ROBOTS IN ASSISTED LIVING ENVIRONMENTS

UNOBTRUSIVE, EFFICIENT, RELIABLE AND MODULAR SOLUTIONS FOR INDEPENDENT AGEING

Research Innovation Action

Project Number: 643892 Start Date of Project: 01/04/2015 Duration: 36 months

DELIVERABLE 4.6

Integrated Robotic Platform I

Dissemination Level	**Public**
Due Date of Deliverable	Project Month 15, 1 July 2016
Actual Submission Date	12 April 2017
Work Package	WP4: *Physical home architecture development and integration of cost-effective, reliable and power-efficient RADIO components for elder monitoring and caring*
Task	T4.3: *Robotic platform design and integration*
Lead Beneficiary	ROBOTNIK
Contributing beneficiaries	NCSR-D, TWG, RUB
Type	DEM
Status	Submitted
Version	Final

Project funded by the European Union's Horizon 2020 Research and Innovation Actions

Abstract

This deliverable reports on the RADIO robotic platform first design, corresponding to the work done at Task 4.3 during months 7 to 15. A second iteration of this document will be performed during months 19 to 30, when the fully functional robot prototype will be delivered. The deliverable is comprised by two main chapters, Chapter 2 documenting and the hardware design of the RADIO robotic platform and Chapter 3 documenting the software developed.

History and Contributors

Ver	Date	Description	Contributors
00	9 June 2016	Document structure	NCSR-D
01	6 July 2016	Hardware platform description	ROBOTNIK
02	7 July 2016	Internal peer review	RUB
03	11 July 2016	Addresses peer review comments	ROBOTNIK
04	15 Mar 2017	RADIO Home deployment and demonstration	NCSR-D, TWG, ROBOTNIK, S&C
05	5 Apr 2017	RADIO Home demonstration video	NCSR-D, ROBOTNIK
06	11 Apr 2017	List of software components	NCSR-D, ROBOTNIK, TWG
07	11 Apr 2017	Internal review comments and corrections	TWG
Fin	11 Apr 2017	Final preparations and submission.	NCSR-D

Abbreviations and Acronyms

NCSR-D	National Centre for Scientific Research “Demokritos”
TWG	Technical Educational Institute of Western Greece
RUB	Ruhr Universitaet Bochum
ROBOTNIK	Robotnik Automation SLL
S&C	Sensing & Control Systems S.L.
AVN	AVN Innovative Technology Solutions Ltd.
FSL	Fondazione Santa Lucia
FHAG	Fundació Hospital Asil de Granollers
FZ	Frontida Zois
ADL	Activities of Daily Life
BLE	Bluetooth Low Energy
CPS	Cyber-Physical Systems
ICT	Information and Communications Technology
IPR	Intellectual Property Rights
RTD	Research, Technological Development
SME	Small Medium Enterprise
ROS	Robot Operating System

CONTENTS

Contents iii
List of Tables iv
List of Figures v
1 Introduction 1
1.1 Purpose and Scope 1
1.2 Approach 1
1.3 Relation to other Work Packages and Deliverables 1
2 Prototype Hardware 2
2.1 Robotic Base 2
2.2 Attached Devices 2
3 Prototype Software 7
3.1 Core Packages 7
3.2 Robot Perception Packages 8
3.3 Robot Behaviour and RADIO Home Integration 8
4 Demonstration 9
4.1 Simulated Demonstration in FSL Environment 9
4.2 Physical Demonstration in FHAG Environment 9

LIST OF TABLES

Table 1: Standard technical specifications of the TurtleBot2 base 3
Table 2: ORBBEC Technical Specifications 3
Table 3: Xtion Technical Specifications 3
Table 4: Hokuyo Technical specifications 5

LIST OF FIGURES

Figure 1: Relation to other Work Packages and Deliverables 1
Figure 2: TurtleBot2 robotic platform 2
Figure 3: Telescopic bar and cameras 4
Figure 4: Hokuyo assembly 5
Figure 5: Radio robot model in ROS 9
Figure 6: FSL floorplan 10
Figure 7: Building conversion for simulation in Gazebo 10
Figure 8: RADIO robot simulation on FSL building 11
Figure 9: RADIO navigation tests in the FSL premises 11

1 INTRODUCTION

1.1 Purpose and Scope

This deliverable demonstrates the first RADIO Robot prototype, including the hardware design and the integration of the ADL recognition prototypes developed in WP3. Within the scope of this deliverable is to demonstrate the hardware prototype to publish the source code of the software developed for the robot, besides the ADL recognition software.

