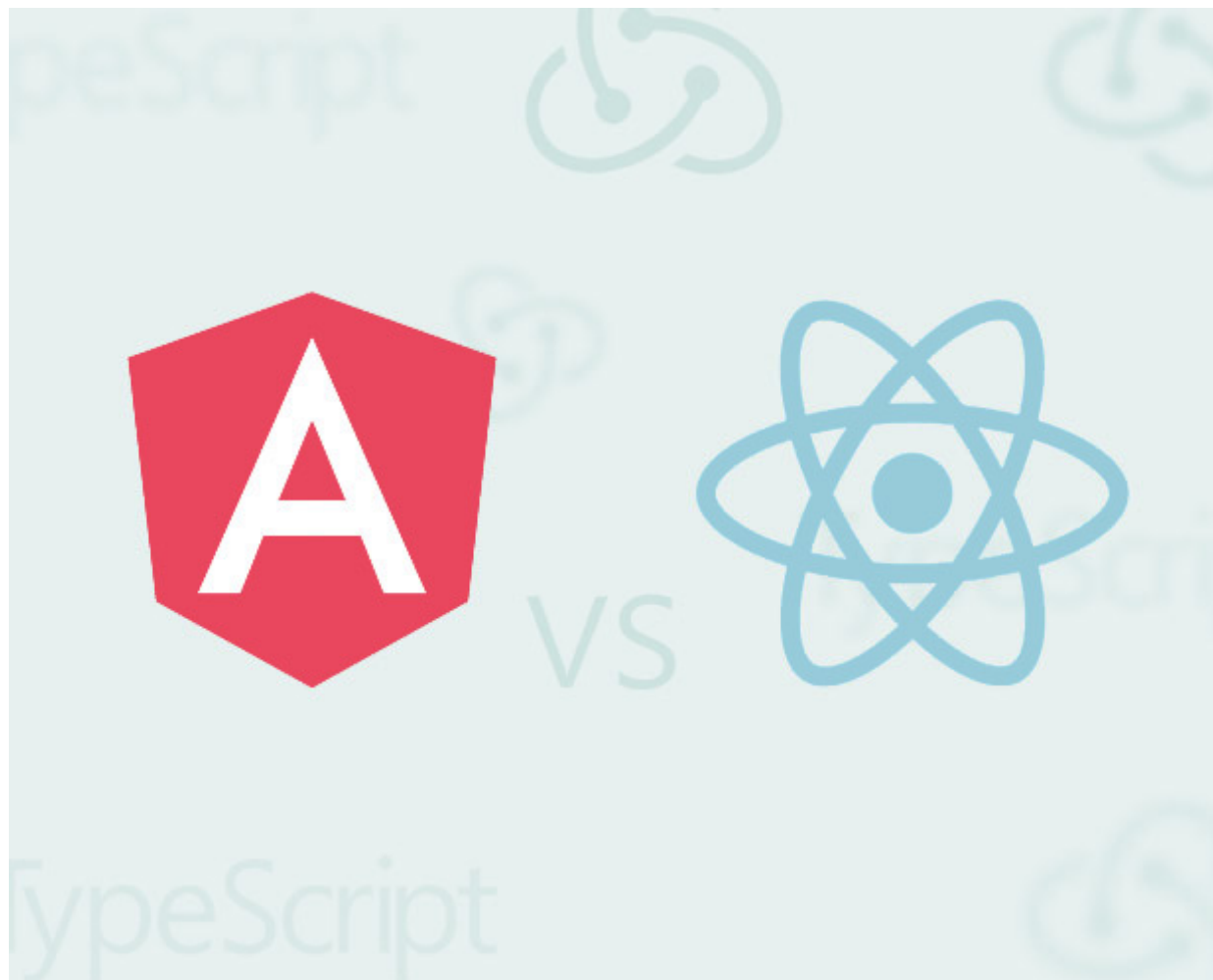


05.06.2018

Angular vs. React - Ein ausführlicher Vergleich. Welche Bibliothek passt zu Ihrem Projekt?



