Deliverable Number D44.3
Factsheet Number 03
Ambient Intelligence Module

Document Owner:	Giovanni Casella – Softeco
Contributors:	Paolo Giovene, Enrico Morten, Simona Stringa - Softeco
Dissemination:	Confidential
Contributing to:	WP44
Date:	31/03/2013
Revision:	1.0

List of Abbreviations

AMI	Ambient Intelligence

Table of Contents

1 AVAILABILITY AND CONTACTS 4

2 ARCHITECTURE AND FUNCTIONALITIES 4

3 TECHNICAL INFORMATION 6

3.1 SERVICE REQUIREMENTS 6
3.2 TECHNICAL DETAILS 6

4 LICENSING 7

4.1 LICENSE 7
4.2 THIRD PARTY LICENSES 7

5 TECHNICAL MANUAL 7

6 USER MANUAL 7

6.1 ACCESS 7
6.2 USER MANUAL 8

7 FUTURE PLANS 19

8 REFERENCES 20

1 Availability and Contacts

Version	Final version
Availability	*http://www.msee-ip.eu/intranet/sp4-workspace/wp44-msee-generic-mobile-business-platform/d44.3-mobile-platform-first-prototype/*
Accompanying specification and design document	*MSEE D44.1 – MSEE D44.3*
Source control	*http://engrep.eng.it/svn/msee/SP4/*
Contact person	*Giovanni Casella – Softeco Sismat* – giovanni.casella@softeco.it

2 Architecture and Functionalities

The main goal of the Ambient Intelligence (AMI) module is to enable the user to discover and identify the devices in his environment. A device is a smart product (i.e. a washing machine, a television, a temperature sensor) able to provide some information about itself and in some case able to communicate and interact with the ambient.

The following figure shows the user with some environments and devices.

For each identified devices the module provides several functionalities like to search for services and mobile applications related to the device. If the device has more interaction capabilities, additional features are available:

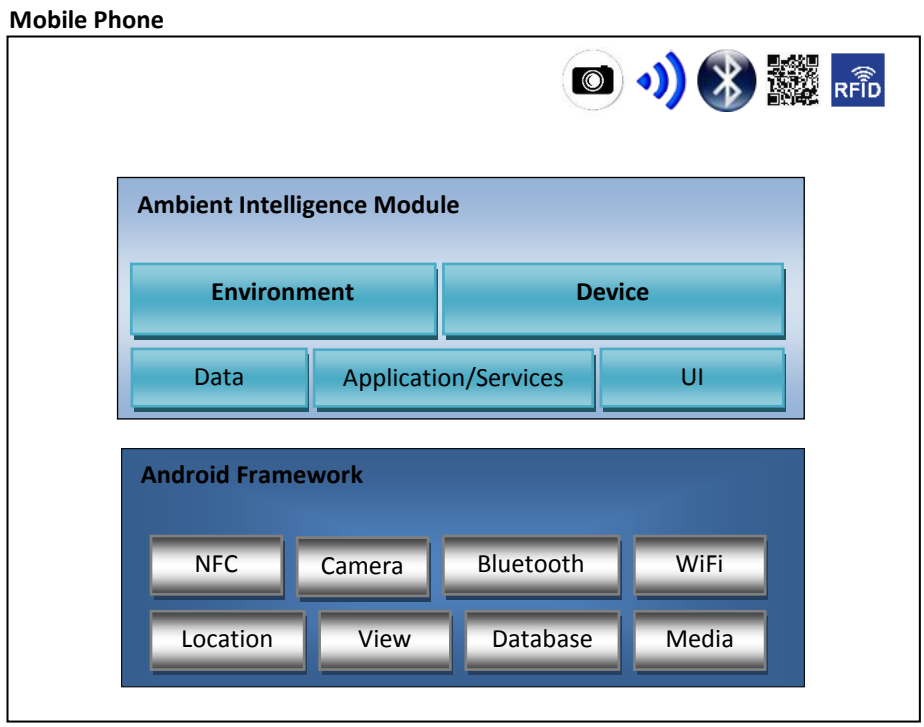
- Store/send device settings: it is possible to send to a device the settings retrieved by another device. Exploiting this functionality, for example, it is possible to set-up the

air conditioner in an hotel with the same settings of the air conditioner located at the home.

