

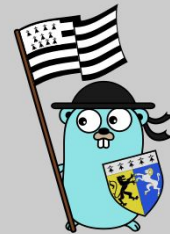


# STENCIL

## and the future of Ionic

**Horacio Gonzalez**  
@LostInBrittany

# Finist Devs



# Horacio Gonzalez



@LostInBrittany

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



<http://cityzendata.com>



# Ionic Framework

---

Have you meet Ionic?

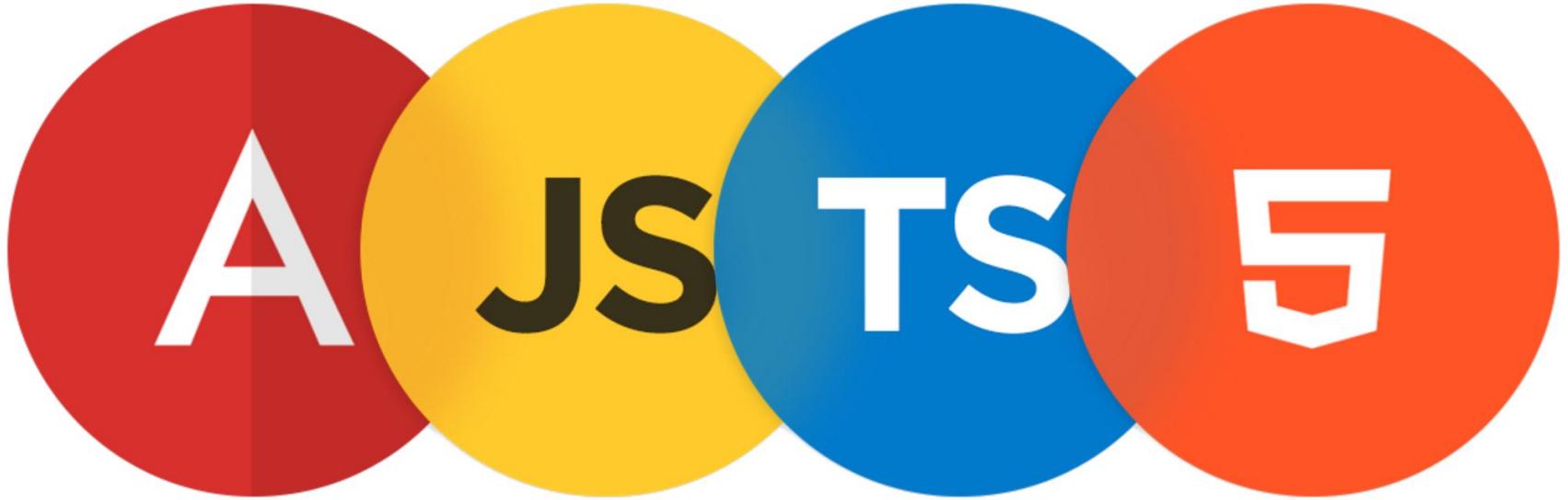


# What is Ionic, and where does it fit?



An open source framework for  
developing hybrid mobile apps

# What is inside Ionic?





# Angular



One framework.  
Mobile & desktop.

GET STARTED

Angular Provides: Components, Filters,  
Two way data binding, Services,  
DI & Testability

# Angular and Ionic

A match made in heavens...



UI components are Angular components  
All the power of Angular available for the app



# The Ionic components



UI controls missing from HTML  
but common on mobile apps

# And is it slow?

Performance is an obsession in Ionic



Minimal DOM manipulation, zero jQuery  
and hardware accelerated transitions

# Fully responsive



Applications adapted to  
multiple device resolutions

# And what about beauty?

Clean, simple, and functional design



Lots of UI components  
and ready-to-use widgets



# But I'm a hardcore developer!

Then you have the power of a powerful CLI

```
$ ionic start myApp
Creating myApp... done

Your app is ready to go!
```

Use just one command to create, build, test, and deploy your Ionic apps onto any platform

