Quick Start Guide

STM32Cube function pack for STEVAL-STWINKT1B evaluation kit for predictive maintenance application based on artificial intelligence (AI)

(FP-AI-PREDMNT2)

Version 1.2 (Jun 10, 2022)

Agenda

1 Hardware and Software overview

2 Setup & Demo Examples

3 Documents & Related Resources

4 STM32 Open Development Environment: Overview

1- Hardware and Software overview

STWIN development kit - STEVAL-STWINKT1B

STWIN - SensorTile Wireless Industrial Node

The STWIN (STEVAL-STWINKT1B) is a development kit and reference design that simplifies prototyping and testing of advanced industrial IoT applications such as condition monitoring and predictive maintenance. The kit supports BLE wireless connectivity through an on-board module and Wi-Fi connectivity through a special plugin expansion board (STEVAL-STWINWFV1), wired RS485 and USB OTG connectivity.

Key Features

- Multi-sensing wireless platform implementing vibration monitoring and ultrasound detection

- Updated version of STEVAL-STWINKT1, now including STSAFE-A110 populated, BlueNRG-M2S module and IMP23ABSU MEMS microphone

- Ultra-low-power ARM Cortex-M4 MCU at 120 MHz with FPU, 2048 kbytes Flash memory (STM32L4R9)

- Micro SD Card slot for standalone data logging applications

- Option to implement Authentication and Brand protection secure solution with STSAFE-A110

- Wide range of industrial IoT sensors: ultra-wide bandwidth (up to 6 kHz), low-noise, 3-axis digital vibration sensor (IIS3DWB), 3D accelerometer + 3D Gyro iNEMO inertial measurement unit (ISM330DHCX) with machine learning core, ultra-low-power high performance MEMS motion sensor (IIS2DH), ultra-low-power 3-axis magnetometer (IIS2MDC), digital absolute pressure sensor (LPS22HH), relative humidity and temperature sensor (HTS221), low-voltage digital local temperature sensor (STTS751), industrial grade digital MEMS microphone (IMP34DT05), analog MEMS microphone with frequency response up to 80 kHz (IMP23ABSU)

- Modular architecture, expandable via on-board connectors: STMOD+ and 40-pin flex general purpose expansions, 12-pin male plug for connectivity expansions, 12-pin female plug for sensing expansions

- Other kit components: Li-Po battery 480 mAh, STLINK-V3MINI debugger with programming cable, Plastic box

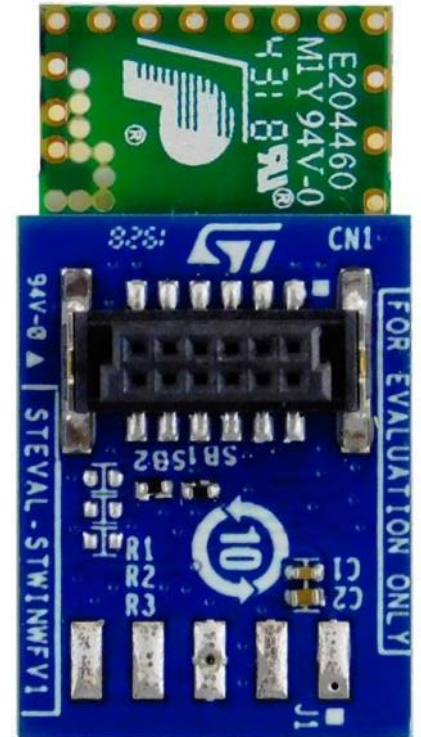Latest info available at
www.st.com/stwin

Wi-Fi expansion for the SensorTile Wireless Industrial Node (STWIN) kit

The STEVAL-STWINWFV1 expansion board adds 2.4 GHz Wi-Fi connectivity to the STEVAL-STWINKT1B.

Application examples that take advantage of this expansion board can be found also in STSW-STWINKT01 firmware package as well as in the FP-IND-PREDMNT1 and FP-AI-PREDMNT2 Predictive Maintenance Function Pack.

Key Features

- Wi-Fi adapter for STEVAL-STWINKT1B
- Plugs into STWIN core system board through dedicated 12-pin connector
- Single 3.3 V power supply input
- ISM43362-M3G-L44-E Wi-Fi module:
    - 802.11 b/g/n Compatible based on Broadcom MAC/Baseband/Radio device
    - Fully contained TCP/IP stack
    - Host interface: SPI up to 25 MHz
    - Network features ICMP (Ping), ARP, DHCP,TCP, UDP
    - Low power operation (3.3 V supply) with built-in low power modes
    - Secure Wi-Fi authentication WEP-128, WPA-PSK (TKIP), WPA2-PSK
    - CE, FCC and IC certified
- RoHS and China RoHS compliant
- WEEE compliant

Latest info available at www.st.com
STEVAL-STWINWFV1

FP-AI-PREDMNT2 Software Description

FP-AI-PREDMNT2 is an **STM32Cube** function pack that programs the **STWIN** as an IoT Edge node, connected to the cloud, able to acquire sensor data, process them and send the results to the **DSH-PREDMNT** cloud dashboard. It includes dedicated algorithms for advanced time and frequency domain signal processing and analysis of 3D digital accelerometers with flat bandwidth up to 6 kHz.

Key features

- Compatible with **NanoEdge™ AI Studio** solution, to enable AI-based solution (NanoEdge™ AI library generation is out of the scope of this function pack and must be generated using NanoEdge™ AI Studio)

- Generic FFT library middleware to enable frequency domain analysis for any kind of sensor through Fast Fourier Transform (with programmable size, overlapping and windowing)

- Motion TD library middleware for vibration analysis in time domain (speed RMS and acceleration peak)

- Configurable alarm and warning thresholds for key parameters

- Compatible with **STBLESensor** application for Android (from v4.12.0) and iOS (from v4.15.0), to perform Wi-Fi configuration and secure certificate provisioning

- Compatible with DSH-PREDMNT web-based predictive maintenance dashboard for monitoring sensor data and device status

- Easy portability across different MCU families, thanks to STM32Cube

- Firmware modular example based on **eLooM** (embedded Light object-oriented fraMework for STM32) to enable code re-usability at application level

- Free, user-friendly license terms

FP-AI-PREDMNT2
Software Overview

Overall Software Architecture
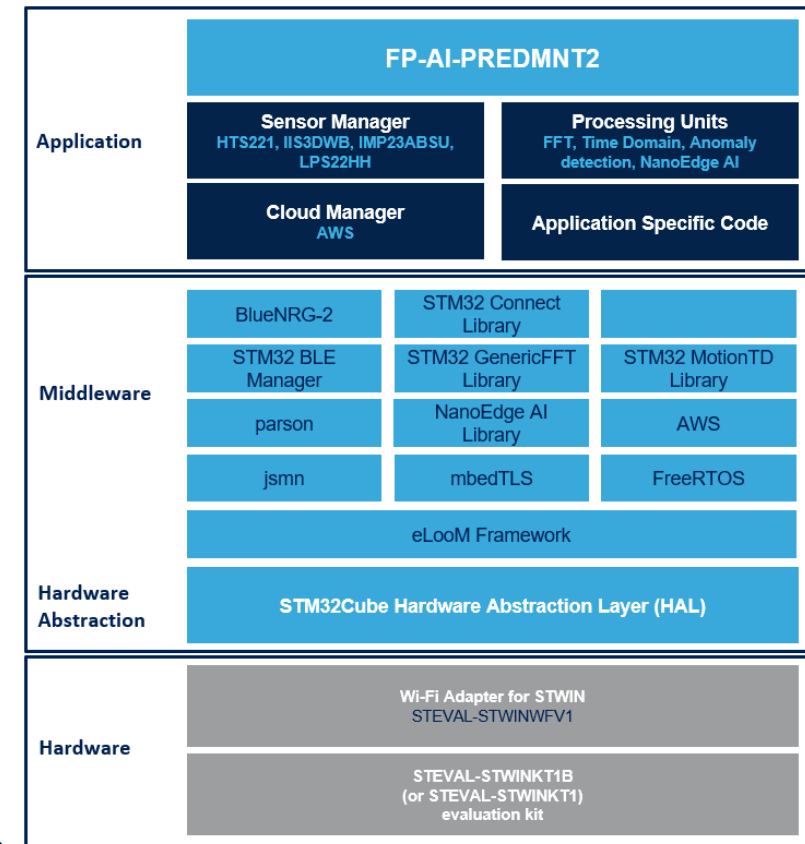
2- Setup & Demo Examples

Setup & Demo Examples

- 1x STEVAL-STWINKT1B development board
- 1x STEVAL-STWINWFV1
- Laptop/PC with Windows 7, 8 or 10
- 2x microUSB cables
- 1 smartphone with ST BLESensor App
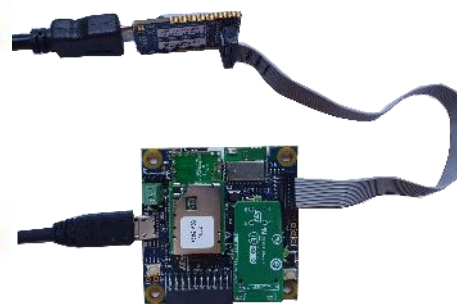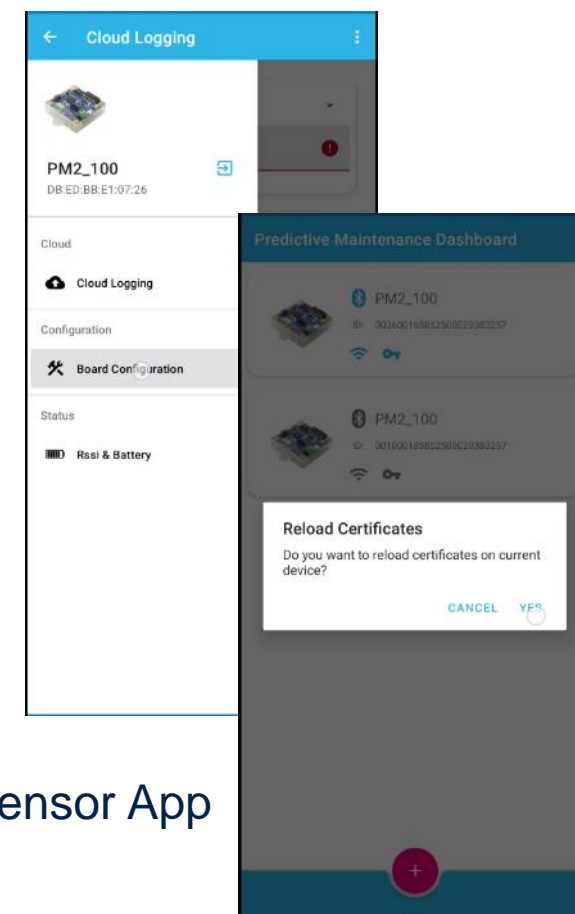- Wi-Fi Router or access to a Wi-Fi network

