Asynchronous communication

Sensor monitoring system and real-time notification

The system consists in two applications, one being a producer which simulates the reading of a sensor which sends data to a messaging queue every 10 minutes, the other being the consumer, which reads from the message queue and preprocesses the data, inserting hourly the computed consumption of the sensor into the database. In case the consumption at some point is greater then the maximum admitted, a notification is sent to the user via a websocket.

Producer

Architecture:

* The application is divided into a service, a controller and a module: the module establishes the connection to the rabbitmq cloud, the service reads the data from the .csv file into an array, the controller sends elements from the array every 10 minutes into the queue

Consumer

Architecture:

* The application has the models and dtos for the devices, energy consumption and users from the first application as well as the device service, which now uses all three repositories, in order to find the corresponding device for which the consumption is being computed, the user to whom the device is assigned and the method to insert the consumption into the database
* The module of the application establishes connections to the rabbitmq cloud and the database
* Here, a websocket server is created
* The preprocessing of the data is being done by accumulating the consumptions received per hour
* If the computed value exceeds at some point the maximum supported value of the device, a notification is being sent to the frontend

Frontend application

Changes have been made to create a websocket client which listens to the server created in the consumer. When a notification is received, it is shown to the user as a push notification.

Graphical user interface, table

Description automatically generated

Deployment diagram

Diagram

Description automatically generated