Deliverable D25.7

User's Manual

Antonio BANDERA RUBIO (UMA)
Rebeca MARFIL ROBLES (UMA)

CONTENTS

CONTENTS 2
1 Overview 5
2 General information 6
2.1 Terms and concepts 6
2.1.1 Hardware concepts 6
2.1.2 The CGAmed server 7
2.2 User Access 7
3 Turning on the CLARC framework 8
3.1 The CLARA robot 8
3.1.1 Turning on the robot 8
3.2 CGAmed web 9
3.2.1 Login in the CGAmed web 9
3.2.1 The Robot Control service on the CGAmed 10
3.2.2 The Schedule service on the CGAmed 11
4 Launching a CGA session 20
4.1 Starting up the components on CLARA robot 20
4.2 Launching a session 21
5 Shutting down 22
6 Charging the robot 24
7 Bugs 25
Overview 27
8 General information 28
8.1 Terms and concepts 28
8.1.1 Hardware concepts 28
8.1.2 The CGAmed server 29
8.2 User Access 30
9 Using the CGAmed tools for session review 31
9.1 Login in the CGAmed web 31
9.2 The Test results service on the CGAmed 32
9.3 Analyzing a recorded session (Barthel example) 33
9.4 Analyzing a recorded session (Get up & Go example) 39
9.5 Comparing the results of several tests 46
10 Bugs 48

Overview 50
11 General information 51
11.1 Terms and concepts 51
11.1.1 Hardware concepts 51
11.1.2 The CGAmed server 53
11.3 User Access 54
12 Setting the CLARC framework 54
12.1 The CLARA robot 54
12.1.1 Turning on the robot 54
12.1.3 Capturing the map of the environment 56
12.1.4 Editing the map 58
12.1.5 Localizing the robot in the map 59
12.1.6 Setting goal poses for Barthel and Get Up & Go tests 61
12.1.7 Setting the goal pose for the Charging Station 63
12.1.8 Updating the map in the start.sh script file 65
13 Administration facilities in the CGAmed 66
13.1 Login in the Administration web 66
13.2 Managing the list of goal poses 67
13.3 Managing the list of patients 69
13.4 Video streaming 71
14 Bugs 72

Part I - Session management

1 Overview

About this manual

This manual describes how to use the **CLARC framework**.

Usage

This manual describes the protocol that allows an user to manage the CLARC framework.

Who should read this manual?

This manual is intended for those users in charge of managing the CLARC framework as a tool for automatizing the Comprehensive Geriatric Assessment (CGA) tests.

Prerequisites

- The reader should have basic skills on managing a personal computer
- The CLARC framework has had to be previously deployed (see User's Manual - Part III)

Organization of chapters

The manual is organized in the following chapters:

Chapter	Title	Contents
1	Introduction	Contains basic information about the CLARC framework, and explanations of the terms and concepts needed for understanding the rest of the document.
2	Turning on	Instructions for turning on the CLARC framework
3	Launching a CGA test	Performing a Barthel or Get Up & Go test
4	Shutting down	Instructions for shutting down the robot
5	Charging the robot	The process for charging the robot

2 General information

2.1 Terms and concepts

CLARC is a complete framework for robotizing two specific tests that are typically part of a Comprehensive Geriatric Assessment (CGA) procedure: the Barthel test and the Get Up & Go test. CLARC consists of two major elements: **CLARA**, a social robot able to interact with the patients, and capture and analyze the obtained data; and the **CGAmed**, a local server able to store a database with all captured data and to provide the physicians with the tools for online monitoring and offline editing and supervision. CLARC provides all hardware items and do not require any specific constraint to be deployed.

2.1.1 Hardware concepts

Overview

This section introduces the hardware in the CLARC framework.

Additional information

The hardware in the CLARC framework is also described in the deliveries

User's Manual - Part III - System deployment

CLARC - Functional prototype

Standard hardware

The table below describes the standard hardware in an CLARC framework

Hardware	Explanation
CLARA robot	The robot is based on a differential driven platform by MetraLabs.
Charging station	The robot has a charging station to be able to charge autonomously.
Remote Control	Portable device connected to the robot that allows the user to interact with the system using large buttons.
Router	CLARC works in a **local network**, in which all the components are connected to the wifi provided by this router.
CGAmed embedded PC	This PC stores all the information about users, sessions, etc.

2.1.2 The CGAmed server

Overview

This section introduces the webs in the CGAmed station

Additional information

The use of the CGAmed web for reviewing the results is described with details in

User's Manual - Part II - Results review

Webs in the CGAmed

The table below describes the webs in an CGAmed station

Web	Explanation
Administration 192.168.0.70	The administration web is used to configure • The positions where the robot is going to perform the tests • The list of patients • The IP address of the camera for online supervision mounted on CLARA robot (Section 1.1.1 - The CLARA robot)
CGAmed 192.168.0.70/cgamed	The CGAMed is used to: • Add new patients. • Add new sessions. • Start/Stop a session. • Pause/Resume a session. • Move the robot to a position (from a list of predefined ones). • See and compare the results of the tests.

2.2 User Access

The table below provides the default user/password data needed to access to the modules in the system.

Module	Access
Linux based PC (CLARA)	Password: scitos

Windows based PC (CLARA)	Accessible from the Linux based PC using the Remmina remote desktop app
CGAmed embedded PC	User / password: isis / grupoisis
CGAmed	URL (CGAmed) 192.168.0.70/cgamed user / password: adminWeb / adminSecure URL (Administration) 192.168.0.70 user / password: admin / adminRobot

Note: All CGAmed stations share currently the same IP Address. Contact us if you need to change this address, as this change implies internal updates on the software modules on CLARA robot.

3 Turning on the CLARC framework

3.1 The CLARA robot

3.1.1 Turning on the robot

Overview

This section explains how to turn on the robot.

The video Starting.mp4 explains how to turn on the robot. As the video shows, when you turn on the robot using the key, the two internal PCs are automatically turned on.

Note: Once the PC's are on, you must connect the Linux based PC to the local wifi network provided by the router (See Section 3.1).

3.2 CGAmed web

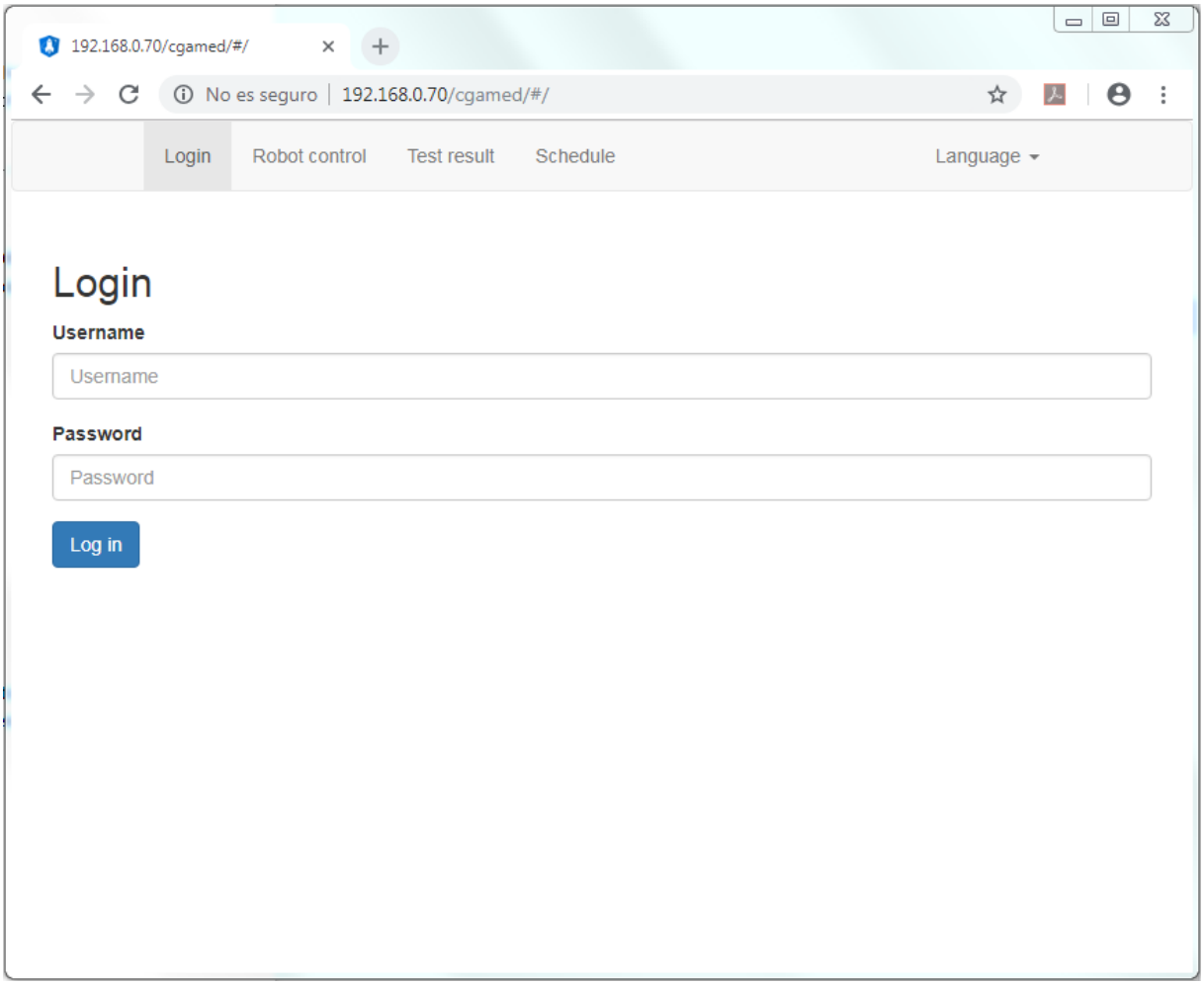
3.2.1 Login in the CGAmed web

Overview

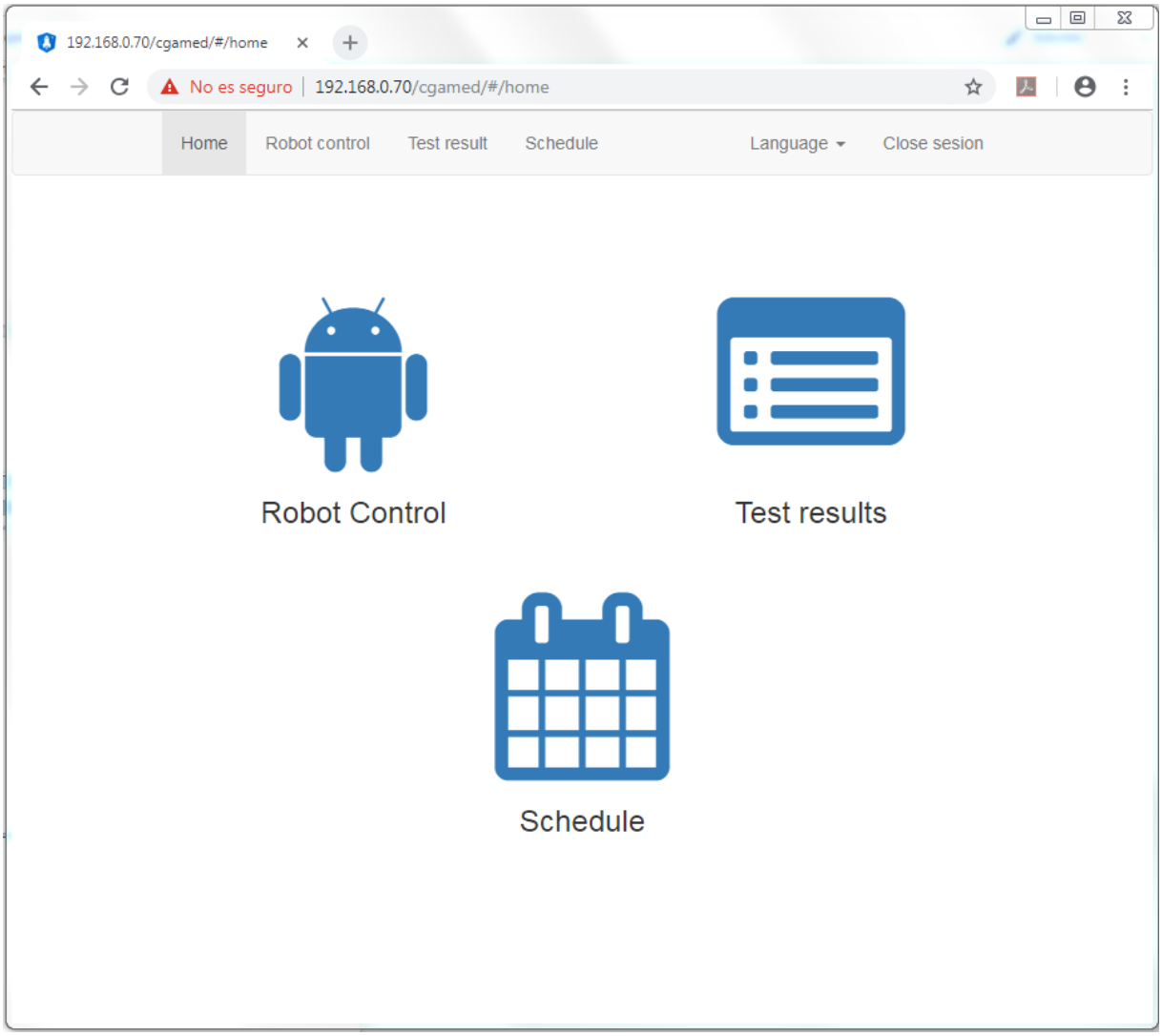
This Section describes the procedure for entering within the CGAmed web. This web provides the tools for (a) scheduling the agenda of a CLARA robot, or (b) manually launching a CGA session.

Login in the CGAmed web

When you connect to the URL of the CGAmed web (http://192.168.0.70/cgamed), you need to add user and password for entering on the web. This data is provided in Section 1.2 (page 7).



Once logged into the web, the main page allows you to access to three different services. It is also possible to choose the Language or to Close the session.



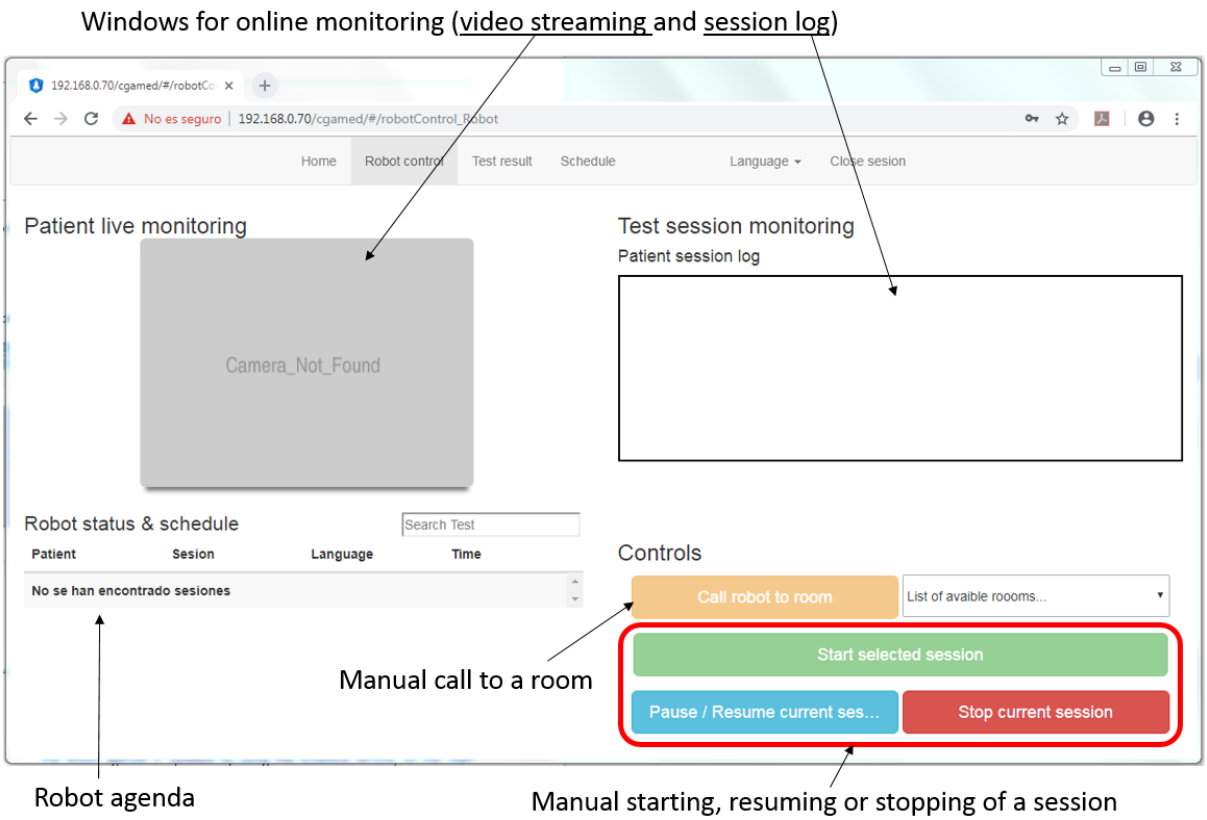
3.2.1 The Robot Control service on the CGAmed

Overview

The Robot Control interface provides the user with the tools for (a) visualizing the agenda of the CLARA robot, (b) manual launching or stopping of a CGA test, and (c) online supervising the session.

The Robot Control interface

Just clicking on the Robot Control icon in the main page of the CGAmed you access to the interface shown below.



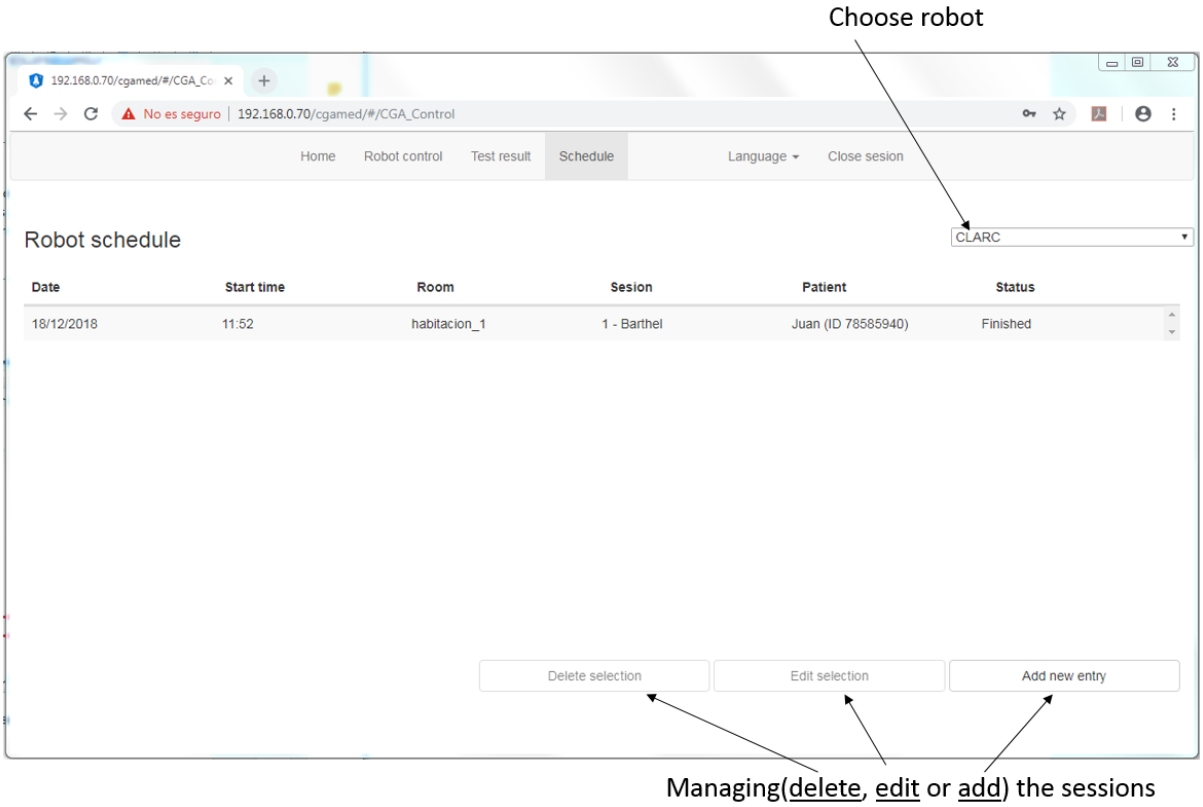
The Robot agenda is updated by using the Schedule service, on the main page (see Section 2.2.3, page 12).

3.2.2 The Schedule service on the CGAmed

Overview

The Schedule interface provides the user with the tools for managing the agenda of the CLARA robot. It allows to add patients and sessions to the agenda of the robot. This data can be visualized in the Robot Control interface.

