



Angular 2

@webtrainingmx



Introduction

¿What will we learn?

- NPM, JSPM, Gulp.js, Git
- Quick Start
- The Angular 2 app structure
- Components
- Services
- Inputs/Outputs
- Pipes



The basics

¿What the h... is an *SPA*?

- **Single Page Application**
 - Angular 2
 - React
 - Ember.js
 - ...

No, so fast... ¿What is a RESTful API?

- Mechanism to interact with data sources via a semantic URI convention using HTTP verbs
 - What?

¿When should I use a framework?

- Multiple data sources (databases, file system)
- Integration with other systems
- Third party "head-less" platforms
- Pure front-end teams

¿Why Angular 2?

- Full-stack framework not just *library*
- TypeScript / Dart / ES2015 ready
- “Web-component” oriented
- License friendly!



Installation

Clone the official repository

- Git / Git bash
 - <http://git-scm.com/>

```
git clone https://github.com/angular/quickstart.git
```



App structure

Some important files

- `package.json`
- `systemjs.config.js`
- `tsconfig.json`
- `typings.json`
- `tslint.json`

Some important directories

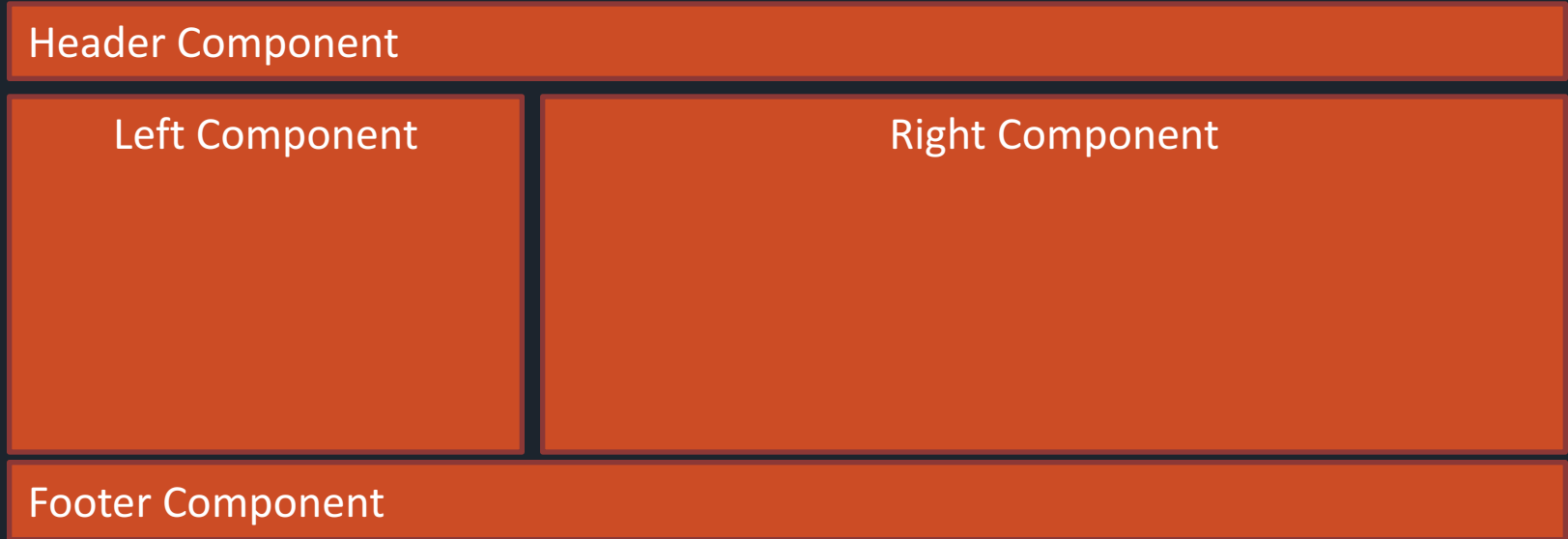
- **app**
- **node_modules**
- **typings**



Components

Angular 2 @Component

- Build UI elements and logic on the page



Minimal component structure

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

@Component({
  selector: 'my-app',
  template: '<h1>My First Angular 2 App</h1>'
})
export class AppComponent { }
```




Templates and directives

Managing data in templates

- Binding with interpolation
- Logic: ngIf
- Iteration: ngFor



Services

Angular 2 @Injectable()

- A service represents a way to interact with data acquisition
- Can use HTTP for communication (via Observables)

Minimal service structure

```
import { Injectable } from '@angular/core';  
  
@Injectable()  
export class HeroService { }
```



Pipes

Angular 2 @Pipe

- Write display-value transformations declared in our HTML
- Similar to a pipe in GNU/Linux or a filter in other template systems

Minimal pipe structure

```
@Pipe({name: 'lowercase'})  
class LowercasePipe {  
  transform(value: string, args: any[]) {  
    return value.toLowerCase();  
  }  
}
```

```
<h2>{{ title | lowercase }}</h2>
```




Inputs / Outputs

Angular 2 @Input

- Send data from parent component to a child component

```
<poke-catalog [pokemon]="pokemonInParentComponent"></poke-catalog>
```

```
@Input() pokemon : Pokemon[];
```

Angular 2 @Output

- Send data from a child component to its parent component

```
<poke-navigation (searchValueUpdated)="searchValueUpdated($event)"></poke-navigation>
```

```
@Output() searchValueUpdated = new EventEmitter();
```



Thank you!

Alex Arriaga

<http://www.alex-arriaga.com/>
[@alex_arriaga_m](#)