Blog', 'Foro', 'Agenda', 'TV', and 'Contáctanos'. To the right of these are buttons for 'Iniciar mi plan' (with a dropdown arrow), 'Iniciar sesión', and a search bar. Below the navigation is a large banner with the text 'Fórmate online como profesional en tecnología'. Underneath this, there are two rows of three student portraits each. The first row shows a woman in a red and orange scarf, a woman with long blonde hair, and a woman with blonde hair. The second row shows a woman with dark hair, a woman with long dark hair, and a man with a beard wearing a cap. On the left side of the page, there are several statistics: '0% de los graduados de Platzi duplican sus ingresos', '0% crean su propia empresa de tecnología o startup', and 'celera tu carrera profesional:'. Below these are input fields for 'Escribe tu correo electrónico' and a 'ENTRA A PLATZI' button. At the bottom, there's a link to 'ingresar aceptas Términos de Servicio y Políticas de privacidad'.

The screenshot shows the Platzi blog homepage. At the top, there's a navigation bar with links for 'Clases', 'Blog', 'Foro', 'Agenda', 'TV', 'Ver Dashboard', and a search bar. Below the navigation, there's a sidebar with 'Blog' selected, followed by 'Firebase', 'Escuela de JavaScript', 'Periodismo', 'Diseño Gráfico', 'Desarrollo Multiplataforma con Xamarin', and a 'Más currículas' link. The main content area has a heading 'Entradas de Juan dc'. It lists several posts:

- React Server Components es una revolución para el Desarrollo Web**
55 Juan dc hace 10 días 3 comentarios
- Mejores sagas de cursos para aprender DevOps, Seguridad Informática y Criptomonedas**
112 Juan dc el mes pasado 12 comentarios
- Aprende cloud computing, devops y/o seguridad informática profesionalmente para aumentar tus ingresos en desarrollo de software a los mejores pagados en la..**
81 Juan dc el mes pasado 17 comentarios
- Mejores sagas de cursos para aprender Desarrollo Web, Multiplataforma y Ciencia de Datos**
81 Juan dc el mes pasado 17 comentarios
- Mejores sagas de cursos para aprender Desarrollo de Software**
88 Las sagas son series de cursos para dominar una tecnología o habilidad en programación profesionalmente. Son impartidos por profesoras expertas en la industria..
Juan dc el mes pasado 15 comentarios

The screenshot shows a search result for "Render tree" on the Plottzi platform. The top navigation bar includes links for Clases, Blog, Foro, Agenda, TV, Ver Dashboard, and a user icon. A search bar at the top right contains the query "Render tree". Below the search bar, there are several filter dropdowns: MARKETING, DISEÑO UX, and CRECIEMIENTO PROFESIONAL (which is highlighted). The main content area displays a card for a video titled "Render tree" by "Plottzi LIVE", which is part of the "Founders by Plottzi" series. The card includes a thumbnail of a person speaking, a play button, and a link to "Ver más". Below this, another card for "Virtual reality: Transportation and giving directions" by "Plottzi LIVE" is shown. Further down, a card for "Curso de Inglés para Peques" by "Plottzi LIVE" is displayed. On the left side, there is a sidebar with a profile picture and the text "Escola de Inglés". A vertical sidebar on the right lists various categories: Escuela de Finanzas Personales, Escuela de Desarrollo Profesional, Escuela de Inglés, Desarrollo de Apps con React Native, Desarrollo Backend con Go, Desarrollo con WordPress, and Seguridad Informática. Each category has a corresponding icon and a "Ver más" link.

The screenshot shows the Platzi learning platform interface. At the top, there's a navigation bar with links for 'Clases', 'Blog', 'Foro', 'Agenda', 'TV', 'Ver Dashboard', and a search bar. Below the navigation, a course card for 'Curso de Inteligencia Emocional' is displayed, featuring a profile picture of the instructor, Carlos van Oosterzee. The course title is 'Curso de Inteligencia Emocional' and it's described as being taught by 'Instruido por: Carlos van Oosterzee'. Below the course title, it says 'Básico' and '2 horas de contenido'. To the right of the course card, there's a sidebar with buttons for 'Nuevos', 'Más Votados', and 'Sin Responder'. A progress bar at the bottom of the course card indicates 'Progreso del curso: 16/25 contenidos (44%)'. Below the course card, there are two buttons: 'Continuar donde me quedé' and 'Presenta el examen'. The main content area below the course card shows a section titled 'Resuélve conflictos en el ámbito laboral' with a brief description and an image of two people in a professional setting. At the bottom of the page, there are sections for 'Contenido del Curso' and 'Tutoriales de estudiantes'.

The screenshot shows a video player interface for a course titled "Curso de Angular Forms: Creación y Optimización de Formularios Web" by Nicolas Molina (@nicobties). The video thumbnail features a portrait of Nicolas Molina. Below the thumbnail, his name is displayed as "Nicolas Molina Monroy". The video progress bar shows 0:03 / 0:48. A "Reportar un problema" button is visible. The video is set to play automatically (Auto (2x)). The main content area contains the course title and a large question "¿Por qué aprender Angular Forms?". To the right, there is a sidebar for the course, including sections for "Aportes", "Preguntas", "Marcadores", and "Archivos y Enlaces". Below these are buttons for "Más Votados", "Nuevos", and "Sin Responder". The sidebar also displays comments from users like Eduardo Antonio Vergara Sánchez and Nicolas Molina, along with their interaction counts (likes, replies).

The screenshot shows the Platzi forums homepage. The header features the Platzi logo and the text "Foros de Platzi". Below the header, there's a subheader: "Comparte y discute sobre temas de programación, diseño, marketing, video, startups y negocios." To the right of the subheader is a stylized graphic of a person standing next to a laptop screen. A search bar with the placeholder "Buscar una discusión" and a "Todos" button are also visible. The main content area displays a discussion titled "+Iniciar una discusión + 5" with a small icon. To the right of this, it says "Discusiones ordenadas por: Mejores". Below this, a post by a user named "alanya" is shown, asking about selecting companies for w3c standards. The post has 0 likes and was made 8 hours ago. A green button labeled "Desarrollo e Ingeniería" is to the right. Another post at the bottom is partially visible, mentioning vim and sudo apt install. The sidebar on the right lists various forum categories: Todos, Marketing, Diseño y Creatividad, Crecimiento, Startups, Desarrollador, Negocios, Contenidos, and Comunidad.

```
const Home = props => ( {  
  <>  
  <Menu user={props.user} />  
  
  <Hero>  
    <Title>Fórmate online...</Title>  
    <Description text="70%" />  
    <Description text="20%" />  
    <RegisterForm />  
    <Description text="Al ingresar..." />  
  </Hero>  
  </>  
} );
```

```
const Menu = props => ( {  
  <Nav>  
    <Logo badge={<Live />} />  
  
    <Links list={props.links} />  
  
    { !props.user ? (  
      <LoginButtons />  
    ) : (  
      <UserPreview {...props.user} />  
    ) }  
  </Nav>  
} );
```

```
const Button = props => ( {  
  <button  
    className={  
      `btn ${props.size}`  
    }  
  />  
  {props.text}  
  </button>  
} ) ;
```

```
const Button = props => ( {  
  <button  
    className={`btn ${props.size}`}  
  >  
  </button>  
} );
```

```
const BigButton = props => ( {  
  <Button size="big" {...props} />  
} );
```