- Check Messages: it is possible to check if a device wants to send some messages to the user. Some devices, like sensors, are not equipped with proper displays or are located where the user cannot easily access them. Exploiting this functionality, for example, the user can check the current temperature in his room or if he needs to replace batteries in his home gas detector.
- Send Context: it is possible to send to a device the information regarding the other devices located in the same environment. Exploiting this information a device can modify its behaviour or interact with other devices directly. For example if a washing machine is aware that in the same environment there is also a dryer then can suggest to the user to increase the spin speed in order to save energy during the drying of clothes.

Devices are organized into environments. An environment is a physical place that can be identified by a geographical position and/or by other spatial attributes. Some examples of environments includes: an house, an office, the interior of the user car, Hilton hotels rooms, wagons of a train. The AMI module is able to store and recognize environments according to their features like the position and/or the presence of a Wi-Fi network and/or the presence of a particular bluetooth device (i.e. a particular phone hands-free car kit).

The following figure shows the internal high level architecture of the module:



Further details can be found in the document D44.1 [1].

In order to test it the module is provided with a Java program simulating a washing machine and a QRCode embedding the information regarding an industrial machine.
The QRCode is the following:

It is possible to generate some QRCodes using the free service provided at http://qrcode.kaywa.com/ (several QRCode generators are available on the web). The information embedded in the code must be a list of key-values encoded in json representing the properties identifying a product. The previous code embeds the following string:
{ "model": "ibarmia zv classic","manufacturer": "ibarmia machines","serial_code": "882IIRP12"}

3 Technical Information

3.1 Service requirements

In order to use the module it is required:

- Mobile Phone or Tablet
  - Including Android Operating System (Android 2.2 or higher)
- PC (only for the machine simulator)
  - Including Windows 7, Java JRE 7 and bluetooth card

3.2 Technical details

Nature	**Android Application**
Programming Language	Java
Development Framework	Eclipse Juno
Additional libraries	-
Application Server	-
Database	Internal SQLLite Database

4 Licensing

4.1 License

Proprietary licensing scheme.

4.2 Third party licenses

The Machine Simulator, provided only for testing purposes, includes:

GSON	Apache License 2.0
JSON-RPC	MIT License
BlueCove bluetooth library	Apache License 2.0

5 Technical Manual

The distribution file of the AMI Module can be installed in any Android phone equipped with Android 2.2 or higher. In order to install the application the installation file must be transferred to the mobile device by bluetooth or by USB cable or by email. Once the apk installation file is on the phone it is possible to execute it starting the Android installation process. Please refer to the mobile device manual for further information.

6 User Manual

6.1 Access

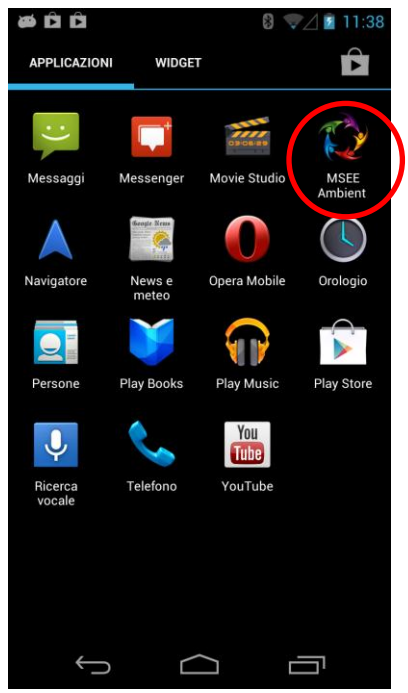
The software component is available for download at:

http://www.msee-ip.eu/intranet/sp4-workspace/wp44-msee-generic-mobile-business-platform/d44.3-mobile-platform-first-prototype/

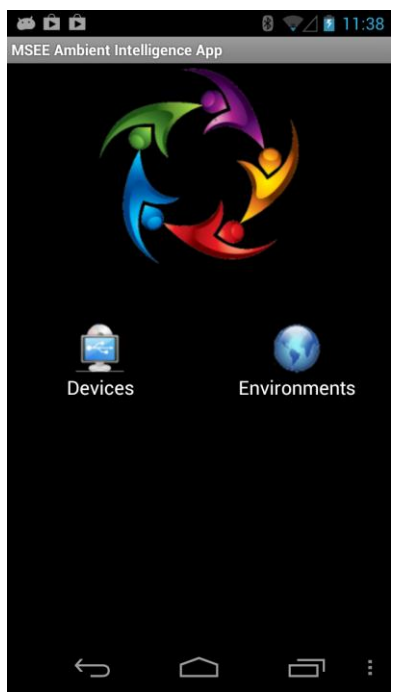
6.2 User Manual

Once the application has been installed it appears in the application list of the phones.

