

# Inhalt Info. Systeme HA Inhalt

## Ziel

In diese Unterrichtseinheit werden die Lehrplaninhalte LB 13.6 und 13.7 mithilfe der OPENSENSE Wetterstation, dabei auch der Umgang mit DLLs und OPP in der Praxis umgesetzt.

Der Ziel dabei ist diese Lerninhalte zu wiederholen und dabei für die Prüfung zu üben.

## Aufbau

Das Projekt besteht aus 3 Komponenten, die in den folgenden Absätzen näher erläutert werden.

Im Prinzip funktioniert es so, dass eine Wetterstation Temperatur, Luftdruck, Luftfeuchtigkeit, Beleuchtungsstärke und Strahlungsintensität eines Raums abmessen. Diese Daten werden in einem Programm gespeichert, das man mittels einer IP aufrufen kann und an unsere Formular anzeigen kann.

## Wetterstation

Die Sensoren und Steckplätze auf der Basisstation werden mittels der Verbindungskabel vernetzt. Als nächste werden diese Teile in der Outdoor-Gehäuse eingebaut.

Um die Wetterstation zu bedienen wird die Linux integrierte Entwicklungsumgebung von Arduino, kurz *Arduino IDE*, benötigt. Ein paar Python Bibliotheken, die bei Software aufgelistet sind, müssen installiert werden, um das Programm für die API zu programmieren.

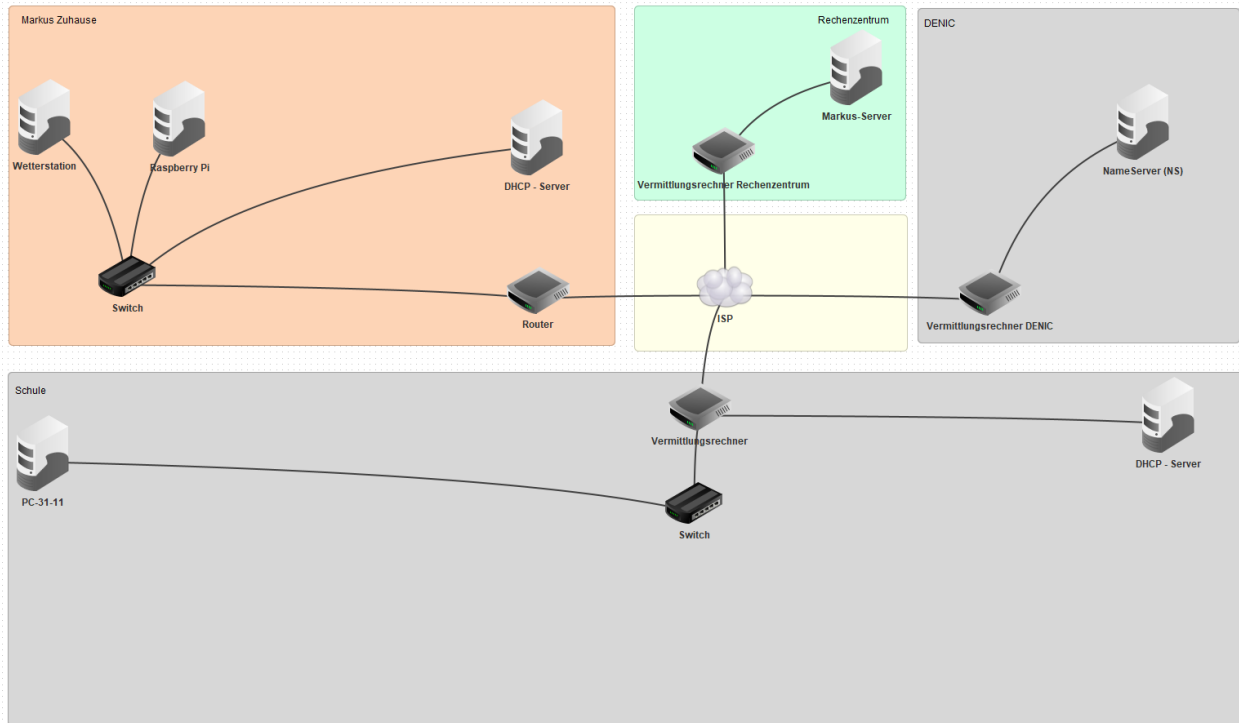
Um den genaueren Aufbau durchzulesen finden Sie die Einleitung von senseBox:home v1 auf [sensebox.de](http://sensebox.de).

Ein Serieller Port verbindet die Wetterstation mit dem Pi physische. Durch NgNix wird eine Website(<http://65.109.108.200/api/v1/senseBox/weather>) gehostet, wo man die Daten der Wetterstation in Echtzeit abfragen kann.

