

Angular

Vor- und Nachteile

Vorteile

- MVC Architecture Implementation
- Erweiterte Design Architecture
- Modules
- Services und Dependency Injection
- Custom Directives
- TypeScript

Nachteile

- Begrenzte SEO Optionen
- Komplex
- Steile Lernkurve
- CLI-Dokumentation fehlt Details
- Minified ~180 KB
- subjektive Meinung Jans: verdammt hässlich
- Keine Fehlermeldungen

Aufsetzen

Um Angular zu verwenden muss das Angular-Framework installiert werden. Dafür muss man zuerst in den Ordner des Projekts wechseln und dort den Befehl `npm -g @angular/cli` eingeben:

```
C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit>mkdir Hello-World
C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit>cd Hello-World
C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit\Hello-World>mkdir server
C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit\Hello-World>npm install -g @angular/cli
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
C:\Users\danie\AppData\Roaming\npm\ng -> C:\Users\danie\AppData\Roaming\npm\node_modules\@angular\cli\bin\ng
> @angular/cli@12.0.1 postinstall C:\Users\danie\AppData\Roaming\npm\node_modules\@angular\cli
> node ./bin/postinstall/script.js
? Would you like to share anonymous usage data with the Angular Team at Google under
Google's Privacy Policy at https://policies.google.com/privacy? For more details and
how to change this setting, see https://angular.io/analytics. No
+ @angular/cli@12.0.1
added 234 packages from 180 contributors in 58.062s
C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit\Hello-World>
```

Jetzt wird die ExecutionPolicy auf Unrestricted gesetzt:

```
PS C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit\GruppeEAngular\hello_world> Set-ExecutionPolicy

Cmdlet Set-ExecutionPolicy an der Befehlspipelineposition 1
Geben Sie Werte für die folgenden Parameter an:
ExecutionPolicy: Unrestricted

Ausführungsrichtlinie ändern
Die Ausführungsrichtlinie trägt zum Schutz vor nicht vertrauenswürdigen Skripts bei. Wenn Sie die Ausführungsrichtlinie
ändern, sind Sie möglicherweise den im Hilfethema "about_Execution_Policies" unter
"https://go.microsoft.com/fwlink/?LinkID=135170" beschriebenen Sicherheitsrisiken ausgesetzt. Möchten Sie die
Ausführungsrichtlinie ändern?
[J] Ja [A] Ja, alle [N] Nein [K] Nein, keine [H] Anhalten [?] Hilfe (Standard ist "N"): a
```

Als nächstes wird ein neuer Workspace angelegt. Angular routing ermöglicht mehrere unterschiedliche Seiten ohne Laden anzuzeigen.

```
PS C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit\GruppeEAngular\hello_world> ng new testSomething
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? CSS
CREATE testSomething/angular.json (3081 bytes)
CREATE testSomething/package.json (1076 bytes)
CREATE testSomething/README.md (1059 bytes)
CREATE testSomething/tsconfig.json (783 bytes)
CREATE testSomething/.editorconfig (274 bytes)
CREATE testSomething/.gitignore (604 bytes)
CREATE testSomething/.browserslistrc (703 bytes)
CREATE testSomething/karma.conf.js (1430 bytes)
CREATE testSomething/tsconfig.app.json (287 bytes)
CREATE testSomething/tsconfig.spec.json (333 bytes)
CREATE testSomething/src/favicon.ico (948 bytes)
CREATE testSomething/src/index.html (299 bytes)
CREATE testSomething/src/main.ts (372 bytes)
CREATE testSomething/src/polyfills.ts (2820 bytes)
CREATE testSomething/src/styles.css (80 bytes)
CREATE testSomething/src/test.ts (743 bytes)
CREATE testSomething/src/assets/.gitkeep (0 bytes)
CREATE testSomething/src/environments/environment.prod.ts (51 bytes)
CREATE testSomething/src/environments/environment.ts (658 bytes)
CREATE testSomething/src/app/app-routing.module.ts (245 bytes)
CREATE testSomething/src/app/app.module.ts (393 bytes)
CREATE testSomething/src/app/app.component.html (23809 bytes)
CREATE testSomething/src/app/app.component.spec.ts (1078 bytes)
CREATE testSomething/src/app/app.component.ts (217 bytes)
CREATE testSomething/src/app/app.component.css (0 bytes)
? Packages installed successfully.
Directory is already under version control. Skipping initialization of git.
```