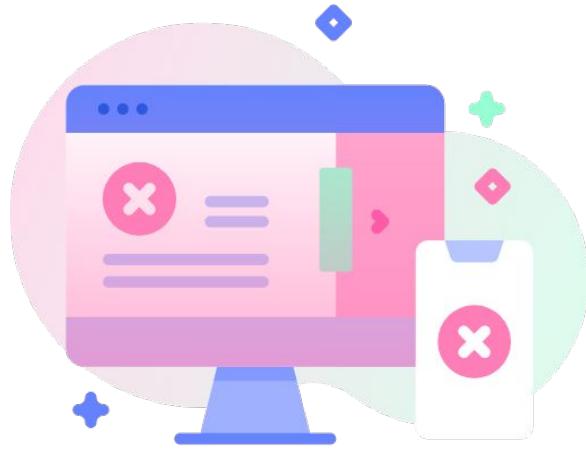
```
const SmallButton = props => ( {  
  <Button size="small" {...props} />  
} );
```

Reactividad, estado y ciclo de vida

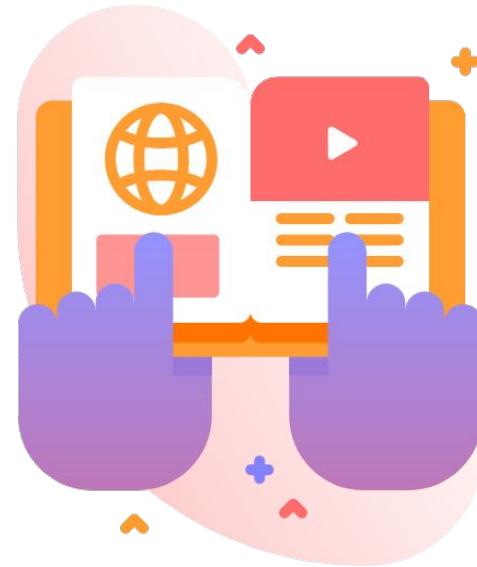
Fundamentos del desarrollo web profesional