1.2 Approach

This deliverable is prepared within *Task 4.3: Robotic platform design and integration*. This task covered all aspects of hardware design and integration, developed drivers and controllers for the hardware components, and handled integration issues from deploying D3.9/D3.10 on the physical system.

1.3 Relation to other Work Packages and Deliverables

This deliverable integrates D3.9 and D3.10, following the *Architecture for extending smart homes with robotic platforms I* (D4.1). The resulting prototype is used for the first prototype of the integrated RADIO Home (D4.8). This prototype is also the basis for the development of the second RADIO Robot prototype (D4.7, due M30).

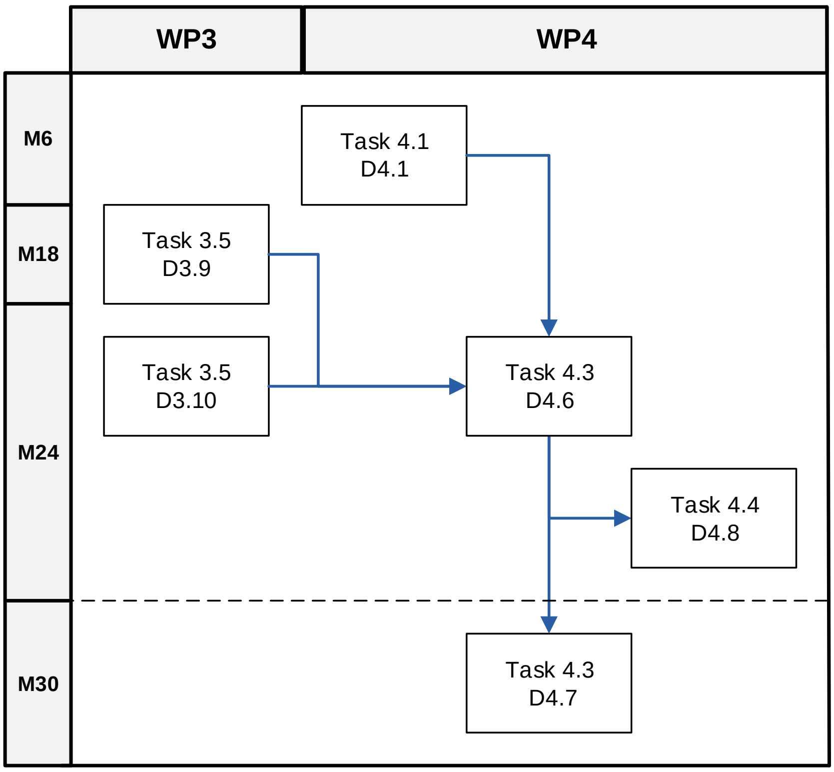

Figure 1: Relation to other Work Packages and Deliverables

2 PROTOTYPE HARDWARE

2.1 Robotic Base

The starting point for the RADIO robotic platform is the TurtleBot2 mobile robot. TurtleBot2 is an open robotic platform designed for education and research on state of art robotics. It is also a powerful tool to teach and learn ROS and make the most of its cutting edge technology. Equipped with a 3D sensor, it can map and navigate indoor environments. Its highly accurate odometry, amended by a factory-calibrated gyroscope, enables precise navigation.