Jetzt kann innerhalb des Workspace die Seite geöffnet werden:

```
PS C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit\GruppeEAngular\hello_world> cd testSomething
PS C:\Users\danie\OneDrive\Desktop\Schule\4.Jg\DBI\Gruppenarbeit\GruppeEAngular\hello_world\testSomething> ng serve --open
- Generating browser application bundles...Compiling @angular/core : es2015 as esm2015
Compiling @angular/common : es2015 as esm2015
Compiling @angular/platform-browser : es2015 as esm2015
Compiling @angular/router : es2015 as esm2015
Compiling @angular/platform-browser-dynamic : es2015 as esm2015
? Browser application bundle generation complete.
```

REST-Call-Funktionalität

Um mit Angular Rest-Calls machen zu können, muss ein Modul in src/app/app.module.ts hinzugefügt werden.

```
import { HttpClientModule } from '@angular/common/http';
```

In dem Imports-Array muss HttpClientModule angegeben werden.

Nun muss in der src/app/api.service.ts der HttpClient dazugegeben werden.

```
import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';

import {Observable} from 'rxjs/Observable';

const httpOptions = {
  headers: new HttpHeaders({ 'Content-Type': 'application/json' })
```

```
};
export class ApiService {

  constructor(private httpClient:HttpClient) { }

}
```

Mithilfe des Observable-imports können die späteren REST-Methoden Observable-Objekte zurück geben.

Nun können wir Methoden anlegen, hier ein paar standard:

```
getTest() {
  return this.http.get('/api/test');
}

createTest(test) {
  let body = JSON.stringify(test);
  return this.http.post('/api/test/', body, httpOptions);
}

updateTest(test) {
  let body = JSON.stringify(test);
  return this.http.put('/api/test/' + test.id, body,
    httpOptions);
}

deleteTest(test) {
  return this.http.delete('/api/test/' + test.id);
}
```

Um die Daten auch wirklich zu bekommen müssen die Komponenten bei Observable registrieren:

```
ngOnInit() {
  this.getTest();
}

getTest() {
  this._apiService.getTest().subscribe(
    data => { this.test = test },
    err => console.error(err),
    () => console.log('done loading testdata')
  );
}
}
```

Besonderheiten des Frameworks

Angular präsentiert nicht nur die Tools, sondern auch das Design-Pattern um Projekte in einem verwaltbaren Weg.

Angular ist mit TypeScript aufgebaut, das wiederum auf JS ES6 aufbaut. Deswegen braucht man keine neue Programmiersprache lernen.

Mit Angular hat man bereits eine Menge Werkzeuge, um sofort mit der Gestaltung der Anwendung beginnen zu können.

Angular ist dafür gedacht, gründlich getestet zu werden und unterstützt sowohl Unit- als auch End-to-End-Tests mit Tools wie Jasmine und Protractor.

Angular ist Mobile- und Desktop-ready, d.h. man hat ein Framework für mehrere Plattformen.

Es gibt eine Zahnpflegemarke mit einem Produkt namens Angular.

Probleme

Microsoft

Es gibt ein altes und ein neues Angular. Viele Tutorials, Dokumentationen und Anleitungen sind für AngularJS(alt).

Extrem kompliziert

Fehlermeldungen sind von fragwürdiger Qualität

Cooler Features

Features von Angular

- Support von mehreren Plattformen und TypeScript 3.4
- einfaches Erstellen von Libraries
- Lazy Loading
- Ivy Preview

Geschichte des Frameworks

Ursprünglich hieß Angular als Neuimplementierung von AngularJS 'Angular 2', da beide Versionen jedoch stark unterschiedlich sind, kam es zu Verwirrung unter den Entwicklern. Daraufhin wurden beide Versionen deutlich abgegrenzt und 'AngularJS' und 'Angular' genannt.

Die derzeit aktuelle Version ist Angular 12.

