****

# Software-Dokumentation Weather App Version 2.1

**Projekt:** Entwicklung einer Desktop-Applikation mit API- und Datenbankanbindung

**Thema:** Weather App

**Klassenbezeichnung:** E2FI3

**Lehrkraft:** Herr Stephan Baldes  
 Herr Benedikt Schumacher

**Schulfach:** BfK-S

**Ausbildungsberuf:** Fachinformatiker für Anwendungsentwicklung

**Abgabetermin:** 09. Juli 2023

Inhaltsverzeichnis

[Software-Dokumentation Weather App Version 2.1 0](#_Toc139288166)

[1 Das Projektteam 2](#_Toc139288167)

[3 Algorithmen 3](#_Toc139288168)

[3.1 Locate Me – Algorithmus 3](#_Toc139288169)

[3.2 Search City – Algorithmus 3](#_Toc139288170)

[4 Applikation 4](#_Toc139288171)

[4.1 Grafische Benutzeroberfläche 4](#_Toc139288172)

[4.2 Application Programming Interface 5](#_Toc139288173)

[4.3 Cities-Datenbank 5](#_Toc139288174)

[4.4 Klassendiagramm 6](#_Toc139288175)

## 1 Das Projektteam



**Luca Schäuffele***Frontend Designer*

Designen der grafischen Benutzeroberfläche in Paint und PowerPoint  
Suchen von passenden Grafiken für die Benutzeroberfläche



**Henrik Heiser***Frontend Developer*

Scrum-Master  
Entwicklung des Overlays der grafischen Benutzeroberfläche  
Anpassen der Steuerelemente der Oberfläche auf das Backend



**Merve Guenes***Frontend Developer*

Entwicklung des Overlays der grafischen Benutzeroberfläche  
Anpassen der Steuerelemente der Oberfläche auf das Backend



**Maurice Hofmann***Backend Developer*  
Auswahl und Testen von geeigneten „Weather“ und „Location“-APIs  
Entwicklung der Demo-Weather-App (V1.2)  
Entwicklung der Datenbankanbindung   
Entwicklung der API-Anbindung  
Entwicklung der Cities-, GeoLocator-, Weather-Klasse sowie des Hauptprogramms



**Hesham Mohamed Awadalla Osman***Backend Developer*  
Auswahl und Testen von geeigneten „Weather“ und „Location“-APIs

Auswahl einer geeigneten „Cities“-Datenbank

Entwicklung der Demo-Weather-App (V1.1) mit den dazugehörigen:

* Datenbankanbindung
* APIs-Anbindungen
* „Search city“ und „Locate me“-Algorithmen

Anbindung des App-Designs vom Frontendteam mit dem Backend

Entwicklung einer analogen Uhr zum Anzeigen der Ortszeiten

## 3 Algorithmen

### 3.1 Locate Me – Algorithmus

Der „Locate Me“-Algorithmus wird beim Starten der Applikation ausgeführt. Dabei wird der aktuelle, geografische Standort mithilfe von Längen- und Breitengraden mithilfe einer API über die IP-Adresse des Geräts ermittelt. Anschließend werden mithilfe dieser Daten, die aktuellen Wettereigenschaften über eine weitere API herangezogen und im Anschluss auf der grafischen Benutzeroberfläche im Main-Bereich dargestellt.



Abbildung 1 - Locate Me Algorithmus

### 3.2 Search City – Algorithmus

Der „Search City“-Algorithmus wird ausgeführt, wenn die Wetterbedingungen einer bestimmten Stadt angezeigt werden sollen. Dabei wird zunächst die „Cities“-Datenbank geladen. In der TextBox, auf der grafischen Benutzeroberfläche, kann eine beliebige Stadt gesucht werden. Für jedes eingegebene Zeichen wird eine SELECT-Abfrage an die Datenbank gesendet, welche die Städte, die der Nutzer-Such-Eingabe entsprechen, zurückliefert. Wählt der Nutzer im Anschluss aus der ListBox die gewünschte Stadt aus, werden aus dem gewählten Datensatz aus der Datenbank-Tabelle die Längen- und Breitengrade ermittelt. Anschließend werden mithilfe dieser Daten, die aktuellen Wettereigenschaften über eine weitere API herangezogen und im Anschluss auf der grafischen Benutzeroberfläche im Main-Bereich dargestellt.