TurtleBot 2 (Kobuki base) is the new version of the successful platform TurtleBot and it has many advantages over its predecessor: odometric measurement precision, open protocol, greater autonomy, greater load, higher speed, and greater mobility, larger diameter wheels and capacity to overcome obstacles up to 12 mm. Figure 2 presents the TurtleBot2 and Table 1 gives its standard specifications.

2.2 Attached Devices

In order to accommodate all the hardware needs of the RADIO project, the following devices have been attached to the robotic platform:

- Telescopic bar
- Orbbec / Asus Xtion camera
- Hokuyo URG-04LX laser
- PC NUC C5CPYH + 19VDC battery

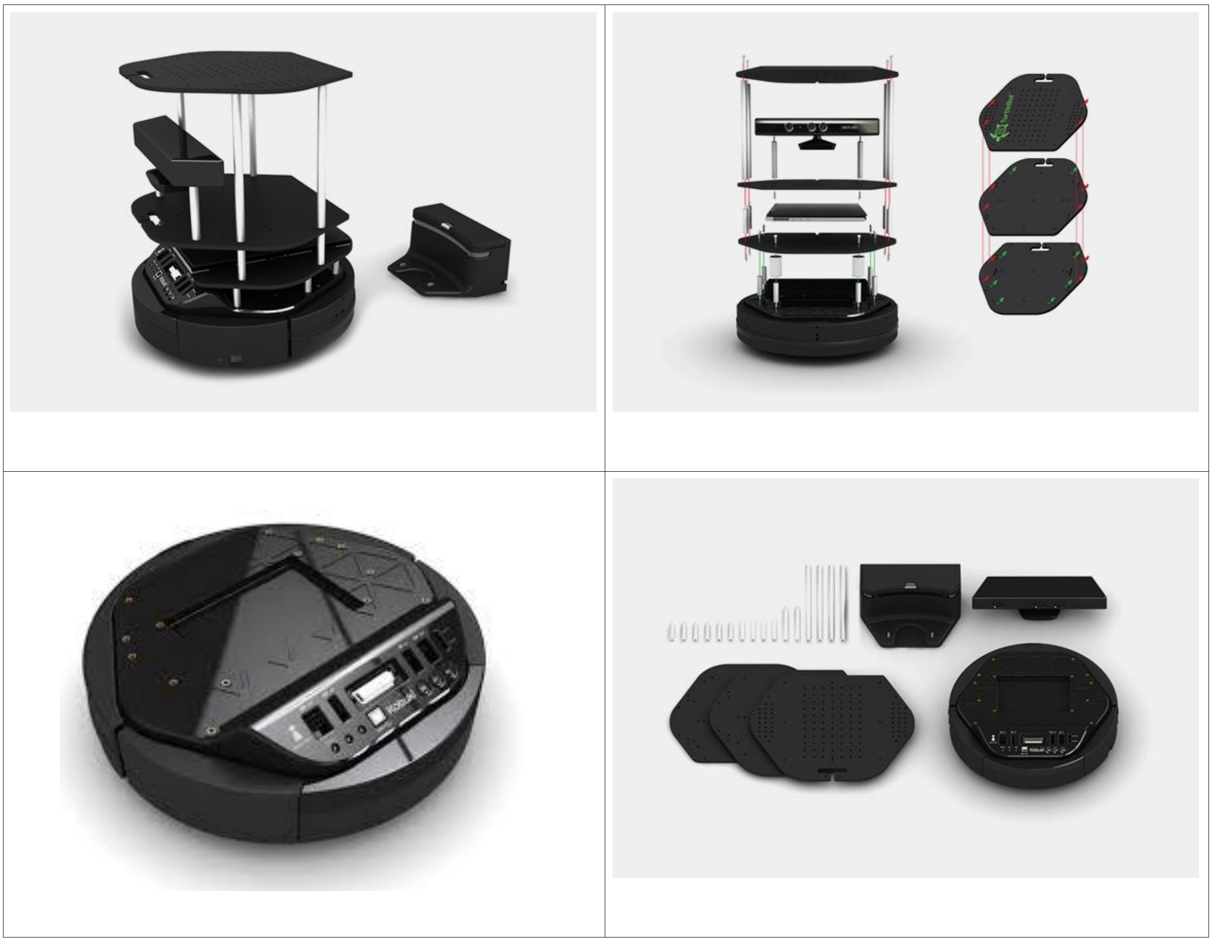

Figure 2: TurtleBot2 robotic platform

The Telescopic bar is the one supporting the Xtion or Orbbec camera that is mounted on the top. It permits 4 different positions for the camera, at 60.7 cm, 86.7 cm, 113.5 cm, and 138.4 cm.

