

Angular 2 Geo-Apps mit YAGA

Stephan Herritsch und Arne Schubert

YAGA ist eine Sammlung von verschiedenen Modulen, Projekten und Tools zur Erstellung von webbasierten Geo-Anwendungen. YAGA wird von einem Entwickler-Team, bestehend aus Arne Schubert, Markus Strauß und Stephan Herritsch, als Hobby in der Freizeit entwickelt und gepflegt. Ein Teil dieser Entwicklungen sind als OpenSource-Projekte veröffentlicht und für jedermann nutzbar. Ebenso ist jeder dazu eingeladen sich mit seinem Know-how an den OpenSource-Projekten zu beteiligen.

Eines der jüngsten Projekte, mit dessen Entwicklung im November 2016 begonnen wurde, ist das Modul "leaflet-ng2". Mit leaflet-ng2 wurde eine Möglichkeit geschaffen das weit verbreitete Leaflet-Framework in Angular2 zu integrieren. Angular2 ist ein, im September 2016 veröffentlichtes, Framework zur Erstellung von Single-Page-Webanwendungen (WebApps) von Google. Mit Hilfe des auf Angular2 basierenden Ionic2, das im Dezember 2016 veröffentlicht wurde, können WebApps als nativ-hybride Apps im typischen "Look & Feel" der Endgeräte erstellt werden.

Bei leaflet-ng2 wurde stark darauf geachtet, dass die typische Leaflet Namensgebung beibehalten bleibt, was wiederum dafür sorgt, dass Außenstehende, die bereits Erfahrungen mit Leaflet haben, sich schnell in YAGAs leaflet-ng2 einarbeiten können.

Es wurde bewusst auf Softwaretests, in Form von Unit-Tests, geachtet, um eine geringe Fehleranfälligkeit zu gewährleisten, aber auch um eine schnelle Möglichkeit zu bieten, Fehler in Weiterentwicklungen zu lokalisieren. Zusätzlich wird auch die Abdeckung der Tests geprüft, die derzeit bei 99% liegt.

Im Gegensatz zu anderen entwickelten Angular-Leaflet-Frameworks, bei denen eine Directive für Leaflet zum Einsatz kommt, wurde bei leaflet-ng2 granular für die einzelnen Leaflet Klassen Angular-Directives erstellt.

Alle Optionen der Leaflet-Klassen wurden durch Input-Decorators mit Angular2 zur Verfügung gestellt, so dass ein nachträgliches Abändern zwar nicht im Funktionsumfang von Leaflet liegt, es jedoch in leaflet-ng2 möglich ist. Dort, wo Leaflet eine Möglichkeit bietet Werte zu ändern, werden über zusätzliche Output-Decorators die veränderten Werte automatisch im eigenen Datenmodell synchron gehalten. Dieses so genannte Two-Way-Databinding hat den Vorteil, dass Leaflet in ein MVC (Model-View-Controller) vollkommen eingebettet ist und damit Datenänderungen automatisch in der gesamten Anwendung übernommen und nicht mehrfach verwaltet werden müssen. Zusätzlich werden sämtliche Leaflet-Events über Output-Decorators an die Directives weitergeleitet.

Dieser Ansatz der Umsetzung bietet die Möglichkeit, direkt im Template, eine App weitgehend deskriptiv mit der bekannten Namensgebung von Leaflet zu erstellen, so dass für einfache Anwendungen hauptsächlich HTML-Kenntnisse von Nöten sind.

Wie mit YAGA leaflet-ng2 über ein Template eine Karte erstellt werden kann, wird im Folgendem exemplarisch dargestellt:

Angular 2 Geo-Apps mit YAGA

```
<yaga-map [lat]="48.5768558" [lng]="13.268283" [zoom]="10">
  <yaga-tile-layer [url]="http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png">
  </yaga-tile-layer>
  <yaga-marker [lat]="48.5768558" [lng]="13.268283">
    <yaga-popup>
      <h1>Hello World</h1>
      <p>This is Passau</p>
    </yaga-popup>
  </yaga-marker>
</yaga-map>
```

