Set of sample modules/applications that operationalize, for all architecture layers, complex functionalities that can be run on different types of objects as well as on an avatar

Deliverable 5.4 (version 1)

Nicolas LE SOMMER

16 September 2017

Project ASAWoO

Adaptive Supervision of Avatar / Object Links for the Web of Objects

Grant Agreement: ANR-13-INFR-0012-04

Abstract

This document presents a set of modules and applications we have developed in project ASAWoO to evaluate our solution in real conditions, as wells to provide a demonstration and code examples meant for developers.

Contents

1 Introduction

The deliverable describing the architecture of the ASAWoO platform should be read before this one in order to understand certain concepts (avatar, platform, functionalities, WoTApp, etc.).
The pieces of code of the functionalities and of the WoT applications described in this document are available on the GIT repository of the ASAWoO project (https://asawoo.gitlab.io/).
The Web of Thing (WoT) application, we have developed for demonstration purposes and we have tested in real conditions, is an application involving an autonomous mobile robot performing both a video surveillance and a collection of environmental data during its patrol. This application relies on a set of functionalities allowing the robot to achieve its navigation, to collect environmental data from sensors, and to perform a video surveillance and to detect persons in the video stream. Other applications have been also developed: one showing the data collected by the robot, one showing the video stream and the persons that have been detected and an application allowing to administrate the physical objects and to deploy WoT applications on them.
These functionalities and applications are presented in the remainder of this document.

2 Functionalities

According to the capacities exhibited by the physical object, some functionalities can be deployed on them. These functionalities are expected to be used either by WoT applications or by remote objects. We describes below the navigation, data collection, video surveillance and person detection, go to base station and patrol surveillance functionalities.

2.1 Navigation

The navigation functionality is designed to be used by an unmanned ground vehicle that must follow a predefined path. Two implementation of this functionality have been developed: one based on an odometry approach to estimate the position changes over the time, and another one based on a GPS.

2.2 Data collection

The data collection functionality is composed of two parts: a part that must be deployed on a sensor, and another part that must be installed on the mobile object that must collect the data. After being deployed on the sensor, the sensing functionality is registered in the service registry of the avatar of the sensor. The content of this registry is advertised in the network so as to be discovered by other avatars. After discovering the sensing functionality, the data collection functionality installed on the avatar of the mobile object invoke this functionality in order to obtain the value provided by the sensor. This data collection functionality stores the data locally. These data can be read by a dedicated WoT application in order to be presented to end users.

2.3 Camera

The camera functionality allows to exploit a local camera, and to and to expose a video stream at a given URL. This functionality can be exploited by a WoT application in order to show this video stream to a user,

or by high level functionalities such as person detection.

2.4 Person Detection

As indicated by its name, The person detection functionality makes it possible to detect people in a video stream. This functionality relies on the camera functionality.

2.5 Ping Notification

The Ping Notification functionalities makes it possible to notify a remote device of an event. This notification is achieved by sending a HTTP POST request to a specific URL (i.e. /ping).

2.6 Video surveillance

The Video Surveillance functionality is a composite functionality. It relies both on the Person Detection functionality and on the Ping Notification functionality. When a person is detected in the video stream provided by the Camera functionality, a notification is sent to a given remote device.

2.7 Go to base station

The ”Go to base station” functionality is a functionality allowing an unmanned ground vehicle (UGV) to return to its base station. This functionality preempts the Navigation functionality. It can be used for example to return to the base station when the power budget of the UGV is low.

2.8 Patrol Surveillance

The Patrol Surveillance functionality is a composite functionality based on the ”Go to base station”, Video surveillance and Data collection functionalities. This functionality makes it possible to collect data provided by sensors while performing a video surveillance. When a person is detected during the patrol, the UGV returns to its base station to notify a remote application of this event. The UGV also delivers the data its has collected during its patrol when it reaches its base station.

3 WoT Applications

This section presents the WoT applications we have developed to conduct our experiment in real conditions, as well as to provide a demonstration.

3.1 Administration

The administration application, whose a screenshot is given in Figures 1,2 et 3 makes it possible to configure a device, to discover remote ASAWoO platforms, to install and uninstall WoT applications on the avatar of the considered device, to browse the service registry of the avatar, as well as to invoke services using the API they exhibit (implementation of the OpenAPI specification).

