

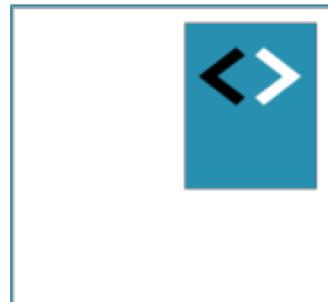


Angular Fundamentals

Module 1 – Core



A CANON COMPANY



Peter Kassenaar
info@kassenaar.com

Peter Kassenaar

On Peter Kassenaar:

- Trainer, author, developer – since 1996
- Specialty: "*Everything JavaScript*"
- JavaScript, ES6, Angular, NodeJS, TypeScript,

www.kassenaar.com/blog

info@kassenaar.com

Twitter: [@PeterKassenaar](https://twitter.com/PeterKassenaar)



Angulartraining.nl

Home Training Dates Information Contact

2018 dates now available!

```
const routes: Routes = [
  { path: '', redirectTo: 'home', pathMatch: 'full' },
  { path: 'home', loadChildren: './home/home.module#HomeModule' },
  { path: 'training', loadChildren: './training/training.module#TrainingModule' },
];
const config: ExtraOptions = {
  enableTracing: false,
  preloadingStrategy: PreloadAllModules
};
@NgModule({
  imports: [RouterModule.forRoot(routes, config)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```

World-class Angular training in Dutch and English

Live classrooms - focused on today's developers

LEARN MORE SIGN UP!

www.angulartraining.nl

github.com/PeterKassenaar/oce

The screenshot shows the GitHub repository page for the user PeterKassenaar with the repository name oce. The page has a dark header with navigation links for Code, Pull requests, Issues, Marketplace, and Explore. Below the header, there's a search bar and a link to the repository's URL. The main content area displays the repository's title, "PeterKassenaar / oce", and its description, "Slides and example code on the course Angular, Océ - spring 2019". It shows 1 commit, 1 branch, 0 releases, and 1 contributor. A green button labeled "Clone or download" is visible. The repository's history shows one initial commit by PeterKassenaar, which is a README.md file. The commit message is "Initial commit", it was made 8 minutes ago, and it has a SHA of e0aabb1. The repository's README.md file content is displayed below the commit history, showing the text "oce" and the repository's description.

About you...



Introduce yourself shortly

Current knowledge, mobile apps, Angular apps?

Previous AngularJS 1.x- knowledge?