STEVAL-STWINWFV1
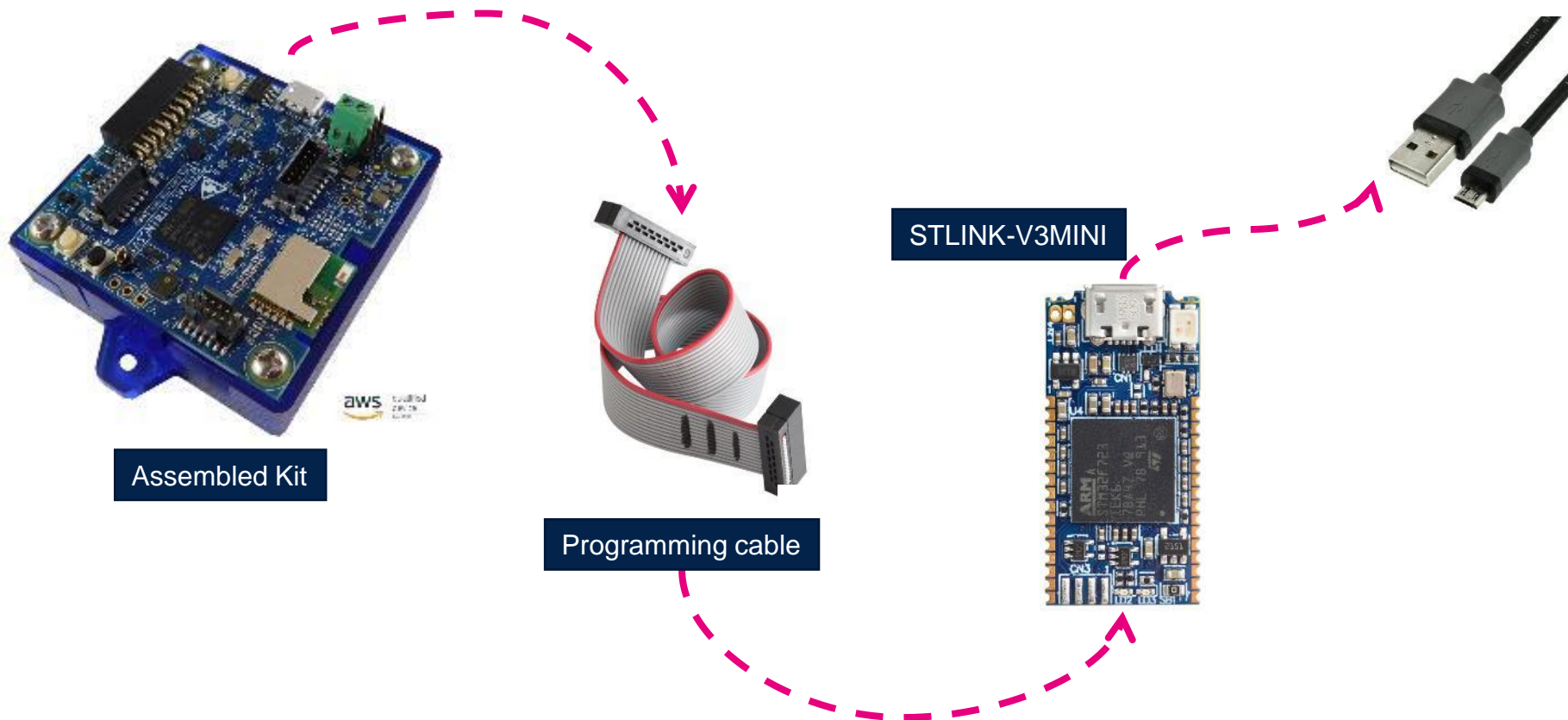
STEVAL-STWINKT1B

MicroUSB Cable

ST BLESensor App

8

- How to re-program/debug the STWIN

Assembled Kit

Programming cable

STLINK-V3MINI

- **STM32CubeProgrammer Software**
  - Download and install STM32CubeProgrammer

- **FP-AI-PREDMNT2**
  - Copy the .zip file content into a folder on your PC. The package will contain source code example (Keil, IAR, STM32CubeIDE) based on **STEVAL-STWINKT1B**

- **ST BLE Sensor** Application for Android (from v4.12.0) / iOS (from v4.15.0) to download from Google Store / App Store

- **ST DSH-PREDMNT** Account

FP-AI-PREDMNT2 **is not** the default firmware on STEVAL-STWINKT1B.
To update the firmware, please follow the instructions available in slide 11

FP-AI-PREDMNT2 **is not** the default firmware on the STEVAL-STWINKT1B, so it needs to be downloaded on the board by the user.

The easiest way is to use the **pre-compiled binary** provided in the package in the folder Projects\STM32L4R9ZI-STWIN\Applications\PREDMNT2\Binary.

To update the firmware:

- Connect the STWIN core system board to the STLINK-V3MINI programmer.

- Connect both the boards to a PC using micro USB cables.

- Open STM32CubeProgrammer, select the proper binary file and download the firmware.

For further details, see **UM2937**
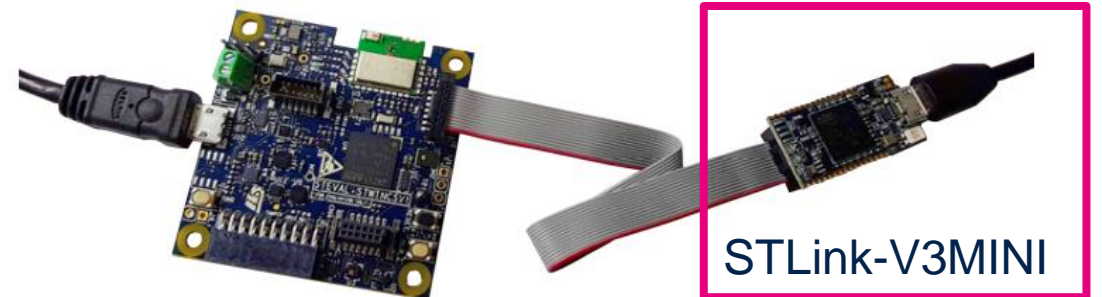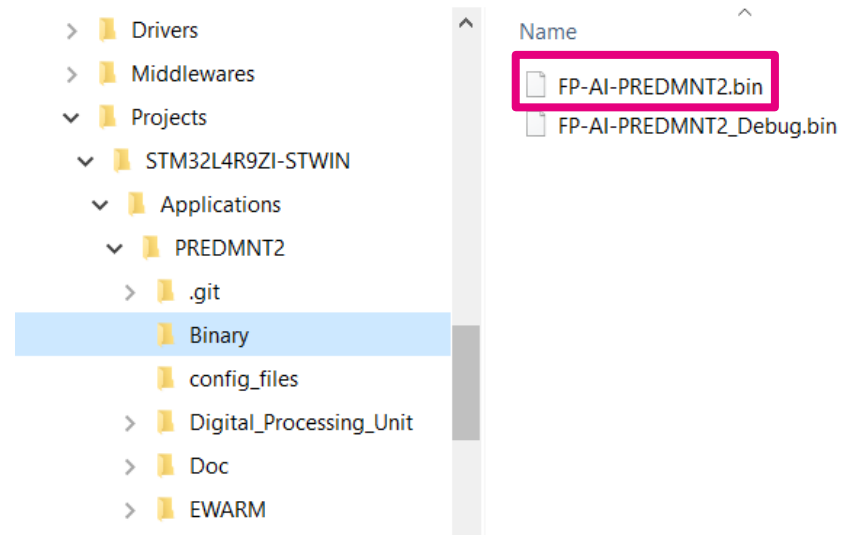
STLink-V3MINI

1 www.st.com/stm32ode-fp

2

3 Download & unpack
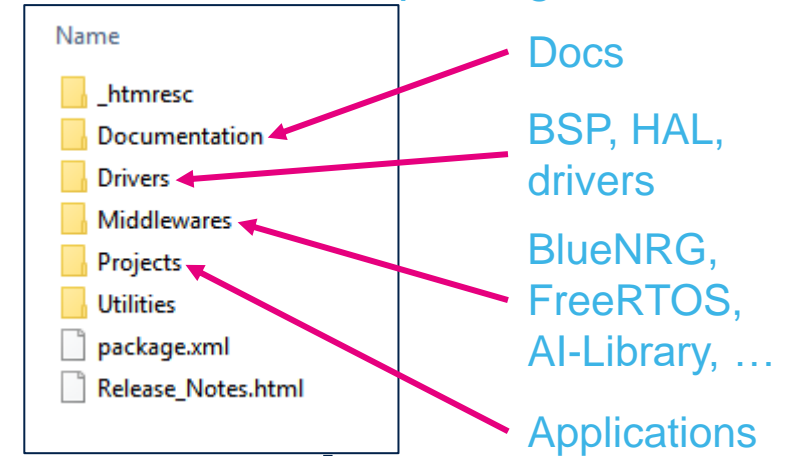
Select Function Pack: FP-AI-PREDMNT2

FP-AI-PREDMNT2 package structure

Name
_htmresc
Documentation
Drivers
Middlewares
Projects
Utilities
package.xml
Release_Notes.html

