

# One hour to learn Stencil

Horacio Gonzalez - @LostInBrittany



@LostInBrittany



# Who am I?

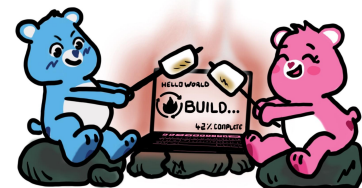
Introducing myself



# Horacio Gonzalez

**@LostInBrittany**

Spaniard lost in  
Brittany, developer,  
dreamer and all-around  
geek



**@LostInBrittany**



# The 3 minutes context

What the heck are web component?



# Web Components



Web standard W3C

# Web Components



Available in all modern browsers:  
Firefox, Safari, Chrome

# Web Components



Create your own HTML tags  
Encapsulating look and behavior

# Web Components



Fully interoperable

With other web components, with any framework



# Web Components



CUSTOM ELEMENTS



SHADOW DOM



TEMPLATES

# Custom Element



To define your own HTML tag

```
<body>
  ...
  <script>
    window.customElements.define('my-element',
      class extends HTMLElement {...});
  </script>
  <my-element></my-element>
</body>
```

# Shadow DOM



To encapsulate subtree and style in an element

```
<button>Hello, world!</button>  
<script>  
var host = document.querySelector('button');  
const shadowRoot = host.attachShadow({mode: 'open'});  
shadowRoot.textContent = 'こんにちは、影の世界!';  
</script>
```

Hello, world!



こんにちは、影の世界!

# Template



To have clonable document template

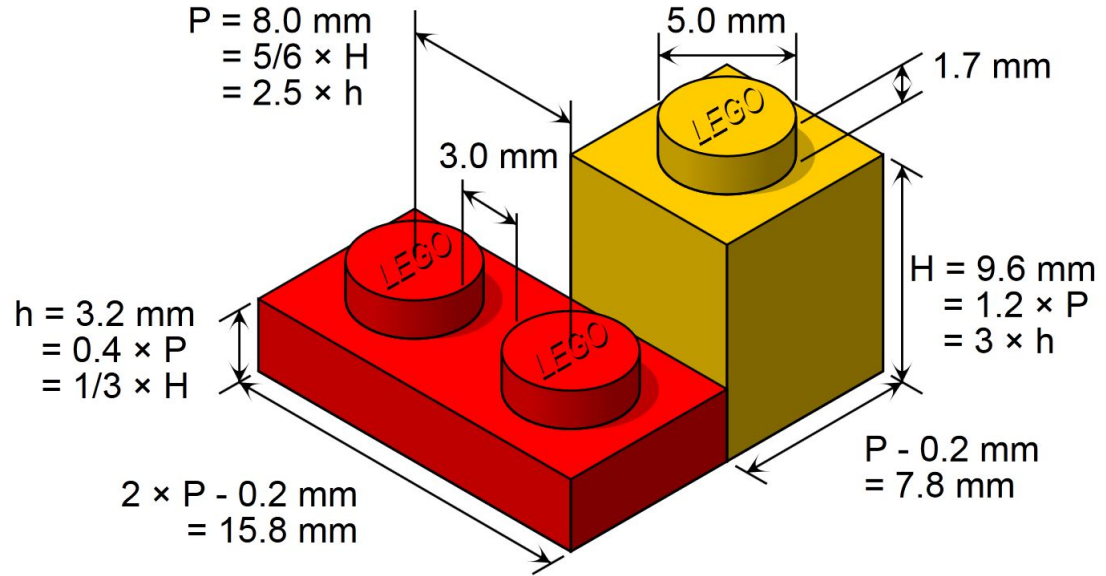
```
<template id="mytemplate">  
  <img src="" alt="great image">  
  <div class="comment"></div>  
</template>
```

```
var t = document.querySelector('#mytemplate');  
// Populate the src at runtime.  
t.content.querySelector('img').src = 'logo.png';  
var clone = document.importNode(t.content, true);  
document.body.appendChild(clone);
```

# But in fact, it's just an element...



- Attributes
- Properties
- Methods
- Events





# Stencil

Powering Ionic 4+



# Not another library



[Docs](#) [Resources](#) [Blog](#) [Twitter](#) [GitHub](#) [LinkedIn](#)



## Build. Distribute. Adopt.

Stencil is a toolchain for building reusable, scalable Design Systems. Generate small, blazing fast, and 100% standards based Web Components that run in every browser.

GET STARTED

WHY STENCIL?



