



# Reference Guide (en)

Date: 09/2020

Revision: v.2.1



# MiR Robot interface

## Copyright and disclaimer

All rights reserved. No parts of this document may be reproduced in any form without the express written permission of Mobile Industrial Robots A/S (MiR). MiR makes no warranties, expressed or implied, in respect of this document or its contents. In addition, the contents of the document are subject to change without prior notice. Every precaution has been taken in the preparation of this document. Nevertheless, MiR assumes no responsibility for errors or omissions or any damages resulting from the use of the information contained.

Copyright © 2017-2020 by Mobile Industrial Robots A/S.

Contact the manufacturer:

Mobile Industrial Robots A/S

Emil Neckelmanns Vej 15F

DK-5220 Odense SØ

[www.mobile-industrial-robots.com](http://www.mobile-industrial-robots.com)

Phone: +45 20 377 577

Email: support@mir-robots.com

CVR: 35251235

# Table of contents

---

<b>1. About this document .....</b>	<b>6</b>
1.1 Where to find more information .....	6
1.2 Version history .....	6
<b>2. MiR robot interface .....</b>	<b>10</b>
2.1 Signing in .....	10
2.2 Navigating the MiR robot interface .....	13
<b>3. Dashboards .....</b>	<b>15</b>
3.1 Dashboards .....	16
3.2 Widgets .....	19
<b>4. Setup .....</b>	<b>27</b>
4.1 Missions .....	27
4.2 Maps .....	63
4.3 Sounds .....	86
4.4 Transitions .....	88
4.5 I/O modules .....	91
4.6 Paths .....	92
4.7 Path guides .....	94
4.8 Marker types .....	97
4.9 Footprints .....	101
4.10 Users .....	105
4.11 User groups .....	108
<b>5. Monitoring .....</b>	<b>113</b>
5.1 Analytics .....	114

---

5.2 System log .....	115
5.3 Error logs .....	116
5.4 Hardware health .....	117
5.5 Safety system .....	118
5.6 Mission log .....	120
<b>6. System .....</b>	<b>123</b>
6.1 Settings .....	124
6.2 Processes .....	143
6.3 PLC registers .....	146
6.4 Software versions .....	150
6.5 Backups .....	151
6.6 Robot setup .....	152
6.7 Triggers .....	154
<b>7. Help .....</b>	<b>158</b>
7.1 Robot information .....	158
7.2 API documentation .....	159
7.3 Remote access .....	160
7.4 Service book .....	161
7.5 Manual .....	162
<b>8. Hook .....</b>	<b>163</b>
8.1 Controlling MiR Hook .....	164
8.2 Carts .....	167
8.3 Settings .....	173
8.4 Setup .....	174

---

<b>9. Modbus register reference .....</b>	<b>176</b>
9.1 Status messages .....	176
9.2 PLC triggers .....	177
9.3 Mission triggers .....	177
9.4 Action commands .....	177

# 1. About this document

This document describes the MiR robot interface and is intended for administrators of the system and users responsible for updating the system regularly.

## 1.1 Where to find more information

At [the MiR website](#), you can find the following resources under the **Manuals** tab on each product page:

- **Quick starts** describe how you start operating MiR robots quickly. This document is in print in the box with the robots. Quick starts are available in multiple languages.
- **User guides** provide all the information you need to operate and maintain MiR robots, and how to set up and use top modules and accessories, such as charging stations, hooks, shelf lifts, and pallet lifts. User guides are available in multiple languages.
- **Getting started guides** describe how to set up products that are mainly software-based, such as MiR Fleet.
- **Reference guides** contain descriptions of all the elements of the robot interface and MiR Fleet interface. Reference guides are available in multiple languages.
- **REST API references** for MiR robots, MiR hooks, and MiR Fleet.
- **MiR network and WiFi guide** specifies the performance requirements of your network and how you must configure it for MiR robots and MiR Fleet to operate successfully.

## 1.2 Version history

This table shows current and previous versions of this document and their interrelations with software releases.

Revision	Release date	Description	SW version
1.0	2017-03-02	First edition.	2.0
1.1	2017-10-30	General improvements. Continue/Pause button added to top bar. Pause action added to Missions.	2.0.2
1.2	2017-12-06	Update to SW version 2.0.14: new widgets, improved mapping editor.	2.0.14

