Safectory Config APP 2.3.0 (Android)

safectory GmbH

Status - 15.04.2025

Compatibility - tested on Android OS versions 10 and higher.

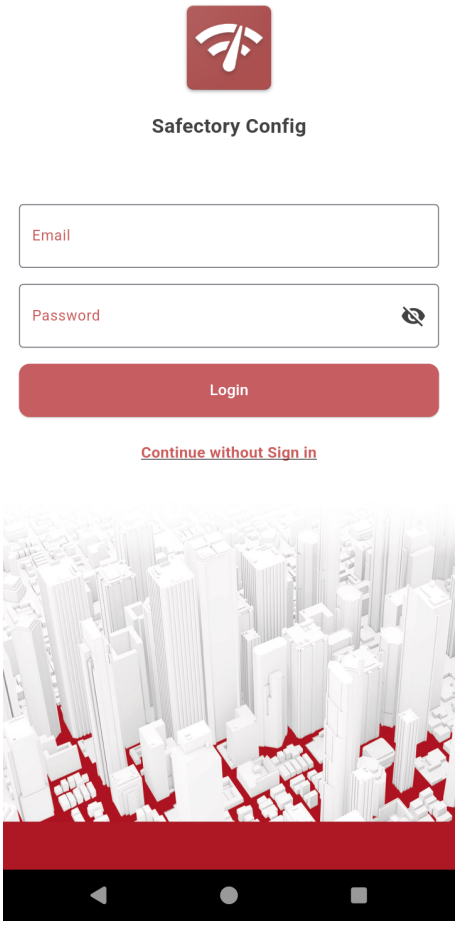
Proven interoperability

- Ascom: Myco 3, Myco 4
- Samsung: Galaxy S8, Galaxy S23
- Redmi A3

1 Intro

After starting the APP the Splash Screen and the Login Screen will be displayed.

Splash Screen



Login Screen

2 Login Screen

Unlock additional features such as:

- Connecting to your device to read and reconfigure settings (learn more in the Connecting Devices section)
- Uploading read-out device configurations to Track
- Import production CSV data and synchronize scanned devices with the device name from the CSV file.

To gain access, please contact Safectory support to request the creation of an account with the `ASN Config Definition` attribute.

This attribute, which will be assigned to your user profile, determines which characteristics and settings you can view or modify once connected to a device. Depending on the permissions granted by the TrackUI administrator, some characteristics and services may be read-only, hidden, or fully editable. If certain features appear missing or restricted, please contact Safectory support to have the necessary attribute added or customized to fit your needs.

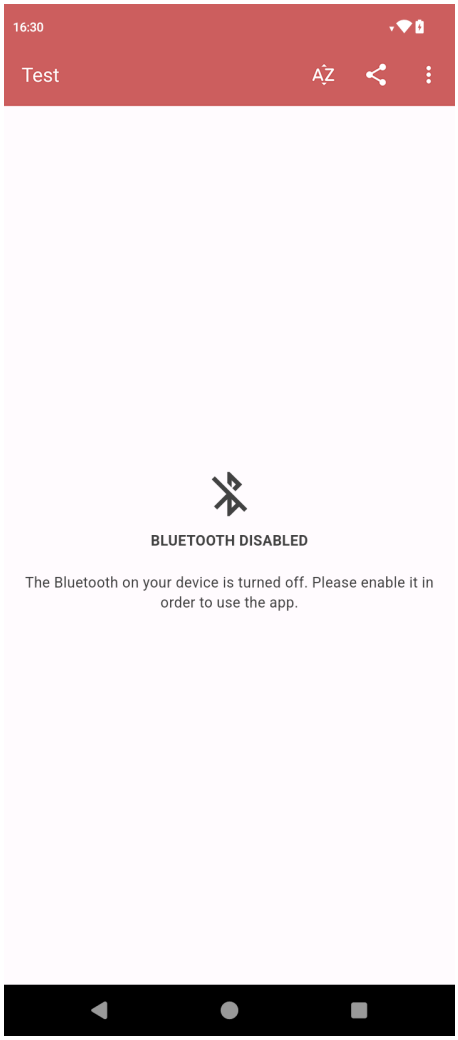
2.1 “Continue without Sign in”

- see devices list screen
- see properties of the device

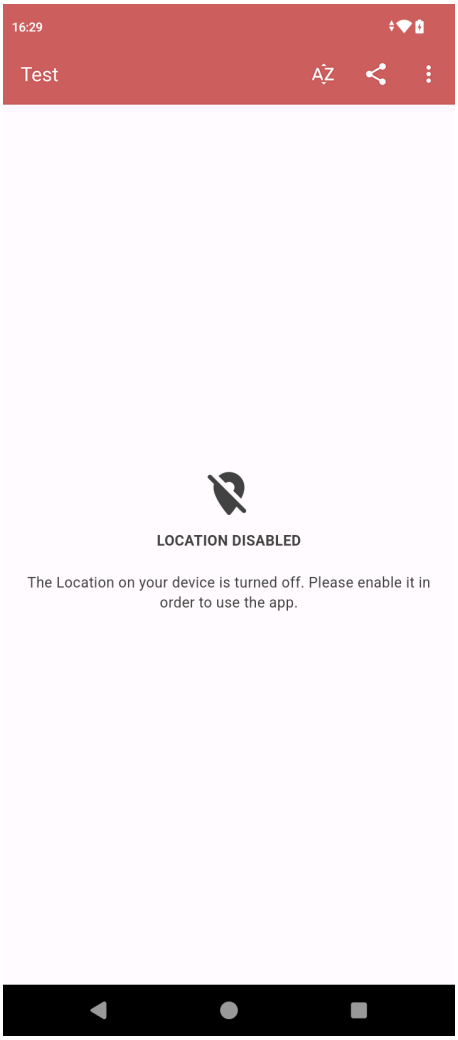
3 Devices List Screen

3.1 Enable bluetooth and location permissions

To use the app Bluetooth and Location permission need to be enabled. In case they are not the following screens will be dispaled

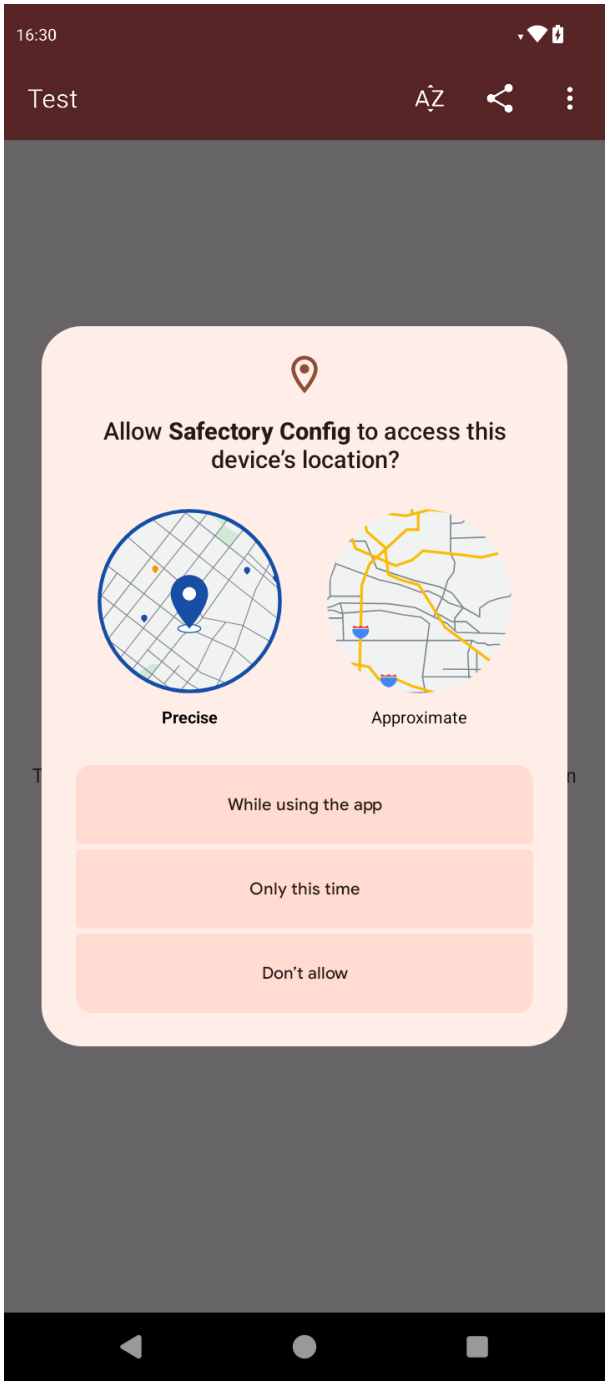


Bluetooth disabled



Location disabled

Furthermore the app will ask you to enable, only during the first time, the following permissions



