



Atos

Global Knowledge.®

# Angular Module 1 - Inleiding

Peter Kassenaar –  
[info@kassenaar.com](mailto:info@kassenaar.com)

WORLDWIDE LOCATIONS

BELGIUM CANADA COLOMBIA DENMARK EGYPT FRANCE IRELAND JAPAN KOREA MALAYSIA MEXICO NETHERLANDS NORWAY QATAR  
SAUDI ARABIA SINGAPORE SPAIN SWEDEN UNITED ARAB EMIRATES UNITED KINGDOM UNITED STATES OF AMERICA

# Peter Kassenaar

- Trainer, author, developer – since 1996
- Specialty: “*Everything JavaScript*”
- JavaScript, ES6, Angular, NodeJS, TypeScript, React, VueJS, Phonegap

[www.kassenaar.com](http://www.kassenaar.com)

[info@kassenaar.com](mailto: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](http://www.angulartraining.nl)

# github.com/PeterKassenaar/atos

The screenshot shows the GitHub repository page for `PeterKassenaar / atos`. The repository has 1 commit, 1 branch, 0 packages, 0 releases, and 1 contributor. The latest commit was made 6 minutes ago by `PeterKassenaar`. The repository contains files: `.gitignore`, `README.md`, and `README.md` (a second one). The second `README.md` file is open in a preview pane, showing the word `atos`.

Slides en voorbeeldcode bij training Angular Fundamentals + Advanced - ATOS Zaventem (B).

Manage topics

1 commit 1 branch 0 packages 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

PeterKassenaar Initial commit Latest commit 87ee157 6 minutes ago

.gitignore Initial commit 6 minutes ago

README.md Initial commit 6 minutes ago

README.md

atos

**Over jullie**



Voorkennis webdevelopment, (mobile/web-) apps?

(Kennis AngularJS 1.x?)

Voorkennis andere (web)talen?

Verwachtingen van de cursus?

Concrete projecten?

# Materialen

- Software (Angular, NodeJS, editor, downloads)
- Handouts (Github - PDF)
- Oefeningen (Github)
- Websites (online)

The image displays two side-by-side screenshots of the Angular framework. On the left is a screenshot of a personal blog post by Peter Kassens, featuring a blue header with his name and a circular profile picture, followed by text about Angular and a globe icon. On the right is the official Angular.js website, which has a dark blue header with the word 'ANGULAR' and navigation links for 'FEATURES', 'DOCS', 'ABOUT', and 'CONTRIBUTE'. The main content area features a large red 'A' logo with the text '2.0 NOW IN BETA!', followed by the slogan 'One framework. Mobile and desktop.' and a 'GET STARTED' button. Below this, there are three sections: 'Fast', 'Mobile', and 'Flexible', each with a brief description.

Peter Kassens  
Angular  
Web Development Library

ANGULAR FEATURES DOCS ABOUT CONTRIBUTE

2.0 NOW IN BETA!

One framework.  
Mobile and desktop.

GET STARTED

**Fast**  
Angular computes updates based on changes to data, not DOM, for fast updates that scale to the largest data sets with minimal memory overhead.

**Mobile**  
With Angular Universal for server-side rendering and Web Workers for smooth scrolling and transitions, Angular 2 solves the core challenges in mobile web performance.

**Flexible**  
Supports several languages including plain JavaScript, TypeScript, and Dart. Also supports both object-style data structure with POJO data-binding and functional reactive style with unidirectional data flow.

[angular.io/](http://angular.io/)

# Agenda - globaal

20-21 Januari 2020 – ma + di - “Fundamentals”

22-23 Januari 2020 – wo + do – “Advanced”

~9:00 start

~ 10:30 Break

~12:00 lunch

~ 14:30 Break

~16:15-16:30 einde

# Agenda

- Day 1 - **Intro**
  - Theorie - Introductie & geschiedenis - waarom Angular
  - Hello World in Angular – inzicht in boilerplate-code
  - Concepten, context & architecture
  - Angular CLI
  - Components
  - ECMAScript 2015 + TypeScript
  - Data binding

# Agenda

- Day 2 – Services & Communication
  - Vervolg data binding
  - Services & DI
  - HTTP, Live API's, Observables (RxJS)
  - Intercomponent communication
  - Routing
  - Forms
    - Template Driven Forms
    - Model Driven Forms

# Agenda

- Day 3 – Advanced architecture
  - Applications w/ multiple modules
  - Advanced routing & Lazy loading
  - Advanced Components
    - Content Projection
    - Smart components/View Components
  - Introduction State Management & @ngrx/store

# Agenda

- Day 4 – State management & Architecture
  - Vervolg State Management
  - Angular in the Enterprise / monorepo's (or not?)
  - Adding extra capabilities to your app
    - PWA
    - Elements
    - ...

OVERALL:

## BEST PRACTICES / STYLE GUIDE

## 2 Richtlijnen

### 1. Oefeningen

- Maar: neem ook vooral zijpaden, experimenteer, lees verder, maak een eigen project, app, website...

### 2. Voorbeeldcode

- Als ondersteuning bij de oefeningen, zie boven
- Work in progress – check de Angular-site!
- [github.com/PeterKassenaar/voorbeeldenAngular2](https://github.com/PeterKassenaar/voorbeeldenAngular2)

# Globale werkwijze



...  
...

# Vragen?



# Angular vs. The Rest

Differences, similarities, new features

# Addressing the “WHY” question!

WHY, would we want to use a frontend framework.

It is all HTML, CSS and JavaScript right?

Rethorical question:

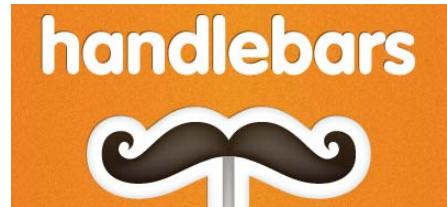
**“Do we want to go back**

***to the jQuery days?”***

speed,  
consistency,,  
community,  
performance,  
testing....

# Old school web apps

HTML + templates



Data Binding



Routing



DOM-manipulation



Mobile development

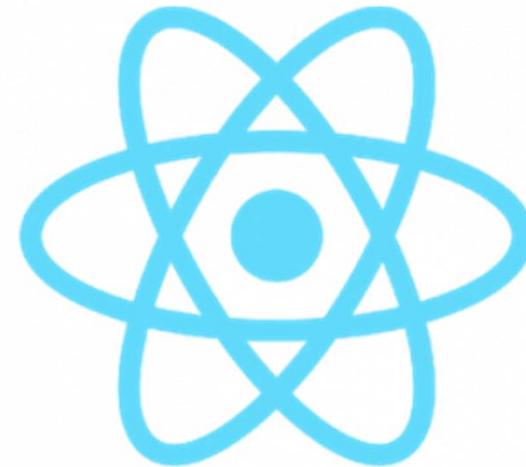


...