# And soon Ionic 4!

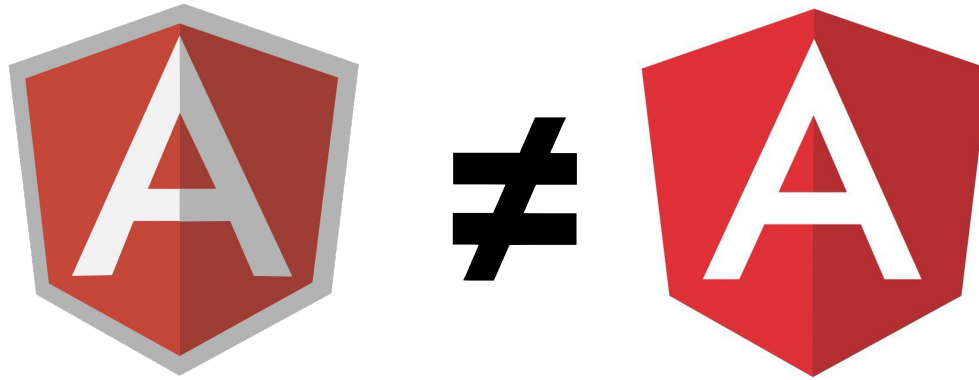
---

Where everything changes... or not?



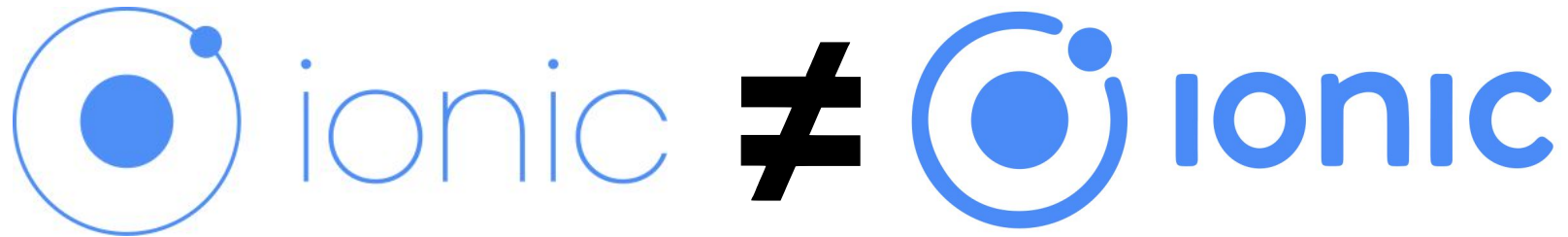
# From Angular JS to Angular...

Migrating isn't easy



Especially when you have a full catalog of directives...