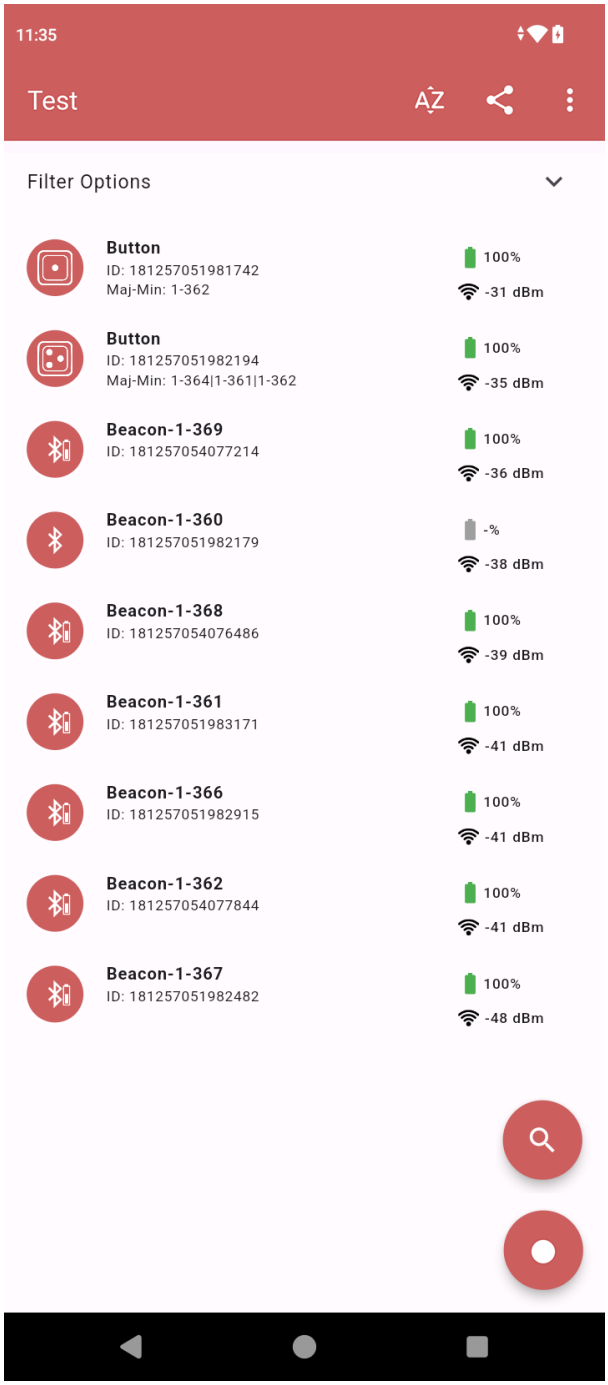
Device Location (select `While using the app` to proceed)

Find and connect to nearby devices (select `allow`)

Now, you will be able to scan the Bluetooth devices around you by tapping the magnifying glass icon.

3.2 Device list

Main Screen (Devices List Screen)



Devices list

- Bluetooth devices around are sorted by RSSI (closest will appear at the top of the list) by default but they can be also sorted (only ascending) by major or minor by using the sorting button in the menu.
- If scanning is started it will be done continuously until the stop button is clicked (lower right side).
- It is also possible to start scanning by changing the filters in the filter menu (e.g change RSSI filter)
- “The”share” button  in the menu allows you to easily share the list of discovered devices in CSV format after a scan process has been completed. If you have applied filters, the device list will contain only the devices that meet the filter criteria.

- **Note**: this this option is only available **after stopping** the scan.
- This exported CSV file can be used to import the device list into the backend.

NOTE: Imported Button/Scan tags will not appear online in TrackUI because the required UUID information (needed as an attribute) is missing.

3.3 Icon legend

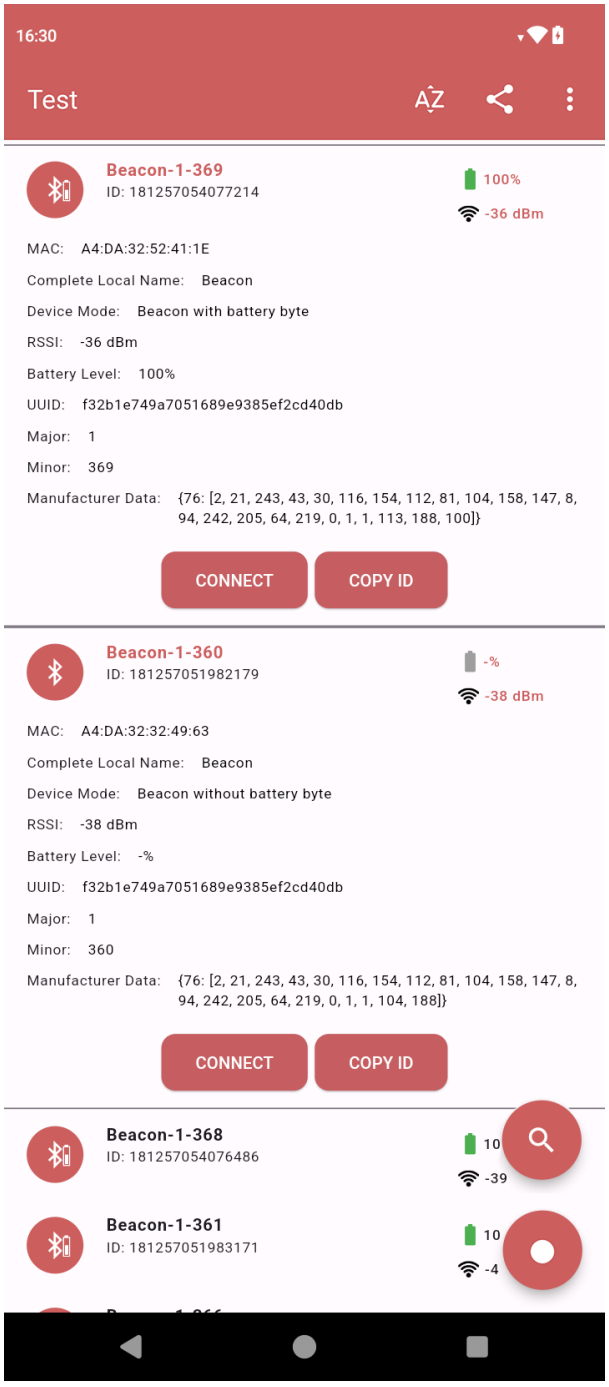
The available Safectory devices can be easily distinguished by their icons:

-  **Bluetooth symbol:** Safectory beacons not sending battery level (no battery byte)
-  **Bluetooth and battery symbol:** Safectory beacons with battery level (with battery byte)
-  **One dot:** Safectory Asset Tags scanning for one location beacon (`Button Tag`)
-  **Three dots:** Safectory Asset Tags scanning for three location beacons (`Scan Tag`)
-  **Bluetooth symbol with gray background:** Unknown devices (available Bluetooth devices)