Other (web) languages?

Expectations of the training?

Specific or current projects?

Agenda - 3 + 2 days

- Introduction & short history – Why Angular?
- Key features of Angular 2 to 7
- Hello World in Angular – Looking at the boilerplate-code - CLI
- Angular 2 in depth (modules):
 - Components
 - ECMAScript 2015 + TypeScript
 - Data binding
 - Dependency Injection (DI) – more components
 - Services and Http, Observables (RxJS)
 - Routing, Forms, Unit testing
- BEST PRACTICES / STYLE GUIDE

Agenda - 3 days, 8 modules

Module 1

**Introduction, concepts
and architecture**

Module 2

Databinding

Module 3

Services

Module 4

Observables and RxJS

Agenda - 4 days, 8 modules

Module 5
Multiple Components

Module 6
Routing

Module 7
Reactive Forms

(Module 8)
Unit testing)

Overall – Océ specific stuff

Océ specifics (Roel)

- “Ocean” component library
 - OTB, IPC, CEP

Schedule, roughly...

- 9:00 ~ 12:00 Morning session
 - Coffee/tea break
- 12:00 – 12:45 Lunch
- 12:45 ~ 16:00-16:30 Afternoon session
 - Coffee/tea break
- Intense! Talks & hands-on workshops

Materials

- Software (NodeJS & NPM, Editor, browser)
- Handouts (Github, Océ Repo)
- Workshops (Github)
- Websites (online)



angular.io/

2 Guidelines

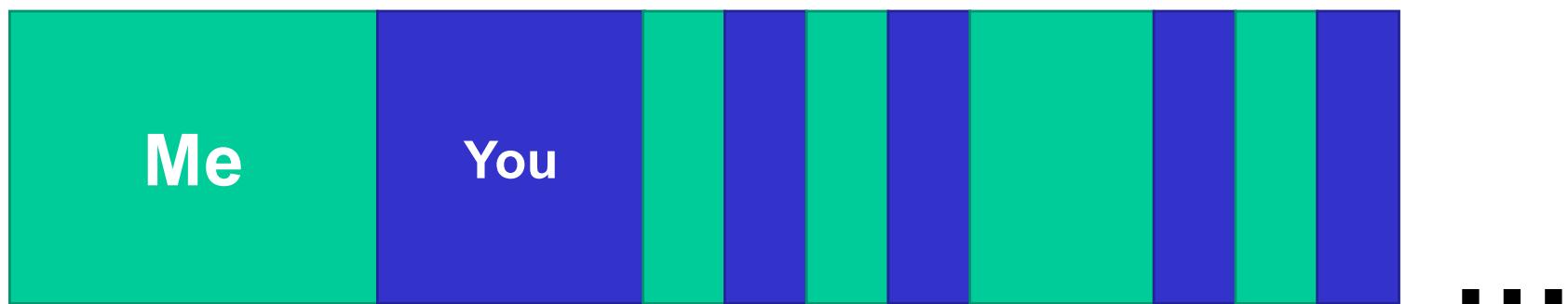
1. Workshops / Exercises

- But: get off the beaten path! Create your own project, app, website...

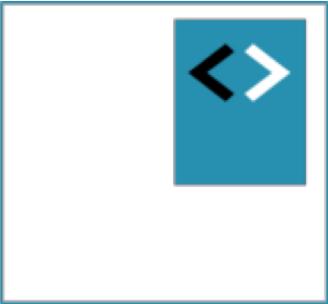
2. Example code – Updated to Angular V7.

- To support the exercises – ready made examples
- Work in progress – check Angular-site!
- <https://github.com/PeterKassenaar/voorbeeldenAngular2> (Dutch)

How I work...



Questions?



AngularJS vs. Angular 2-7

Key features, differences
And similarities



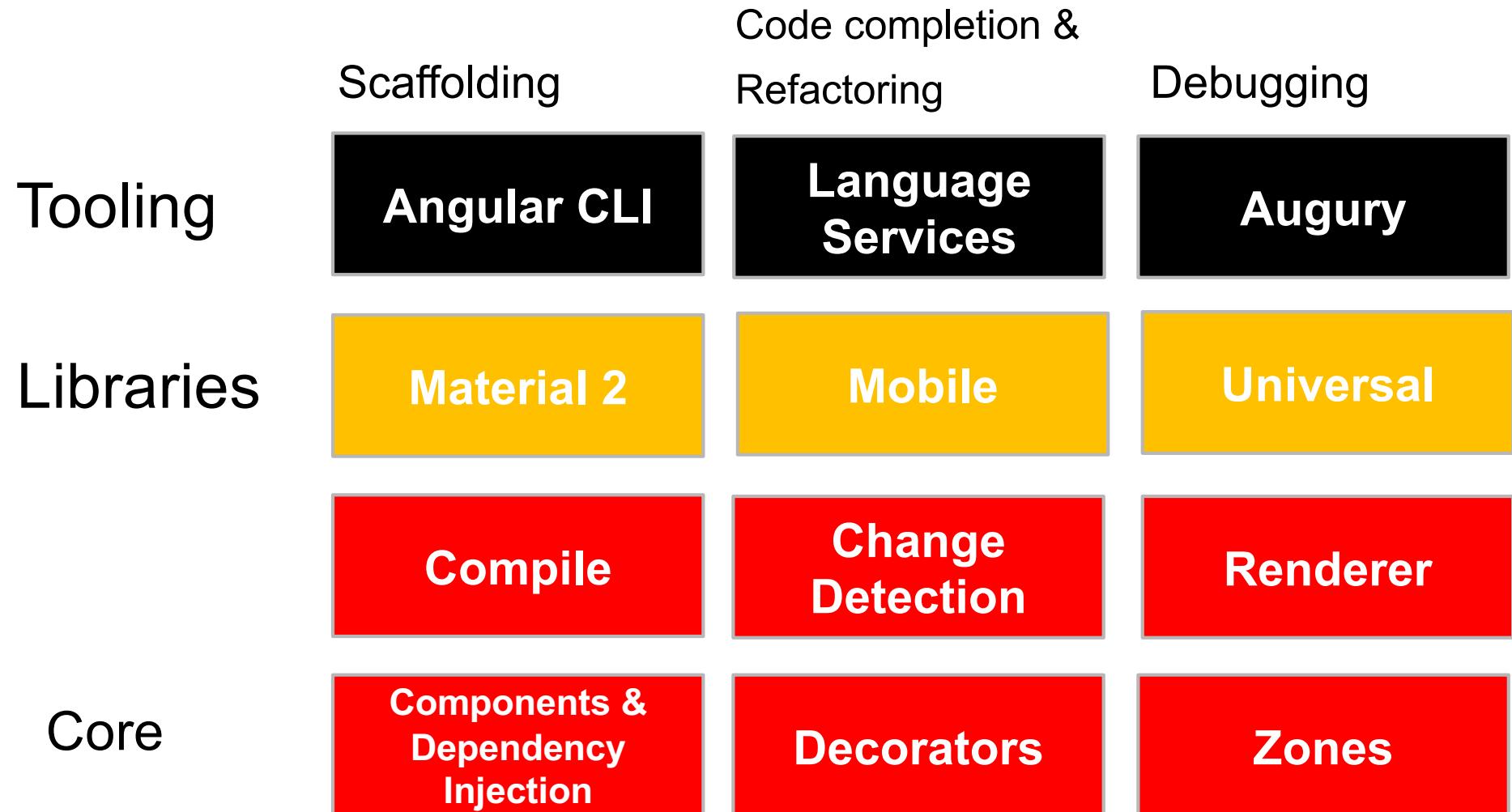
MV*

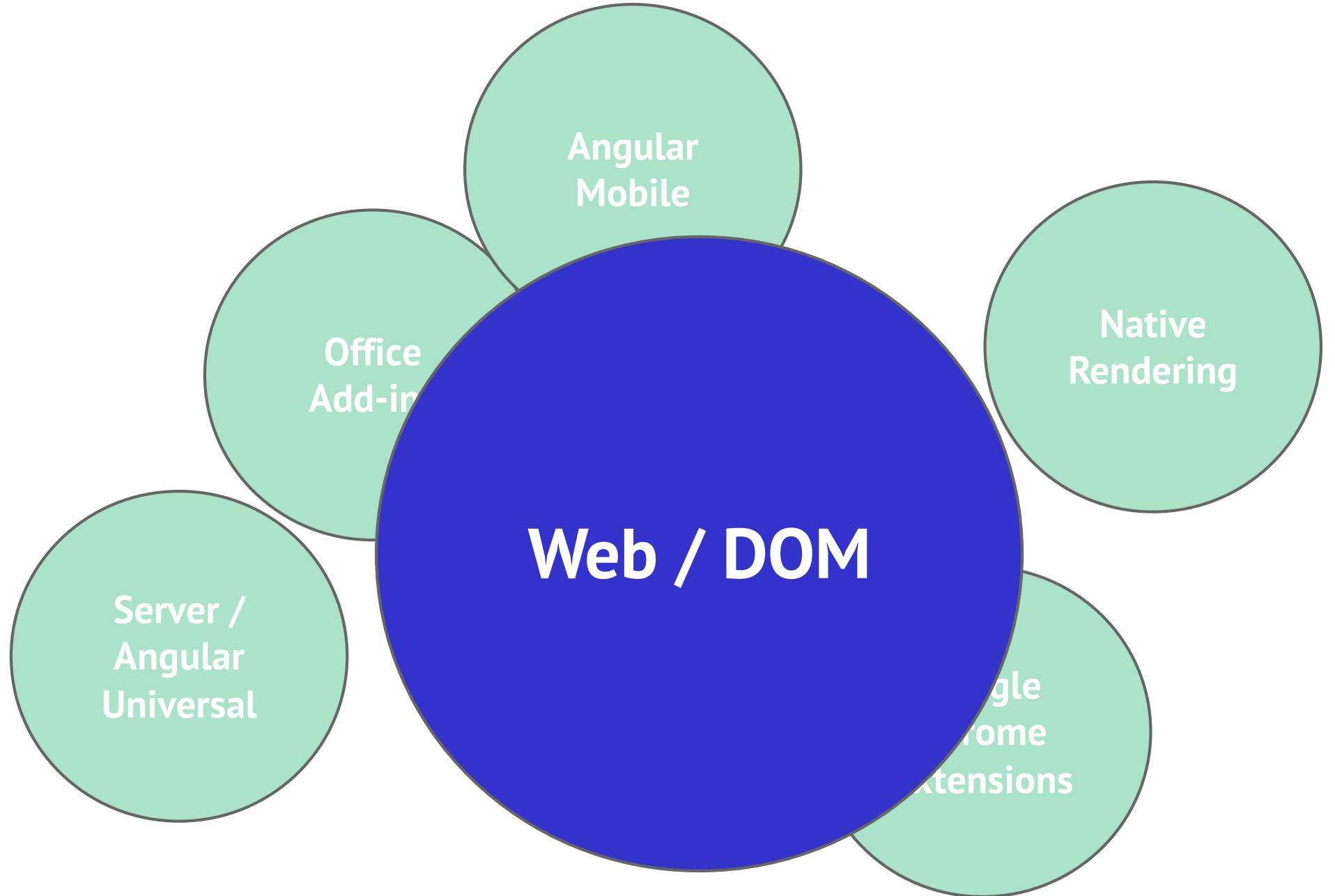
Framework



Platform

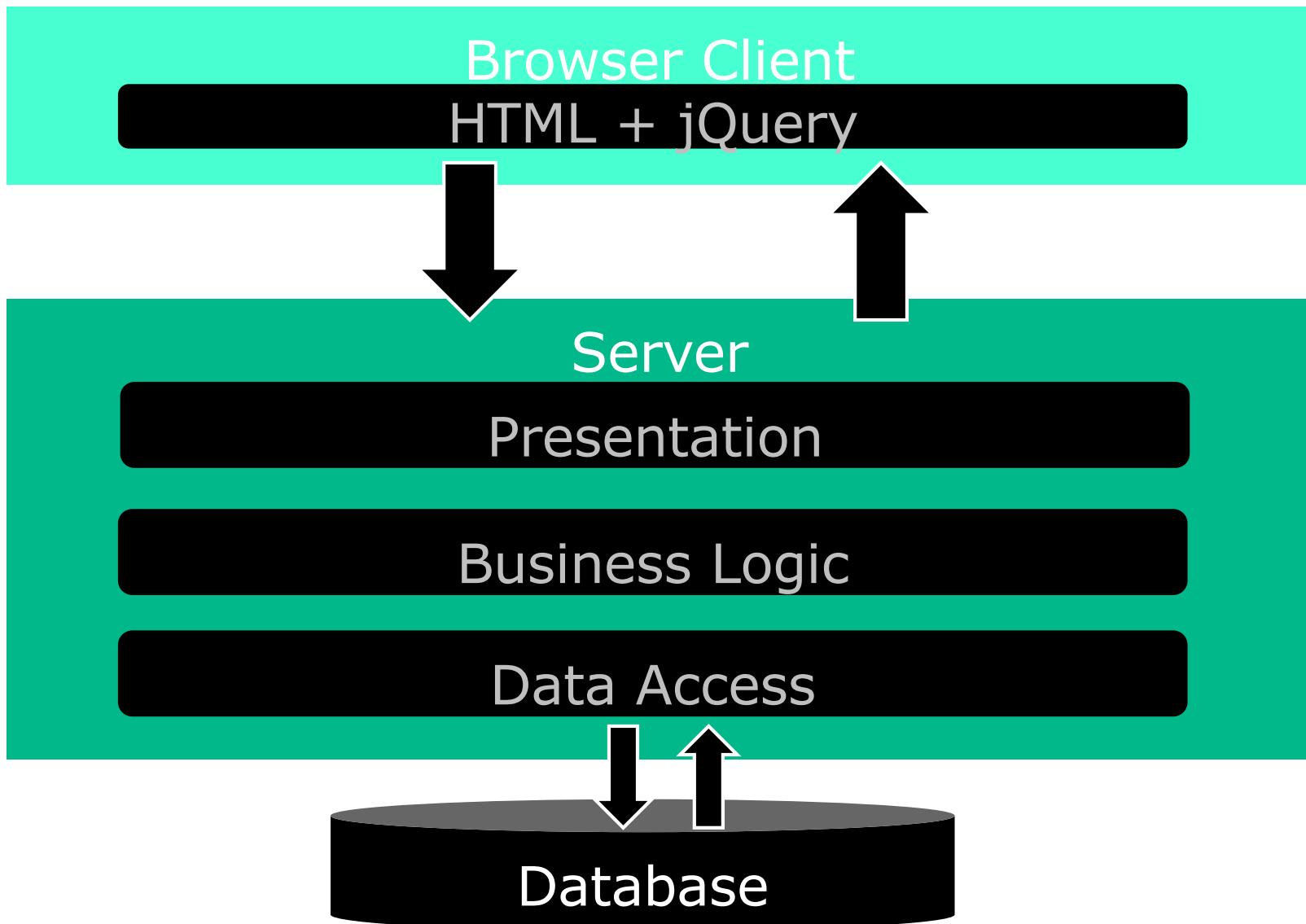
Framework to Platform



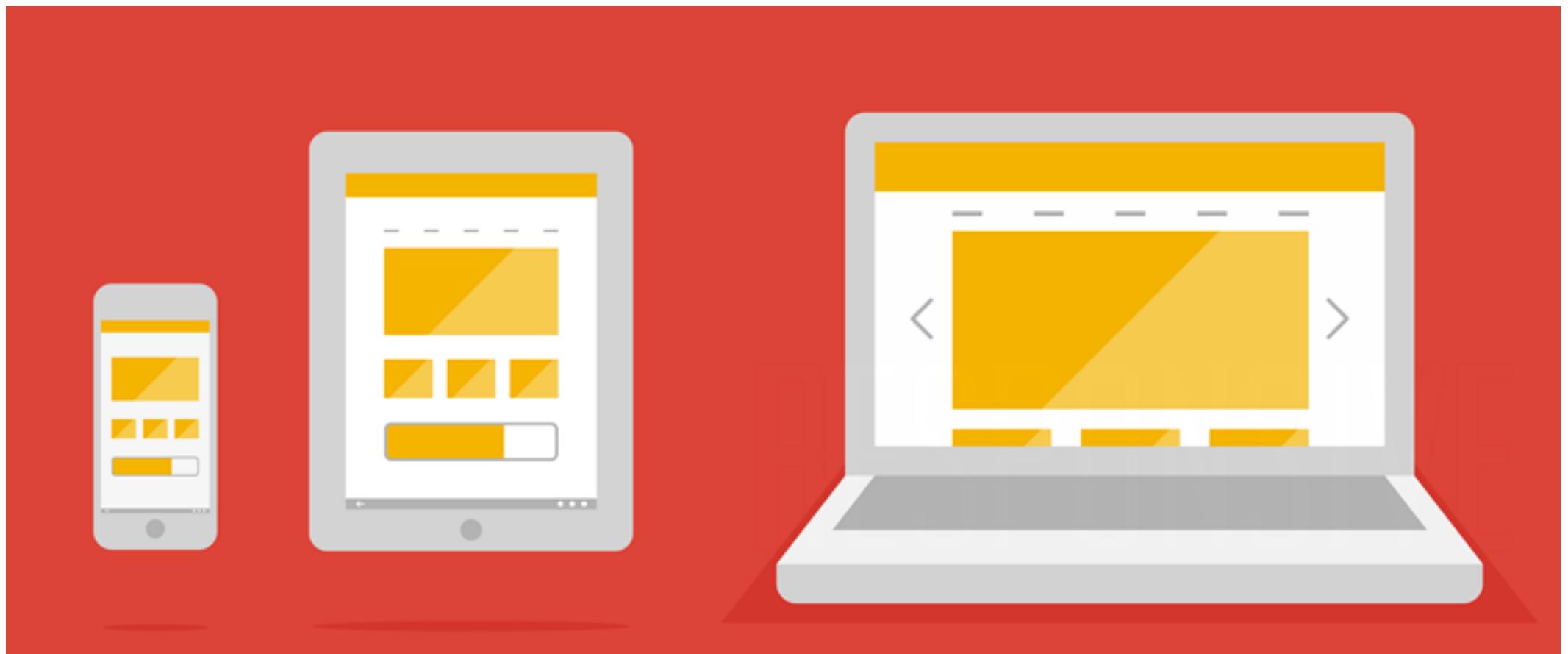


Conventional Web App

2000 - 2013

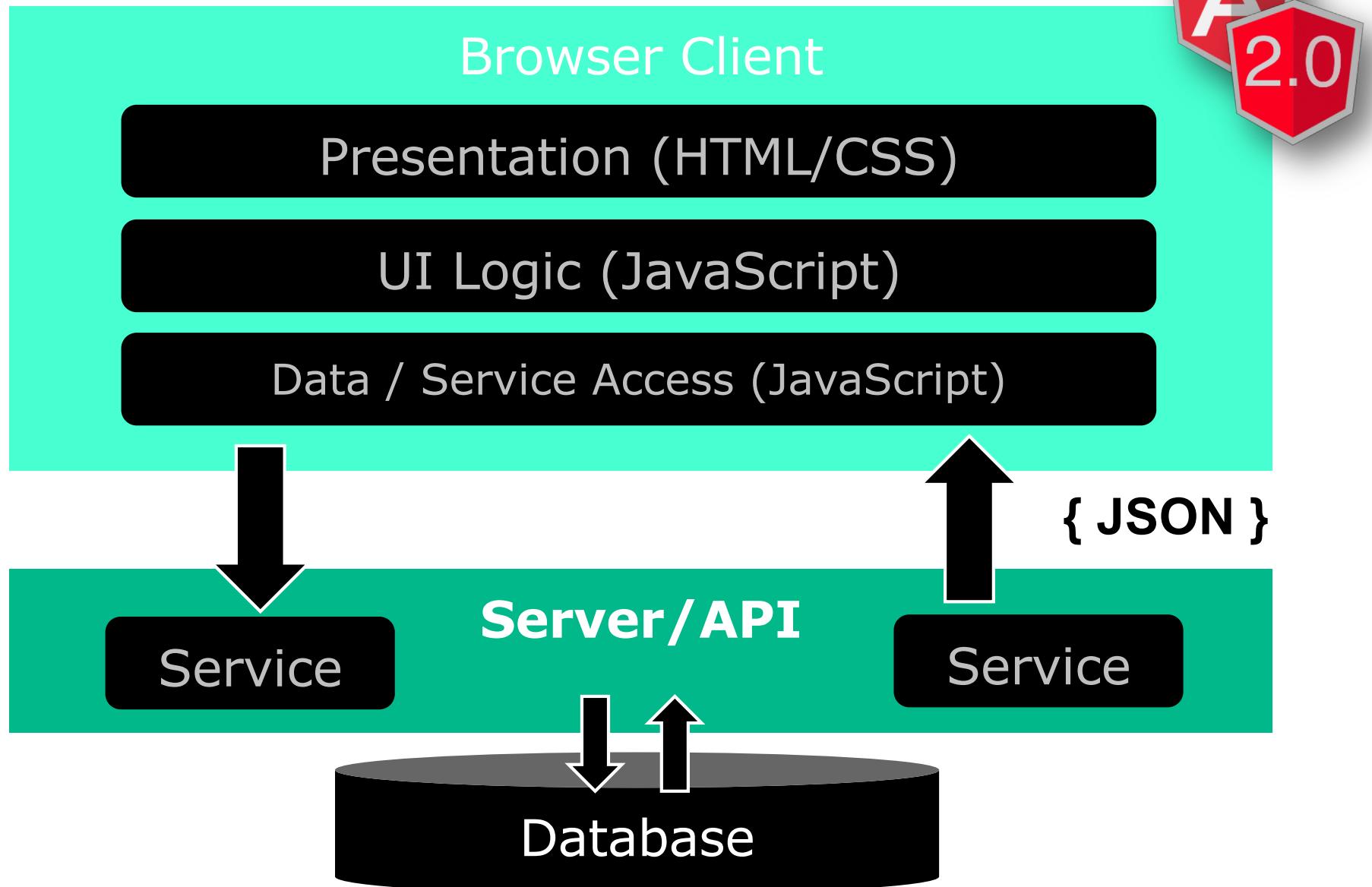


However, ca. 2010:



Single Page Application

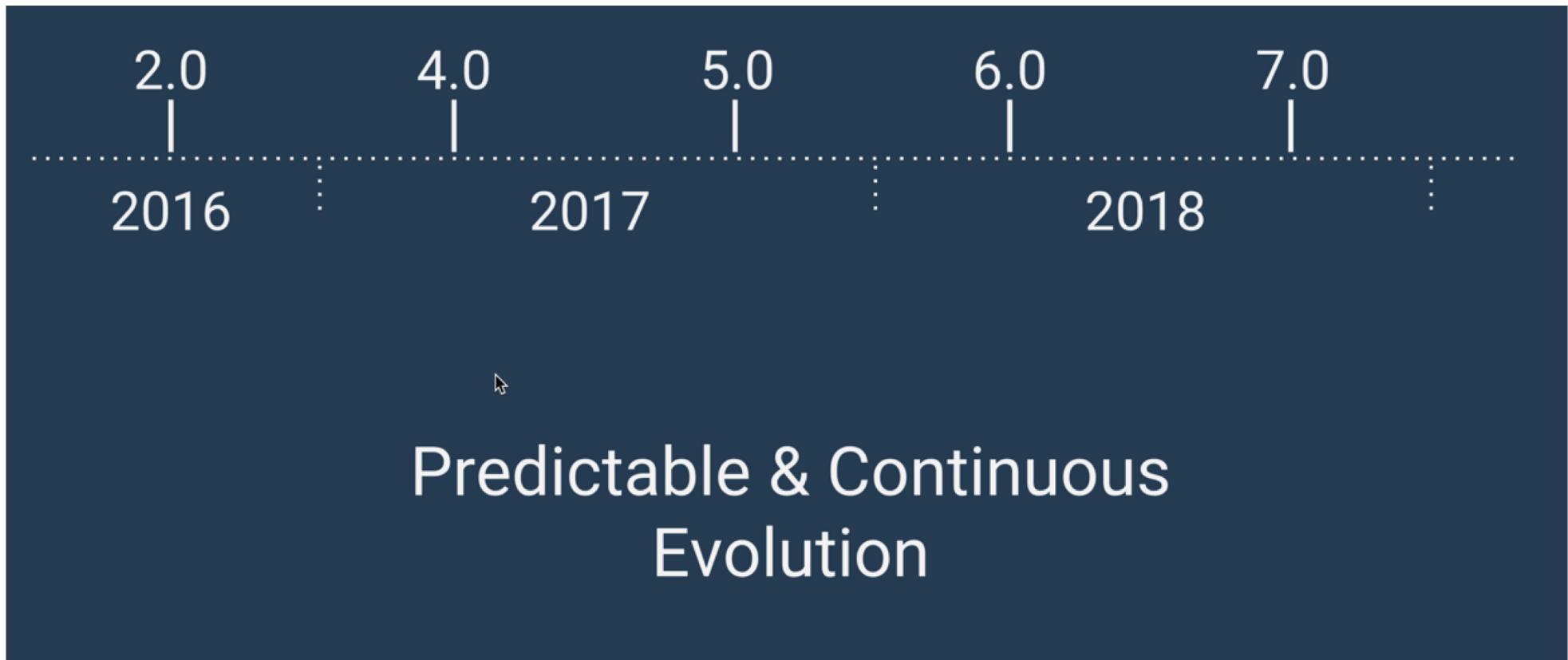
2010 – 20??

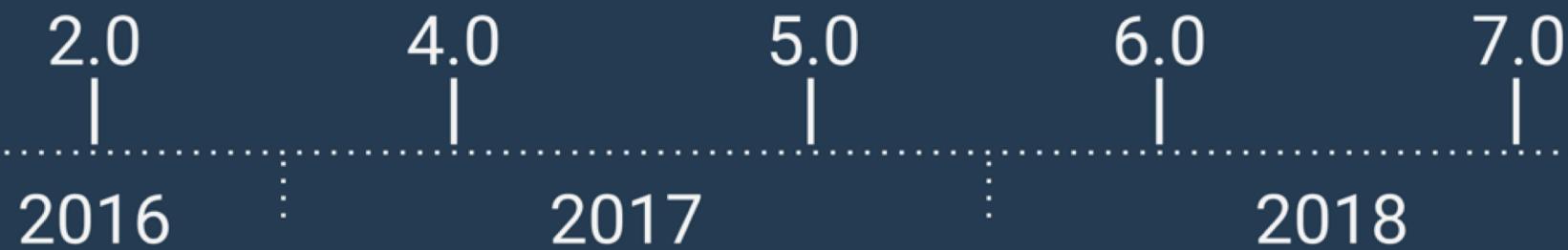


