

WEA5-WETR

Manuel Hinterreiter (S1610307018)

20.01.2019

Contents

1	Weather-Tracer	2
2	Installationsanleitung	2
3	Angular-Komponenten	3
3.1	Search-Location	3
3.2	Login	4
3.3	Welcome	4
3.4	Dashboard	4
3.5	Station-List	4
3.6	Station-Detail-View	5
3.7	Admin	6
4	Angular-Services	7
4.1	PreferencesService	7
4.2	Wetr-Rest-Client	7
4.3	Auth-Service	8

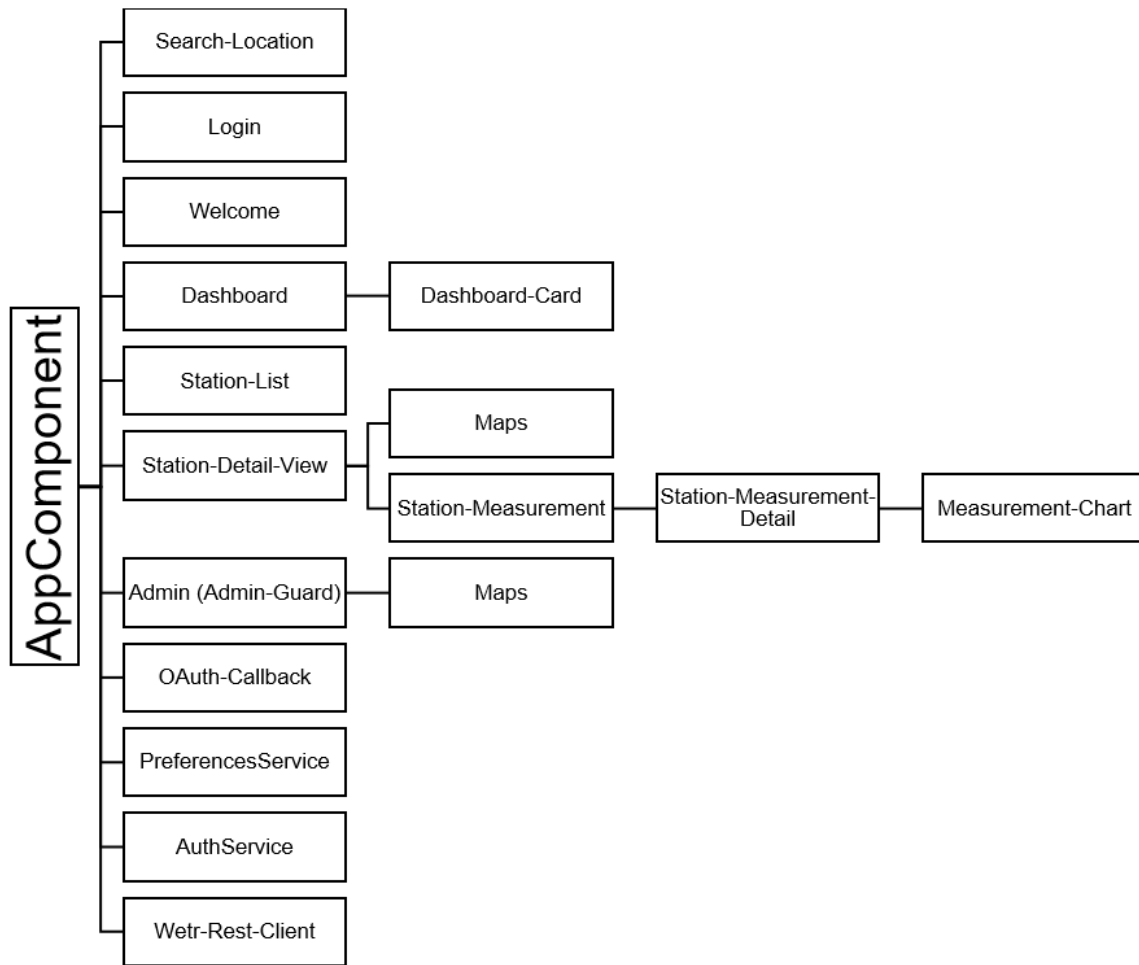
1 Weather-Tracer

Bei der Applikation **Weather-Tracer** handelt es sich um eine Angular Single-Page-Application. Die Applikation dient zur Darstellung von Wetterdaten. Authentifizierte Benutzer können sich beliebige Station in Ihr Dashboard speichern. Die Daten für die Applikation kommen über einen REST-Webservice, welche im Rahmen der Lehrveranstaltung SWK5 erstellt wurde. Die Swagger Definition für die REST-API befindet sich im Anhang dieses Dokuments. Als UI-Framework wurde **Angular Material** (<https://material.angular.io/>) verwendet. Für die Imports der dazu benötigten Komponenten wurde ein eigenes Modul dafür erstellt, welches die Imports dafür an zentraler Stelle übernimmt. Dieses Modul ist im AppModule als import definiert und somit verfügbar. Als OAuth Provider wurde eine eigene Applikation bei Auth0 angelegt (<https://auth0.com/>).