Application Launch

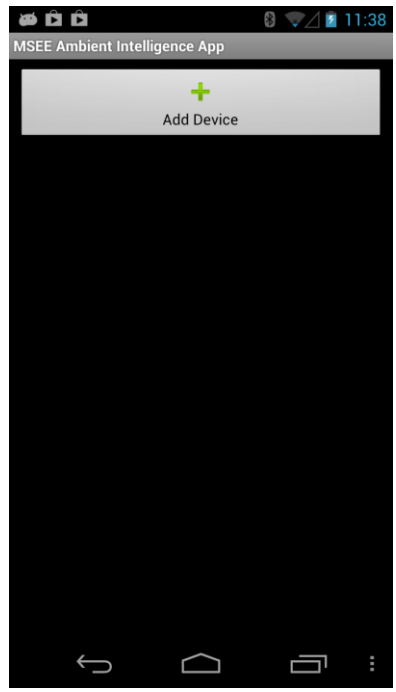


The first screen of the application enables the user to choose if he wants to manage his devices or his environments.

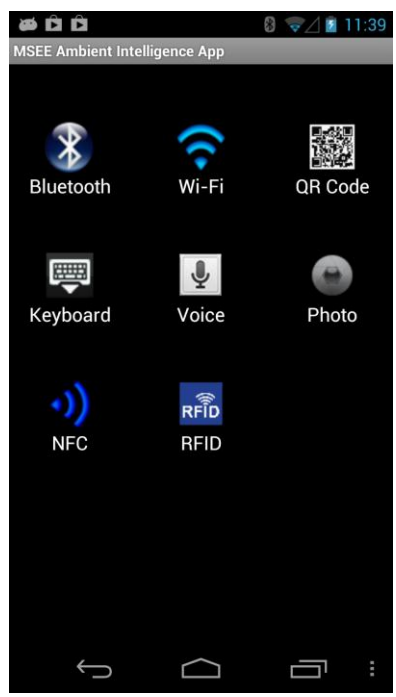


Device Addition

Pressing the devices button it is possible to access the device list.



The device list is initially empty and contains only the button to add new devices.
It is possible to add a device exploiting several methods.



The current version of the software supports:

- Bluetooth: the device to add must be equipped with a bluetooth card and must be able to interact with the AMI module according a precise protocol (further information can be found in D44.1 [1]). The provided Machine Simulator application supports the protocol and can be used for testing purposes.

- QRCode: the device identity information are reported on a QRCode (usually attached on the device).
- Keyboard: the user can add the information regarding the device using the mobile device keyboard (this is useful for legacy devices).

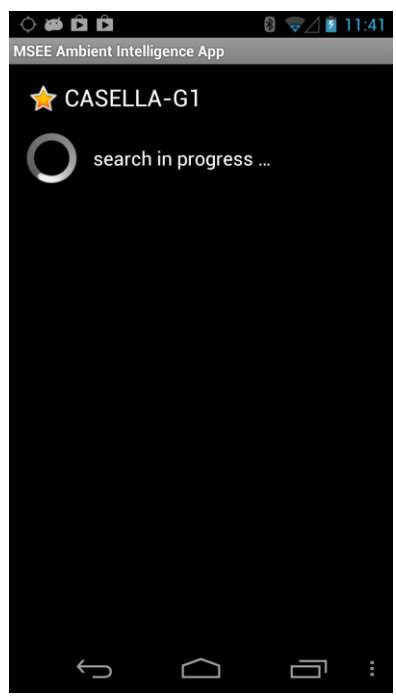
Identity information retrieved when the device is added are used later to find applications and services related to the product.