Reactividad

Responsive



Resiliente



Escalable

Message Driven

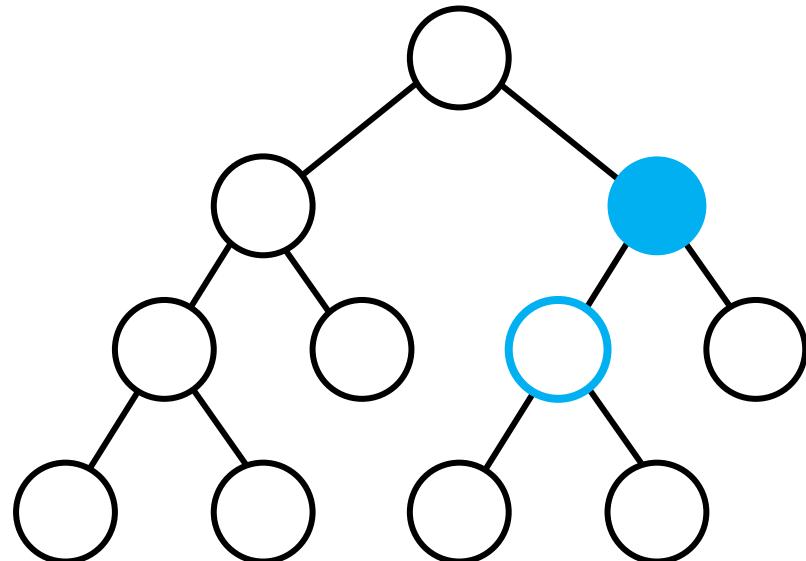


Estado

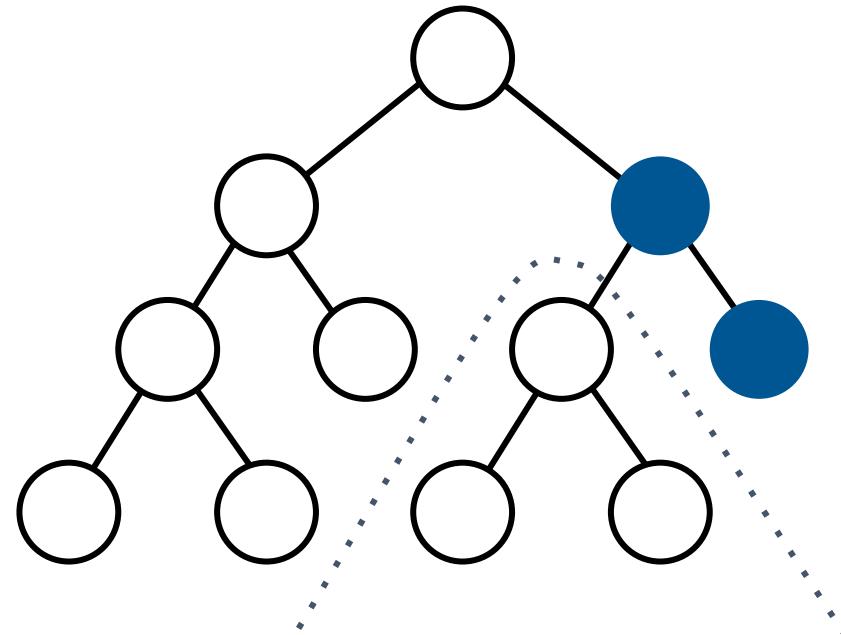
Render

Virtual DOM

Dirty

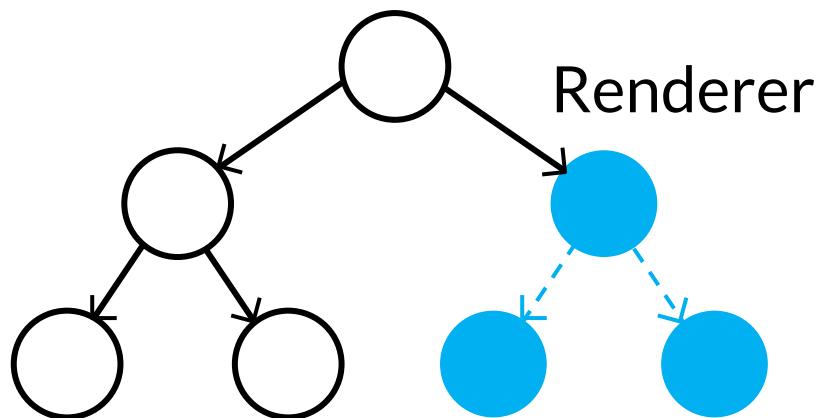


Re-rendered

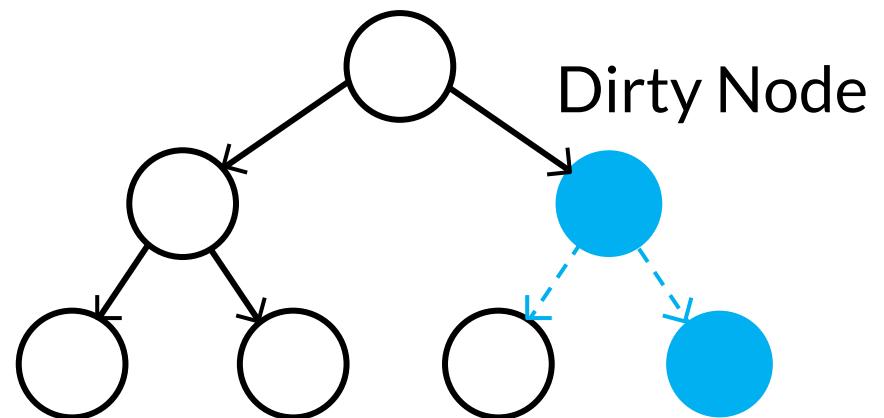


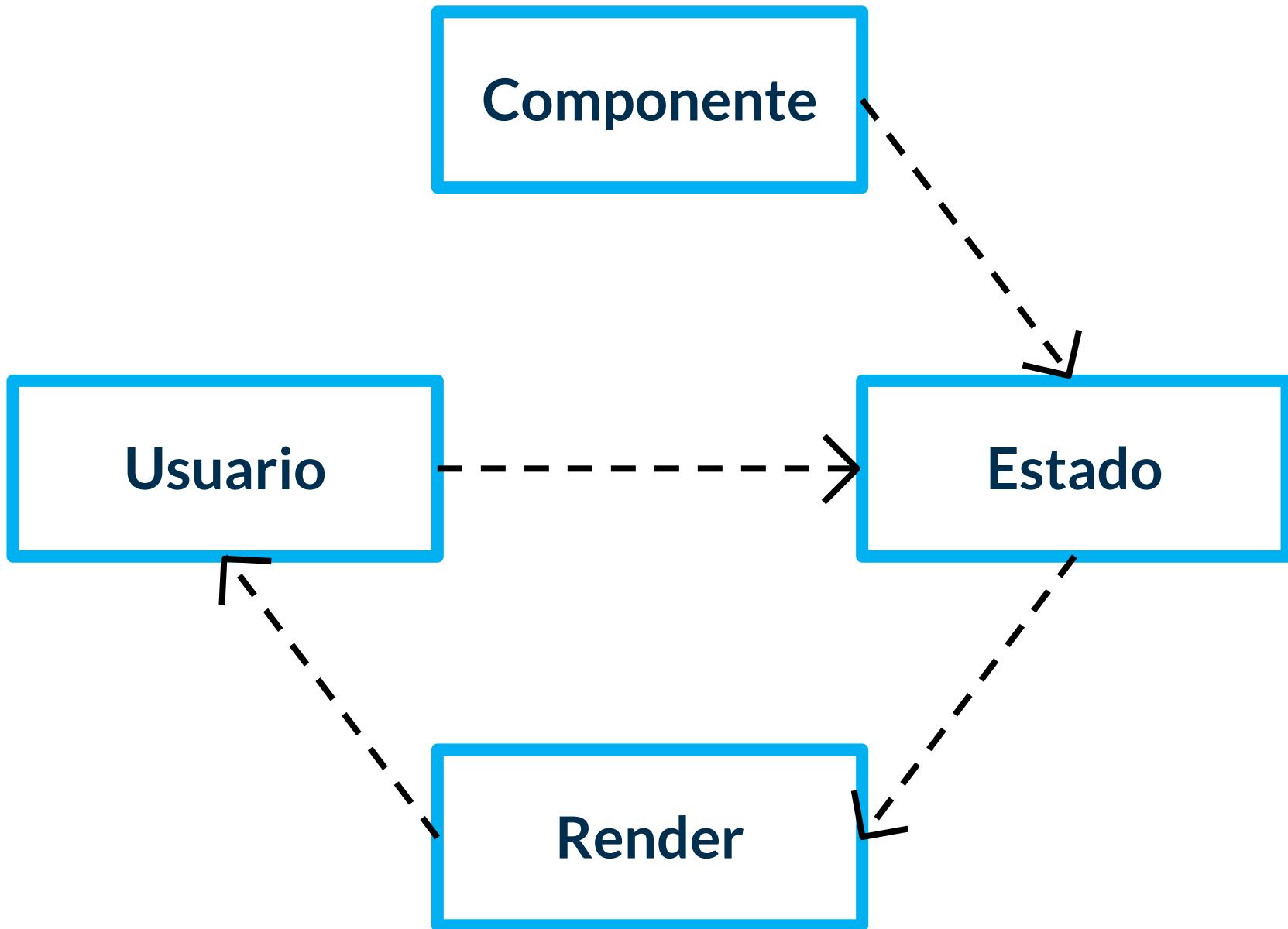
No Virtual DOM

Real DOM



Virtual DOM





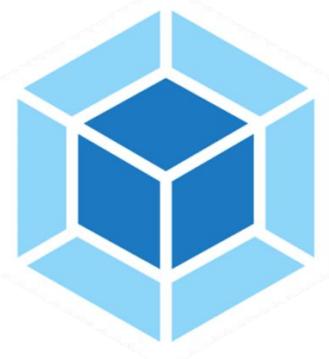
Librerías vs. Frameworks

Fundamentos del desarrollo web profesional

Ecosistema de frameworks y librerías JavaScript

Fundamentos del desarrollo web profesional

Empaquetadores



Webpack



Parcel



Rollup

A meme image featuring Joaquin Phoenix as the Joker. He is leaning out of the open driver-side door of a yellow taxi. He is wearing his signature green face paint, red lips, and a multi-colored suit (red, orange, and black). He has a wide, manic grin. In the background, there's a building with an "EXIT" sign above a doorway. The overall tone is dark and chaotic.

Complains on Twitter about how bad
webpack is

Trying to write your own bundler

"Compiladores"

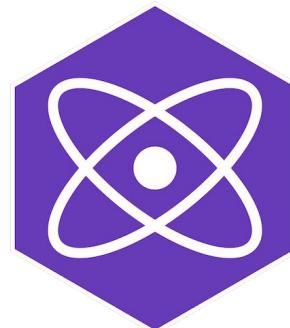
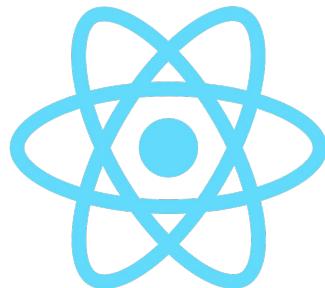


Babel



TypeScript

Herramientas para UI



CSS y preprocesadores

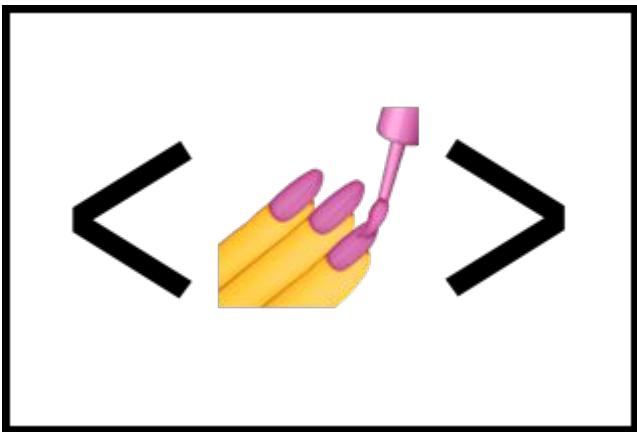


CSS



Sass, Less, Stylus,
PostCSS...

CSS-in-JS



Styled Components



Emotion

Routers

- React Router
- Vue Router
- Svelte Router
- LitElement Router
- Whatever Router

Frameworks (*todo en uno*)



Angular

Entornos de desarrollo (*todo en uno*)

- Create React App
- Vue CLI
- Svelte CLI
- Polymer CLI
- Whatever CLI

Manejo del estado



Redux



XState

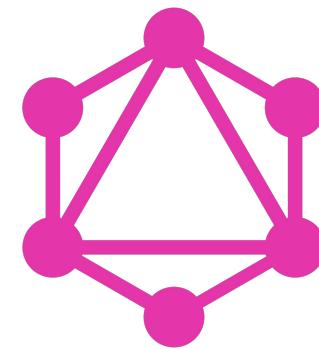


MobX

Consulta de datos

{ REST:API }

API REST



GraphQL

¿Cómo construyeron React.js?