2 Installationsanleitung

Vorbedingung für eine korrekte Ausführung der Applikation ist die Installation von NPM. Weiters muss der dafür benötigte REST-Service erreichbar sein. Sind diese Bedingungen erfüllt, kann der beigelegte Source Code entpackt werden. Im Root-Folder der Applikation die entsprechenden node modules mittels *npm install* nachinstallieren. Danach auf der selben Ebene in der Kommandozeile *ng serve* ausführen. Browser starten und die URL *localhost:4200* eingeben - FERTIG.

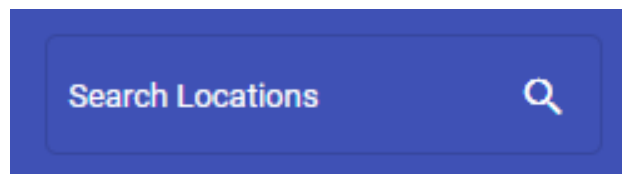
3 Angular-Komponenten



Hier werden die Zusammenhänge der implementierten Komponenten in Angular dargestellt. Die implementierten Services werden nur eine Ebene unterhalb der **AppComponent** dargestellt, da diese in sehr vielen Komponenten verwendet werden, und nicht jedes mal extra angeführt sind. Im Folgenden werden die einzelnen Komponenten kurz erklärt:

3.1 Search-Location

Diese Komponente ist in der Kopfzeile der Applikation eingebunden und in jeder Ansicht sichtbar. Hier kann nach PLZ-Codes, Gemeinden, Bezirke, usw gesucht werden.



3.2 Login

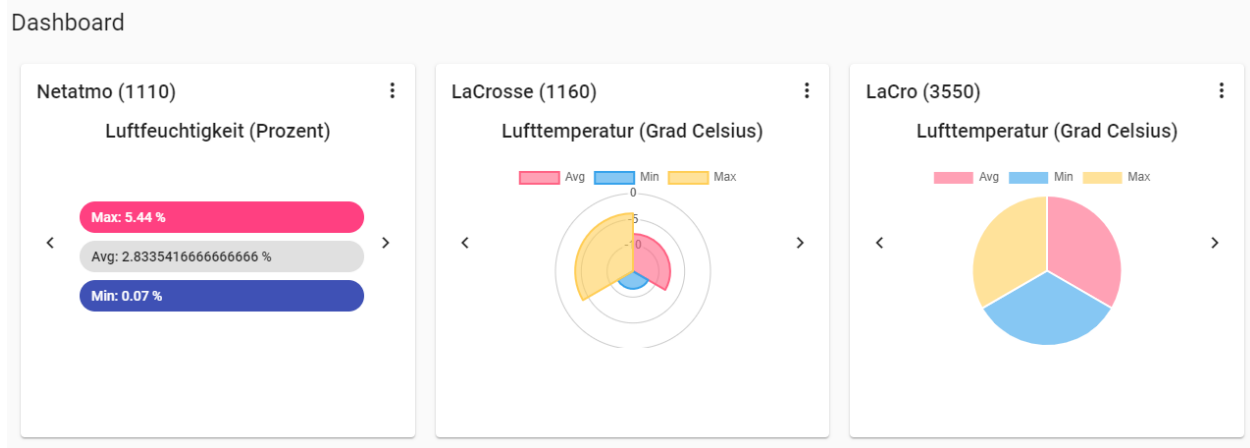
Diese Komponente wird dem User ebenfalls immer angezeigt. Hier kann ein Login bzw. ggf. ein Logout durchgeführt werden. Beim Login wird man auf die entsprechende auth0-Seite weitergeleitet. Danach wird man auf die OAuth-Callback Komponente zurückgeschickt. Im LocalStorage wurde die ursprüngliche Route zwischengespeichert und so wird der User automatisch zur ursprünglichen Seite zurücknavigiert.

3.3 Welcome

Nur ein Willkommensgruß.

3.4 Dashboard

Das Dashboard ist ausschließlich für authentifizierte Benutzer. Benutzer können sich Stationen im Dashboard abspeichern. Mit den Pfeilen nach links und nach rechts kann zwischen den Messwerten hin und her navigiert werden. Weiters kann der Benutzer zwischen 4 verschiedenen anzeigen (Pie-Chart, PolarArea-Chart, ...) wechseln. Diese Ansicht wird in der **Dashboard-Card**-Komponente verwaltet. Die Ansicht wird im LocalStorage gespeichert, sodass Sie für den User beim nächsten mal wieder bereitsteht.