# “The Frankenstein Framework”



# Front-end Frameworks – the big four

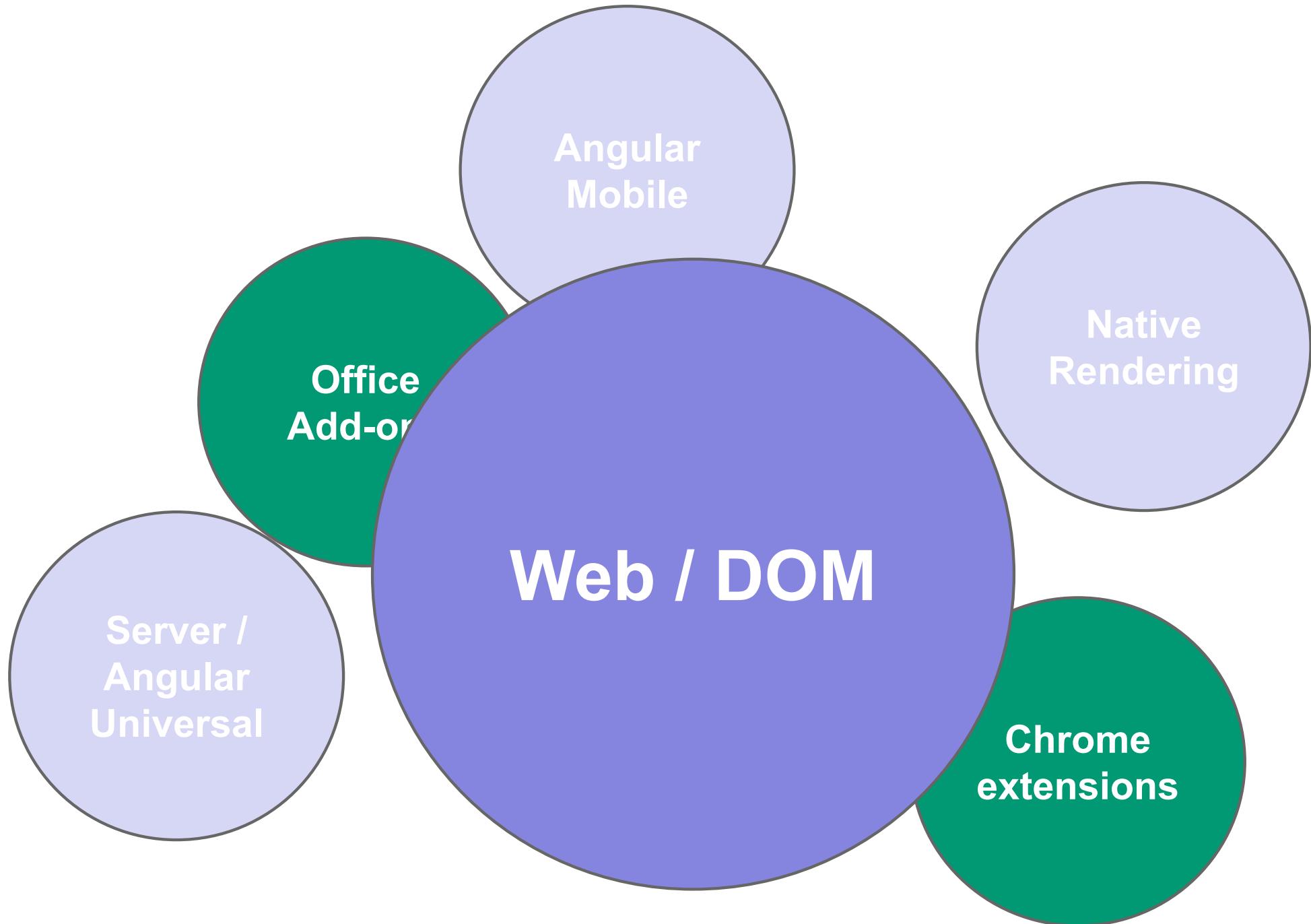




Platform

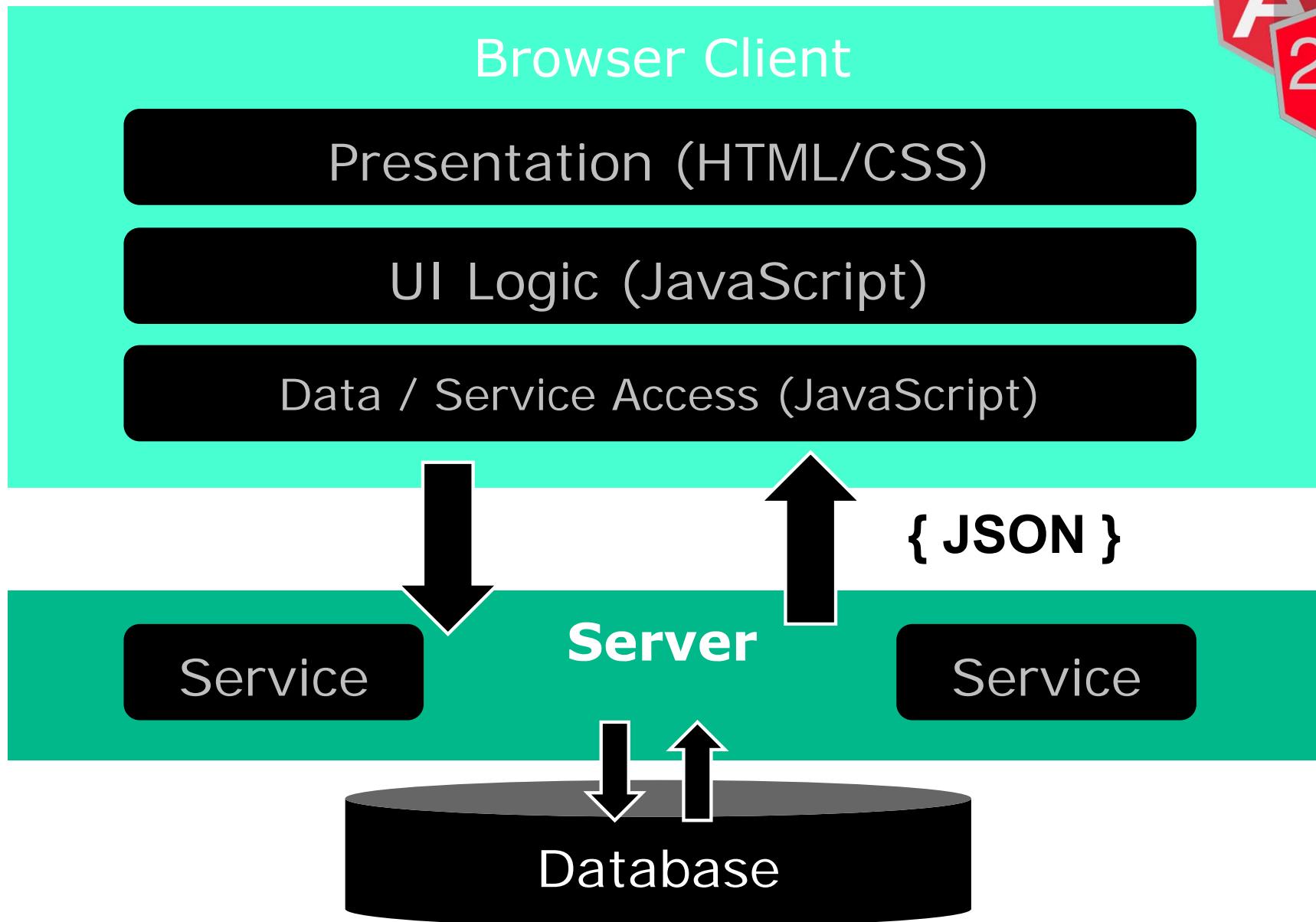
# Platform Features

	Scaffolding	Code completion & Refactoring	Debugging
Tooling	Angular CLI	Language Services	Augury
Libraries	Material 2	Mobile	Universal
	AOT- Compile	Change Detection	Renderer
Core	Components & Dependency Injection	Decorators	Zones



# Single Page Application

2010 – 20??





## Time-based Releases



Predictable & Continuous  
Evolution



- Predictability
- Painless Updates
- Long Term Support

# Angular Versies en -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 to show Internationalization, Language Service, Security, Setup & Deployment, Service Workers), Keeping Up-to-Date (expanded to show Updating Your Projects, Angular Releases, Upgrading from AngularJS), Server-side Rendering, Visual Studio 2015 QuickStart, Style Guide, Glossary, and API (stable (v6.0.4)). The main content area starts with a bulleted list about support frequency and LTS. It then shows a table of support status for versions ^4.0.0, ^5.0.0, and ^6.0.0. A sidebar on the right lists Angular versioning and releases topics, with 'Support policy' selected.

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

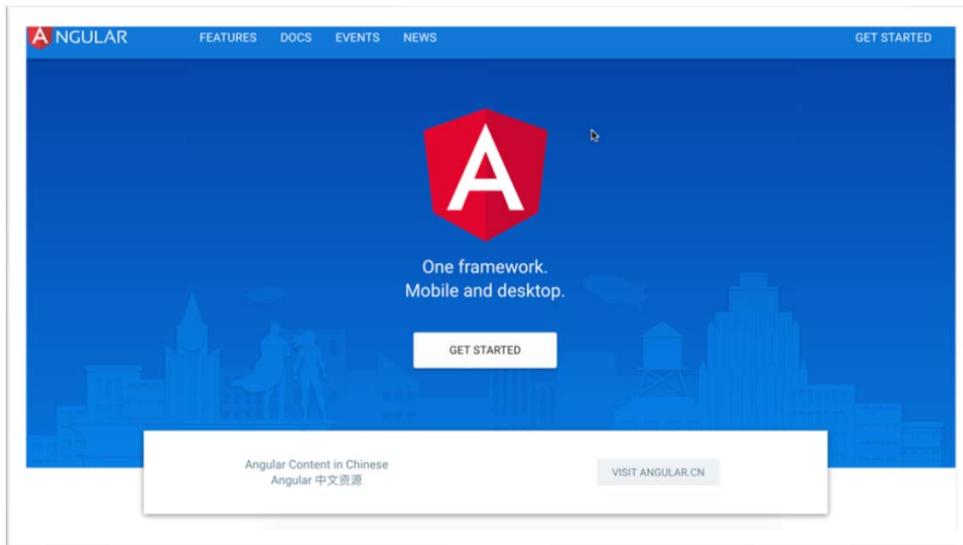
Show me how to update!