2.2.3a The Schedule interface

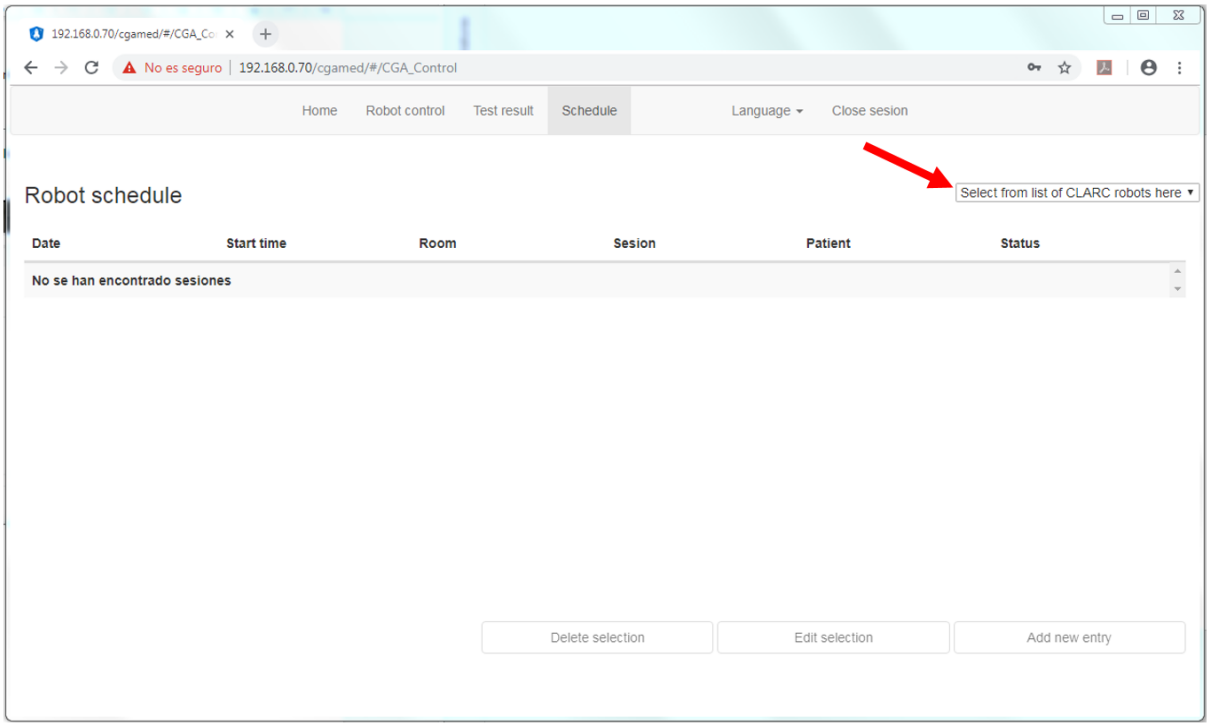


2.2.3b Adding a new patient

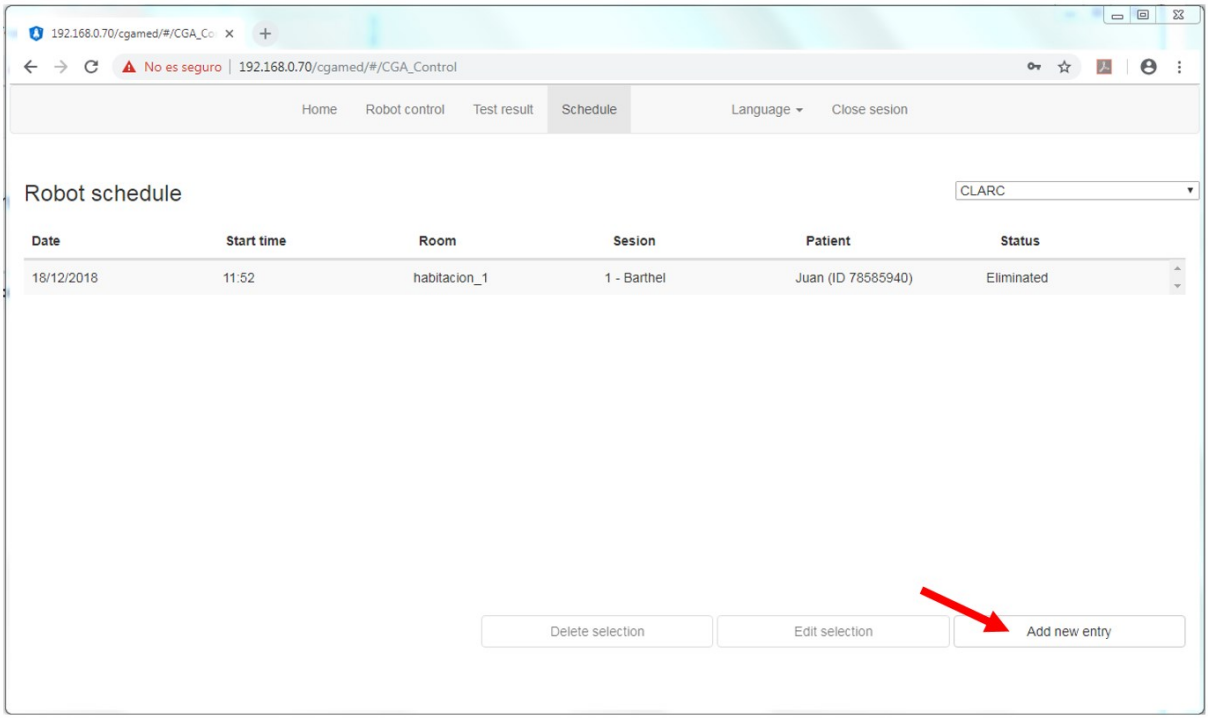
The process to add a new patient is explained in the **SetUp.avi** video (from 2:37).

The procedure for adding a new patient consists of the following steps:

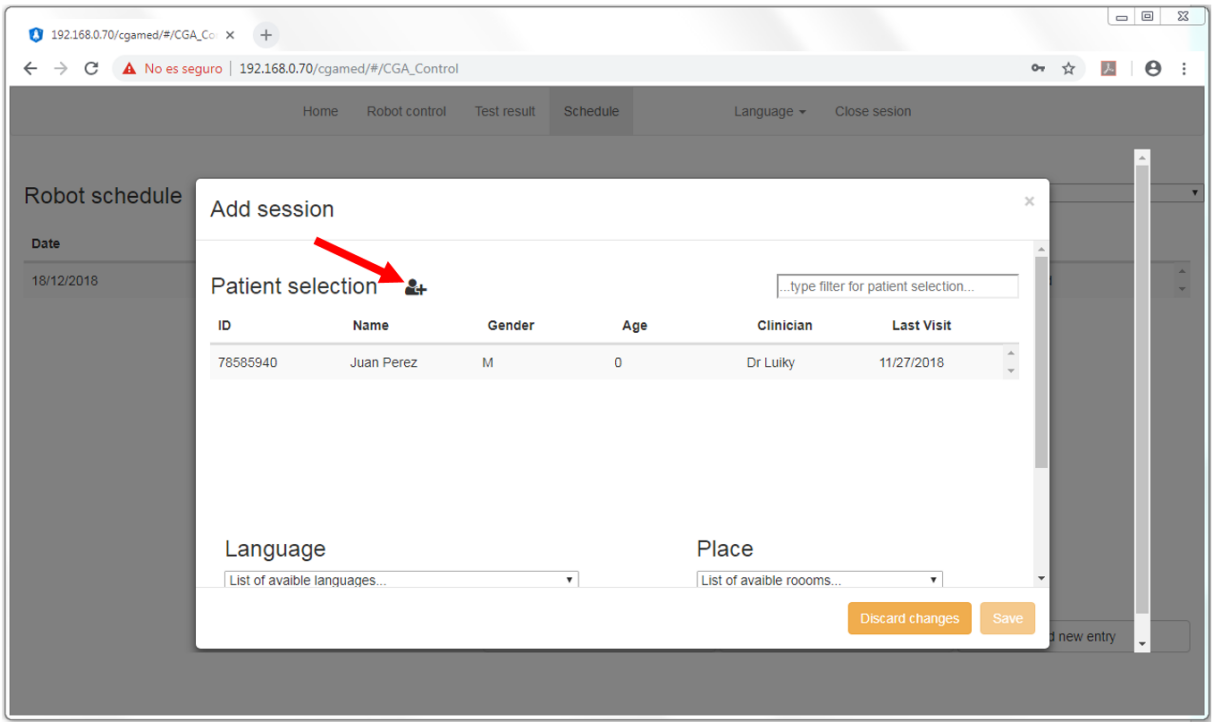
1. Choosing a robot



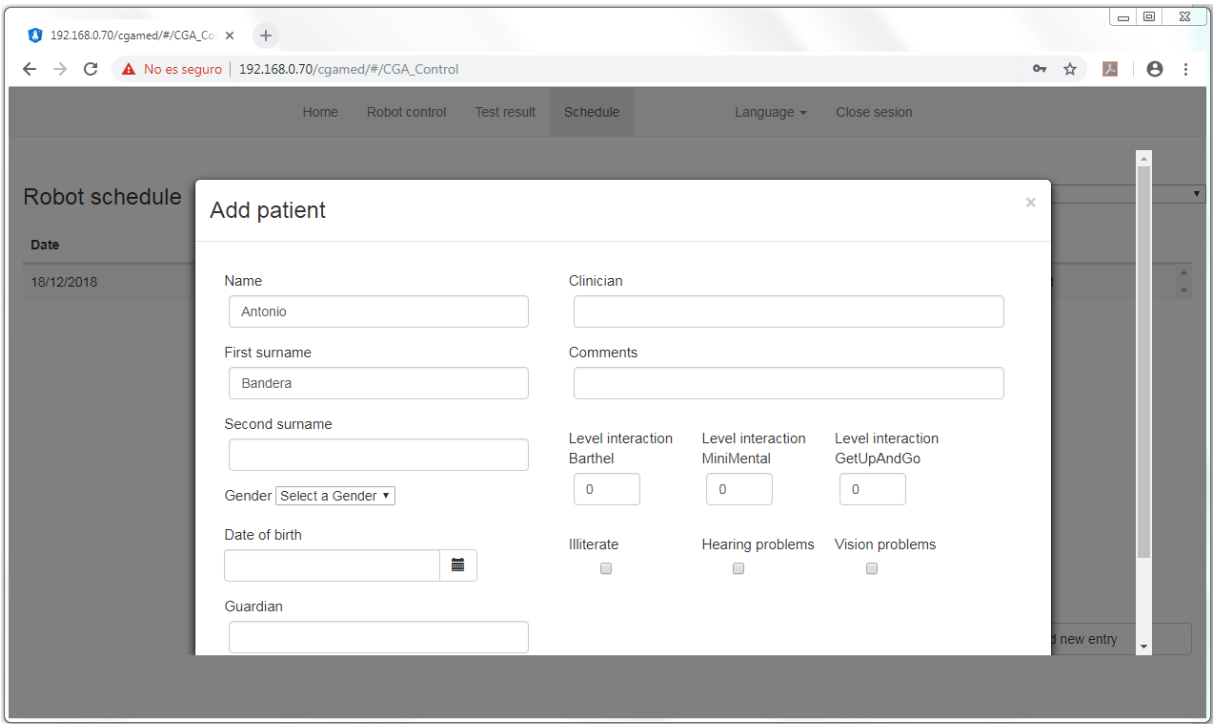
2. Add a new entry



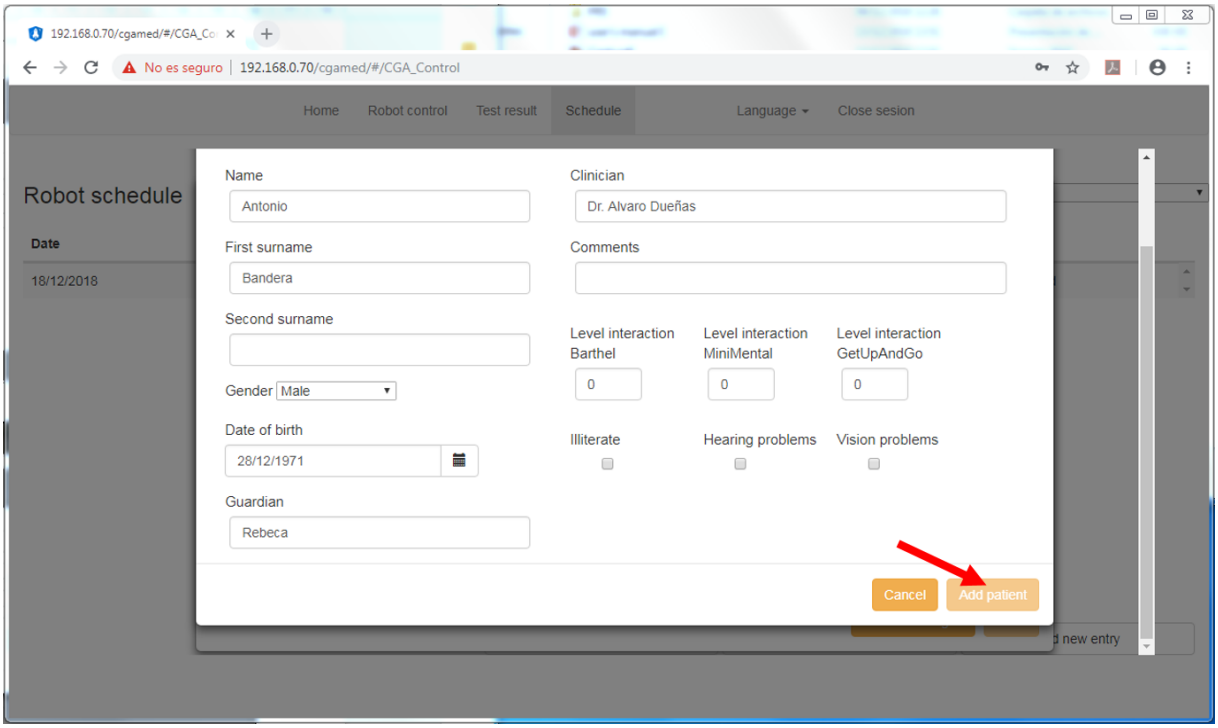
3. Click on the icon for adding a new patient



4. Add information about the patient



5. Close and save the patient profile.

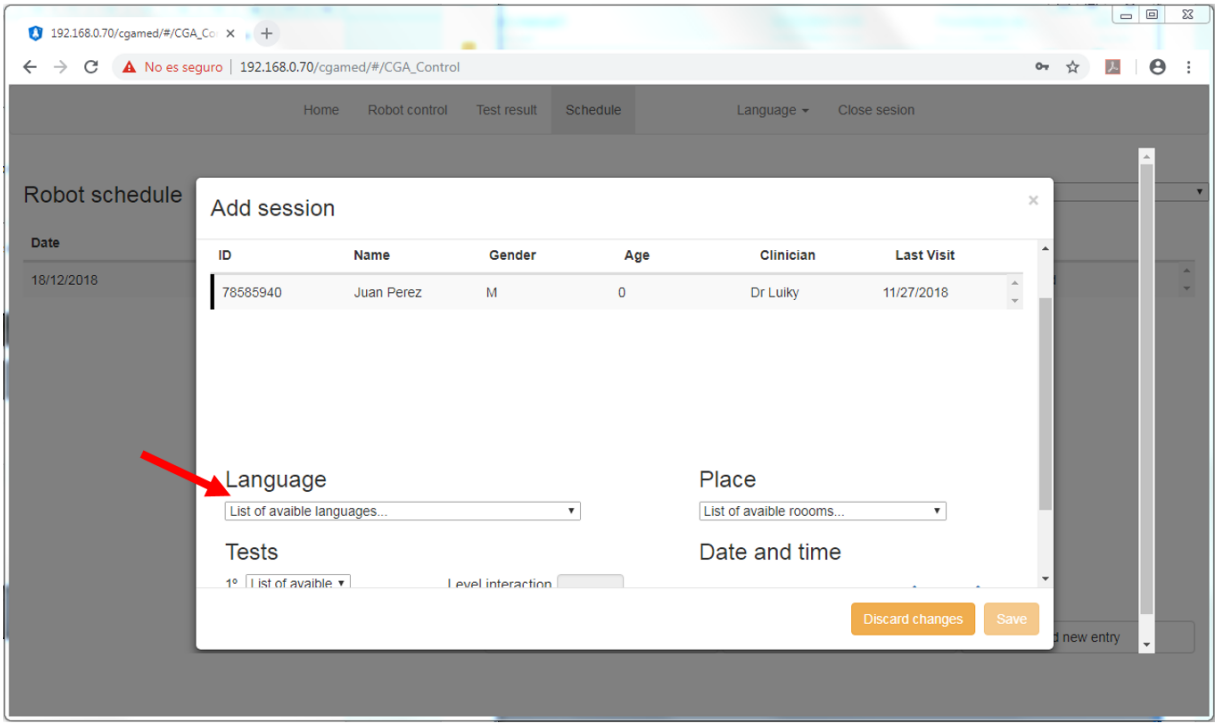


2.2.3c Adding a new session

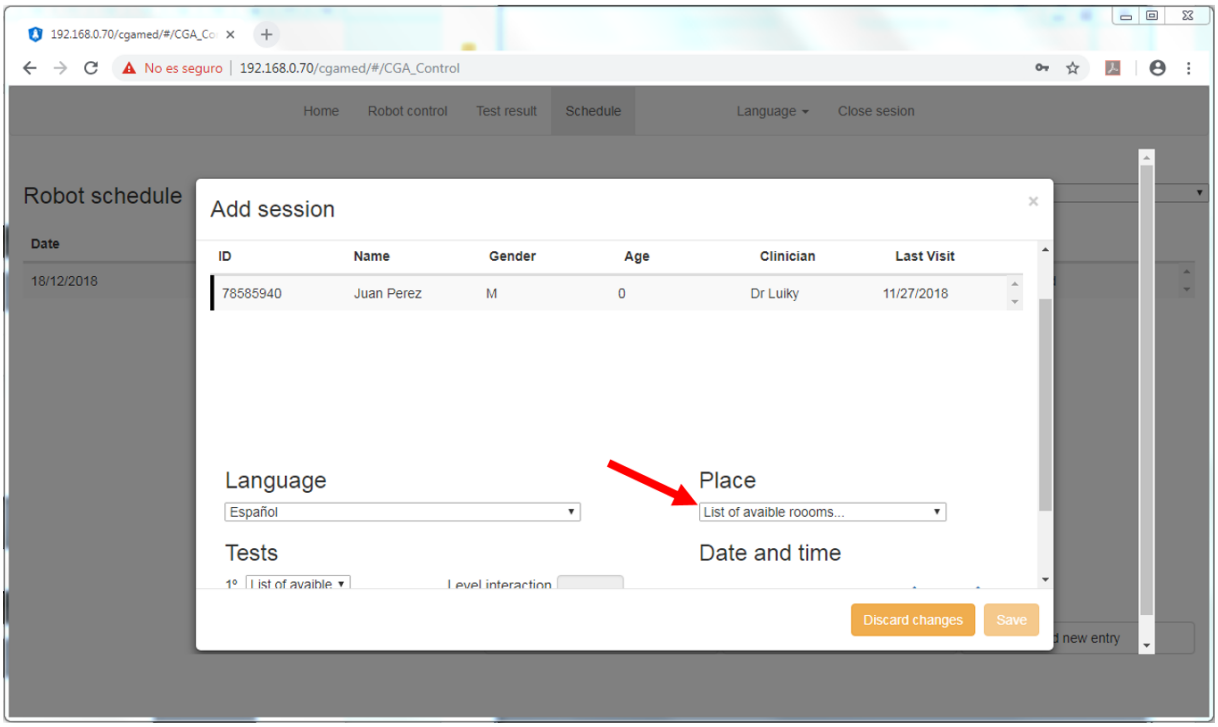
The process to add a new session is explained in the **SetUp.avi** video (from 3:35).

Once a patient has been chosen (for adding a new one, see Section 2.2.3b, page 13), the procedure for adding a new session consists of the following steps:

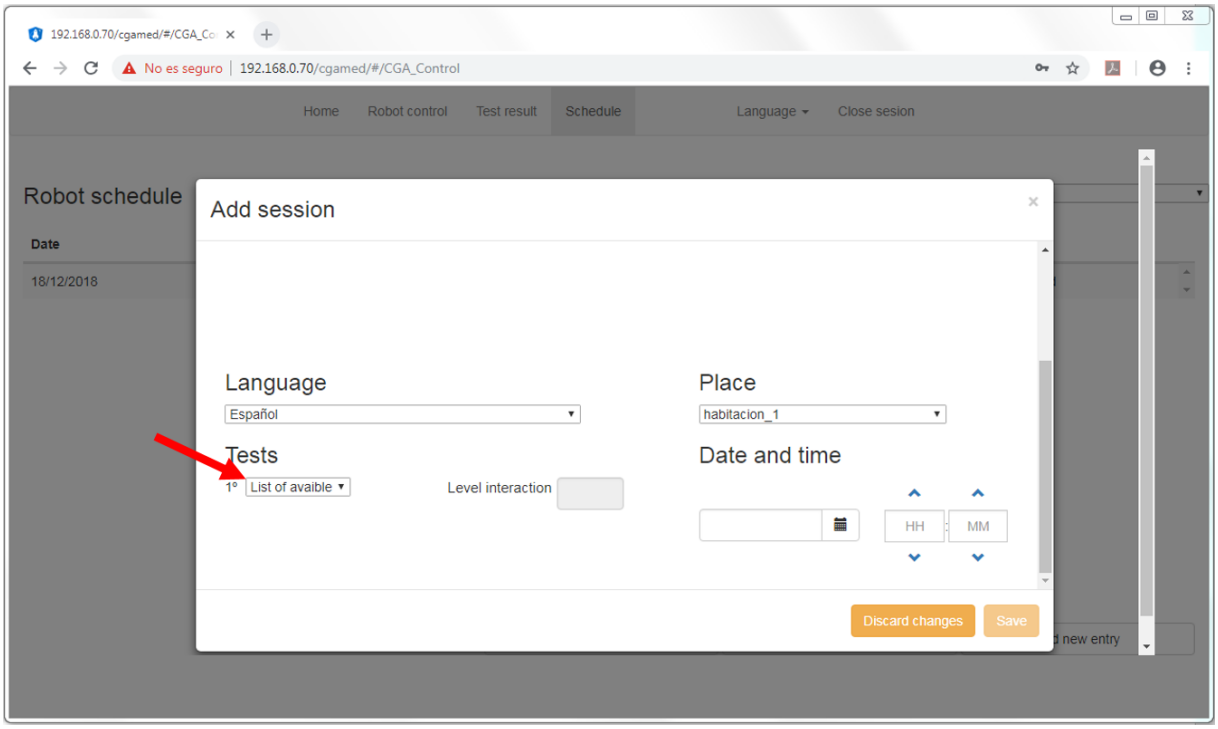
1. Choosing the Language for the test



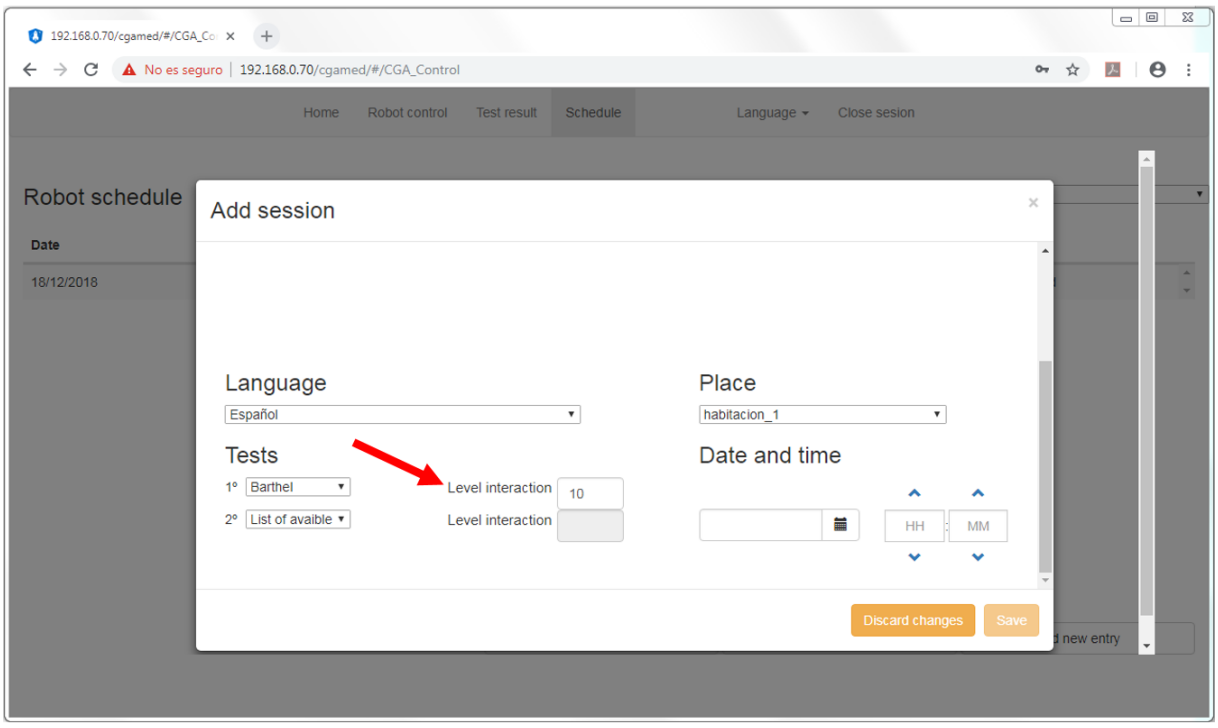
2. Choosing the room



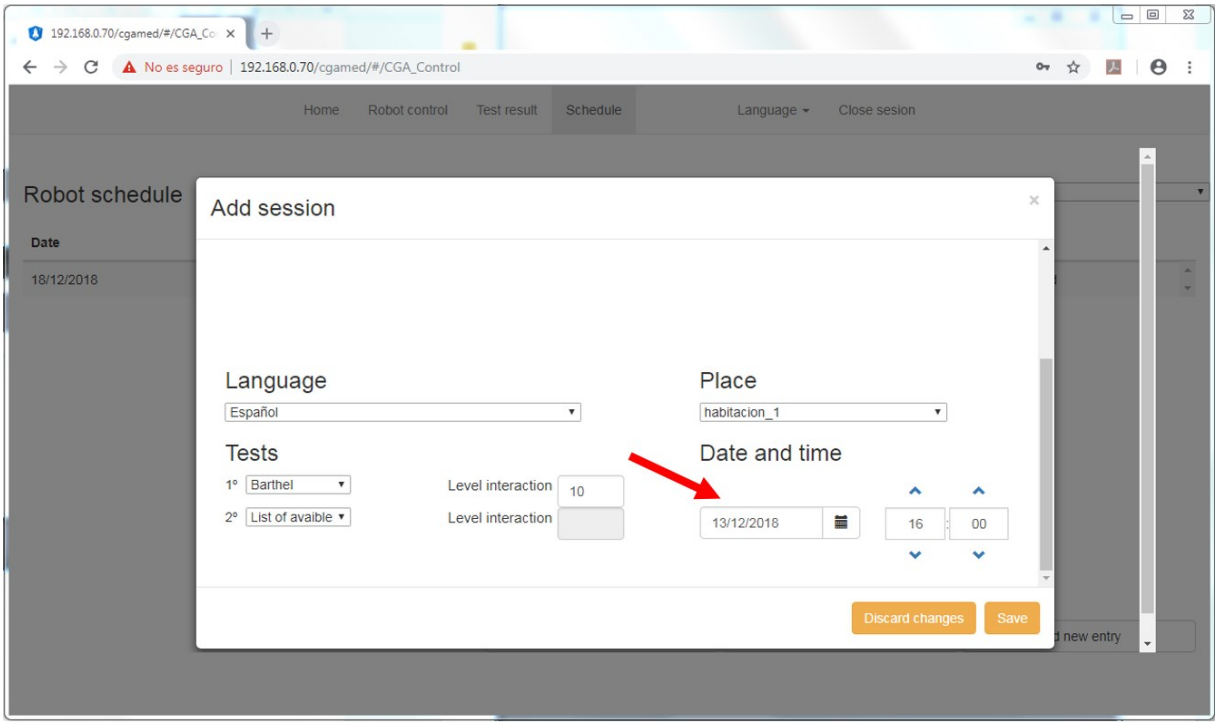
3. Choosing the test



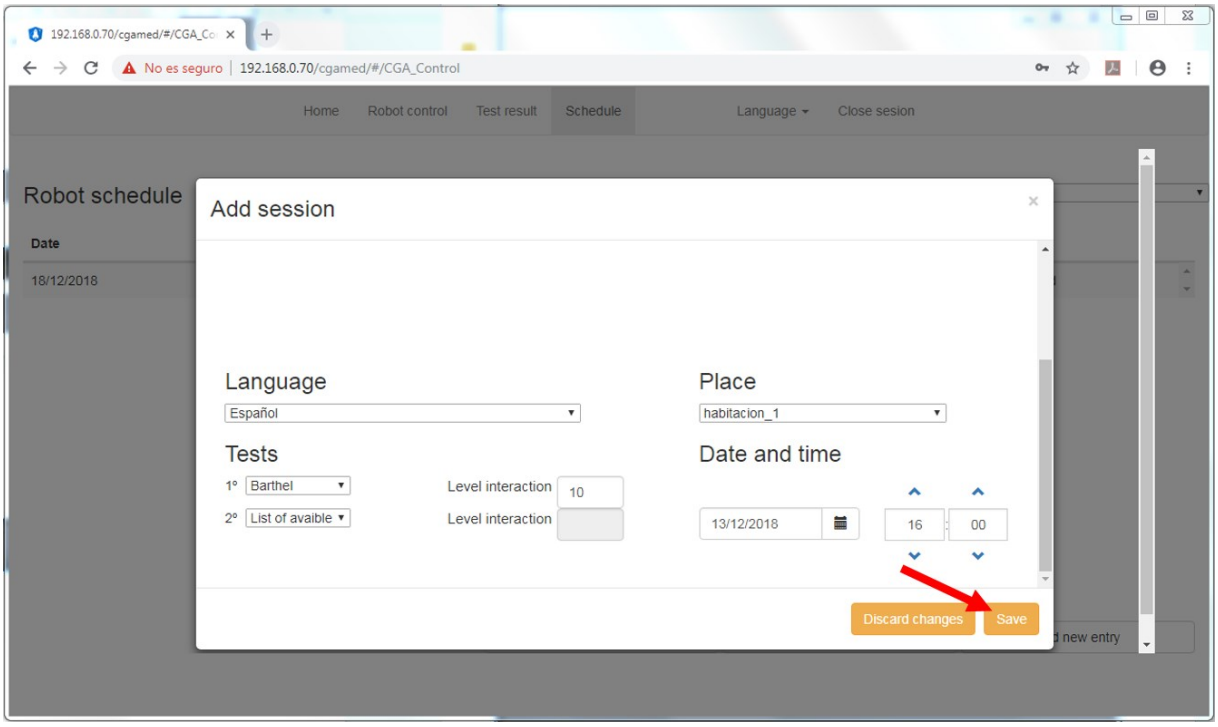
4. Choosing the level of interaction of the text (10 if you do not want that the patient performs the training test, and 0 in the other case).



5. Choosing the date and hour



6. Close and save the session



Once the session has been closed, you can visualize the new entry on the Schedule and Robot Control interfaces.

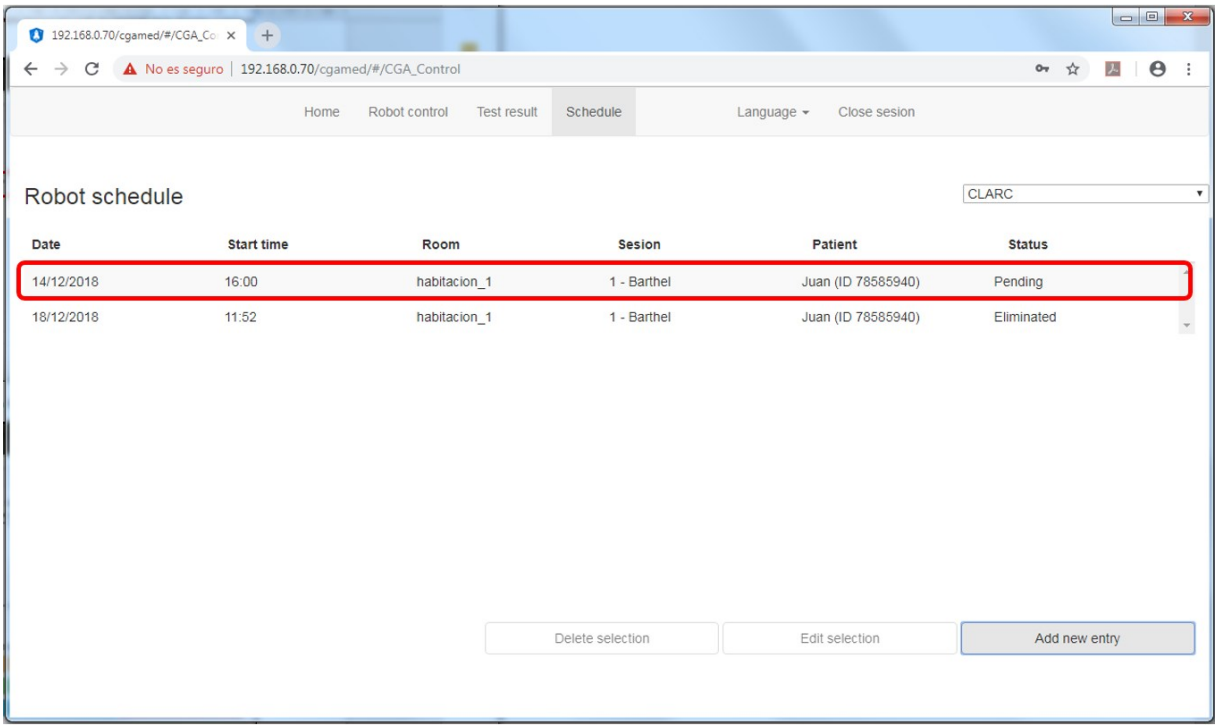


Figure 1: Schedule interface - New entry on the robot's agenda

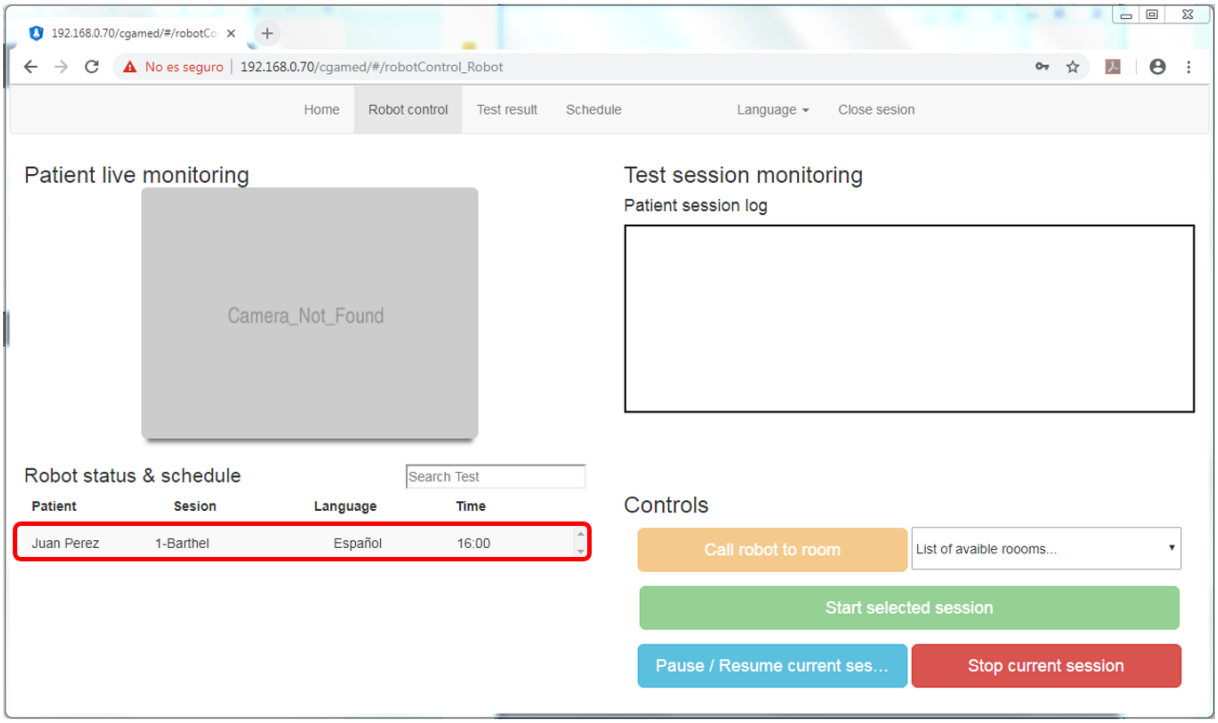


Figure 2: Robot control interface - New entry on the robot's agenda

4 Launching a CGA session

The process of launching a session, which has been previously programmed in the schedule service on the CGAmed (see Section 2.2.3c in 2.2.3 The Schedule service on the CGAmed), implies to firstly start up the software architecture endowed in the CLARA robot (the CORTEX architecture).

4.1 Starting up the components on CLARA robot

Overview

This Section describes how to start up the CORTEX architecture in the CLARA robot.

The process to start up the components on CLARA robot is explained in the **Starting up robot components.ogv** video.

Starting up the components on CLARA robot

Previous steps

1. Turn on the router
2. Turn on the embedded PC with the CGAmed database and connect it to the local wifi network provided by the router
3. Turn on the Remote Control device (using the small black switch)

Starting up the software components on CLARA

4. Open a yakuake session and launch the `start.sh` script.

```
>>cd robocomp_clarc/robocomp/components/cajasvaciasechord/

>>./start.sh
```

The different agents on the CORTEX software architecture will be wake up.

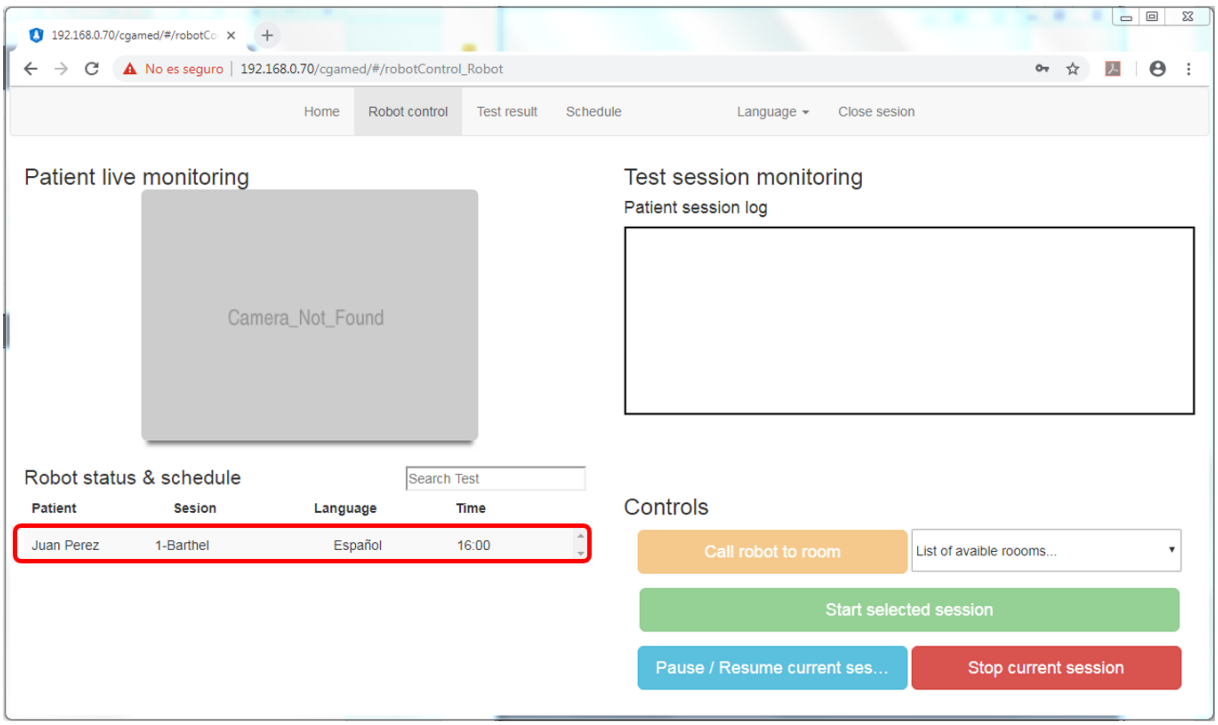
Once all the components are started and the planner (Execution - Pelea shell terminal) is waiting for click, you can launch a test from the CGAMed web.

4.2 Launching a session

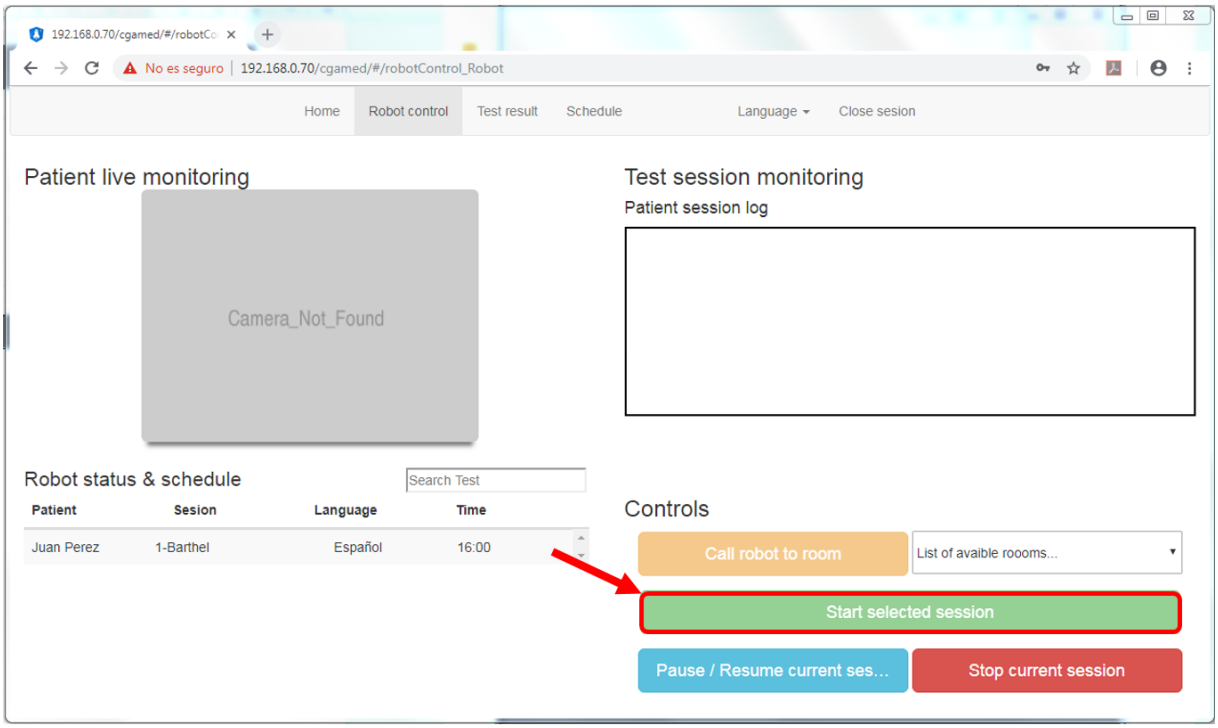
Overview

This Section describes how to launch a session previously scheduled (the CORTEX architecture in the CLARA robot has been started as explained in Section 3.1).

In the Robot Control tab of the CGAmed (see Section 2.2.3) appears the scheduled session



Select the desired session in the **Robot status & schedule** list and push the **Start selected session** button.



5 Shutting down

Overview

This section describes how to shut down the robot in an correct way.

The process to shut down the robot is explained in the **Shutting down.ogv** video. After following the steps in the video you can shut down the Linux based PC.

Stopping the software components on CLARA

1. Launch the `kill.sh` script in "robocomp_clarc/robocomp/components/cajasvaciasechord/"
2. Close all the yakuake sessions.

Stopping the hardware components on CLARA

3. **Shutting down the windows pc:** connect to the windows pc using the remmina desktop application and click in "apagar equipo".

4. **Shutting down the Raspberri Pi (old CLARA robot):** connect to the Raspberry pi using ssh and stop it:

```
ssh pi@192.168.0.59

>>sudo poweroff
```

5. Shut down the linux pc.

Shutting down the remote control

6. To shut down the remote control, first press the small red button and then, when the red light will be off, move the black switch to the off position.

6 Charging the robot

Overview

If the robot detects that its battery is discharged, it will leave all it is doing (e.g. a test) and will go automatically to charge. Hence, it is recommended to **charge the robot every night** in order it be ready for doing tests during the day.

The process to charge the robot is presented in the Charging.mp4 video.

To charge the robot, it must be correctly placed (centered) on the charging station and the key must be in the ON position. When it is charging, the small light close to the key of the robot is blinking and the “Charging” light in the base station is on.

Note: You can command the robot to go to the Charging Station by clicking the “call robot to room” button in the CGAMed web, selecting previously the base_station in the list of available rooms. In this case, you must control that, when the robot finish to move, it is well located in the charging station and it is charging. It must be noted that this option is only possible if the components of the robot are running and the base_station have been correctly calculated and written in the goalPositions.txt file and in the CGAMed database.

7 Bugs

Module	Problem
CGAmed	The IP Address is currently the same in all CGAmed stations. This will provoke conflicts when several robots work in the same environment.

Part II - Results review

About this manual

This manual describes how to review the results captured of a Comprehensive Geriatric Assessment (CGA) session by the **CLARC framework**.

Usage

This manual describes how the physician can access and review the results captured by the **CLARA** robot during a CGA session using specific interfaces on the **CGAmed** app.

Who should read this manual?

This manual is intended for those physicians in charge of monitoring the state of a set of patients helped by the CLARC framework.

Prerequisites

The reader should have basic knowledge of the protocol associated to the Barthel and Get Up & Go tests (CGA session).

Organization of chapters

The manual is organized in the following chapters:

Chapter	Title	Contents
1	Introduction	Contains information about the hardware elements in the CLARC framework, and basic explanations of the terms and concepts needed for understanding this manual
2	Using the CGAmed tools for session review	Describes how to review the results provided by the CLARC framework.