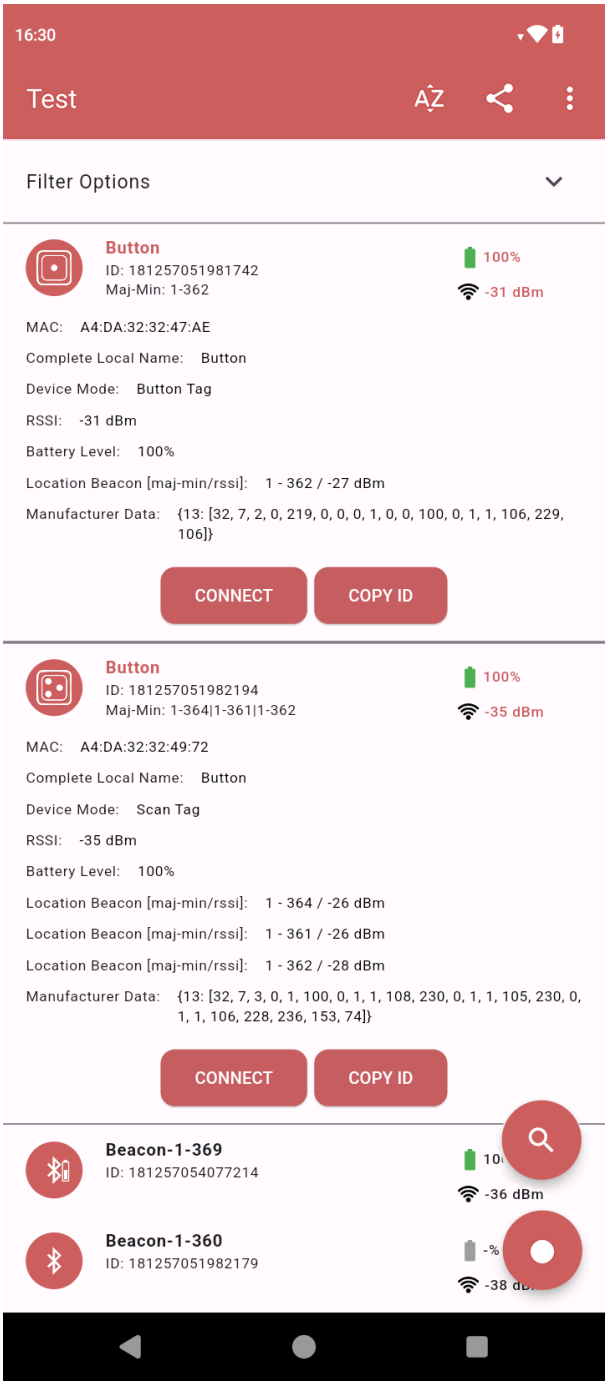
3.4 Device's Info

- The device interface boasts an intuitive design that simplifies the identification of devices following a scanning process. Each device is represented by distinct icons and names (either "button" or "beacon").
- Beneath the name, additional information is provided, offering insights into the device type.
  - **Beacons:** major and minor details are incorporated into the device name
  - **Buttons**: advertising button or scan frames, information such as the major and minor values of advertised devices (three in the case of scan frames) is included.
- **Battery level**
  - If battery level >= 60 && batteryLevel <= 100, show a green battery icon plus the percentage value (e.g. icon + 70%)
  - If battery level <= 59, show a yellow battery alert icon plus the percentage value
  - If battery information is not available on the device, a grey battery icon plus the text "-%" is shown
  - In scenarios where the device operates as both a button and a beacon without a designated battery level, the battery value is preserved during button operation and is revealed when the device functions as a beacon without a battery level.
- **RSSI status** can be monitored in the list, values and icon change in real-time while scanning:
  - If RSSI >= -50, icon showed is  .
  - If RSSI >= -70, icon showed is  .
  - If RSSI >= -80, icon showed is  .
  - Otherwise, icon showed is  .
- Clicking on devices will expand and show information depending on device type like:
  - Device Name
  - Device mode
    - Beacon with battery level (with battery byte)
    - Beacon without battery level (without battery byte)
    - Asset Tag scanning for one location beacons (`Button Tag`)
    - Asset Tag scanning for 3 location beacons (`Scan Tag`)
  - Received signal strength indicator (RSSI) between App and BLE-device
  - Battery Level

  - UUID (beacon only)
  - Configured Major (beacon only)
  - Configured Minor (beacon only)
  - One to three strongest Location Beacon(s) (maj-min-rssi) and their RSSI advertised by Asset Tag
- Depending on the device type (device mode), the displayed information can vary.
  - For example, in beacons without a battery byte, the battery level field will be absent.
  - In addition to the information on the device, a “COPY ID” button is displayed, which allows to copy the device ID to the clipboard



Device list, showing beacon withand without battery byte



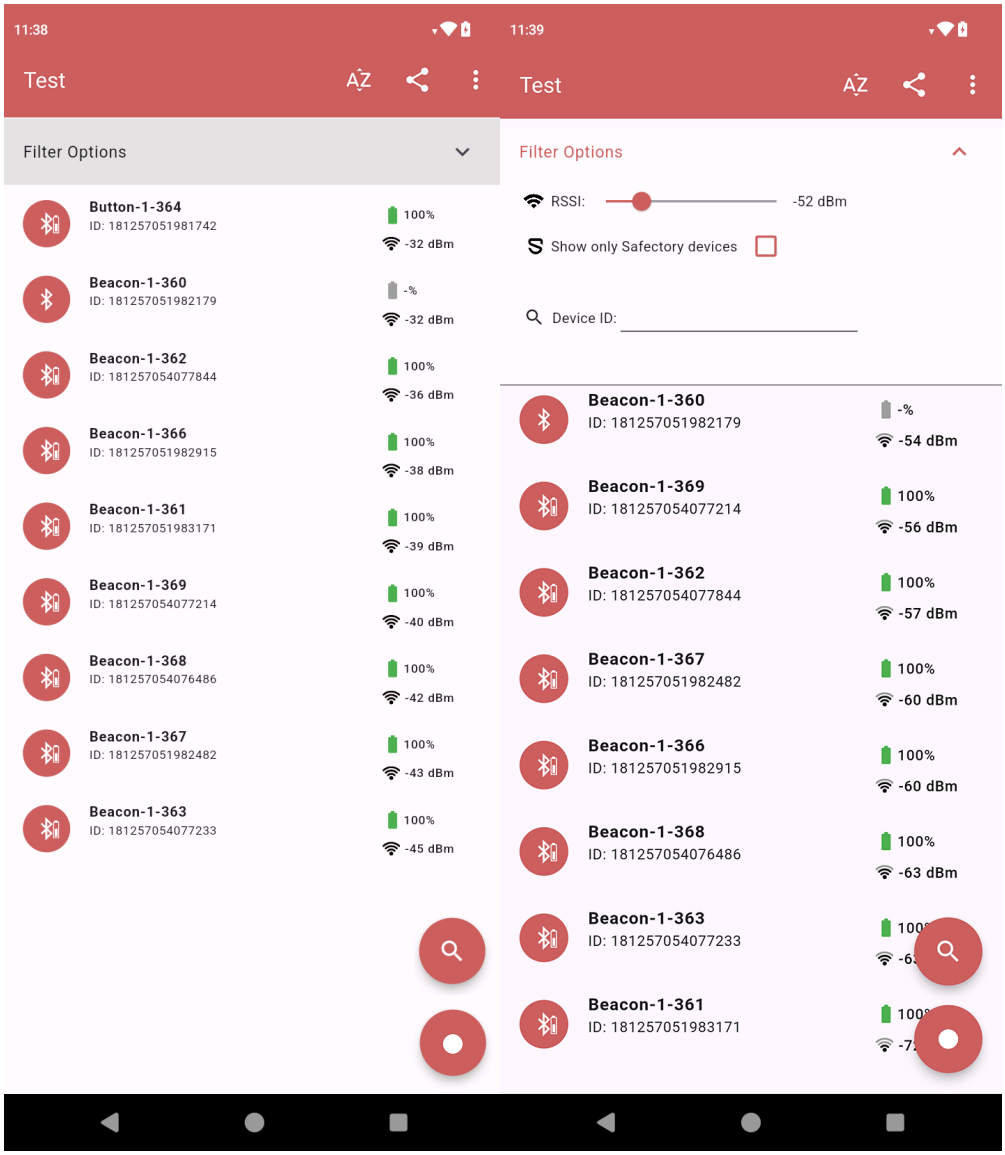
Device list, showing `Scan Tag` and `Button Tag`

3.5 Filtering options

The App also provides device filtering operations. Click on `Filter Options` to open a drop-down menu with the following possibilities:

- **RSSI**:
  - Display devices within a specific signal strength range.
  - Choose a range between -40 dBm (near) and -100 dBm (far).
- **Only Safectory devices** (set as default) will be shown
  - uncheck to scan for all BLE-devices
- **Device ID**:
  - filter by device ID to isolate BLE-devices and better observe updates.