On Versioning Numbering



Planned – six months major release cycle





- Predictability
- Painless Updates
- Long Term Support

Angular Versions and -Long Time Support

→ <https://angular.io/guide/releases>

The screenshot shows the Angular documentation page for releases. The left sidebar has sections like Getting Started, Tutorial, Fundamentals, Techniques (expanded), Internationalization (i18n), Language Service, Security, Setup & Deployment, Service Workers, Keeping Up-to-Date (expanded), Updating Your Projects, Angular Releases, Upgrading from AngularJS, Server-side Rendering, Visual Studio 2015 QuickStart, Style Guide, Glossary, API, and a stable branch (v6.0.4). The main content area discusses support frequency and LTS periods. It includes a table for versions 4.0.0, 5.0.0, and 6.0.0, and a section on deprecation practices.

• 6 months of active support, during which regularly-scheduled updates and patches are released, as described above in [Release frequency](#).

• 12 months of long-term support (LTS). During the LTS period, only critical fixes and security patches will be released.

The following table provides the support status and key dates for Angular version 4.0.0 and higher.

Version	Status	Release Date	LTS Start Date	LTS End Date
^4.0.0	LTS	March 23, 2017	September 23, 2017	September 23, 2018
^5.0.0	LTS	November 1, 2017	May 1, 2018	May 1, 2019
^6.0.0	Active	May 3, 2018	November 3, 2018	November 3, 2019

Deprecation practices

Sometimes "breaking changes", such as the removal of support for select APIs and features, are necessary to innovate and stay current with new best practices, changing dependencies, or changes in the (web) platform itself.

To make these transitions as easy as possible, we make two commitments to you:

- We work hard to minimize the number of breaking changes and to provide migration tools when possible.
- We follow the deprecation policy described here, so you have time to update your apps to the latest APIs and best practices.

<https://update.angular.io/>

Angular Update Guide

Select the options matching your project:

Angular Version

4.0 6.0

