iSupport Platform

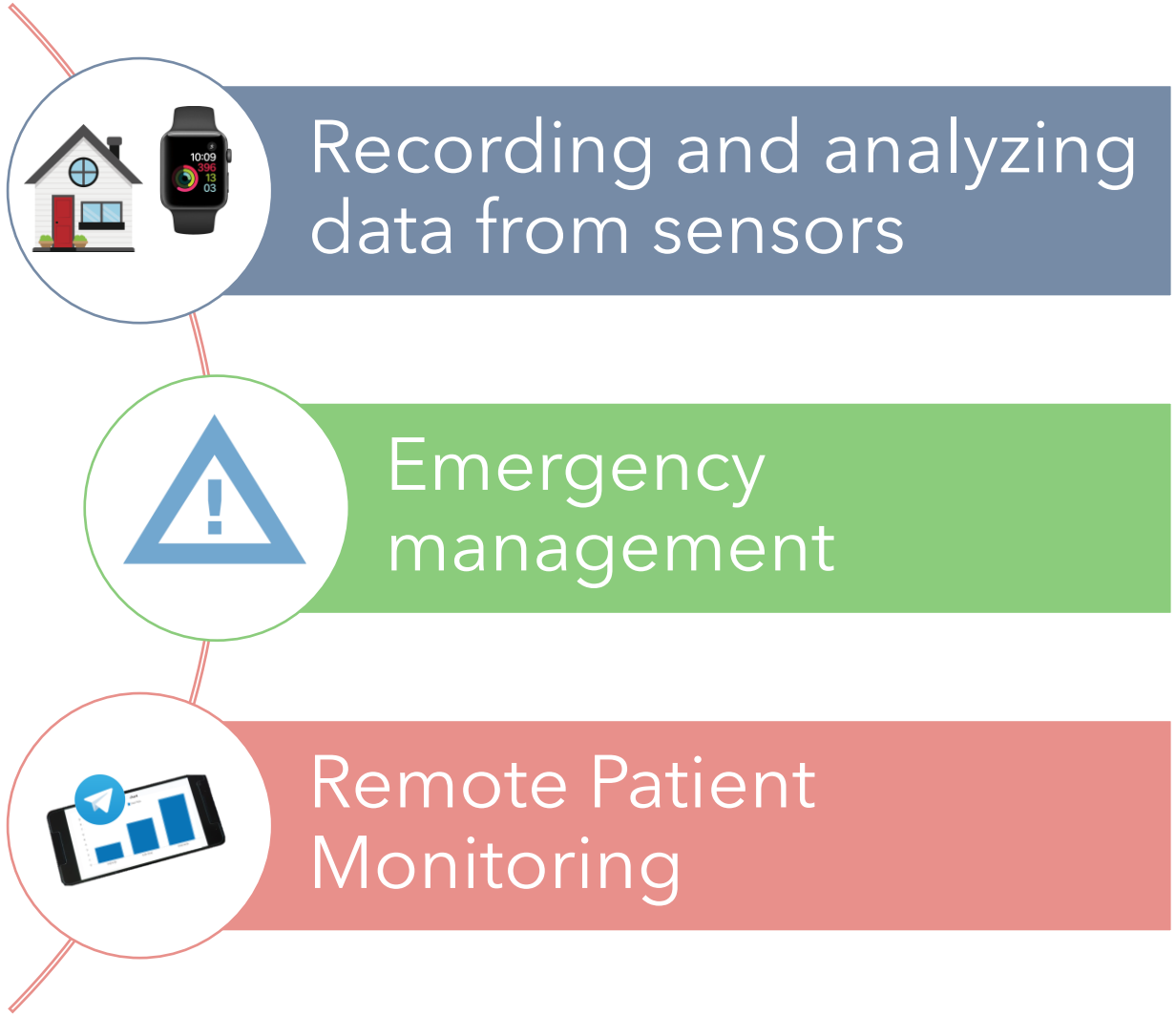
For your daily care

Team members:
Burdisso Sara s275424
Centonze Ilenia s277389
Ferrara Mauro s276932
Pasotti Andrea s275100

The idea - overview

Objective:

To help clinicians and caregivers follow individuals in depressive states during their treatment period by providing a remote monitoring and support platform.



Protocolli di Comunicazione

- *REST Web Service*

It allows to synchronize the communication between the parties.

Catalog: accesses the system information present in a JSON file and communicates it to the other actors;

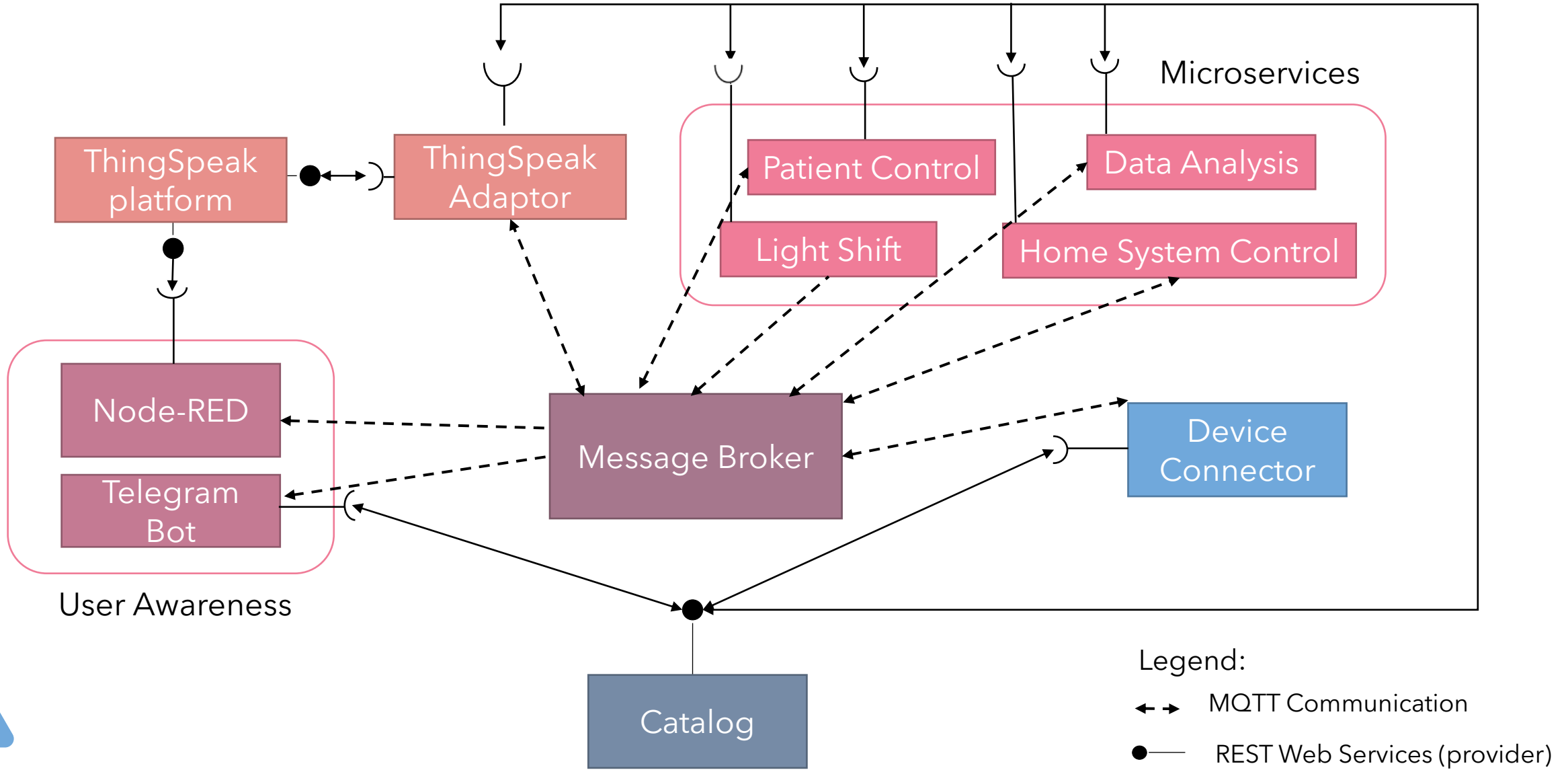
ThingSpeak: database with data coming from the sensors. The data is accessed with a GET request and provided via a message in JSON format.