Inhaltsverzeichnis

1. [Einleitung](#)

2. [Angular](#)

3. [React](#)

4. Vergleich

5. Fazit

Die Anzahl der Frameworks für eine schnelle Entwicklung von Webseiten und Webanwendungen wächst ständig und stellt Entwickler immer wieder vor die Qual der Wahl, welche Technologie sich am besten für die Umsetzung eines neuen Projektes eignet.

In diesem Artikel werden die zwei derzeit sehr beliebten JavaScript-Technologien Angular und React verglichen, deren Vor- und Nachteile aufgezeigt, sowie auf deren Unterschiede bezüglich Komponenten, Templates, Data Binding, Performance, Features und Ecosystem eingegangen.

1. Einleitung

Angular und **React** werden für die Entwicklung von **Single-Page-Applikationen (SPA)** verwendet. Bei SPAs handelt es sich um webbasierte Anwendungen, die lediglich ein HTML-Dokument vom Server laden und Inhalte ohne serverseitiger Seitennavigation dynamisch nachladen. Webseiten können so auf Interaktionen und Events reagieren und Inhalte ohne Page-Reload laden. Somit wirkt die Website für den Nutzer dynamischer. Außerdem können Serverkosten für den Betreiber und der Trafficverbrauch für den Nutzer verringert werden, da bei Serveranfragen nicht ganze Seiten, sondern nur die benötigten Daten geliefert werden. Als Nachteile können (vom Projekt abhängig) höherer Entwicklungsaufwand und hohe Anforderungen an Entwickler genannt werden.

Da die Anforderungen an Entwickler bei SPAs viel höher sind, ist es sehr empfehlenswert, zu einem Framework zu greifen. Frameworks sind Gerüste für Softwareentwicklung, die Entwickler nach den eigenen Bedürfnissen einsetzen können.

Vorteile bei Einsatz eines Frameworks sind unter anderem, dass alles in einem erprobten Rahmen stattfindet und es meist Communities und eine Dokumentation dahinter gibt. Außerdem wird der Programmieraufwand durch Frameworks enorm erleichtert, da viele nützliche Funktionen oder Strukturen bereits dem Entwickler zur Verfügung gestellt werden. Durch klare Strukturen erleichtern Frameworks dem Entwickler auch, sauberen und strikt nach Patterns (bspw. MVC) getrennten Code zu schreiben und eliminieren so auch gängige Fehlerquellen. Zudem ist es erwähnenswert, dass die meisten Frameworks frei zur Verfügung stehen und für Entwickler keine zusätzlichen Kosten entstehen. Sowohl Angular, als auch React unterstehen der weit verbreiteten MIT Lizenz. Diese freizügige Lizenz enthält kaum Beschränkungen oder Verpflichtungen, erlaubt eine kommerzielle Nutzung und auch Wiederverwendung in Closed-Source Projekten.

2. Angular

Angular ist ein von Google entwickeltes Open Source Framework und wurde auf Basis von JavaScript entwickelt. Es wurde als Nachfolger von AngularJS von Grund auf neu geschrieben, weswegen sich Angular und AngularJS hinsichtlich einiger Punkte unterscheiden. Das zentrale Architekturkonzept ist eine Hierarchie aus Komponenten. Komponenten sind Javascript Klassen im MVC Prinzip welche beim Laden Template-Marker durch das bereit gestellte HTML ersetzen und durch eine spezielle Attribute-Syntax auf Events etc. reagieren können

Angular baut auf dem Model-View-View Model (MVVM) auf, daher sind Logik und Präsentation gut getrennt. Bekannte Unternehmen, die Angular zur Entwicklung deren Webseiten nutzen, sind unter anderem Youtube und PayPal.