App Complexity

Basic Medium Advanced

ngUpgrade

I use ngUpgrade

Package Manager

npm yarn

Show me how to update!

Warning: We do not recommend moving across multiple major versions.

"It's just

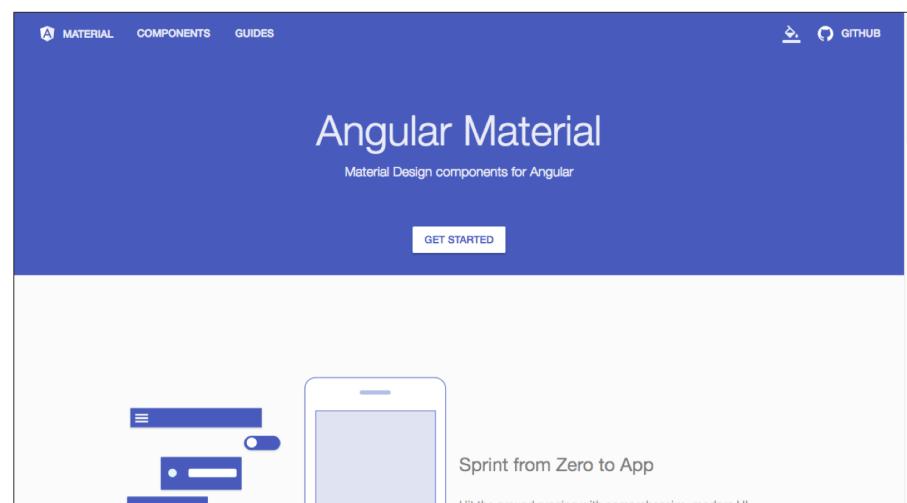
Angular

Angular as a Platform



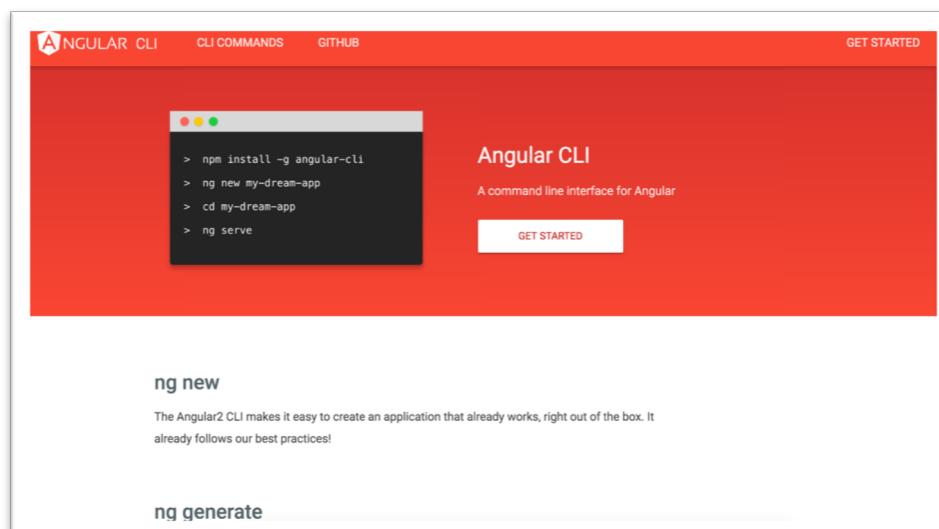
The screenshot shows the main Angular website. At the top, there's a navigation bar with links for FEATURES, DOCS, RESOURCES, EVENTS, and BLOG. A search bar is located at the top right. The main content area features a large red hexagon with a white 'A' logo. Below it, the text "One framework. Mobile & desktop." is displayed, along with a "GET STARTED" button. In the bottom left corner, there's a callout for "ANGULAR MIX" with the text "Join us at our newest event, October 2017" and a "LEARN MORE" button.

<https://angular.io/>



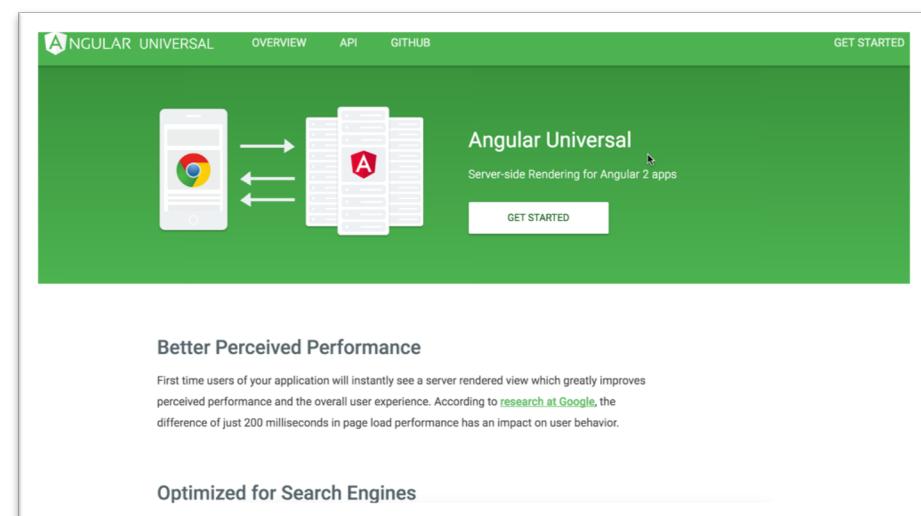
The screenshot shows the Angular Material website. The top navigation bar includes links for MATERIAL, COMPONENTS, and GUIDES, along with a GitHub icon. The main title "Angular Material" is prominently displayed, followed by the subtitle "Material Design components for Angular". A "GET STARTED" button is visible. The page features illustrations of a smartphone and a laptop, with the text "Sprint from Zero to App" and "Hit the ground running with comprehensive, modern UI".

<https://material.angular.io/>



The screenshot shows the Angular CLI website. The top navigation bar has links for ANGULAR CLI, CLI COMMANDS, and GITHUB. A "GET STARTED" button is at the top right. The main section features a terminal window showing command examples: "npm install -g angular-cli", "ng new my-dream-app", "cd my-dream-app", and "ng serve". To the right, the text "Angular CLI" and "A command line interface for Angular" is displayed, with another "GET STARTED" button. Below this, sections for "ng new" and "ng generate" provide descriptions of their respective commands.

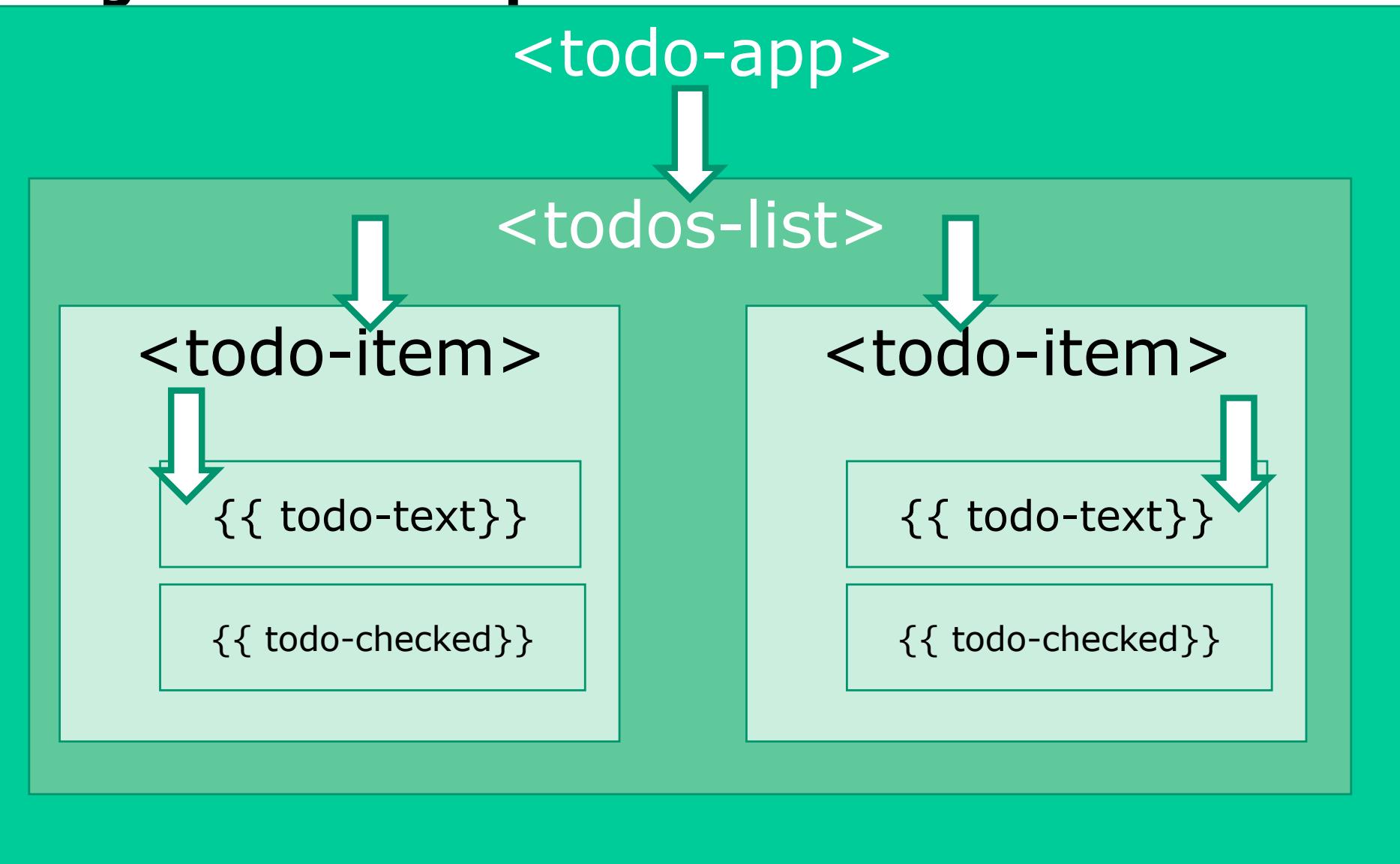
<https://cli.angular.io/>