8 General information

8.1 Terms and concepts

CLARC is a complete framework for robotizing two specific tests that are typically part of a Comprehensive Geriatric Assessment (CGA) procedure: the Barthel test and the Get Up & Go test. CLARC consists of two major elements: **CLARA**, a social robot able to interact with the patients, and capture and analyze the obtained data; and the **CGAmed**, a local server able to store a database with all captured data and to provide the physicians with the tools for online monitoring and offline editing and supervision. CLARC provides all hardware items and do not require any specific constraint to be deployed.

8.1.1 Hardware concepts

Overview

This section introduces the hardware in the CLARC framework.

Additional information

The hardware in the CLARC framework is also described in

User's Manual - Part III - System deployment

CLARC - Functional prototype

Standard hardware

The table below describes the standard hardware in an CLARC framework

Hardware	Explanation
CLARA robot	The robot is based on a differential driven platform by MetraLabs.
Charging station	The robot has a charging station to be able to charge autonomously.

Remote Control	Portable device connected to the robot that allows the user to interact with the system using large buttons.
Router	CLARC works in a **local network**, in which all the components are connected to the wifi provided by this router.
CGAmed embedded PC	This PC stores all the information about users, sessions, etc.

8.1.2 The CGAmed server

Overview

This section introduces the webs in the CGAmed station

Additional information

The use of the Administration web on the CGAmed is described with details in

User's Manual - Part III - System installation

The use of the CGAmed web for managing a session is described with details in

User's Manual - Part I - Session management

Webs in the CGAmed

The table below describes the webs in an CGAmed station

Web	Explanation
Administration 192.168.0.70	The administration web is used to configure • The positions where the robot is going to perform the tests • The list of patients • The IP address of the camera for online supervision mounted on CLARA robot (Section 1.1.1 - The CLARA robot)

CGAmed 192.168.0.70/cgamed	The CGAMed is used to: ● Add new patients. ● Add new sessions. ● Start/Stop a session. ● Pause/Resume a session. ● Move the robot to a position (from a list of predefined ones). ● See and compare the results of the tests.

8.2 User Access

The table below provides the default user/password data needed to access to the modules in the system.

Module	Access
Linux based PC (CLARA)	Password: scitos
Windows based PC (CLARA)	Accessible from the Linux based PC using the Remmina remote desktop app
CGAmed embedded PC	User / password: isis / grupoisis
CGAmed	URL (CGAmed) 192.168.0.70/cgamed user / password: adminWeb / adminSecure URL (Administration) 192.168.0.70 user / password: admin / adminRobot

Note: All CGAmed stations share currently the same IP Address. Contact us if you need to change this address, as this change implies internal updates on the software modules on CLARA robot.

9 Using the CGAmed tools for session review

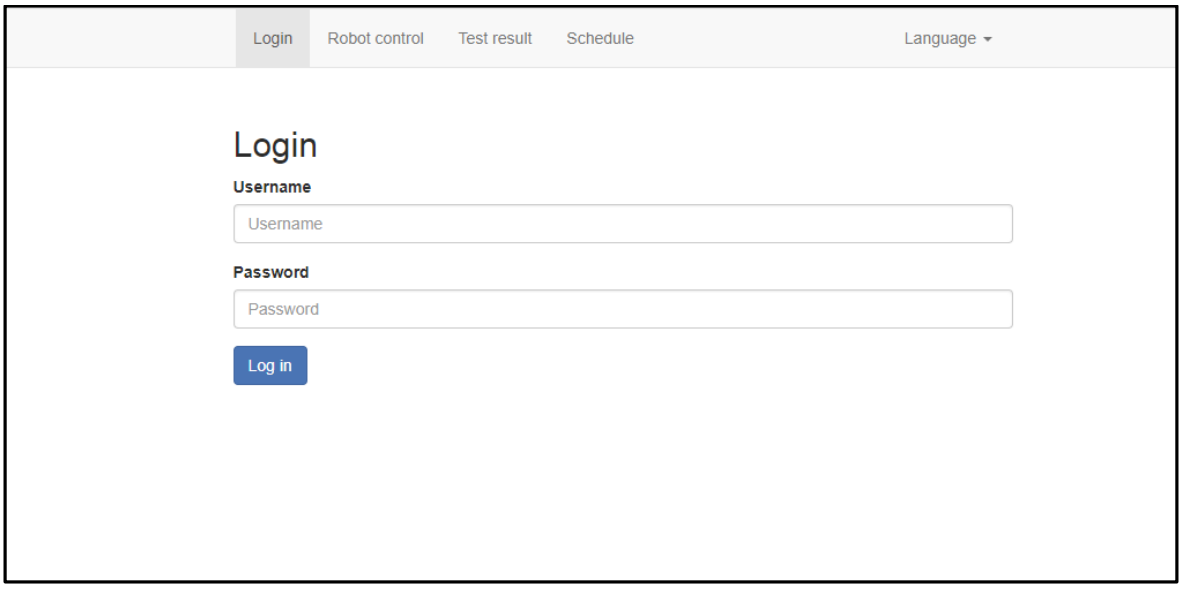
9.1 Login in the CGAmed web

Overview

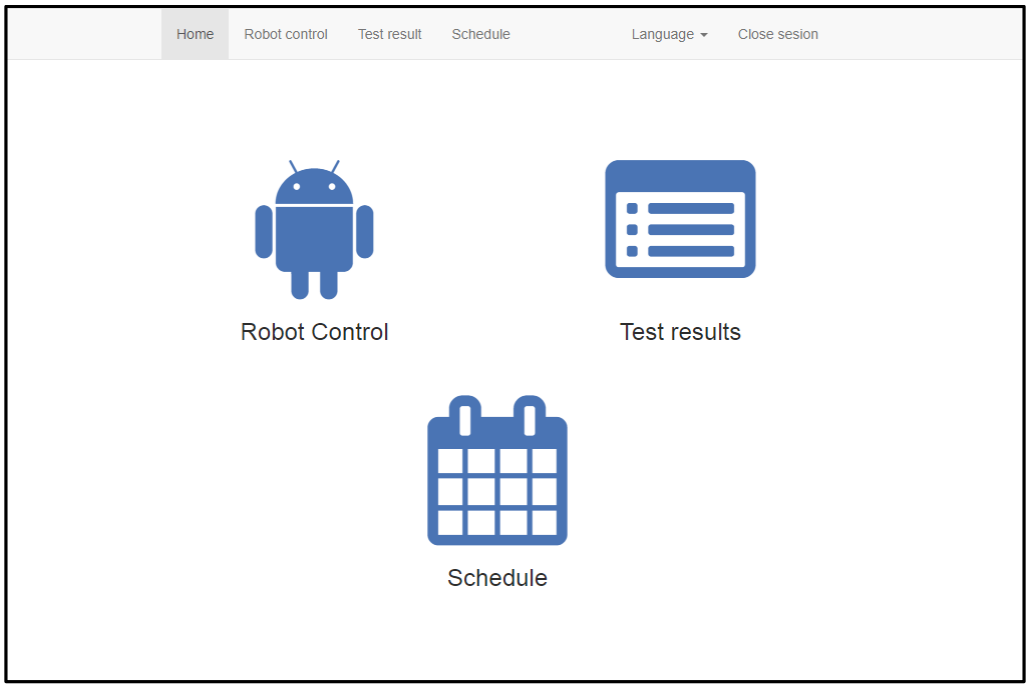
This Section describes the procedure for entering within the CGAmed web.

Login in the CGAmed web

When you connect to the URL of the CGAmed web (http://192.168.0.70/cgamed), you need to add user and password for logging into the web. This data is provided in Section 1.2.



Once logged into the web, the main page allows you to access to three different services. It is also possible to choose the Language or to Close the session.



Additional information

This Manual focuses on the reviewing of the results provided by the CLARC framework (**Test results** service). Further information about the **Robot Control** and **Schedule** services can be revised in

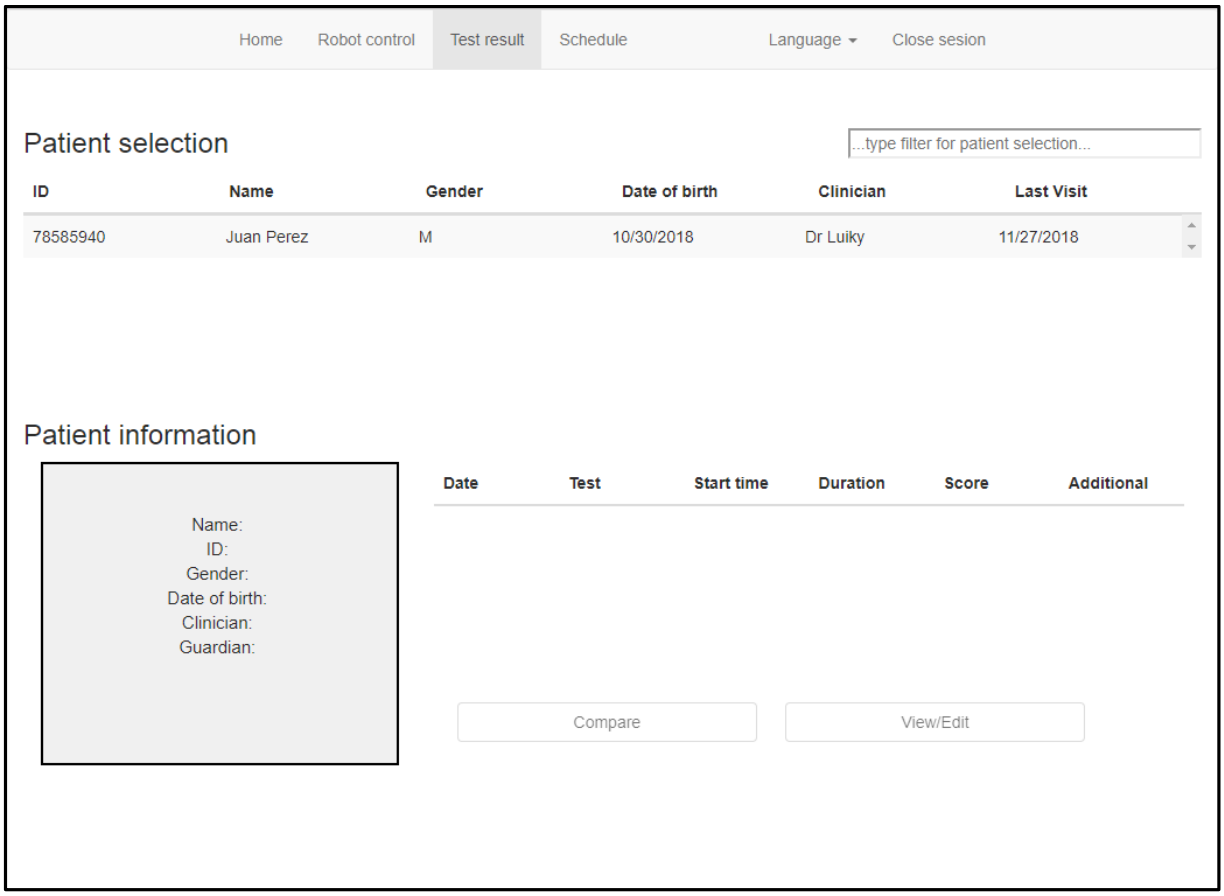
User's Manual - Part I - Session management

9.2 The Test results service on the CGAmed

Overview

The **Test results interface** provides the physician with the tools for offline analysis of a recorded session: visualizing videos, editing the scores, comparing results from several sessions, or managing the automatic report generated to resume the session.

The Test results interface

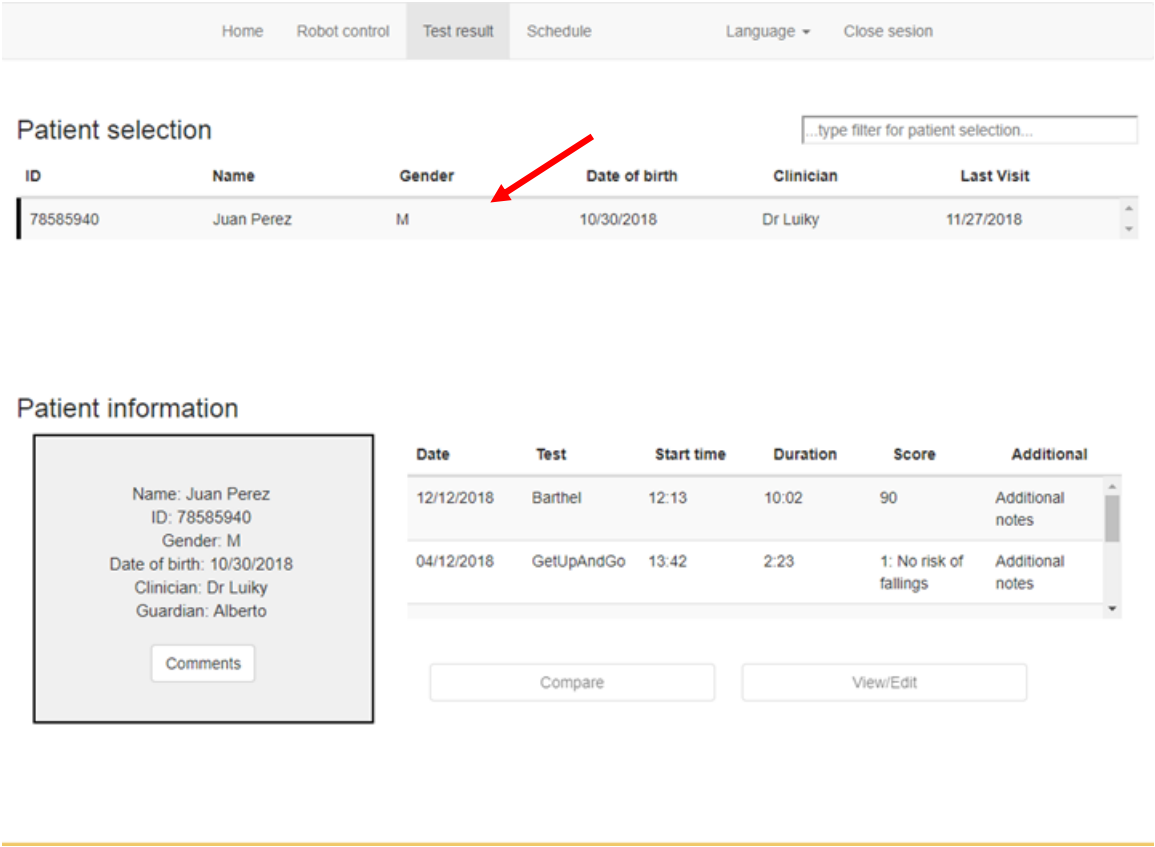


9.3 Analyzing a recorded session (Barthel example)

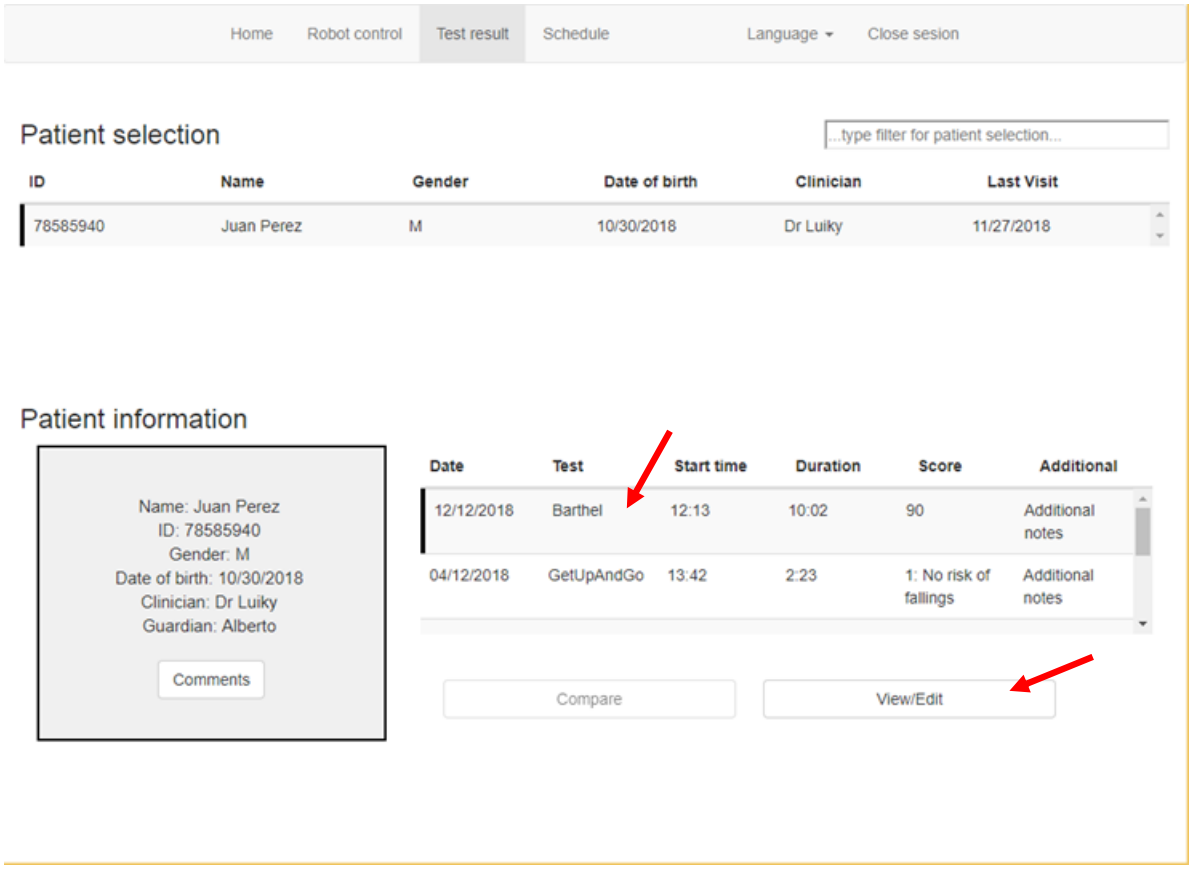
The information stored of a recorded Barthel test is presented in the **BarthelFaseIII.avi** video (from 8:07).

Once a session has been recorded, it is possible to analyse the stored information and editing the scores. The analysis of a Barthel test implies:

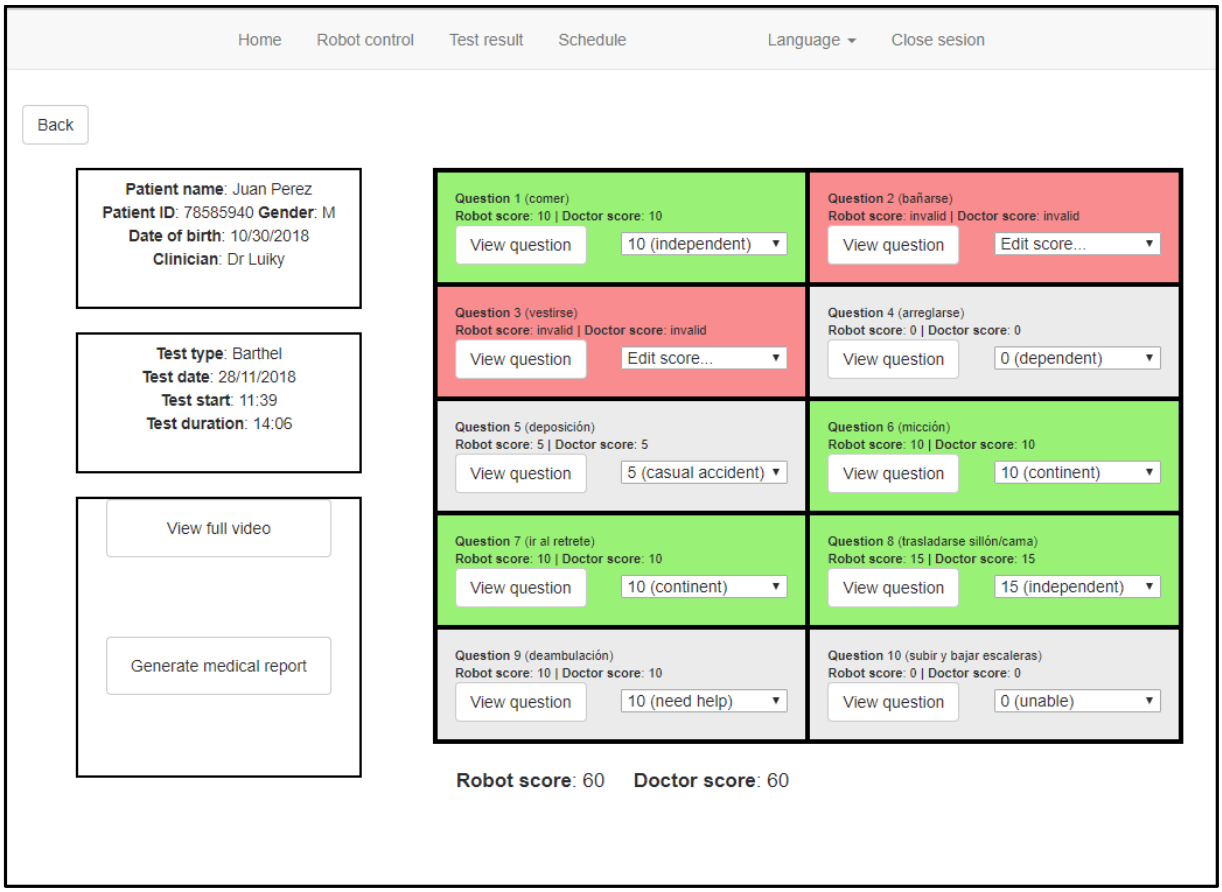
1. Selecting the patient of the desired session from the list shown in the Test results interface.