Vorteile

Da Angular in TypeScript geschrieben ist, empfiehlt es sich für die Entwicklung auch TypeScript zu verwenden. Dies kann aber je nach Entwickler auch als Nachteil gesehen werden.

Webapplikationen, die mit Angular gebaut sind, haben (eine gute Clientleistung vorausgesetzt) eine sehr gute Performance.

Durch die Einführung von Werkzeugen wie dem Kommandozeilenwerkzeug Angular-CLI können Projekte schneller erstellt, erweitert und in Produktion überführt werden.

Angular liefert ein Grundgerüst zur Erstellung von Webanwendungen und kommt ohne zusätzliche Bibliotheken aus, die Verwendung ist aber möglich.

Unit- und End-to-End Testing wird erheblich erleichtert.

Nachteile

Angular ist durch das vorgegebene Gerüst relativ starr und unflexibel.

Um mit Angular arbeiten zu können, benötigt es eine gewisse Einarbeitungszeit.

Wenn ein User JavaScript im Browser deaktiviert hat, ist die Verwendung einer JavaScript-basierten SPA nicht möglich. Außerdem werden alte oder unbekannte Browser manchmal nicht unterstützt.

3. React

Bei **React** handelt es sich genau genommen um eine JavaScript-Bibliothek, welche von Facebook entwickelt wurde und seit 2013 als Open-Source-Lösung zur Verfügung steht. Unter anderem wurden Webanwendungen, wie Facebook, Instagram, Netflix oder Airbnb auf Basis von React entwickelt.

Im Gegensatz zu anderen MVC-Frameworks besteht React nur aus der View-Komponente. Diese wird in einzelne Funktionen bzw. Komponenten heruntergebrochen, welche den UI-Teil und die Logik der Applikation, also HTML und JavaScript beinhalten. Um den Code einfacher lesen zu können, stellt React in

Form von JSX eine optionale JavaScript-Syntax-Erweiterung zur Verfügung. JSX ist eine von React unabhängige Technologie und muss nicht zwingend verwendet werden. Es handelt sich dabei um eine statisch typisierte, objektorientierte Programmiersprache, die zu eigenständigem JavaScript kompiliert wird.

Vorteile

React ermöglicht Trennung von Daten und Präsentation.

Der Einstieg im Vergleich zu Angular ist simpel und erfordert nicht viel Vorarbeit.

Da React nur eine Bibliothek ist und nicht so viele Vorgaben hat, ist es für die Entwickler einfacher zu erlernen.

React arbeitet mit einem virtuellen DOM, mit dem der Aufwand einer Veränderung am tatsächlichen DOM extrem gering gehalten wird. Das sorgt dafür, dass die Anwendung sich immer sehr flüssig verhält, selbst wenn die dahinterliegenden Operationen oder Datenbankabfragen durchaus komplex sind.

Nachteile

React alleine reicht oft nicht aus um eine Webapplikation zu erstellen, weshalb die Verwendung zusätzlicher Bibliotheken in den meisten Fällen empfehlenswert ist.

Der Einsatz lohnt sich nur, wenn Webanwendungen sehr interaktiv gestaltet sind.

Die empfohlene Entwicklungssprache JSX muss meist noch erlernt werden. Dies sollte für Entwickler aber kein großes Problem darstellen.

4. Vergleich

Angular und React unterscheiden sich hinsichtlich einiger Punkte. Um einen akkuraten Vergleich zu ermöglichen, werden beide Bibliotheken zusätzlich hinsichtlich Komponenten, Templates, Data Binding, Performance, Features und Ecosystem verglichen.

Komponenten

Angular und React sind komponentenbasiert. Der große Vorteil von Komponenten ist, dass diese einfach wiederverwendbar, zusammensetzbar und einzeln testbar sind. Sie bieten die Möglichkeit eine Webanwendung in kleine Teile herunterzubrechen und mit einer konsistenten API zu schreiben. Eine Komponente erhält meist über Parameter eine Eingabe und wird dementsprechend angepasst und gerendert. Komponenten bestehen in der Regel aus einer View, die Daten dynamisch und aufbereitet erhält und anschließend anhand der HTML-Struktur der View anzeigt.