The Orbbec and the Xtion are RGBD cameras providing the capability to sense the environment in the visual and the 3D modality. Their specifications given in Tables 2 and 3.

.

Table 1: Standard technical specifications of the TurtleBot2 base	
Dimensions	31.5 x 43 x 34.7 cm
Weight	5 kg, also depending on configuaration
Load capacity	5 kg
Speed	0.65 m/s
Controller	Open architecture ROS
Autonomy	7h. (big battery), 3h. (small battery)

Table 2: ORBBEC Technical Specifications	
Power	5V (USB 2.0)
Range	0.4 - 8 m
Depth Image Size	640*480 (VGA) 16bit @ 30FPS
RGB Image Size	1280*960 @ 10FPS
Data Interface	USB 2.0
Microphones	2
Weight	300 g
Software	Orbbec Astra SDK + Openni (ROS compatible)

Table 3: Xtion Technical Specifications	
Field of view	Horizontal field of view: 58 degrees Vertical field of view: 45 degrees Diagonal field of view: 70 degrees Depth sensor range: 0.8m – 3.5m (only indoor or outdoor outside direct sunlight)
Data streams	320x240 QVGA @ 60 frames/sec 640x480 VGA @ 30 frames/sec The graphic display uses a SXGA resolution (1280x1024)
Sensor	RGB & Depth & Microphone

	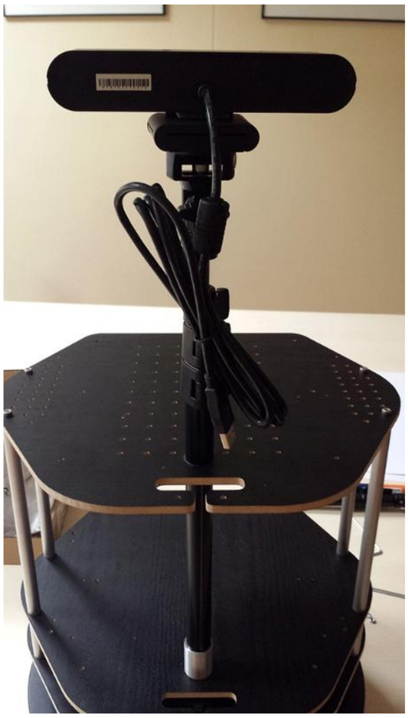
Asus Xtion camera	Orbbec camera

Figure 3: Telescopic bar and cameras

The bar is mounted on the middle and top plates of the TurtleBot and the next parts are needed for its assembly:

- Telescopic bar adapter
- Hexagonal Head M6 Screw
- M6 washer
- Middle plate with M6 hole
- Allen M6

Figure 3 shows the assembly of the telescopic bar and the cameras onto it.

Next device attached to the robotic platform is the Hokuyo URG-04LX laser scanner. It is a high accuracy, high resolution and wide angle sensor which provides a good for autonomous robots moving in an unknown environment. Table 4 shows its technical specifications.