2. Selecting the test within the CGA session and clicking the View/Edit button.

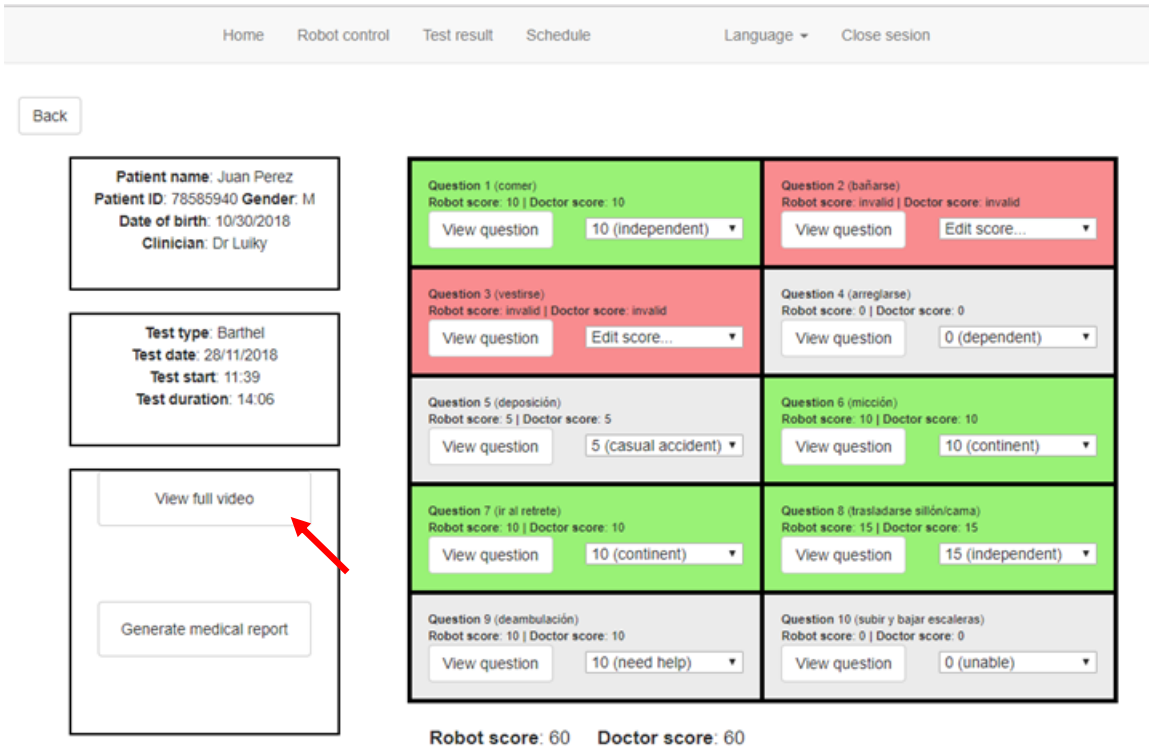

Home
Robot control
Test result
Schedule
Language
Close sesion
Patient selection
...type filter for patient selection...
ID
Name
Gender
Date of birth
Clinician
Last Visit
78585940
Juan Perez
M
10/30/2018
Dr Luiky
11/27/2018
Patient information
Name: Juan Perez
ID: 78585940
Gender: M
Date of birth: 10/30/2018
Clinician: Dr Luiky
Guardian: Alberto
Comments
Date
Test
Start time
Duration
Score
Additional
12/12/2018
Barthel
12:13
10:02
90
Additional notes
04/12/2018
GetUpAndGo
13:42
2:23
1: No risk of fallings
Additional notes
Compare
View/Edit

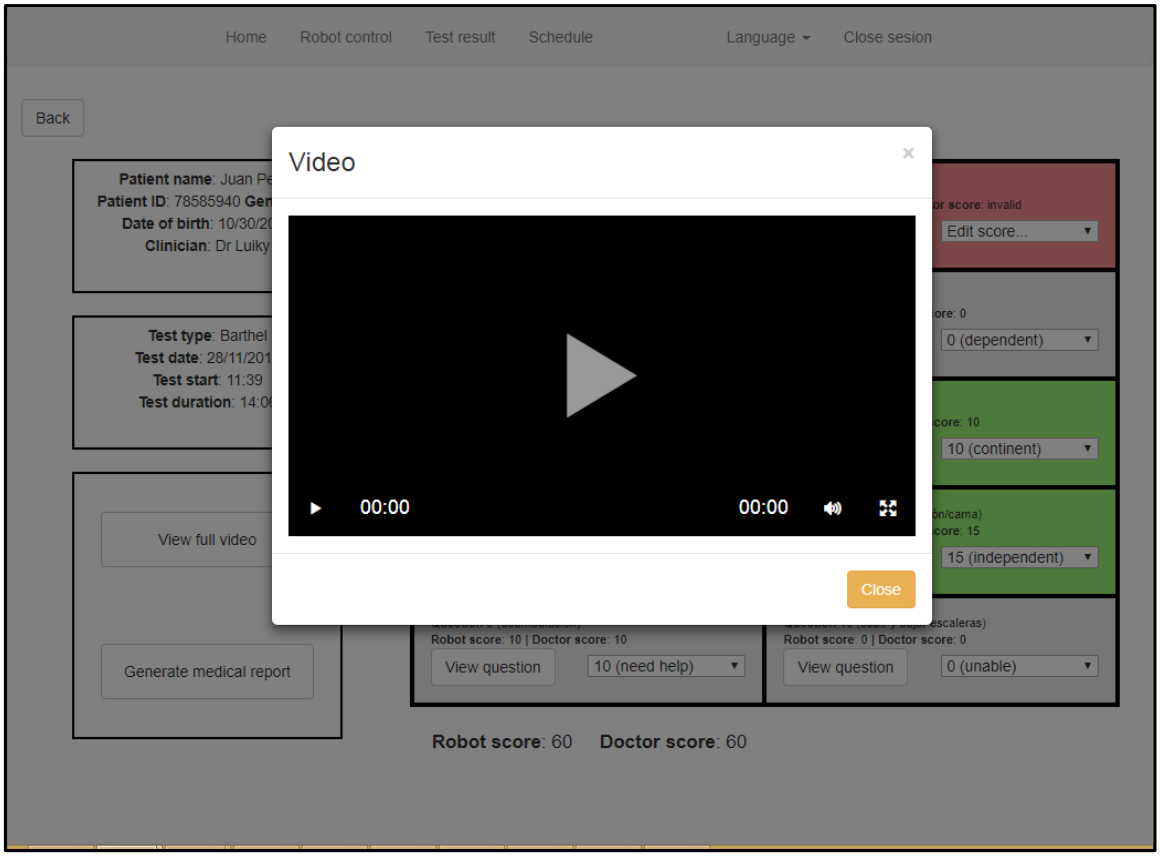
The Barthel test results interface



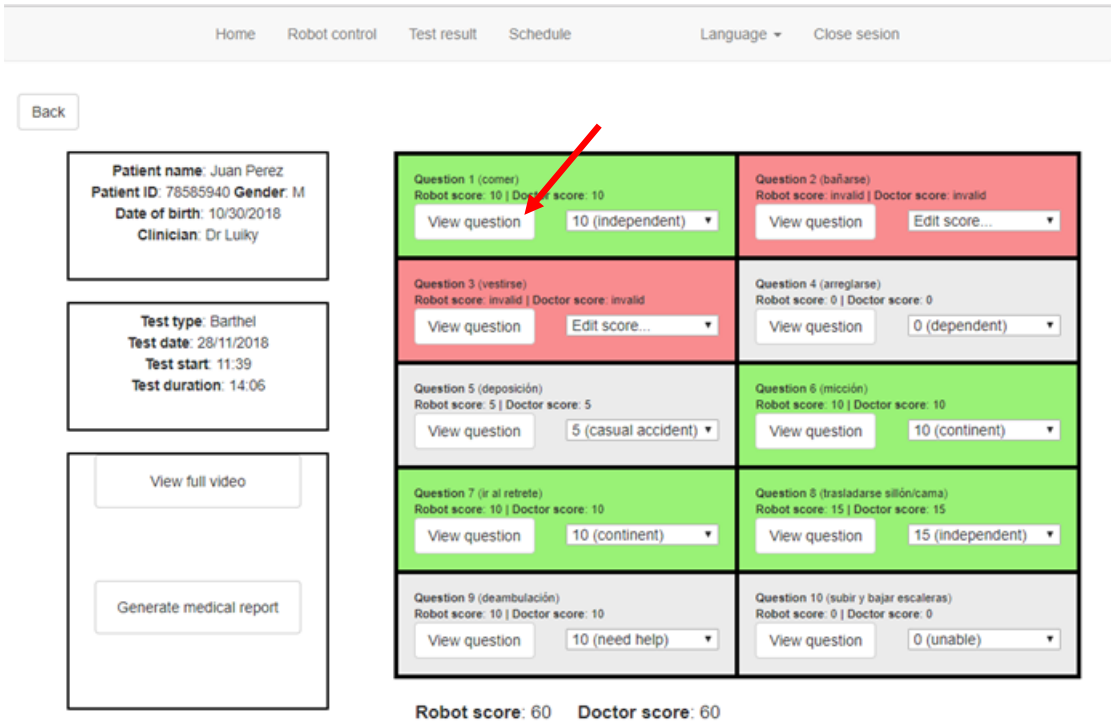
In this interface the questions not answered by the patient are shown in red, the questions in which he/she answered that is independent are shown in green and the rest of answers are shown in grey.

Option 1: See the full video of the test

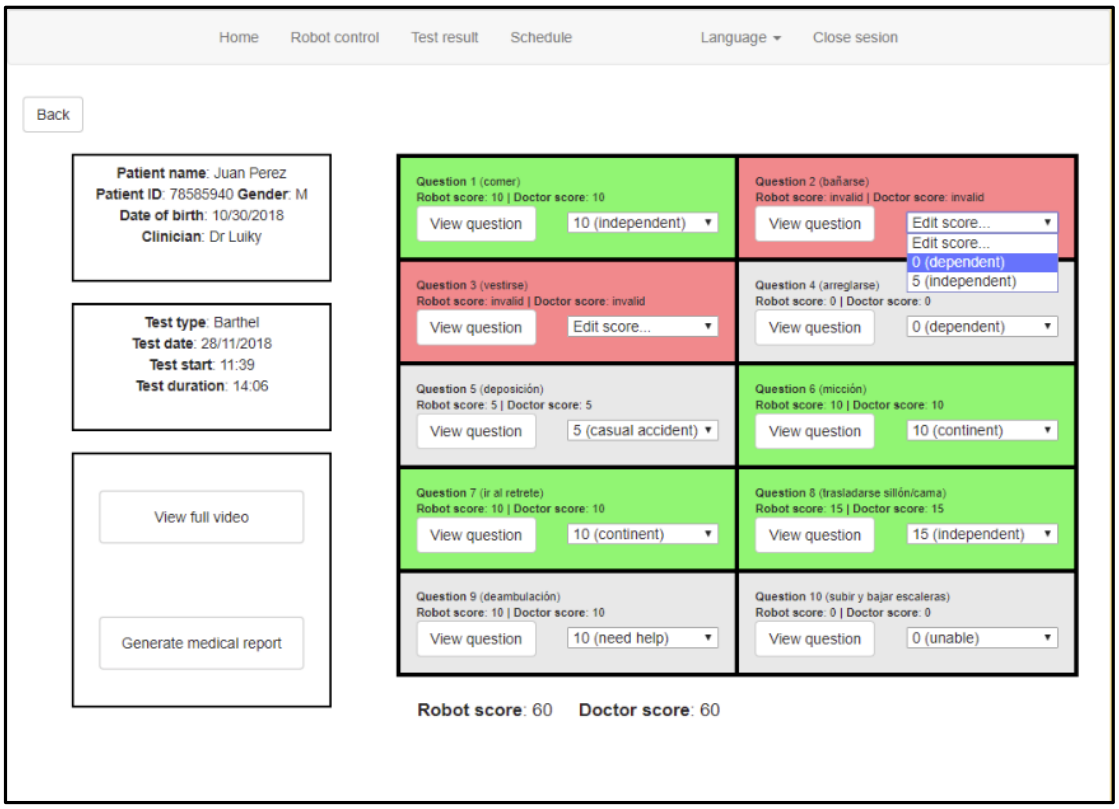




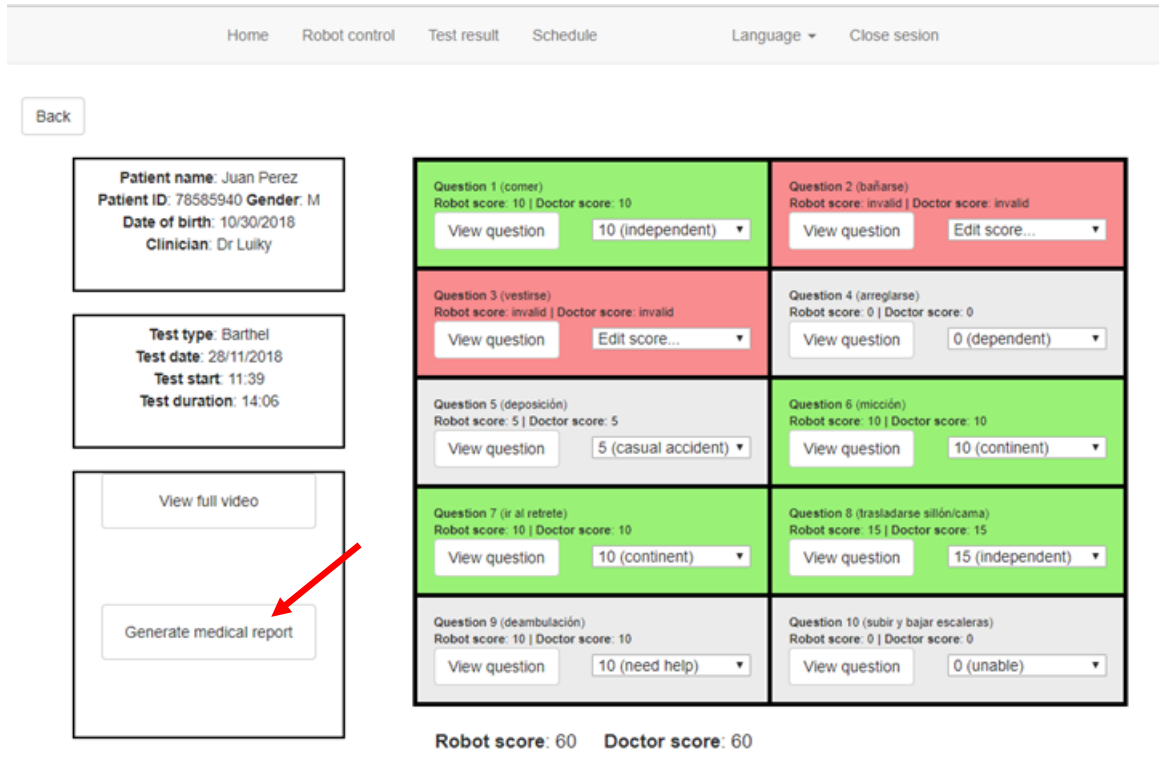
Option 2: See the video associated to a specific question on the Barthel test

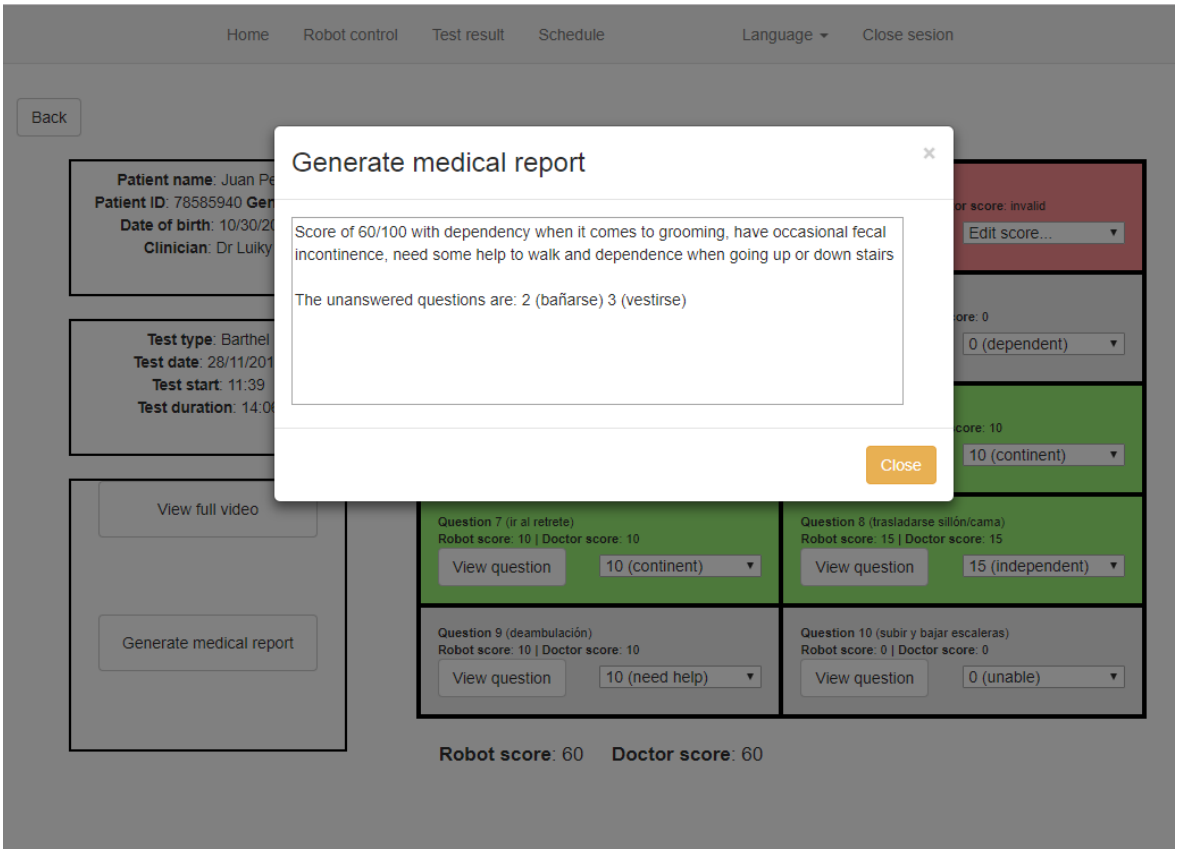


Option 3: Editing the score of a specific question on the Barthel test



Option 4: Generating the medical report that can be copied and pasted.



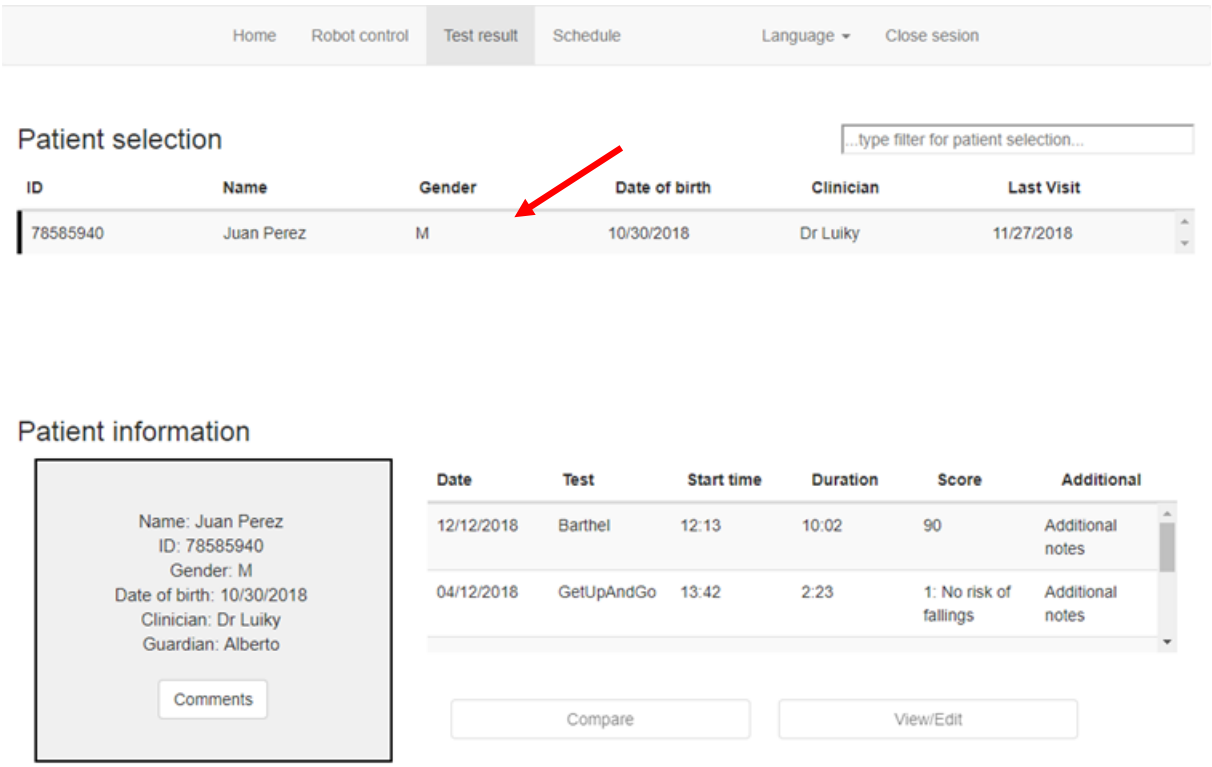


9.4 Analyzing a recorded session (Get up & Go example)

The information stored of a recorded Get up & Go test is presented in the [GetUpAndGo.avi] video (from 2:07).