The screenshot shows the Angular Universal website. The top navigation bar includes links for ANGULAR UNIVERSAL, OVERVIEW, API, and GITHUB, along with a GitHub icon. The main title "Angular Universal" is shown with the subtitle "Server-side Rendering for Angular 2 apps". A diagram illustrates the process where a browser (Chrome) communicates with a server, which then renders the application. Below this, sections discuss "Better Perceived Performance" and "Optimized for Search Engines".

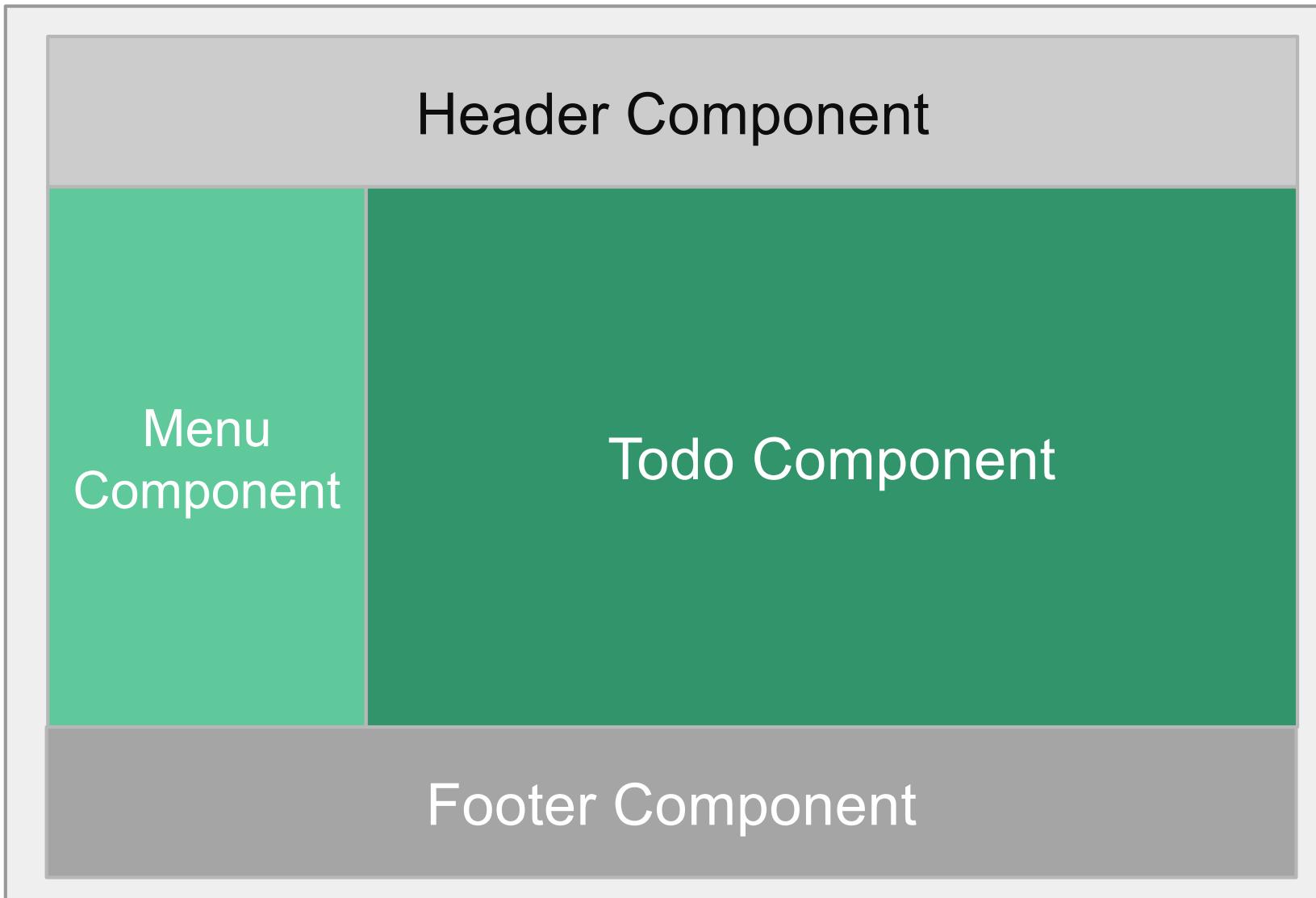
<https://universal.angular.io/>

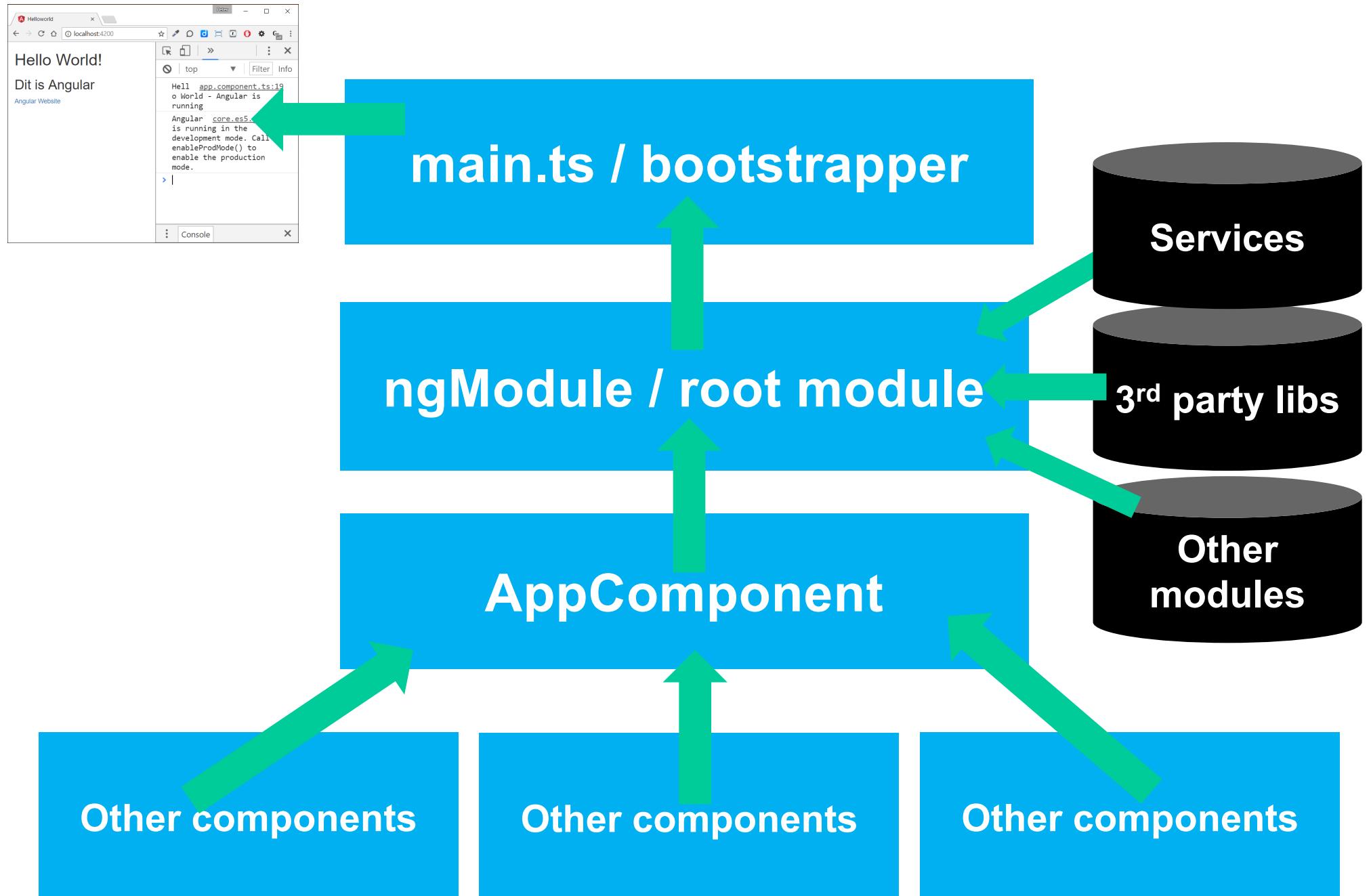
Angular 2 - components

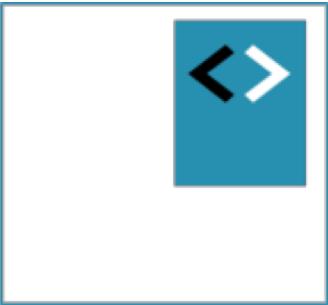


*"An Angular-app is a
tree
of components"*

Components – visually







Let's write some code

Hello World in Angular

Angular 1:

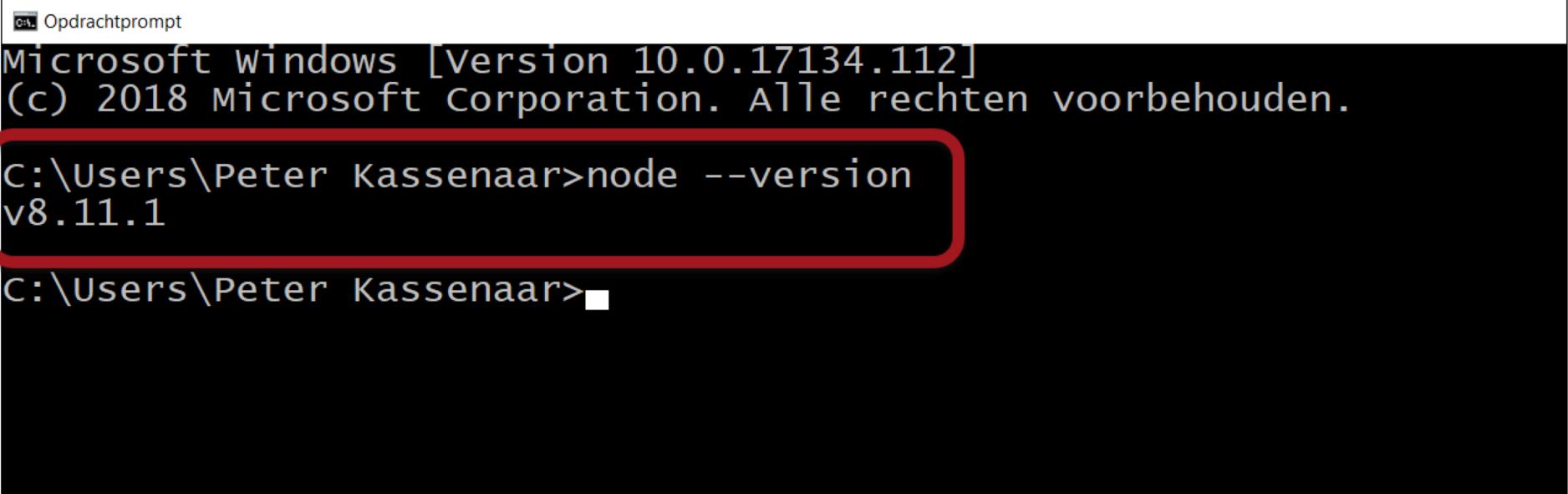
```
<script src="angular.min.js"></script>
```

Angular development dependency: NodeJS 8.0+



The screenshot shows the official Node.js website. At the top, there's a dark navigation bar with the Node.js logo and links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, NEWS, and FOUNDATION. The FOUNDATION link is highlighted with a green background. Below the navigation, a large text area states: "Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine." A green callout box says "Important June 2018 security upgrades now available". The main content area features a "Download for Windows (x64)" heading. It offers two options: "8.11.3 LTS" (Recommended For Most Users) and "10.6.0 Current" (Latest Features). Below these are links for "Other Downloads | Changelog | API Docs" for each version. Further down, it says "Or have a look at the [Long Term Support \(LTS\) schedule](#)." A call to action encourages users to "Sign up for [Node.js Everywhere](#), the official Node.js Weekly Newsletter." At the bottom, there's a footer with the Linux Foundation logo, a link to "COLLABORATIVE PROJECTS", and links for reporting issues and getting help. It also includes a copyright notice: "© Node.js Foundation. All Rights Reserved. Portions of this site originally © Joyent."

Node – check your version



A screenshot of a Windows Command Prompt window titled "Opdrachtprompt". The window shows the following text:

