

1 Teilnehmer/innen des Teams:

Klasse: AP23b	Team: Rahel Hüppi
------------------	----------------------

2 Anforderungsdefinition (Meilenstein A)

Wetter-APP

Fachlicher Inhalt:
(Allgemeine Beschreibung)

Kundennutzen: Mit dem Skript soll man über eine Webseite Informationen über das Wetter verschiedener Städte ansehen können.

Setup und Automation:

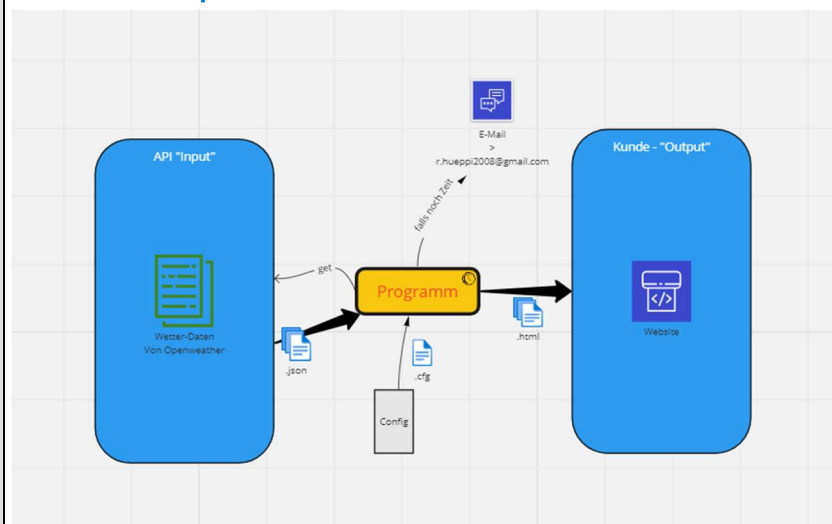
Der Kundenserver / -dienst ist eine Webseite über das Wetter verschiedener Städte.

Das Skript verarbeitet Informationen einer Wetter-API und stellt diese in einer Webseite dar.

Details:

- Konfiguration (.cfg): URL (API), API-Key
- Get-Prozedur (.raw): Die API abfragen und die Daten einzeln herauslesen.
- Verarbeitung (process): Daten so verarbeiten, dass ich sie danach darstellen kann. Das beinhaltet zum Beispiel das Umrechnen der Unix-Zeit des Sonnenuntergangs in unsere Zeit oder das Umrechnen der Grad in Celsius.
- Weiterreichung (.fmt): Die Daten in mein HTML-File bringen, damit diese dargestellt werden können.
- Sicherheitsaspekte: Falls es Probleme beim Abrufen der API gibt, soll eine Fehlermeldung ausgegeben werden.

Skizze / Mockup



Erkenntnisse aus der Machbarkeitsabklärung in Bash (oder Python):