Powering design systems and cross-framework components at



amazon

PORSCHE

arm

Panera

Microsoft

## A Web Component toolchain


@LostInBrittany

# A mature technology



**ionic** @Ionicframework · Jan 23  
Ionic Framework 4.0 is here! 🎉

- ⚙️ Web Components = Stability + Compatibility
- ⚡ Better performance & startup times
- 👤 Made for developers & their preferred tooling
- 🎨 Powerful theming & CSS variables
- 🔮 Future-proof development, built in
- 📄 New 4.0 docs



**Introducing Ionic 4: Ionic for Everyone**  
Today I am thrilled to announce the 4.0 release of Ionic Framework, lovingly known as "Ionic for Everyone." 🎉 Ionic 4 represents the culmination of [blog.ionicframework.com](http://blog.ionicframework.com)

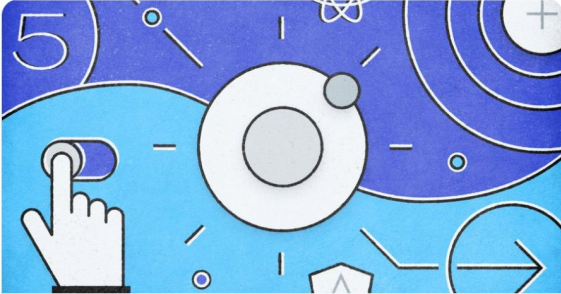


**ionic** @Ionicframework · Feb 11  
IONIC 5 IS HERE! 🎉

The biggest design update ever in the history of Ionic:

- ✓ New iOS & Android component designs
- ✓ New React & Angular starters
- ✓ New Ionicons icon pack
- ✓ New Animations API

Check the Blog for the full story and get building! 🏗️



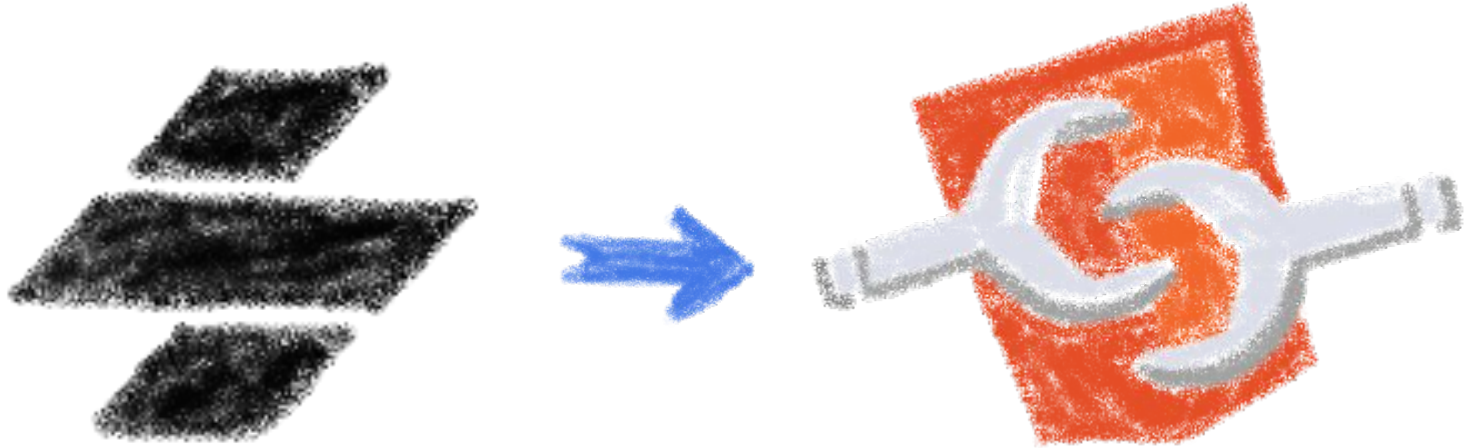
**Announcing Ionic 5! - Ionic Blog**  
Today, I'm thrilled to announce Ionic Framework 5 (Magnesium)! 🎉 This release includes iOS 13 design updates, a brand new API for creating ... [ionicframework.com](http://ionicframework.com)