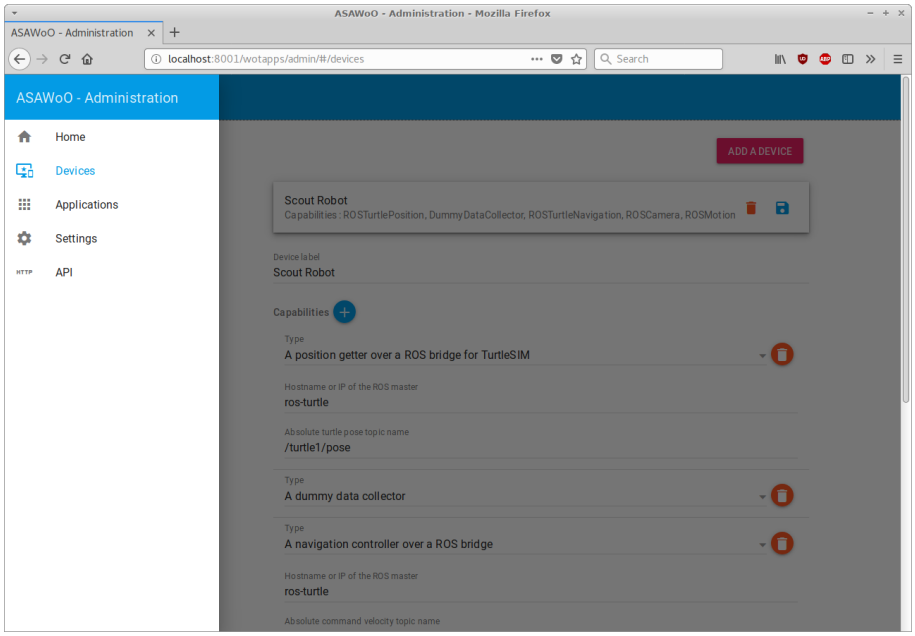

Figure 1: User interface of the administration application.

3.2 Data collection

The data collection application presents the data collected from the sensors as a graph. The user interface of this application is given in Figure .
This application relies on the data collection functionality, and uses the API provided by this functionality.

3.3 Patrol surveillance

The patrol surveillance application provides a view of the video stream, the images in which a person has been detected, the mean to specify the path the unmanned ground vehicle (UGV) must follow, and the mean to generate an event forcing the UGV to return to its base. A capture of the user interface of this application is given in Figure 5.

4 Experimentation

We have evaluated the above-presented functionalities and applications in real conditions using a Husky unmanned ground vehicle (UGV)[1] and sensors based on Raspberry devices. This experiment consisted of collecting data provided by Raspberry devices using the Husky UGV while performing a video surveillance. Some photos of this experiment are given in Figure 6. The Husky UGV runs a Linux and ROS based system. This UGV has been controlled by the ASAWoO platform during this experiment. To do so, bindings between the ROS and the ASAWoO platform have been developed.

[1] https://www.clearpathrobotics.com/husky-unmanned-ground-vehicle-robot/

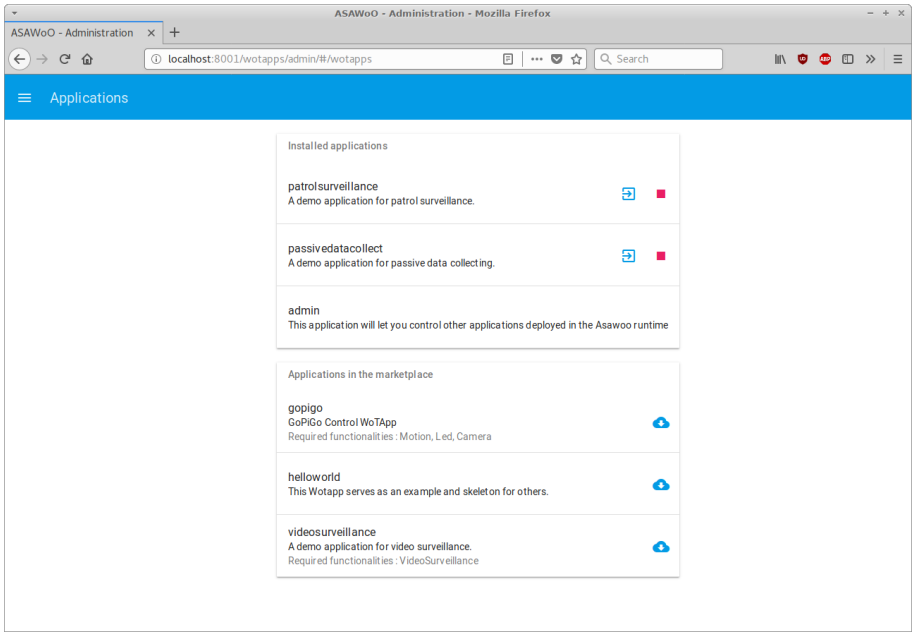

Figure 2: User interface of the administration application.