Docs

BSP, HAL, drivers

BlueNRG, FreeRTOS, AI-Library, …
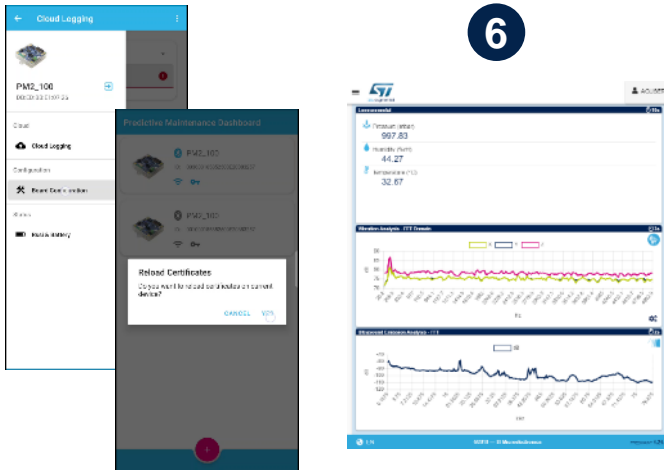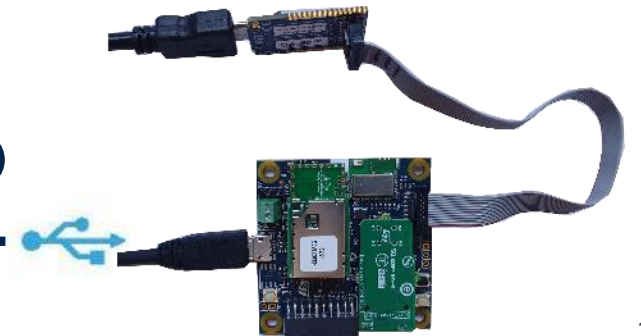
Applications

4

Configure the board through ST BLESensor app and monitor the application through the ST DSH-PREDMNT dashboard

6

Projects\STM32L4R9ZI-STWIN\Applications\PREDMNT2\

KEIL Tools by ARM

IAR SYSTEMS

STM32 CubeIDE

5

12

STWIN Power ON/OFF

- Battery only (no USB cable):
  - Power ON
    - Long-press the PWR button until the red led turns off (~1 sec)
  - Power OFF
    - Press the PWR button

- Plugged mode (USB cable)
  - Power ON
    - When USB is plugged-in, the STWIN is always on. It doesn't matter if the battery is present or not
  - Power OFF
    - Unplug the cable and, if the battery is connected, press the PWR button.

PWR

2.1- Setup wi-fi and security certificates through ST BLESensor app
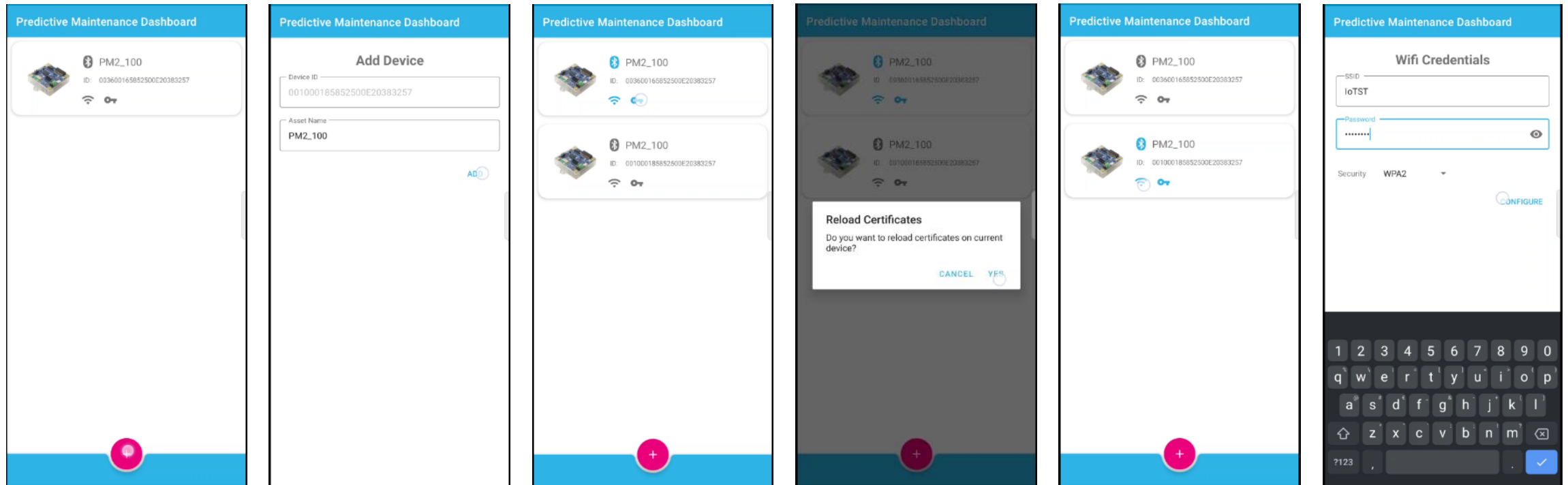
Wi-fi configuration and certificate request

- Once the firmware has been loaded, press the [**Reset**] black button on the STWIN core system. Then, open **STBLESensor** app and connect to PM2_xyz device (where xyz describes the firmware version).

- Open the menu, select [**Board configuration**] and open [**Certificate Request**].

- Select [**Predictive Maintenance**] and sign in with your MyST credentials.
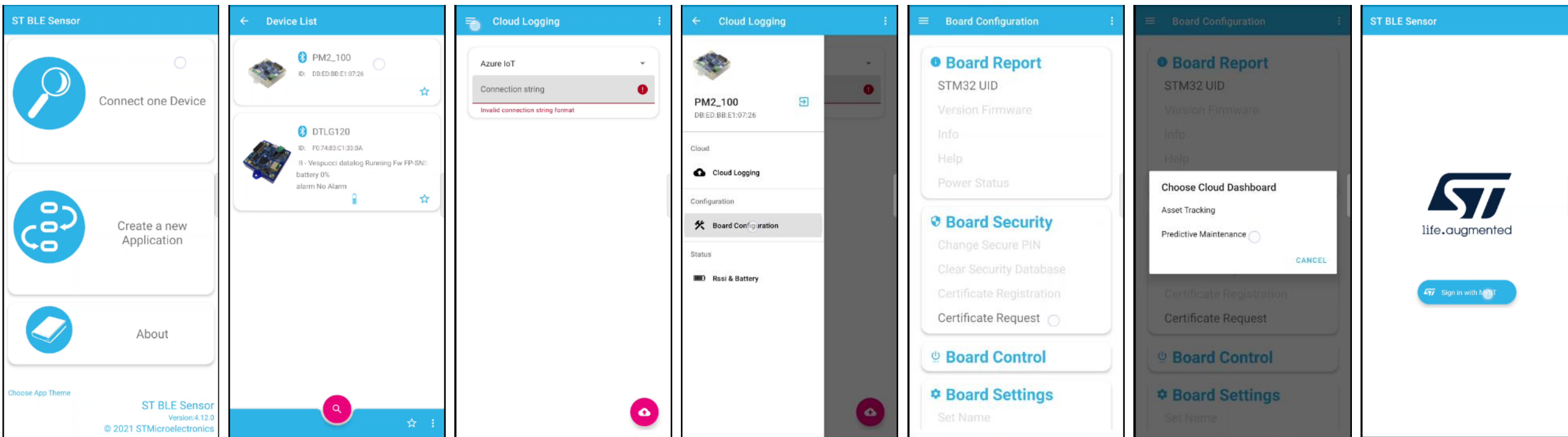
Wi-fi configuration and certificate request
Android v4.12 and following – iOS v4.15 and following

- Ask for a new certificate by tapping the + symbol placed at the bottom of the screen or retrieve the ones already associated to your myST account. The certificates valid for your STWIN are highlighted in blue

- Tap on the key to download the security certificates into the STWIN and tap on the Wi-Fi symbol to update Wi-Fi credentials.

Once the procedure ends correctly, the orange LED switches on.
Now the STWIN is connected to the AWS PREDMNT dashboard and is ready to stream data.