Revision	Release date	Description	SW version
		Jumping from 2.0.2 to 2.0.14 to align with old webinterface versions 1.8.14/1.9.14.	
1.3	2018-01-26	<p>Update to SW version 2.0.15:</p> <ul style="list-style-type: none"> <li>• Redesigned Dashboard with flexible widgets and new options, for example control of Bluetooth functions.</li> <li>• New feature: Path guides for precise control of robots' paths between two positions.</li> <li>• New mapping method: Cartographer.</li> <li>• Positions &amp; Mapping section removed and fully integrated in Mapping section.</li> </ul>	2.0.15
1.4	2018-04-19	<p>Update to SW version 2.0.17.</p> <p>New features in the interface:</p> <ul style="list-style-type: none"> <li>• New Hook widget: Hook widget described in chapter 3 Dashboard.</li> <li>• Modbus: a section, Triggers, is added to chapter 6 System, and a new chapter, 11 Modbus registers, has been added.</li> </ul>	2.0.17
1.5	2018-05-24	<p>Update to SW version 2.0.18.</p> <p>New features in the robot interface:</p> <ul style="list-style-type: none"> <li>• New section, Mission log, has been added to the Monitoring menu.</li> <li>• A WiFi watchdog parameter has been added to the Advanced settings section.</li> <li>• Minor corrections and improvements throughout the document.</li> </ul>	2.0.18
1.6	2018-06-18	<p>Update to SW version 2.1.0.</p> <p>New features in the robot interface:</p>	2.1.0

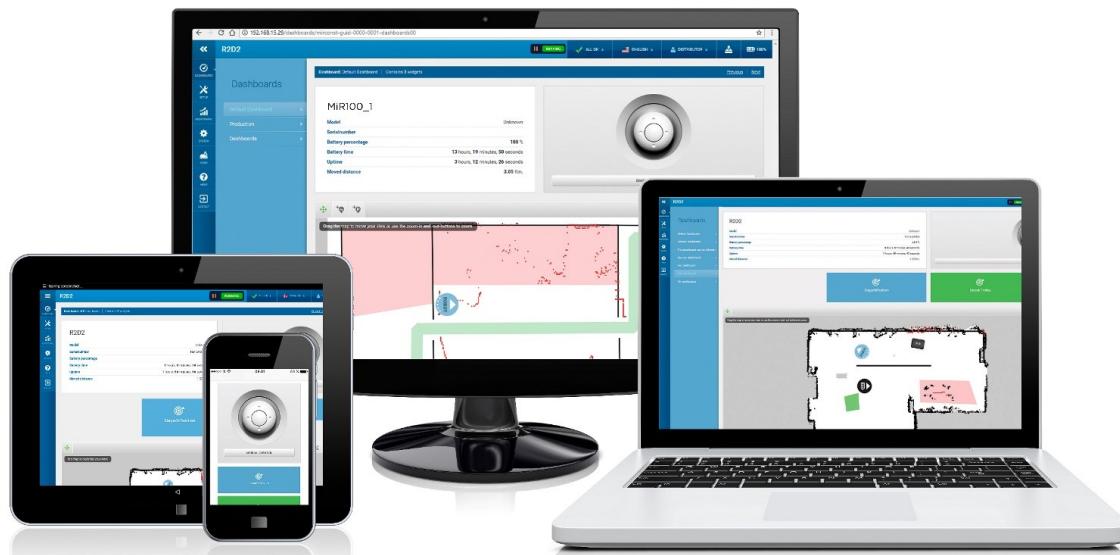
Revision	Release date	Description	SW version
		<ul style="list-style-type: none"> <li>• The Directional zones functionality is added to section 4.3 Maps.</li> <li>• Minor corrections and improvements throughout the document.</li> </ul>	
1.7	2018-07-20	<p>Update to SW version 2.2.0.</p> <p>New features in the robot interface:</p> <ul style="list-style-type: none"> <li>• I/O module feature replaces Bluetooth feature. Changes to chapters 3. Dashboards, 4.2 Missions, and 4.3 Maps. Chapter 4.9 I/O modules replaces 4.9 Bluetooth relays.</li> <li>• Sound feature has been updated. Changes to chapters 4.2 Missions, and 4.3 Maps.</li> </ul>	2.2.0
1.8	2018-10-01	<p>Update to SW version 2.3.0.</p> <p>New features in the robot interface:</p> <ul style="list-style-type: none"> <li>• A speed control functionality is added to the Mission editor.</li> <li>• Minor corrections and improvements throughout the document.</li> </ul>	2.3.0
1.9	2019-03-06	<p>Update to SW version 2.6.0.</p> <p>New features in the robot interface:</p> <ul style="list-style-type: none"> <li>• Map zones have been reconstructed and new zone settings are available.</li> <li>• Minor corrections and improvements throughout the document.</li> </ul>	2.6.0
2.0	2020-03-30	<p>Update to SW version 2.8.0</p> <p>New features in the robot interface:</p> <ul style="list-style-type: none"> <li>• Marker types, used for robots driving with shelves, has been added to the Setup section.</li> </ul>	2.8.0

Revision	Release date	Description	SW version
		<ul style="list-style-type: none"> <li>• A graphic Footprint editor has been added to the Setup section.</li> </ul>	
2.1	2020-09-11	<p>Updated to SW 2.10.0</p> <p>New features in the robot interface:</p> <ul style="list-style-type: none"> <li>• Changes in cart calibrations for MiR hooks.</li> <li>• Removed default users.</li> <li>• Minor corrections and improvements throughout the document.</li> </ul>	2.10.0

## 2. MiR robot interface

This section gives a quick overview of the MiR robot interface.