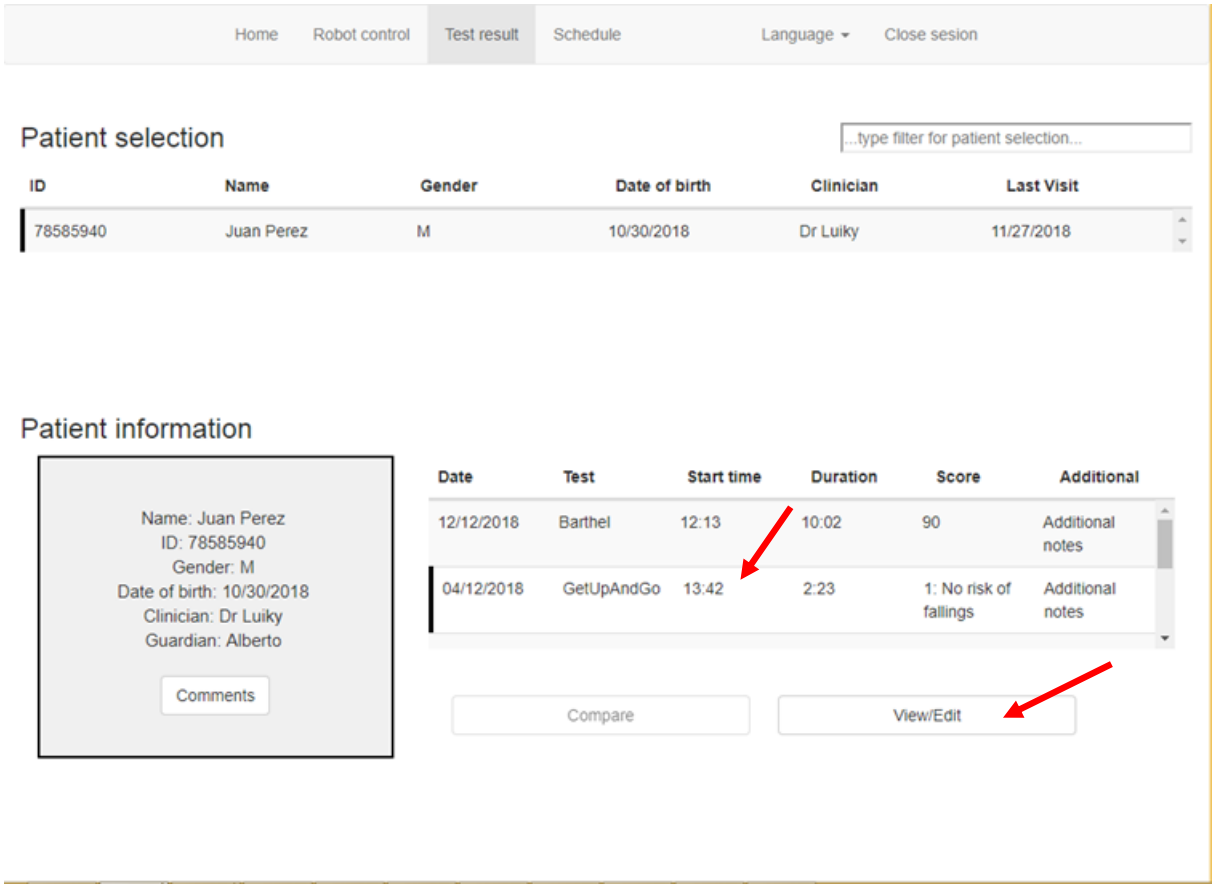
Once a session has been recorded, it is possible to analyse the stored information and editing the scores. The analysis of a Get up & Go test implies:

1. Selecting the patient of the desired session from the list shown in the Test results interface

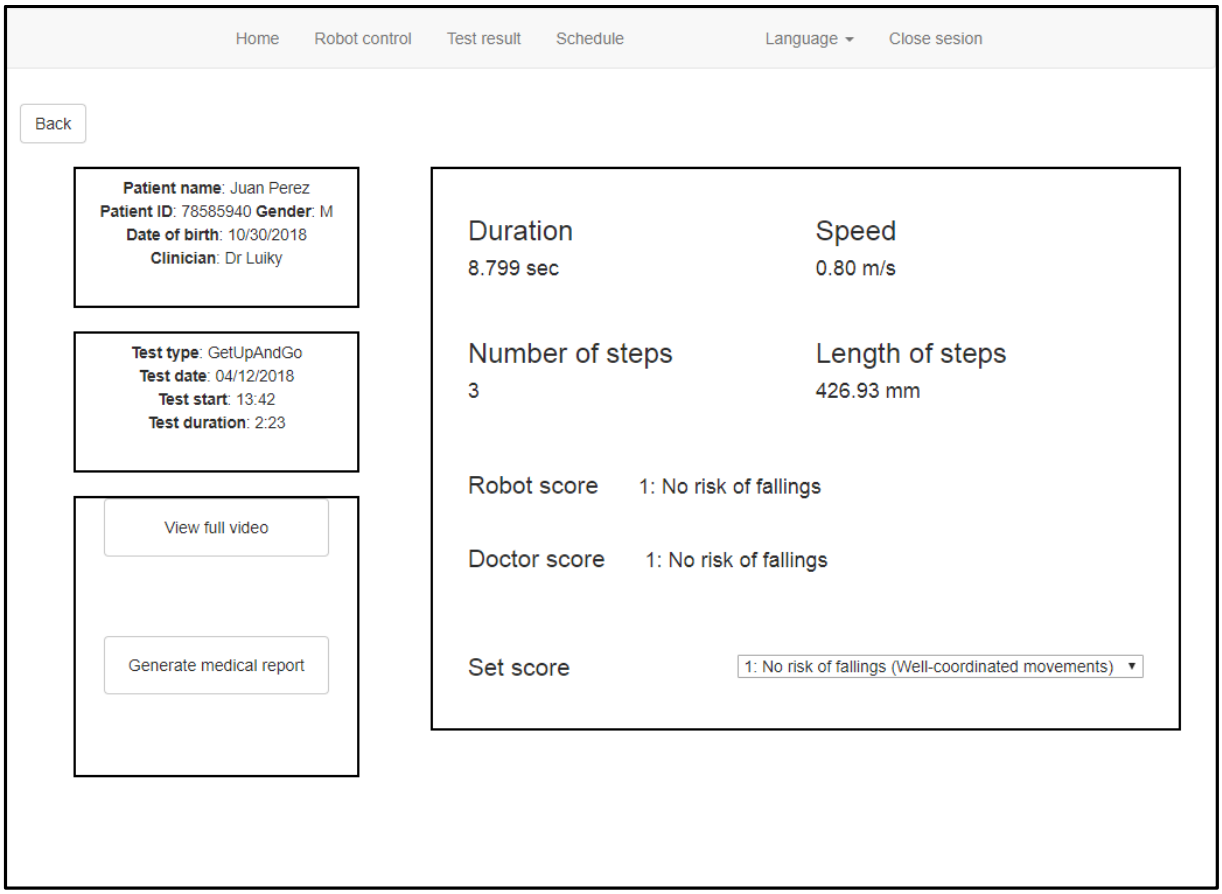


---

2. Selecting the test within the CGA session and clicking the View/Edit button.



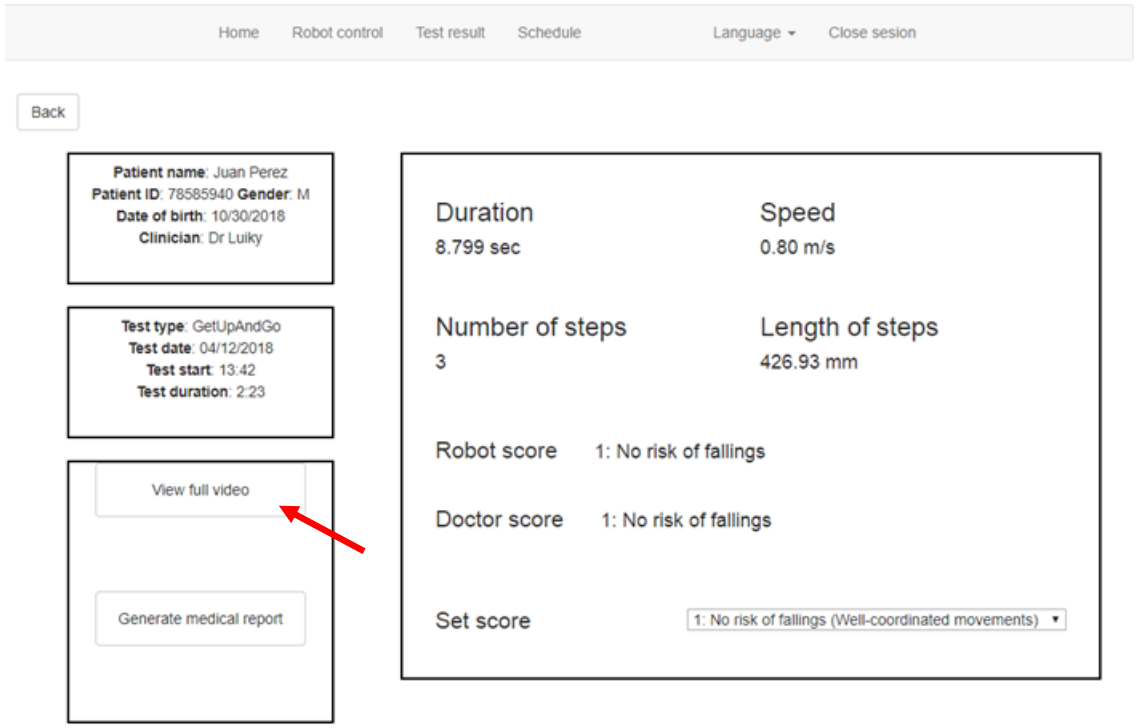
The Get up & Go test results interface

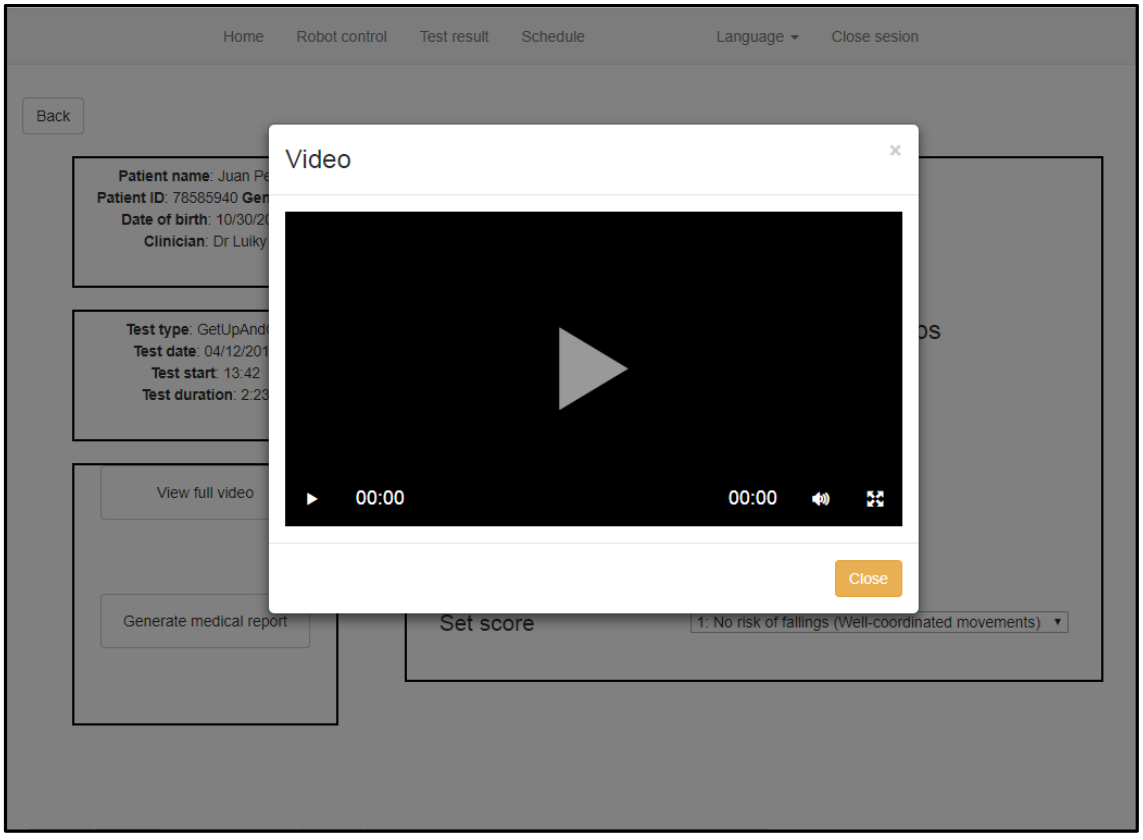


In the big square region of this interface, some information about the execution of the test is shown:

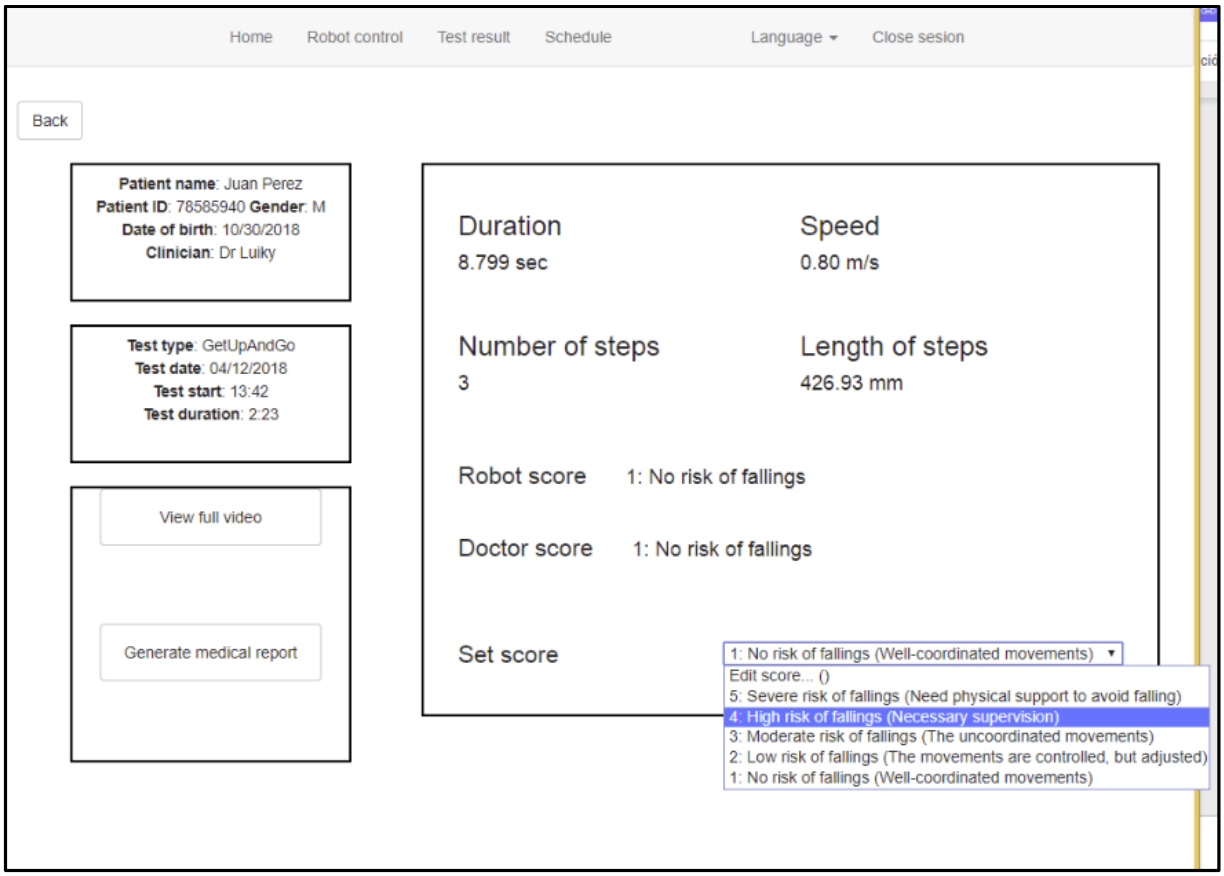
- **Duration:** The time used by the patient to perform the test. It can be used to evaluate the Time Up & Go.
- **Speed:** The mean walking speed.
- **Number of steps:** The number of steps made in the first straight walk.
- **Length of steps:** The mean length of a step.

Option 1: See the video of the test

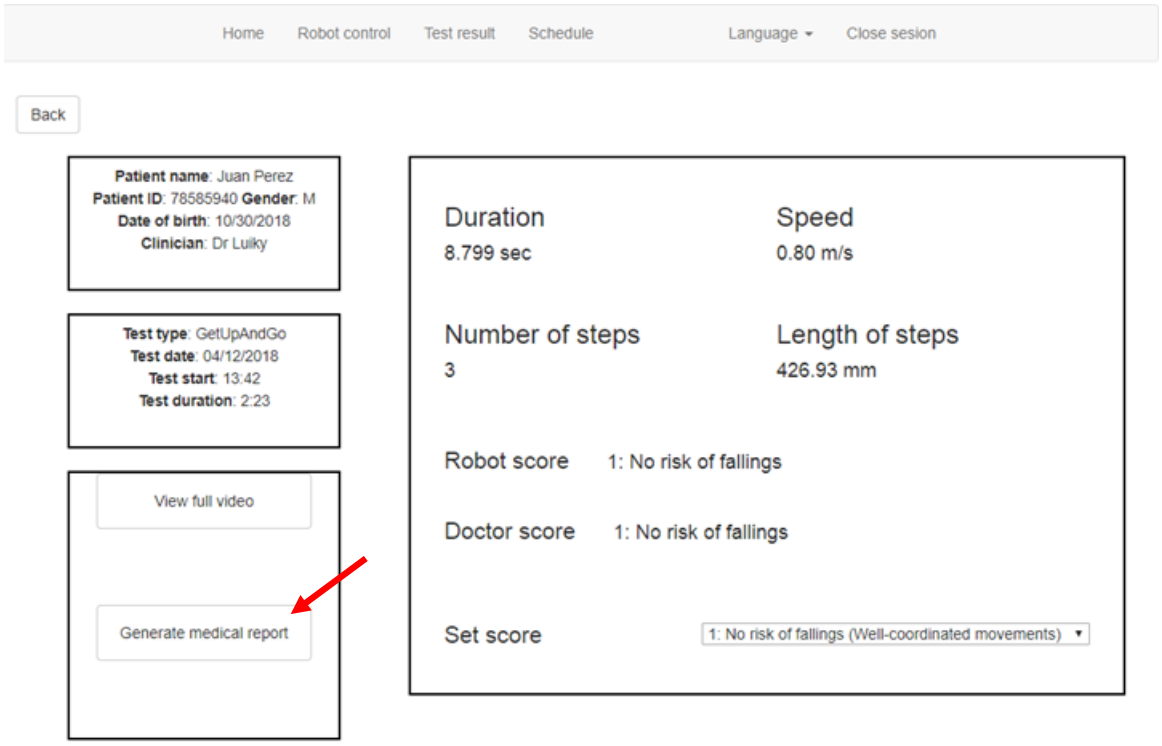


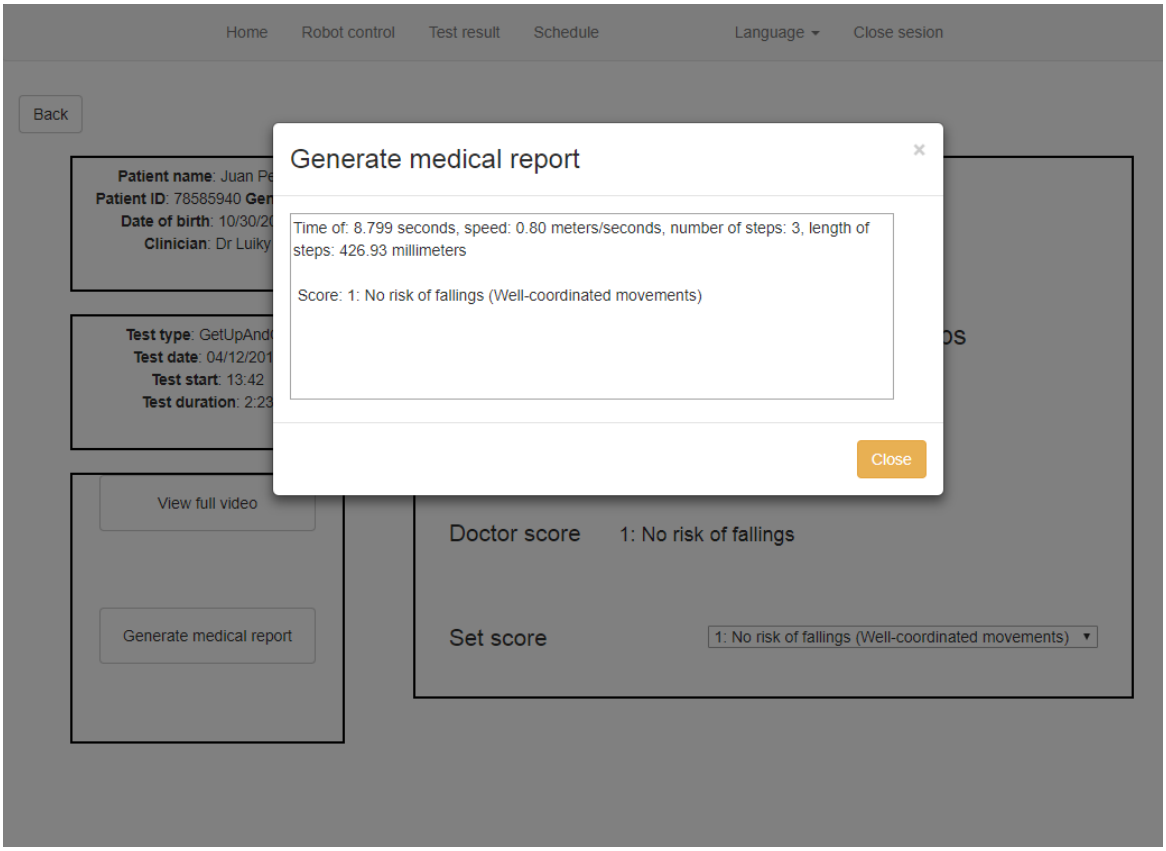


Option 2: Edit the score


Home
Robot control
Test result
Schedule
Language
Close sesion
Back
Patient name: Juan Perez
Patient ID: 78585940 Gender: M
Date of birth: 10/30/2018
Clinician: Dr Luiky
Test type: GetUpAndGo
Test date: 04/12/2018
Test start: 13:42
Test duration: 2:23
View full video
Generate medical report
Duration
8.799 sec
Speed
0.80 m/s
Number of steps
3
Length of steps
426.93 mm
Robot score
1: No risk of fallings
Doctor score
1: No risk of fallings
Set score
1: No risk of fallings (Well-coordinated movements)
Edit score... ()
5: Severe risk of fallings (Need physical support to avoid falling)
4: High risk of fallings (Necessary supervision)
3: Moderate risk of fallings (The uncoordinated movements)
2: Low risk of fallings (The movements are controlled, but adjusted)
1: No risk of fallings (Well-coordinated movements)

Option 3: Generating the medical report that can be copied and pasted.