5 Demonstration

We have reproduced our experiment in a demonstration using several devices running in different Docker containers. This demonstration is available in the GIT repository of the ASAWoO project, and can be run by the users of the ASAWoO platform or by the developers. Figure 7 shows the user interface of the demonstration we have developed.

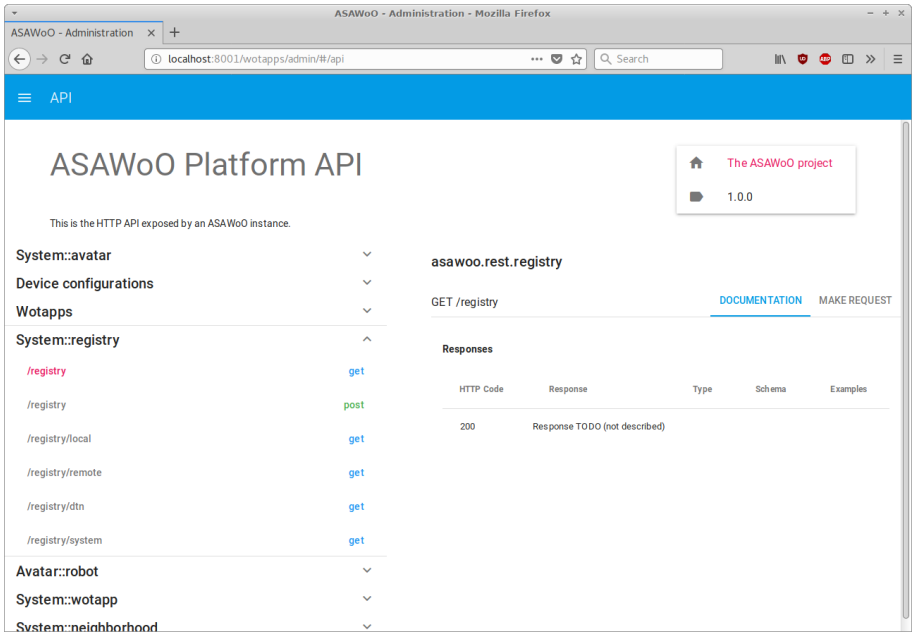

Figure 3: User interface of the administration application.

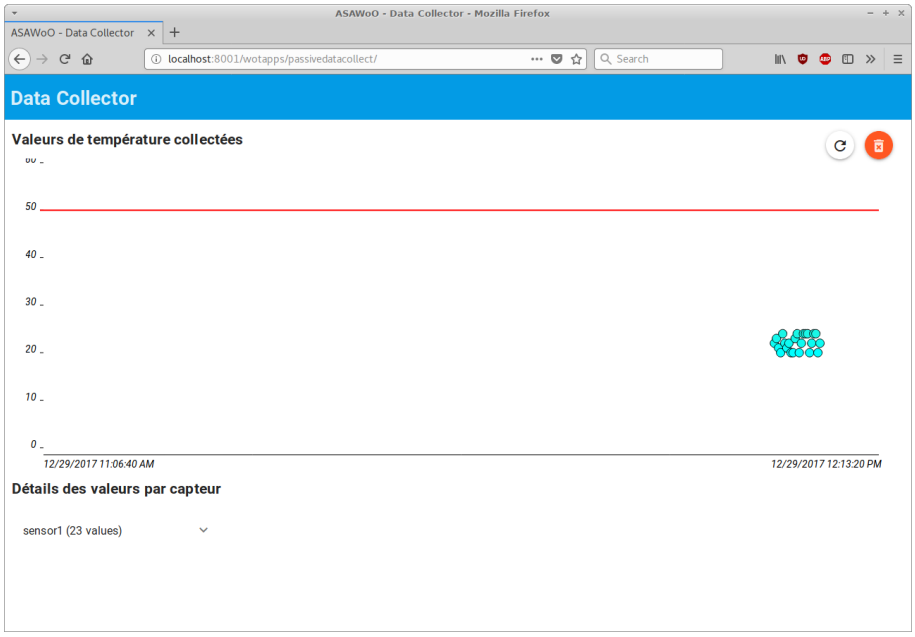

Figure 4: User interface of the data collection application.

