PROJECT

Overview:

The general purpose of the project is to transmit the location information of the sensors from the 2 existing sensors and the angles they determined towards the target to the central unit. Central unit be able to determine the X and Y coordinates of the target by processing the incoming data from sensors.

Tech Stack & Architecture & Structure:

- Apache Kafka was used as a tool for the sensors to transmit data to the central unit.
- A producer-consumer structure was established between the sensor and the main unit, and the sensors were designed as threads.
- The line equation was created with the coordinate and angle information coming from the sensor and the intersection point of these lines was determined as the target.
- apache.kafka.clients version 3.3.1 library is used.
- Java 17 is used.

Operation:

With the producer-consumer structure created, the sensors act like threads and each sensor sends the location and target angle information within the "run" method using the send method of the KafkaProducer object created in this class. Since I don't really have a sensor in my hand and there is no real data coming in, I used randomly assigned numbers. Then, I listened to these messages sent with KafkaProducer on the Consumer, namely Main Unit side. For this, I used KafkaConsumer structure. Also, since we have more than one producer structure (thread) here, I used ThreadPoolExecutor. I kept the data from the Consumer in the ConsumerRecords structure and created the MainUnitThreadHandler structure for each incoming record. To summarize, this structure actually prints the value of the incoming ConsumerRecords to the console. When the number of consumer records that follow is 2, I ran the method I wrote named findTarget. This method calculates the line equations according to the given angle and coordinate, finds the intersection point of the 2 lines, that is, our target, and writes the x and y values it finds to the console.
Actually, before I set up the multi-thread structure here, I wanted to use Spring Boot and I was sending the sent messages with a REST service, but then when the multi-thread structure was created, I thought that there should be 2 different requests for 2 different sensors, not a single request. At the same time, since I do not take information such as location, angle as input and create it in a random way, it made more sense to use Kafka as an intermediary.

Hilal Yavuz