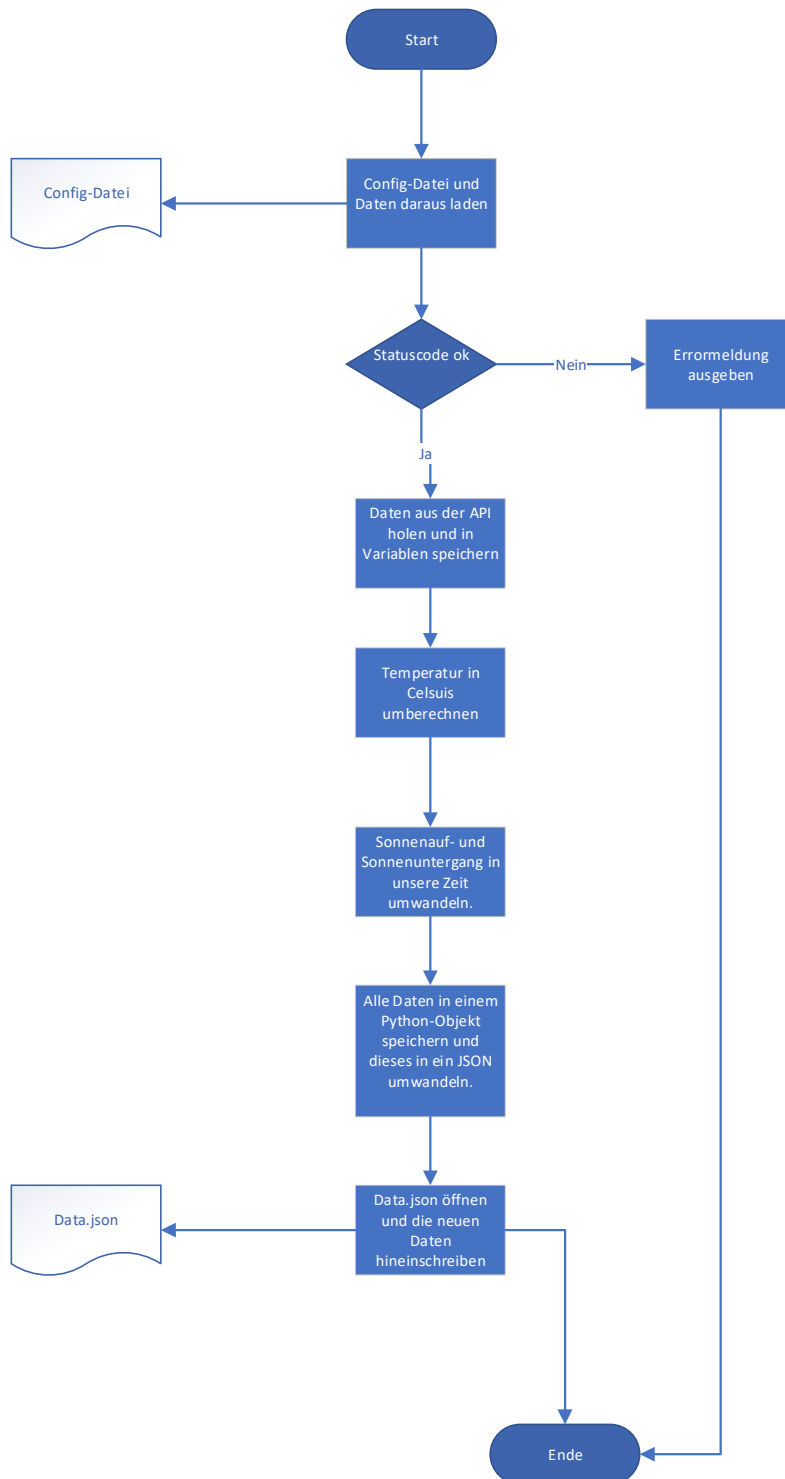
Folgende Features sind vorab untersucht worden und

	<ul style="list-style-type: none">• API: https://openweathermap.org/
MUSS Kriterien: (Konkrete Features, die umzusetzen sind)	Folgende Features sollen implementiert werden, um einen produktiven Ablauf sicherzustellen: (Siehe Vorgaben «Muss») <ul style="list-style-type: none">• API anfragen und Daten extrahieren• Umrechnen der Unix-Zeit in unsere Zeit.• Temperatur in Grad Celsius umrechnen• Daten in HTML, um es auf der Webseite abzubilden.• CSS, damit es etwas schöner aussieht (es muss nicht sehr schön sein)
KANN Kriterien: (Konkrete Features, die optional sind)	Folgende Features können zusätzlich implementiert werden: (Varianten, Kreativität) (Siehe Vorgaben «Gewünscht») <ul style="list-style-type: none">• Mein Mail einbauen. Z.B. falls angewählt soll die Abfrage ans Mail gesendet werden oder falls die Temperatur ungewöhnlich tief oder hoch ist soll eine Warnung kommen.• Skript automatisieren• Auf Webserver hochladen (automatisch)

3 Betriebsdokumentation (Meilenstein C: individuelle Aufgabe 3)

Für Administrator und Benutzer wird folgende Anleitung ausgeliefert

3.1 UML Aktivitätsdiagramm: weather.py



3.2 Installationsanleitung für Administratoren

Das Programm ist folgendermassen zu installieren und konfiguriert ...

Mein Bash-Script, um die Webseite auf dem Webserver hochzuladen funktioniert nicht.

Zum Aufsetzten würde es jedoch folgendermassen gehen.

1. Bashscript und Program-Files lokal abspeichern.
2. Pfade zu den Files im Bashscript anpassen.
3. Bashscript ausführen

3.3 Bedienungsanleitung für Benutzer

Das Programm ist folgendermassen zu bedienen ...

Wenn es auf den Webserver hochgeladen wurde, kann man es über den Browser ansehen. Das wäre mit dieser URL: hueppi-haab.ch/rahel.