Contexto y funcionamiento de los
Frameworks JavaScript

```
var sales = <sales vendor="John">
    <item type="peas" price="4" quantity="6" />
    <item type="carrot" price="3" quantity="10" />
    <item type="chips" price="5" quantity="3" />
</sales>

alert( sales.item.@type == "carrot" ).@quantity );
alert( sales.@vendor );
for each( var price in sales..@price ) {
    alert( price );
}
delete sales.item[0];
sales.item += <item type="oranges" price="4" />;
sales.item.(@type == "oranges").@quantity = 4;
```

```
$post =  
  <div class="post">  
    <h2>{$post}</h2>  
    <p><span>Hey there.</span></p>  
    <a href={$post->getAttribute("url")}>  
      {$post->getAttribute("title")}</a>  
    </div>;  
  
$list = <ul />;  
$items = ...;  
  
foreach ($items as $item) {  
  $list->appendChild(<li>{$item}</li>);  
}  
  
$list->appendChild(<li>{$post}</li>);
```

¿Cómo construyeron React.js?

Contexto y funcionamiento
de los Frameworks JavaScript

React

- Declarativo
- Basado en Componentes
- Multiplataforma

```
}); then response => {
  setProgress('finished')
}

};

return (
  <div className={progress}>
    <span className="load">
      <button className="cancel">
        <span className="cancel">
          </button>
        <span className="percent">
          </div>
    </span>
  </div>
);
```

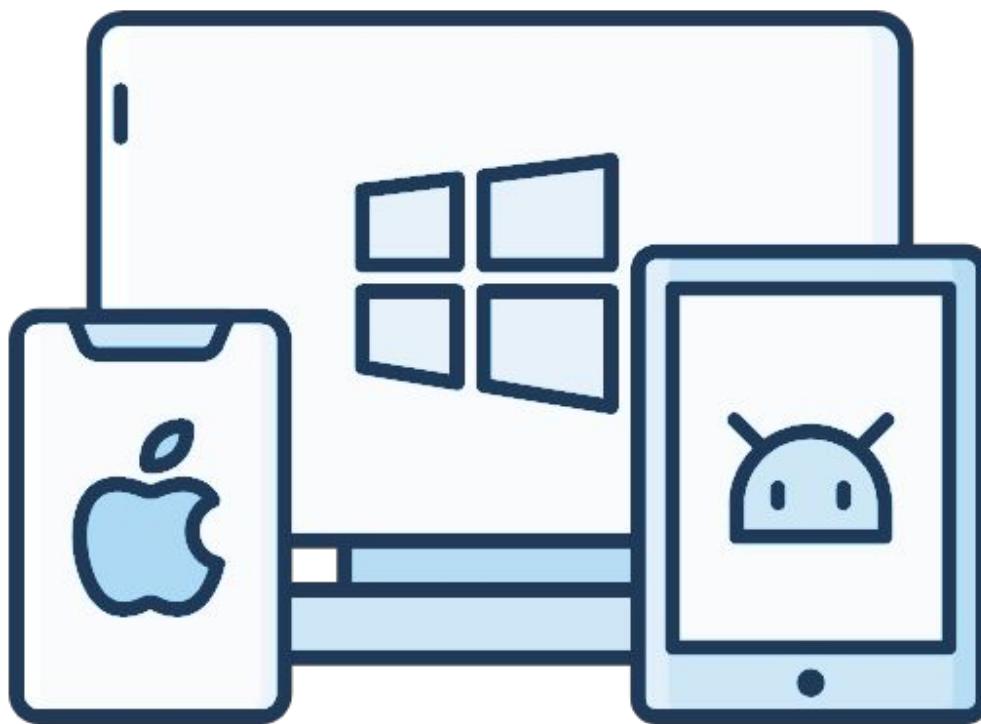
Declarative

```
11
12 ReactDOM.render((
13   <BrowserRouter>
14     <Switch>
15       <Route path="/login" component={Login} />
16       <ProtectedRoute exact={true} path="/" component={Home} />
17       <ProtectedRoute path="/settings" component={Settings} />
18       <ProtectedRoute component={Dashboard} />
19     </Switch>
20   </BrowserRouter>
21 ), document.getElementById('root'));
```

Component-Based



Learn Once, Write Anywhere



```
class Persona extends React.Component {  
  render() {  
    return (  
      <div>  
        Este nombre de esta  
        persona es {this.props.name}  
      </div>  
    );  
  }  
}
```

```
class Persona extends React.Component {  
  render() {  
    return React.createElement(  
      "div",  
      null,  
      "Este nombre de esta persona es",  
      this.props.name  
    );  
  }  
}
```

```
class Persona extends React.Component {  
  render() {  
    return (  
      <div>  
        Este nombre de esta  
        persona es {this.props.name}  
      </div>  
    );  
  }  
}
```

```
ReactDOM.render(  
  <Persona name="Homero" />,  
  document.getElementById('root')  
) ;
```

```
ReactDOM.render(  
  <Persona name="Enrique" />,  
  document.getElementById('root')  
) ;
```

```
class Click extends React.Component {  
  state = { clicks: 0 };  
  render() {  
    return (  
      <div>  
        <p>Clicks: {this.state.clicks}</p>  
        <button  
          onClick={() => this.setState(  
            ({clicks}) => ({clicks:clicks+1}))  
          }>Click</p>  
      </div>  
    );  
  }  
}
```