2.2- Setup and use DSH-PREDMNT dashboard
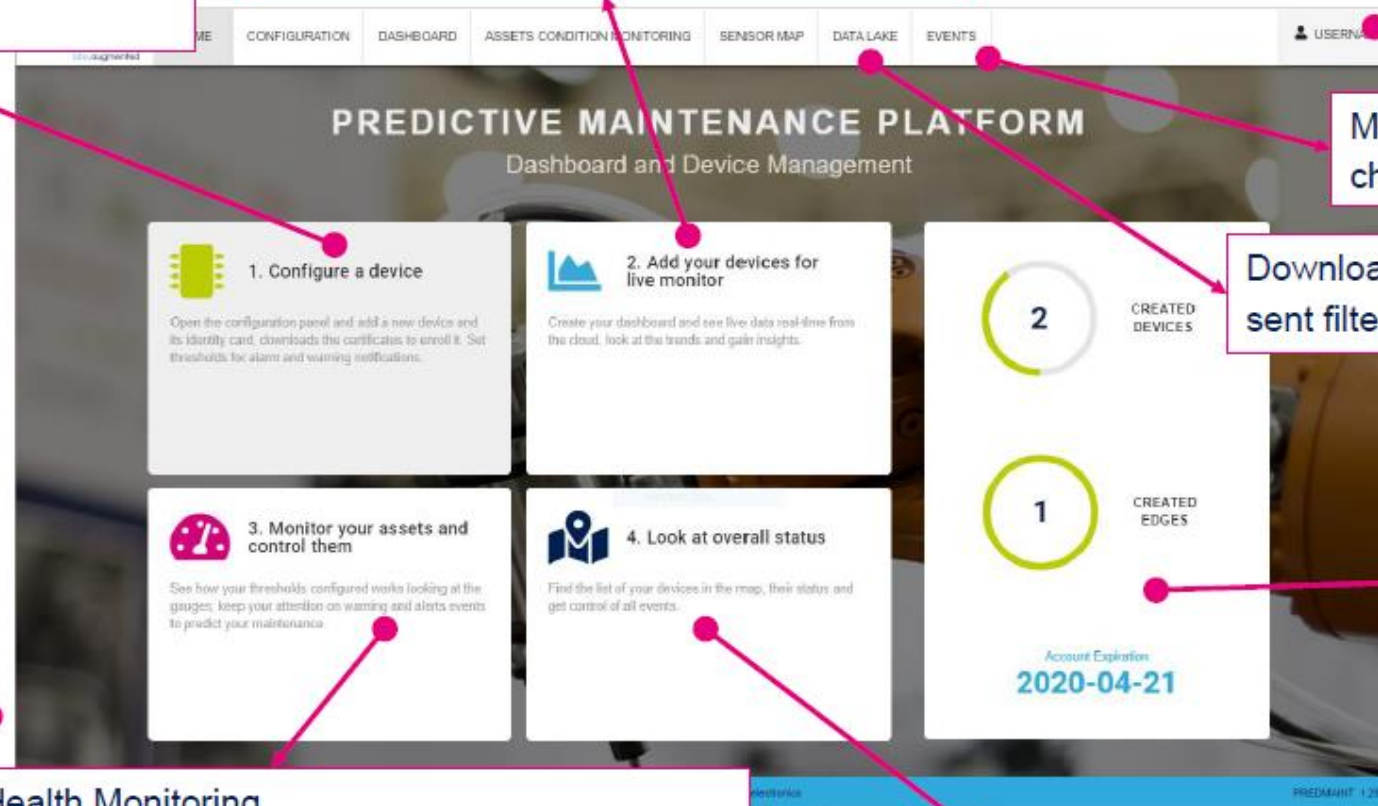
How to use <u>DSH-PREDMNT</u> dashboard

DSH-PREDMNT features

- Device and AWS Greengrass Edge registration and configuration:
  - Provisioning
  - Association to assets
  - Streaming time

- Live data visualization
  - Add a device to live monitoring

- User login and data segregation

Monitor live events or check events history

Download telemetry data sent filtering by device

Info Panel for the user about its terms

- English and Chinese

- Asset Health Monitoring
  - Collect data
  - Analyse Historical trends
  - Apply failure thresholds for alerts and warning

- Assets Map

PREDICTIVE MAINTENANCE PLATFORM
Dashboard and Device Management

1. Configure a device
Open the configuration panel and add a new device and its identity card, downloads the certificates to enroll it. Set thresholds for alarm and warning notifications.

2. Add your devices for live monitor
Create your dashboard and see live data real-time from the cloud, look at the trends and gain insights.

3. Monitor your assets and control them
See how your thresholds configured works looking at the gauges; keep your attention on warning and alerts events to predict your maintenance.

4. Look at overall status
Find the list of your devices in the map, their status and get control of all events.

2 CREATED DEVICES

1 CREATED EDGES

Account Expiration
2020-04-21

18

How to use DSH-PREDMNT dashboard

- Once connected you can

Click on **Events** to visualize anomaly events

Click on **Dashboard** tab to add a device to live monitoring

Click on **Configuration** tab to see the registered device and configure update time, measures and condition monitoring thresholds
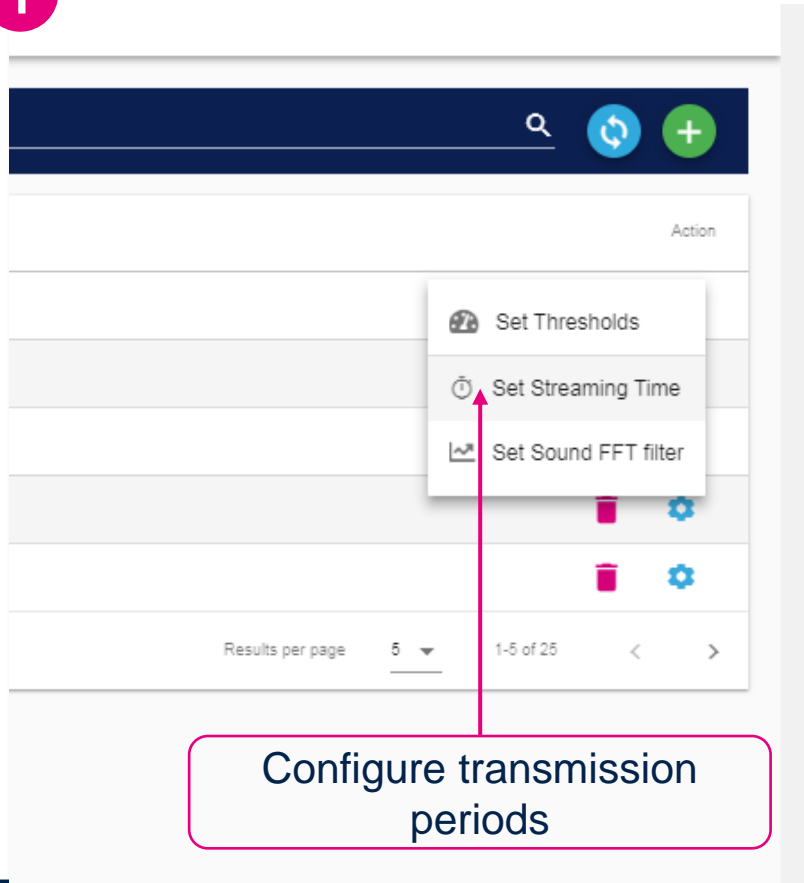
How to use <u>DSH-PREDMNT</u> dashboard