54 622 1.2K

## Powering Ionic Framework



# A build time tool



To generate standard web components

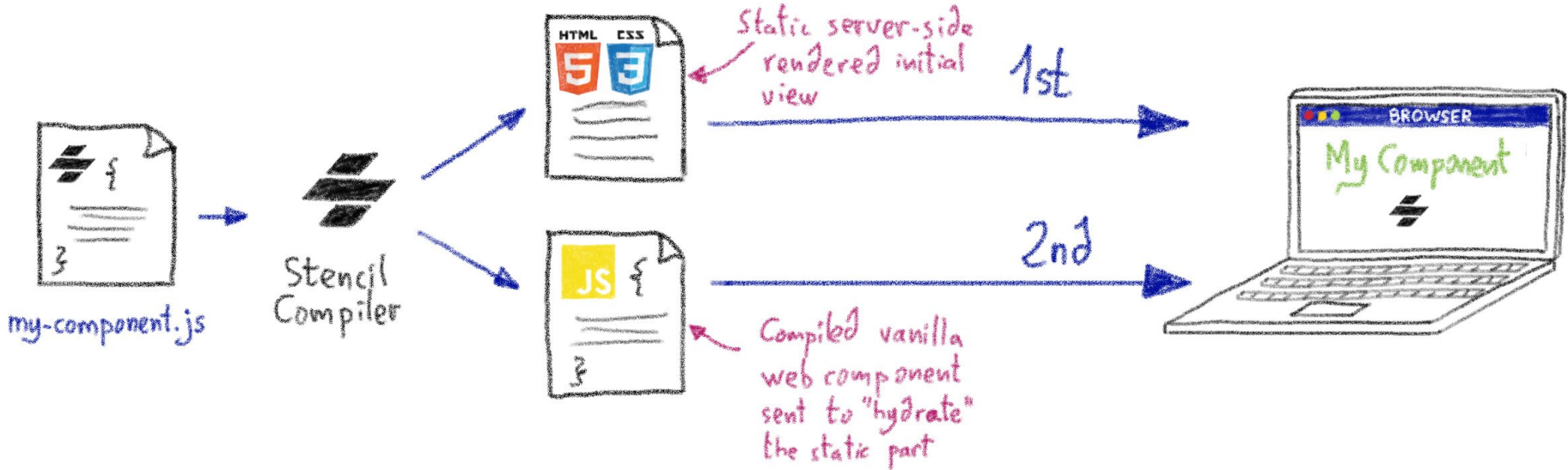
# Fully featured



- Web Component-based
- Asynchronous rendering pipeline
- TypeScript support
- Reactive Data Binding
- Component pre-rendering
- Simple component lazy-loading
- JSX support
- Dependency-free components



# And the cherry on the cake



Server-Side Rendering

# Stencil leverages the web platform



Stencil doesn't fight the web platform. It embraces it.



## Simple

With intentionally small tooling, a tiny API, and zero configuration, Stencil gets out of the way and lets you focus on your work.



## Lightweight

A tiny runtime, pre-rendering, and the raw power of native Web Components make Stencil one of the fastest compilers around.



## Future proof

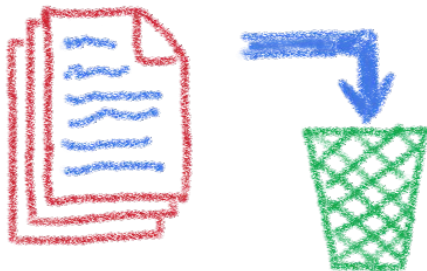
Build cross-framework components and design systems on open web standards, and break free of Framework Churn.