The interface is responsive and automatically adapts to your use of smartphone, tablet, or PC.



### 2.1 Signing in

The interface comes with three default access levels:

- Distributor - the MiR distributor
- Administrator - the end-customer's production engineer with technical responsibility for the robot
- User - the daily operator(s) of the robot

There are two ways in which you can sign in to the MiR robot interface:

- Username and password
- PIN code

System permissions are handled per user group whereas login credentials are handled per individual user. Read more in the sections [Users on page 105](#) and [User groups on page 108](#).

## Accessing the interface

The user interface is accessed by connecting to the robot's WiFi and opening your preferred web browser. Enter the IP address of the robot or enter **mir.com** in the browser's address bar.



The interface can be accessed via Google Chrome, Apple Safari, Mozilla Firefox, and Microsoft Edge.

## Username and password

Enter your username and password to sign in to the robot interface.



After signing in the first time, we recommend to change the passwords of the default users: Distributor, Administrator, and User. See [Users on page 105](#) to change the passwords.

The screenshot shows the MiR\_U0070 login screen. At the top, it says "Please choose a way to sign in:" with two options: "Username and password" (selected) and "PIN code". Below this, there is a section titled "Sign in by username and password". It contains instructions: "Enter your username and password to sign in to the robot.", "Your username and password should be given to you by either the robot administrator or found in the robot manual.", and "If you don't have a username and password, please contact the robot administrator.". There are input fields for "Username" (placeholder "Enter your username...") and "Password" (placeholder "Enter your password..."). A green "Sign in" button with a key icon is at the bottom.

**NOTICE**

The original username and password for the robot's web interface are in the document *Getting the robot online*.

The unique password for the WiFi access point is in the *Passwords* document.  
Both documents are in the box with the product.

## PIN code

Select the PIN code tab and enter a four-digit PIN code. This PIN code will now function as an access code to the interface. There is no preconfigured PIN code.

The screenshot shows the MiR S125 login interface. At the top, it says "S125" and "Please choose a way to log in:". Below that are two tabs: "Username and password" (which is selected) and "PIN code".  
  
The "PIN code" section has a heading "Log in by PIN code". It contains two paragraphs:

- If you are registered in the robot as a PIN code-enabled user, you can use your PIN code to log in to the robot.
- If you don't have a PIN code but would like to be able to log in with a 4-digit PIN, please contact your robot administrator.

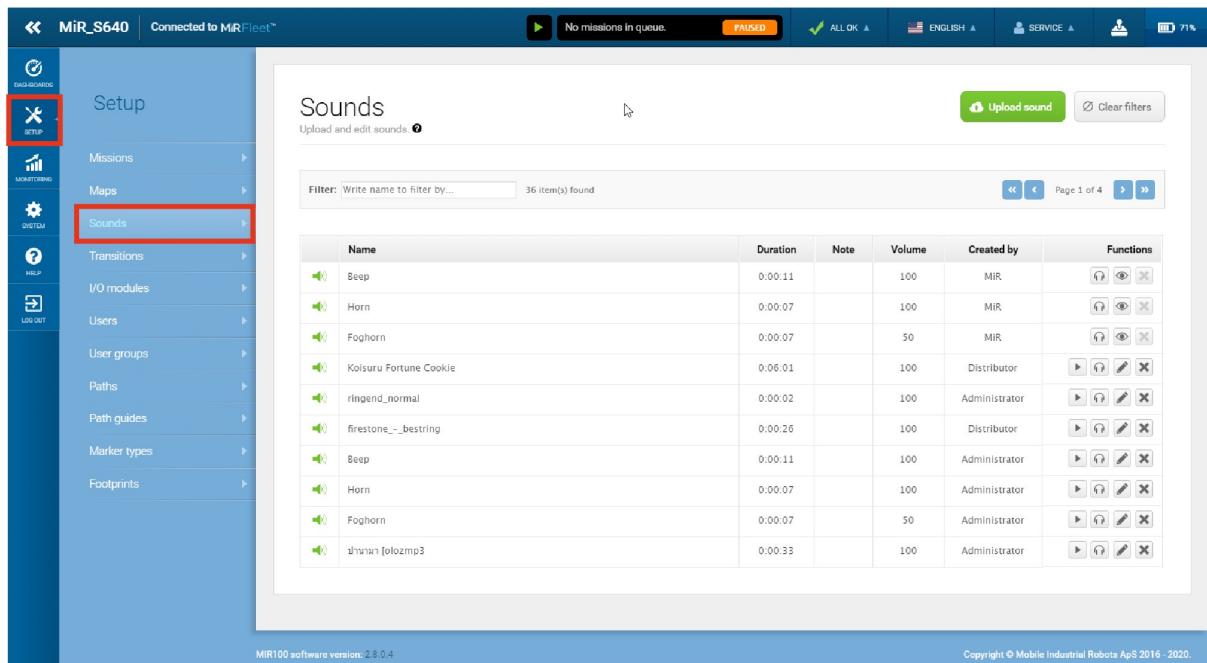
Below the text are four empty input fields for entering digits, each containing a single dot ("."). To the right is a numeric keypad grid with a 4x3 layout:

1	2	3
4	5	6
7	8	9
0	X	

## 2.2 Navigating the MiR robot interface

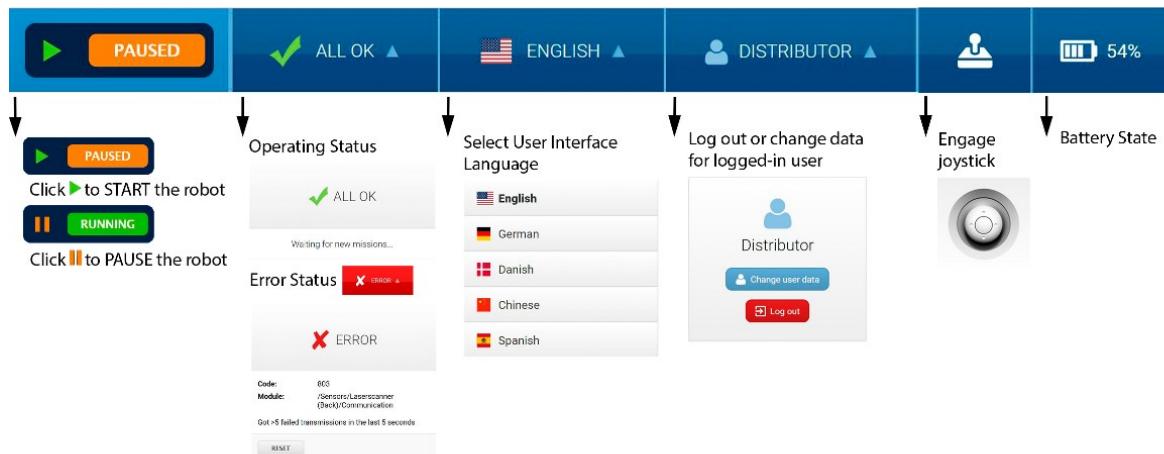
To access a section in the MiR robot interface, first select an item on the main menu, then select the relevant sub-menu. The section appears in the main window.

For example, to go to the **Sounds** section, select **Setup** on the main menu, then select **Sounds** on the submenu bar.



## Top bar

The top bar shows information on the current state of the robot, and you can start and pause the robot by selecting the button.

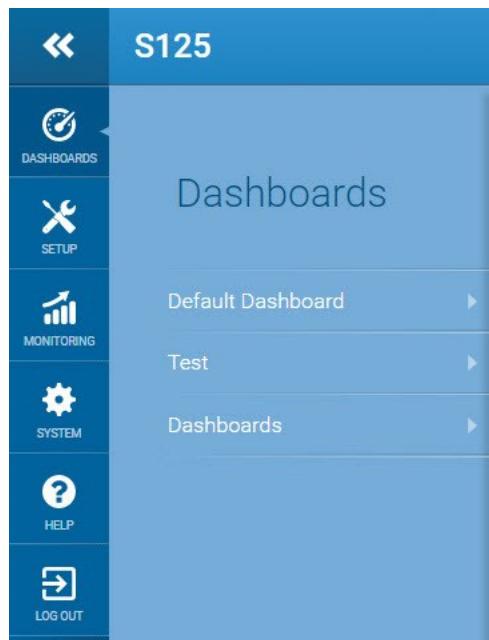


# 3. Dashboards

This section describes the items in the Dashboards menu.

The **Dashboards** menu displays all dashboards currently available on the robot.

In the subsection **Dashboards**, you can create new dashboards and edit existing ones. Select **Dashboards** to open the list of dashboards, and select the **Create dashboard** button to open the dashboard designer.