```
Microsoft Windows [Version 10.0.17134.112]
(c) 2018 Microsoft Corporation. Alle rechten voorbehouden.

C:\Users\Peter Kassenaar>node --version
v8.11.1

C:\Users\Peter Kassenaar>
```

The command "node --version" and its output "v8.11.1" are highlighted with a red rounded rectangle.

Exercise

- Download or clone
<https://github.com/PeterKassenaar/voorbeeldenAngular2>
- Unpack the repository and cd into /voorbeeldenAngular-master

```
cd examples
```

```
cd 100-helloworld
```

```
npm install
```

```
npm start
```

- Go to browser: <http://localhost:4200>

The screenshot shows a browser window displaying the text "Hello World!" and "This is Angular". Below the browser is a code editor interface.

Browser View:

- Address bar: localhost:4200
- Content:
 - Header: Hello World!
 - Text: This is Angular
 - Link: Angular Website

Code Editor View:

- Project Explorer (left):
 - Project: voorbeeldenAngular2
 - Folder: Examples
 - Folder: 100-helloworld
 - File: node_modules (library root)
 - Folder: src
 - Folder: app
 - File: app.component.ts
 - File: app.module.ts
 - Folder: assets
 - Folder: environments
 - File: favicon.ico
 - File: index.html
 - File: main.ts
 - File: polyfills.ts
 - File: styles.css
 - File: tsconfig.app.json
 - File: angulardoc.json
 - File: .gitignore
 - File: angular.json
 - File: package.json
 - File: package-lock.json
 - File: tsconfig.json
 - File: yarn.lock

The folder structure under 'src/app' is highlighted with a red rounded rectangle.

Code Editor (right):

```
1 import {Component, OnInit} from '@angular/core'
2
3 @Component({
4   selector: 'hello-world',
5   template: `
6     <h1>Hello World!</h1>
7     <h2>This is Angular</h2>
8     <a href="http://angular.com">Angular Website</a>
9   `,
10 })
11
12 export class AppComponent implements OnInit {
13   // optional: add constructor()
14   constructor() {
15   }
16 }
```

Boilerplate code for Hello World

Steps

1. Set up environment, boilerplate & libraries
2. Write Angular Root Component
3. Write @NgModule Component
4. Bootstrap module
5. Write HTML-pagina (`index.html`)



Boilerplate files #1 - package.json

```
{  
  "name": "hello-angular",  
  "description": "Voorbeeldproject bij de training Angular (C) - info@kassenaar.com",  
  "version": "0.0.1",  
  "license": "MIT",  
  "scripts": {  
    "ng": "ng",  
    "start": "ng serve",  
    "build": "ng build",  
  },  
  "private": true,  
  "dependencies": {  
    "@angular/animations": "7.0.0",  
    "@angular/common": "7.0.0",  
    "@angular/compiler": "7.0.0",  
    "@angular/core": "7.0.0",  
    "@angular/forms": "7.0.0",  
    "rxjs": "^6.1.0",  
    "zone.js": "^0.8.26"  
  },  
  "devDependencies": {  
    "@angular-devkit/build-angular": "~0.6.0",  
    "@angular/cli": "6.0.0",  
    "typescript": "3.2.1"  
  },  
  "author": "Peter Kassenaar <info@kassenaar.com>"  
}
```

Boilerplate files #2 - tsconfig.json

```
{  
  "compileOnSave": false,  
  "compilerOptions": {  
    "outDir": "./dist/out-tsc",  
    "baseUrl": "src",  
    "sourceMap": true,  
    "declaration": false,  
    "moduleResolution": "node",  
    "emitDecoratorMetadata": true,  
    "experimentalDecorators": true,  
    "target": "es5",  
    "typeRoots": [  
      "node_modules/@types"  
    ],  
    "lib": [  
      "es2016",  
      "dom"  
    ]  
  }  
}
```

Boilerplate files #3 - angular.json

```
{  
  "$schema": "./node_modules/@angular/cli/lib/config/schema.json",  
  "version": 1,  
  "newProjectRoot": "projects",  
  "projects": {  
    "helloworld": {  
      "root": "",  
      "sourceRoot": "src",  
      "projectType": "application",  
      "architect": {  
        "build": {  
          "builder": "@angular-devkit/build-angular:browser",  
          "options": {  
            "outputPath": "dist",  
            "index": "src/index.html",  
            "main": "src/main.ts",  
            "tsConfig": "src/tsconfig.app.json",  
            ...  
          }  
        }  
      }  
    }  
  }  
}
```

"Nice to have" - non-essential files

The screenshot shows a GitHub repository page for 'angular / quickstart'. The repository has 250 forks and 2,567 stars. The 'Code' tab is selected. A commit by 'wardbell' is shown, adding 'bs-config.e2e.json' to the file. The file contains a list of non-essential files:

```
1 .git  
2 .gitignore  
3 .travis.yml  
4 *.spec*.ts  
5 bs-config.e2e.json  
6 CHANGELOG.md  
7 e2e  
8 favicon.ico  
9 karma.conf.js  
10 karma-test-shim.js  
11 LICENSE  
12 non-essential-files.txt  
13 non-essential-files.osx.txt  
14 protractor.config.js  
15 README.md
```

<https://github.com/angular/quickstart/blob/master/non-essential-files.txt>

Step 2 – Component

Convention - components in directory /src/app

Or: edit in angular.json

Filename: src/app/app.component.ts

```
import {Component} from '@angular/core';
@Component({
  selector: 'hello-world',
  template: '<h1>Hello Angular</h1>'
})
export class AppComponent { }
```

Step 3 – @NgModule

Convention - filename: /src/app.module.ts

```
// Angular Modules
import {NgModule}      from '@angular/core';
import {BrowserModule} from '@angular/platform-browser';

// Custom Components
import {AppComponent} from './app.component';

// Module declaration
@NgModule({
  imports      : [BrowserModule],
  declarations: [AppComponent],
  bootstrap    : [AppComponent]
})
export class AppModule {
```

Root Module of the application

Some background info on Root Module

The screenshot shows a blog post titled "Introducing Angular Modules - Root Module" by John Papa. The post is dated September 5, 2016. It discusses the `@NgModule` decorator and its purpose in organizing Angular applications. To the right, there is a sidebar featuring a bio of John Papa and links to his most recent posts.

Introducing Angular Modules - Root Module

05 SEPTEMBER 2016

The `@NgModule` is a new decorator that has recently been added in Angular 2. `NgModule` defines an Angular Module, which (from the official docs) are defined as "Angular Modules help organize an application into cohesive blocks of functionality."

John

Hi, I'm John Papa. I author this blog, create courses for Pluralsight and am a Google Developer Expert and Microsoft Regional Director. I speak at events and I train technology thought leaders →

Most Recent

- [Introducing Angular Modules - Root Module](#)
- [Learning Angular 2 this Fall](#)
- [The Ultimate Angular 2 Workshop in Ft](#)

<https://johnpapa.net/introducing-angular-modules-root-module/>

Step 4 - bootstrap component

Best practice: bootstrap app in separate component

Convention: main.ts, **of** app.main.ts.