- *MQTT*

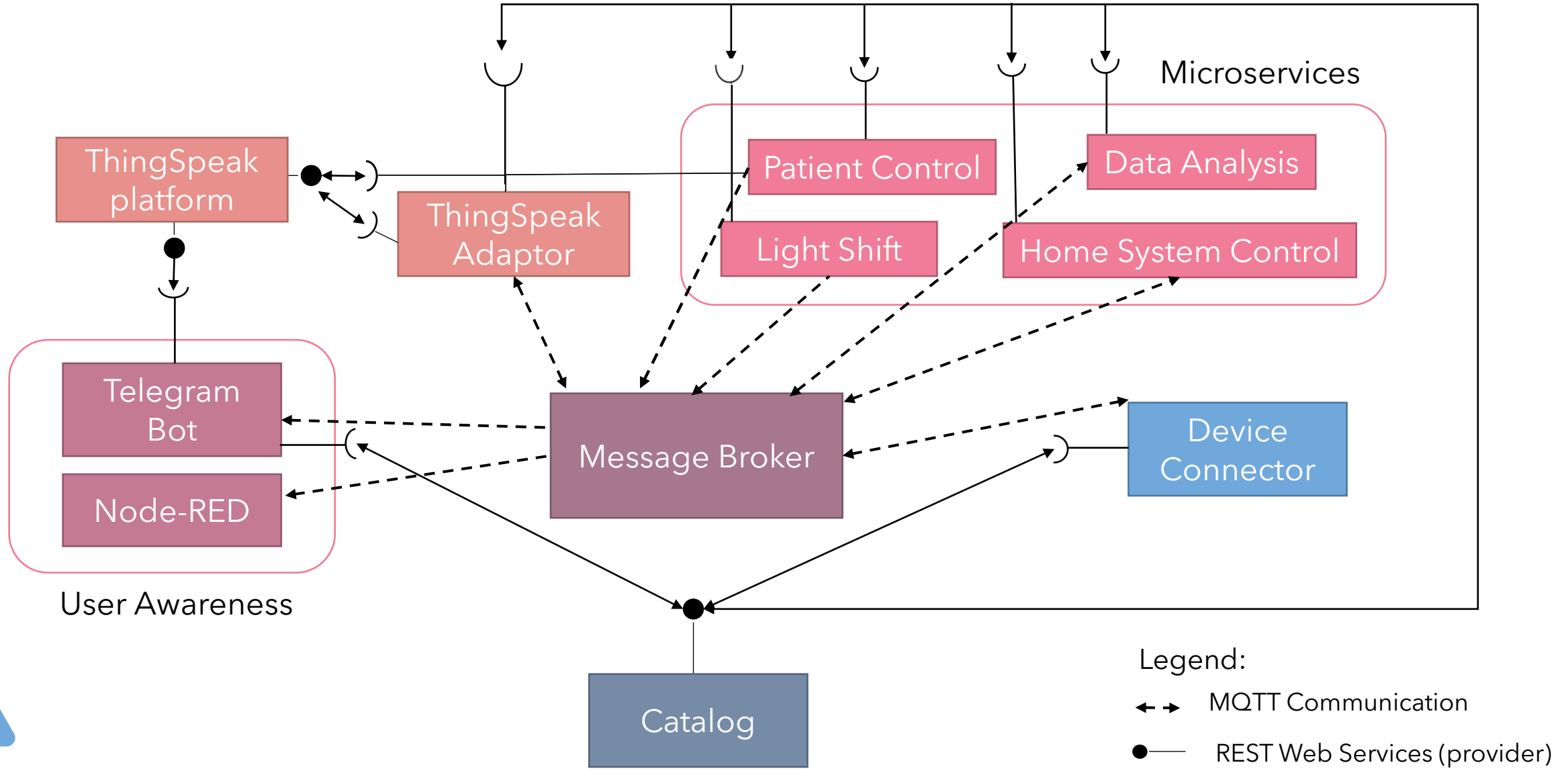
Lightweight and flexible protocol suitable for ioT applications.

- **Publish** messages in JSON format using QoS 2
- **Receive** messages using QoS 2
- **Message topic**: hierarchical, containing patient information

Use Case Diagram - proposal



Use Case Diagram - modified



Catalog

- It communicates with all the other actors in the platform exploiting REST communication.

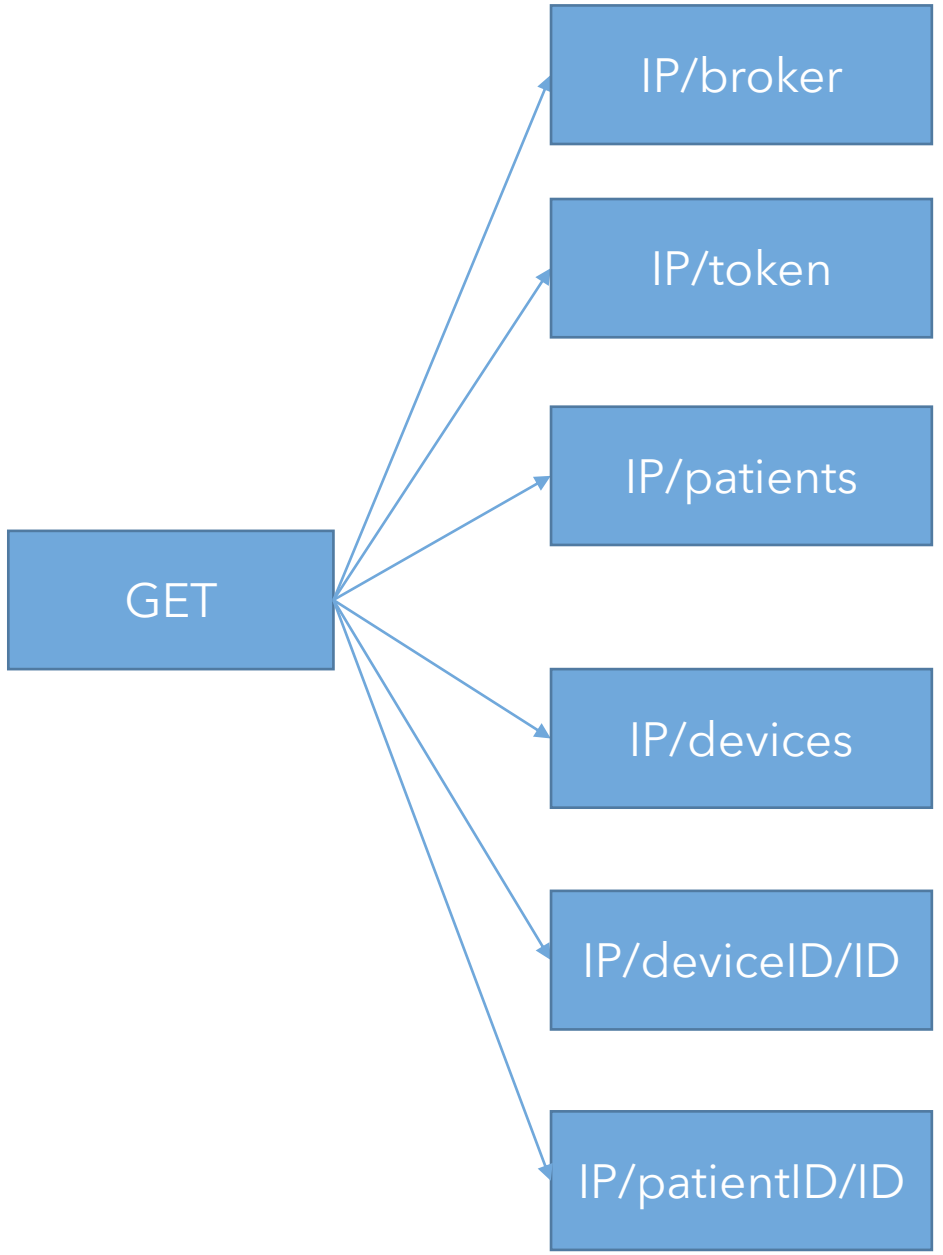
Catalog

- Catalog.json contains information about general configuration, registered devices and registered patients.


```
{
    "broker": {
        "IPaddress": "test.mosquitto.org",
        "port": 1883
    },
    "token": "1711013176:AAH6bCdvA0dOdL8fX4FIfCPGEqpEFNjN1d4",
    "deviceList": [
                {
            "deviceName": "smartwatch_1",
            "deviceID": "1",
            "measureType": [
                "HeartRate",
                "Accelerometer"
            ],
            "availableServices": [
                "MQTT"
            ],
            "servicesDetails": [
                {
                    "serviceType": "MQTT",
                    "topic": [
                        "iSupport/1/sensors/Body"
                    ]
                }
            ],
            "lastUpdate": 1621764472.704841
        }],
    "patientList": [
        {
            "patientID": "1",
            "patientName": "Mario",
            "patientSurname": "Rossi",
            "LightsSchedule": "7:30-8:30",
            "connectedDevices": [
                {
                    "measure": [
                        "Heart Rate",
                        "Acceration"
                    ],
                    "deviceID": "smartwatch_1"
                },
                     "telegramIDs": [1234567],
                     "thingspeakInfo": {
                "apikeys": [
                    "ZQ2AJZABDSC4MXXG",
                    "SV1HOET58CE7225M"
                ],
                "channel": 1342741
            },
            "lastUpdate": 1622981412.5148401
        }
    ]
}
```

