

EYE App Configuration

EYE App Configuration

Devices work constantly and are ready to perform out of the box. Default basic Sensor settings are set to:

- Transmitting at 2 dBm power.
- Data advertising at 3 second intervals.
- Eddystone protocol

If you would like to change these settings you will need to:

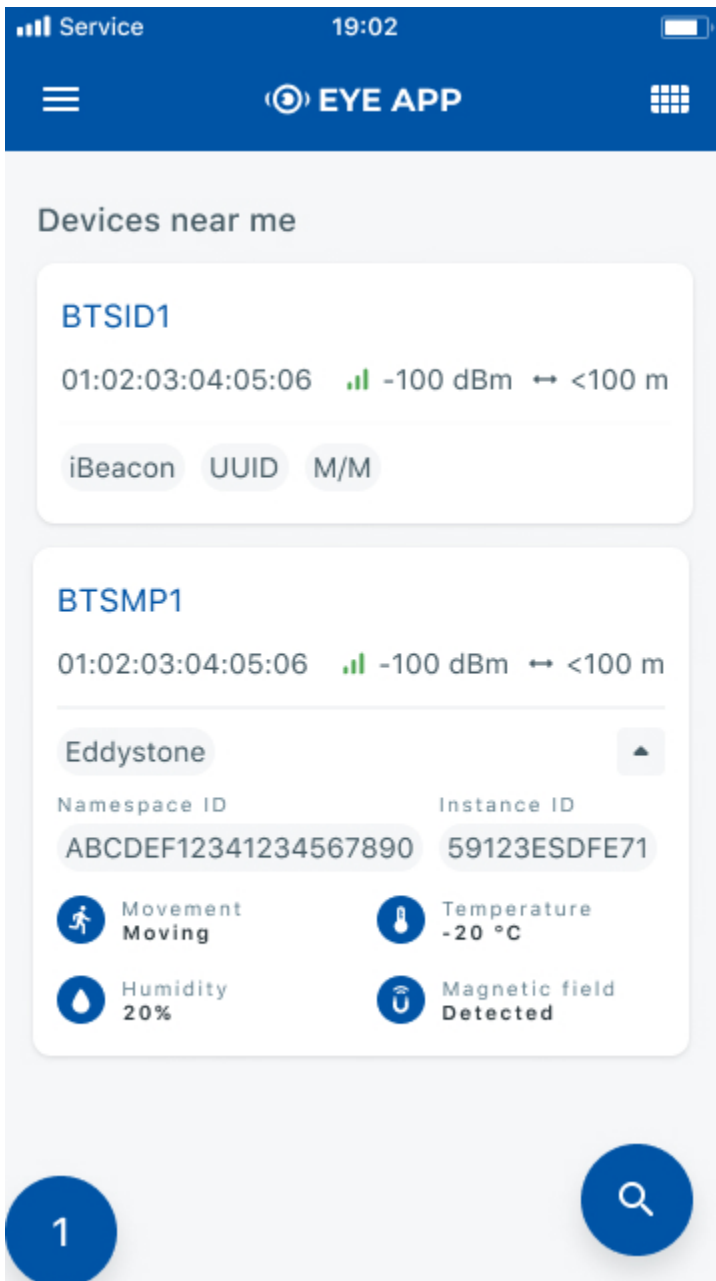
- 1) Download and install EYE APP - Teltonika application to change sensor settings.






*iOS doesn't show devices with iBeacon protocol

Eye App Overview


With the application you will be able to scan for visible Teltonika EYE devices, check their statuses or connect for configuration or firmware update.



✘ **Devices in view:** In this window you will see all visible devices. You have options to see devices in Short list  or in default list. When looking at devices in default list you will be able to open Detailed view  of devices and check transmitted data statuses. Additionally if you are looking for specific devices you will be able to use Search  function to filter search options. When in this window select a device of your choice to connect and after passing pin code (default pin code is 123456) you will go to device overview window

✘ **Overview window:** In overview window you can see device details, check firmware version and update if available , go to device configuration settings. If you select to Configure device new window will open with Basic and Advanced settings.

✘ **Configure window:** In this window you can check and change device configuration settings. In main tab Basic Settings you can change main settings. Change Device name, Power signal strength, Advertising interval and Packet transmission type*. For more settings go to Advanced settings tab to enable various events.

 **Advanced configuration window:** Beacon information (iBeacon ID*, Eddystone ID).
UUID/MINOR/MAJOR parameter can be changed for iBeacon ID*. Namespace ID/Instance ID can be changed for Eddystone ID



***iOS doesn't show devices with iBeacon protocol**