Device Addition via Bluetooth

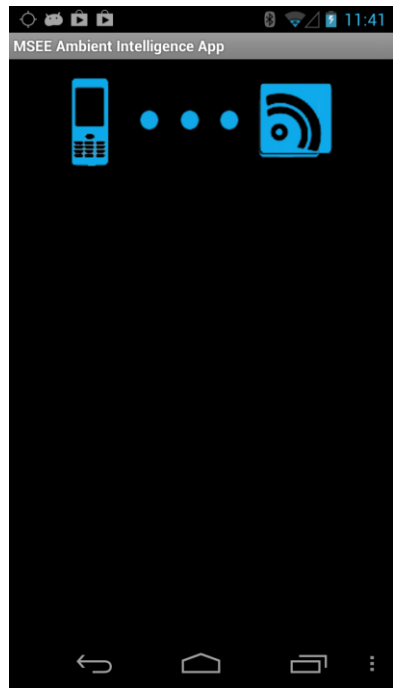
Note: to test this feature you can use the provided AMIMachineSimulator. To start the application execute the file machinesimulator.jar . Before starting the application activate the bluetooth card and check that the PC is “searchable” by other bluetooth devices.

If when this option is selected the mobile device bluetooth is disabled a dialog will enable the user to activate it. The application will search for all “searchable” bluetooth devices. Once a device is selected the bluetooth communication is started.

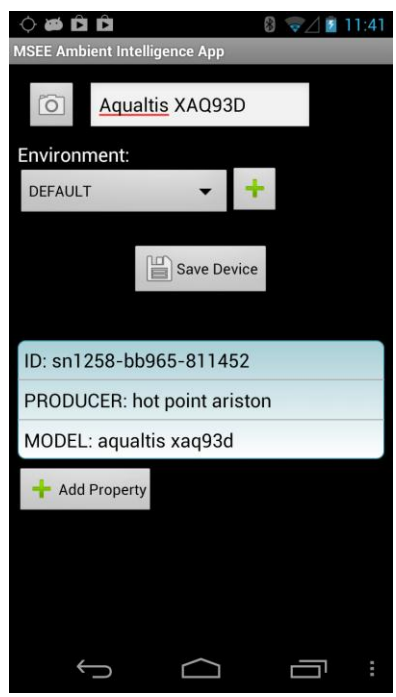
In the screenshot the PC running the simulator is found.



Once a device has been selected the protocol to identify the device is started by the application.

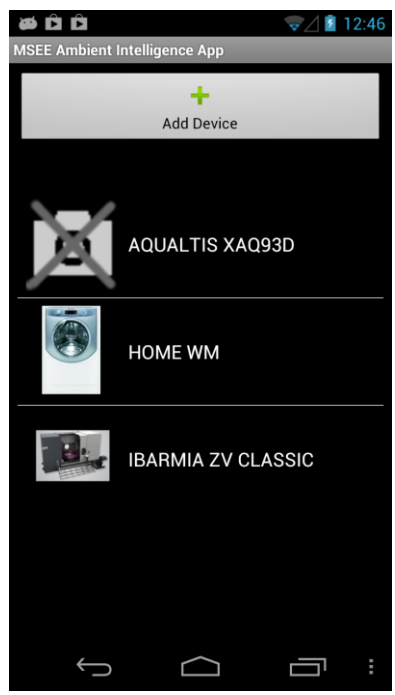


After few seconds the application shows the device identified.



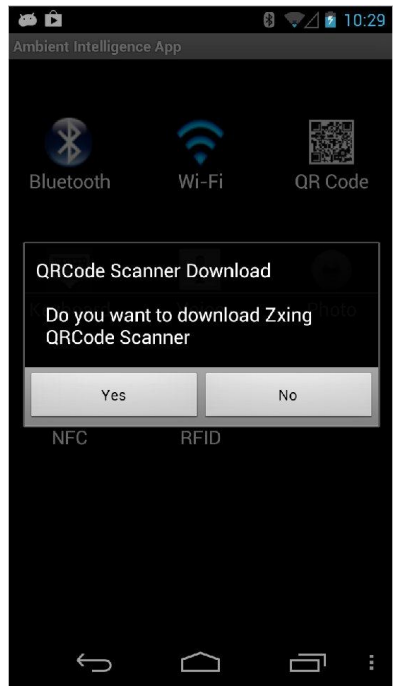
The user can assign a descriptive name to the device, can take a photo of the device and can add some properties manually.
The device must be associated to an environment. A default environment is available, the user can press the "plus" button to add other environments.
The following figure shows the application after the definition of a "home" environment and after the addition of the device name and picture.