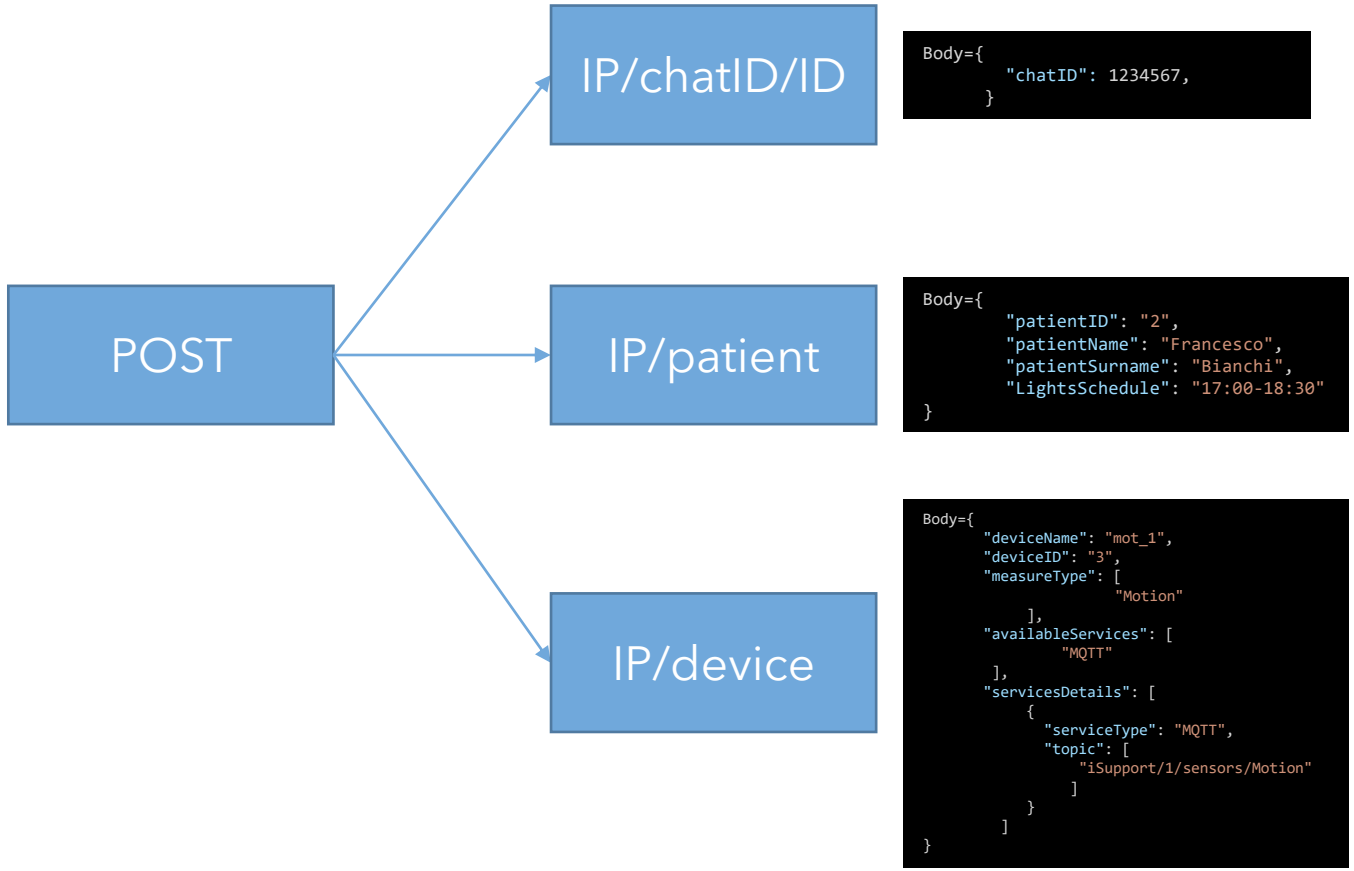
Catalog

- It communicates with all the other actors in the platform exploiting REST communication.
- It has 3 main functions:
  - Retrieve (**GET**)
  - Add (POST)
  - Update (PUT)



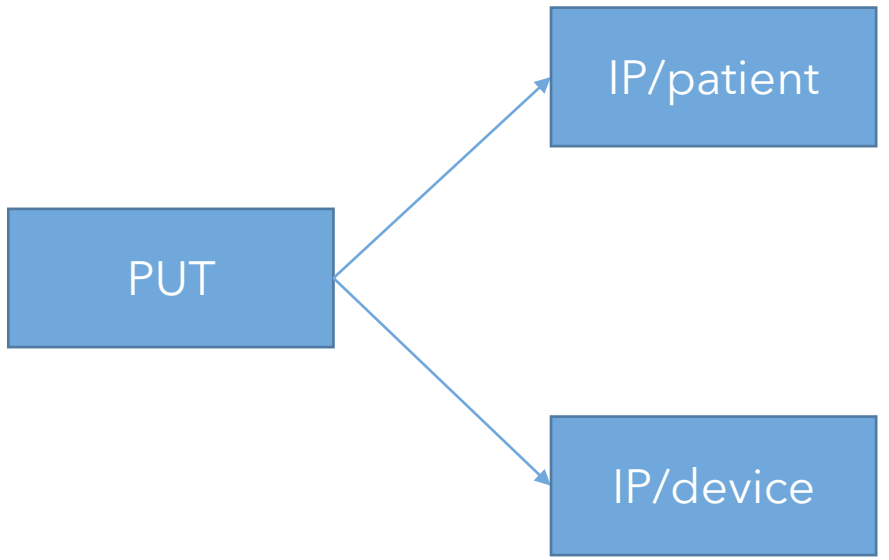
Catalog

- It communicates with all the other actors in the platform exploiting REST communication.
- It has 3 main functions :
  - Retrieve (GET)
  - Add (**POST**)
  - Update (PUT)



Catalog

- It communicates with all the other actors in the platform exploiting REST communication.
- It has 3 main functions:
  - Retrieve (GET)
  - Add (POST)
  - Update (**PUT**)

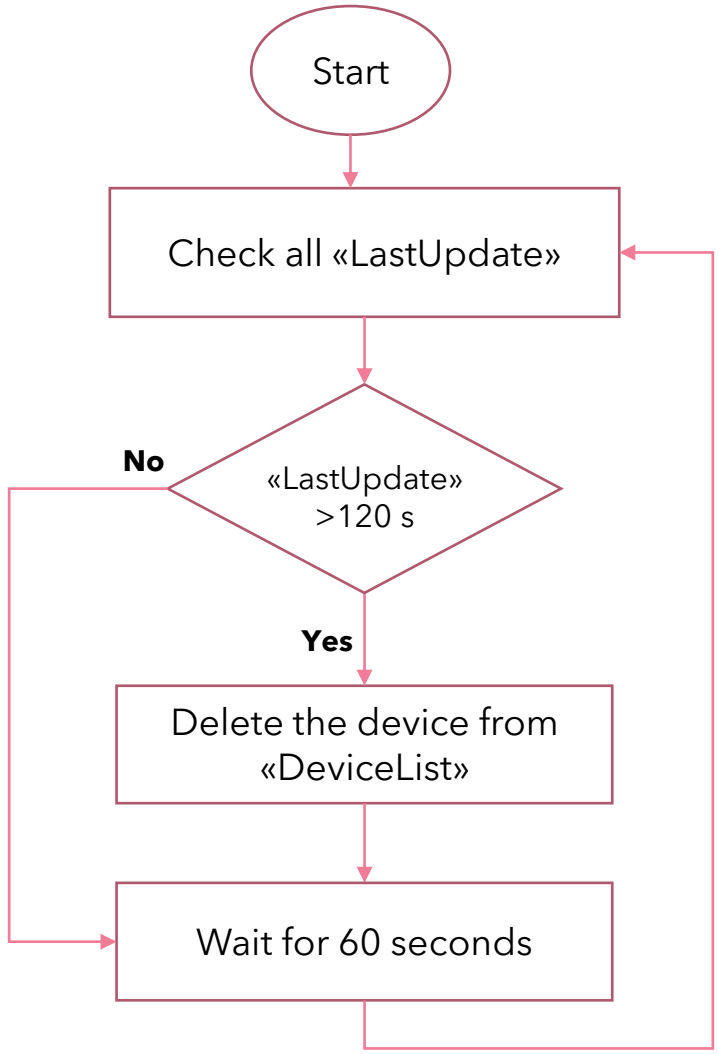



```
Body={
        "patientID": «1",
        "patientName": "Francesco",
        "patientSurname": "Bianchi",
        "LightsSchedule": "17:00-18:30",
        "connectedDevices": [
            {
                "measure": "Temperature",
                "deviceID": "t_1"
            }
        ],
        "telegramIDs": [],
        "thingspeakInfo": {
            "apikeys": [
                "ZQ2AJZABDSC4MXXG",
                "SV1HOET58CE7225M"
            ],
            "channel": 1342741
        },
        "lastUpdate": ""
    }
```

```
Body={
        "deviceName": "smartwatch_1",
        "deviceID": "1",
        "measureType": [
            "HeartRate",
            "Accelerometer"
        ],
        "availableServices": [
            "MQTT"
        ],
        "servicesDetails": [
            {
                "serviceType": "MQTT",
                "topic": [
                    "iSupport/1/sensors/Body"
                ]
            }
        ],
        "lastUpdate": 1623097970.3872042
    }
```

Catalog

- Catalog keeps updated the "deviceList" with the devices correctly connected and removes all the devices that are disconnected.