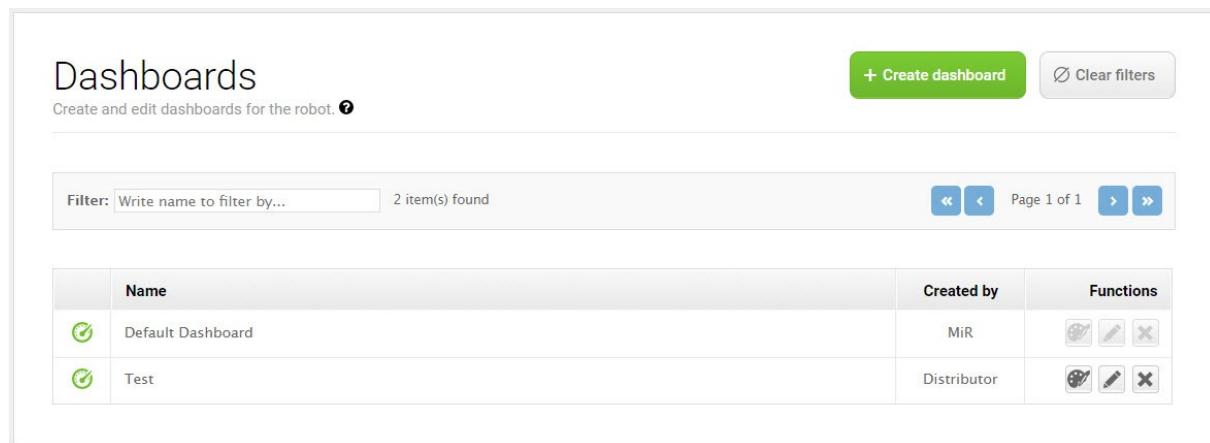
The Dashboards menu contains the following items:

**3.1 Dashboards** ..... **16**

**3.2 Widgets** ..... **19**

## 3.1 Dashboards

Dashboards are an easy way for different user groups to control the robot giving direct access to the individual key functions. A dashboard is made up of a number of widgets, each representing a feature in the system, such as a particular mission, the map the robot is running in, or the current mission queue.

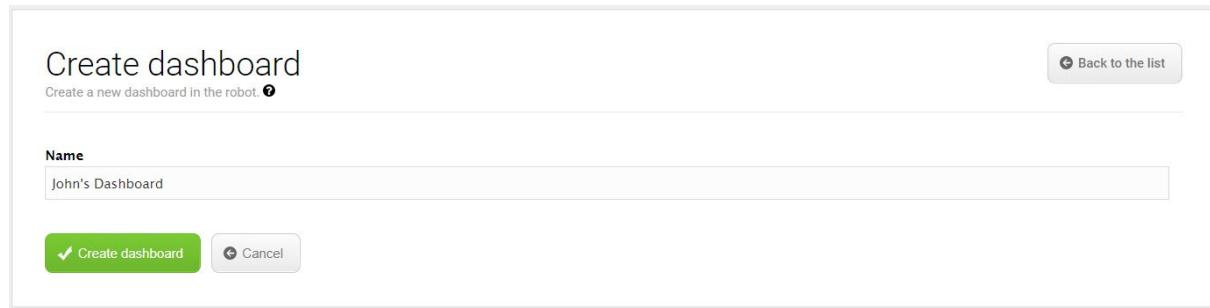


The screenshot shows a web-based application for managing dashboards. At the top, there's a header with the title "Dashboards" and a sub-instruction "Create and edit dashboards for the robot." Below the header are two buttons: "+ Create dashboard" and "Clear filters". A search bar labeled "Filter: Write name to filter by..." is followed by a message "2 item(s) found". To the right of the search bar are navigation icons for pages and rows. The main area contains a table with two rows of data:

	Name	Created by	Functions
	Default Dashboard	MiR	
	Test	Distributor	

## Create dashboard

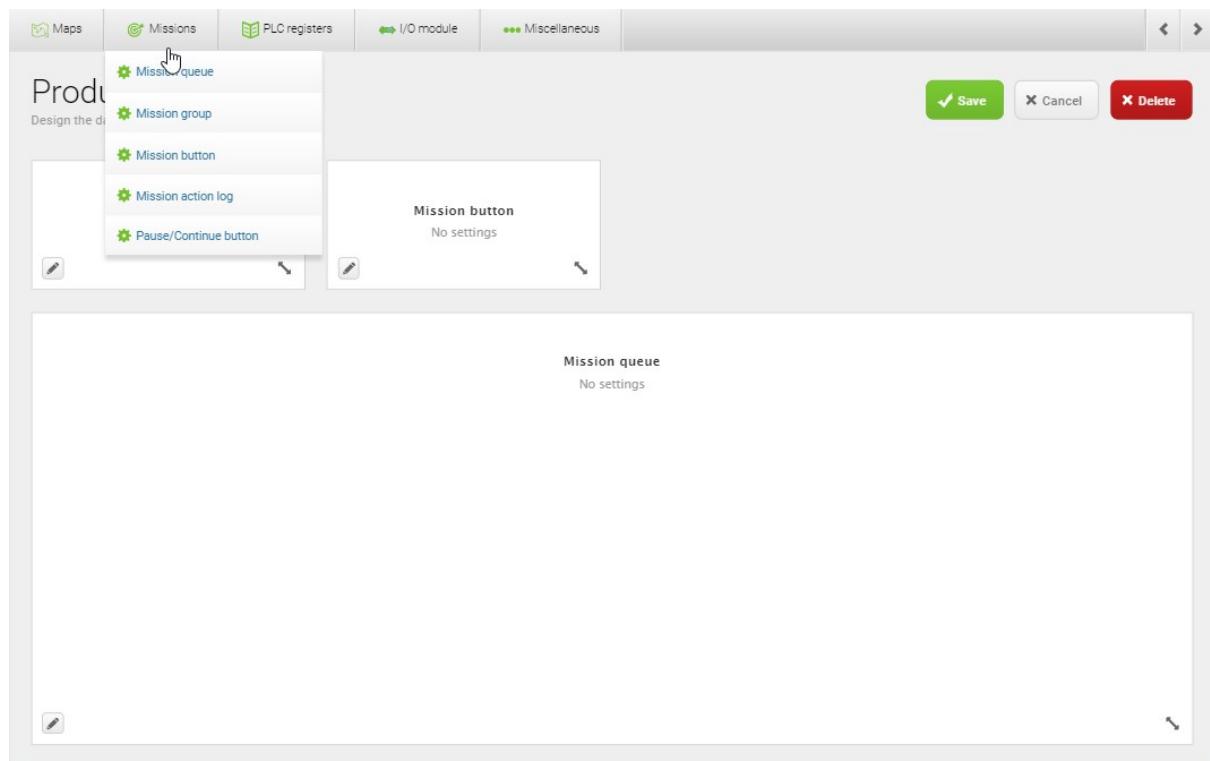
Enter a name in the **Name** field to create a new dashboard. Select **Create dashboard** to continue to the design section. Design the dashboard by adding widgets that represent the features you want to assign to the dashboard.



The screenshot shows a form titled "Create dashboard" with the sub-instruction "Create a new dashboard in the robot." There is a "Back to the list" button in the top right corner. The form has a single input field labeled "Name" containing "John's Dashboard". At the bottom are two buttons: a green "Create dashboard" button with a checkmark icon and a grey "Cancel" button.

## Dashboard designer

Design the dashboard by selecting widgets from the menus in the top bar. Resize the widgets by pulling the arrow in the lower right-hand corner and rearrange their order by selecting and dragging them. Some widgets require further settings. For example, you must select a particular mission for mission buttons. To do this, select the pen icon in the lower left corner and select the wanted action.

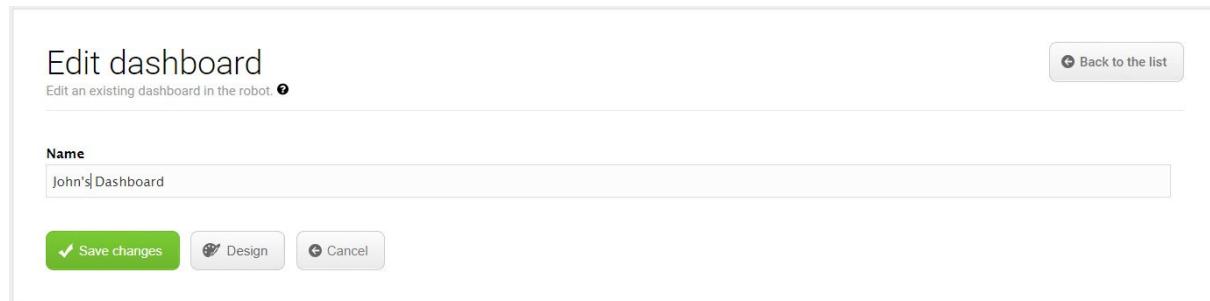


## Edit dashboard

The dashboard design can be edited and widgets added or removed.

Select **Edit**  next to the dashboard you want to edit.

You can change the name of the dashboard and select  **Design** to make changes in the dashboard widgets.