Diese Daten können dann in der Form als 'requests' abgefragt werden, und die Website wird ein Jason String zurückgeben. In dem Formular werden die Daten gespeichert und

im Ausgabefenster wiedergegeben. Der genaue Ablauf ist unter 'schulprojekt\_OPENSENSE-BOX/UploadData.py' zu finden. Folgendes finden sie einen Netzwerkdiagramm um den Ablauf zu veranschaulichen.

### 💡 Netzwerkdiagramm



Eine **API**(The PI Web API is a **RESTful interface to the PI syste**) besteht aus Protokollen und anderen Code Bausteinen zur Entwicklung und Integration von Anwendungssoftware. In einer PI für unseren Wetterdienst kann man beispielsweise die Datenquelle auswählen(die von Markus oder deine eigenen AP-Adresse), wo die entsprechende Daten herkommen.

## Form

Das Formular besteht aus eine GUI, wo man entsprechende Daten in Echtzeit abfragen kann, dem der Knopf "Daten Abfragen" gedrückt wird.

An der ersten Stelle der Methode X wird überprüft, ob eine IP ausgewählt wurde, falls nicht (if (string.IsNullOrEmpty(apiIP)) wird eine Warnung zurückgegeben mit der Anweisung eine IP auszuwählen. Wenn eine gültige IP ausgewählt wurde, werden die

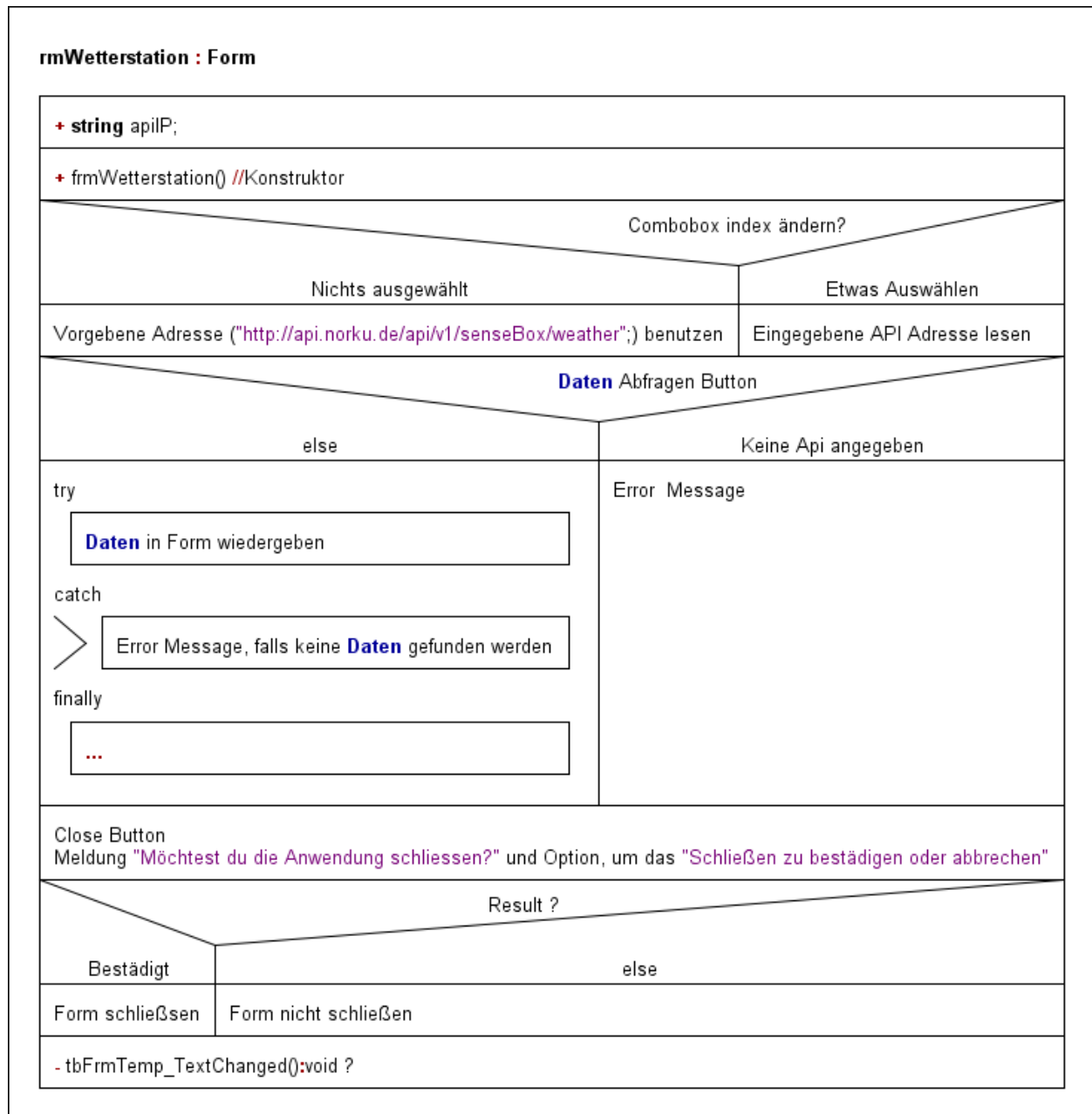
entsprechende Daten in die Textfelder zurückgegeben und ein Infotext mit der Uhrzeit der letzten Abfrage angezeigt.