Angular 2 wurde auf der ng-Europe-Konferenz vom 22-23. Oktober 2014 angekündigt. Am 23. März 2017 wurde schon Angular 4 veröffentlicht, um Verwirrung wegen eines Pakets mit der Versionsnummer 3 zu vermeiden. Version 5 erschien am 1. November 2017, womit progressive Web-Apps und ein Build Optimizer unterstützt werden sowie Verbesserungen im Material Design (Dateisprache von Google). Die Integration von Angular Elements im

Framework und die Einführung der Command-Line-Befehle 'ng update' und 'ng add' wurden mit Version 6 vom 3. Mai 2018 hinzugefügt. Fokus auf Performance-Verbesserung, Neuerung bei Angular Material und CDS, Virtual Scrolling und das Verwenden von Typescript 3.1 und RxJS 6.3 kamen mit Angular 7 (18. Oktober 2018) dazu. Am 28. Mai 2019 erschien Angular 8, hier lag der Fokus auf Differential-Loading, inkrementelles Kompilieren und Optimieren durch Ivy, Multithreading, Anpassungen beim Lazy Loading (Datenobjekte stellen Werte oder andere Objekte bereit), mehr Angular Befehle und Typescript 3.4, Node 12 und RxJS 6.5.1 werden verwendet. Am 6. Februar 2020 erschien Angular 9, am 25. Juni 2020 Angular 10, am 11 November Angular 11 und am 13. Mai 2021 Angular 12.

Berühmte Applikationen mit Angular

- Gmail
- Nike
- HBO
- Sony
- Forbes
- Upwork (Engagement von Freelancern)
- PayPal
- Weather.com
- Wikiwand (Alternative Darstellung für Wikipedia-Artikel)

Tooling hinter Framework

Welche IDEs gibt es?

Am schnellsten und effektivsten für die Entwicklung ist die "Angular IDE". Sie ist sehr einfach zu verstehen.

Auch Webstorm von IntelliJ und Visual Studio Code sind effektiv. Des Weiteren gibt es noch Sublime Text, Brackets, Atom und Aptana Studio, die nützlich sind.

Wie sieht die Library-Infrastruktur aus?

Eine Angular Library ist ein Angular Projekt, das alleine nicht laufen kann und vorher in eine Applikation eingebaut werden muss. Jeder kann die als npm-Package veröffentlichten Libraries in das eigene Projekt einbauen. Das Packaging von Funktionalitäten als Library zwingt die Artifacts in der Library dazu, von der Geschäftslogik der Anwendung entkoppelt zu werden.

Lernkurve des Frameworks

Die Lernkurve wirkt so als, ob sie nach unten gehen würde, das heißt sie ist sehr steil.

Konzepte hinter dem Framework

Die Architektur stützt sich auf einige grundlegende Konzepte. Die grundlegenden Bausteine des Angular-Frameworks sind Angular-Components, die in NgModules organisiert sind. NgModules sammeln zusammengehörigen Code in funktionalen Sets. Eine Angular-Anwendung besteht aus mehreren solchen Sets.

Components definieren Views, welche Elemente am Bildschirm sind, die Angular verwenden und modifizieren kann. Components verwenden Services, die Funktionen bereitstellen, diese hängen nicht direkt mit den Views zusammen. Services können als Abhängigkeiten in Components injiziert werden.

Modules, Components und Services sind Klassen, welche Decorators verwenden. Decorators sind Funktionen, die Metadaten an Klassen, Methode, Properties oder Parameter hängen.

Die Metadaten für eine Component-Klasse sind mit dem Template, welches eine View definiert, assoziiert. Ein Template verbinden HTML mit Angular.

Die Metadaten für eine Service-Klasse liefern Informationen, um die Dependency Injection mit Components durchzuführen.

Persönliche Meinung zum Framework

Wir sind keine großen Angular-Fans und würden viel lieber ein anderes Framework verwenden. Bei einigen Funktionen verstehen wir zwar, was die Funktionalität sein soll, finden aber, dass das entweder nicht deutlich genug funktioniert oder wir es falsch verwenden.

Empfehlung?

Nein

Quellen

Starten mit Angular

<https://angular.io/docs>

https://www.youtube.com/watch?v=unWCXyBOy8E&list=PLwG0hu96Rc0oK1MJ3zfFRPqpmMJJ7Bn_g&index=3 [19.Mai.2021]

Zusätzliche Informationen:

<https://de.wikipedia.org/wiki/Angular>

<https://clockwise.software/blog/best-angular-applications/>

<https://www.edureka.co/blog/advantages-and-disadvantages-of-angular/> [26.Mai.2021]

<https://programmertoday.com/angular-rest-api-calls/>

<https://www.willtaylor.blog/complete-guide-to-angular-libraries/>

<https://www.javatpoint.com/features-of-angular-8> [02.Juni.2021]

<https://angular.io/guide/architecture>

<https://hackr.io/blog/best-angular-ide-editor>

<https://angular.io/guide/schematics-for-libraries> [09.Juni.2021]