Table 4: Hokuyo Technical specifications	
Power source	5 VDC ± 5% (USB Bus Power)
Light source	Semiconductor laser diode(λ=785nm), Laser safety class 1
Measuring area	60 to 4095mm(white paper with 70mm), 240°
Accuracy	60 to 1,000mm : ±10mm, 1,000 to 4,095mm : 1% of measurement
Angular resolution	Step angle : approx. 0.36°(360°/1,024 steps)
Repeatability	60 to 1,000mm : ±10mm
Scanning time	100ms/scan
Noise	25dB or less
Interface	USB, RS-232C(19.2k, 57.6k, 115.2k, 250k, 500k, 750kbps), NPN open-collector (synchronous output of optical scanner : 1 pce)
Command System	SCIP Ver.2.0
Ambient illuminance	Halogen/mercury lamp: 10,000Lux or less, Florescent: 6000Lux(Max)
Ambient temperature/humidity	-10 to +50 degrees C, 85% or less (Not condensing, not icing)
Vibration resistance	10 to 55Hz, double amplitude 1.5mm each 2 hour in X, Y and Z directions
Impact resistance	$196 m/s^2$, Each 10 time in X, Y and Z directions
Weight	Approx. 160 g

Figure 4: Hokuyo assembly

The Hokuyo is mounted in the middle plate of the TurtleBot2, and the following parts are needed to assemble it:

- Metallic extension plate
- 3 M3 bolts
- 3 M3 bolts
- 3 M3 nut
- 2 M2 countersunk head screw

Figure 4 shows the assembly of the Hokuyo laser sensor.

The last attached special device is the NUC C5CPYH PC in the bottom plate. It is a very powerful mini PC able to provide extended processing capabilities. It is composed by an Intel Celeron N3050 Dual Core Processor (1.6 - 2.16GHz), 64-bit and Intel HD Graphics (320 - 600MHz). It supports up to 8GB of memory (DDR3L; 1.35V; 1333MHz Minimum) in 1 Slot and a 2.5" SATA Drive (6.0Gbps MAX.), including Intel 802.11ac wireless and Bluetooth 4.0.
The PC is powered by an extra 19VDC battery that has been integrated into the robot.

3 Prototype Software

All the developed software is integrated on the ROS middleware (www.ros.org).

The project has registered a Github organization (https://github.com/RADIO-PROJECT-EU) in order to gather and integrate all the Git repositories used for development. When adapting and extending existing software, either by one of the RADIO beneficiaries or by third parties, the original repository is forked into RADIO-PROJECT-EU and then the RADIO fork is used to track development.

3.1 Core Packages

The core motor control, sensor drivers, localization, and navigation packages. Where possible adapted from TurtleBot and Kobuki packages, with some packages specifically developed for the RADIO Robot.

Package name and description	Source code repository and release used for this deliverable
turtlebot: The basic drivers for running and using a TurtleBot with ROS. Forked from the official Turtlebot repository and modified and adapted for the RADIO Robot, creating a new model of the robot and sensors.	https://github.com/radio-project-eu/turtlebot
turtlebot_apps: A group of demos and examples used as templates for TurtleBot/ROS packages. Forked from the official Turtlebot repository and modified and adapted, creating configuration files specific to the RADIO Robot.	https://github.com/radio-project-eu/turtlebot_apps
kobuki: Software controllers for the Kobuki mobile base. Forked from the official Turtlebot repository and slightly modified for the RADIO Robot.	https://github.com/radio-project-eu/kobuki
kobuki_core: Software controllers for the Kobuki mobile base. Forked from the official Turtlebot repository and modified for the auto-docking procedure of the RADIO robot.	https://github.com/radio-project-eu/kobuki_core
turtlebot_radio_bringup: Bringup files for the robot. It contains all the configuration and launch files to run all the RADIO Robot components.	https://github.com/radio-project-eu/turtlebot_radio_bringup
robotnik_msgs: Definition of messages and services used by the core packages.	https://github.com/radio-project-eu/robotnik_msgs
turtlebot_radio_emergency: Node that implements safety stop. It manages the emergency button of the robot and disables any movement.	https://github.com/radio-project-eu/turtlebot_radio_emergency
marker_mapping: Package to localize ar_track_alvar markers and use them to initialize the localization module with a known pose every time the robot initializes. To avoid manual initialization, this node localizes visual QR markers onto the map and uses them inversely to calculate the correct pose of the robot.	https://github.com/radio-project-eu/marker_mapping

3.2 Robot Perception Packages

The robot perception stack, delivered as Integrated Data Analysis System (D3.9/D.10) is also installed on the RADIO Robot. For completeness, the list of packages is repeated here. Note that some of the packages listed in D3.9/D.10 are omitted, as they are relevant to the integrated RADIO Home. These will be listed in D4.8 *Integrated smart home with robotic platform extensions*.