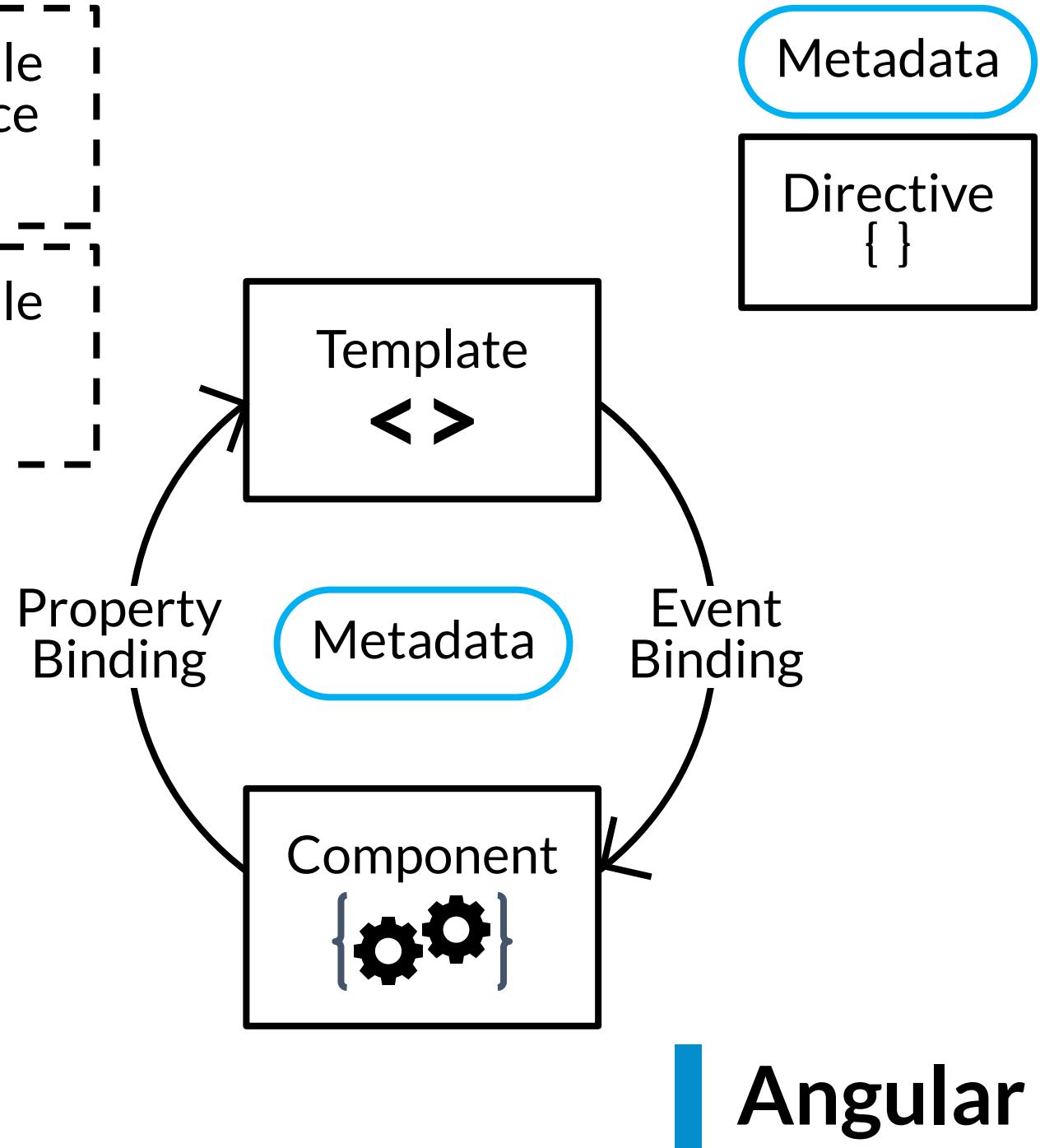
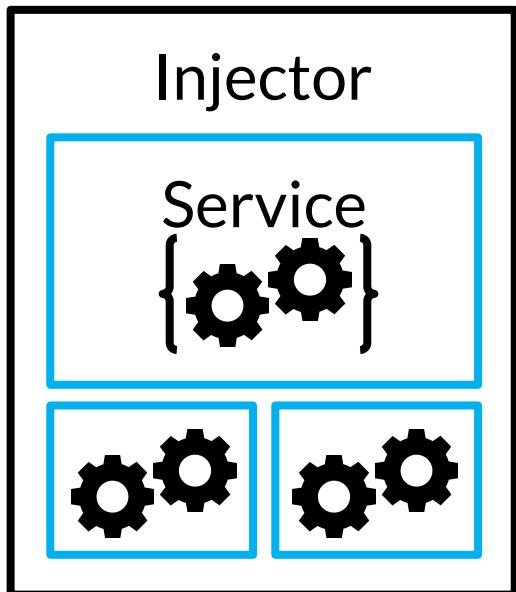
```
function Click() {  
  const [clicks, setClicks] = React.useState(0);  
  
  return (  
    <div>  
      <p>Clicks: {clicks}</p>  
      <button  
        onClick={  
          () => setClicks(count + 1)  
        }  
        >Click me</button>  
    </div>  
  );  
}
```

Demo con React

¿Cómo construyeron Angular?

Contexto y funcionamiento
de los Frameworks JavaScript

Module Component	Module Service
{ }	{ }
Module Value	Module Fn
3.1415	λ



Incremental DOM y Angular Ivy vs. Virtual DOM

```
▶<style>...</style>
▶<nav class="navbar navbar-expand-md navbar-dark bg-dark">...</nav>
▼<main class="container">
  ▼<todo-form>
    ▶<style>...</style>
    ▶<div class="card todo-form">...</div>
  <hr>
  ▼<todo-list ref="list">
    ▶<style>...</style>
    <h2>Tasks:</h2>
    ▼<ul ref="todos" class="list-group">
      ▶<todo-task ref="task-1517176192142" id="task-1517176192142">
      ...</todo-task> == $0
      ▶<todo-task ref="task-1517176320397" id="task-1517176320397">
      ...</todo-task>
      ▶<todo-task ref="task-1517176329096" id="task-1517176329096">
      ...</todo-task>
      ▶<todo-task ref="task-1517176334849" id="task-1517176334849">
      ...</todo-task>
    </ul>
  </todo-list>
</main>
</todo-application>
</div>
<script src="b+>
```

```
::before {
  box-sizing: bo
  box;
}

Inherited from ul.li
ul, user agent st
menu
• dir {
  display: block;
  list-style-type:
  -webkit-margin-t
  1em;
  -webkit-margin-a
  1em;
  -webkit-margin-s
  0px;
  -webkit-margin-e
  0px;
  -webkit-padding-s
  40px;
}

Inherited from div/app
body { _reboot.s
margin: 0;
```

Demo con Angular

¿Cómo construyeron "el cloncito" Vue.js?

Contexto y funcionamiento de los
Frameworks JavaScript

El framework progresivo

Escalable, pero flexible

Reactividad

Virtual DOM

+ HTML al principio
+ JS cuando creces

"El mejor
performance"

Demo con Vue

¿Cómo construyeron Svelte? ¿React es realmente reactivo?

Contexto y funcionamiento de los Frameworks JavaScript

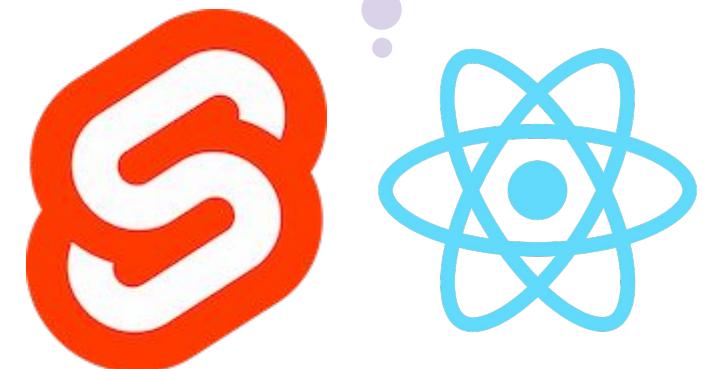


Eres débil,
te falta
reactividad



Los amo a
los dos,
chicos <3

Awww





Svelte para
sitios web



Svelte para
aplicaciones web

Arquitectura de Svelte





Rich-Harris commented on Dec 12, 2017

Member

...

Yay! This could make for a good blog post. I have no idea when I'd get round to it, but I like the idea.

Let me try and do the short version for the bullet points you raised:

Code generation and static analysis

As with any compiler, the first step is to generate an abstract syntax tree (AST). Svelte actually uses three parsers — its own internal one, for HTML and `<{(tags)}>`, Acorn for JS, and css-tree for CSS. Put those together, and you get a tree-like data structure that you can 'traverse' to find out things about the component (such as which data properties it expects, what methods it declares, what lifecycle hooks it has and so on.)

Once we've figured out the shape of the component, we can move on to the code generation step. Each component has a `main_fragment`, which is what all the markup represents — this gets created by the `create_main_fragment` block. As we traverse the AST and find things inside the component (like elements and text nodes), we add code to the block's methods using an internal utility called `CodeBuilder` which makes it slightly easier to combine different statements.

Sometimes, we'll encounter `if` blocks and `each` blocks, etc — these contain their own fragments, so we create a new block for each of those. Each block has its own methods for creating, updating and destroying the DOM, which allows Svelte to work with the DOM directly rather than incurring the overhead of a virtual DOM.