Working with the web, not against it

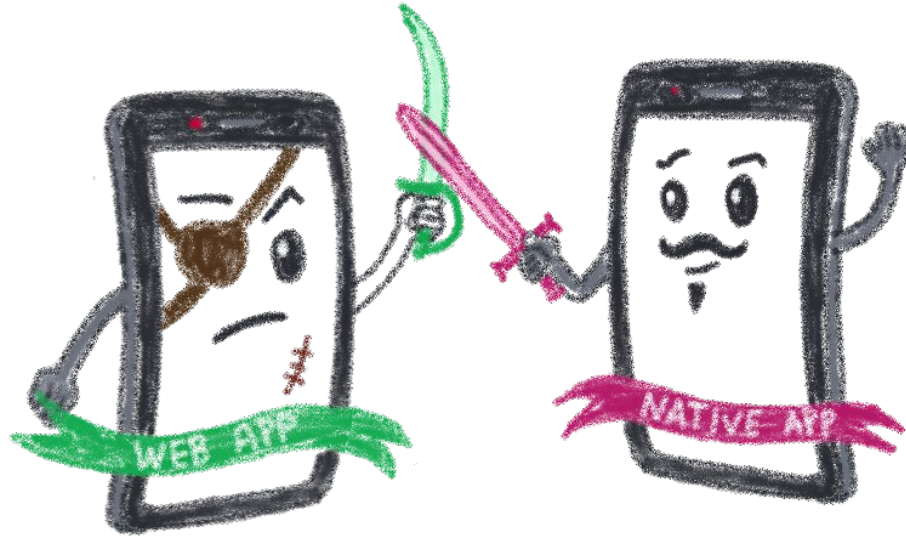


# The Stencil story

A company tired of putting good code in the bin



# Once upon a time there was a fight



Between native apps and web app on mobile

# A quest to the perfect solution



Hybrid apps, leveraging on web technologies

# A company wanted to do it well



The perfect technology for mobile web and hybrid apps

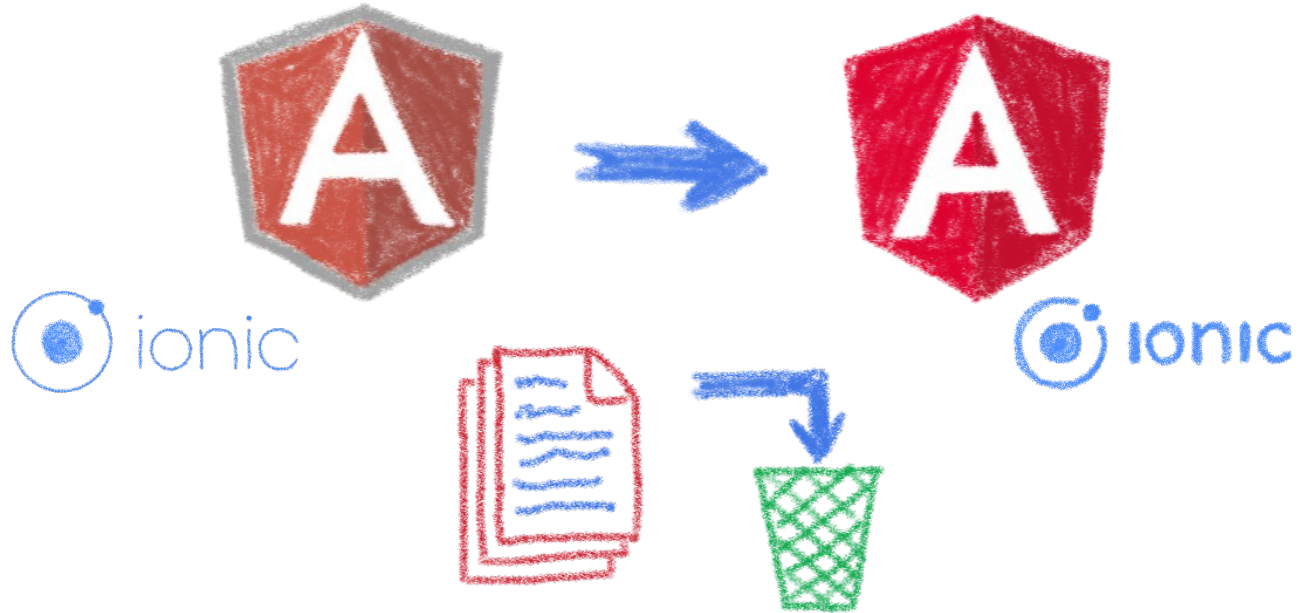


# The time is 2013



So what technology would you use?

# Really soon after launch...



*Hey folks, we are killing AngularJS!*

# What did Ionic people do?



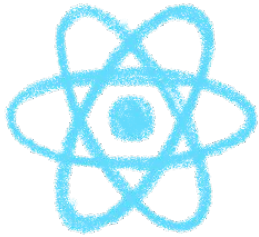
Let's put everything in the trash bin and begin anew