To close the drop-down menu click on `Filter Options` again.



Filter menu | Filter menu open

4 Characteristics Screen

- After Log-in, the Device List Screen will display your user name in the toolbar
  - **NOTE**: User names longer than 15 characters are truncated with "…" at the end.

- With the login is also possible to access the beacons recording feature. Give a look at the Beacon Recording section for more information.
- In the 3 dot menu you will find:
  - About: to go to the About screen and read info about the App (version and commit number) and the company
  - Log out: to gets back to the Login screen
- Device display and filtering will work as described in the device list
- On closing and reopening the app the user information stored in track will be downloaded to the APP.
- Another way to update the user data is to perform in the device list a pull to refresh.
- In this case a notification will appear on the lower end of the screen to notify you that the user data has been successfully updated.

App menu

User data updated successfully

4.1 Beacon Recording

Beacon recording allow you to check th status of the nearby devices starting from a production `.csv` file provided by Safectory or a consume track device list exports (exported from TrackUI).

To use the recording feature, you must be logged in. Once logged in, a **Record** button will appear below the **Scan** button in the bottom-right corner.

Clicking on the recording button a file picker will open allowing you to import the `.csv` file .

Once the file is successfully imported, the recording will begin automatically, and a message will confirm **"Recording Started."** During the beacon recording, any detected devices that match those in the .csv file will be displayed with their corresponding names from the file.

The recording duration will be displayed in the app toolbar, and all imported devices from the `.csv` file will be listed in the devices list. All other active devices nearby that do not belong to the imported list will be filtered out and will not show in the devices list.

During the scan, detected devices will be updated in real-time. Devices that are found will have a **green icon**, and their **battery level** and **RSSI (signal strength)** will be displayed. Devices that are **not detected** will have a **grey icon**, with **"Not seen"** displayed for RSSI and **"-%"** for battery level.

To stop the recording, press the **Stop** button. While the recording is active, you cannot connect to or copy the IDs of detected devices. Once the recording is stopped, a **summary dialog** will appear, showing the recording duration, total devices imported, and the number of detected devices. The dialog will also provide two options: **Share**, which exports the recording data as a CSV file, and **Cancel**, which closes the dialog.

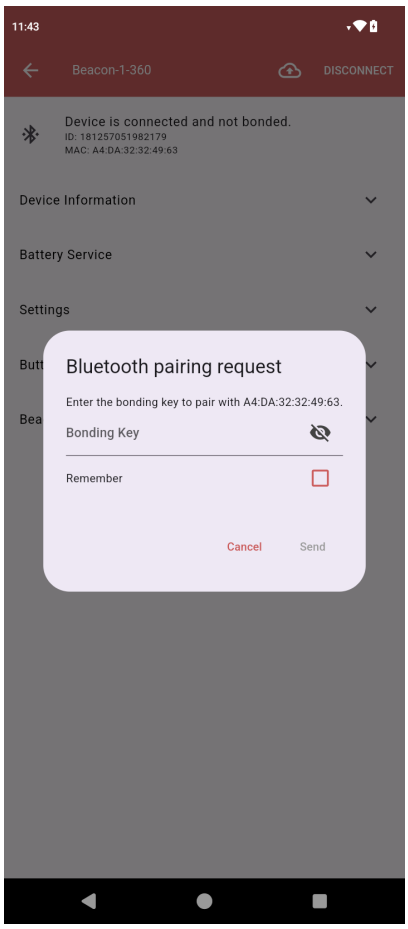
When sharing the data, a CSV file identical to the imported `.csv` will be generated, with three additional columns: **Battery Level**, **Last Seen Time**, and empty rows/columns indicating undetected beacons. The CSV file name will include the **date of the recording** and the **name of the logged-in user** for easy identification.

4.2 Connecting devices

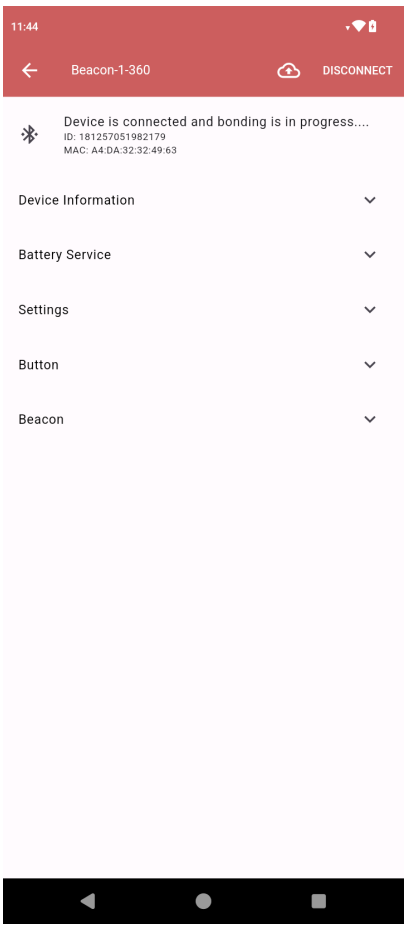
- Select a device from the list.
- The "CONNECT" button is displayed. After clicking it, after few seconds, the characteristics list will be displayed.
- If a device is not already bonded, a bonding request should appear as soon as you attempt to:
  - Read/write a characteristic that requires bonding (all except "Device Information" and "Battery Service").
    - You should first see a notification saying: "You can't read/write this characteristic because the device isn't bonded or the bond was lost. Initiating bonding process now..."
  - Transmit the data to the backend (cloud button).
    - You should first see a notification saying: "You can't send the device info to the backend because the device isn't bonded or the bond was lost. Initiating bonding process now..."
- Insert bonding key to pair it with your smartphone: the bonding key should be a 6-digit number and not contain any 0s or white spaces.
  - The "eye symbol" will reveal your bonding key.
  - Hit "remember" to save the bonding key. Only the last bonding key will be remembered.
  - Trying to bond with a BLE device with another bonding key will reopen the Bluetooth pairing request
- Options to remove a saved bonding key:
  - re-install the app
  - delete the app cache via Android OS:
    - `setting > application > safectory config > storage memory > clear cache`
    - or `setting > application > safectory config > storage memory > clear APP-data`

Bonding may fail if the insert bonding key is incorrect, if the device (Scan Tag / Button Tag) is not in motion, or if it is too far from the smartphone. In such cases, an error message will notify you of the failed bonding attempt. In case bonding is lost (if the native Android bonding dialog appears), clicking "Cancel" should disconnect the device and update the bonding status to "not bonded." The app detect when a reading fails

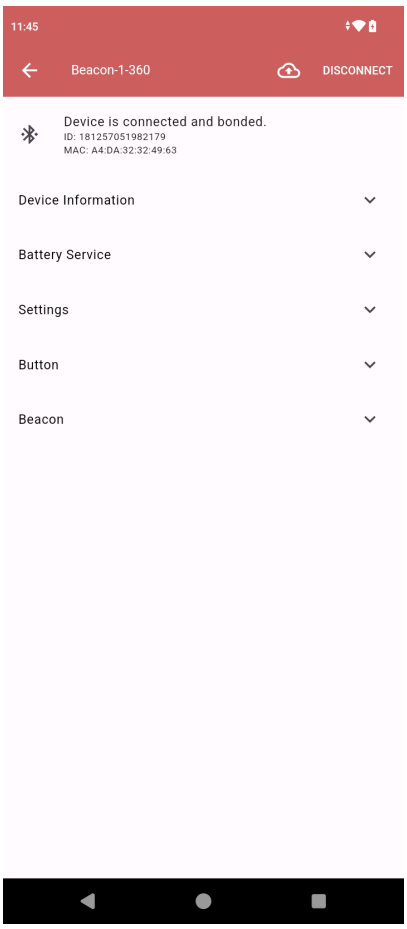
due to a lost bond, disconnect the device, and display ‘Not bonded.’ Reconnecting the device and attempting to read a feature that requires bonding should enable reading of that feature again.