At the end of that process, we basically just concatenate all those blocks and add some boilerplate. There's a little sleight-of-hand involved, for the sake of debugging — when we're using code that the component author actually wrote (as opposed to stuff the compiler creates from scratch) we don't just copy the code, we insert a marker like this: `[-42-46-]`. That means 'insert the characters between 42 and 46', which allows us to record the source location to generate a sourcemap from the finished code.

Role of TypeScript

TypeScript is a purely internal choice, and not something that users of Svelte need to know about at all (though we may provide some form of TypeScript-in-components support in the future). I was sceptical for a long time, but I've become a huge fan. It makes for a much nicer development experience. For example, you start typing the first few characters of a method or property, and it will offer to autocomplete the rest. Or if you're not sure what the arguments to a function or method should be, just hover over the name and it'll look them up for you. I tend to be a fairly careless programmer, and TypeScript means I write way fewer bugs (and when I do write bugs, they're much easier to find).

Downsides: there's quite a lot of work updating an existing project to use TypeScript, and it complains if your dependencies aren't also written in TS (or at least have type definitions available). But overall I'm a happy customer, and I don't think it's deterred any would-be contributors (though that's obviously unknowable).

Motivation

Partly it's just the same itch that I think most programmers share — solving tricky problems and whittling something down to the most efficient possible implementation is a satisfying thing to do. And there are less savory motivations, like the ego-inflating buzz you get from seeing people using your work or borrowing your ideas.

But it's also because of my job. I work as a graphics editor for the New York Times, and before that the Guardian, and as everyone knows news websites tend to be full of stuff that slows your browser down — analytics, ads, social media crap, comment widgets, etc (the NYT and the Graan are among the lesser offenders, but the entire industry needs to improve on this front). Because of that, loading a framework like React or Vue for a simple interactive embedded in an article just isn't a good option. We need tools that allow us to build extremely quickly, and not worry about performance, app size, CSS leakage, and all that stuff. And there's a constant tension between the desire to reuse code and the fact that every news graphic is unique, with unique requirements. A declarative component-driven approach is helpful in that context.

All devs have these issues to some degree, but I believe they're particularly acute in the news biz, where deadlines are often measured in hours (no time for optimization!), the browser environment is unusually hostile, and programming might only be about 30% of the job. So we're probably even more motivated than most devs to find good solutions. (I don't think it's a coincidence that D3, Underscore, Backbone and CoffeeScript all came out of the news business, just to list projects by former members of the NYT graphics desk.)

Roadmap

The big priority for me at the moment is Sapper. The idea is to have a first-rate way to build Node apps using Svelte. It's inspired by Next.js, but departs from it in a few important ways. It's a very early work-in-progress but I'm very excited about — the early results are very encouraging.

Beyond that, there are a few things I want to work on:

- Transitions are pretty good in Svelte already, but they could get a lot better — more control over timing, methods of transitioning between nodes, and so on
- As custom element support improves, we'll need to spend more time making sure that Svelte-generated custom elements are rock solid
- There are some syntax improvements we could make. For example I'm not sure you need both curly braces either side of a `<{(tag)}>` — JSX seems to do fine with just one.
- Devtools and editor plugins!
- The next frontier in app performance, I believe, is whole-app optimisation. Right now, if you create a component that has a `<{(foo)}>` tag, then Svelte has to generate code that updates the tag if the value of `foo` changes. But if it could see that the only time that property is set is when a parent component does `<{thing foo='bar'}>`, then it could skip that — less code, and even less to do when it's checking for state changes. There are lots of similar situations where Svelte could generate better code if only it knew how components were being consumed (similar to how tree-shaking allows a module to skip some of its exports if it knows they're unused). That's a tricky problem that the current generation of bundlers aren't really equipped to deal with.
- Glimmer are doing some interesting work around generating binary bytecode representations of templates. I'm waiting to see the numbers, but there are definitely some theoretical wins from doing that instead of generating JavaScript. Svelte could have a 'bytecode mode' that switched to bytecode if your app was large enough to warrant the overhead of the interpreter that you need to ship in order to use it. (This is the sort of thing that Sapper could even do automatically, based on certain heuristics.)

In short, lots to do!

Sorry this answer got a bit long, I didn't have time to write a shorter one 😊

74 18

Demo con Svelte

—

¿Qué es CSS-in-JS?
¿Reemplaza los estilos
normales con CSS?

Integración de herramientas
para CSS-in-JS

CSS normal

```
5 .padding {  
6   padding: 0.5em;  
7 }  
17 .padding-H-05em {  
18   padding-left: 0.5em;  
19   padding-right: 0.5em;  
20 }  
21 .padding-V-05em {  
22   padding-top: 0.5em;  
23   padding-bottom: 0.5em;
```

Preprocesadores y postprocesadores de CSS

```
@keyframes ripple {
  from {
    width: .1%;
    height: .1%;
    opacity: 1;
  }
  to {
    width: 100%;
```

Componentes y CSS-in-JS

```
2   ${scope.tags.map(tag => `  .  
3     ${() => { tag.classes = (tag.classes || [])  
4       .push(tag.name.matches('js') ? 'tag-blue' : "")  
5     }}()  
6     <a href="${tag.link}" class="${tag.classes.join(" ")}>${tag.name}</a>  
7   `).join('')}</div>;  
8
```

article.html.js x

```
1 module.exports = (scope) => `<article>  
2   <header>  
3     <h1><a href="${scope.link}">${scope.title}</a></h1>  
4   </header>  
5   ${require('./tags.html.js')(scope)}  
6   <div>  
7     ${scope.body}  
8   </div>  
9 </article>;  
0
```

```
h2 {  
    color: white;  
    font-size: 60px;  
}
```

```
const Title = styled.h2`  
  color: white;  
`;
```

```
const Title = styled.h2`  
  color: white;  
`;
```

```
<Title>¡Título!</Title>  
<Title>¡Título 2!</Title>
```

```
const Title = styled.h2`  
  color: ${props =>  
    props.error  
      ? 'red'  
      : 'white'  
  } };  
`;
```

```
const Title = styled.h2`  
  color: ${props =>  
    props.error  
      ? 'red'  
      : 'white'  
  } };  
  
<Title>¡Título blanco!</Title>  
<Title error>¡Título rojo!</Title>
```

```
const Title = styled.h2`  
color: ${props =>  
  props.theme.mode === 'student'  
    ? 'green'  
    : 'blue'  
} ;  
`;
```

```
const Title = styled.h2`  
  color: ${props =>  
    props.theme.mode === 'student'  
      ? 'green'  
      : 'blue'  
  } ;  
`;  
  
<Title>¡Título!</Title>
```

“

**Los estilos globales no
son una desventaja de
CSS-in-JS**

”