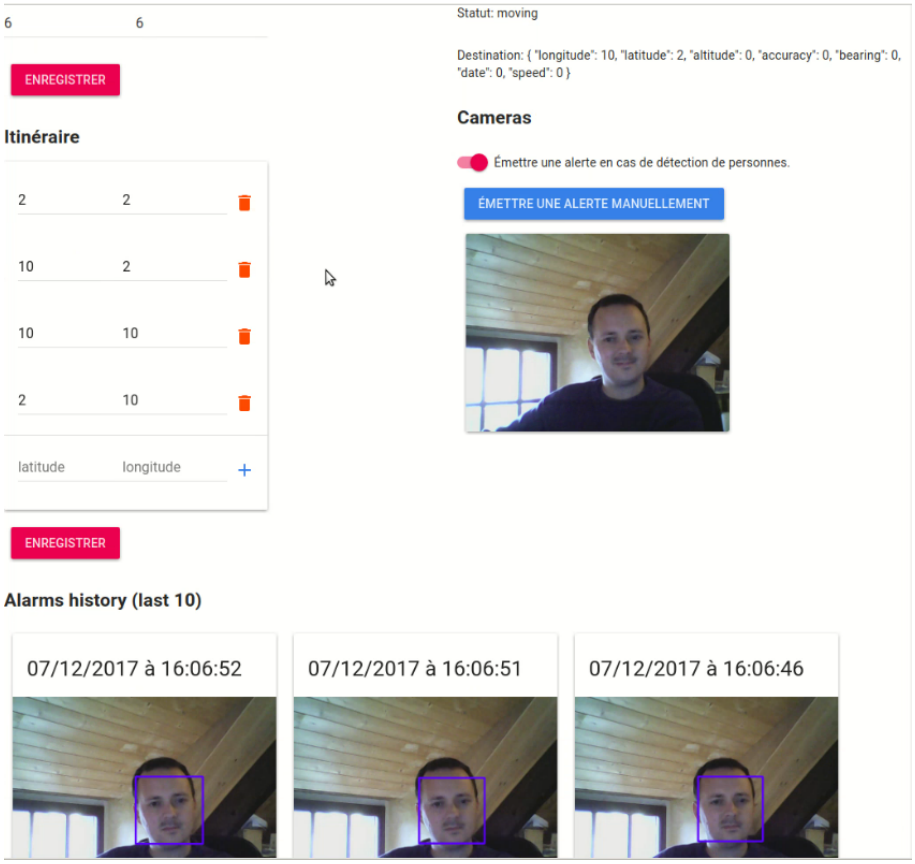

Figure 5: User interface of the patrol surveillance application.

Figure 6: Photos of the ASAWoO experiment.

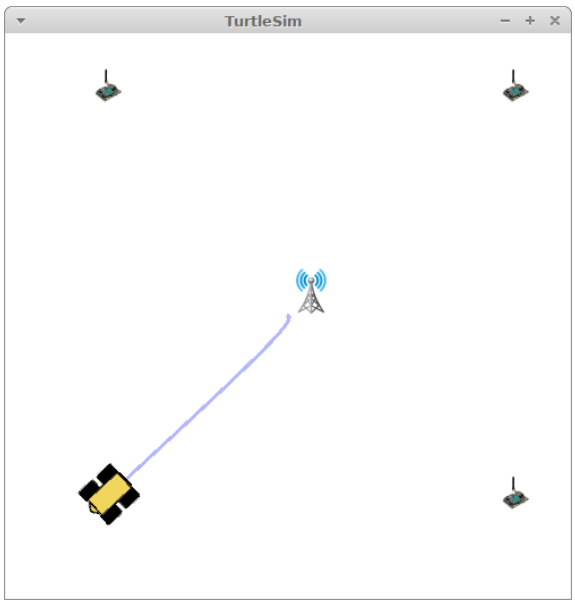

Figure 7: Capture of the user interface of the demonstration.