Bonding process



Bonding Status

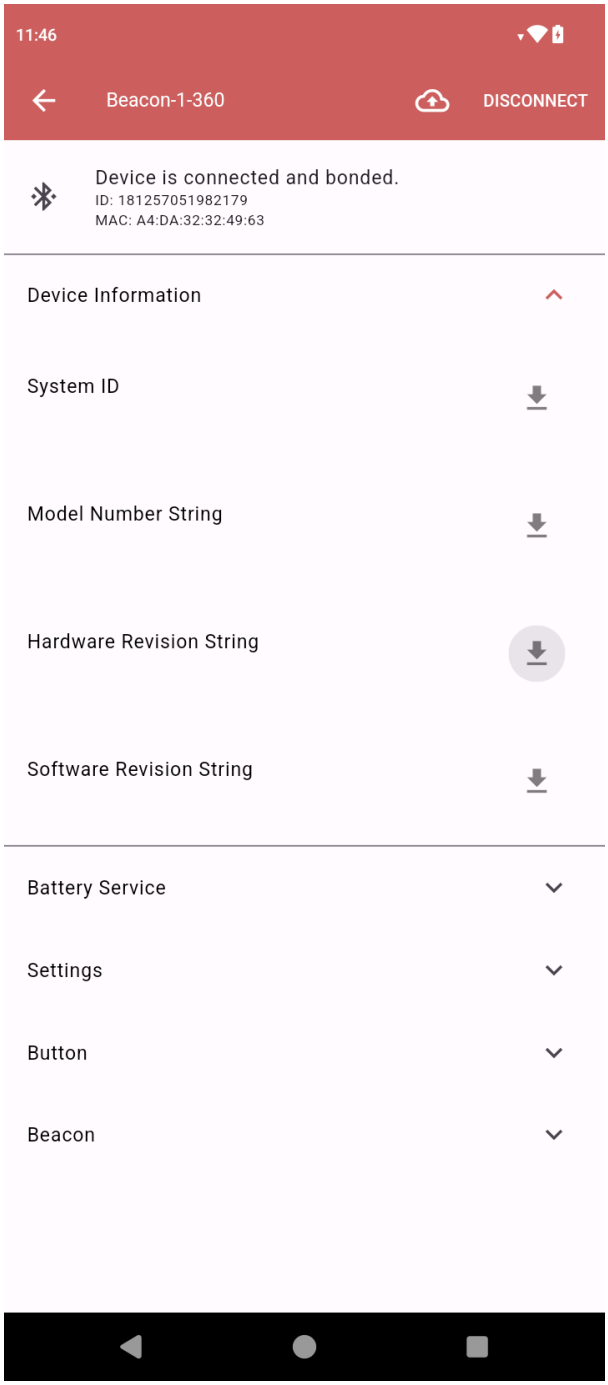


Bonding successful

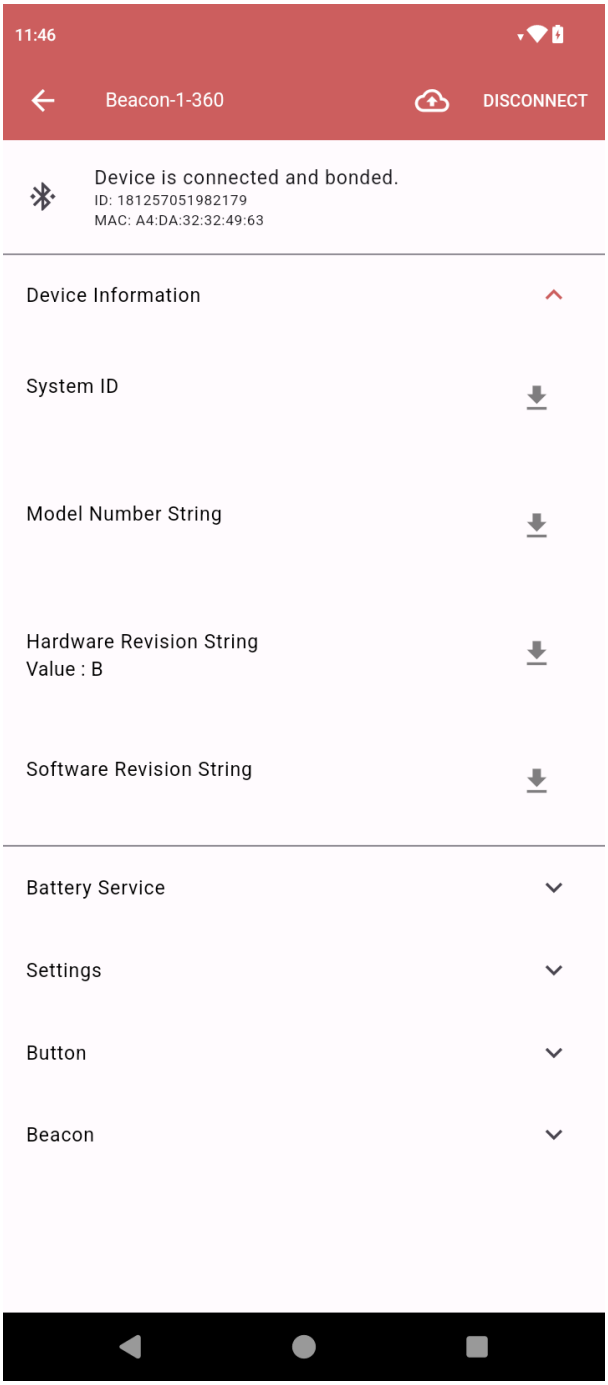
- The start of the bonding process is displayed via message “bonding is in process”.
- Wait for the Device to be connected (depending on the device, the connection can take a few minutes).
- On connection a list of services is shown.
- When connecting to a Scan Tag / Button Tag an alert dialog is displayed showing the text: “Scan Tag / Button Tag detected. Keep the device in motion during bonding and configuration.”

4.2.1 Read and Edit a characteristic

- Clicking on the services will reveal their associated characteristics.
  - Generally, services and characteristics are presented in a human-readable format.
  - Depending on the characteristic, it is possible to perform read and edit operations on the value (up and down arrows), or read only or write only.
  - The characteristics of the services “Device Information” and “Battery Service” do not require bonding and can be read without it.
- Bonding is necessary to read or write certain characteristics.
- “Reading…” is displayed when pressing the “Read” button until the value is displayed.