Templates

In React werden Templates mit JSX geschrieben. JSX ist eine HTML ähnliche Syntax, welche mit Inline-JavaScript erweitert wird. Dies hat zwar den Vorteil, dass alles an einem Ort ist und Code-Vervollständigung und Kompilierungs-Checks besser funktionieren, jedoch bricht es bewährte Best Practices, bei denen Template und Logik getrennt werden.

Angular Templates sind HTML-Strukturen, welche um Direktiven erweitert werden können. Somit sind die Templates flexibel und können auf verschiedene Inhalte anders reagieren. Während React JavaScript-Kenntnisse verlangt, ist es bei Angular erforderlich, die Angular spezifische Syntax der Direktiven zu erlernen. Dies sollte aber kein großes Problem darstellen, da diese Direktiven einfach zu erlernen sind und keine hohe Komplexität aufweisen.

Data Binding

Angular nutzt das Double-Bind-Feature, dadurch sind Elemente aus View und Model einander zugeordnet und ein Austausch in beide Richtungen ist problemlos möglich. Passiert also auf einer Seite eine Änderung, wird diese direkt auf die andere Seite übertragen, ohne dafür spezielle Funktionen schreiben zu müssen.

React hingegen nutzt One-Way Data Binding, das heißt Daten können nur in eine Richtung weitergegeben und verarbeitet werden. Dadurch ist es einfacher nachzuvollziehen, wie die Daten innerhalb der Applikation verteilt und verarbeitet werden. Es macht die Applikation performanter und erleichtert vor allem auch das Debuggen.

Performance

Um alle Änderungen am DOM zu erfassen, erstellt Angular für jede Bindung einen Watcher. Jedes Mal, wenn die View upgedated wird, werden die neuen Werte mit den alten Werten verglichen. Dadurch kann es bei größeren Anwendungen zu einer schwächeren Leistung kommen. Da React mit einem virtuellen DOM arbeitet, wird bei Änderungen der View das neue DOM mit dem virtuellen DOM verglichen und dementsprechend aktualisiert.

Features

Wie bereits erwähnt, bringt Angular mehr Features als React mit. Dies kann sowohl als Vorteil oder Nachteil betrachtet werden. Angular bietet viele Features, die für eine moderne Webanwendung erforderlich sind. Dazu zählen beispielsweise [Dependency Injection](#), Templates, Routing, Ajax Requests oder Unittest. All diese Features sind bei Angular von Beginn an verfügbar, bei React hingegen müssen Features erst über Packages installiert werden, da React selbst eher minimalistisch aufgebaut ist.

Ecosystem

Das Tolle an Open Source Frameworks ist, dass oft viele zusätzliche Tools oder Bibliotheken zur Verfügung gestellt werden. Sowohl für Angular, als auch für React gibt es dank der aktiven Community und Entwickler eine große Anzahl an Erweiterungen, die für die unterschiedlichsten Fälle eingesetzt werden können. Vor allem für erleichtertes Bootstrappen eines Projektes ([Angular CLI](#), [Create React App](#)), für die Entwicklung von Apps (Ionic 2 für Angular, React App), für Designangelegenheiten (Material Design, Material UI) oder auch für die Verwaltungen von States ([@ngrx/store](#) für Angular und [Redux](#) für React).

5. Fazit

Für Projekte, deren Ergebnis sehr interaktiv ist, **ist React zu empfehlen**. Auch wenn man auf weitere Bibliotheken angewiesen ist oder die Entwickler bereits Erfahrungen mit JSX haben, eignet sich React. Da es keine klaren Strukturen vorgibt, ist eine enge Zusammenarbeit des Entwicklerteams nötig. Außerdem wird serverseitiges Rendering von React besser unterstützt, was die Bibliothek **SEO-freundlich** macht.

