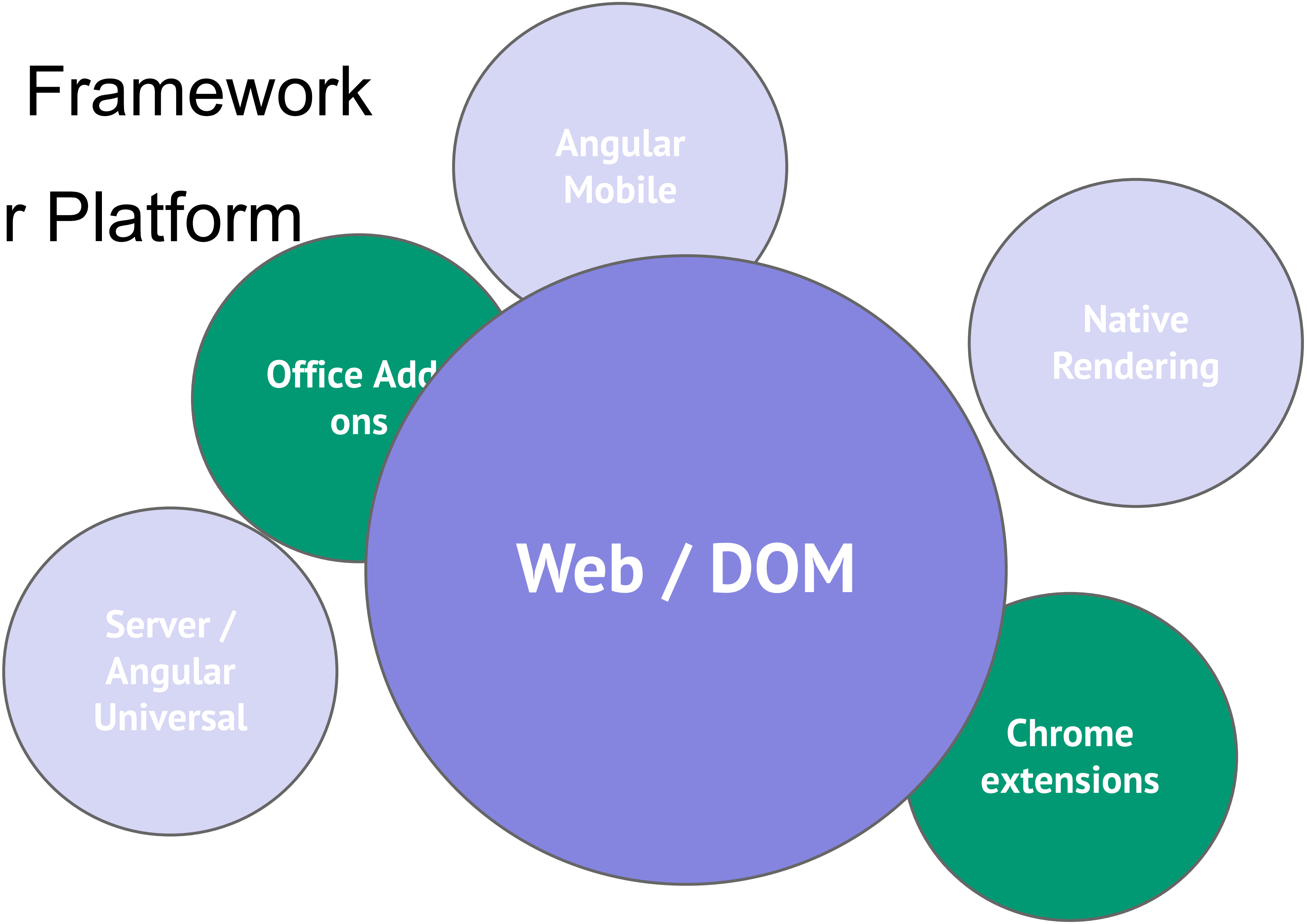
Angular Fundamentals
Korte herhaling dag 1