```
import {enableProdMode} from '@angular/core';
import {platformBrowserDynamic} from '@angular/platform-browser-dynamic';

import {AppModule} from './app/app.module';
import {environment} from './environments/environment';

if (environment.production) {
  enableProdMode();
}

platformBrowserDynamic().bootstrapModule(AppModule);
```

Step 5 – index.html

index.html - simple HTML file - expanded at runtime by WebPack

```
<html>

<head>
  <meta charset="utf-8">
  <title>Helloworld</title>
  <base href="/">

  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="icon" type="image/x-icon" href="favicon.ico">
</head>
```

Body of index.html

Element reference (selector) of root-component:

```
<body>  
  <hello-world>  
    loading...  
  </hello-world>  
</body>
```

Run the app

npm start - run start script from package.json.

ng serve - start global angular-cli instance

The terminal window shows the following output:

```
Terminal
+ chunk {0} polyfills.bundle.js, polyfills.bundle.js.map (polyfills) 157 kB {4} [initial] [rendered]
x
chunk {1} main.bundle.js, main.bundle.js.map (main) 3.73 kB {3} [initial] [rendered]
chunk {2} styles.bundle.js, styles.bundle.js.map (styles) 9.78 kB {4} [initial] [rendered]
chunk {3} vendor.bundle.js, vendor.bundle.js.map (vendor) 2.1 MB [initial] [rendered]
chunk {4} main.bundle.js, main.bundle.js.map (main) 3.73 kB {1} [initial] [rendered]
webpack: C:\Users\...{[redacted]}
```

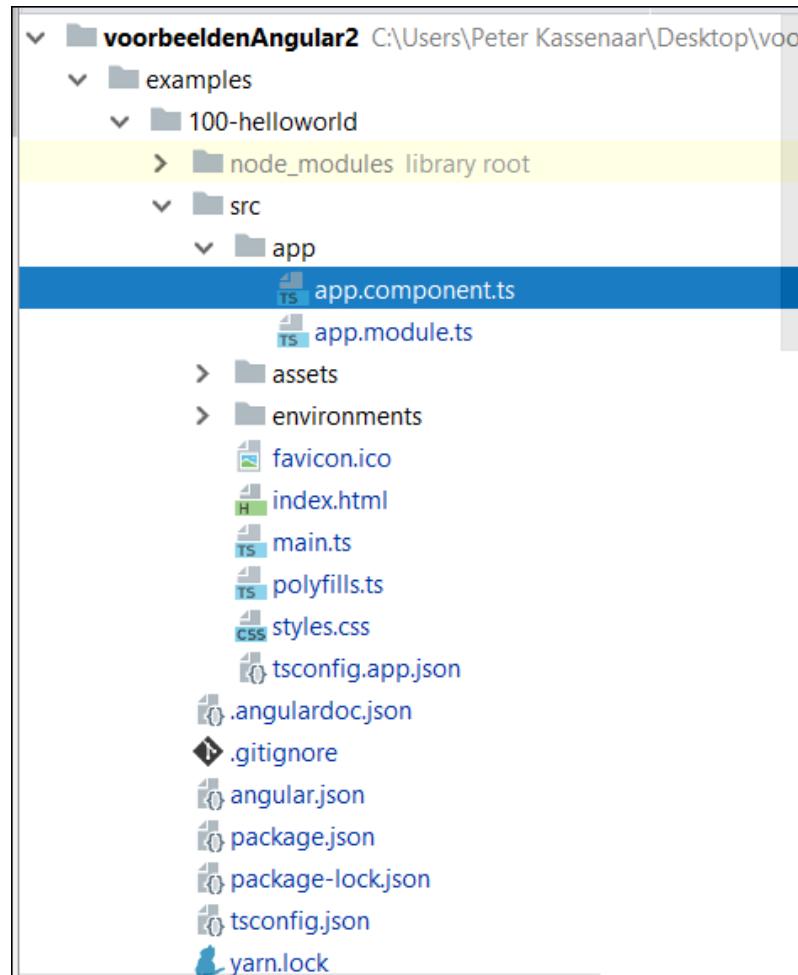
The browser screenshot shows a page titled "Hello World!" with the subtext "Dit is Angular" and "Angular Website". The developer tools console tab shows the following logs:

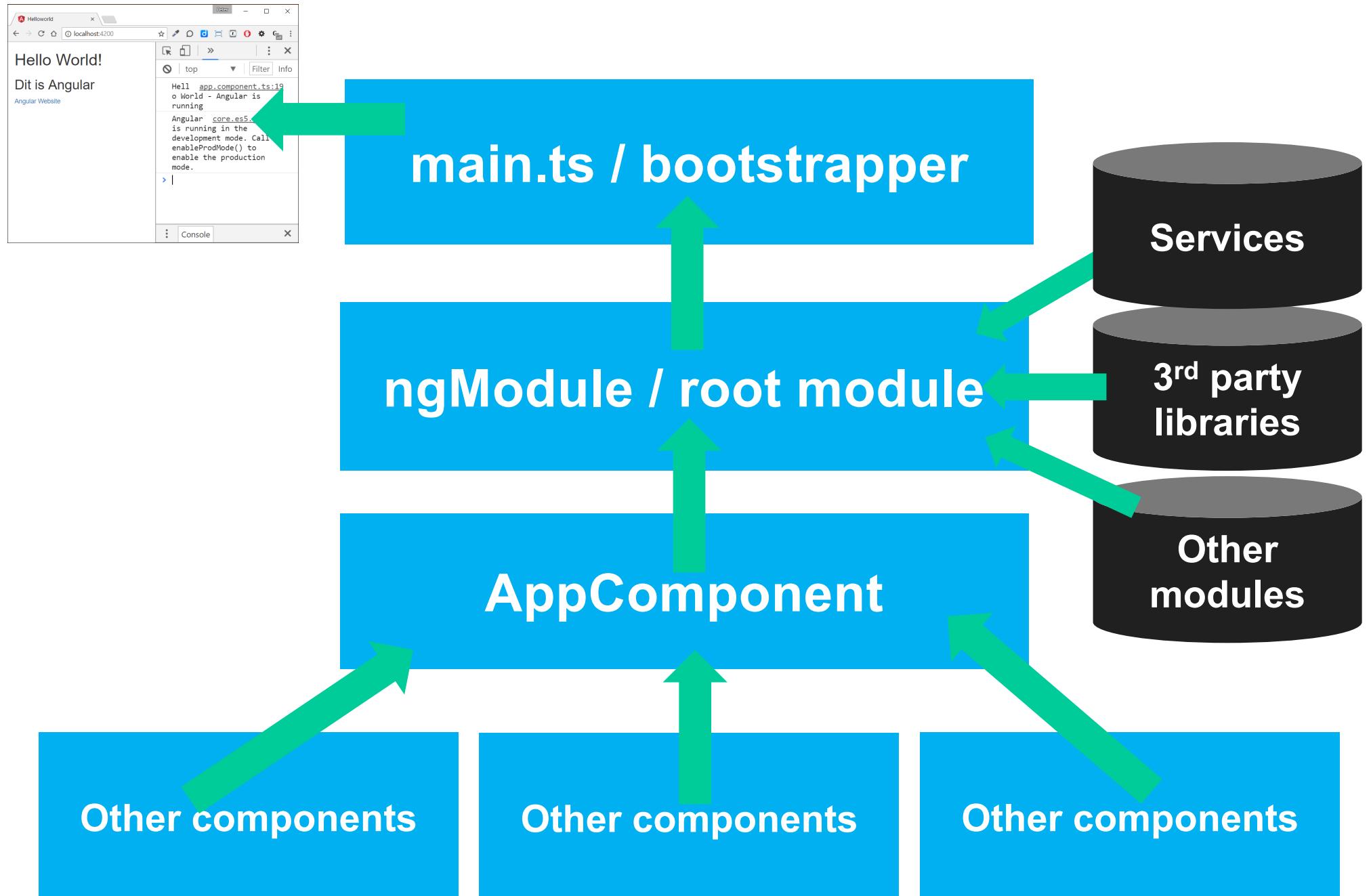
```
localhost:4200
Elements Console Sources Network Performance Memory > | : X
Filter Info
Hello World - Angular is running
Angular is running in the development mode. Call enableProdMode() to enable the production mode.
app.component.ts:19
core.es5.js:3025
> |
```

After that: edit app.component.ts

– Automagically refreshed through Live Reload

So, a Basic Project Structure and Architecture





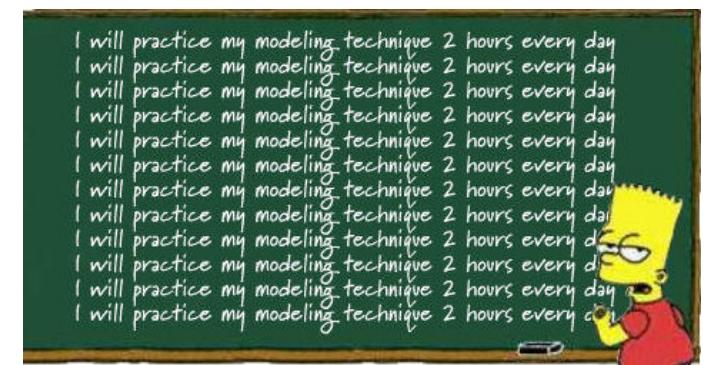
Checkpoint

You need a lot of boilerplate code to start an Angular project.

(At least) Five steps:

1. Set up environment, boilerplate & libraries
2. Write Angular Root Component for app
3. Bootstrap component (`main.ts`)
4. write HTML-pagina (`index.html`)
5. Run the app : `npm start`

Then: work on your components, services, etc.

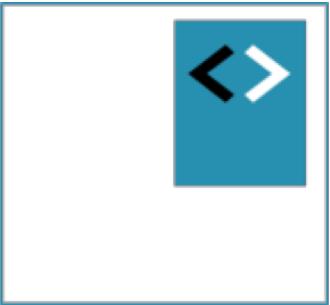


Exercise....

Assets

<https://github.com/PeterKassenaar/oce>

Exercises and example code



Tooling - Angular CLI

Quickly set up new projects
via command line interface

Angular-CLI to the rescue

- It *is* possible to start new Angular projects from scratch
- But by using the CLI it is *much* simpler
- CLI-options:
 - Scaffolding
 - Generating
 - Testing
 - Building
 - AOT-Compiling
 - ...

Scaffolding - Angular CLI

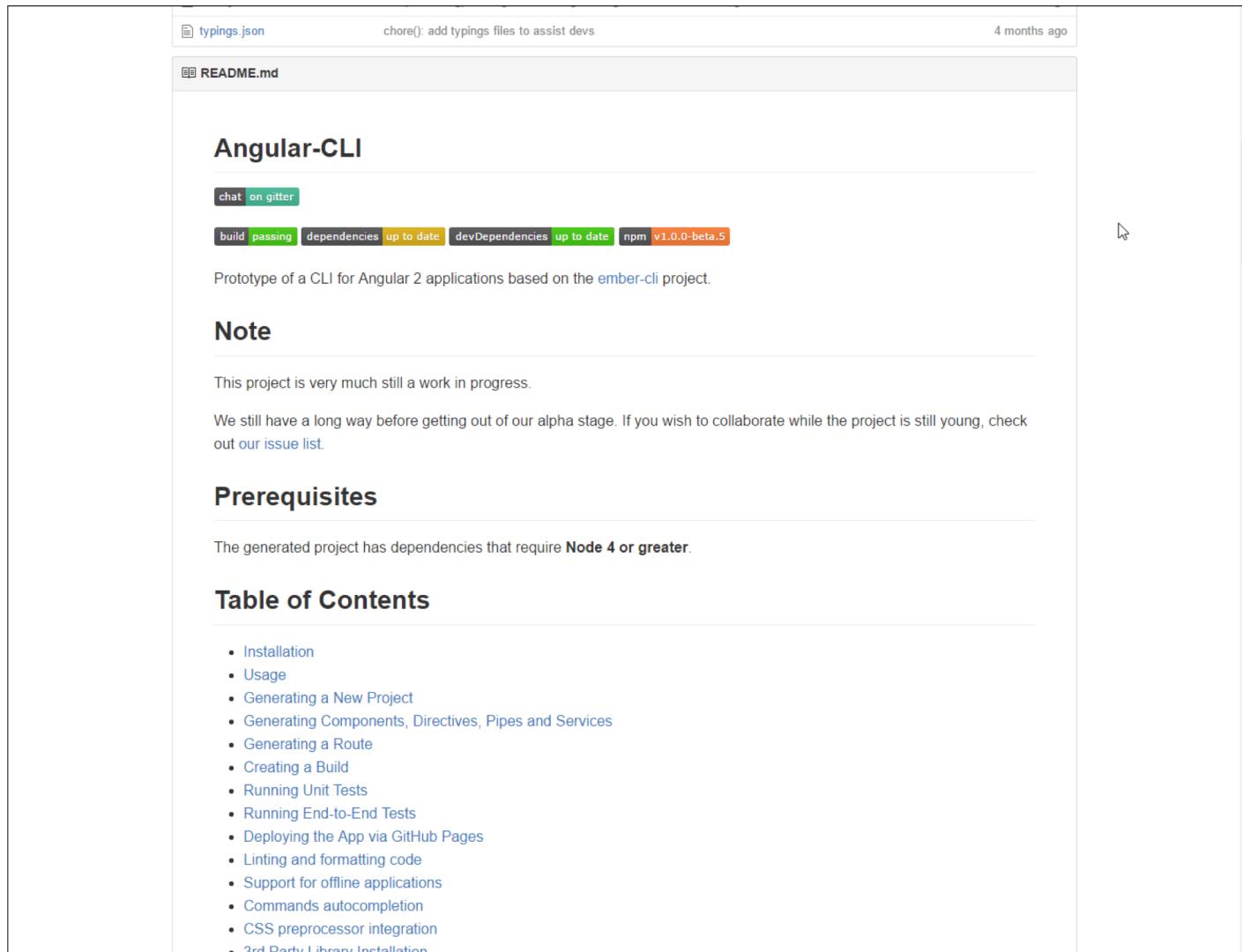
First : install CLI globally

<https://github.com/angular/angular-cli>

en

<https://cli.angular.io/>