- Set streaming times to reduce bandwidth according to the system dynamics

1

Configure transmission periods

2

Streaming Time for Device: myDevice

Environmental time domain period

40s
10 ──────────────● 60

Inertial time domain period

20s
10 ──●──────────── 60

Inertial FFT time domain period

60s
30 ────●────────── 120

Acoustic time domain period

50s
2 ──────────────●── 60
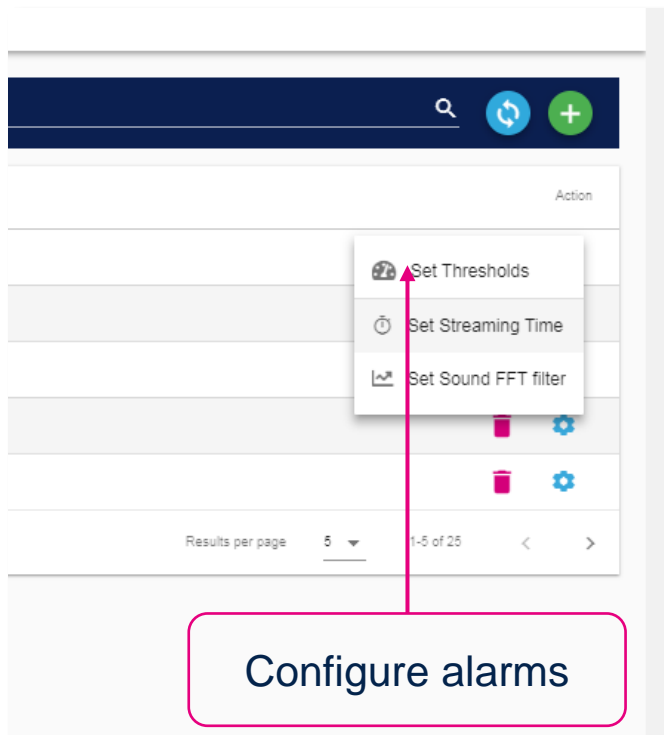
SAVE   CLOSE
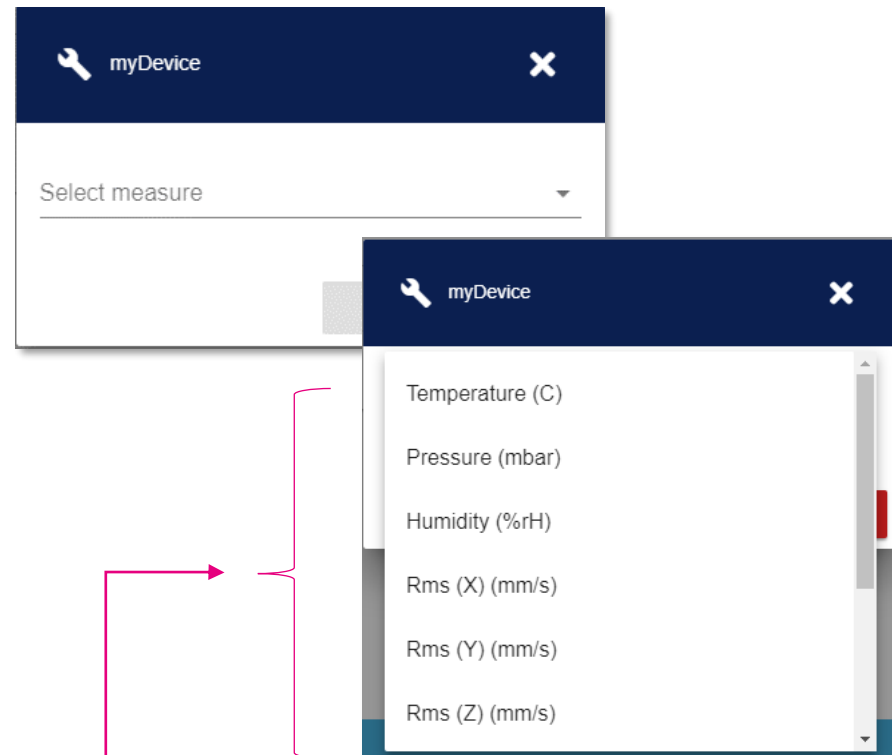
Drag balloons
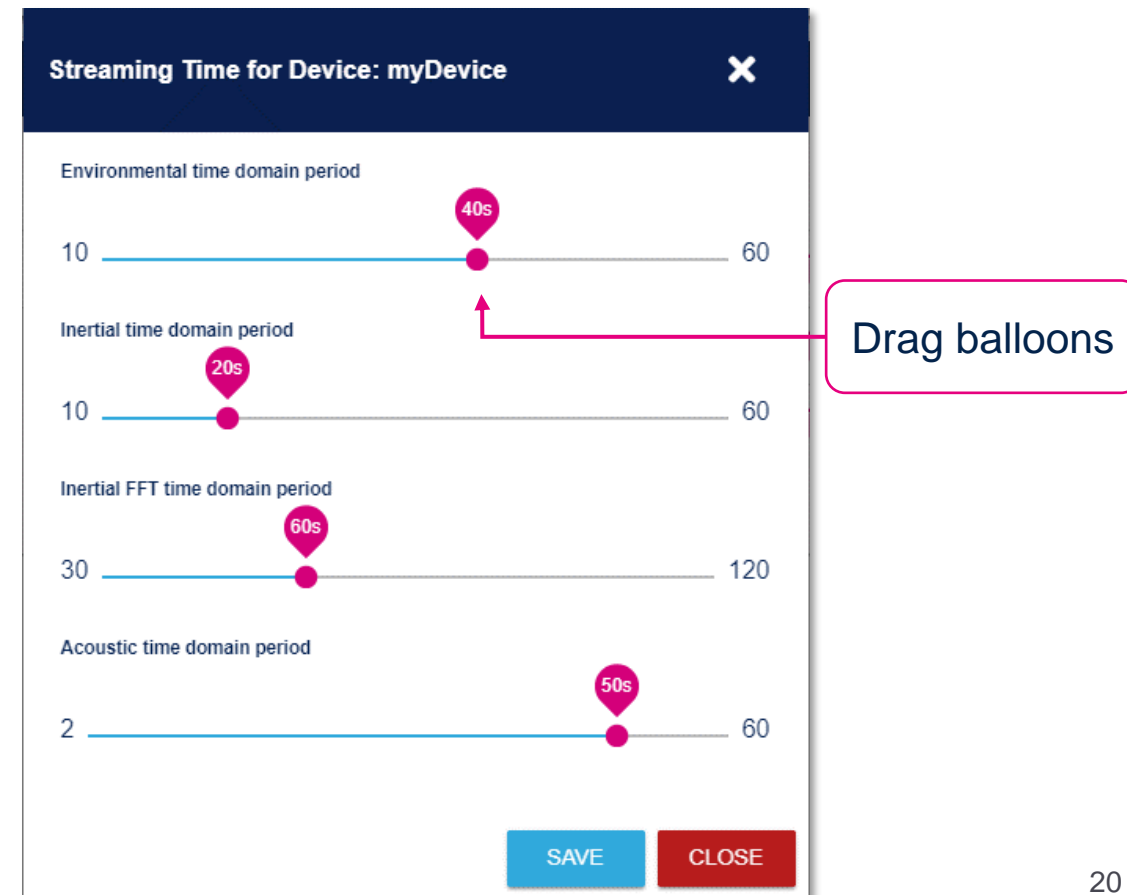
How to use DSH-PREDMNT dashboard

Configure measures and condition monitoring thresholds

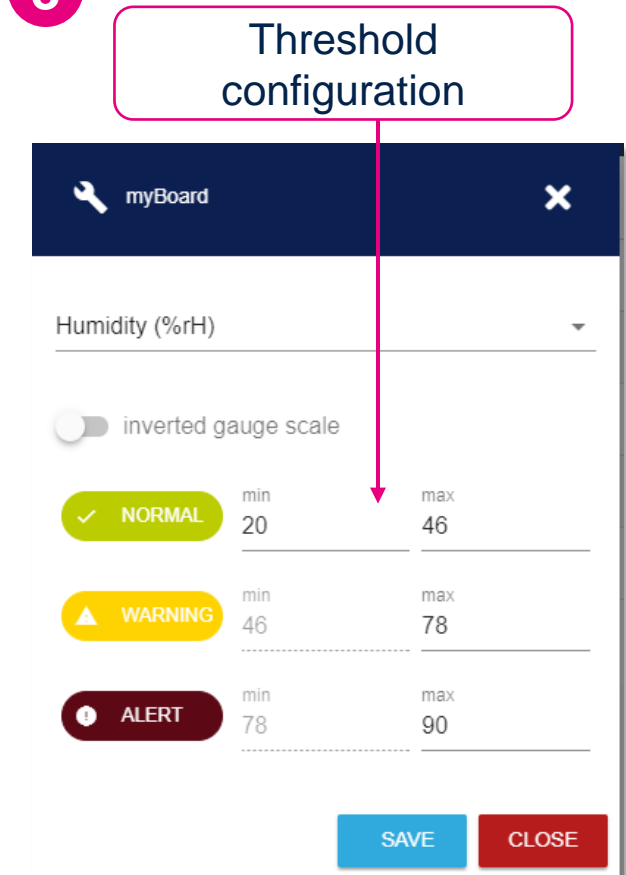- Configure the measures to take under control and set thresholds for warning and alerts

1 Configure alarms

2 Measures domains

3 Threshold configuration

How to use <u>DSH-PREDMNT</u> dashboard

Add a device to live monitoring

- Once devices are selected in the live dashbaord, a new cardbox will be showed and, if your physical device is transmitting, you will see live data monitoring refreshing.

- Data showed are customized according to the FW release of the device, they are:

  - Environmental Analysis

  - Vibration Analysis in Time Domain

  - Vibration Analysis in Frequency Domain

  - Sound Emission Analysis

Select your device for live monitoring

22
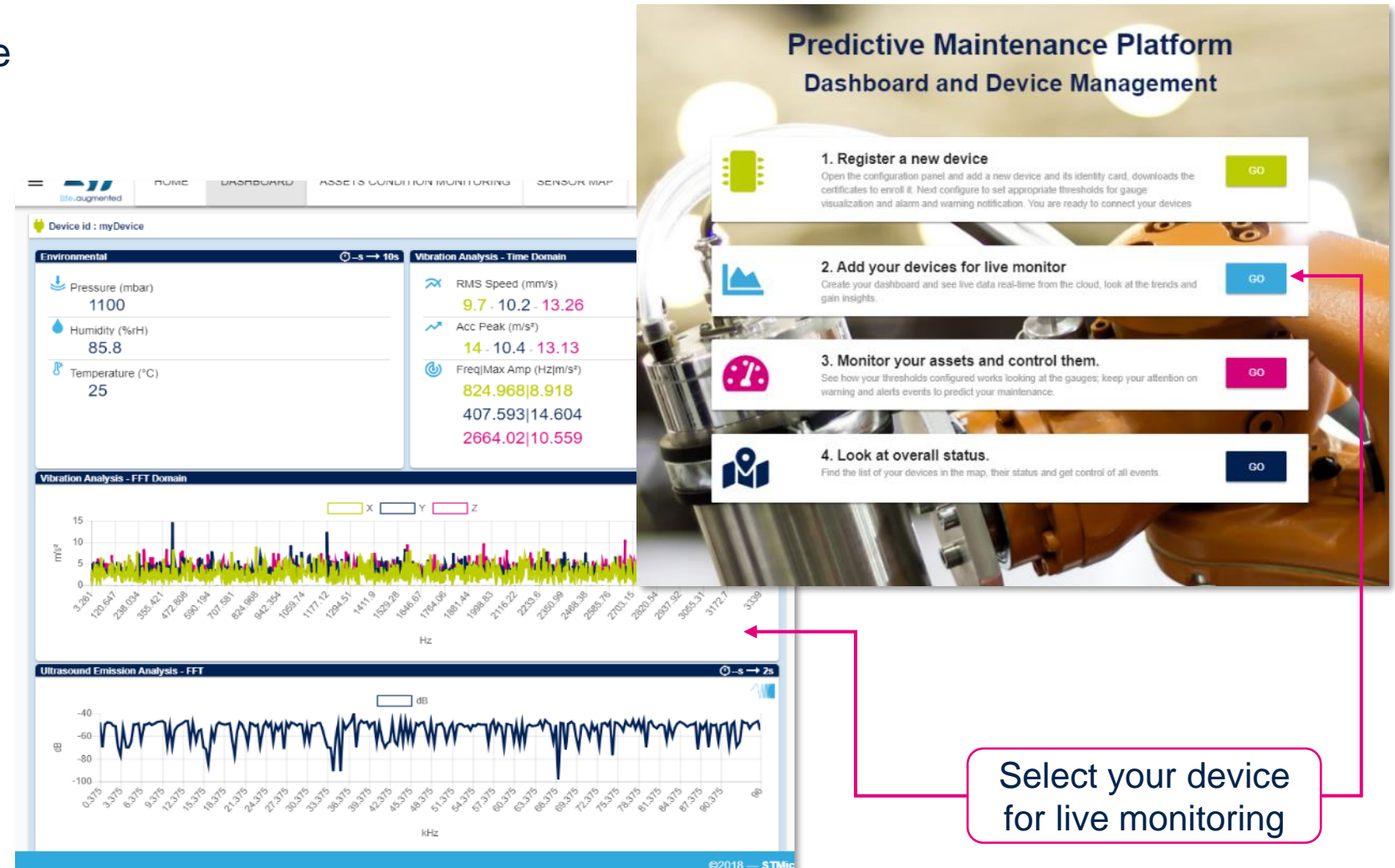
How to use <u>DSH-PREDMNT</u> dashboard
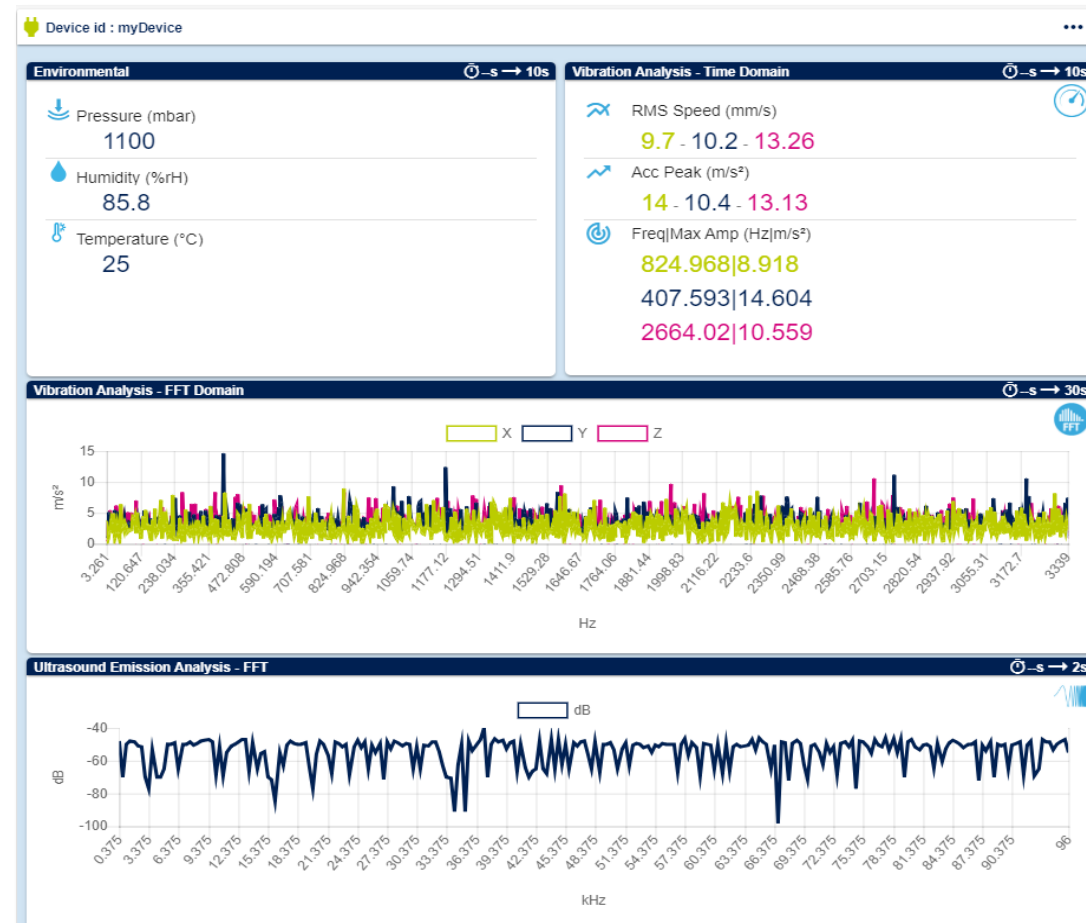## Hand Shaking Device – Flexible visualization