After the addition of some devices the device list displays them as in the following figure.

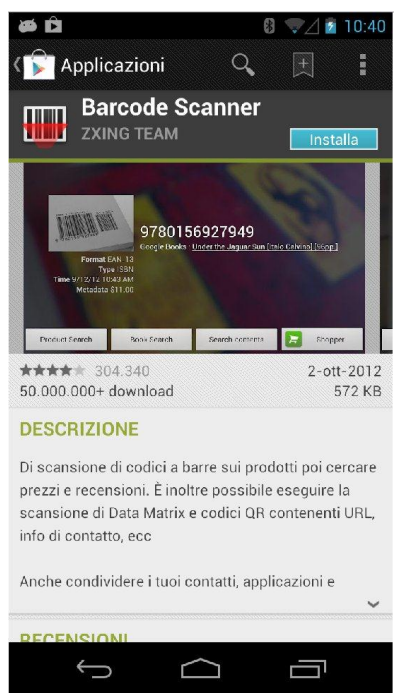


Device addition using QRCode

In order to identify and add a device scanning a QRCode the free open-source Barcode Scanner[1] provided by the zxing team must be installed on the mobile device. If it is not already installed the AMI application requests to the user to download it from the Android Market.

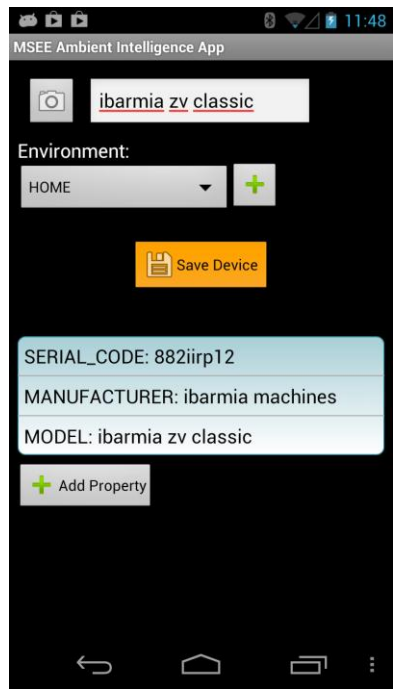


If the user accepts the request he will be driven in the installation process.



[1] Zxing Barcode Scanner: https://play.google.com/store/apps/details?id=com.google.zxing.client.android

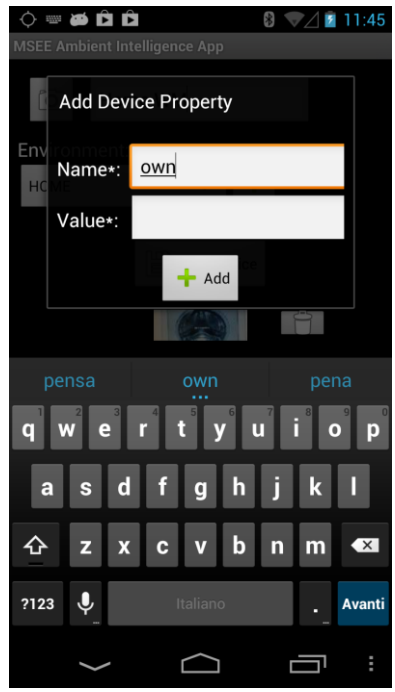
Once the barcode scanner is installed the AMI module exploits it internally (in a transparent way for the user) to decode QRCodes. The information retrieved scanning the code are presented into the Add Device screen.



The previous picture reports the device retrieved scanning the QRCode provided in this user manual.