Falls irgendein Fehler aufkommt, der in die DLL definiert wurde, wird eine Warnung angezeigt mit der Anweisung sich bei einem unserer Experten zu melden.

Das Programm kann unterbrochen werden, wenn man den "X" Knopf drückt, oder geschlossen, wenn man den Knopf "Close" auswählt. Dabei wird ein neues Fenster geöffnet, wo der Benutzer bestätigen muss, dass er die Anwendung schließen möchte.

Die Struktur ist im folgende Struktogramm zusammengefasst:

 Struktogramm

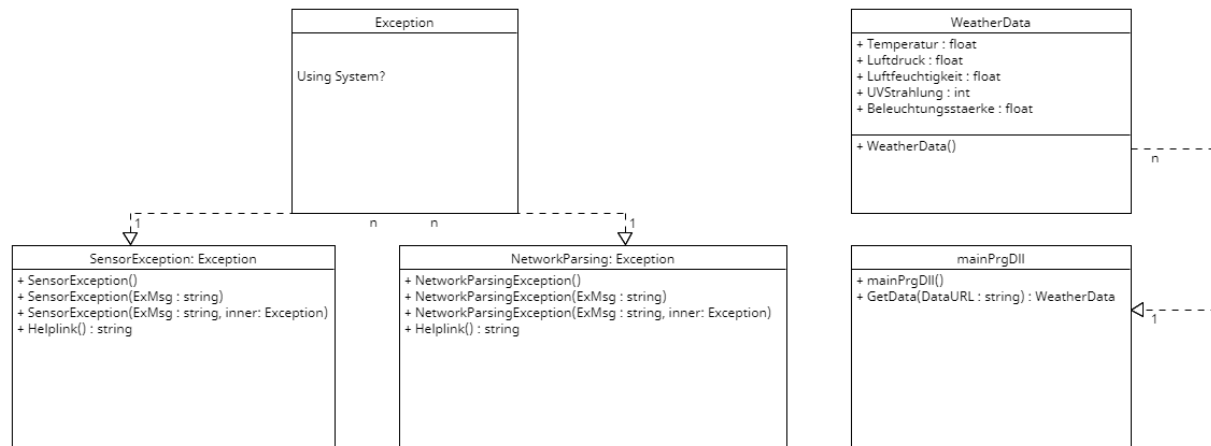


## DLL

Eine Dynamic Link library (DLL) beinhaltet Funktionen und Code Bausteine, die man in mehreren Programmen gleichzeitig verwenden kann.



Klassendiagramm



Bei Weatherdll werden die Daten Temperatur, Luftdruck, Luftfeuchtigkeit, Beleuchtungsstärke und Strahlungsintensität als Variablen und der Konstruktor `WeatherData()` in die Klasse `WeatherData` eingespeichert.

Die Klasse `mainPrgDll` hat eine Methode `GetData()`, um den Konstruktor von `weatherdll` aufzurufen.

Die vom System `Exeption` vererbte `SensorExeption` und `NetworkParsing` sind mit den `mainPrgDll` indirekt verbunden. Die Sensoren werden beim Aufrufen von `GetData()` auf ihre Gültigkeit überprüft, falls ein Fehler auftritt wird die entsprechende Error Message auftreten, sonst werden die entsprechende Daten an dasFormular von ein JSON String zu Text umgewandelt (JSON-Response Deserialize).

## Feedback

Was habt ihr von die Stunde mitgenommen?

Link

Ich könnte euch gedanklich Folgen 0 - 10

Die Aufgabbbben waren umsetzbar 0 - 10

Ich verstehe wie man DLLs in ein Programm implementiert 0- 10

Verbesserungsvorschlägen gern Privat ansprechen bzw. anschreiben.

## Impressung

Dieses Dokument dient nur zu Unterrichtszwecken.