- The Dashboard recognizes the device, update the device shadow (that will be used for other purposes also) and adapt the visualization

Environmental domain

Vibration (FFT) domain
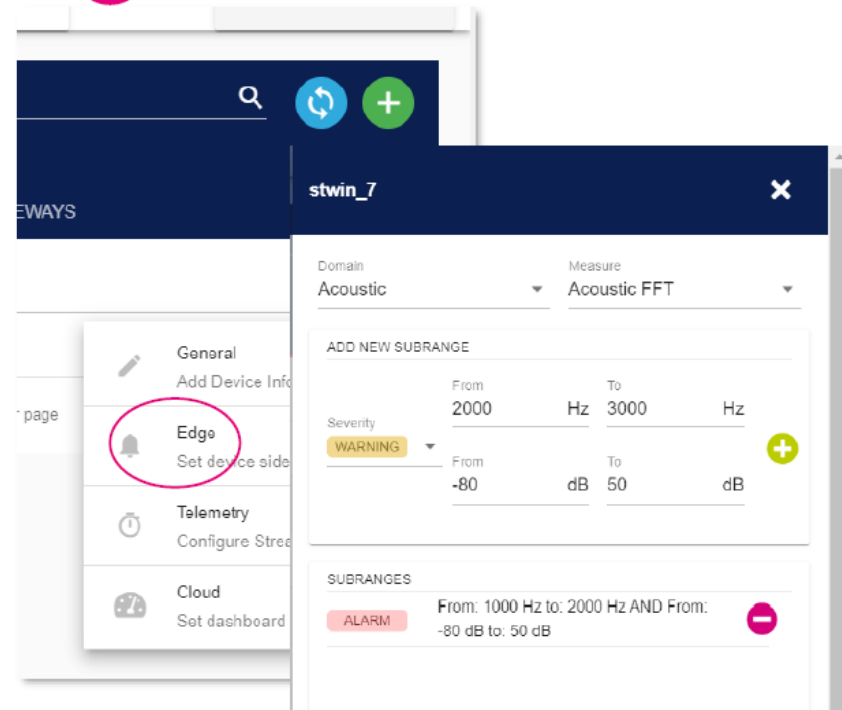
Vibration (TDM) domain

Acoustic domain

DHS-PREDMNT dashboard to visualize data and setup STWIN

1 Add thresholds to device

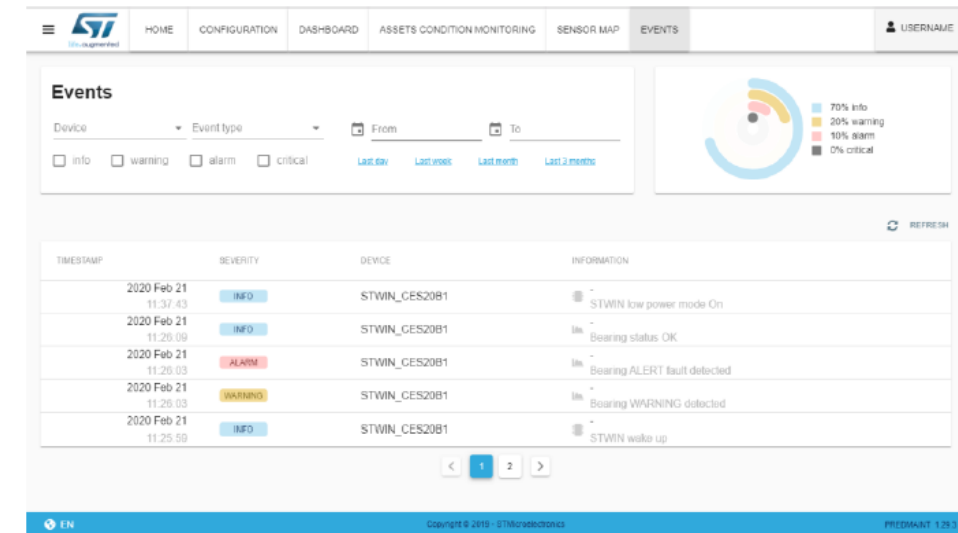**2** View events coming

Telemetry

Set Thresholds

Anomaly events

2.3- Setup NanoEdge™ AI library

Setup a NanoEdge™ AI library into FP-AI-PREDMNT2

Generating contextual data using FP-SNS-DATALOG1	Build library using NanoEdge™ AI Studio	Embed into FP-AI-PREDMNT2	Use with DSH-PREDMNT

Capture data	**Label data**	**Build NanoEdge Lib**	**Embed NanoEdge Lib**	**Process new data**

Accelerometer	Normal / Anomaly	Modeling / Results	NanoEdge integration / NanoEdge implementation	Real-time learning

1 **2** **3** **4** **5**
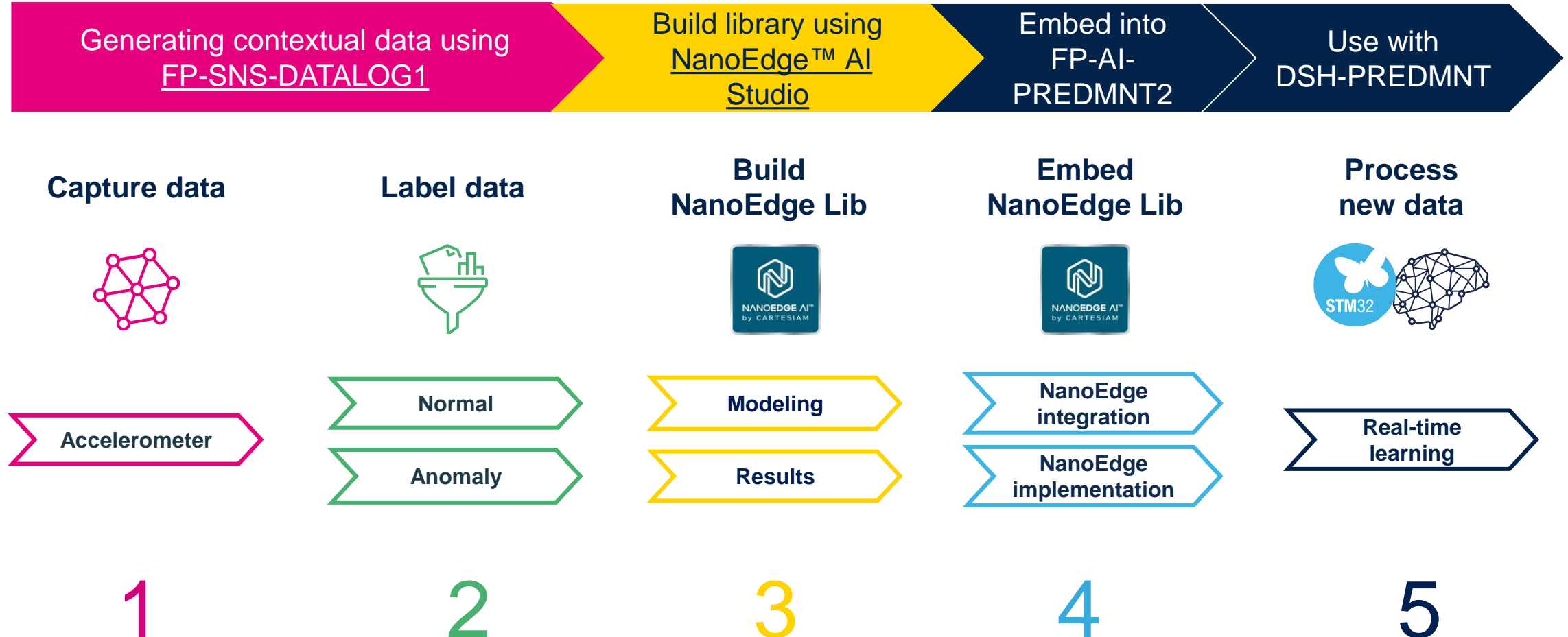
Capture data and create a labelled dataset

