

# Web.API

**Formål:**

At demonstrere opfyldelse af læringsmålet:

- Beskrive hvorledes man opbygger et Web Api som følger REST arkitekturen.

Her beskrives 2 opgaver: A og B, men du skal kun aflevere den ene, og du kan frit vælge, hvilken opgave du vil aflevere.

**Det skal gruppen aflevere:**

En journalen som indledes med en forside, hvor hver af gruppens medlemmer er anført med studienummer og navn. Journalen skal indeholde en kort beskrivelse af udviklingsforløbet, og dokumentere at serveren virker som krævet i opgaven.

Et bilag til journalen som skal indeholde alt det udviklede kode i en zip-fil, samt en video som demonstrerer at jeres WebAPI opfylder kravene.

# Opgave A

## Krav til anvend teknologi

Du vælger selv, om du vil løse opgaven med C# og ASP.Net Core eller med Python og Flask.

## Opgavebeskrivelse

Der skal programmeres et webAPI hvortil indlejrede systemer kan uploade temperaturmålinger mv. og andre klienter kan læse de uploadede målinger, eller blive notificeret om nye målinger via en WebSocket-forbindelse.

## Krav til web.API

Lav et webAPI, hvortil en vejrstation (et indlejret system) kan uploade vejrobservationer som serveren persisterer (lagrer) i en database.

En vejrobservation (måling) består af følgende data som sendes på json-formatet:

- Tidspunkt (dato og klokkeslæt) – du vælger selv hvilket tidsformat du vil anvende.
- Sted – består af felterne
  - Navn: string (navn på lokalitet).
  - Lat: double (gps koordinaten latitude).
  - Lon: double (gps koordinaten longitude).
- Temperatur – i grader celcius med 1 decimals nøjagtighed.
- Luftfugtighed – et heltal som angiver luftfugtigheden i procent.
- Lufttryk – i millibar med 1 decimals nøjagtighed.

Via web.api'et kan andre klienter hente de 3 seneste uploadede vejrdato.

Via web.api'et kan andre klienter hente alle vejrobservationer for en angiven dato.

Via web.api'et kan andre klienter hente alle vejrobservationer mellem et angivet starttidspunkt og et angivet sluttidspunkt.

## Krav til WebSocket forbindelse

Det skal være muligt for klienter at tilmelde sig til at få live opdatering, hver gang serveren modtager en ny måling. Dette skal implementeres ved brug af en WebSocket forbindelse - evt. ved brug af SignalR teknologi, hvis den valgte server understøtter dette.

## Krav til sikkerhed

En klient (vejrstation) skal være logget ind, for at den kan uploade vejrobservationer til serveren. Serverens api skal derfor indeholde endpoints til oprette nye klienter, og mulighed for at eksisterende klienter kan logge ind. Serveren skal benytte JWT som authenticationstoken.

## Krav til testning

At serveren opfylder de specificerede krav skal dokumenteres ved brug af programmet Postman. Og for studerende, som har eller har haft kurset Softwaretest gælder, at de også skal teste controllerklasserne med nogle udvalgte unittests.

## Opgave B

### Krav til anvend teknologi

Du vælger selv, om du vil løse opgaven med C# og ASP.Net Core eller med Python og Flask.

### Opgavebeskrivelse

Lav en web server som kører på en Raspberry Pi og som har funktionalitet til at tænde og slukke for en lampe/diode samt kan aflæse en værdi på Pi'en - f.eks. om en kontakt er on eller off. Endvidere skal serveren kunne udsende en alarm via WebSocket-protokollen ved tryk på en kontakt.

### Krav til web.API

Lav et webAPI hvormed en klient kan:

- Hente status på en kontakt (open eller closed).
- Tænde eller slukke for en lysdiode.

### Krav til websocket forbindelse

Det skal være muligt for klienter at tilmelde sig til at få alarm, når det trykkes på en kontakt forbundet til R.Pi. Dette skal implementeres ved brug af en WebSocket forbindelse - evt. ved brug af SignalR teknologi, hvis den valgte server understøtter dette.

### Krav til sikkerhed

En klient skal være logget ind, for at kunne tænde eller slukke for en lysdiode. Serverens api skal derfor indeholde endpoints til oprette nye klienter, og mulighed for at eksisterende klienter kan logge ind. Serveren skal benytte JWT som authenticationstoken.

### Krav til testning

At serveren opfylder de specificerede krav skal dokumenteres ved brug af programmet Postman. Og for studerende, som har eller har haft kurset Softwaretest gælder, at de også skal teste controllerklasserne med nogle udvalgte unittests.