# But times have changed...



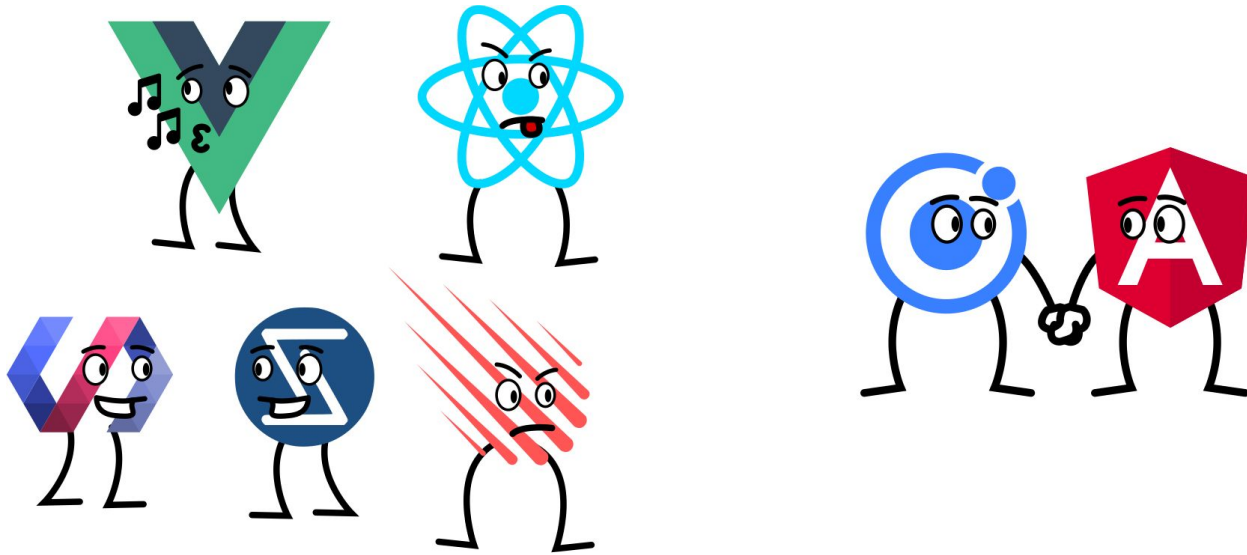
In 2013 Angular JS was the prom queen

# Times have changed...



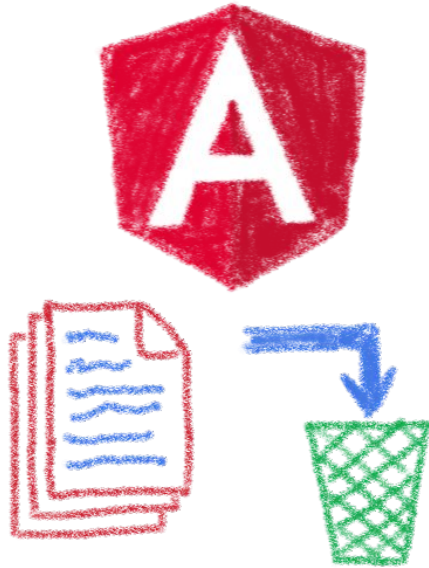
In 2017 Angular is only one more in the clique

# Angular limits adoption of Ionic



Devs and companies are very vocal about JS Frameworks

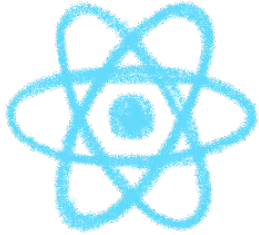
# What did Ionic people do?



Let's put everything in the trash bin and begin anew...

But on which framework?

# What about web components?



A nice solution for Ionic problems:  
Any framework, even no framework at all!



# But what Web Component library?



hybrids



snuggsi ツ



**smart**  
HTML ELEMENTS



slim.js



SkateJS

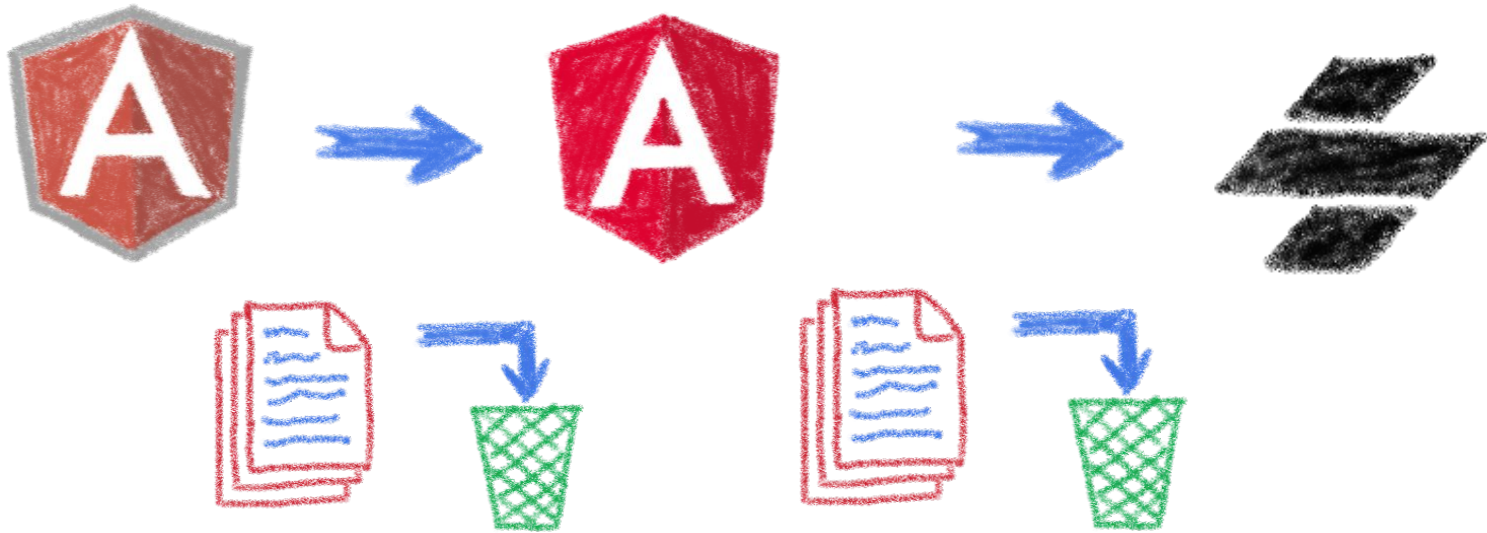
There were so many of them!