- Flash STWIN board with HSDatalog firmware
- Acquire training data onto SDCard
- Extract and convert logs using hsdatalog_to_nanoedge.py Python script

1 Generate data set

2 Label Data

Data log

SDCard

Serial
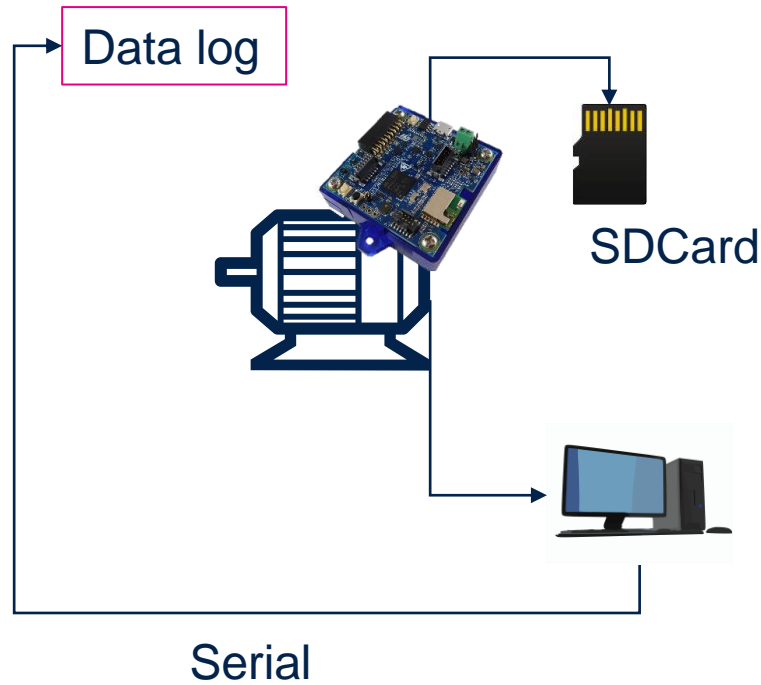
Regular signals file.csv

Abnormal signals file.csv

normalDataFull.csv

abnormalDataFull.csv

life.augmented

Use NanoEdge AI Studio tool

- Generate a Machine Learning library with NanoEdge™ AI Studio based on the provided data
- Download the library from the NanoEdge™ AI Studio

3 Building and Export Library file

➡️
1. Libneai.a
2. NanoEdgeAI.h

life.augmented

Embed into FP-AI-PREDMNT2

- Replace the provided stub in the function pack by the newly generated library
- Rebuild the firmware
- Program the STWIN with the new firmware

4 Integrate Lib files into the project

STM32_PREDMNT1_TMP.git/FP_AI_NANOEDG1/Middlewares/Third_Party/Cartesiam_NanoEdge_AI_Library/
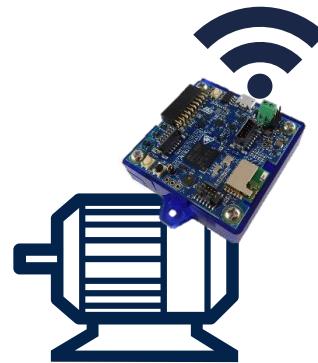
Name

📁 Inc

📁 lib → `NanoEdgeAI.h`

→ `Libneai.a`

5 Re-compile the FP

life.augmented

Running and Learning

- Learn the normal modes on the edge

6 NanoEdge Learn

7 Run the demo: NanoEdgeAI execute
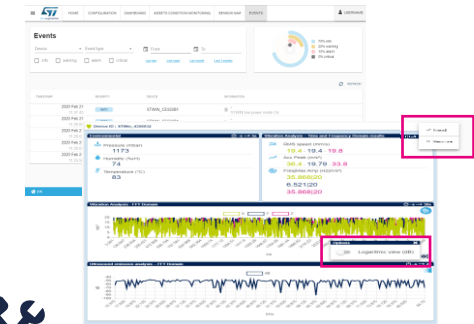
NanoEdgeAI_learn()

NanoEdgeAI_execute()

Running and Re-training

- Monitor anomalies in detection
- Go back to learn mode to improve the learning or to add new normal modes

7 Analyze and decide if new training should be launched
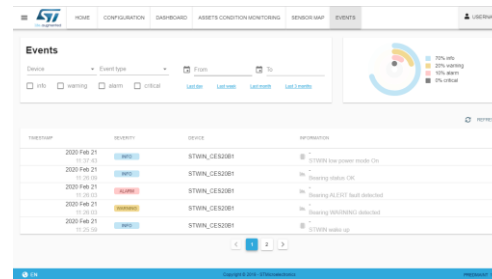
8 Run the demo again
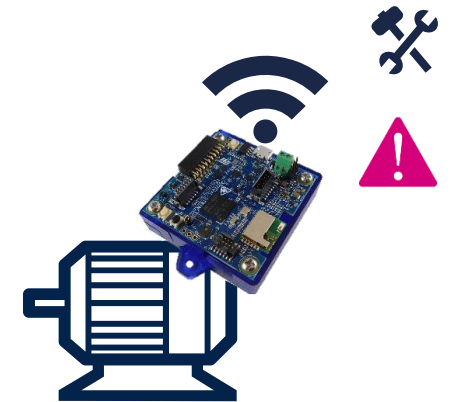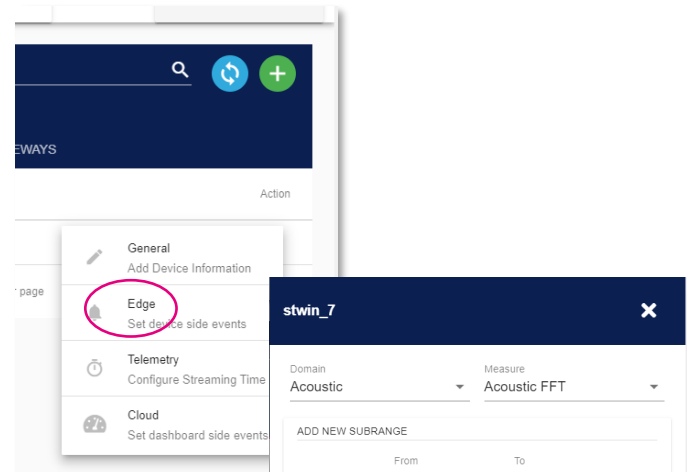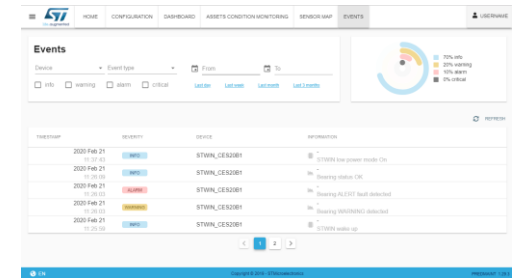
Send a command to NanoEdge™ AI library

- <u>DSH-PREDMNT</u> allow you to configure and setup your custom NanoEdge™ AI library

A. Click on the **Configuration** tab

B. Select the ⚙ symbol of the desired STWIN board

C. Click on **Anomaly detection**

Send a command to NanoEdge™ AI library

- The **settings** tab allow to setup the sensitivity and the threshold parameters

- The **commands** tab allow to chose either train or detect command. The result of the train command is the training of the neai library for the time interval set in the "phase duration" form. The detect command will trigger the anomaly detection phase in the application in according with the generated library

Detection phase of NanoEdge™ AI library

① Start detection

Orange led is blinking

Alarm sent

② Events

- If the detect command is set, in according with the custom NanoEdge™ AI library, an anomaly event is sent to the dashboard and an event will be shown in the EVENTS panel of the dashboard.

- For further details on how NanoEdge™ AI libraries work, read the detailed documentation of NanoEdge™ AI Studio.