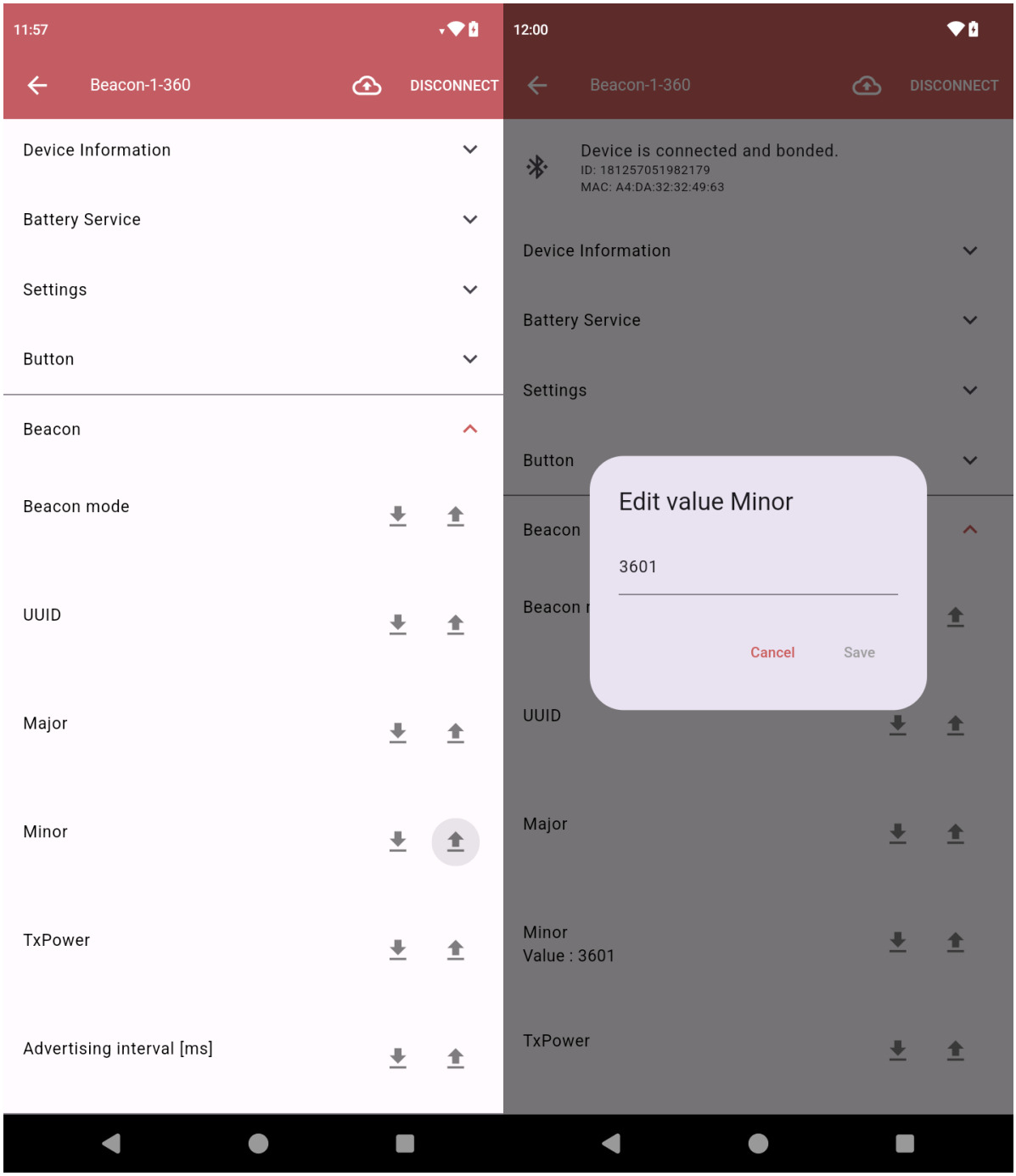


Reading button



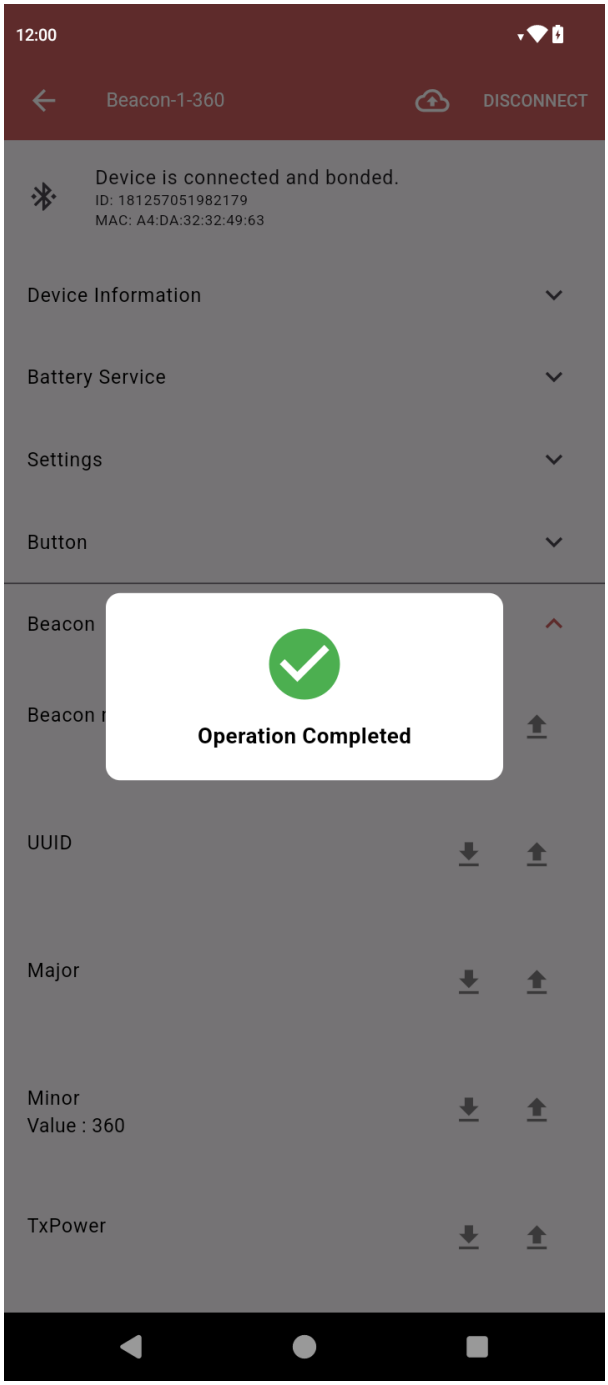
Value displayed

- To edit a characteristic, a click on the up arrow will open a dialog displaying the current value of the characteristic, allowing you to edit and save the new value.
- Please refer to the ‘Value Guidelines’ section to see the allowed values or range of values you can enter for each characteristic. This section also includes a brief description of each services and characteristics.

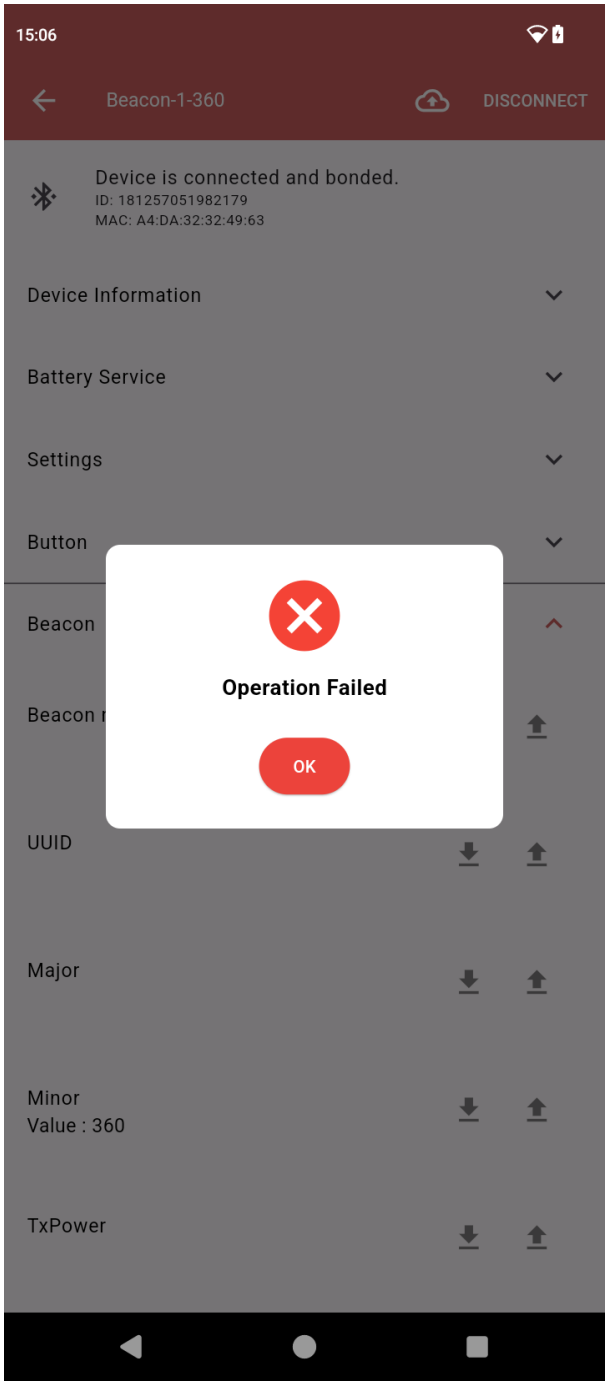


Writing operation Edit characteristic dialog

- The Save button will be enabled only if the entered value is valid.
- If the value is empty or the UUID is invalid, an appropriate error message will be shown below the input field.
- Upon successful saving, a confirmation message with a green icon will be displayed.
- If saving fails, a failure message with a red icon will be shown instead.