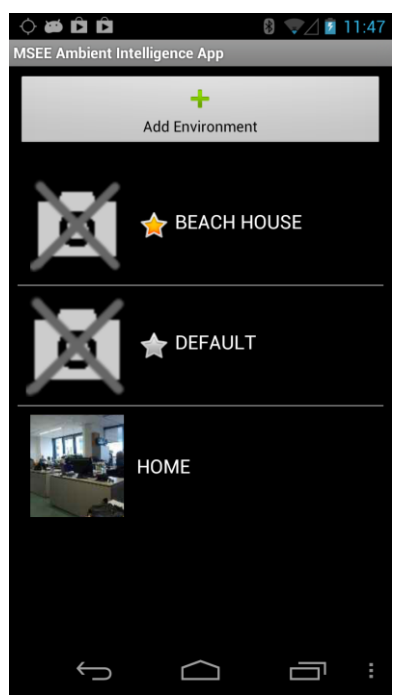
Device addition using the keyboard

Despite some devices are not able to communicate and there is not any QRCode associated to them, it can be useful to add such devices in order to search for services and applications. The user can add the information regarding these devices using the mobile device (virtual) keyboard.



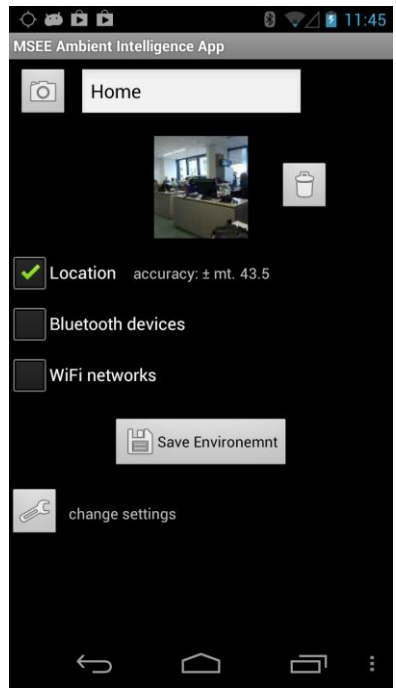
Environment Addition

Starting from the home screen it is possible to access the environment list.



The previous figure shows the screen reporting the environment list with the button to add new environments.

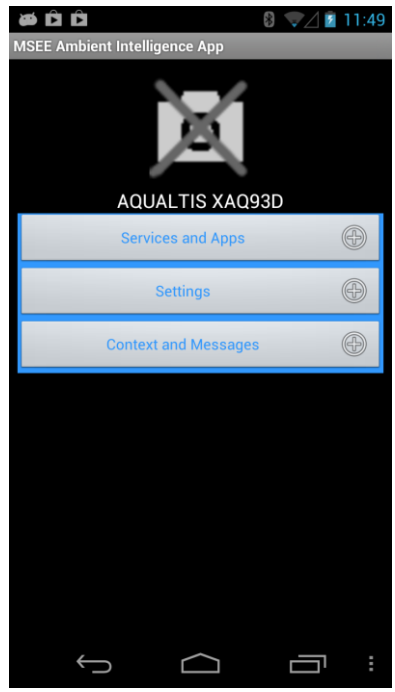
The yellow star highlights the current environment (recognized automatically by the application) while the gray star denotes the "default" environment. Selecting an environment it is possible to view the devices associated to it.
Pressing the "Add Environment" button the user can add new environments.



The user can define the environment name, take a photo of the environment and decide on which bases the environment will be recognized. The "change settings" button enables the user to access the phone settings in order to activate the bluetooth and/or the WiFi if they are required.

Device Action

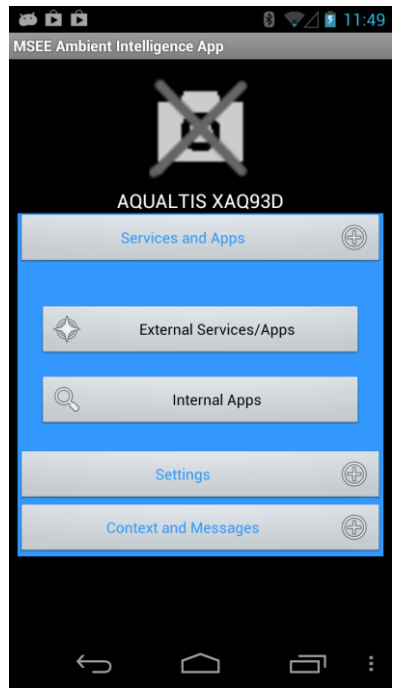
Selecting a device it is possible to open the device action selection screen.