Device Connector

Devices implemented in the simulation:

- **Temperature and humidity sensors**
- **Morion Sensor**
- Heart rate sensor
- Accelerometer sensor
- Actuator for air Conditionair system
- Smart light bulb actuator

- Temperature and Humidity:

Extraction of a value from uniform distributions that have specific limits that change according to the month in which publication occurs. User sets mode of generation:

- in the suitable range
- out of range

- Motion Sensor:

It is a value (1/0) set by the user

Device Connector

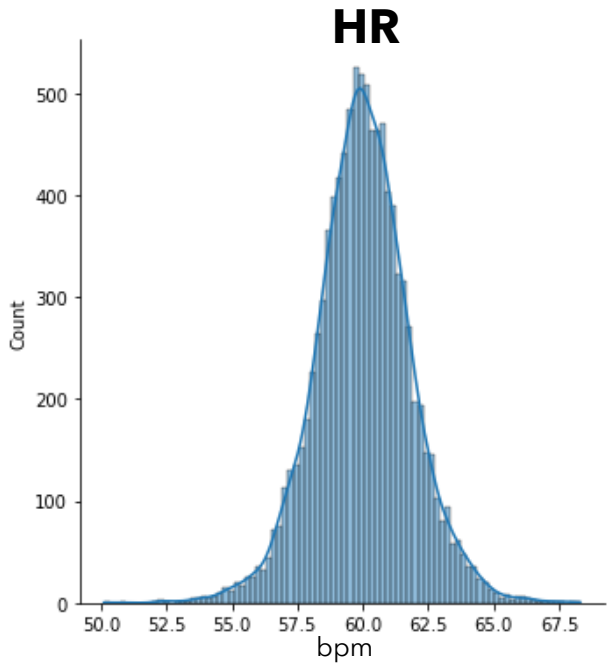
Devices implemented in the simulation:

- Temperature and humidity sensors
- Motion Sensor
- **Heart rate sensor**
- **Accelerometer sensor**
- Actuator for air Conditionair system
- Smart light bulb actuator

-HR and Accelerometer measurments:

Accelerometer measurments are obtained from 2 database: *REST.txt* and *SPORT.txt*

- REST configuration



+

REST.txt

0.06275
0.04957
0.11802
0.28902
0.10596
0.556840…

Device Connector

Devices implemented in the simulation:

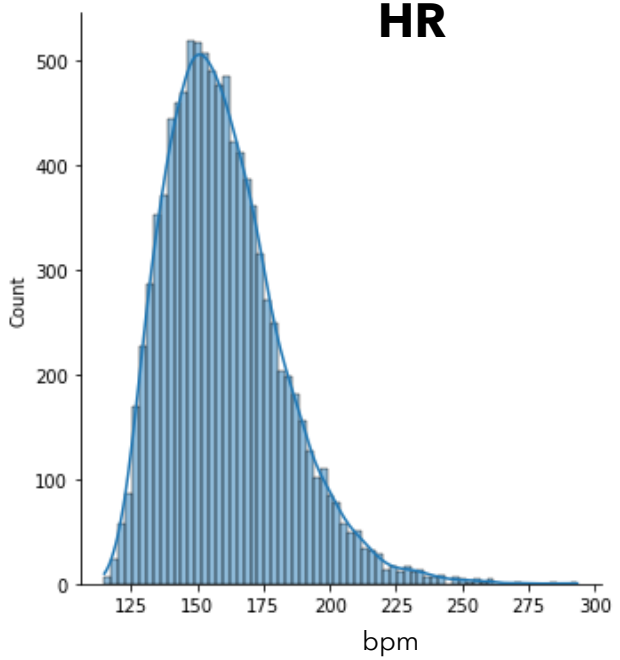
- Temperature and humidity sensors
- Motion Sensor
- **Heart rate sensor**
- **Accelerometer sensor**
- Actuator for air Conditionair system
- Smart light bulb actuator

-HR and Accelerometer measurments:

Accelerometer measurments are obtained from 2 database: *REST.txt* and *SPORT.txt*

- Sport configuration

HR



+

SPORT.txt

6.920512
7.049275
7.176033
7.291157
7.398414
8.597905
8.611896…

Device Connector

Devices implemented in the simulation:

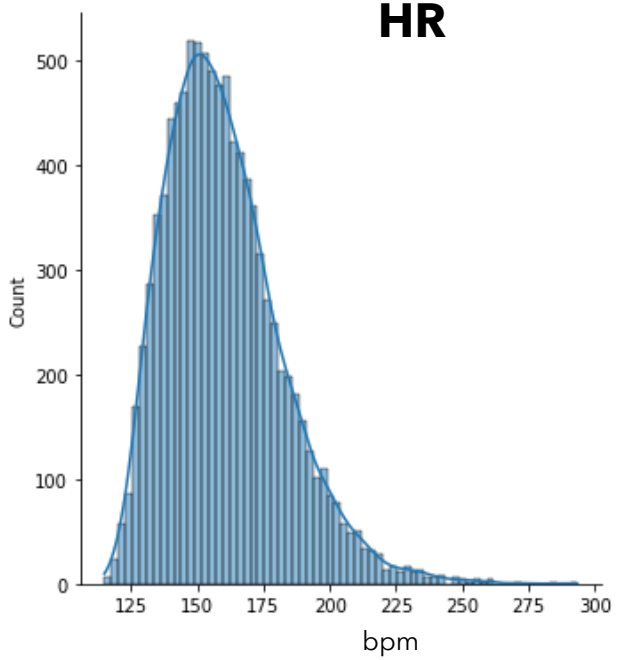
- Temperature and humidity sensors
- Motion Sensor
- **Heart rate sensor**
- **Accelerometer sensor**
- Actuator for air Conditionair system
- Smart light bulb actuator

-HR and Accelerometer measurments:

Accelerometer measurments are obtained from 2 database: *REST.txt* and *SPORT.txt*

- «Danger» configuration

HR



\+

REST.txt

0.06275
0.04957
0.11802
0.28902
0.10596
0.556840…

Device Connector

Devices implemented in the simulation:

- Temperature and humidity sensors
- Presence Sensor
- Heart rate sensor
- Accelerometer sensor
- **Actuator for air Conditionair system**
- **Smart light bulb actuator**

Actuation Commands:

- Switching on the air Conditionair system to maintain optimal temperature and humidity levels.
- Switching on smart lights to enable luminotherapy.

Device Connector

Communication with the other actors is done using **SenML format.**

```
self.__message={
            "bn": 1,
            "e": [{
                        "n": "Heart Rate",
                        "value": 61,
                        "timestamp": "123456",
                        "u": "bpm"
            },
            {
                        "n": "Accelerometer",
                        "value": 0.060369,
                        "timestamp": "123456",
                        "u": "m/s2"
            }
            ]
}
```

Communication with

- Catalog: to register devices or refresh registration with a REST communication.
- ThingSpeak Adaptor: which receives all measurements from the sensors published by Device Connector.
- Home system Control: device connector publishes sensor measurements to this microservice, which sends actuation commands to switch on and off the temperature control system.
- Light Shift: from which device connector receives actuation command to switch on/off smart light bulbs for luminotherapy.