Edit dashboard

Edit an existing dashboard in the robot 

Name

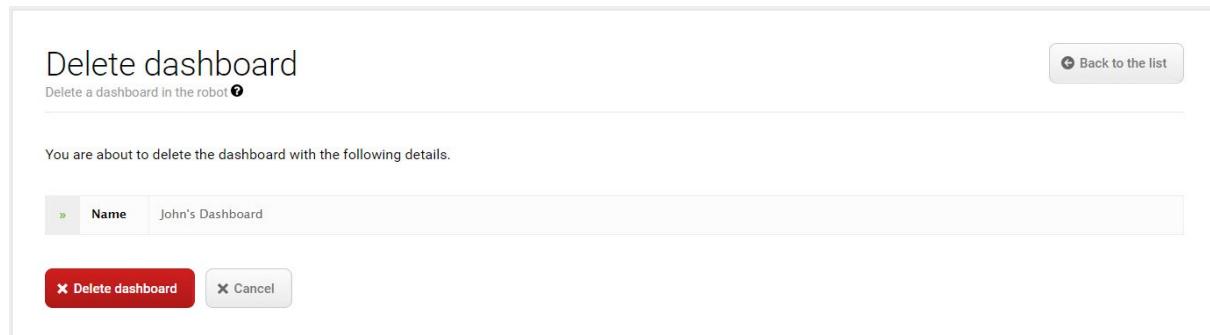
John's Dashboard

✓ Save changes     Design     Cancel

## Delete dashboard

You can delete all dashboards that are created by you or another member of the user group you belong to.

Select **Delete**  next to the dashboard you want to delete. In the next dialog, select  **Delete dashboard** to delete the dashboard.



Delete dashboard

Delete a dashboard in the robot 

You are about to delete the dashboard with the following details.

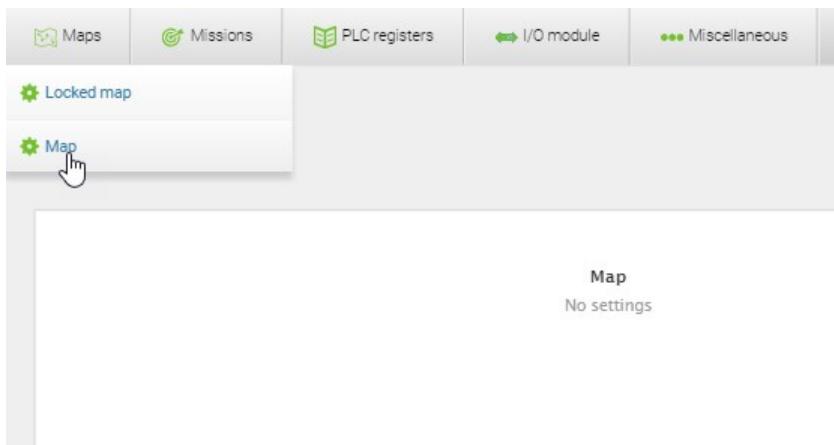
» Name John's Dashboard

 Delete dashboard     Cancel

## 3.2 Widgets

This section describes the dashboard widgets.

### Maps



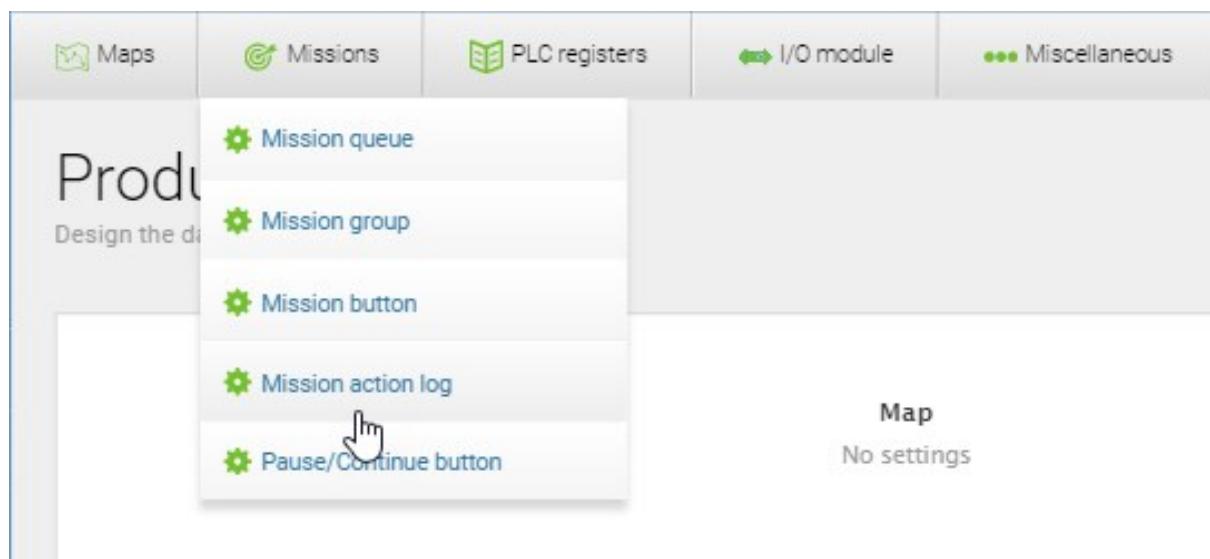
#### Locked map

A **Locked robot map** widget makes the active map of the selected robot visible on the dashboard. The robot is always shown in the middle of a locked map. Select the robot you want shown on the dashboard.