```
npm install -g @angular/cli
```



A screenshot of the Angular-CLI GitHub repository's README page. The page includes a file tree header with 'typings.json' and 'chore(): add typings files to assist devs' (4 months ago). Below the header is a 'README.md' section containing the following content:

Angular-CLI

[chat on gitter](#)

[build passing](#) [dependencies up to date](#) [devDependencies up to date](#) [npm v1.0.0-beta.5](#)

Prototype of a CLI for Angular 2 applications based on the [ember-cli](#) project.

Note

This project is very much still a work in progress.

We still have a long way before getting out of our alpha stage. If you wish to collaborate while the project is still young, check out [our issue list](#).

Prerequisites

The generated project has dependencies that require [Node 4 or greater](#).

Table of Contents

- Installation
- Usage
- Generating a New Project
- Generating Components, Directives, Pipes and Services
- Generating a Route
- Creating a Build
- Running Unit Tests
- Running End-to-End Tests
- Deploying the App via GitHub Pages
- Linting and formatting code
- Support for offline applications
- Commands autocomplete
- CSS preprocessor integration
- [3rd Party Library Installation](#)

```
npm install -g @angular/cli
```

[DOCUMENTATION](#)[GITHUB](#)[GET STARTED](#)

```
> npm install -g @angular/cli
> ng new my-dream-app
> cd my-dream-app
> ng serve
```

Angular CLI

A command line interface for Angular

[GET STARTED](#)

ng new

The Angular CLI makes it easy to create an application that already works, right out of the box. It already follows our best practices!

ng generate

Generate components, routes, services and pipes with a simple command. The CLI will also create

```
ng --version
PeterKassenaar
  / \ - - - - - | | - - - - - | | | | | | | |
  / Δ\ | ' \ / ' | | | | | | | | | | | | | |
  / ____\ | | | | | | | | | | | | | | | | | |
  / / \ \ \ | | | | | | | | | | | | | | | | | |
  |__|
```

Angular CLI: 7.0.4
Node: 10.13.0
OS: darwin x64
Angular:
...

Package Version

@angular-devkit/architect 0.10.4
@angular-devkit/core 7.0.4
@angular-devkit/schematics 7.0.4
@schematics/angular 7.0.4
@schematics/update 0.10.4
rxjs 6.3.3
typescript 3.1.3

MacBook-Pro:~ PeterKassenaar\$

Background info



<https://www.youtube.com/watch?v=wHZe6gGI5RY>

Main commands

```
ng new PROJECT_NAME
```

```
cd PROJECT_NAME
```

```
ng serve
```

Project is served on <http://localhost:4200>

More info

The screenshot shows the Scotch.io website. The left sidebar has a dark background with the Scotch logo (a glass with orange and yellow stones) and the text "scotch.io code on the rocks". It includes sections for "LEARN" (Tutorials, Bar Talk, Community), "Scotch School", and "OUR STUFF" (About, Shop, Join Us On Slack, Hire Us). The main content area has a red header with the Angular logo and the title "Use the Angular CLI for Faster Projects". Below the header, there's a large image of a person working at a computer. The text "Use the Angular CLI For Faster Angular 2 Projects" is displayed, followed by the subtext "Creating Angular 2 projects has never been faster than with the Angular CLI". The footer features a "Popular Scotchers" section with profiles of nine people, each with a small profile picture and their name: Chris Sevilleja, Nicholas Cerminara, Holly Lloyd, Ado Kukic, Ryan Chenkie, Ken Wheeler, Alex Sears, Chris Nwamba, and a link to "Become a Scotcher!". The top right of the page has a search bar, login links, and a "More" menu.

<https://scotch.io/tutorials/use-the-angular-cli-for-faster-angular-2-projects>

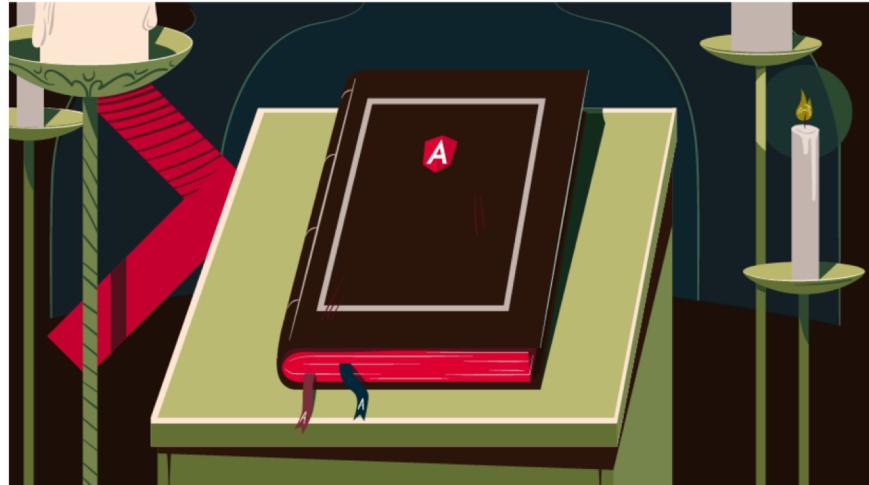
[≡](#)  sitepoint

[Books & Courses](#) [Community](#) [Login](#) [Create Free Account](#)

HTML & CSS **JAVASCRIPT** PHP DESIGN & UX WEB WORDPRESS SEO BY WOORANK MICROSOFT TECH

JavaScript - April 25, 2017 - By Jurgen Van de Moere

The Ultimate Angular CLI Reference Guide



[f](#) [in](#) [t](#)

2017.04.25: As of March 24, Angular CLI v1.0 was released. This article has been updated to reflect the latest changes. If you want to add the latest features of Angular CLI v1.0 to your existing Angular project that was generated with an earlier version of Angular CLI, check out the [Angular CLI v1.0 migration guide](#).

2017.02.17: As of February 9, 2017, the `ng deploy` command has been removed

Related Topics:

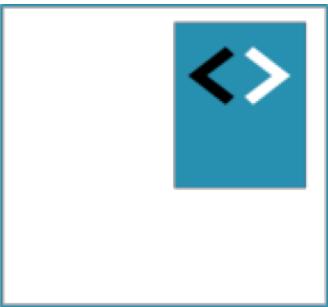
[React](#) [Tools & Libraries](#)
[Node.js](#) [Raw JavaScript](#)
[npm](#) [More...](#)

SPONSORS


● JAVASCRIPT
An Introduction to Component Routing with Angular Router
Jurgen Van de Moere, 12 hours ago



<https://www.sitepoint.com/ultimate-angular-cli-reference/>



Angular Code - “Backend”

On TypeScript en ES6

Programming languages



TypeScript

ES6

ES5

ES6 and TypeScript

The future of JavaScript is ES6/ES2015

Major update from JavaScript as a programming language

Modules, classes and more

Helps in developing Angular apps

TypeScript is a typed superset of ES6:

Annotations & types

Interfaces

Compiler

TypeScript – tooling support

Types, Autocompletion, color coding.

Compile-time checking in editors.

Everything in TypeScript is
optional.

You can always use just JavaScript



Checkpoint

- Angular is a totally different beast than AngularJS
- Component-based vs. Page-based
- New Syntax
- New programming languages and design patterns
- Concepts are – mostly – the same.
- But: *you do need* a lot of boilerplate code to get started
- After that: never look around. Concentrate on components and other content