# Let's do something different



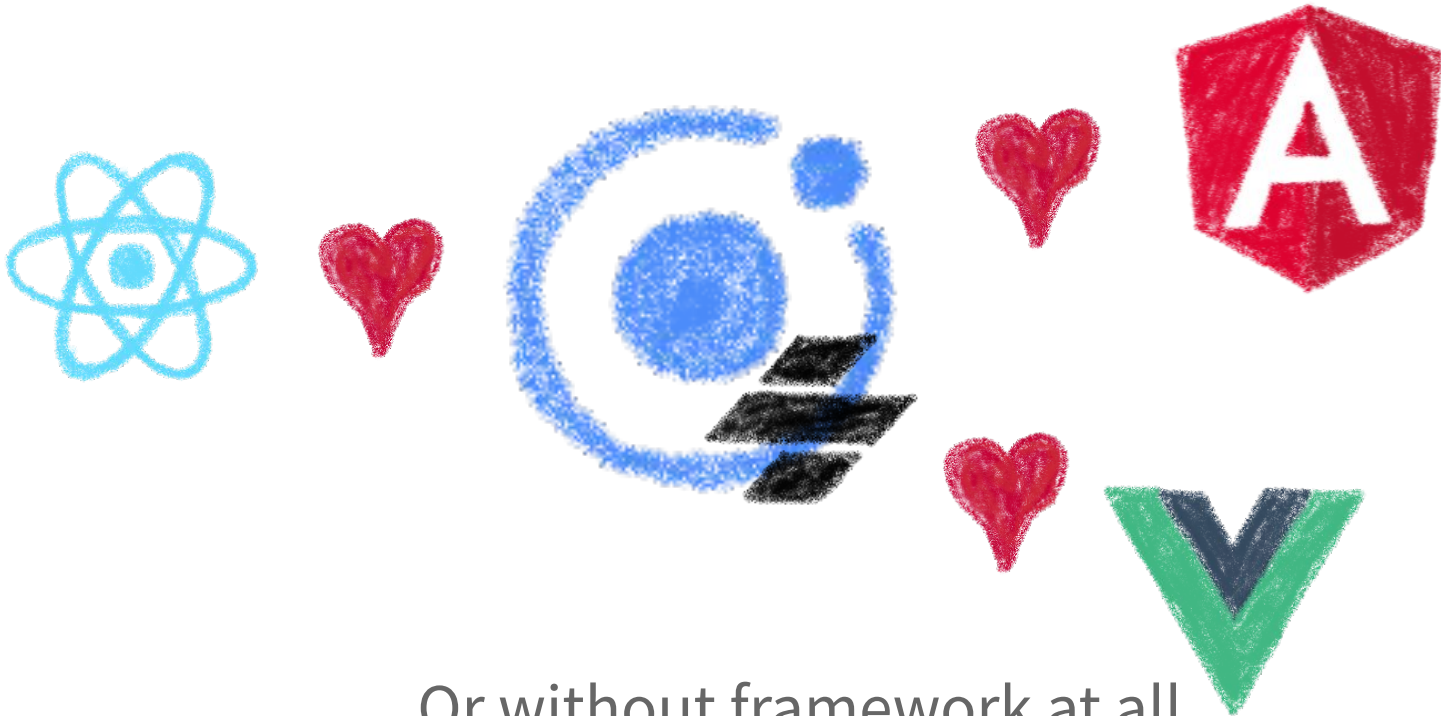
A fully featured web component toolchain  
With all the bells and whistles!

# Ionic rewrote all their code again



Ionic 4 is fully based on Ionic

# Now Ionic works on any framework



Or without framework at all

# And we have Stencil



To use it in any of our projects

@LostInBrittany



# Hey dude, enough stories!

We are here to see some code!