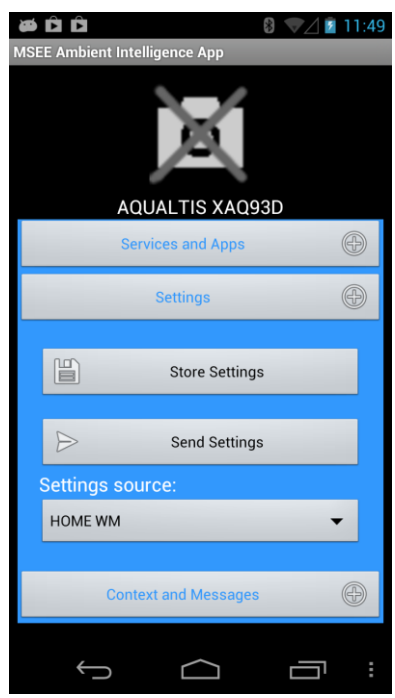
The interface is split in three parts namely:

- Services and Apps: enables to search for services and applications related to the device.
- Settings: enables to request/send settings to the device.
- Context and Messages: enables to send to the device information about the other devices in the same environment and to check if the device has some messages for the user.

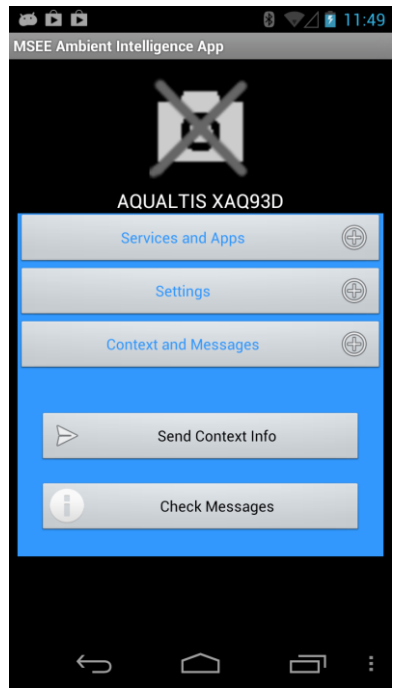
Exploiting the “External Services/Apps” button the user can access the Mobile Delivery module to find services and apps registered on the MSEE Delivery Platform and on other sources (i.e. Google Play). Exploiting the “Internal Apps” button the user can search for applications related to the product already installed on the mobile device.



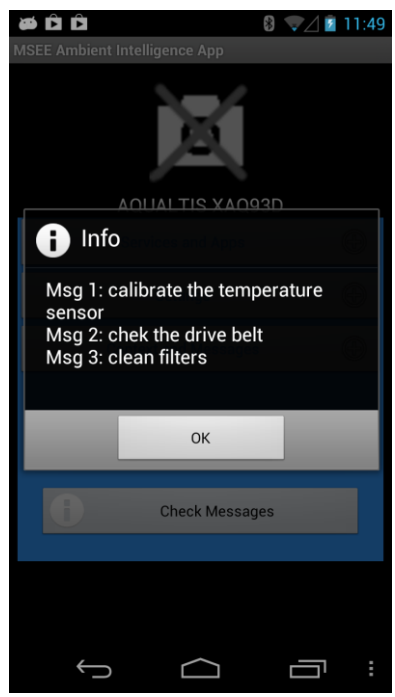
The "Store Settings" button can be used to store on the mobile phone the settings of the device. In this way it will be possible to export these settings to another device.
The "Send Settings" button can be used to send to the device the settings of another device that is possible to select exploiting the "Settings source" list box. The settings buttons are available only if the device is able to communicate.



The "Send Context" button can be used to send to the device the information regarding the other device registered in the same environment.



Finally the "Check Messages" button can be used to request messages to the device. Received messages (if any) are shown on a modal window.



The "Send Context" and "Check Messages" buttons are available only if the device is able to communicate.

7 Future Plans

The prototype will be integrated and used in the framework of the MSEE IT System [3]. Additional developments will be considered during the specification of the final prototype of the MSEE Mobile Business Platform [2].

8 References

[1] D44.1 Mobile Business Platform Specification and Architecture
[2] D44.2 Mobile Business Platform Specification and Architecture (final)
[3] D45.1 MSEE Service-System Integrated