AngularJS

Alexander Ruhdorfer

2016

Inhaltsverzeichnis

[Merkmale von AngularJS: 1](#_Toc464655236)

[Controller 1](#_Toc464655237)

[Routen 2](#_Toc464655238)

[Filter in AngularJS 2](#_Toc464655239)

[Direktiven 2](#_Toc464655240)

[Gründe für die entschiedene Technologie 3](#_Toc464655241)

# Merkmale von AngularJS:

* Dependency Injection
* Two-Way Data-Binding  
  Änderungen im Datenmodell wirken sich automatisch auf die Elemente in der Ansicht aus und umgekehrt   
  (vermeidet Boilerplate-Code).
* Benutzerdefinierte HTML-Elemente und -Attribute über Direktiven.
* Gute Testunterstützung inkl. Mocks

Module sind das gröbste Strukturmittel in AngularJS und kapseln eine Menge zusammenhängender Anwendungskomponenten. Sie dienen u.a. als Container für

* Controller
* Services
* Direktiven

AngularJS-Module sollen die folgenden Eigenschaften besitzen:

* hohe innere Kohäsion
* klar definiertes API
* hohes Maß an Wiederverwendbarkeit

# Controller

Controller sind Konstruktionsvorschriften für Scopes und definieren die Daten und Logik, die für eine bestimmte Ansicht benötigt werden. Jeder Aufruf eines Controllers bekommt einen eigenen Scope.

Wichtig bei der Erstellung von Controller:

* Controller sollten schmal designet werden
* Der Quellcode sollte aussagekräftig sein
* Komplexe Logik gehört in die Services
* Controller halten keine Daten, sondern sollten in Ihren Scopes nur Referenzen auf Daten bereitstellen.

# Routen

Routen dienen der Adressierung einzelner Anwendungsteile über eine URL und der Unterstützung der Vorwärts- bzw. Rückwärtsfunktionalität des Browsers (Deep Linking). Das Ermöglicht die Abbildung von URLs auf Templates mit zugehörigen Controllern.

Beispiel für Routing:



# Filter in AngularJS

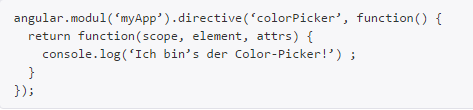
* Formatierungsfilter
* Collection-Filter
* Eigene Collection-Filter

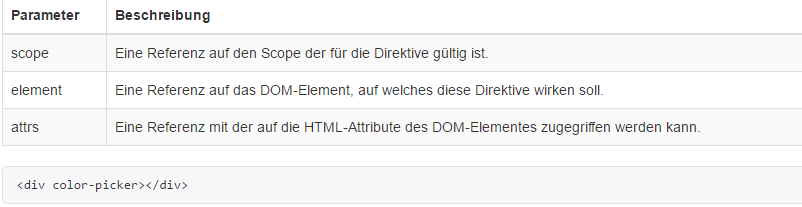
# Direktiven

Direktiven stellen ein Alleinstellungsmerkmal von AngularJS dar. **Direktiven erweitern das Vokabular von HTML**. ***Mit Direktiven können neue HTML-Tags und Attribute erstellt werden***. Sie sind ein mächtiges Feature, das die Erstellung von strukturiertem und wiederverwendbarem Anwendungscode ermöglicht.

Direktiven lassen sich mit der directive () -Funktion definieren. Diese erwartet zwei Parameter, den Direktivennamen und eine Factory-Funktion. Ein Direktivenname ist in den CamelCase-Schreibweise, mit einem Kleinbuchstaben beginnend – z.B. colorPicker - zu notieren. Wird die Direktive als HTML-Tag oder –Attribut verwendet, so ist die SnakeCase-Notation – z.B. color-picker - zu verwenden.

Beispiel:





# Gründe für die entschiedene Technologie

Für die Web Applikation wird AngularJS verwendet, weil:

* Es eine leichtere Kommunikation mit Node.js ermöglicht
* AngularJS deklarative Markups bietet
* Es werden Controller und Scopes in Angular verwendet
* Leichtere Einarbeitung als in ASP.Net
* AngularJS ein Front-End Framework ist
* Leichte Erstellung von Single Page Applications
* ASP.Net mehr ein Back-End Framework ist