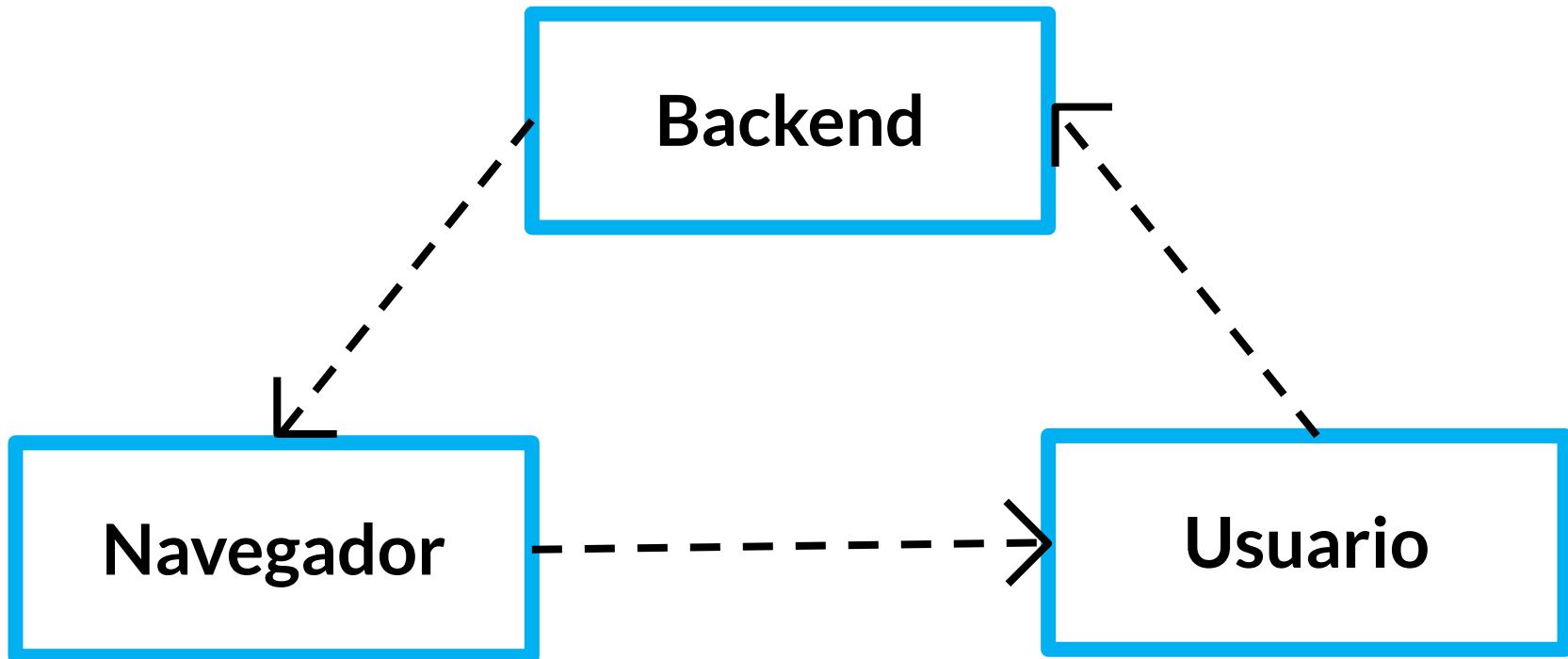
Cualquier persona que ha usado CSS-in-JS

Demo con CSS

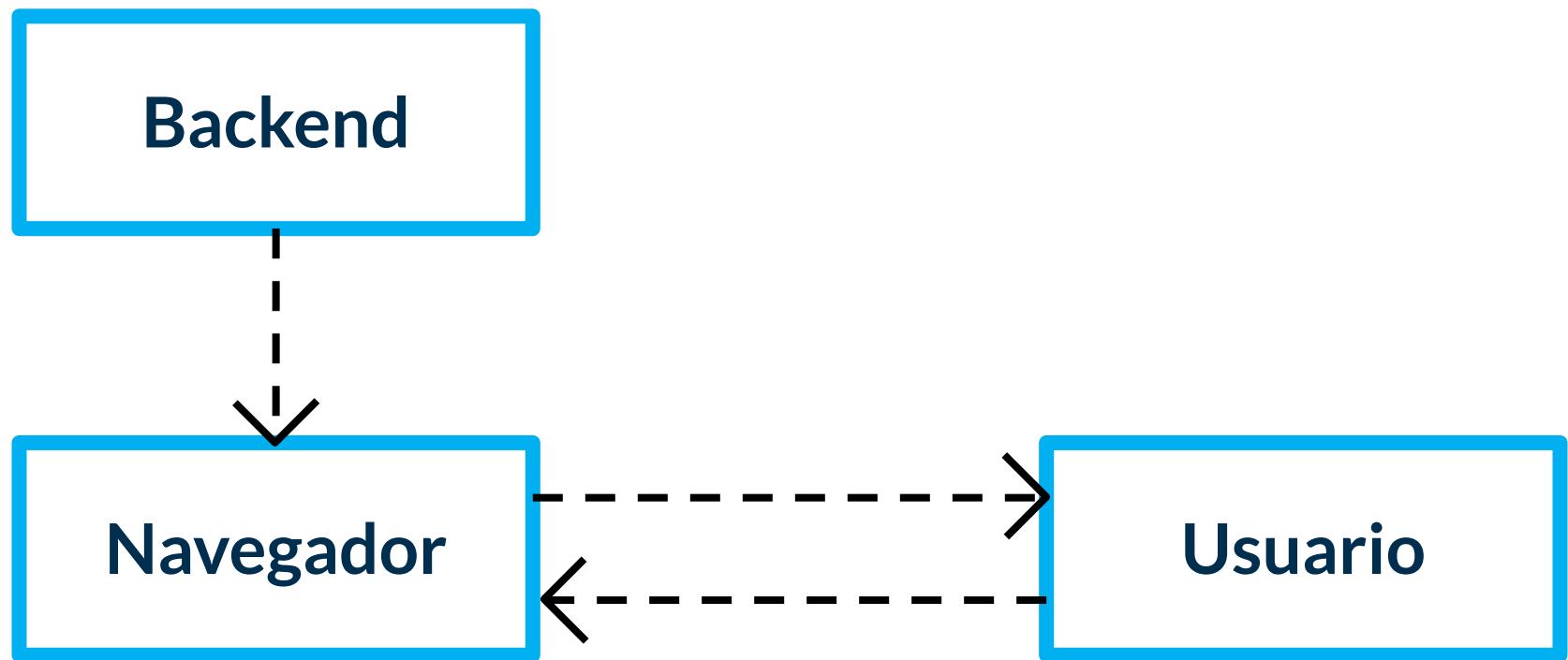
Tipos de aplicaciones según su router: SPAs vs. SSR