Angular bietet im Gegensatz zu React sehr klare Strukturen und viele Features. Dadurch können Angular-affine Entwicklerteams schnell zur Umsetzung schreiten, ohne beispielsweise vorher Strukturen festzulegen oder zusätzliche Bibliotheken zu suchen. Da Angular so umfangreich ist, ist es allerdings für kleine Projekte oft zu überladen und bringt unnötigen Ballast mit sich.

Doch wie so oft kann keine allgemeine Empfehlung ausgesprochen werden, da die Wahl des richtigen Frameworks stark von den Anforderungen an das Projekt sowie von dem Entwicklerteam abhängig ist. Daher ist es empfehlenswert sich vor der Entscheidung folgende Fragen zu stellen und anhand dessen zu entscheiden, welches Framework sich am besten sowohl für das Team, als auch für das Projekt eignet.

Wie groß ist das Projekt?

Angular bringt bereits alles Nötige für den Aufbau einer Webanwendung mit, daher ist die Orientierung für den Anfang leichter, weil keine zusätzlichen Tools nötig sind. Ist das Projekt jedoch kleiner kann dies als unnötiger Ballast gesehen werden und es empfiehlt sich eher React zu verwenden.

Ist die gesamte Funktionalität klar definiert oder wird Flexibilität erwartet?

Angular bringt bereits viele Funktionen mit, bei React müssen oft zusätzliche Bibliotheken verwendet werden.

Welche Funktionen werden benötigt?

React eignet sich vor allem für sehr interaktive Projekte, in denen ohnehin weitere Bibliotheken zum Einsatz kommen und JSX für niemanden im Team ein Fremdwort ist.

Wird serverseitiges Rendering benötigt?

React unterstützt serverseitiges Rendering besser als Angular und macht die Webanwendung dadurch SEO-freundlicher.

Wie erfahren ist das Entwicklerteam?

Je nachdem mit welchen Frameworks das Team bereits gearbeitet hat und wie zufrieden sie damit waren, kann entweder auf bereits Bekanntes zurückgegriffen werden, da die Umsetzung so einfacher ist oder es wird ein neues Framework ausprobiert, um die Kompetenzen im Team zu erweitern.

WEITERE INTERESSANTE ARTIKEL



#blog #react #strapi #headlesscms #gatsby #graphql

Ist das Headless CMS Strapi bereit für die breite Masse?



Frontend-Entwicklung 2017 - Ein Vergleich aktueller Techniken zu 2007



Der 6-Schritte-Plan für einen erfolgreichen Relaunch Ihrer Internetseite

Alle Artikel →

Du bist auf der Suche nach Angular oder React Spezialisten

Abhängig von der konzipierten Lösung setzen wir die passende Technologie ein. Ob Native oder Cross-Platform, unsere Entwickler sind Full-Stack Spezialisten. Durch Agile Methodik hast du als Kunde stets den Überblick und die Kontrolle über das Projekt.

030 / 505 671 67

Montag - Freitag 10:00 - 18:00 Uhr

info@ventzke-media.de

Projekte

Agentur

Karriere

Services

Magazin

Kontakt

Downloads

Agenturpräsentation (PDF, 866 KB)

Projektablauf (PDF, 672 KB)

Blog

coopz 2.0 ist am Start - was kannst du erwarten von der Social Dating App?

Offene Stellenangebote

Frontend Full-Stack Entwickler (m/w/y)

Screendesigner / UX Designer (m/w/y)

Verbinden Sie sich mit uns



Services

Frontend Entwicklung

React JavaScript Experten

Angular Entwicklung
TYPO3 Agentur
WordPress Agentur
Produktkonfiguratoren

© 2020 VENTZKE MEDIA

[IMPRESSUM](#) [DATENSCHUTZ](#) [AGB](#)