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

"It's just

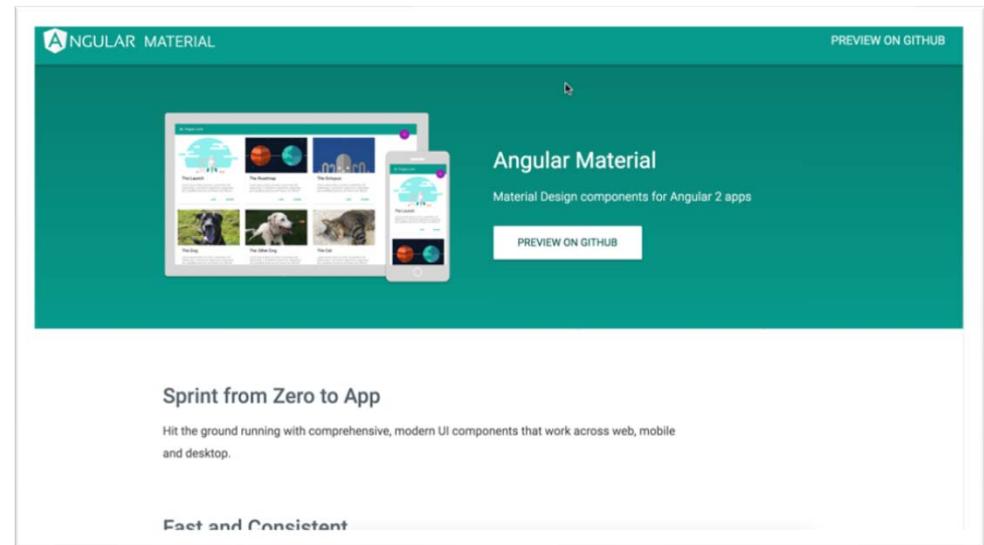
*Angular*

# Angular as a Platform



The Angular homepage features a large red Angular logo with a white 'A' inside a hexagon. Below it, the text 'One framework. Mobile and desktop.' is displayed. A 'GET STARTED' button is located at the bottom left. At the bottom right, there are links for 'Angular Content in Chinese' and 'VIST ANGULAR.CN'. The top navigation bar includes 'FEATURES', 'DOCS', 'EVENTS', 'NEWS', and 'GET STARTED'.

<https://angular.io/>



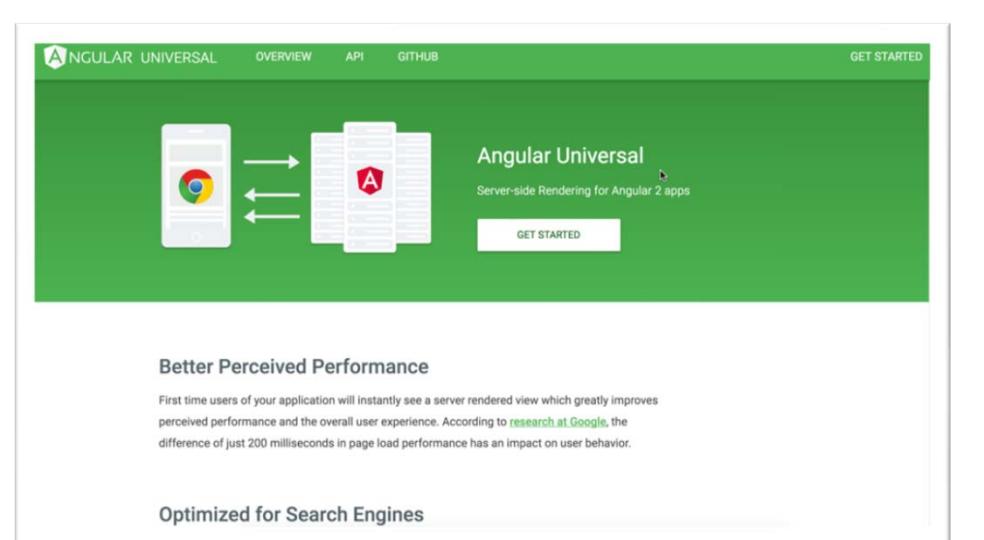
The Angular Material homepage features a green header with the 'ANGULAR MATERIAL' logo and a 'PREVIEW ON GITHUB' button. It shows a mobile phone and a laptop displaying material design components. Below this, a section titled 'Angular Material' describes it as 'Material Design components for Angular 2 apps'. A 'PREVIEW ON GITHUB' button is also present. The bottom section is titled 'Sprint from Zero to App' with a sub-section 'Fact and Conjecture'.

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



The Angular CLI homepage has an orange header with the 'ANGULAR CLI' logo, 'CLI COMMANDS', 'GITHUB', and 'GET STARTED' button. It features a terminal window showing command examples: 'npm install -g angular-cli', 'ng new my-dream-app', 'cd my-dream-app', and 'ng serve'. Below this, sections for 'ng new' and 'ng generate' are shown. The 'ng new' section explains that the Angular2 CLI makes it easy to create an application that already works, right out of the box. The 'ng generate' section shows a command example: 'ng generate component hero'.

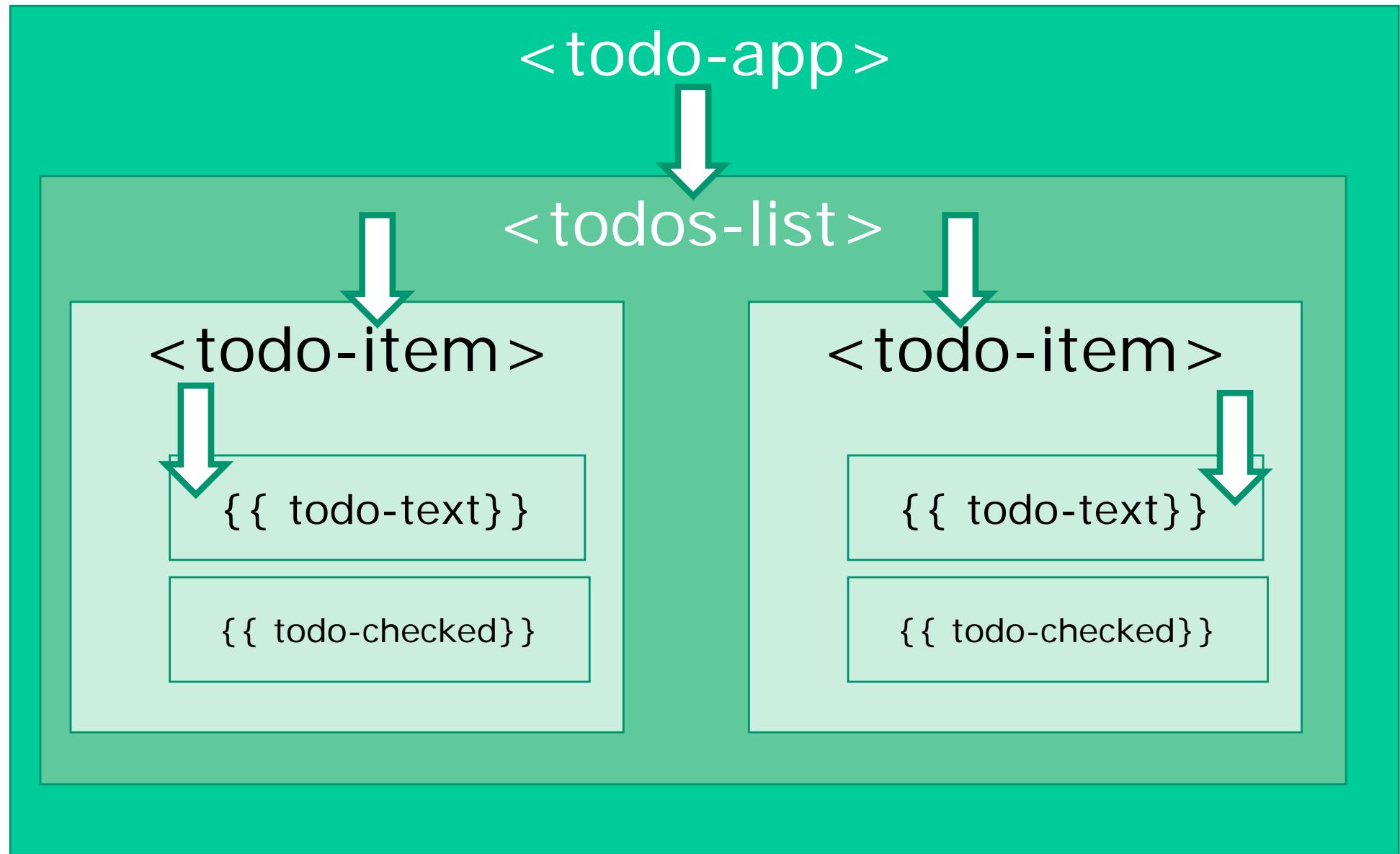
<https://cli.angular.io/>



The Angular Universal homepage has a green header with the 'ANGULAR UNIVERSAL' logo, 'OVERVIEW', 'API', 'GITHUB', and 'GET STARTED' button. It illustrates the server-side rendering process with a diagram showing a browser connected to a server stack. The 'Angular Universal' section describes it as 'Server-side Rendering for Angular 2 apps'. Below this, sections for 'Better Perceived Performance' and 'Optimized for Search Engines' are shown. The 'Better Perceived Performance' section includes a note about research at Google.

