APIs to Fetch Data

The Application developer needs to develop a logic server that handles the logic of the query given to the platform. The logic server will process the query sent to the platform requesting the desired data. For e.g. receiving an alert when the temperature of sensor 1 rises above 30 degree Celsius. These kind of conditions are handled by the logic server. The logic server has at its disposal three APIs available:

1. Query API: The Query API has parameters such as type, location (GPS coordinates) and range. This Query API return a list of all the sensors that match the criteria of the condition. This is required for instance when Application demands a temperature of a specific area only. Then the logic server needs to check whether there is a sensor available in that are or in a location nearby. The response is in JSON format containing the list of sensors.

{"sensors":[  
    {"id":"1", "type":"temperature", "location":"103", "alive":"yes"},

{"id":"2", "type":"humidity", "location":"101", "alive":"yes"},

{"id":"3", "type":"smoke", "location":"105", "alive":"no"},  
 ]}

1. Command API: The Logic Server can register commands onto a Filter server to fulfil the user applications demands. The command can be of any form. Eg. Fetch the data of sensor with id = 2 or Fetch the data of sensor with type = “temperature” and location = “103”. This call is directly registered with the Filter server and it starts sending the required data to logic server as and when it receives it.
2. Callback API: The CallBack API is the one in which data is sent by filter server based on some condition. For e.g. send me the humidity data when the humidity crosses 25 unit mark. In this case, the data is only sent when the value crosses this limit. All the other times when the value is lower than 25, the data is discarded by Filter server and not forwarded to logic server. The response received is in JSON format which needs to be parsed accordingly by the developer to display on the application.

{"sensors":[  
    {"id":"1", "type":"humidity", "location":"103", "data":"26.7", "timestamp":"reading\_time"},

{"id":"2", "type":"humidity", "location":"101", "data":"30.7", "timestamp":"reading\_time"},

{"id":"3", "type":"humidity", "location":"105", "data":"27.2", "timestamp":"reading\_time"},  
 ]}