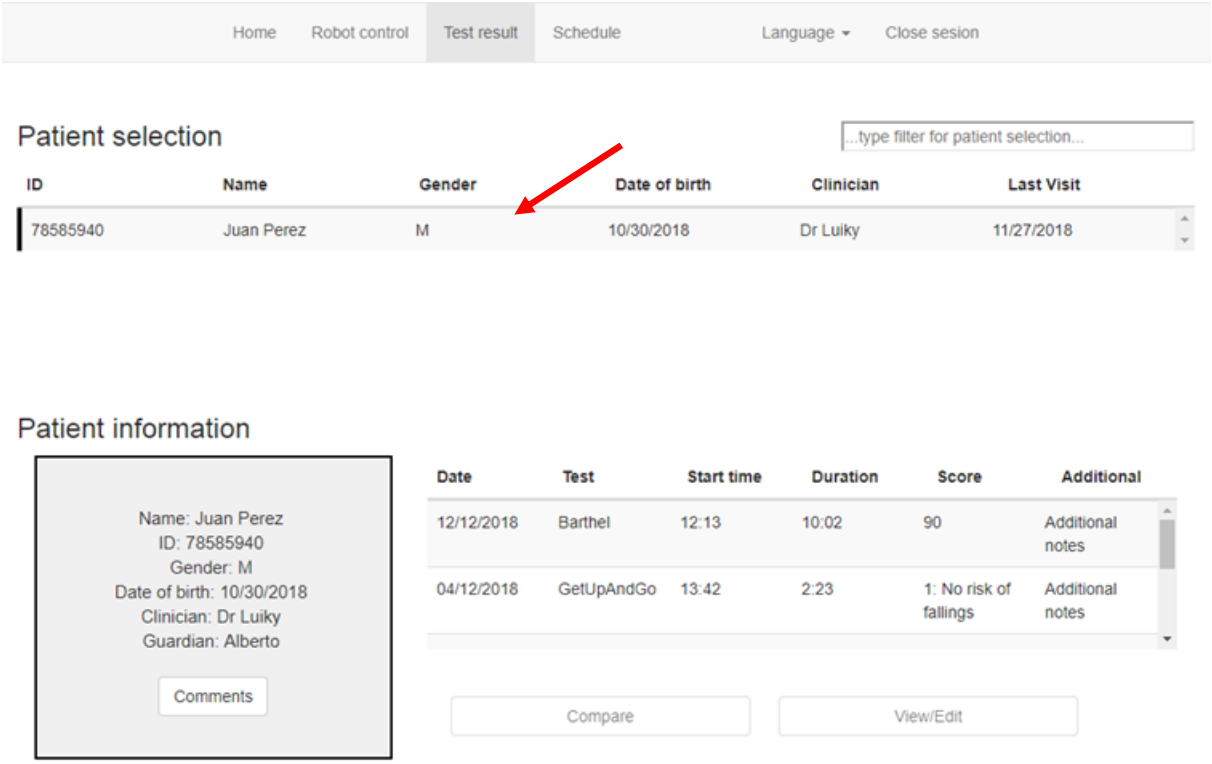




9.5 Comparing the results of several tests

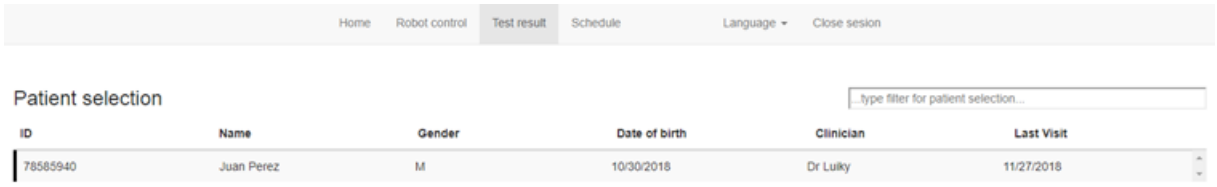
Once several sessions of a same patient have been recorded, it is possible to compare the results obtained for the same type of tests. The comparison of several tests implies:

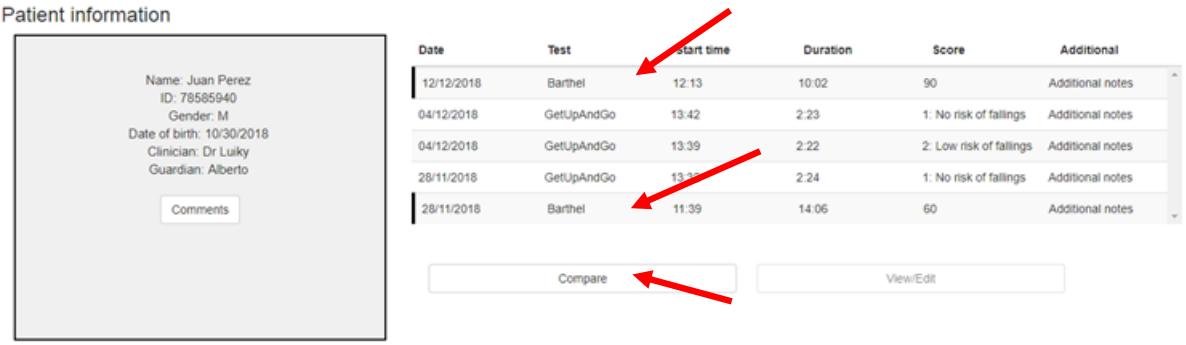
1. Selecting the patient of the desired session from the list shown in the Test results interface

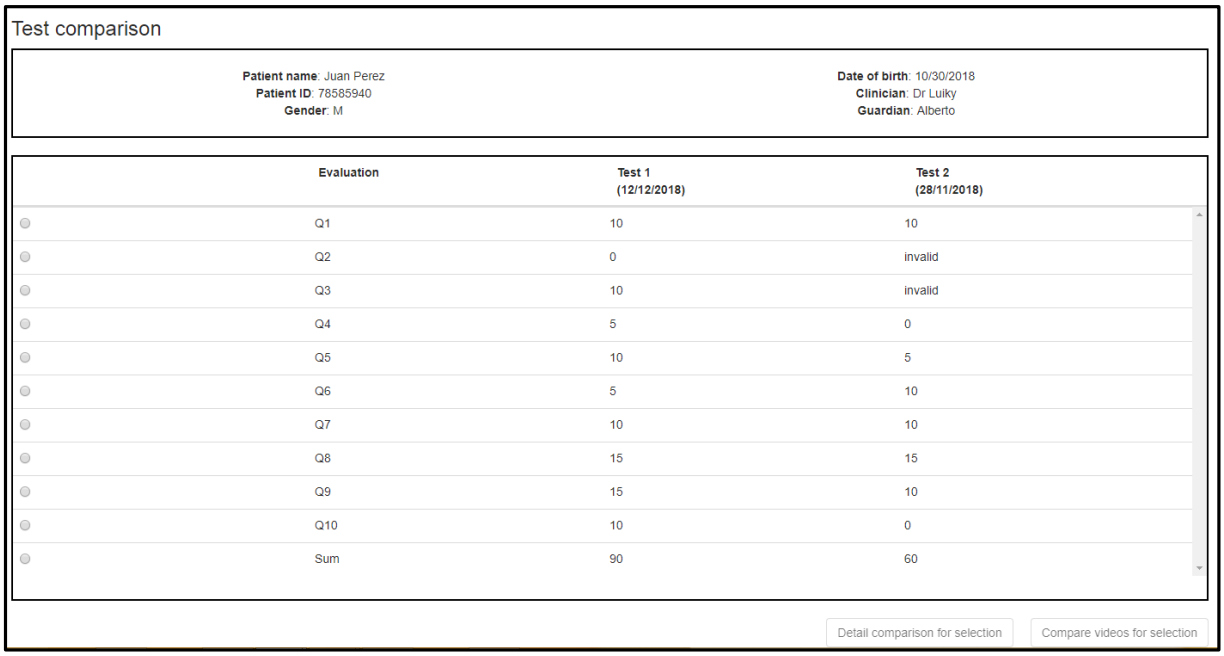


2. Selecting the tests which are going to be compared and clicking the “Compare” button.

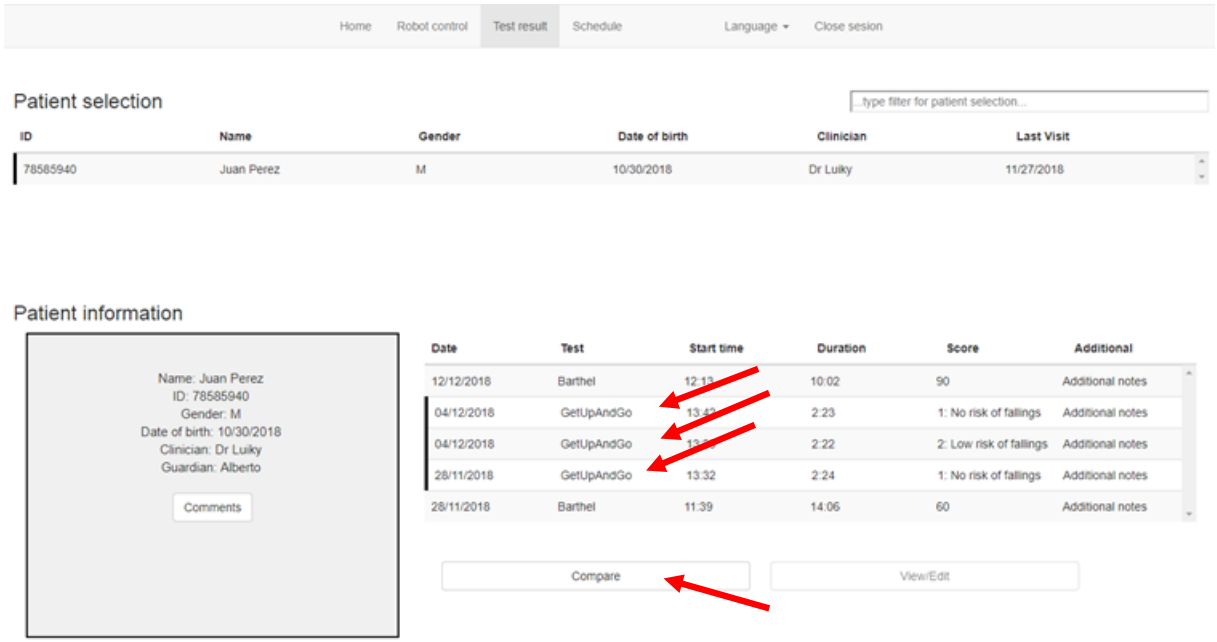
Barthel example

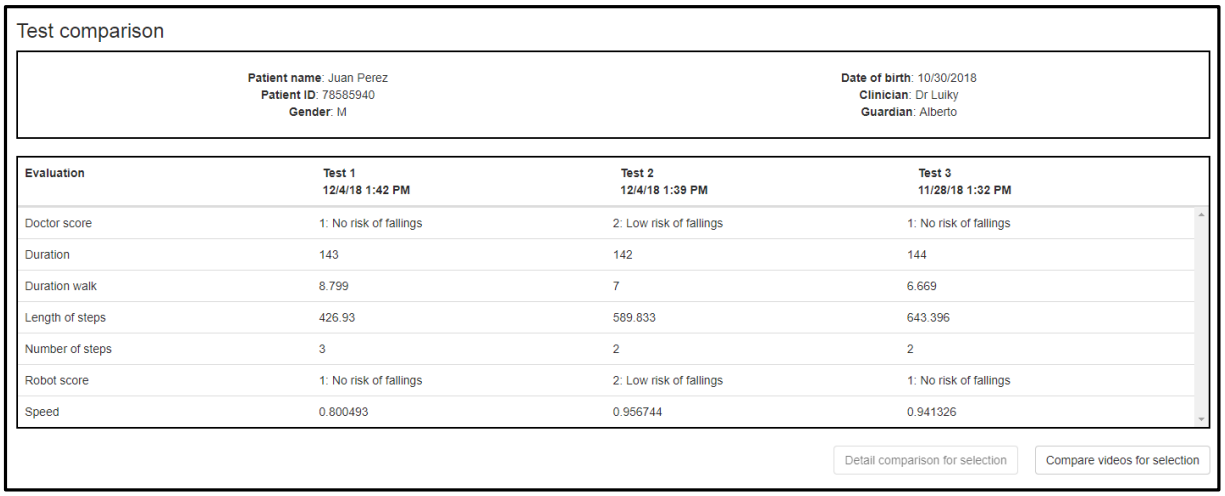






Get up & go example





10 Bugs

Module	Problem
CGAmed	The IP Address is currently the same in all CGAmed stations. This will provoke conflicts when several robots work in the same environment.
CGAmed (Administration web)	The current version of the Administration web only runs in Spanish.

Part III - System installation

About this manual

This manual describes how to deploy a **CLARC framework**.

Usage

This manual describes how to deploy the CLARC framework. It must be carefully used each time the CLARC framework is deployed in a new scenario, or when significant changes have been accomplished in the scenario (because of moving wardrobes, tables, etc.).

Who should read this manual?

This manual is intended for those technicians in charge of deploying the CLARC framework.

Prerequisites

The reader should have basic knowledge of:

- Windows and Ubuntu operating systems, including network configuration
- Configuration of routers

Organization of chapters

The manual is organized in the following chapters:

Chapter	Title	Contents
1	Introduction	Contains information about the hardware elements in the CLARC framework, and basic explanations of the terms and concepts needed for achieving a successful deployment of the framework
2	Setting the CLARC framework	Describes how to capture the map of the environment and set the goal poses, those from where the robot will interact with the patients for conducting the CGA sessions
3	Administration	Describes the Administration web on the

	facilities in the CGAmed	CGAmed

11 General information

11.1 Terms and concepts

CLARC is a complete framework for robotizing two specific tests that are typically part of a Comprehensive Geriatric Assessment (CGA) procedure: the Barthel test and the Get Up & Go test. CLARC consists of two major elements: **CLARA**, a social robot able to interact with the patients, and capture and analyze the obtained data; and the **CGAmed**, a local server able to store a database with all captured data and to provide the physicians with the tools for online monitoring and offline editing and supervision. CLARC provides all hardware items and do not require any specific constraint to be deployed.

11.1.1 Hardware concepts

Overview

This section introduces the hardware in the CLARC framework

Additional information

A detailed description of the CLARC framework can be found in

CLARC - Functional prototype

Standard hardware

The table below describes the standard hardware in an CLARC framework

Hardware	Explanation
CLARA robot	The robot is based on a differential driven platform by MetraLabs. Main components are listed in Section **The CLARA robot**.
Charging station	The robot has a charging station to be able to charge autonomously. The charging station is powered by standard main supply. In case of charging the power output is 400 W.
Remote Control	Portable device connected to the robot that allows the user to interact with the system using large buttons.

Router	CLARC works in a **local network**, in which all the components are connected to the wifi provided by this router.
CGAmed embedded PC	This PC stores all the information about users, sessions, etc.

Optional hardware

The table below describes the optional hardware in an CLARC framework

Hardware	Explanation
Remote Control (XL size)	Portable device connected to the robot that allows the user to interact with the system using large buttons and a small touchscreen.

The CLARA robot

The table below describes the standard hardware in the CLARA robot

Hardware	Explanation
The motors & gearboxes	
MetraLabs HG4 main control unit	Safety motor controller and power supply, battery charging
Battery 40 Ahrs	
Bumper	Stops the robot in case of collision
Safety LIDAR	Measures distances to walls for orientation, measures distances to obstacles to avoid collisions, reduces the velocity of the robot if it is close to a person
Embedded PC Shuttle DH170	Linux based PC that runs the CORTEX architecture and CogniDrive
Embedded PC Intel NUC	Windows based PC for person detection, human motion capture and speech recognition
Microsoft Kinect2	Sensor for motion detection
Network camera Edimax IC-3115W WiFi	IP camera for online supervision
Webcam Logitech C310 HD Logitech	Webcam for recording the session
Soundkarte USB 2.0 ROCCAT	Converts USB to Microphone

Display 13,3" with PCAP-Touchpanel	Touchscreen for tactile interaction
Shotgun Microphone	Directional microphone for speech capture
Speakers	

11.1.2 The CGAmed server

Overview

This section introduces the webs in the CGAmed station

Webs in the CGAmed

The table below describes the webs in an CGAmed station

Web	Explanation
Administration 192.168.0.70	The administration web is used to configure ● The positions where the robot is going to perform the tests ● The list of patients ● The IP address of the camera for online supervision mounted on CLARA robot (Section 1.1.1 - The CLARA robot)
CGAmed 192.168.0.70/cgamed	The CGAMed is used to: ● Add new patients. ● Add new sessions. ● Start/Stop a session. ● Pause/Resume a session. ● Move the robot to a position (from a list of predefined ones). ● See and compare the results of the tests.

11.3 User Access

The table below provides the default user/password data needed to access to the modules in the system.

Module	Access
Linux based PC (CLARA)	Password: scitos
Windows based PC (CLARA)	Accessible from the Linux based PC using the Remmina remote desktop app
CGAmed embedded PC	User / password: isis / grupoisis
CGAmed	URL (CGAmed) 192.168.0.70/cgamed user / password: adminWeb / adminSecure URL (Administration) 192.168.0.70 user / password: admin / adminRobot

Note: All CGAmed stations share currently the same IP Address. Contact us if you need to change this address, as this change implies internal updates on the modules on CORTEX architecture.

12 Setting the CLARC framework

12.1 The CLARA robot

12.1.1 Turning on the robot

Overview

This section explains how to turn on the robot.

The video **Starting.mp4** explains how to turn on the robot. As the video shows, when you turn on the robot using the key, the two internal PCs are automatically turned on.

Note: Once the PC's are on, you must connect the Linux based PC to the local wifi network provided by the router.

12.1.3 Capturing the map of the environment

Overview

The first time that the robot is going to be deployed in a new place, it is necessary to build a map of the environment. This will be addressed using MIRA and the CogniDrive application from MLAB.

Additional information

https://www.mira-project.org/MIRA-doc/toolboxes/MapBuilder/MappingIntroduction.html
https://www.mira-project.org/MIRA-doc/toolboxes/MapBuilder/SimpleMapperPage.html

---

The process to build the map is explained in the **Mapping.ogv** video.

Capturing the map

1. **Opening the MIRA mapping application:** run the following command in a shell:

```
Miracenter SCITOSConfigs:etc/SCITOS-mapping.xml
```

2. **Setting the origin of coordinates:** the initial position of the robot will be stored in MIRA as the origin of coordinates of the environment

Note: It is recommended to mark this initial position on the floor, because it will be used many times to localize the robot, as it will be explained later in this manual

3. **Recording a map:** in the MIRA top menu, click the “Window” menu and then the “Show view” tag. Select the “Simple Mapper” view. Click on the “Record” button and move the robot manually around the place to allow MIRA catching the information to build the map. The robot can be moved by pushing it, using the keyboard arrows, or the arrows of MIRA application (to do it, you must put the focus on the arrows section of MIRA by clicking there). The movement of the robot around the place must to finish at the same position where it started (the initial position). After that, you can stop the recording by clicking the “Finish” button of the Simple Mapper.

Note: When the robot is moved to record the map, please, be careful not step in front of the robot, to avoid your legs to be mapped as obstacles!!

4. **Saving the map:** the result of the previous step is a map that must be saved to use it later. You have to save it as a MCF file using the “Save MCF” button in the “Simple mapper” view.

12.1.4 Editing the map

Overview

Once the map has been built, you can edit it for erasing noise and adding areas of NOGO (an area which the robot must no enter).

Additional information

https://www.mira-project.org/MIRA-doc/toolboxes/MapBuilder/SimpleMapEditorPage.html
https://www.mira-project.org/MIRA-doc/toolboxes/MapBuilder/MCFReference.html

---

The map can be edited to remove noise or to mark places in which the robot is not able to enter, or where we don't want the robot to go. This is explained in the **Editing map.ogv** video.

To edit the map open the "Simple map editor" view in the MIRA mapping application (Window->Show view).

12.1.5 Localizing the robot in the map

Overview

When a new map of the environment is built, it is necessary to localize the robot in that map as a previous step to define the goal positions for the tests. This step is also necessary when an error in the localization of the robot is detected.

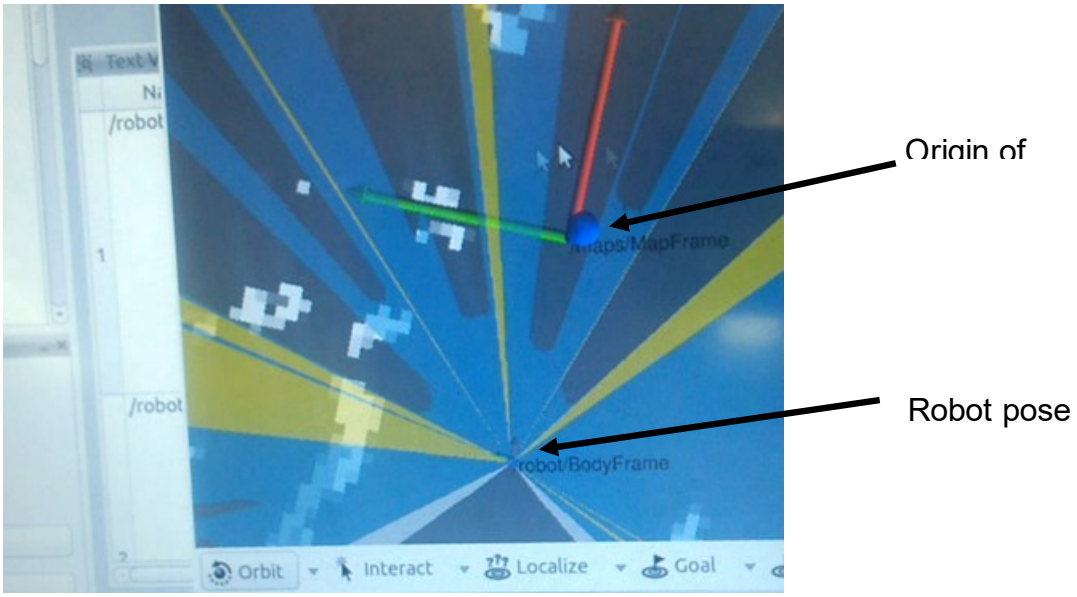
> **Note:** A localization error can occur if the robot is moved by hand by pushing it. Because of that, it is recommended don't move the robot by pushing it. It can be moved using the keyboard or the arrows in the MIRA application.

The process to localize the robot is explained in the **Setting up MIRA.mp4** video.

Localizing the robot in the map

1. The first step is to turn on MIRA using the previously built map. The easiest way to do it is making a shell script (.sh file) as the one shown in the **Setting up MIRA.mp4** video.

   When MIRA starts and load the map, we could see the position of the origin of coordinates (the initial position of the robot in the mapping process) marked with a big coordinate axes, and the position estimated by MIRA for the robot, marked with a small coordinate axes.



2. The easiest way to localize the robot is to physically place it in the environment position corresponding to the origin of coordinates, and then, set in MIRA that the robot is in the origin of coordinates. To do it, first, in the "Text View" of MIRA look at the "Pose" of the robot odometry. If this pose is different from (0.0,0.0,0.0), click the Reset Odometry button. After that, click in the "localize" button and then, in the origin of coordinates of the map to make a correspondence between the axes of the origin of coordinates and the axes of the robot. Now the robot is localized in the origin of coordinates. After that, it is good to make the robot spin to improve the localization of the robot in the map. Remember, for moving the robot you can use the arrows of the keyboard or the arrows of MIRA application (to do it, you must put the focus on the arrows section of MIRA by clicking there).