Description	Source code repository and release used for this deliverable	
HumanPatterRecognition: Recognizes human walking patterns in laser scans and tracks walking.	https://github.com/radio-project-eu/HumanPatternRecognition	v2.0
HPR Wrapper: Uses HPR output to recognize and time "walked 4m" events.	https://github.com/radio-project-eu/hpr_wrapper	v1.0
ROSVisual: Tracks moving objects in the RGB/depth modality and classifies motion as bed or chair transfer.	https://github.com/radio-project-eu/ros_visual	v1.1
ROSVisual Wrapper: Uses the output from ROSVisual to time chair and bed transfer events and to recognize and time "walked 4m" events.	https://github.com/radio-project-eu/ros_visual_wrapper	v1.0
Motion Analysis: Recognizes motion and classifies it as "bed transfer" and "pill intake" events.	https://github.com/radio-project-eu/motion_analysis	v1.2
Motion Analysis Wrapper: Uses the output from motion analysis to time the bed transfer event.	https://github.com/radio-project-eu/motion_analysis_wrapper	v1.0
AUROS: Recognizes talking, watching TV, listening to music, and doing housework by acoustic analysis of the audio modality.	https://github.com/radio-project-eu/AUROS	v1.0
BLE Localization: Estimate the position of an object/Robot by observing RSSI values. Implementing filters, like Kalman filter, for normalize RSSI values and predict position.	https://github.com/radio-project-eu/robot_ble_localization	
Map convergence: Consumes robot pose messages from the localization package and converts them to the corresponding coordinates in the RADIO Home model.	https://github.com/radio-project-eu/map_convergence	

3.3 Robot Behaviour and RADIO Home Integration

Although some of the them execute on the robot's on-board computer, the packages that implement robot behaviour, control by the end user, and interactions with the RADIO Home Main Controller are listed in D4.8 *Integrated smart home with robotic platform extensions*, as they relate to integrating the robot into the RADIO Home environment.

4 Demonstration

4.1 Simulated Demonstration in FSL Environment

In order to test the localization and navigation capabilities of the RADIO robot in preparation of the actual trials at FSL (July 2016), a simulated RADIO robot and FSL environment have been created for the Gazebo simulator.[1]

These are at the **turtlebot_simulator** package, available at https://github.com/RADIO-PROJECT-EU/turtlebot_simulator

This package contains the configurations and definitions needed to have a virtual RADIO robot in a Gazebo simulation. The robot model is based on the standard TurtleBot2 model, but has been significantly adapted to account for the modifications carried out in RADIO (cf. Section 2). Figure 5 shows the model of the RADIO robot. A new simulated environment of the trials area at FSL has also been created (Figures 7 and 8) based on the floorplan provided by FSL (Figure 6).

By testing the localization and navigation capabilities of the RADIO robot in this simulated environment (Figure 9), these capabilities have been adapted to guarantee a robust and reliable navigation and localization of the robot before the actual deployment at FSL. This has considerably shortened the time needed to install and configure the RADIO prototype for the trials.

4.2 Physical Demonstration in FHAG Environment

The first prototype of the RADIO Home (including the robot) was deployed in March 2017 at FHAG premises, in preparation for the first round of the piloting study.

A demonstration is at https://vimeo.com/211673690 The demonstration of the robot starts at 1:30.