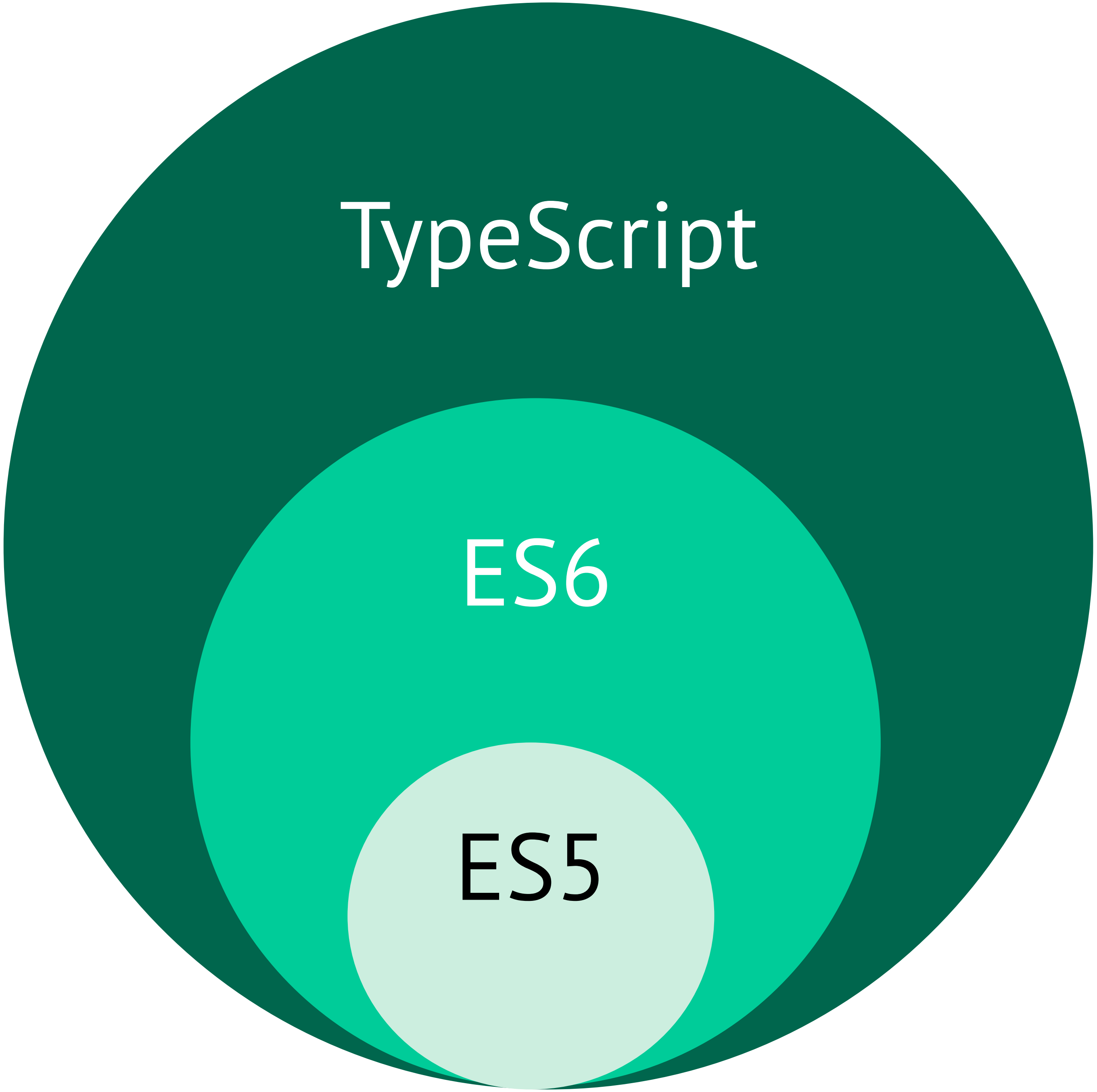
Peter Kassenaar –
info@kassenaar.com

Framework to Platform

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

Van Framework naar Platform



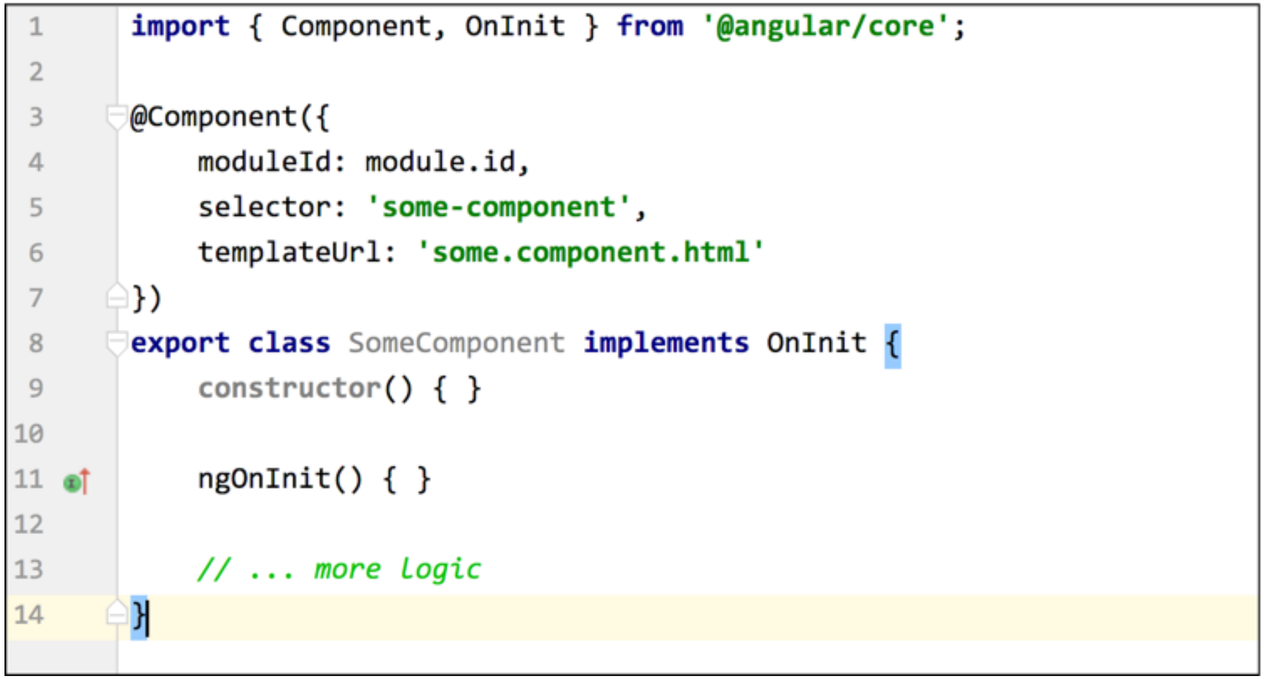

TypeScript
ES6
ES5

Boilerplate code - stappenplan

1. Set up environment, boilerplate & libraries
   - `npm install`
2. Schrijf Angular Root Component & Module voor de app
3. Bootstrap
4. Schrijf HTML-pagina (`index.html`)
5. Run applicatie
   - `npm start`