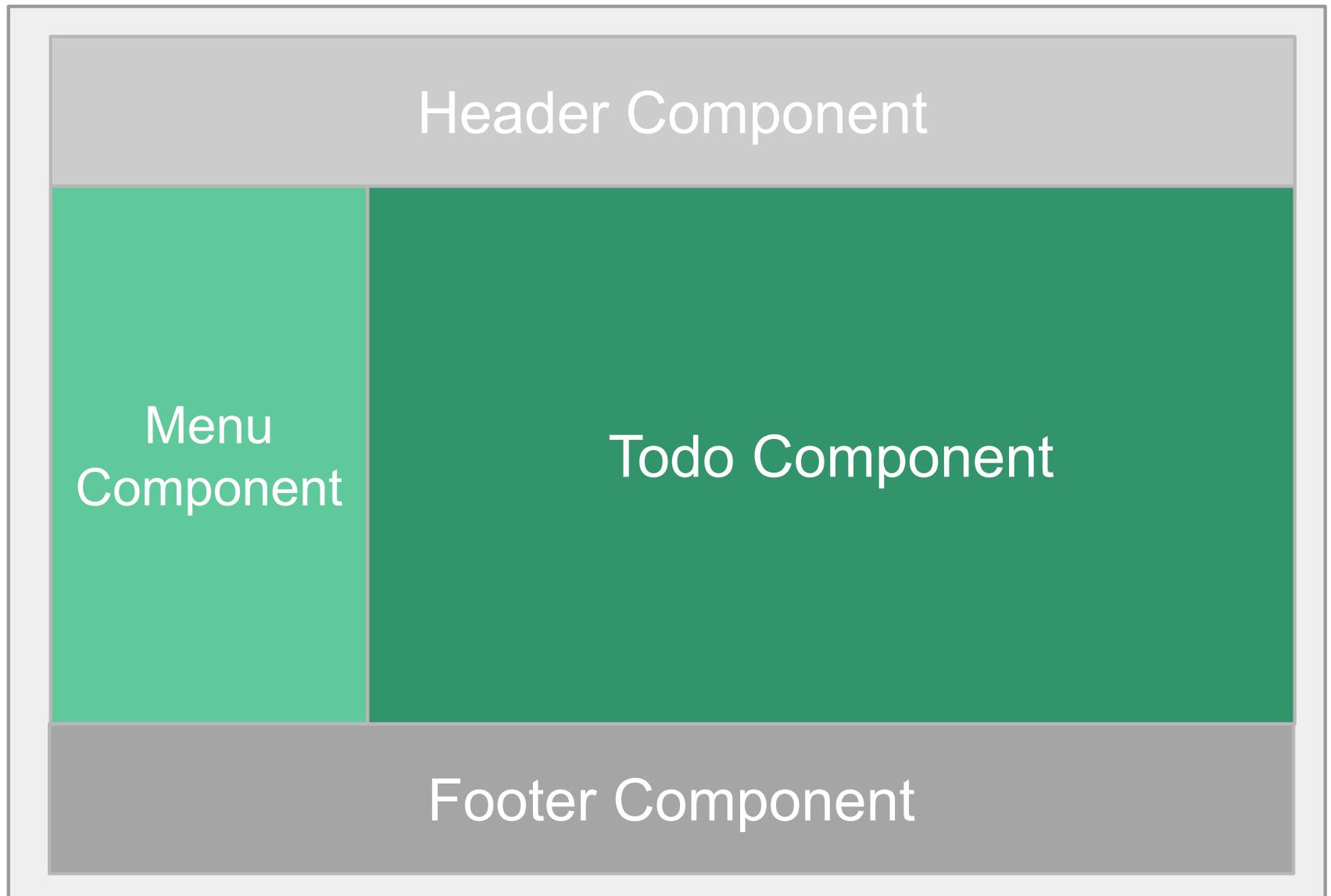
<https://universal.angular.io/>

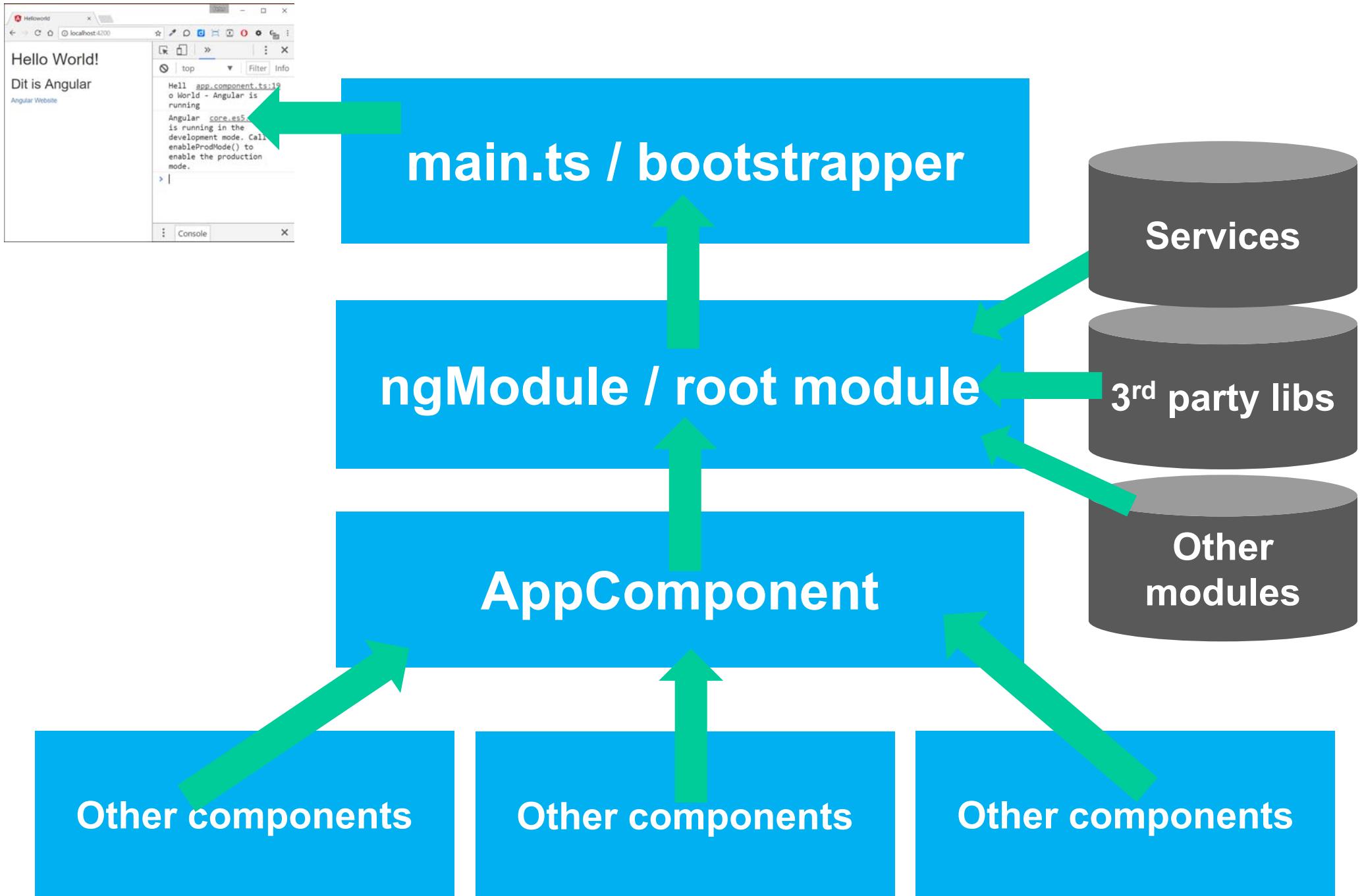
# Angular 2 - components



*“An Angular-app is a tree  
of components”*

# Components – visual representation







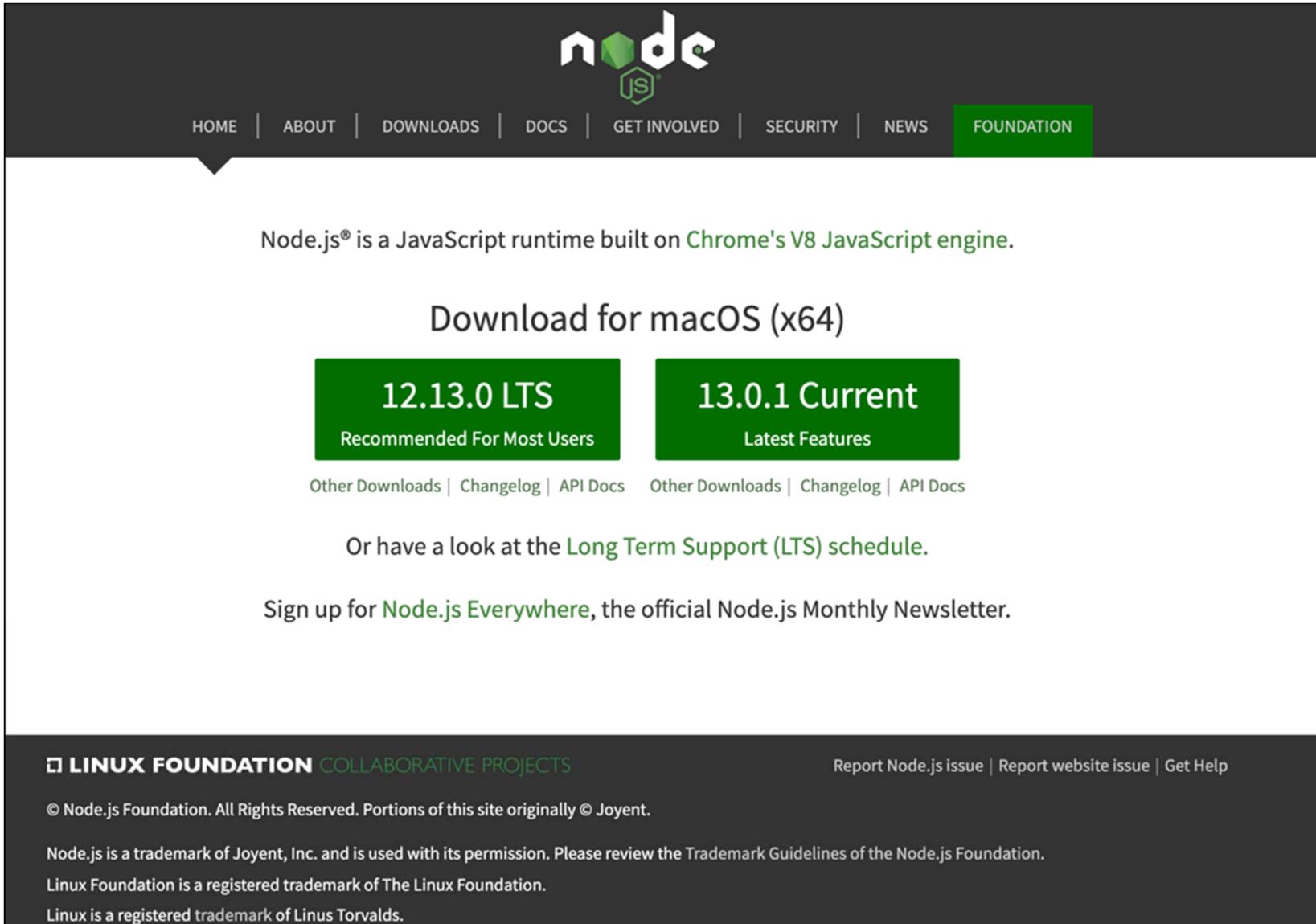
# Let's write some code

Hello World in Angular 2

# Angular 1:

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

# Angular development dependency: NodeJS 10+



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 says "Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine." It features two prominent download buttons: a green one for "12.13.0 LTS Recommended For Most Users" and a white one for "13.0.1 Current Latest Features". Below these buttons are links for "Other Downloads | Changelog | API Docs" and "Other Downloads | Changelog | API Docs". A section titled "Or have a look at the Long Term Support (LTS) schedule." is present, along with a link to "Sign up for Node.js Everywhere, the official Node.js Monthly Newsletter.". At the bottom, there's a footer with links to "Report Node.js issue | Report website issue | Get Help", "LINUX FOUNDATION COLLABORATIVE PROJECTS", and various legal and trademark information.

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Download for macOS (x64)

**12.13.0 LTS**  
Recommended For Most Users

**13.0.1 Current**  
Latest Features

Other Downloads | Changelog | API Docs    Other Downloads | Changelog | API Docs

Or have a look at the [Long Term Support \(LTS\) schedule](#).

Sign up for [Node.js Everywhere](#), the official Node.js Monthly Newsletter.

□ **LINUX FOUNDATION COLLABORATIVE PROJECTS**

© Node.js Foundation. All Rights Reserved. Portions of this site originally © Joyent.

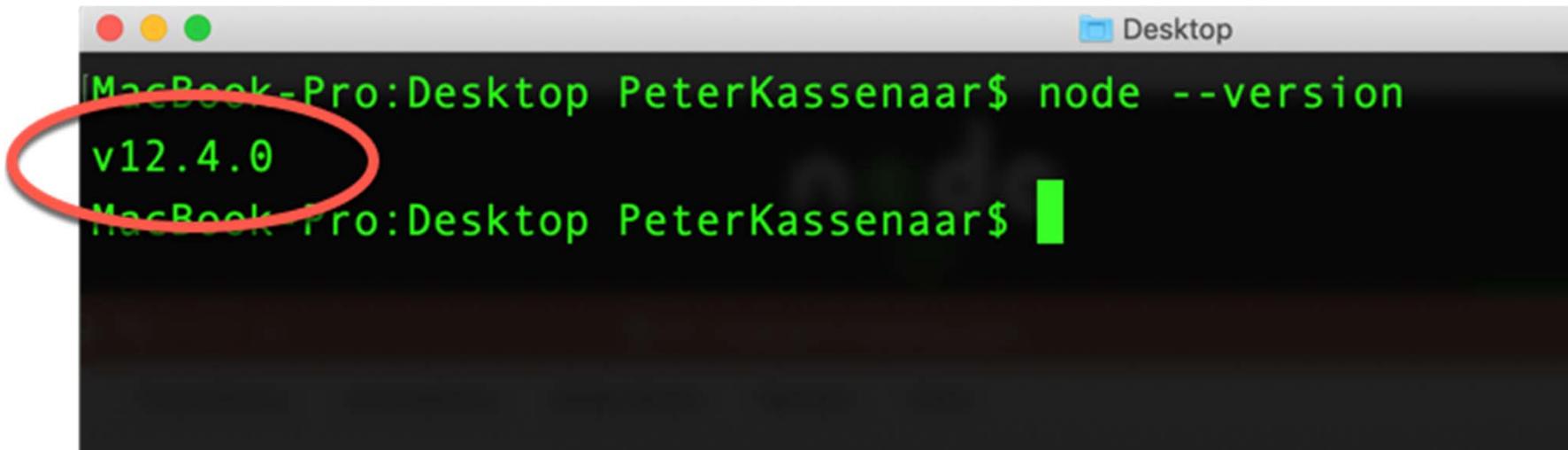
Node.js is a trademark of Joyent, Inc. and is used with its permission. Please review the Trademark Guidelines of the Node.js Foundation.

Linux Foundation is a registered trademark of The Linux Foundation.

Linux is a registered trademark of Linus Torvalds.

Report Node.js issue | Report website issue | Get Help

# Node – check your version



```
[MacBook-Pro:Desktop PeterKassenaar$ node --version
v12.4.0
MacBook-Pro:Desktop PeterKassenaar$ ]
```

# Exercise

- Download or clone  
<https://github.com/PeterKassenaar/voorbeeldenAngular2>

```
cd examples
```

```
cd 100-helloworld
```

```
npm install (or yarn install)
```

```
npm start
```

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