Cómo escalar sitios
o aplicaciones web

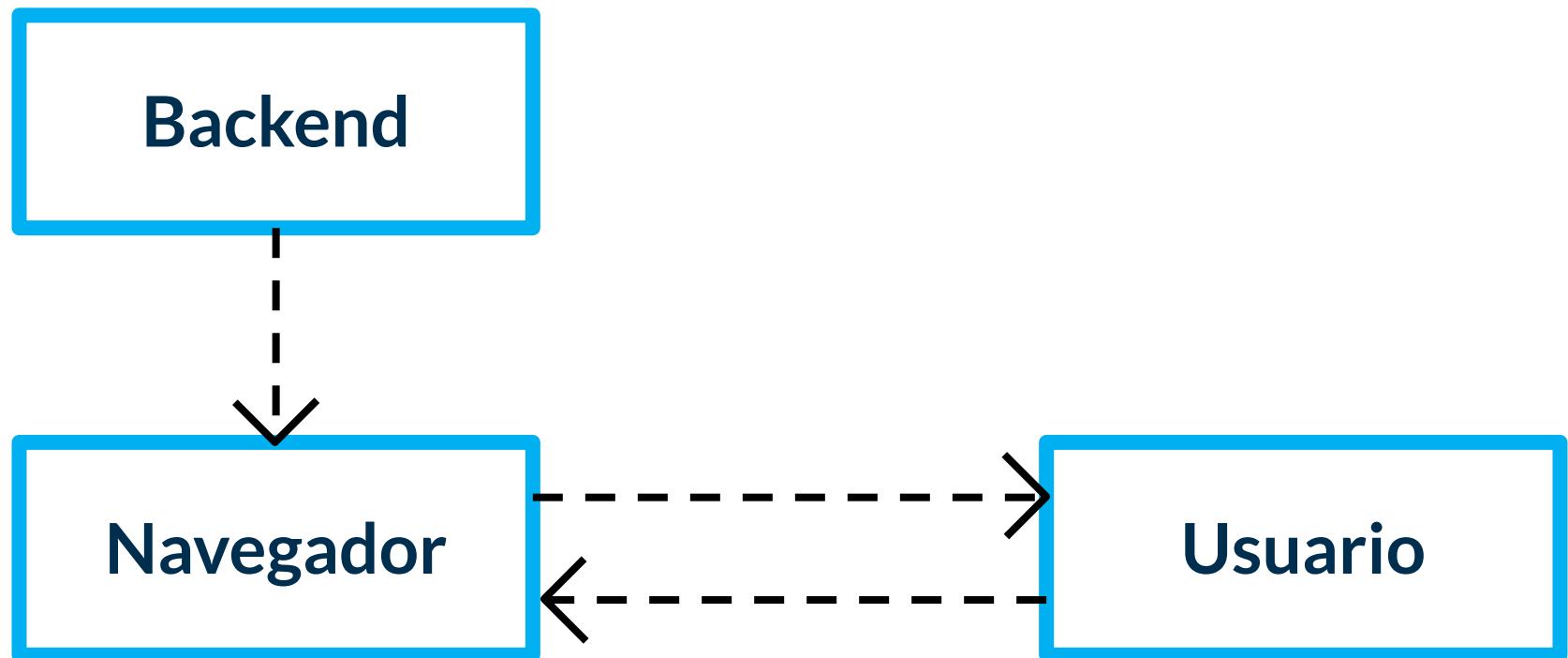
SSR "viejo"



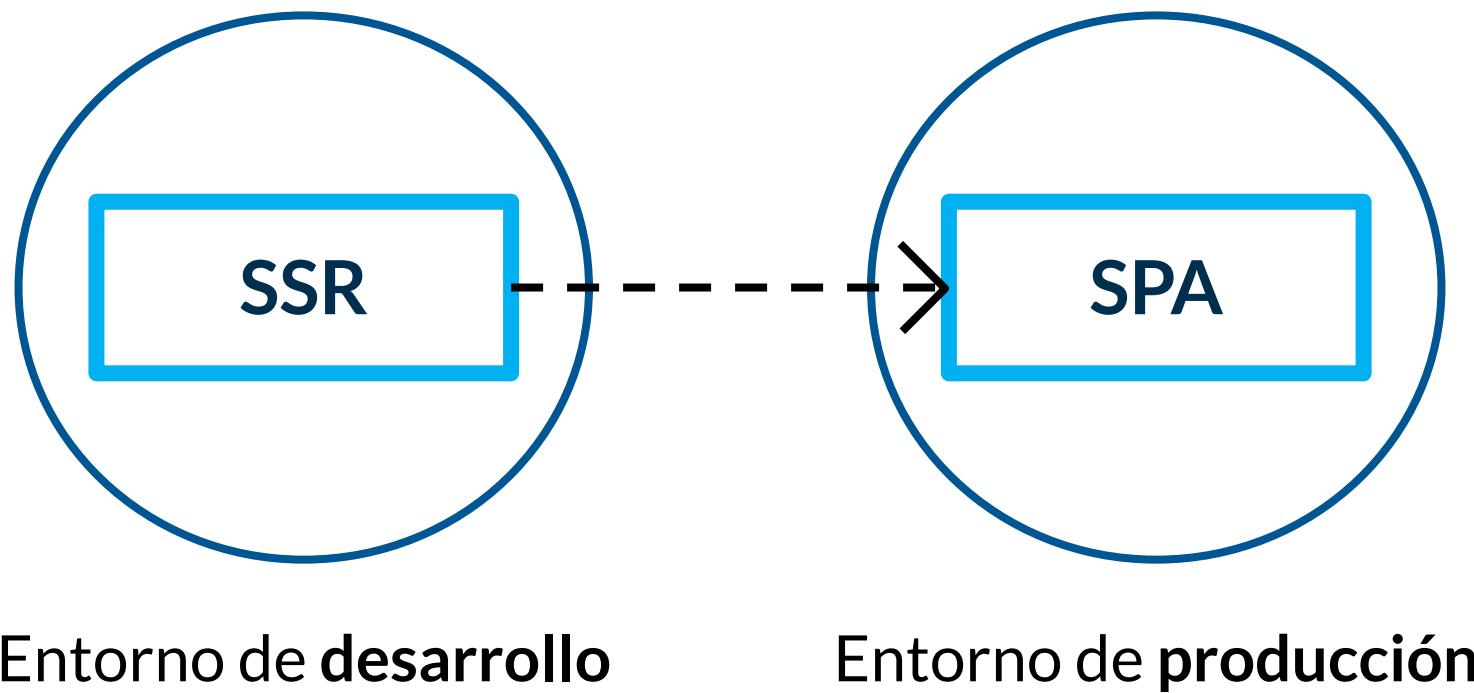
SPAs y CSR



Progressive SSR



Generación de sitios estáticos



Frameworks sobre frameworks: Next.js y Nuxt.js

Organización de archivos en el frontend

Cómo escalar sitios
o aplicaciones web

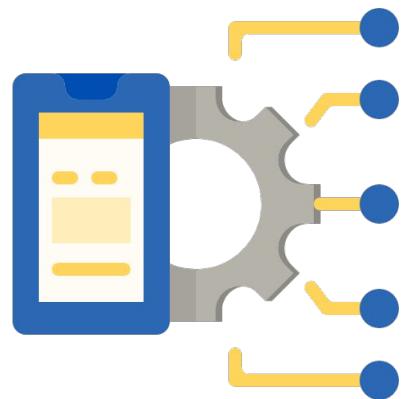
TECHNICAL DEBT



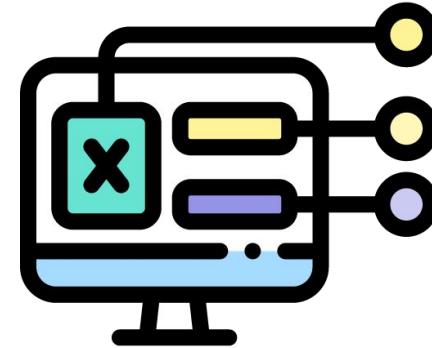
I DON'T
UNDERSTAND
WHY IT TAKES
SO LONG TO
ADD A NEW
WINDOW.

@VINCENTDNL

Lógica vs. Interfaz



Lógica



UI

File Type First

Una carpeta
para **cada tipo**
de componente.

/components

- MenuComponent.js
- FormComponent.js

/containers

- MenuContainer.js
- FormContainer

/routes

- HomeRoute.js
- AboutRoute.js

Feature first

Una carpeta
para **cada feature**.

/Menu

- MenuComponent.js
- FormComponent.js

/Form

- FormUI.js
- FormLogic.js

/Button

- Button.js
- Button.styles.js

Apps for apps

Una carpeta
para cada aplicación.

/StudentsPlatform

- /components
- /containers

/BusinessPlatform

- /Form
- /Button

/SalesLanding

- /html
- /css
- /js

¿Y ahora?

Cursos para especializarte
en Desarrollo Web Frontend
con Frameworks de JavaScript