3.5 Station-List

Die Stationsliste ist für alle Benutzer der Applikation sichtbar. Sie dient dazu, sich einen Überblick über die verfügbaren Stationen zu machen. Beim Auswahl einer Station kommt man auf die Detailansicht zu der jeweiligen Station. Authentifizierte Benutzer können mit dem Herz-Button auf der linken Seite beliebig viele Stationen zum Dashboard hinzufügen. Die Stationen können auch auf alle Felder beliebig gefiltert werden, um so zum Beispiel nur die Stationen aus einem bestimmten Bereich angezeigt zu bekommen.

Filter						
	Id	Name	Type	Post Code	Longitude	Latitude
♥	11448	Netatmo	NWS01	1110	47.0611191	13.4722157
♥	11444	LaCrosse	WS3650	1160	48.0222092	14.9661107
♥	11437	LaCrosse	WS2801	3390	48.0972061	15.3138847
♥	11434	LaCro	WS2500	3550	48.7249985	15.9722176
♥	11431	LaCrosse	WS2350	3580	48.7055511	15.5361118
♥	11426	LaCro	WS1613	3665	48.9110947	15.0277805
♥	11418	Huger	WMR918H	3950	46.0277863	13.3416634
♥	11453	Oregon	Scientific WMR180	4063	48.3527908	14.8805504
♥	11571	New Station	txxxxx	4232	0	0
♥	11573	coolq	nett type	4232	48.271925	48.271925

Items per page: 10 1 - 10 of 110 |< < > >|

3.6 Station-Detail-View

Die Detailansicht gliedert sich in zwei Teile: **Stationsdaten** und **Messdaten**. Im ersten Teil auf der linken Seite werden die Stammdaten der Station angezeigt. Die Stationsdaten können von authentifizierten Benutzer geändert werden. Nicht-Authentifizierte Benutzer sehen den **Create** nicht. Auf der rechten Seite ist eine Karte (**Google Maps**) eingebunden, welche den aktuellen Standort der Station anzeigt (Komponente **Maps**). Diese Ansicht kann aus platztechnischen Gründen auch zugeklappt werden.

11448 - Netatmo

Station Data

Name *

Netatmo

Type

NWS01

Address

Bad Ischl

Post Code

1110

Community

Wien

District

Wien 11_Simmering

Province

Wien

Latitude

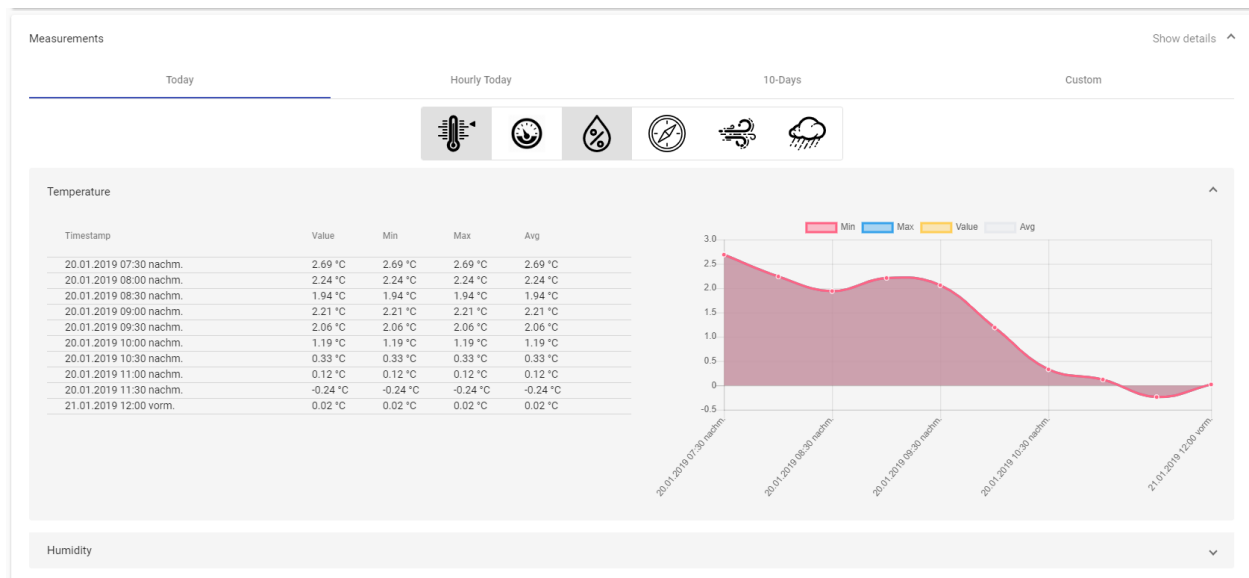
13.4722157

Longitude

13.4722157

Update

Im zweiten Teil der Ansicht bekommt der Benutzer die Messdaten für die Station zu sehen. Zuerst kann der Benutzer entscheiden, für welchen Zeitraum er die Daten sehen will. Es kann unterschieden werden zwischen: **Heute**, **Stündlich Heute**, **10-Tagesprognose** und **Benutzerdefiniert**. Danach kann der benutzer die für ihn wichtigen Messarten auswählen. Für jede ausgewählte Messart wird ein neues Accordion hinzugefügt, welches die Messdaten beinhaltet. Die Daten werden in tabellarisch und grafisch dargestellt.