The screenshot shows a comparison between a running Angular application and its development environment.

**Top Left:** A browser window displaying the Angular application. The title bar says "localhost:4200". The page content includes:

- A large **Hello World!** heading.
- A sub-section titled **This is Angular**.
- A link labeled **Angular Website**.

**Bottom Right:** An IDE interface showing the project structure and the source code for `app.component.ts`.

**Project Structure:** The project is named `voorbeeldenAngular2`. It contains a `Examples` folder, which has a `100-helloworld` folder selected. This folder contains the following files and folders:  
- `node_modules` (library root)  
- `src` folder:

- `app` folder:
  - `app.component.ts`
  - `app.module.ts`
- `assets`
- `environments`
- `favicon.ico`
- `index.html`
- `main.ts`
- `polyfills.ts`
- `styles.css`
- `tsconfig.app.json`

- `.angulardoc.json`
- `.gitignore`
- `angular.json`
- `package.json`
- `package-lock.json`
- `tsconfig.json`
- `yarn.lock`

**Code Editor:** The `app.component.ts` file is open in the editor. The code is as follows:

```
import {Component, OnInit} from '@angular/core';
@Component({
  selector: 'hello-world',
  template: `
    

# Hello World!



## This is Angular


      Go to the Angular website
    
  `
})
export class AppComponent implements OnInit {
  constructor() {
  }
}
```

# Boilerplate code for Hello World

## Steps

1. Set up environment, boilerplate & libraries
2. Write Angular Root Component
3. Write @NgModule Component
4. Bootstrap component
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": "6.0.0",  
    "@angular/common": "6.0.0",  
    "@angular/compiler": "6.0.0",  
    "@angular/core": "6.0.0",  
    "@angular/forms": "6.0.0",  
    "rxjs": "^6.1.0",  
    "zone.js": "^0.8.26"  
  },  
  "devDependencies": {  
    "@angular-devkit/build-angular": "~0.6.0",  
    "@angular/cli": "6.0.0",  
    "typescript": "2.7.2"  
  },  
  "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 stars and 2,567 forks. The 'Code' tab is selected, showing the file 'non-essential-files.txt'. The file was last updated by 'wardbell' on branch 'master' with commit 'df3'. It contains 16 lines (15 sloc) and 226 Bytes. The file lists the following 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 discusses the @NgModule decorator and its purpose in organizing applications. The sidebar features a bio of John Papa and links to his most recent posts.

