

## Opis komponentów i komunikacji pomiędzy nimi:

### Serwer:

Zadaniem serwera jest gromadzenie danych z pomiarów temperatury, przechowywanie ich oraz udostępnianie statystyk, w naszym przypadku jest to średnia temperatura zebranych odczytów temperatury. Serwer składa się z 3 komponentów:

- > serwer REST, służący do odbioru i zapisu pomiarów. Aplikacje klienckie mogą przysyłać dane za pomocą metody POST na endpoint

`/measurements/temperature`. Format danych:

```
{"deviceId":123, "value":23.46, "date":"2020-05-29"}
```

- > dokumentowa baza danych Mongo DB służąca do przechowywania wartości pomiarów

- > Kafka - producent wiadomości, udostępnia średnią temperaturę pomiarów na topicu `temp`

Wszystkie elementy serwera są skonteneryzowane i zestawione w łatwe do uruchomienia środowisko.

### Generatory danych:

Celem generatorów danych jest zbieranie informacji o temperaturze i eksportowaniu ich na Serwer. Zaimplementowano dwa generatory, jeden w środowisku .NET/C#, drugi w języku Python. Obie wersje posiadają po dwa komponenty zbierające dane: tworzenie danych losowo i odczytywanie temperatury z procesora przy pomocy api do OpenHardwareMonitor. Każdy odczyt temperatury zostaje spakowany do obiektu JSON i wysłany na serwer zgodnie z modelem opisanym powyżej.

### Klienci:

Ostatnim elementem tej aplikacji są komponenty klientów, które wyświetlają wiadomości nadane przez Serwer przy pomocy potoków Kafki. Klienci zostali zrealizowani jako aplikacje konsolowe w języku C i Python.