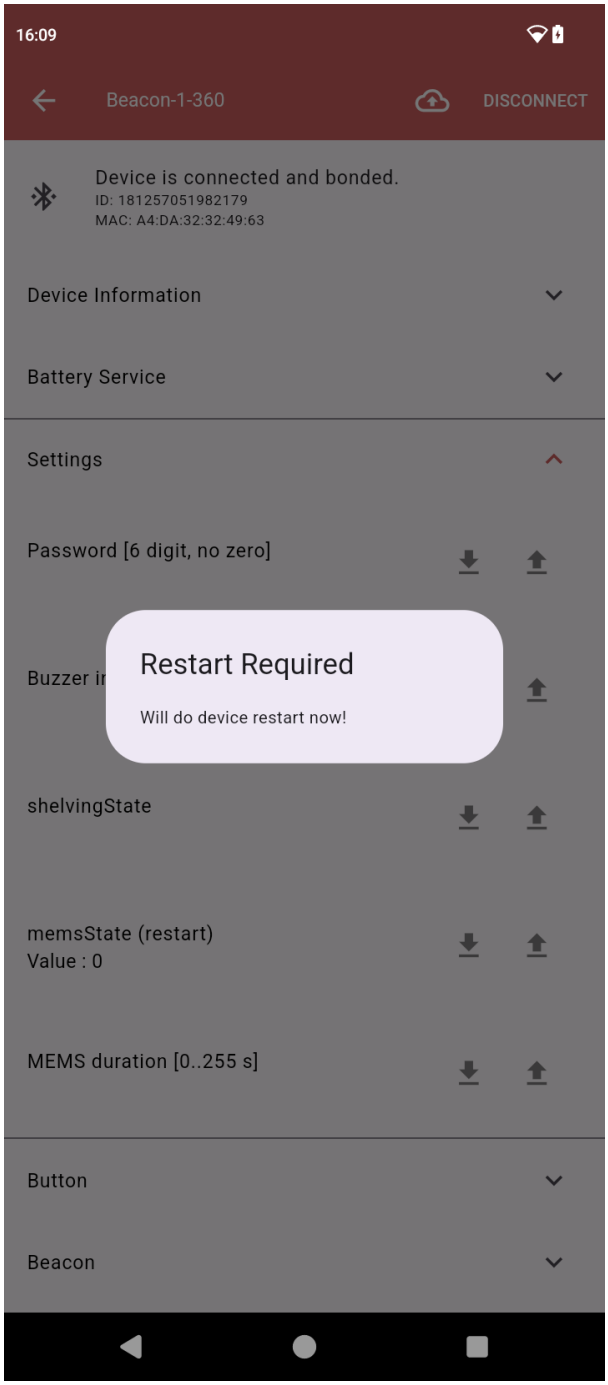


Editing Completed

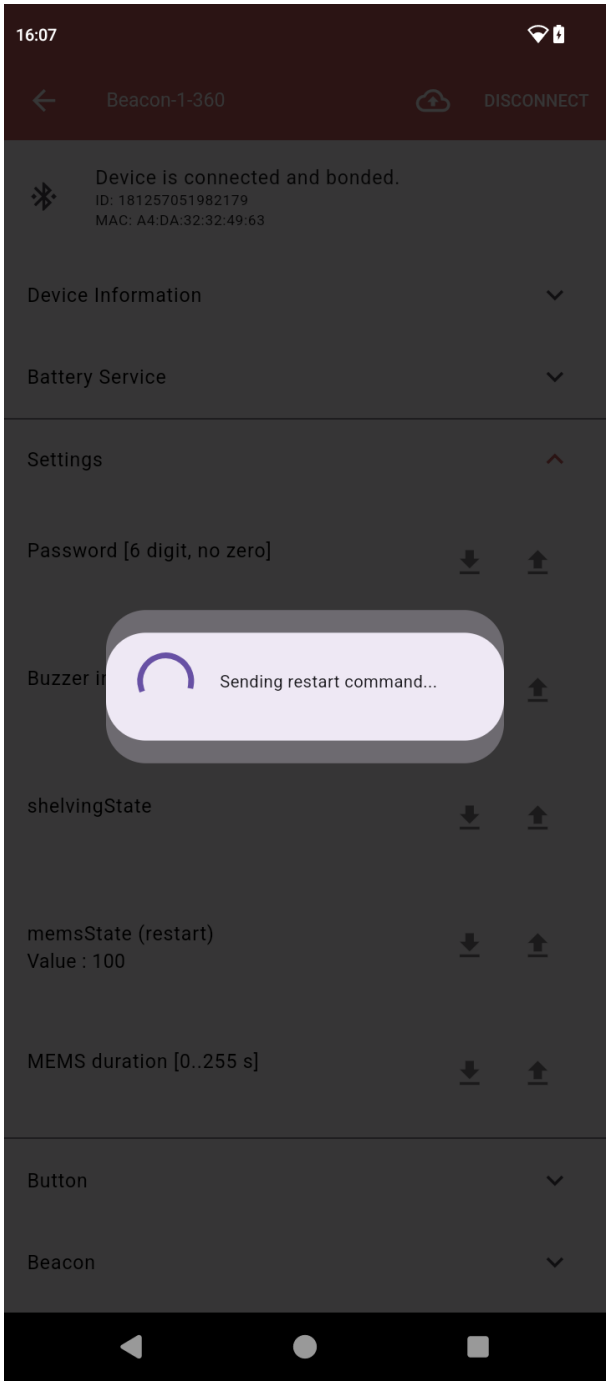


Editing failed

- Characteristics list value will be updated
- characteristics marked “restart” will trigger a BLE-device restart on change. (e.g., memsState). When one of these characteristics is edited, a ‘Write in Progress’ notification is observed, followed by a ‘Restart’ notification. The BLE-device is indicating a restart by a LED light and a beep sound (if the volume is greater than 0).
- Press the back button to return to the device list and automatically disconnect the BLE-device.



Restart required



Sending restart

NOTE:

If the bonding process is initially skipped (e.g., by clicking 'cancel' on the bonding dialog that automatically appears after establishing the connection), and then an attempt is made to read a characteristic that requires bonding, the bonding dialog will be displayed again.

4.2.2 Value Guidelines

In the device screen you will see depending of you user permission the following **Services**: * **Device information** * **Battery Service** * **Settings** * **Button** * **Beacon** * **Mioty**

Each **Services** has its own **Characteristics**. The **Characteristics** of the **Services Device information** and

Battery Service are read only and do not require bonding in order to be read. Below are the **Characteristics** belonging to the other **Services** with the possible ranges of values and a brief description.

Settings

In the Settings service general settings of the device can be configured.

Name	Description	Possible Values
Password	PIN code that is used when pairing	Number of exactly 6 digits, no zeros allowed
Buzzer Intensity	Volume of the buzzer	0–100
Shelving State	Device in shelve mode (i.e. inactive)	`8` or `0` (unshelvemode = false) / `9` or `1` (unshelvemode = true)
Mems State	Use MEMS and set MEMS threshold from 3..15 (48 mg .. 1200 mg). Other values will be set to default 3. If modified, a restart of the device is required	Byte 0xXY is split in high and low nibble: X:3..15, Y:0=Inactive, 1=Active, with LEDs, 2=Active, without LEDs, 3=Active, Heartbeat after Scan Inactive Time, with LEDs, 4=Active, Heartbeat after Scan Inactive Time, without LEDs
MEMS Duration	Time for which the observer stays active after a MEMS interrupt (in seconds)	1–255

Button

In the Button service all settings related to button mode can be configured.