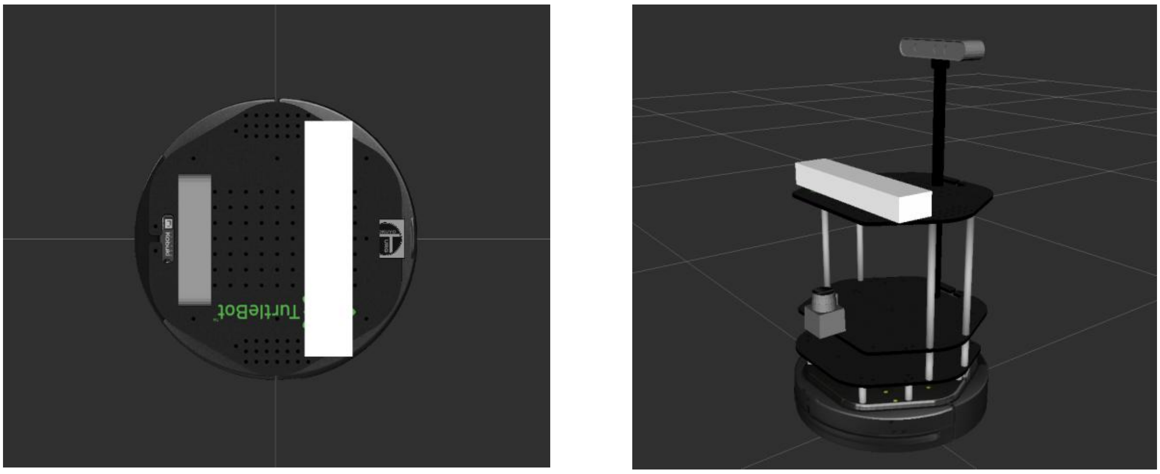

Figure 5: Radio robot model in ROS

[1] Cf. http://gazebosim.org

Figure 6: FSL floorplan

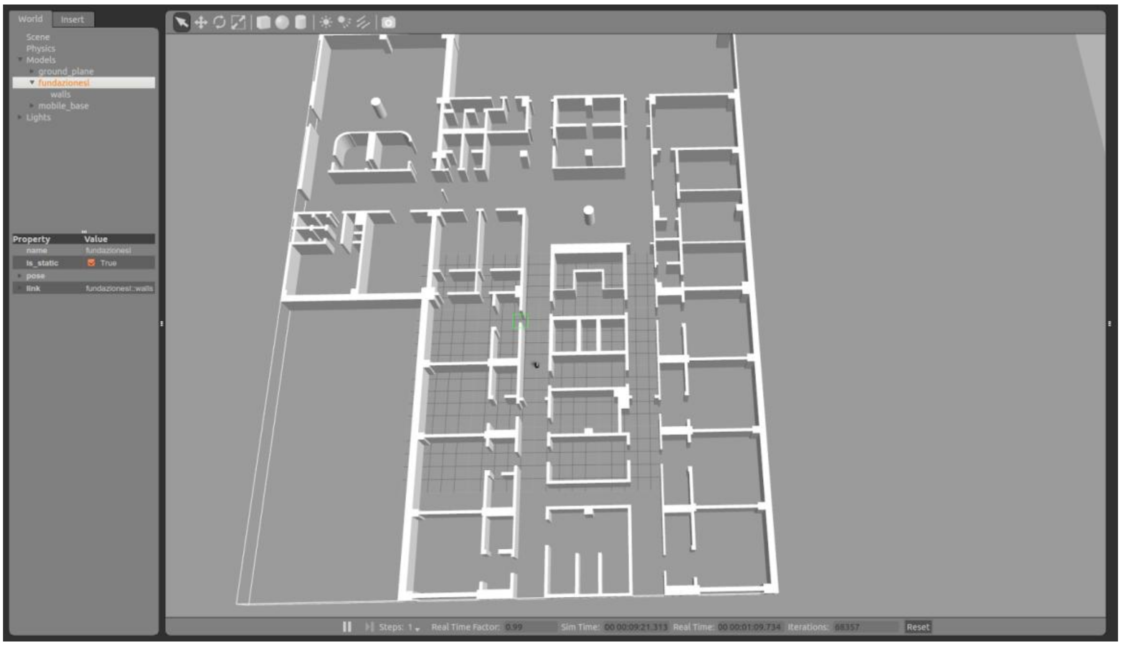

Figure 7: Building conversion for simulation in Gazebo