**Header:** Angular Modules

**Post Title:** Introducing Angular Modules - Root Module

**Author:** John

**bio:** 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 →

**Recent Posts:**

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

**Text in Post:** 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."

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

# Step 4 - bootstrap component

Best practice: bootstrap app in separate component

Convention: main.ts, or 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

Header:

```
<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 van index.html

Verwijzing naar de root-component:

```
<body>  
  <hello-world>  
    Bezig met laden...  
  </hello-world>  
</body>
```

# App draaien

npm start – draait de scriptopdracht start uit package.json.

ng serve - start globale angular-cli instantie

The terminal window shows the following output:

```
+ 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} [initial] [rendered]

webpack: C
```

The browser window displays the Angular application with the following content:

Hello World!  
Dit is Angular  
Angular Website

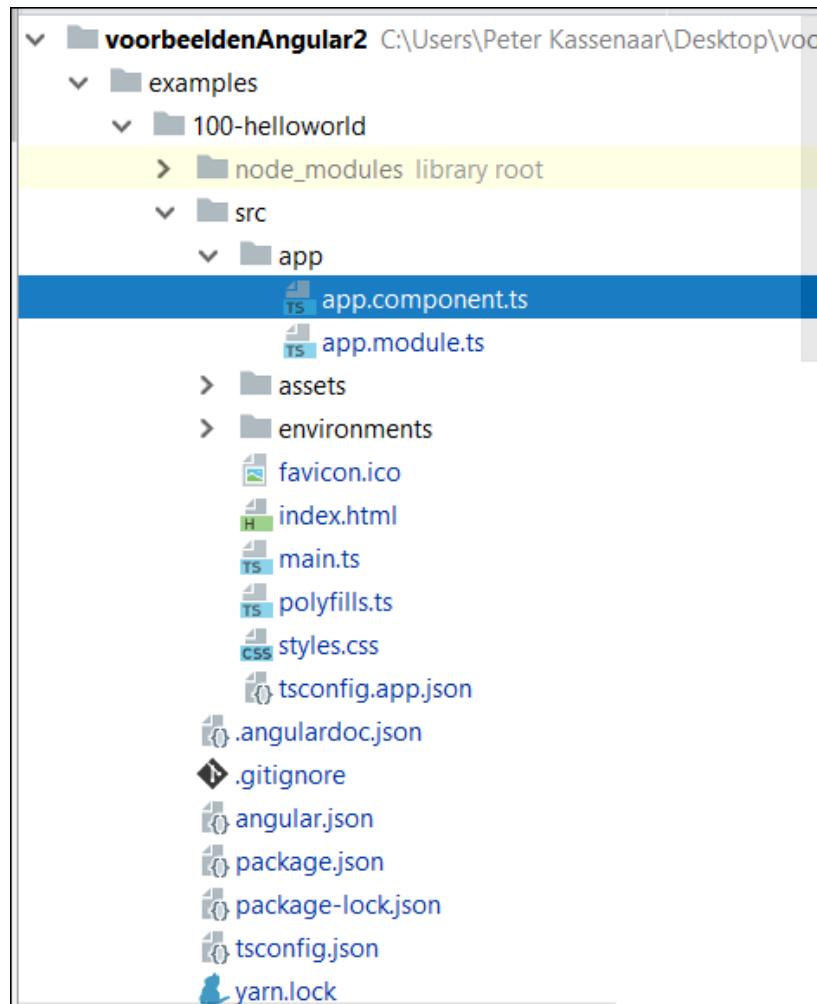
The developer tools' Console tab shows the following logs:

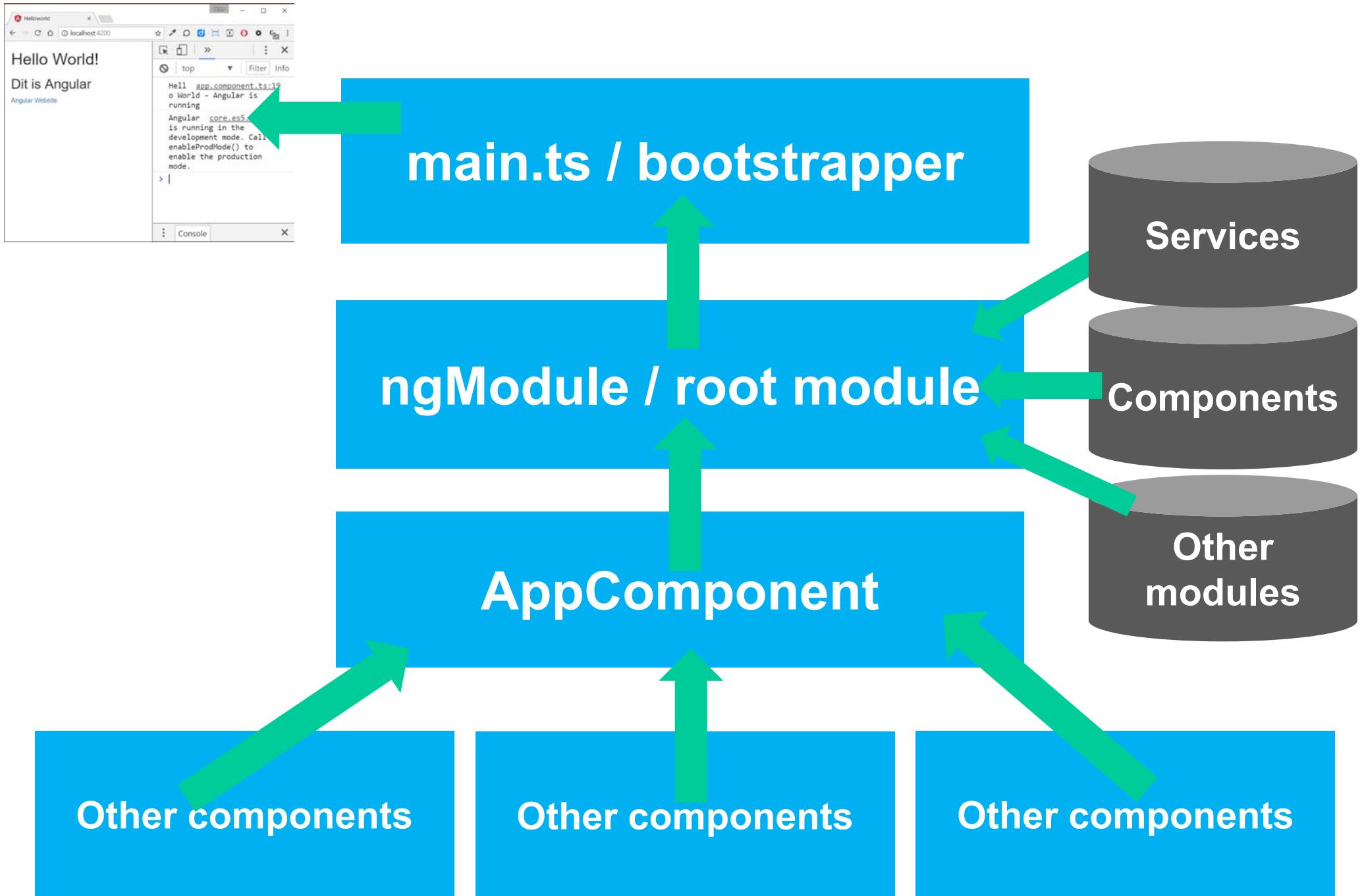
```
Hello World - Angular is running
Angular is running in the development mode. Call
enableProdMode() to enable the production mode.
app.component.ts:19
core.es5.js:3025
```

Daarna: wijzigingen aanbrengen in app.component.ts

– worden opgepikt door Live Reload

# Basic Project Structure





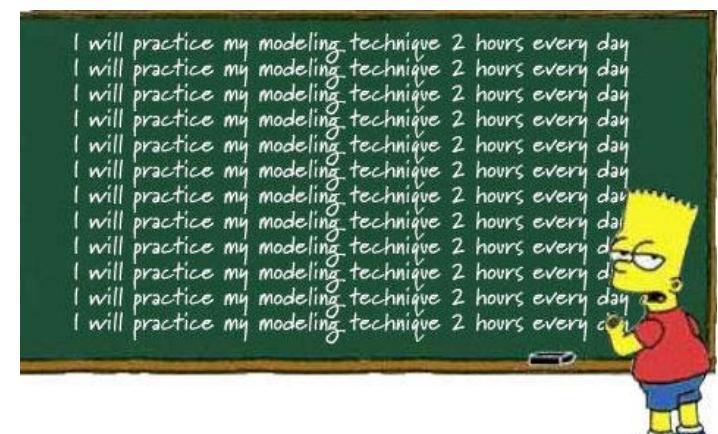
Screenshot of a browser console showing the output of the application:

```
HelloWorld -> app.component.ts:19 o World - Angular is running
Angular core.es5 is running in the development mode. Call enableProdMode() to enable the production mode.
```

# Checkpoint

- Er is aardig wat boilerplate code nodig om een Angular-app te starten
- Vier stappen
  1. Set up environment, boilerplate & libraries
  2. Schrijf Angular Root Component voor de app
  3. Bootstrap de component
  4. Schrijf HTML-pagina (`index.html`)
- Daarna: app gaan uitbreiden
- Oefening 1a), 1b), 1c), 1d)

# Oefening.....



# Assets

[github.com/PeterKassenaar/voorbeeldenAngular2](https://github.com/PeterKassenaar/voorbeeldenAngular2)

Oefeningen en meer voorbeeldcode



# Angular CLI

Snel nieuwe projecten instellen via de command line

# Angular-CLI to the rescue

- Het *is* mogelijk nieuwe Angular-projecten from scratch te starten.
- Met de CLI is eenvoudiger.
- CLI-options:
  - Scaffolding
  - Generating
  - Testing
  - Building
  - AOT-Compiling
  - ...

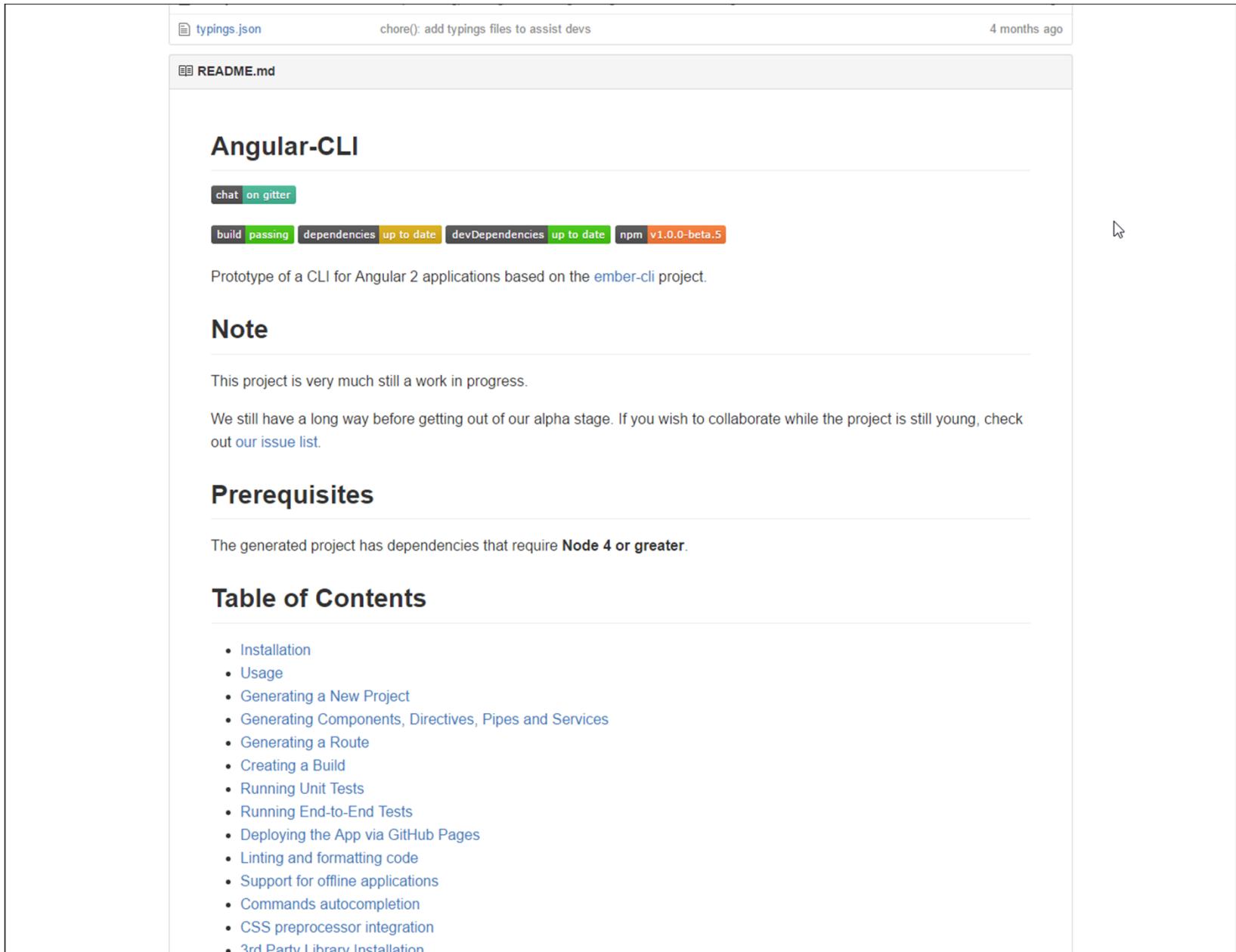
# Scaffolding - Angular CLI

Projecten, componenten, routes en meer definiëren  
vanaf de command line

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

en

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



A screenshot of the Angular-CLI project's GitHub README page. The page shows a commit history at the top, followed by the README content. The README includes sections for 'Note' and 'Prerequisites', and a 'Table of Contents' with a list of 19 items.

File: typings.json  
Commit message: chore(): add typings files to assist devs  
Time: 4 months ago

## 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](#).

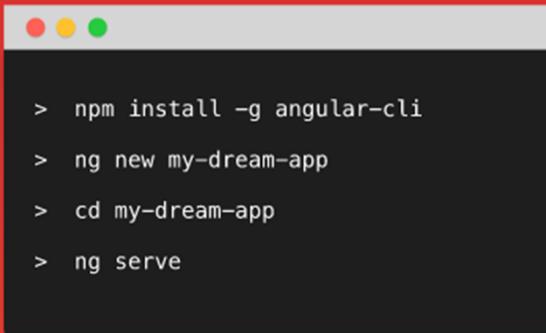
### Prerequisites

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 autocompletion
- CSS preprocessor integration
- 3rd Party Library Installation

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

A screenshot of a Mac OS X terminal window. The title bar shows the standard red, yellow, and green window controls. The main area of the terminal contains the following text:

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

## Angular CLI

A command line interface for Angular

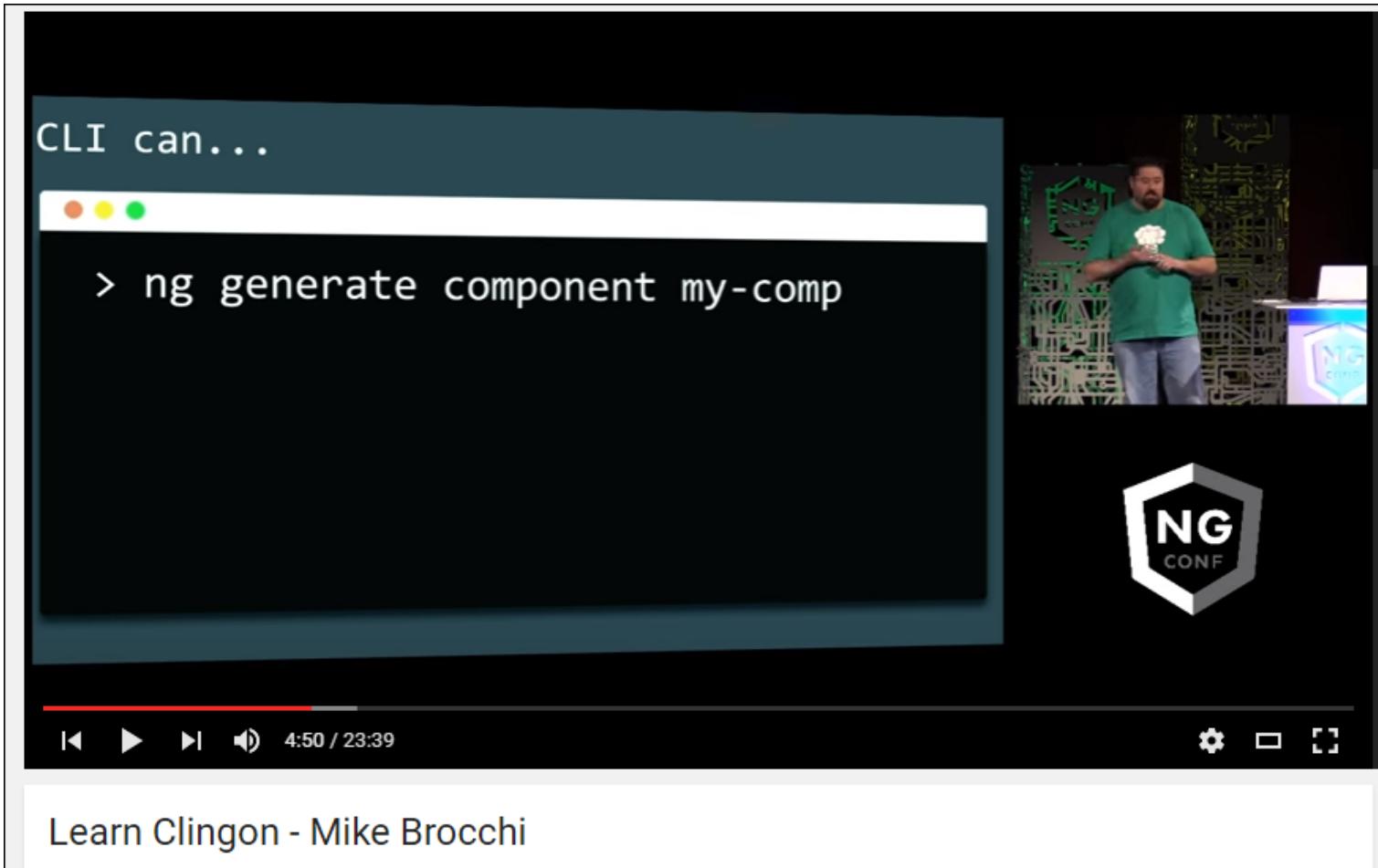
[GET STARTED](#)

## ng new

The Angular2 CLI makes it easy to create an application that already works, right out of the box. It already follows our best practices!

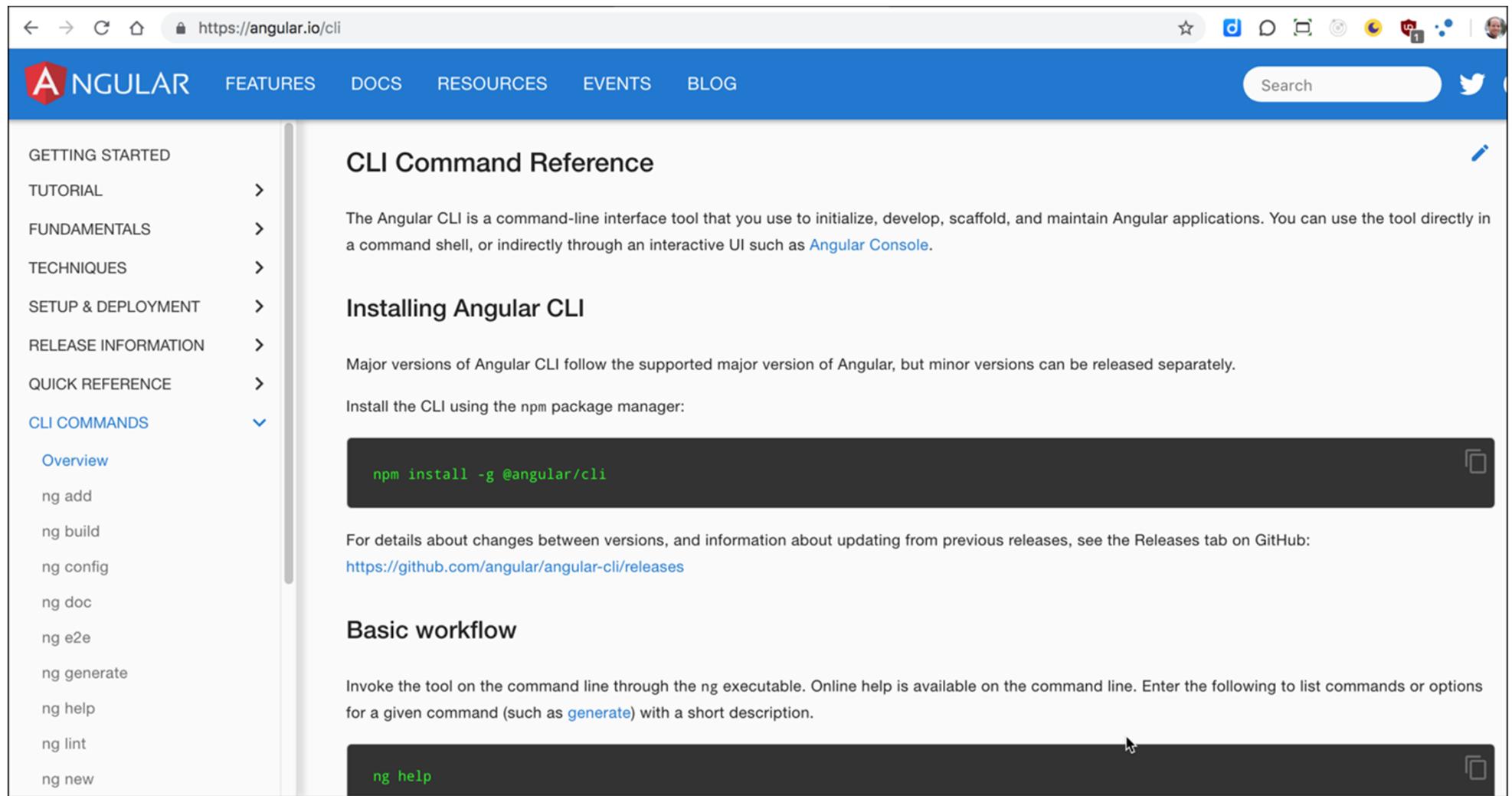
## ng generate

Generate components, routes, services and pipes with a simple command. The CLI will also create simple test shells for all of these.



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

# Documentatie - in de Angular Docs



The screenshot shows a browser window displaying the Angular CLI Command Reference page at <https://angular.io/cli>. The page has a blue header with the Angular logo and navigation links for FEATURES, DOCS, RESOURCES, EVENTS, and BLOG. A search bar and social sharing icons are also in the header. The main content area has a sidebar on the left with sections like GETTING STARTED, TUTORIAL, FUNDAMENTALS, TECHNIQUES, SETUP & DEPLOYMENT, RELEASE INFORMATION, QUICK REFERENCE, and CLI COMMANDS. The CLI COMMANDS section is expanded, showing sub-commands: Overview, ng add, ng build, ng config, ng doc, ng e2e, ng generate, ng help, ng lint, and ng new. The main content on the right is titled "CLI Command Reference". It describes the Angular CLI as a command-line interface tool for initializing, developing, scaffolding, and maintaining Angular applications. It includes sections on "Installing Angular CLI", "Basic workflow", and command examples like "npm install -g @angular/cli" and "ng help". There are also links to GitHub releases and a "RELEASES" tab.

Angular

FEATURES DOCS RESOURCES EVENTS BLOG

Search

GETTING STARTED

TUTORIAL >

FUNDAMENTALS >

TECHNIQUES >

SETUP & DEPLOYMENT >

RELEASE INFORMATION >

QUICK REFERENCE >

CLI COMMANDS >

Overview

ng add

ng build

ng config

ng doc

ng e2e

ng generate

ng help

ng lint

ng new

## CLI Command Reference

The Angular CLI is a command-line interface tool that you use to initialize, develop, scaffold, and maintain Angular applications. You can use the tool directly in a command shell, or indirectly through an interactive UI such as [Angular Console](#).

### Installing Angular CLI

Major versions of Angular CLI follow the supported major version of Angular, but minor versions can be released separately.

Install the CLI using the npm package manager:

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

For details about changes between versions, and information about updating from previous releases, see the Releases tab on GitHub:  
<https://github.com/angular/angular-cli/releases>

### Basic workflow

Invoke the tool on the command line through the ng executable. Online help is available on the command line. Enter the following to list commands or options for a given command (such as `generate`) with a short description.