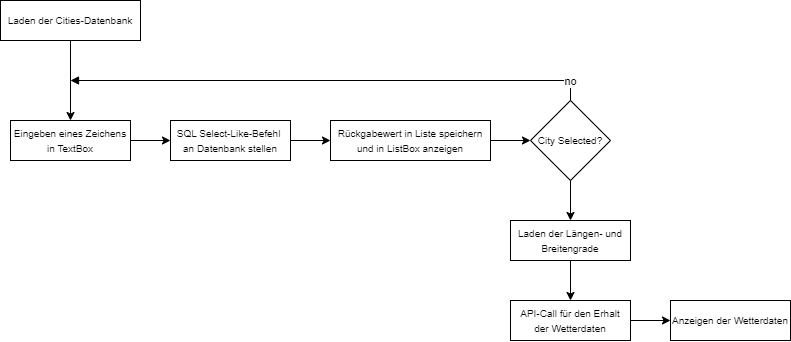


Abbildung 2 - Search City Algorithmus

yes

## 4 Applikation

### 4.1 Grafische Benutzeroberfläche



Abbildung 3 – Design idee

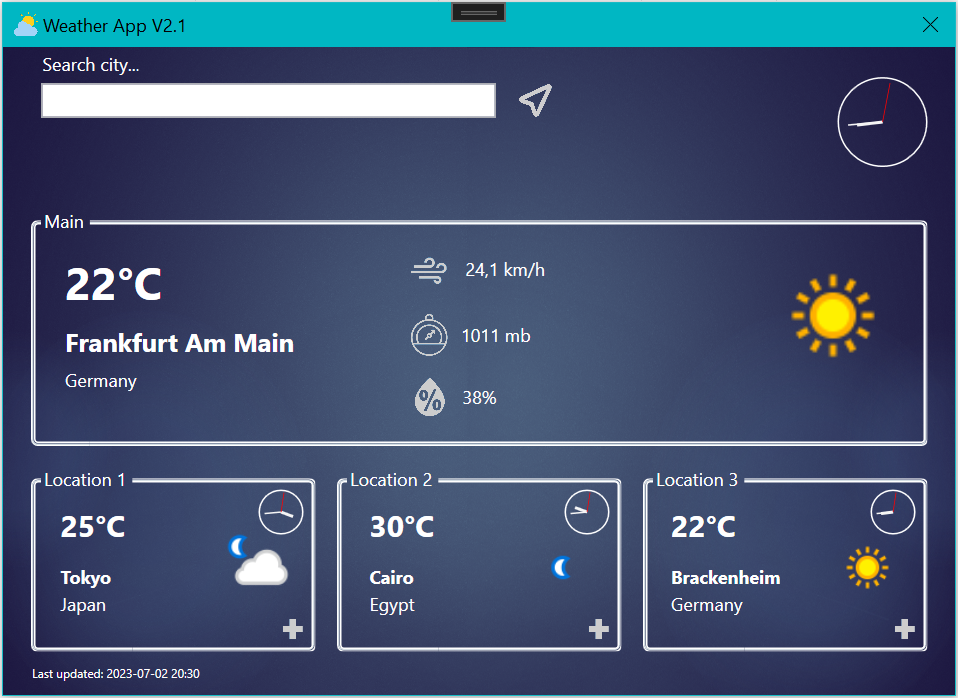


Abbildung 4 - Grafische Benutzeroberfläche Weather App 2.1

### 4.2 Application Programming Interface

Location Endpoint-URL: <https://api.ipgeolocation.io/ipgeo/>

Diese API liefert geografische Informationen mithilfe der IPv4- oder IPv6-Adresse im JSON-Format über eine sichere HTTPS-Verbindung. Es werden Daten wie Ländername, Ländercode, Stadt, Zeitzone, Bundesland, sowie Längen- und Breitengrade zurückgeliefert. Diese Längen- und Breitengrade werden zur Ermittlung der aktuellen Wetterverhältnisse benötigt, welche über die folgende API rückgeliefert werden.

Wetter Endpoint-URL: http://api.weatherapi.com/v1/current.json

Diese API liefert Wetterinformationen im JSON-Format nahezu in Echtzeit für Millionen von Standorten weltweit durch übergebene Längen- und Breitengrade. In der Benutzeroberfläche werden Grad Celsius, Stadt, Land, Localtime, Wind, Luftdruck, Luftfeuchtigkeit sowie zugehörige Sinnbilder grafisch aufbereitet und dargestellt.

### 4.3 Cities-Datenbank

Untenstehend finden Sie ein Entity-Relationship-Modell der „Cities“-Datenbank. Hierbei handelt es sich ausschließlich um eine Tabelle,



Abbildung - Tabelle cities aus der Datenbank

### 4.4 Klassendiagramm