Patient Control

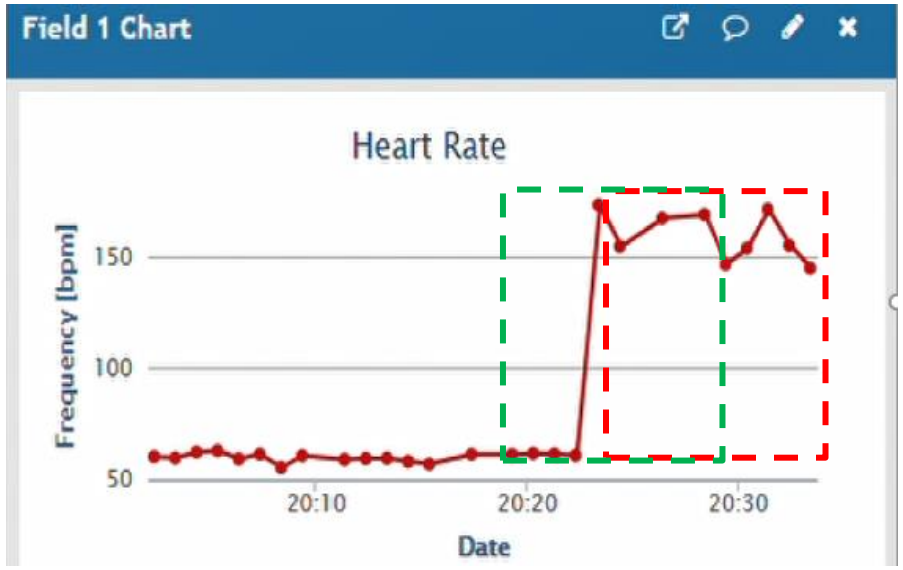
Goal: Monitoring the patient and detecting of panic attack

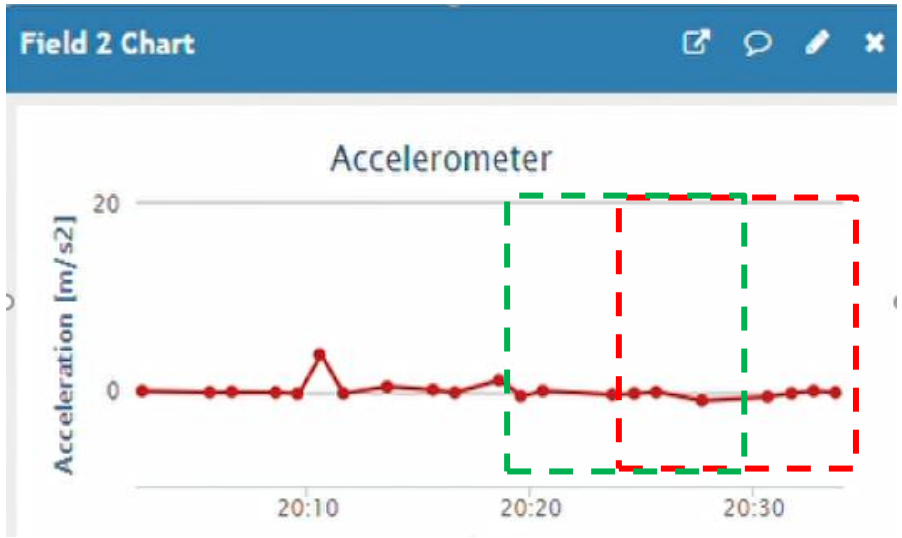
How it works

- Analyze heart rate and acceleration data

Communication with

- ThingSpeak: for getting measurement of 10 minutes windows.

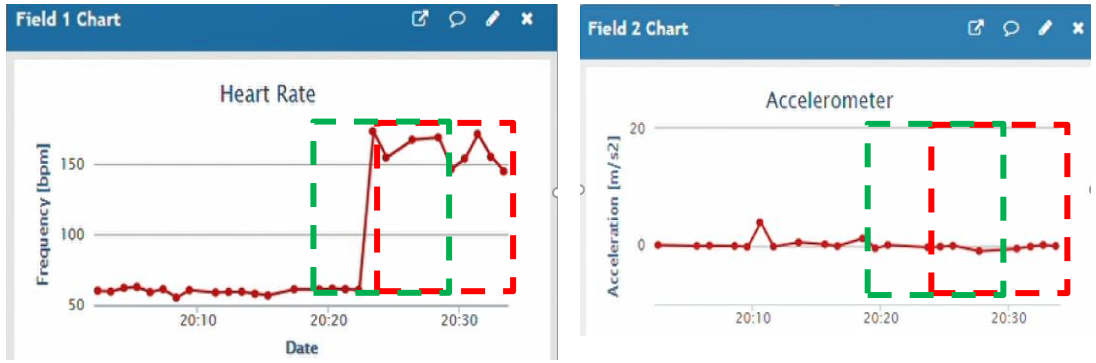




Patient Control

How it works

- Analyze heart rate and acceleration data



- Communicate an alert in case of panic attack to the doctor and caregiver

Communication with

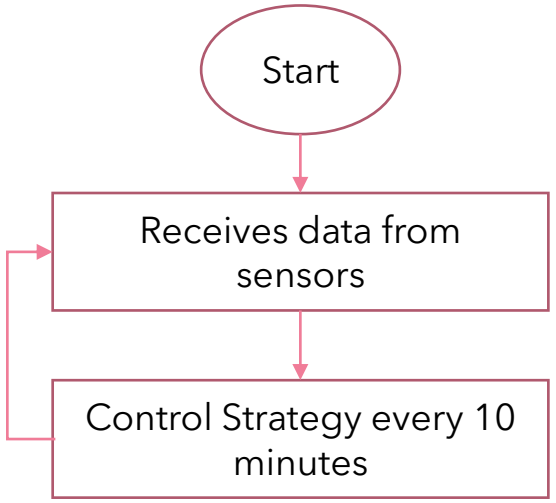
- ThingSpeak: for getting measurement of 10 minutes windows.
- ThingSpeak: for recording the panic attack event
- Telegram bot: for communicating the panic attack event

Home system Control

Goal: Monitoring home for having a comfortable enviroment

How it works

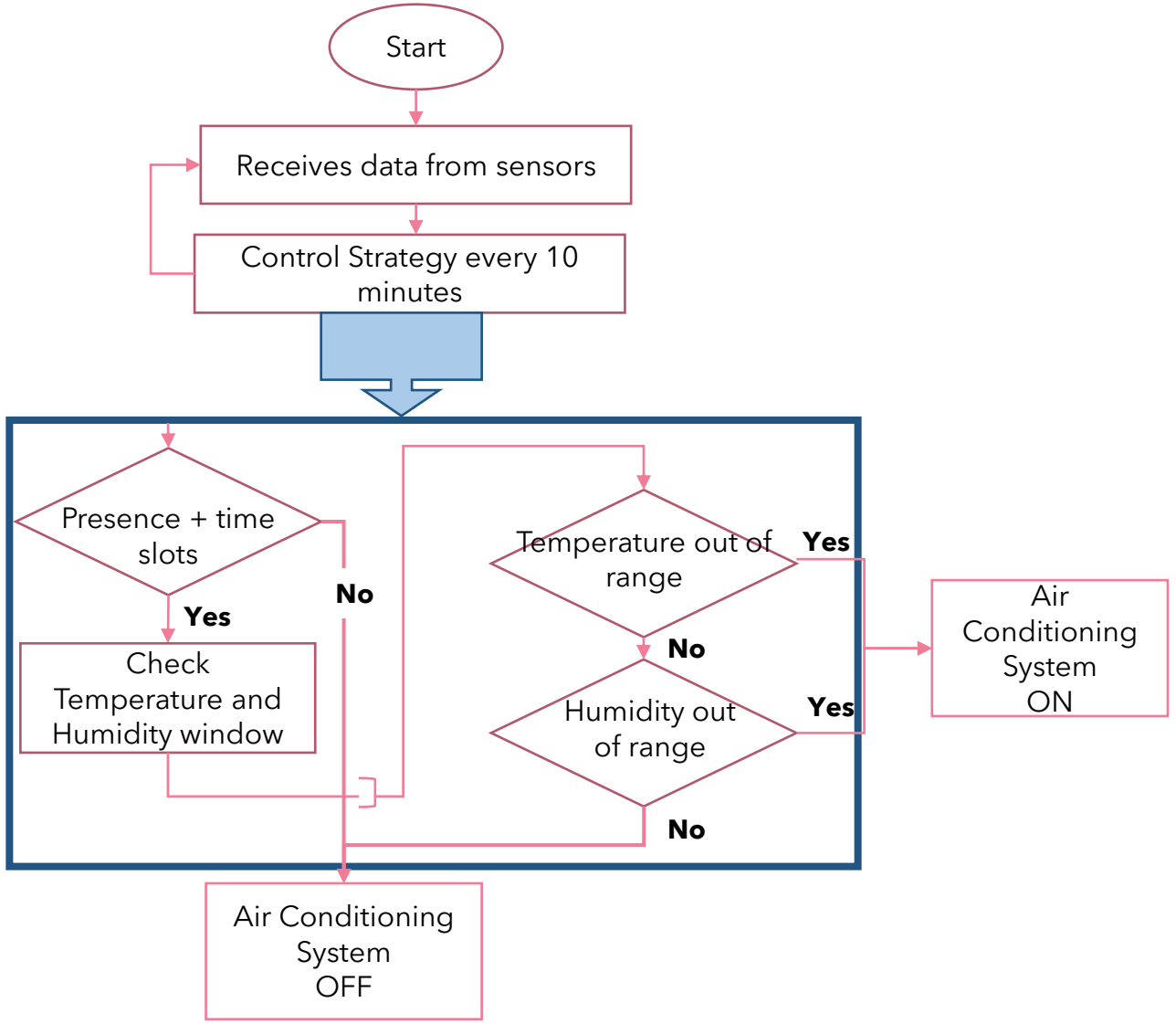
- Manages temperature and humidity inside the house
- Verifies patient presence in the home
- Avoid wasting energy by establishing activation time slots