# Ionic endured the migration



Ionic 1 based on Angular JS

Ionic 2 based on Angular

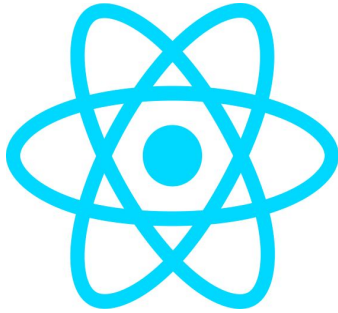


# Times have changed...



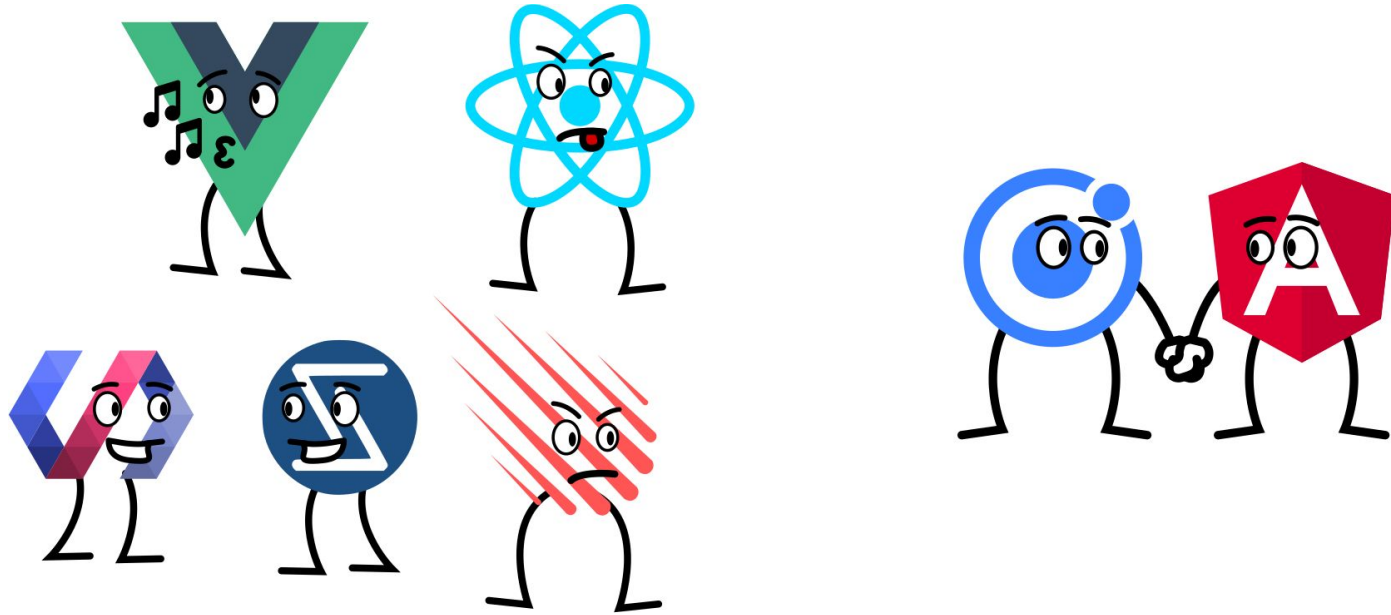
In 2013 Angular JS was  
the prom queen

# 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

# A new hope...





---

# A new breed of Web Components

# Next generation Ionic



**STENCIL**

Ionic 4 will be fully based on web components  
using a new toolkit: Stencil

# New kid on the block



 **Max Lynch**  
Co-founder and CEO of Ionic  
@maxlynch

Using Web Components in Ionic (Polymer Summit 2017)

15,345 views

👍 282 🗨️ 3 ➦ SHARE



**Google Chrome Developers**  
Published on Aug 23, 2017

SUBSCRIBE 159K

Developers and businesses are struggling to build fast mobile web apps to reach the next billion users. This talk explores the challenges faced and lessons learned as the Ionic Framework team ported over their collection of mobile-first UI components from a traditional frontend framework to

SHOW MORE

Up next

AUTOPLAY



**Stencil | Getting Started**  
Academind  
6.8K views



**Intro to Web Components with StencilJS**  
Madness Labs  
701 views



**Faster Web Apps Using Stencil**  
Paul Halliday  
3.3K views



**What You See is What You Deserve: Simple Visual Tools**  
Google Chrome Developers  
12K views



**Polymer Summit 2017**  
Google Chrome Developers



**Practical lessons from a year of building web components -**  
Google Chrome Developers  
47K views



**ES6 Modules in the Real World (Polymer Summit 2017)**  
Google Chrome Developers  
8.4K views

# Announced during Polymer Summit



# Not another library



## The magical, reusable web component compiler



### Simple

With intentionally small tooling, a tiny API, zero configuration, and TypeScript support, you're set.



### Performant

6kb min+gzip runtime, server side rendering, and the raw power of native Web Components.



### Future proof

Build versatile apps and components based 100% on web standards. Break free of Framework Churn.