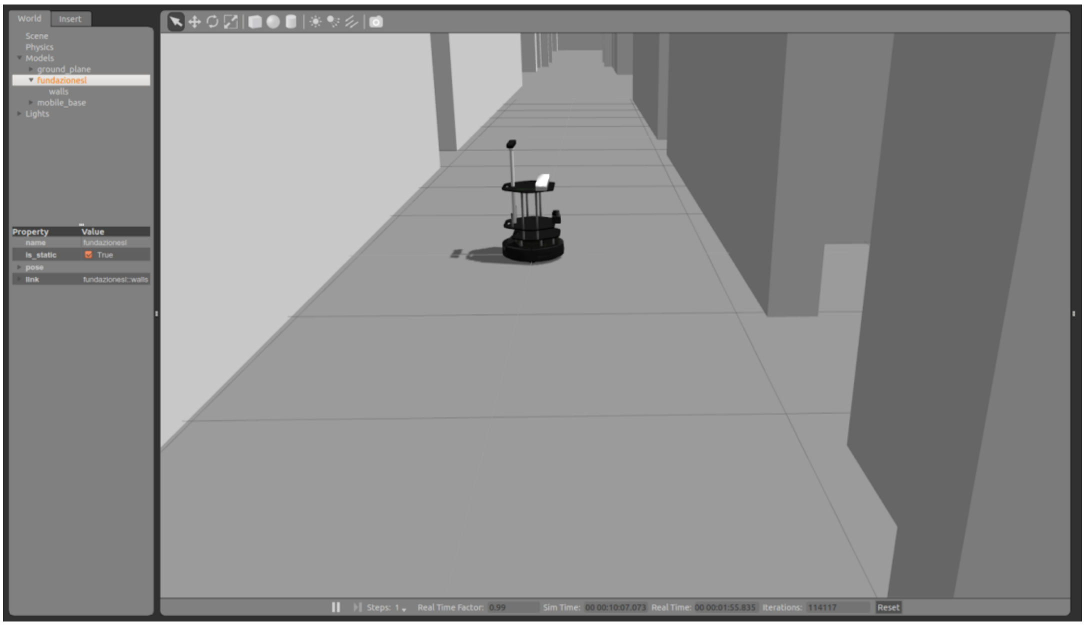

Figure 8: RADIO robot simulation on FSL building

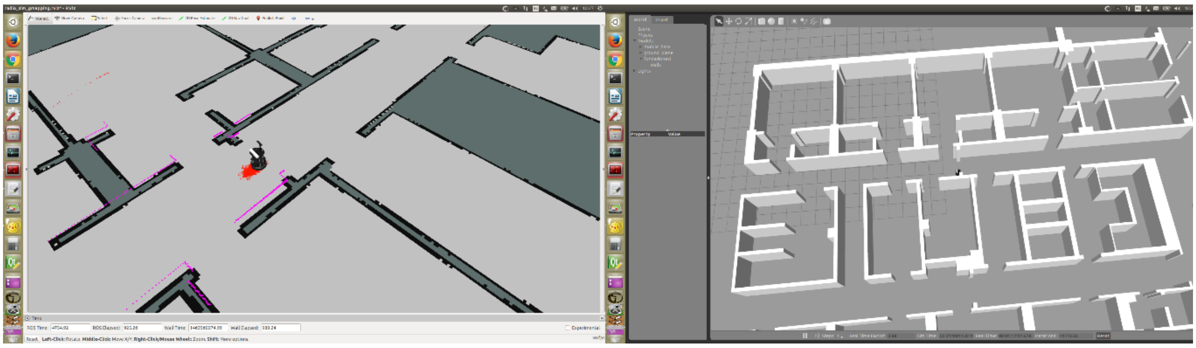

Figure 9: RADIO navigation tests in the FSL premises