Home system Control

How it works

- Manages temperature and humidity inside the house
- Verifies patient presence in the home
- Avoid wasting energy by establishing activation time slots



Home system Control

How it works

- Manages temperature and humidity inside the house
- Verifies patient presence in the home
- Avoid wasting energy by establishing activation time slots

Communication with

- Device connector:
  - Continuously receives data from Temperature and Humidity sensors
  - Sends actuation commands if it is identified a condition to switch ON or OFF the Air Conditioning System

Light Shift

How it works

- Manages the lighting system in relation to the physician's established schedule for luminotherapy

Communication with

- Catalog:
  - to obtain the times in which to activate the luminotherapy
- Device Connectors:
  - to send actuations commands to turn the lights ON or OFF

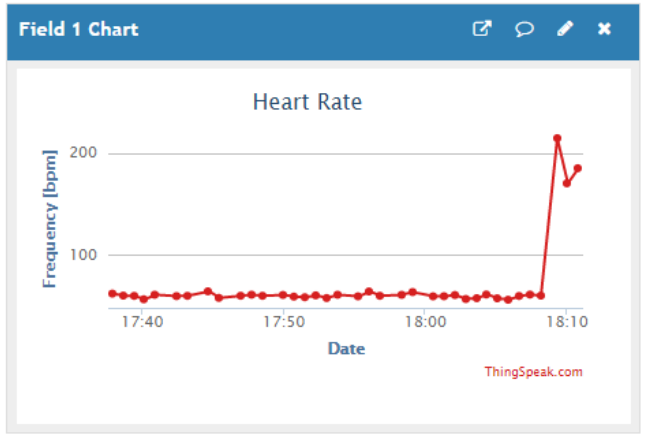
ThingSpeak

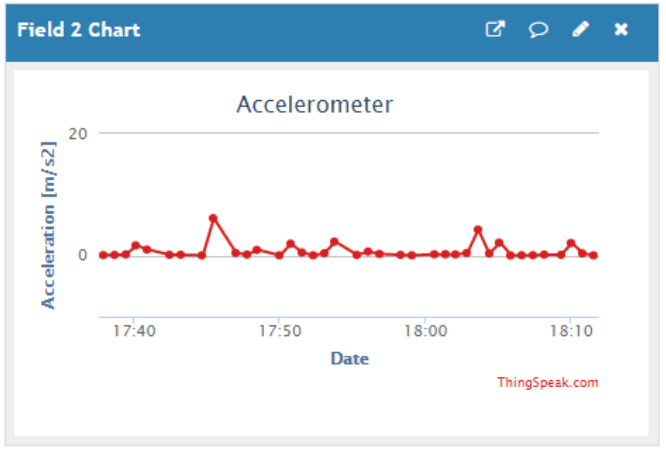
It’s a platform used like a database in which there are all measurement got from different sensors implemented on iSupport platform

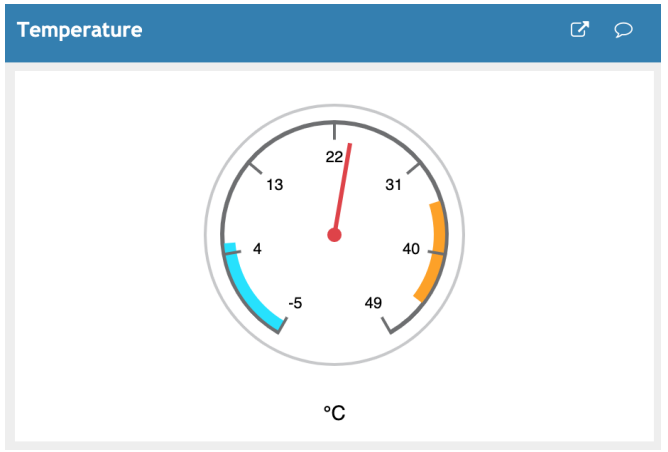
It uses an adaptor *ThingSpeak Adaptor* which:

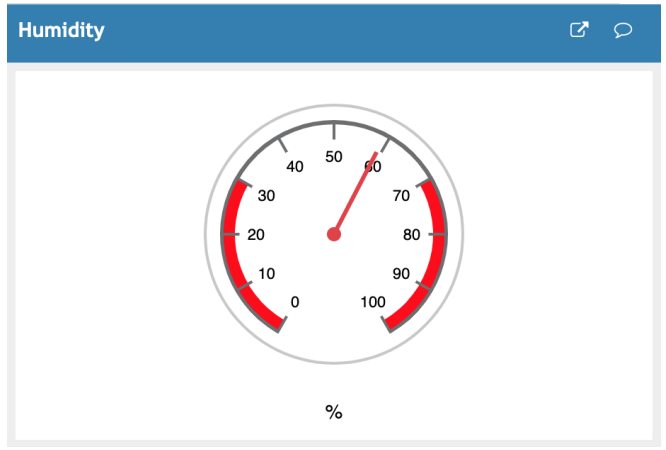
- Allows you to implement communication via MQTT.
- Increases the scalability of the platform
- Increases interoperability between components.

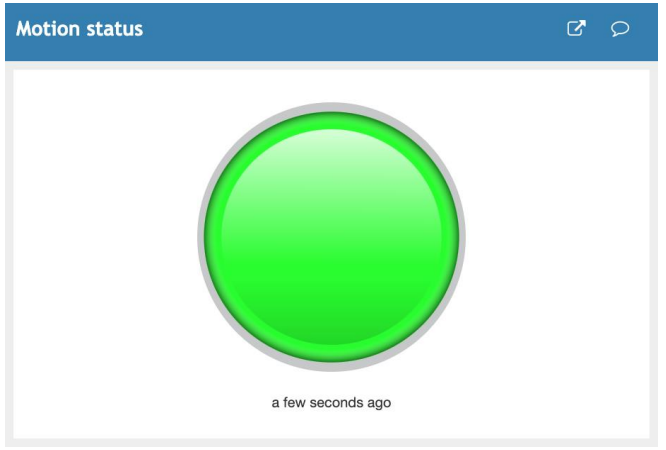
ThingSpeak third part Application User Interface











ThingSpeak

It's a platform used like a database in which there are all measurement got from different sensors implemented on iSupport platform

It uses an adaptor *ThingSpeak Adaptor* which:

- Allows you to implement communication via MQTT.
- Increases the scalability of the platform
- Increases interoperability between components.

Communication with

- Device Connector: continous updating of measurements
- Data Analysis: sending measurements received from the device connector.
- Patient Control: sending measurements related to specific observation windows.

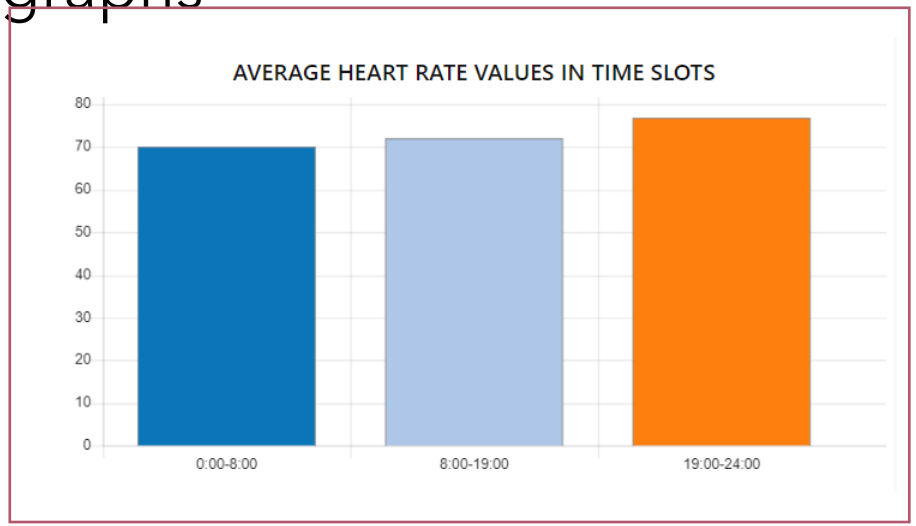
Data Analysis

How it works

- Process patient measurements to get weekly or monthly statistics:
  - **Weekly report about average heart rate in 3 time bands.**
  - Weekly report about patient's activity status (percentage of low, medium, high intensity obtained as comparing rate and acceleration measurements
  - Number of monthly panic attacks
  - Weekly report about how much time patient spends in the bedroom (level of sedentary lifestyle)