In diesem Beispiel wurde im Template eine YAGA-Map-Component erstellt, mit einer OSM-Karte, sowie einem Marker und einem Popup. Nach diesem Prinzip lassen sich sämtliche leaflet-ng2 Directives zum YAGA-Map-Component hinzufügen. Eine Auflistung der verfügbaren Directives ist in der Dokumentation zu finden. Das Beispiel mit Ionic2 für Android, iOS und Windows-Phone umgesetzt sieht wie folgt aus:

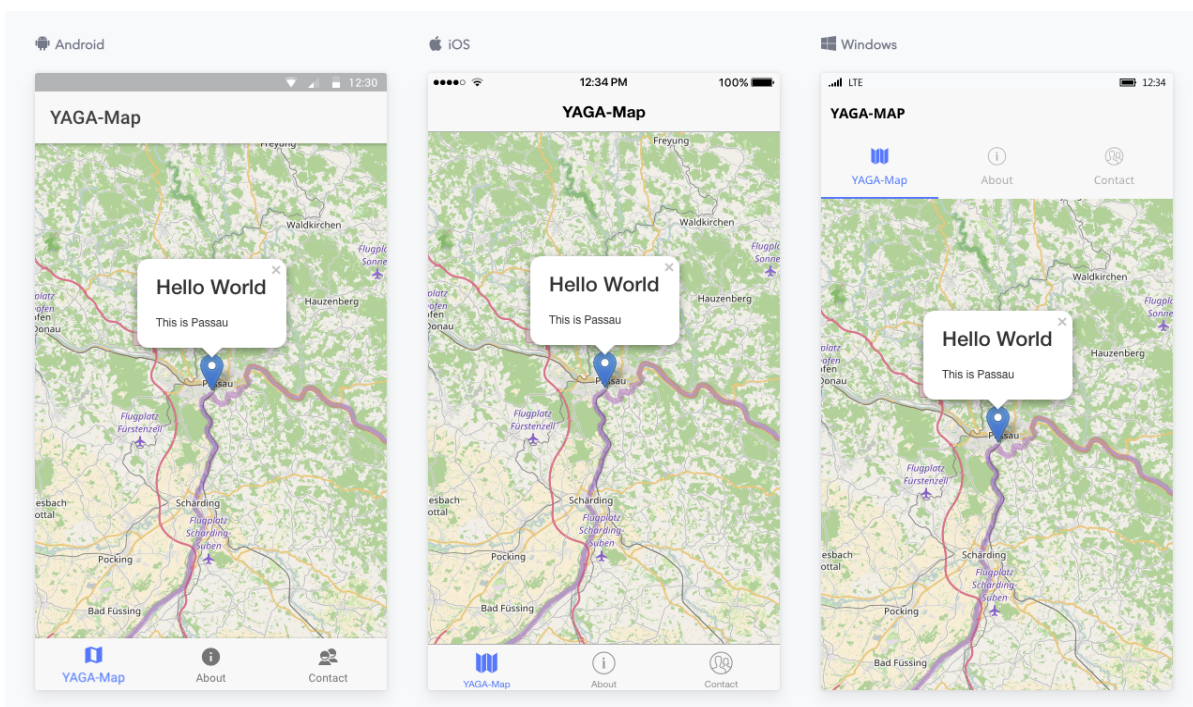


Abbildung 1: Beispiel mit Ionic2 für Android, iOS und Windows

Das YAGA Modul leaflet-ng2 beschränkt sich ausschließlich auf den Funktionsumfang von Leaflet. Durch die modulare Struktur kann problemlos mit weiteren Modulen eine Erweiterung des Funktionsumfangs realisiert werden. Vorstellbare Module sind unter anderem Gerätesensoren (z.B. GPS), Offline-Datenquellen oder Digitalisierung von Raster- und Vektordaten. Auf der Projekt-Webseite (leaflet-ng2.yagajs.org) sind neben der Dokumentation, die Softwaretests und - soweit umgesetzt - zu den einzelnen Directives, Beispiele aufgeführt.

Kontakt zu den Autoren:

Stephan Herritsch: s.herritsch@yagajs.org

Arne Schubert: a.schubert@yagajs.org