3.7 Admin

Der Admin-Bereich kann nur von authentifizierten Benutzer eingesehen werden. Hier können entweder neue Stationen hinzugefügt werden und neue Messwerte hinzugefügt werden.

New Station

Name *	Type *
Address	
Post Code *	Community
District	Province
Latitude *	Longitude *

Create

New Measurement

Station Id *	Type of measure *	Value *
--------------	-------------------	---------

Insert

4 Angular-Services

4.1 PreferencesService

Der PreferenceService ist für die Speicherung der benutzerdefinierten Ansicht pro Benutzer für das Dashboard. Im LocalStorage wird für den Benutzer ein Preferences-Objekt abgelegt, welches die benötigten Daten zur Wiederherstellung des Dashboards beinhaltet. Die Preferences Klasse sieht folgendermaßen aus:

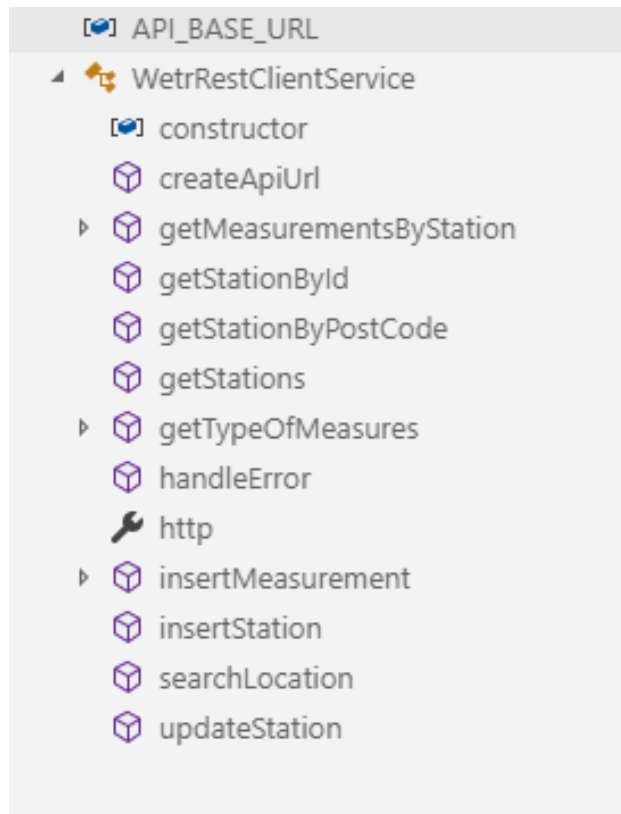
```
export class Preference {  
  constructor(  
    public stationId: number,  
    public tabType?: TabType,  
    public typeOfMeasures?: Array<number>,  
    public chartType?: string,  
    public typeOfMeasureIndex?: number  
  ) {}  
}
```

Hier ein Beispiel Preferences-Objekt, welches das Dashboard aus dem obigen Screenshots des Dashboards widerspiegelt.

```
▼ [{stationId: 11448, chartType: "", typeOfMeasureIndex: 2},...]  
  ▼ 0: {stationId: 11448, chartType: "", typeOfMeasureIndex: 2}  
    chartType: ""  
    stationId: 11448  
    typeOfMeasureIndex: 2  
  ▼ 1: {stationId: 11444, chartType: "polarArea", typeOfMeasureIndex: 0}  
    chartType: "polarArea"  
    stationId: 11444  
    typeOfMeasureIndex: 0  
  ▼ 2: {stationId: 11434, chartType: "pie", typeOfMeasureIndex: 0}  
    chartType: "pie"  
    stationId: 11434  
    typeOfMeasureIndex: 0
```

4.2 Wetr-Rest-Client

Jeder Request an das REST-API wird über diesen Services abgewickelt, somit ist der Zugriff auf den Webservice einheitlich und übersichtlich. Dieser Service beinhaltet folgende Methoden:



4.3 Auth-Service

Der Auth-Service kümmert sich um die Authentifizierung der Benutzer. Beim einem Login wird der Benutzer auf die `auth0.com`-Seite weitergeleitet. Bei erfolgreichen Login kommt der User über die OAuth-Callback-Komponente auf die vorherige Seite zurück.