## A Web Component compiler



# A build time tool



To generate standard web components



# Fully featured

- Virtual DOM
- Async rendering
- Reactive data-binding
- TypeScript
- JSX

# And the cherry on the cake



# SSR

## Server-Side Rendering



# Hands on Stencil

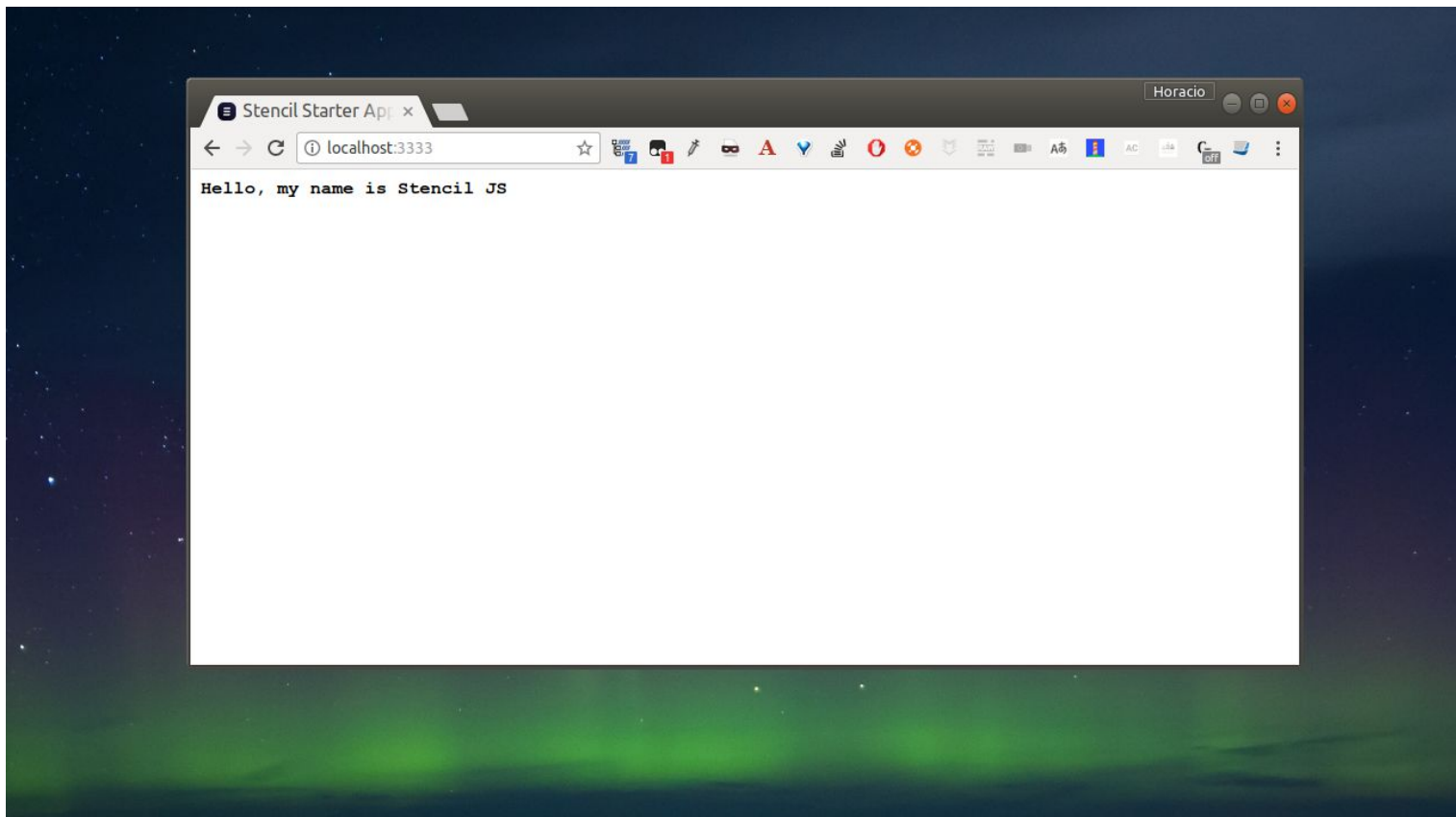
## Clone the starter project

```
git clone https://github.com/ionic-team/stencil-app-starter my-app  
cd my-app  
git remote rm origin  
npm install
```

## Start a live-reload server

```
npm start
```

# Hands on Stencil





# Hands on Stencil

The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left and the Code Editor on the right. The Explorer sidebar shows a project structure for a Stencil application, with the file `my-name.tsx` selected. The Code Editor displays the following TypeScript code:

```
1 import { Component, Prop } from '@stencil/core';
2
3
4 @Component({
5   tag: 'my-name',
6   styleUrls: 'my-name.scss'
7 })
8 export class MyName {
9
10  @Prop() first: string;
11
12  @Prop() last: string;
13
14  render() {
15    return (
16      <div>
17        Hello, my name is {this.first} {this.last}
18      </div>
19    );
20  }
21 }
22
```

The status bar at the bottom indicates the current file is `my-name.tsx` at line 22, column 1, using UTF-8 encoding and LF line endings. The editor is running TypeScript React 2.6.1.

# Some concepts

```
render() {  
  return (  
    <div>Hello {this.name}</div>  
  )  
}
```

```
render() {  
  return (  
    <div>{this.name ? <p>Hello {this.name}</p> : <p>Hello World</p>}</div>  
  );  
}
```

## JSX declarative template syntax

# Some concepts

```
import { Component } from '@stencil/core';

@Component({
  tag: 'todo-list',
  styleUrls: 'todo-list.scss'
})
export class TodoList {
  @Prop() color: string;
  @Prop() favoriteNumber: number;
  @Prop() isSelected: boolean;
  @Prop() myHttpService: MyHttpService;
}
```

## Decorators



# Some concepts

```
import { Event, EventEmitter } from '@stencil/core';

...
export class TodoList {

  @Event() todoCompleted: EventEmitter;

  someAction(todo: Todo) {
    this.todoCompleted.emit(todo);
  }

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

## Events

# Some concepts



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

}
```

## Optional Shadow DOM



# Some concepts

stencil.config.js

```
exports.config = {  
  namespace: 'myname',  
  generateDistribution: true,  
  generateWWW: false,  
  ...  
};
```

## Generate distribution

# Conclusion

---

That's all folks!