Communication with

- ThingSpeak Adaptor: to get measurements
- Node-RED: to create statistical graphs



Node-RED interface

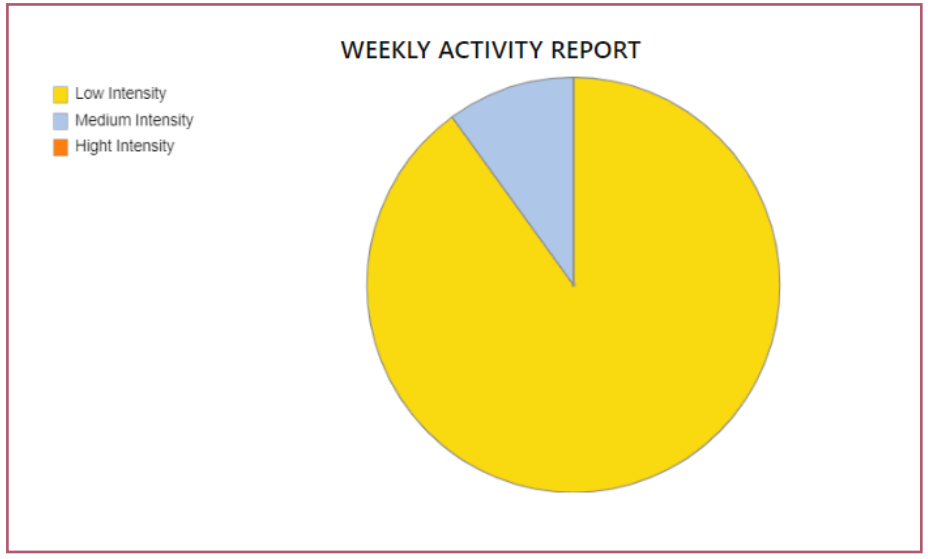
Data Analysis

How it works

- Process patient measurements to get weekly or monthly statistics:
  - Weekly report about average frequency values in 3 time bands.
  - **Weekly report about patient's activity status (percentage of low, medium, high intensity obtained as comparing rate and acceleration measurements.**
  - Number of monthly panic attacks.
  - Weekly report about how much time patient spends in the bedroom (level of sedentary lifestyle).

Communication with

- ThingSpeak Adaptor: to get measurements
- Node-RED: to create statistical graphs



Node-RED interface

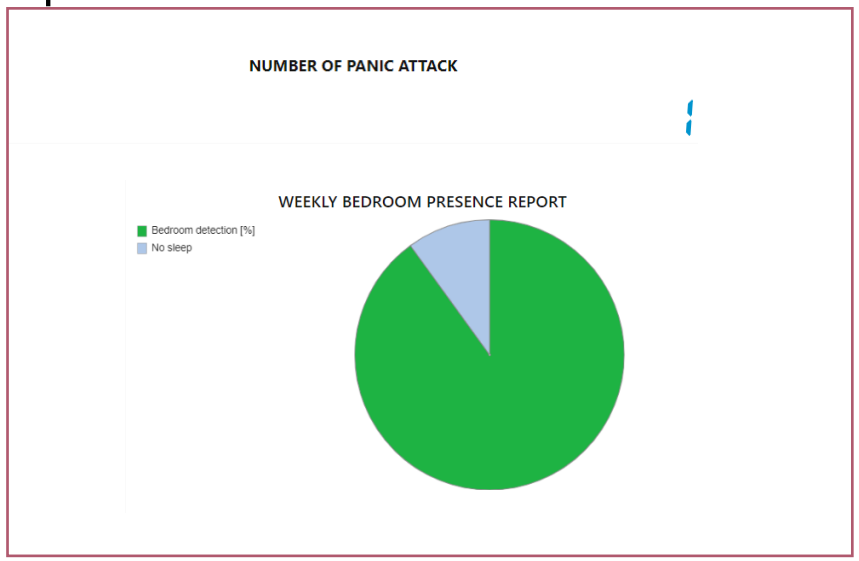
Data Analysis

How it works

- Process patient measurements to get weekly or monthly statistics:
  - Weekly report about average frequency values in 3 time bands.
  - Weekly report about patient's activity status (percentage of low, medium, high intensity obtained as comparing rate and acceleration measurements.
  - **Number of monthly panic attacks.**
  - **Weekly report about how much time patient spends in the bedroom (level of sedentary lifestyle).**

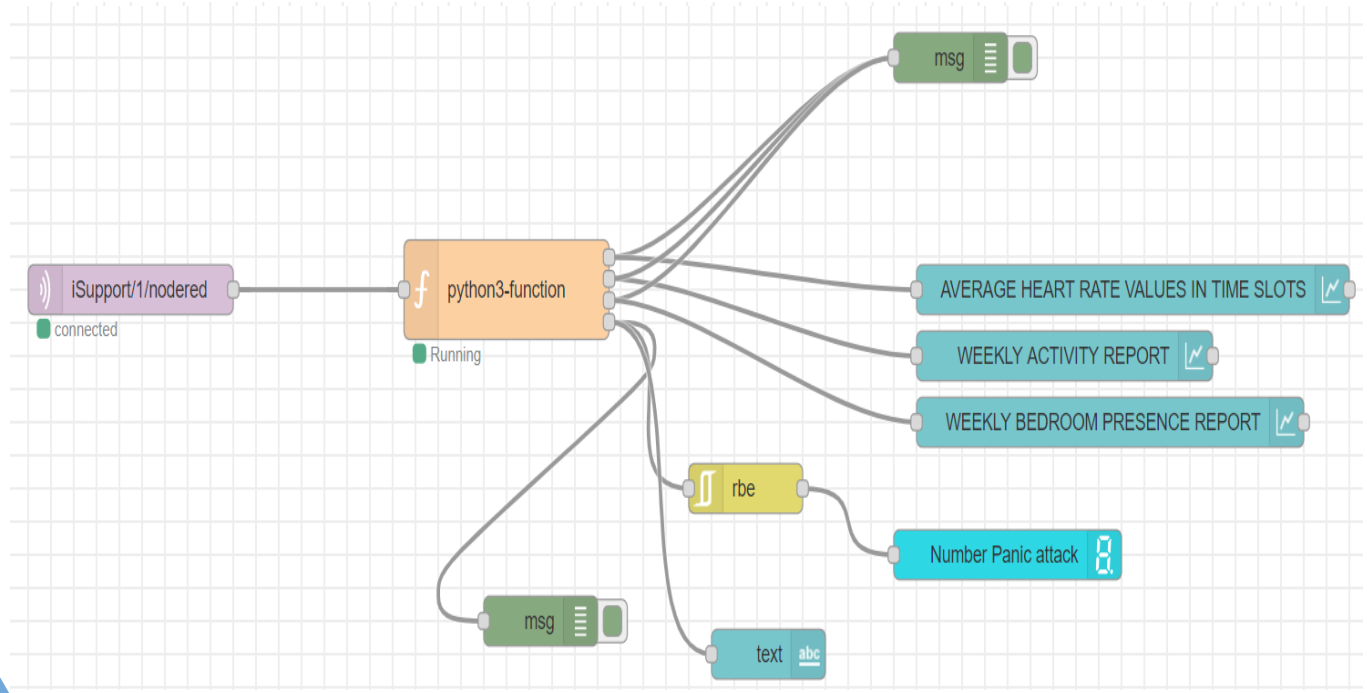
Communication with

- ThingSpeak Adaptor: to get measurements
- Node-RED: to create statistical graphs



Node-RED interface

Node-RED



How it works

- Nodes:
  - MQTT communication node
  - *Python function*
  - Two blocks to visualize msg.payload in debug *display*
  - A *rbe* block
  - A *digital display* and three *chart*

Telegram Bot

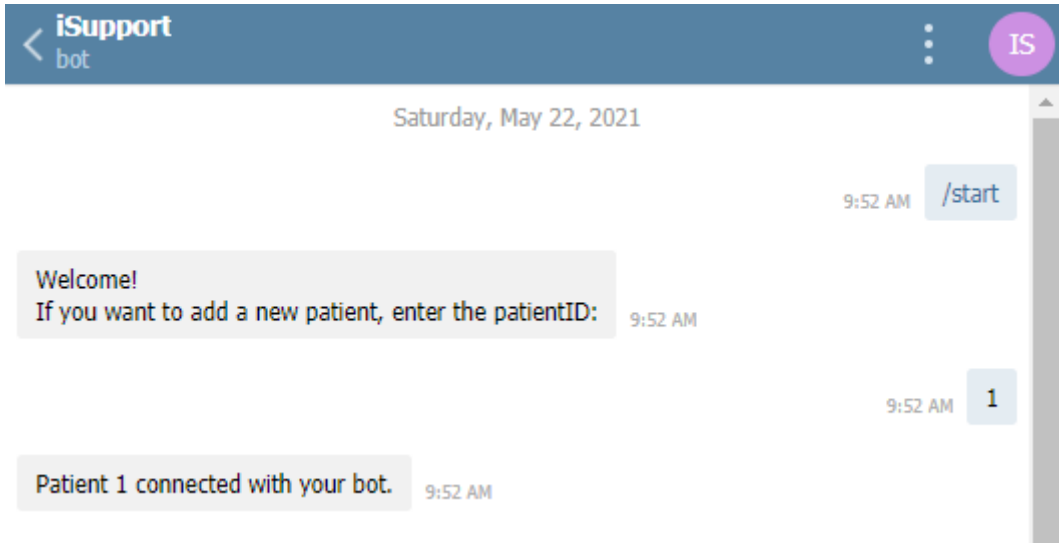
- Emergency management
- Allow the doctor and the caregiver to always be updated on the Heart Rate, Movement and Recurrence trends of panic attacks.