Components

- `@Component` decorator
- View
- Controller / Class



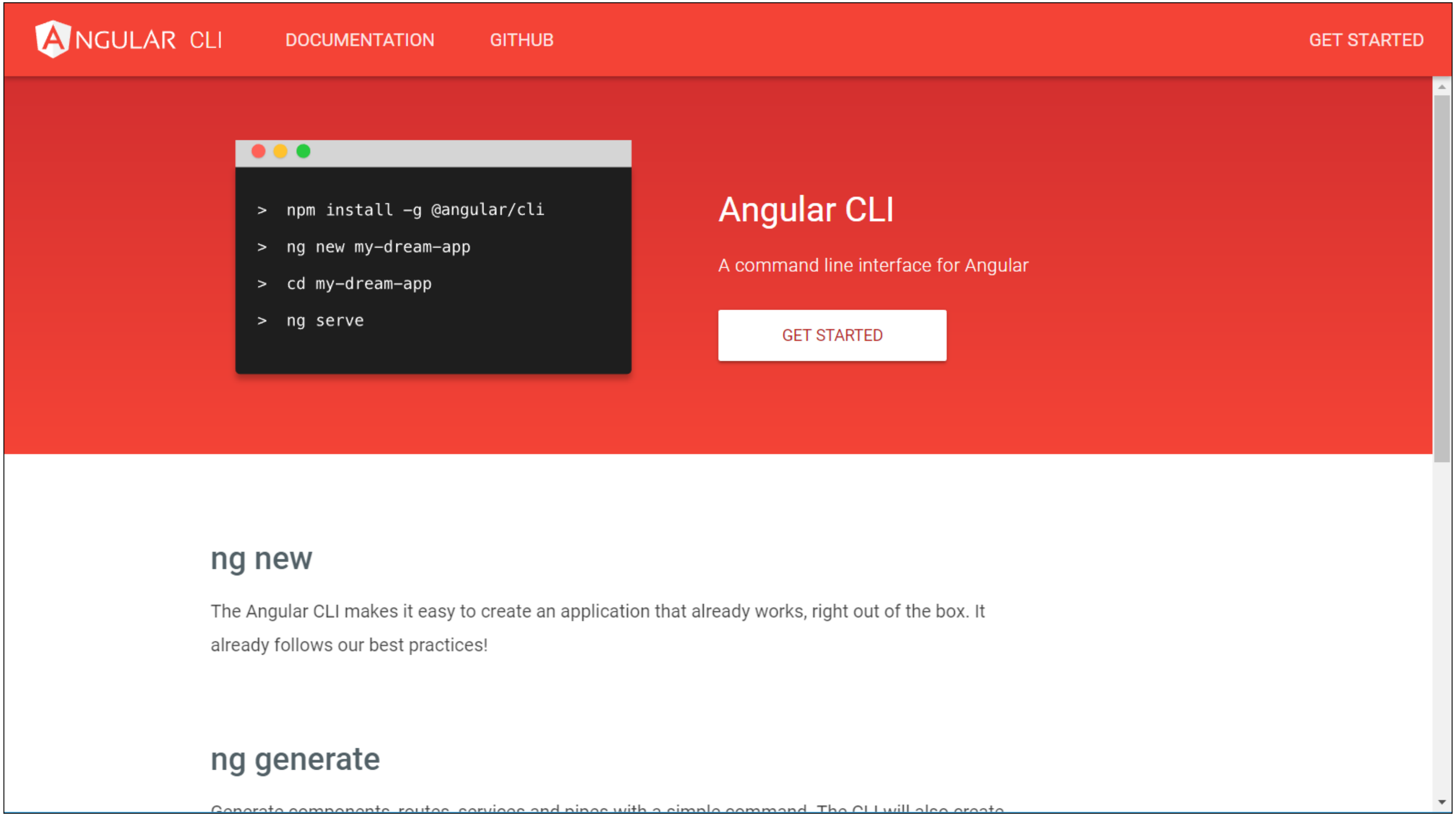
```
import { Component, OnInit } from '@angular/core';

@Component({
    moduleId: module.id,
    selector: 'some-component',
    templateUrl: 'some.component.html'
})
export class SomeComponent implements OnInit {
    constructor() { }

    ngOnInit() { }

    // ... more logic
}
```

Angular CLI



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

Angular CLI

A command line interface for Angular

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 serve

- Vervult verschillende rollen (*kan* ook handmatig)
    1. TypeScript compilation
    2. (SASS compilation)
    3. Fire up webserver on http://localhost:4200
    4. Fire up Live Reload
    5. Analyzing and bundling of application by firing up WebPack
    6. Watch for changes and recompile, rebundle, etc.

Data binding

- Nieuwe notatiewijzen in HTML-views/partials.

    1. Simple data binding **{{ … }}**
    2. Event binding **(…)="..."**
    3. One-way data binding **[…]="..."**
    4. Two-way data binding **[(ngModel)] = " …"**

Event binding

- Niet-DOM events binden: `@HostListener()`
- Event van View → Controller?
  - `$event` doorgeven als parameter
- Local Template Variable, `#`
- `@ViewChild(‘varName’)` voor doorgeven DOM-element aan controller/class

Vandaag

- Afronden Attribute binding
- two-way Data Binding
- Services
  - Static Services
  - RxJS / Observables, Live API’s
- Tree of components
  - Applicaties opbouwen met meerdere componenten
  - Data flow tussen de componenten
  - `@Input()` en `@Output()`
- Case

Dag 3

- Bespreking Case
- Multiple components
  - Event bus / Service bus
- POST-ing data / Live CRUD-operations
- Routing
- Forms
- Examen