010  
0101010  
1011010101

# Hands on Stencil



Simply use npm init

```
npm init stencil
```

Choose the type of project to start

```
? Pick a starter › - Use arrow-keys. Return to submit.  
  
› ionic-pwa      Everything you need to build fast, production ready PWAs  
  app           Minimal starter for building a Stencil app or website  
  component     Collection of web components that can be used anywhere  
Updating Stencil
```

# Hands on Stencil



And the project is initialized in some seconds!

```
✓ Pick a starter > component
✓ Project name > codemotion-online
✓ All setup in 17 ms

$ npm start
  Starts the development server.
$ npm run build
  Builds your components/app in production mode.
$ npm test
  Starts the test runner.

We suggest that you begin by typing:
$ cd codemotion-online
$ npm install
$ npm start

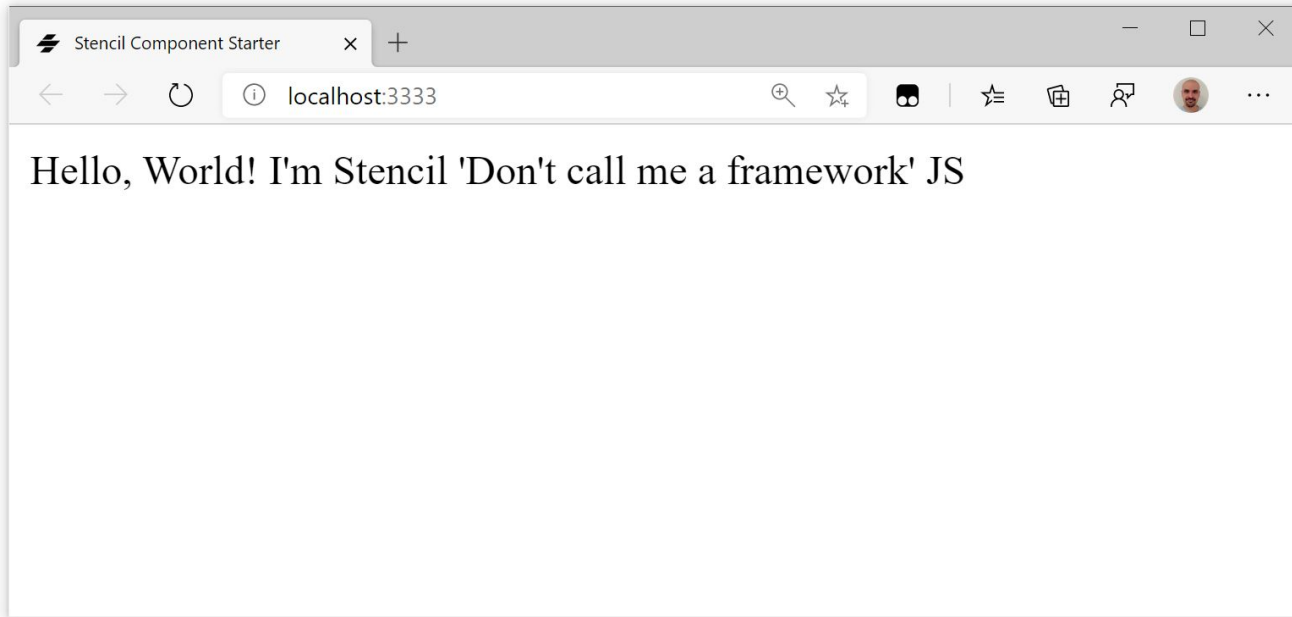
Happy coding! 🍷
```



# Starting the development server



```
npm start
```



# Let's look at the code



```
File Edit Selection View Go Debug Terminal Help my-component.tsx - Untitled (Workspace) - Visual Studio Code

EXPLORER
OPEN EDITORS
  my-component.tsx src/components/my-com...
UNTITLED (WORKSPACE)
  sthlm-js
  .stencil
  dist
  node_modules
  src
    components / my-component
      # my-component.css
      TS my-component.e2e.ts
      my-component.tsx
      readme.md
    utils
      TS components.d.ts
      index.html
      TS index.ts
    www
      .editorconfig
      .gitignore
      LICENSE
      package-lock.json
      package.json
      readme.md
      TS stencil.config.ts
      tsconfig.json
  OUTLINE
  NPM SCRIPTS

my-component.tsx
1 import { Component, Prop, h } from '@stencil/core';
2 import { format } from '../utils/utils';
3
4 @Component({
5   tag: 'my-component',
6   styleUrls: 'my-component.css',
7   shadow: true
8 })
9 export class MyComponent {
10   /**
11    * The first name
12    */
13   @Prop() first: string;
14
15   /**
16    * The middle name
17    */
18   @Prop() middle: string;
19
20   /**
21    * The last name
22    */
23   @Prop() last: string;
24
25   private getText(): string {
26     return format(this.first, this.middle, this.last);
27   }
28
29   render() {
30     return <div>Hello, World! I'm {this.getText()}</div>;
31   }
32 }
33
```

# Some concepts



```
import { Component, Prop, h } from '@stencil/core';
import { format } from '../../utils/utils';

@Component({
  tag: 'my-component',
  styleUrls: 'my-component.css',
  shadow: true
})
export class MyComponent {

  @Prop() first: string;
```

Decorators

# Some concepts



```
@Prop() first: string;  
  
@Prop() middle: string;  
  
@Prop() last: string;  
  
@State() nickname: string;
```