```
ng help
```

<https://angular.io/cli>



# Angular 2 Code - Backend

Kort over TypeScript en ES6

# Programmeertalen



TypeScript

ES6

ES5

# **ES6 en TypeScript**

**De toekomst van JavaScript is ES6/ES2015**

Major update van JavaScript als programmeertaal

Modules, classes en meer

Helpt bij het ontwikkelen in Angular 2

**TypeScript breidt ES6 verder uit**

Annotaties & types

Interfaces

Compiler

## TypeScript – tooling support

Types, Autocompletion.

Compile-time checking in editors.

Alles is *optioneel*. Je kunt altijd nog gewoon JavaScript gebruiken.

# Onderdelen van een Component Class

imports

```
import { Component } from '@angular/core';
import { DataService } from './services/data-service';
```

annotations

```
@Component({
  selector: 'orders',
  directives: [DataService],
  templateUrl: 'orders-component.html',
})
```

class

```
export class OrdersComponent {
  ...
}
```

# Checkpoint

- Angular 2 is een totaal ander framework dan Angular 1
- Component-based vs. Page-based
- Nieuwe syntaxis
- Nieuwe programmeertalen en andere nieuwe kenmerken
- Concepten komen – grotendeels – overeen
- Veel boilerplate-code nodig voor een Quickstart
- Daarna: niet meer naar omkijken. Concentreren op de componenten