Name	Description	Possible Values
Button Mode	Button Operation Mode. Byte 0xXY split in high and low nibble: X controls filtering, Y controls scanning behavior	X: 0 = read all tags, 1 = ignore tags with own major Y: 0 = Inactive, 1 = Scan for 3min for one beacon after key push, 2 = Scan for 1 beacon (Button frame), 3 = Scan for 3 beacons (Scan frame)
Longpress Mode	Increase activation level with each second the key continues to be pressed down	bit 0: longpress_active (0 = inactive, 1 = active) bit 1: SCAN_RESULT_BLINK (0 = inactive, 1 = active) bit 2: CONTACT_TIMER_ENABLED (Vagno only) (0 = inactive, 1 = active)
Scan Active Time	Time for how long the scanning stays active (in seconds)	0 – 21840
Scan Inactive Time	Time for how long the scanning stays inactive (in seconds)	0 – 21840
Advertising Interval	Advertising interval (in ms)	20 – 10000
TxPower	TX Power used for Button Frames	0 = -21 dBm, 1 = -18 dBm, 2 = -15 dBm, 3 = -12 dBm, 4 = -9 dBm, 5 = -6 dBm, 6 = -3 dBm, 7 = 0 dBm, 8 = 1 dBm, 9 = 2 dBm, 10 = 3 dBm, 11 = 4 dBm, 12 = 5 dBm
UUID Filter	UUID Filter for beacons to be scanned	Array of 16 Bytes
Scan Rssi Thresholds	RSSI thresholding and delta configuration. Threshold is minimum RSSI required. Dynamic thresholding enabled if threshold = 0. Delta is the required RSSI difference between top 2 signals. Note: byte order is reversed in nRF Connect.	2 bytes (XXYY): XX = delta (1–255), YY = threshold (1–255); e.g., 0x0A32 = delta 10 dB, threshold -50 dBm; to achieve this in nRF Connect, enter hex value 0x320A

Beacon

In the Beacon services all settings related to iBeacon mode can be configured.

Name	Description	Possible Values
Beacon Mode	Activate/Deactivate iBeacon Frame. Byte 0xXY split in high and low nibble: X = battery byte setting, Y = activation mode.	X: 0 = attach battery byte, 1 = no battery byte Y: 0 = Inactive, 1 = Active, 2 = MEMS mode (1× mems duration in modes 1/2, 2× mems duration in modes 3/4)
UUID	iBeacon UUID	Array of 16 Bytes
Major	iBeacon Major ID	1 – 65535
Minor	iBeacon Minor ID	1 – 65535
Tx-Power	TX Power used for iBeacon Frames	0 = -21 dBm, 1 = -18 dBm, 2 = -15 dBm, 3 = -12 dBm, 4 = -9 dBm, 5 = -6 dBm, 6 = -3 dBm, 7 = 0 dBm, 8 = 1 dBm, 9 = 2 dBm, 10 = 3 dBm, 11 = 4 dBm, 12 = 5 dBm
Advertising Interval	Advertising interval in milliseconds	20 – 10000

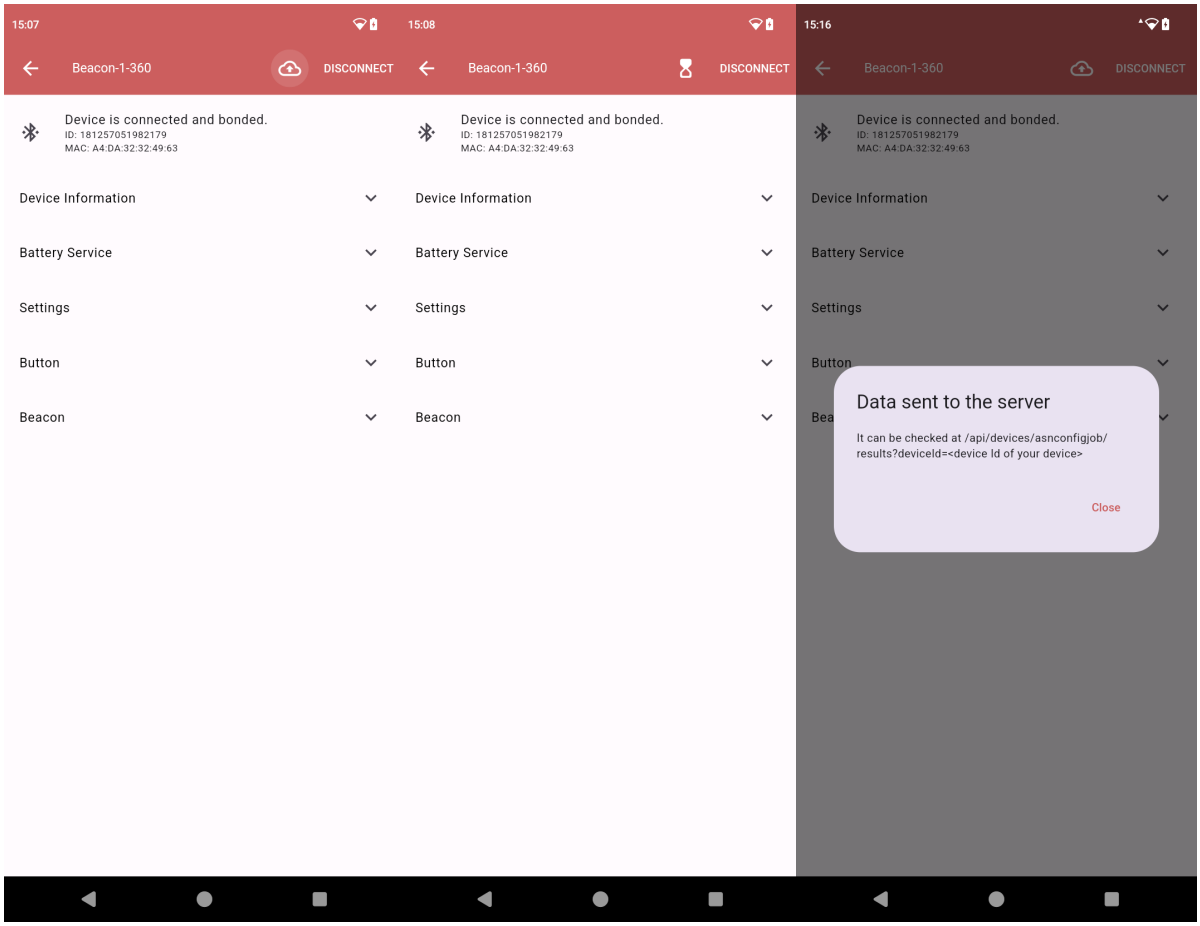
Mioty

Only available for Multitags

Name	Description	Possible Values
Mioty Mode	Defines Mioty operation mode.	0 = Off (no Mioty) 1 = Send all button/scanframes as Mioty data (3 beacons) at min interval; no BLE scan or buttonframes 2 = Send only social and POI beacons as Mioty data
Mioty Minimum Interval	Minimum interval for Mioty messages, in seconds.	Range: 15 – 21600 Any other value resets to default (60)
Mioty Profile	Regional Mioty configuration.	0 = eu0, 1 = eu1, 2 = eu2, 3 = us0
Mioty Uplink Pattern Group	Defines which uplink pattern group is used for Mioty transmission.	0 = upg1, 1 = upg2, 2 = upg3

4.3 Send device information to backend

- It is also possible to send the information of the BLE-device to the backend, via the cloud button on the top (button get enabled as soon as the device is connected) This functionality works only if the device is connected and bonded.
- The BLE-device has to exist in your backend account
- Using the cloud button without being connected or bonded to the device results in error message: “The device should be connected and bonded first”.
- Using the cloud button with a BLE device that does NOT exist in your backend account results in error message: “The data cannot be sent because the device doesn’t exist in the backend. Please create or import it.”
- Sending to server is indicated by an hourglass icon. The sending percentage will be shown near the hourglass icon.
- Once sending is complete, the icon will return to the cloud shape and a dialog will inform the user. Data can be checked at `/api/devices/asnconfigjob/results?uniqueId=<unique Id of your device>`



Cloud button	Sending process	Device info sent

- In case of problems when sending the characteristics related to the internet connection, an alert dialog will appear showing the message “Impossible to communicate with the server. Check your internet connection and retry!”
- One can check for updated characteristics in Track in the configuration window (ASNCTFUI) of the BLE-device.