Properties and States

# Some concepts



```
render() {  
  return <div>Hello, World! I'm {this.getText()}</div>;  
}
```

Asynchronous rendering using JSX

# Some concepts



```
@Prop() value: number;

@Watch(value)
valueChanged(newValue: boolean, oldValue: boolean) {
  console.log(`The new value is ${newValue}, it was ${oldValue} before`);
}
```

Watch

# Some concepts



```
@Event() actionCompleted: EventEmitter;

someAction(message: String) {
  this.actionCompleted.emit(message);
}
```

## Emitting events

```
@Listen('actionCompleted')
actionCompletedHandler(event: CustomEvent) {
  console.log('Received the custom actionCompleted event: ', event.detail);
}
```

## Listening to events

# Some concepts



```
@Method()
async sayHello() {
  this.hello = true;
}

render() {
  return (
    <Host>
      <h2>{ this.hello ? `Hello sthlm.js` : ''}</h2>
    </Host>
  );
}
```

Asynchronous public methods



# Some concepts



```
@Component({
  tag: 'my-component',
  styleUrls: ['my-component.css'],
  shadow: true
})
export class MyComponent {
```

Optional Shadow DOM



# Stencil for design systems

Because web components really shine for that



# What the heck is a design system?



Components

Patterns

No more time spent  
rewriting once again

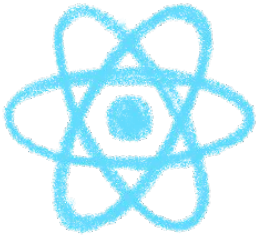
the same base UI elements

Visual language

Design artifacts  
&

Code implementation

# Why Stencil is so good for design systems?



Web Components work everywhere!



# One more thing...\*

Let's copy from the master

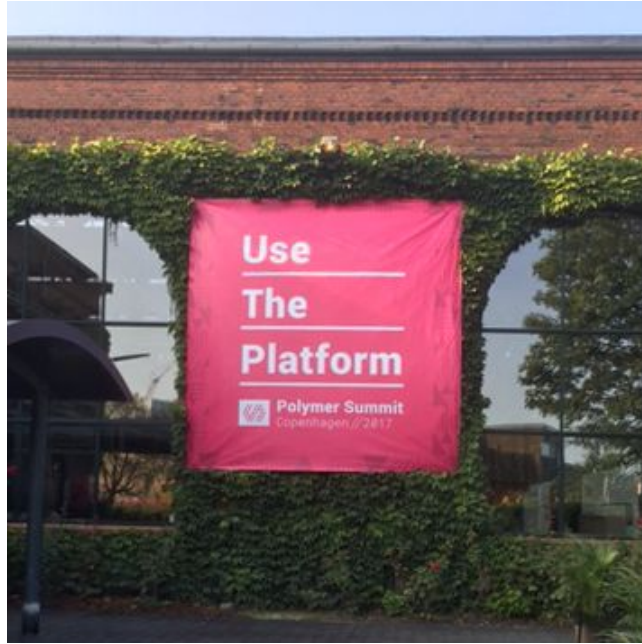


# Stencil is not so important

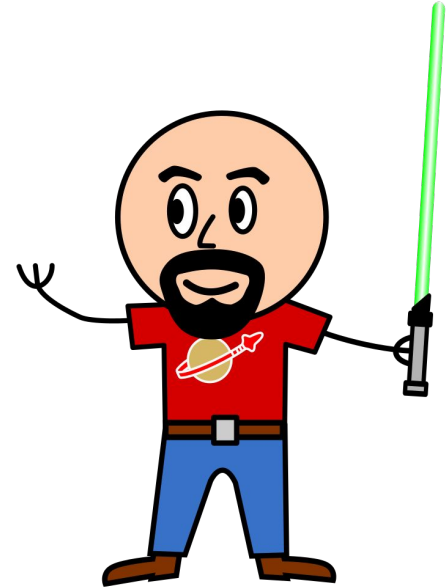


WebComponents ARE

# Use the Platform, Luke...



WebComponents ARE native



# Do you love your framework?



Oh yeah, we all do



# Would you marry your framework?



Like until death...

# How much does cost the divorce?



Do you remember when you dropped AngularJS for Angular?

# Why recode everything again?



Reuse the bricks in your new framework

# Lots of web components libraries



slim.js

LitElement



hybrids



snuggsi ツ



SkateJS



**smart**  
HTML ELEMENTS



**stencil**

For different need and sensibilities

# And some good news



Angular Elements



Vue Web Component  
Wrapper

Frameworks begin to understand it

# So for your next app



Choose a framework, no problem...

But please, help your future self

# Use Web Components!