Note: It is important, when you turn on the robot, to always check that the robot is well localized. You can do it by starting MIRA using the built map and visually check in the map that the robot is well located with respect to the origin of coordinates.

12.1.6 Setting goal poses for Barthel and Get Up & Go tests

Overview

The first time that the robot is going to be used in a new place, and after the localization step at 2.1.3, you must select the poses (position and orientation) on the map where you want that the robot be for doing the tests. You have to store these positions in two places: in the robot components and in the CGAMed database.

Goal poses

There are three types of goal poses to be defined in the CLARC framework. They are listed in the Table below. The setting of the base_station pose will be described in Section 2.1.5.

Goal pose	Explanation
getupandgo_test	The position on the room from which the robot captures the Get Up & Go test (observing how the patient performs the test)
base_station	The position from which the robot can autonomously access to the Charging station
habitacion_x	With x ranging from 1 to N, you can define different rooms in the environment. The robot goes to these positions for performing a Barthel test or introducing a Get Up & Go one.

Setting goal poses (CLARA robot)

1. Create a .txt file named “goalPositions.txt” in the cajasvaciasechord/etc folder of the Linux based PC. The structure of this file is the following:

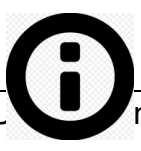
```
getupandgo_test -1.0 0.0 0.0

base_station 0.0 0.0 0.0

habitacion_1 -1.0 0.0 0.0

habitacion_2 -1.0 0.0 0.0

habitacion_3 -1.0 0.0 0.0
```



Note: The positions of all goal poses are in the form (X,Y,angle in radians). Please, use the same labels that in this example (you can adapt the number of rooms (habitacion_x) to your environment).

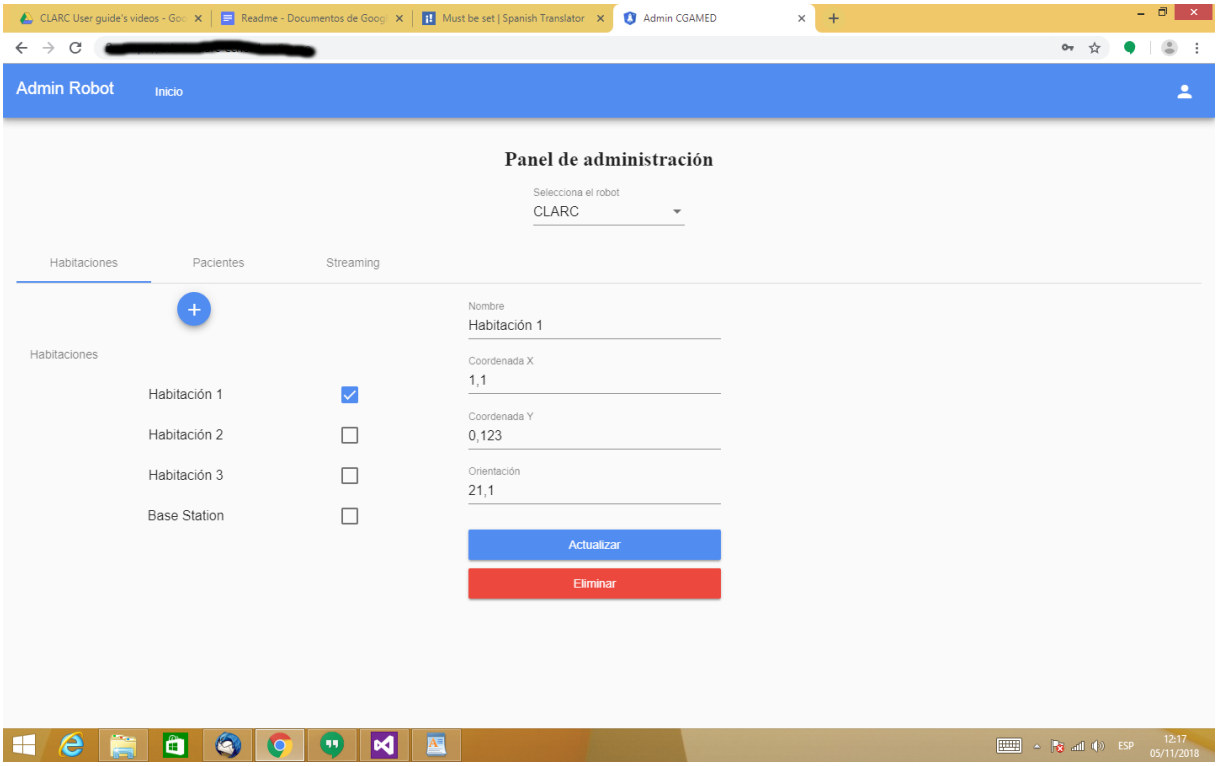
2. Start MIRA with your map (see Section 2.1.3).
3. Reset the odometry and localize the robot (Section 2.1.3).
4. Move the robot to the desired position using the MIRA arrows or the keyboard.
5. Look at the robot odometry pose in the text view of MIRA, there you have the X, Y coordinates of the position and also the angle.
6. Write the positions in the. txt file.

Note: Be careful because the angle in MIRA is in degrees and you have to convert it into radians.

Setting goal poses (CGAmed)

1. To configure de positions of the rooms and the base station in the CGAMed you need to access to the administration web (see Section 3.1.2).

Note: It must be noted that you do not need to write the getupandgo_test pose on the CGAmed.



2. Insert in the correct fields the same X,Y and angle values that in the “goalPositions.txt” file.

12.1.7 Setting the goal pose for the Charging Station

Overview

This Section describes the procedure for setting the pose of the Charging Station. It is different for the one described in Section 2.1.4.

Setting the goal pose for the Charging Station

1. Make sure you have a good map of the environment, which also includes the charging station.
2. Start "miracenter SCITOSConfigs:etc/SCITOS-application.xml" and localise the robot correctly (the localisation accuracy must be fine for teaching the charging station).
3. Push the robot onto the charging station, and make sure that it is firmly and centrally on the charging station, and most importantly, make sure that the robot charges! (Yellow LED on the charging station)
4. Use a 3D view with /maps/static/Map and /robot/frontLaser/Laser visualized, and make sure you can see the area around the charging station in the 3D view (i.e. where the laser is).

5. Use the "Station tool" (in the bottom bar of the 3D view) to create a new charging station (remember which ID you give it. We typically just use 0). Leave all the parameters (except for the ID) as they are set by default, then press "Teach".
6. You will now have to mark the area around the docking station in the 3D view using the mouse. Each left-click adds a corner of a polygon. Create a polygon around the outline of the charging station in the laser, and try to include all static and characteristic features of the environment that are in the immediate vicinity of the charging station. E.g. if the charging station is next to a corner, make sure the polygon includes the corner as well, as this will help the robot localise itself correctly when docking on.
7. When you are satisfied with the polygon, finish with a right-clíck. The robot will now start driving backwards and stop four times to record laser templates at different distances. After that he should tell you that he is finished, but I'm not sure. Either way, as soon as he stops for good, he is finished, and you can now dock on to that docking station from the point where the robot is standing right now (this is called the "base point").
8. The docking station will be saved in a file "stations.xml", which needs to be in the directory where you start MIRA from. You might have to exchange "SCITOS-Pilot.xml" for "SCITOS-application.xml" in your startup scripts, as only SCITOS-application.xml includes the docking stuff.

Note: You can read more about the general process and how to dock on in C++ here:

http://www.mira-project.org/MIRA-doc/domains/navigation/DockingProcess/index.html

From a procedural point of view, you'll have to drive to the "base point" of the station first using regular navigation. The robot will have to be located in front of the charging station roughly the same as when he finished recording the templates. Only then can you let the DockingProcess dock on to this station.

> **Note:** The “base point” is the pose that must be set as base_station (converting the degrees of the angle into radians) in the goalPositions.txt file and the CGAmed tool.

12.1.8 Updating the map in the start.sh script file

Overview

The first time that the robot is going to be used in a new place, and after the map of the environment has been built, the script file in charge of start all the components of the robot (start.sh) must be updated with the new map so that the robot knows its position during the sessions.

---

1. Within the Linux based PC in the CLARA robot, edit the start.sh script, which is in the “robocomp_clarc/robocomp/components/cajasvaciasechord/” folder.
2. Replace the name of the MCF file in the code line

```
qdbus org.kde.yakuake /yakuake/sessions org.kde.yakuake.runCommand 'cd ~ &&
source .bashrc && mira MiraNavigation:etc/SCITOS-application.xml MiraNaviga-
tion:etc/MiraNavigation.xml -v MCFFile=labPhaseTwo.mcf -p 1234'
```

with the name of your MCF map (see 2.1.2 Capturing the map of the environment).

13 Administration facilities in the CGAmed

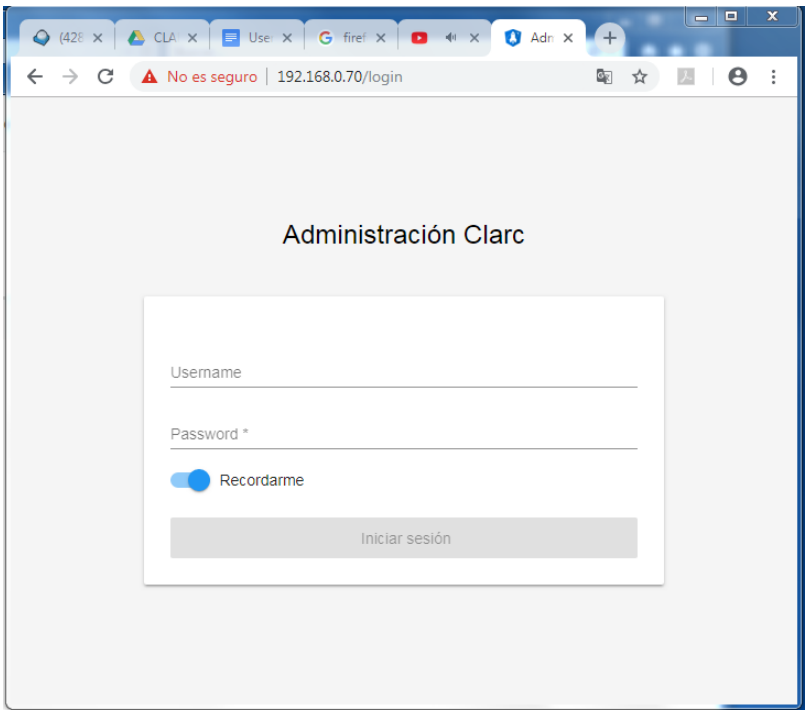
13.1 Login in the Administration web

Overview

This Section describes the procedure for entering within the Administration web on the CGAmed.

Login in the Administration web

When you connect to the URL of the Administration web (http://192.168.0.70), you need to add user and password for entering on the web. This data are provided in Section 1.2.



> **Note:** The current version of the Administration web only runs in Spanish.

13.2 Managing the list of goal poses

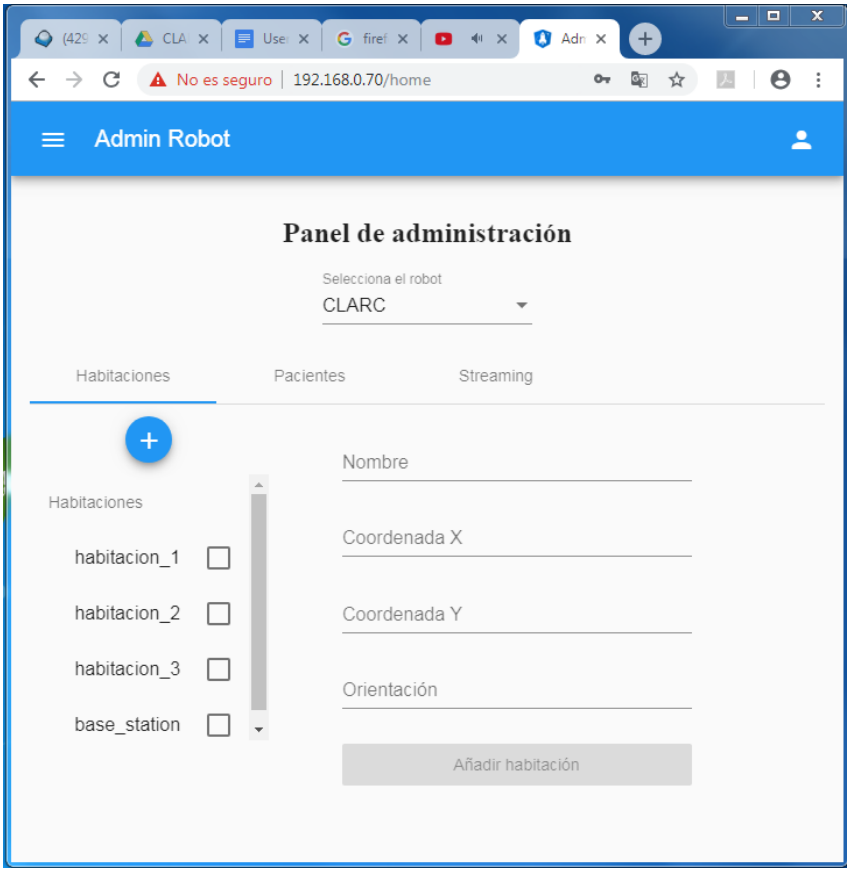
Overview

This Section describes the procedure for managing the list of goal poses within the Administration web on the CGAmed.

Managing the list of goal poses (CGAmed)

Once logged into the Administration web, clicking on the **Habitaciones** tab you have access to the list of goal poses (rooms and base_station).

➔ Clicking on the symbol you can add new poses. When coordinates and angle are added, you should click on the **Añadir habitación** tab.


No es seguro | 192.168.0.70/home
Admin Robot
Panel de administración
Selecciona el robot
CLARC
Habitaciones
Pacientes
Streaming
Habitaciones
habitacion_1
habitacion_2
habitacion_3
base_station
Nombre
Coordenada X
Coordenada Y
Orientación
Añadir habitación

13.3 Managing the list of patients

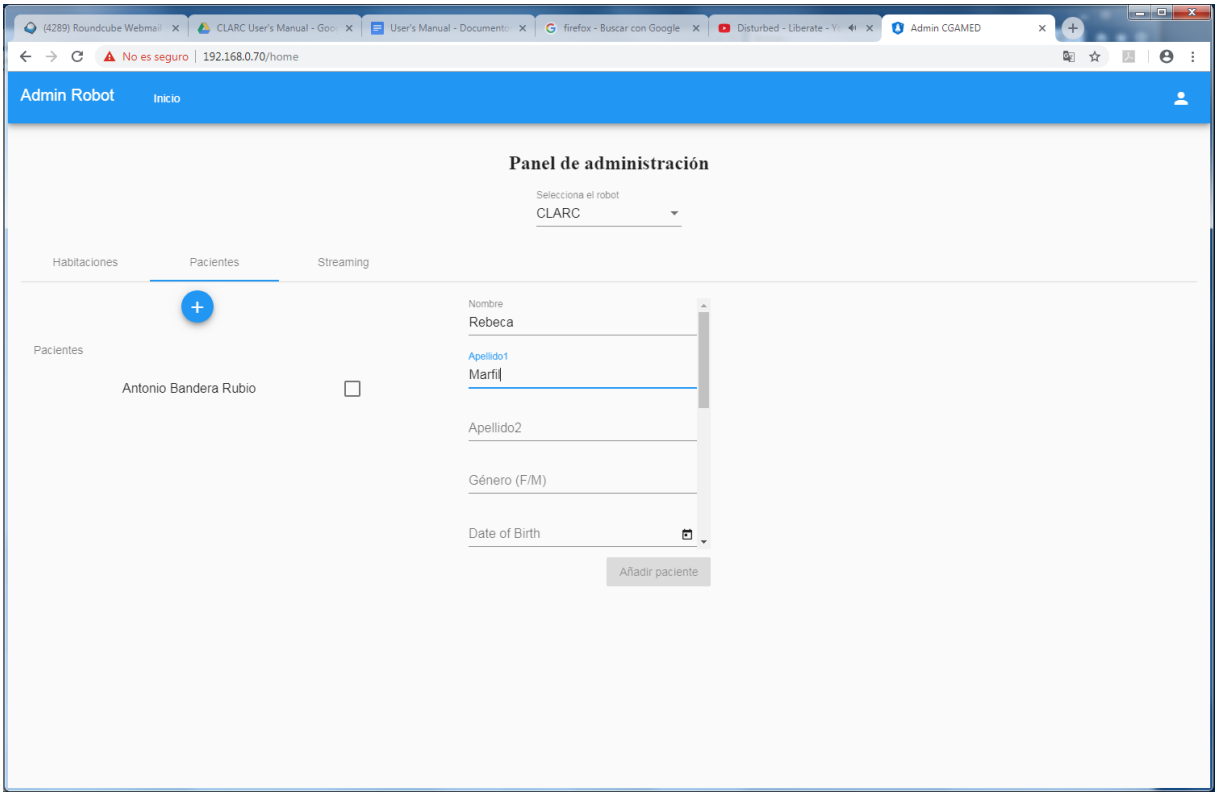
Overview

This Section describes the procedure for managing the list of patients within the Administration web on the CGAmed.

Managing the list of patients

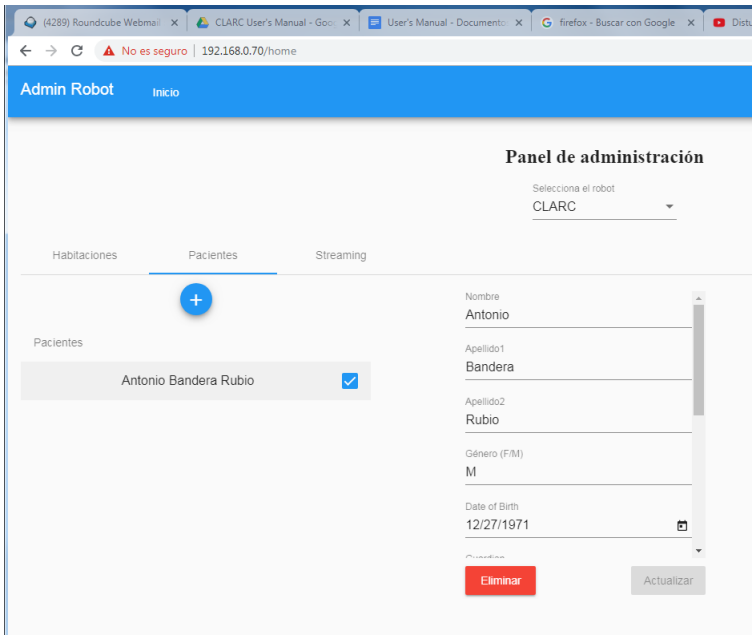
Once logged into the Administration web, clicking on the **Pacientes** tab you have access to the list of patients.

➔ Clicking on the symbol you can add new patients. When all data about the patient has been added, you should click on the **Añadir paciente** tab.



Note: The application does not allow to save a patient with incomplete fields!

➔ Clicking on the square-shaped box inline with the patient's name on the list, it is possible to edit the data stored about the patient. Clicking on the **`Eliminar`** tab is also possible to remove her/him from the list.



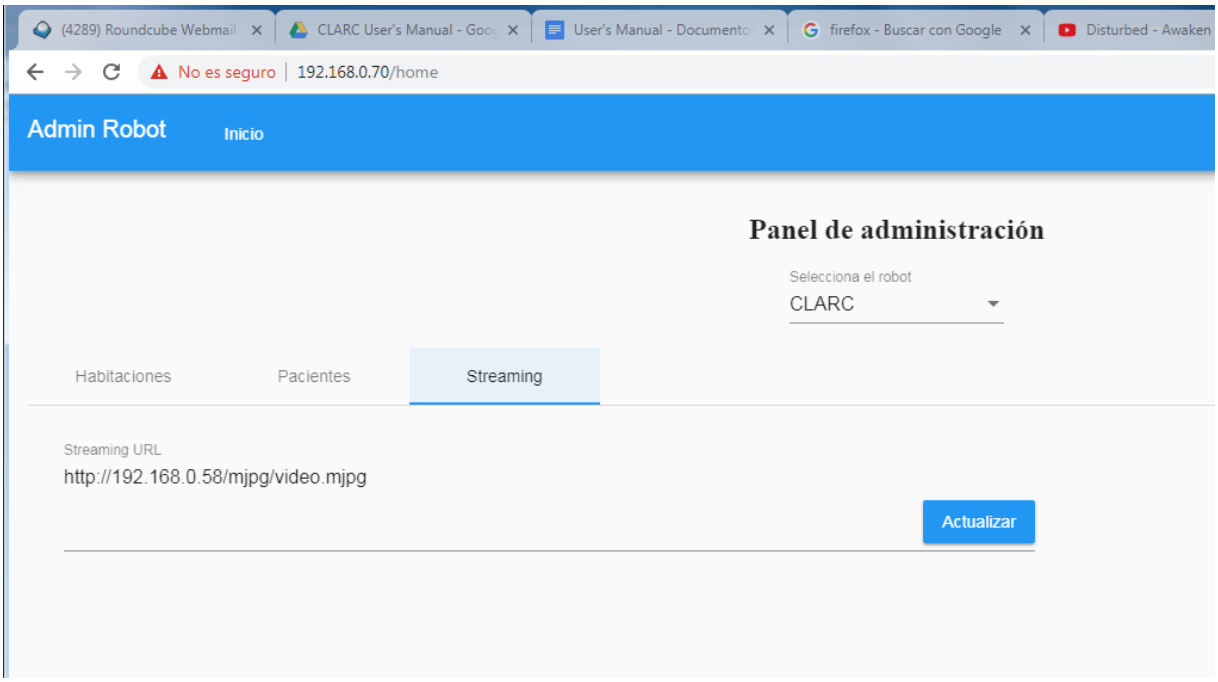
13.4 Video streaming

Overview

The CGAmed web offers the physician the possibility of monitoring, through video streaming, the CGA session. This video streaming is provided by an IP camera mounted on the CLARA robot. This Section describes the procedure for configuring the camera address within the Administration web on the CGAmed.

Configuration of the IP camera

Once logged into the Administration web, clicking on the **Streaming** tab you can update the URL address of the IP camera. If you change the URL, click on the **Actualizar** tab for approve the update.



14 Bugs

Module	Problem
CGAmed	The IP Address is currently the same in all CGAmed stations. This will provoke conflicts when several robots work in the same environment.
CGAmed (Administration web)	The current version of the Administration web only runs in Spanish.