3- Documents & Related Resources

Documents & Related Resources

FP-AI-PREDMNT2:

- **DB4566:** STM32Cube function pack for STEVAL-STWINKT1B evaluation kit for predictive maintenance application based on artificial intelligence (AI) – **databrief**

- **UM2937:** Getting started with the STM32Cube function pack for STEVAL-STWINKT1B evaluation kit for predictive maintenance application based on artificial intelligence (AI) – **user manual**

- **Software setup file**

STEVAL-STWINKT1B:

- **Gerber files, BOM, Schematic**

- **DB4345**: STWIN SensorTile Wireless Industrial Node development kit and reference design for industrial IoT applications – **databrief**

- **UM2777**: How to use the STEVAL-STWINKT1B SensorTile Wireless Industrial Node for condition monitoring and predictive maintenance applications – **user manual**

STEVAL-STWINWFV1:

- **Gerber files, BOM, Schematic**

- **DB3971:** Wi-Fi expansion for the SensorTile Wireless Industrial Node (STWIN) kit – **databrief**

Consult www.st.com for the complete list

Documents & Related Resources

DSH-PREDMNT:

- Cloud based web application for condition monitoring and predictive maintenance

To build up your custom NanoEdge AI library:

NanoEdge AI Studio:

- Automated Machine Learning (ML) tool for STM32 developers

FP-SNS-DATALOG1:

- STM32Cube High Speed Datalog function pack for STWIN evaluation kits

Consult www.st.com for the complete list

4- STM32 Open Development Environment: Overview

FAST, AFFORDABLE PROTOTYPING AND DEVELOPMENT

The STM32 Open Development Environment (ODE) is an **open**, **flexible**, **easy** and **affordable** way to develop innovative devices and applications based on the STM32 32-bit microcontroller family combined with other state-of-the-art ST components connected via expansion boards. It enables fast prototyping with leading-edge components that can quickly be transformed into final designs.

The STM32 ODE includes the following five elements:

o STM32 Nucleo development boards. A comprehensive range of affordable development boards for all STM32 microcontroller series, with unlimited unified expansion capability, and with integrated debugger/programmer

o STM32 Nucleo expansion boards. Boards with additional functionality to add sensing, control, connectivity, power, audio or other functions as needed. The expansion boards are plugged on top of the STM32 Nucleo development boards. More complex functionalities can be achieved by stacking additional expansion boards

o STM32Cube software. A set of free-of-charge tools and embedded software bricks to enable fast and easy development on the STM32, including a Hardware Abstraction Layer, middleware and the STM32CubeMX PC-based configurator and code generator

o STM32Cube expansion software. Expansion software provided free of charge for use with STM32 Nucleo expansion boards, and compatible with the STM32Cube software framework

o STM32Cube Function Packs. Set of function examples for some of the most common application cases built by leveraging the modularity and interoperability of STM32 Nucleo development boards and expansions, with STM32Cube software and expansions.

The STM32 Open Development Environment is compatible with a number of IDEs including IAR EWARM, Keil MDK, mbed and GCC-based environments.

STM32 Nucleo

STM32 Cube

| STM32 Nucleo development boards | STM32Cube development boards |

STM32 Open Development Environment

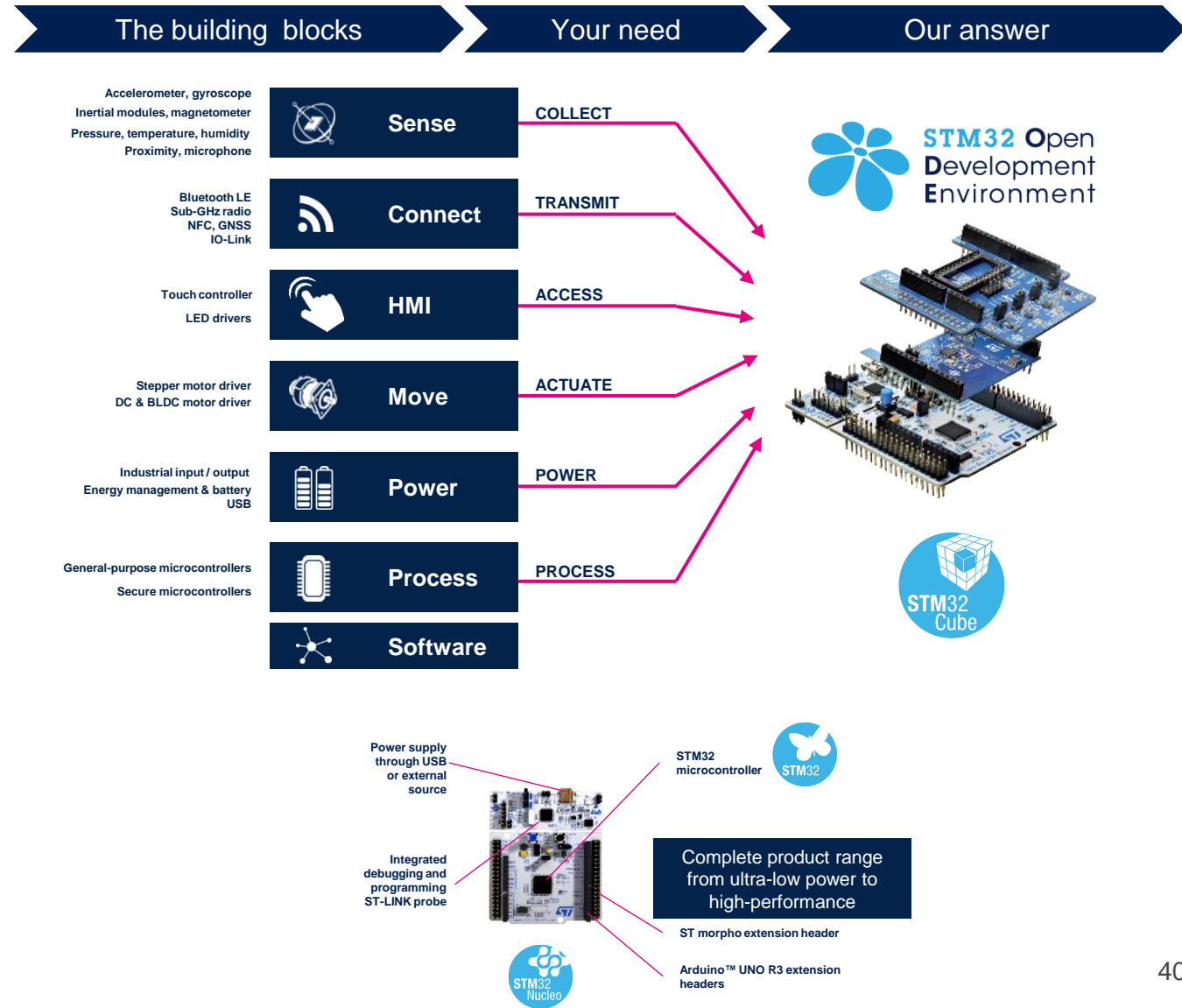| STM32 Nucleo expansion boards (X-NUCLEO) | STM32Cube expansion software (X-CUBE) |

Function Packs
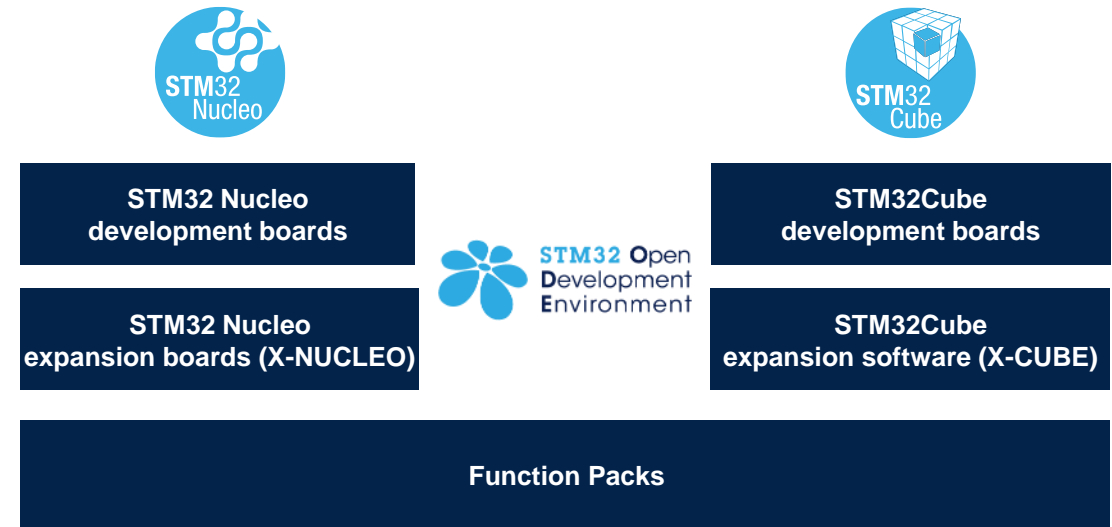
STM32 Open Development Environment: all that you need

The combination of a broad range of expandable boards based on leading-edge commercial products and modular software, from driver to application level, enables fast prototyping of ideas that can be smoothly transformed into final designs.

To start your design:

- Choose the appropriate STM32 Nucleo development board (MCU) and expansion (X-NUCLEO) boards (sensors, connectivity, audio, motor control etc.) for the functionality you need
- Select your development environment (IAR EWARM, Keil MDK, and GCC-based IDEs) and use the free STM32Cube tools and software.
- Download all the necessary software to run the functionality on the selected STM32 Nucleo expansion boards.
- Compile your design and upload it to the STM32 Nucleo development board.
- Then start developing and testing your application.

Software developed on the STM32 Open Development Environment prototyping hardware can be directly used in an advanced prototyping board or in and end product design using the same commercial ST components, or components from the same family as those found on the STM32 Nucleo boards.

The building blocks | Your need | Our answer

Building block	Components	Need
Sense	Accelerometer, gyroscope, Inertial modules, magnetometer, Pressure, temperature, humidity, Proximity, microphone	COLLECT
Connect	Bluetooth LE, Sub-GHz radio, NFC, GNSS, IO-Link	TRANSMIT
HMI	Touch controller, LED drivers	ACCESS
Move	Stepper motor driver, DC & BLDC motor driver	ACTUATE
Power	Industrial input / output, Energy management & battery, USB	POWER
Process	General-purpose microcontrollers, Secure microcontrollers	PROCESS
Software		

STM32 Open Development Environment

STM32 Cube

Power supply through USB or external source

STM32 microcontroller

Integrated debugging and programming ST-LINK probe

Complete product range from ultra-low power to high-performance

ST morpho extension header

Arduino™ UNO R3 extension headers

STM32 Nucleo

life.augmented

Thank you

life.augmented