- ‘/start’ function:
  - Telegram user registration on platform
- ‘/status’ function:
  - Visualization of ThingSpeak graphs about vital signs
- Emergency management:
  - Getting alert message from the Patient Control microservice in case of Panic Attack.

Telegram Bot

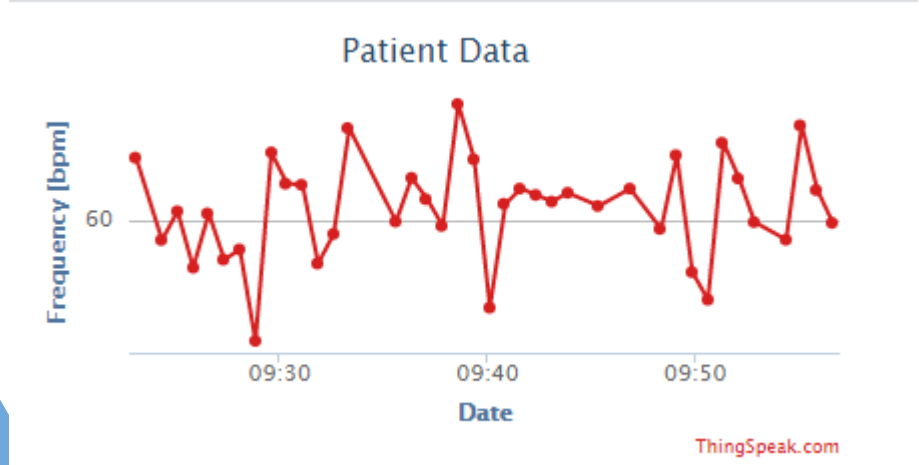
- Emergency management
- Allow the doctor and the caregiver to always be updated on the Heart Rate, Movement and Recurrence trends of panic attacks.

- ‘/start’ function:
  - Telegram user registration on platform

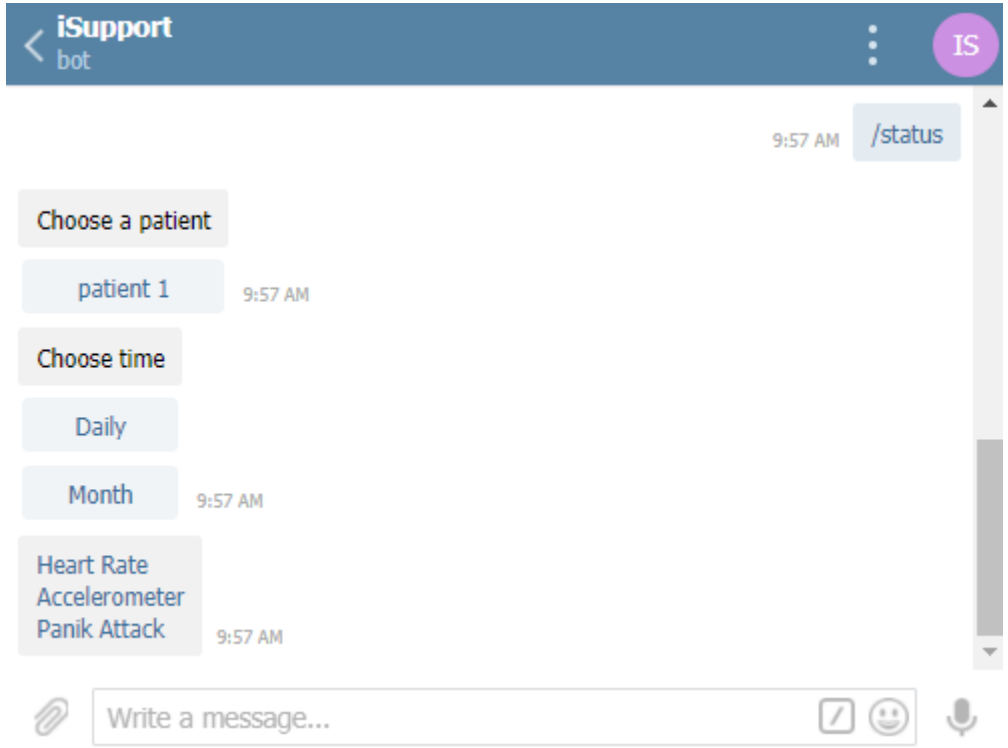


Telegram Bot

- Emergency management
- Allow the doctor and the caregiver to always be updated on the Heart Rate, Movement and Recurrence trends of panic attacks.



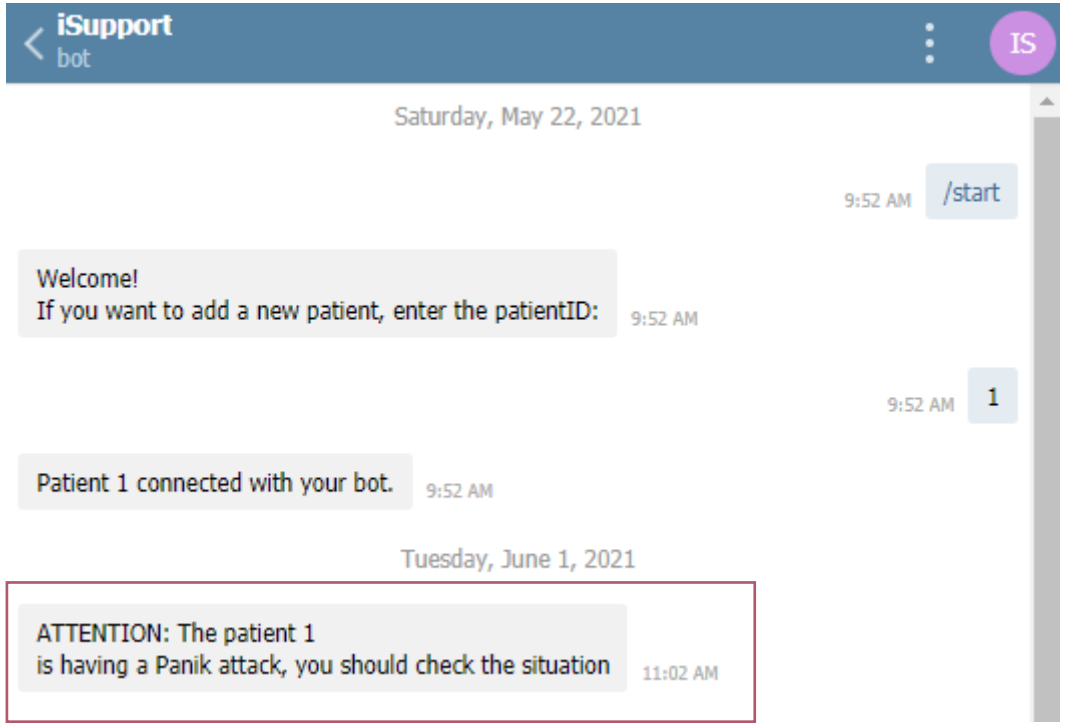
- ‘/status’ function:
  - Visualization of ThingSpeak graphs about vital signs



Telegram Bot

- Emergency management
- Allow the doctor and the caregiver to always be updated on the Heart Rate, Movement and Recurrence trends of panic attacks.

- Emergency management:
  - Getting alert message from the Patient Control microservice in case of Panic Attack.



Telegram Bot

- Emergency management
- Allow the doctor and the caregiver to always be updated on the Heart Rate, Movement and Recurrence trends of panic attacks.

Communication with

- Patient Control: which publishes a message to the telegram bot if it detects a panic attack
- ThingSpeak: required for viewing graphs
- Catalog: retrieve chatId and token

Thanks for your attention

Team 3:
Burdisso Sara s275424
Centonze Ilenia s277389
Ferrara Mauro s276932
Pasotti Andrea s275100