#### Map

A **Map** widget makes the active map visible on the dashboard. You can add and edit positions and markers in the widget and adjust the robot's position.

## Missions



### Mission button

You can start a mission from the dashboard by adding a **Mission button** widget and selecting a predefined mission.

### Pause/Continue button

The **Pause/Continue button** functions the same way as the pause/continue icon on the top bar of the robot interface, but may be inserted as wizard if you want it in a larger size.

### Mission queue

You can have the mission queue displayed on the dashboard by selecting a **Mission queue** widget.

### Mission action log

The **Mission action log** widget displays the individual actions being performed during the execution of a mission.

## Mission group

You can select a mission group and have all missions from that group displayed on the dashboard by adding a **Mission group** widget.

## PLC registers

### PLC button/display

Get easy access to PLC functions from the dashboard. A **PLC widget** can be designed as a button, for example, to shift between two stages, or a display button, for example, for monitoring read-out values.

## I/O module

### I/O configuration

The **I/O configuration** widget lets you program one or more actions that you want the I/O module to perform when the outputs are in a certain state and when you select the button. Add states to the widget and configure the conditions that trigger the state and the outputs that the robot sets on the I/O module when you select the widget. Use the **Reset** section to configure a default output configuration.

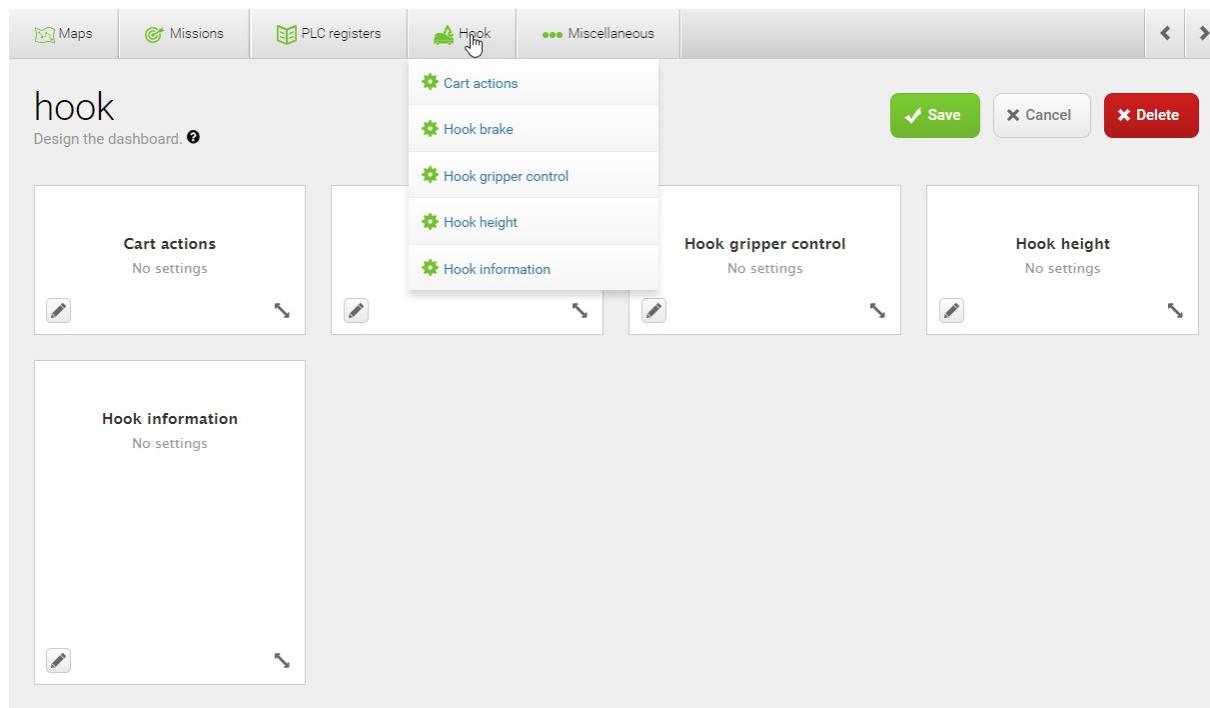
### I/O module

The **I/O module** widget lets you connect and disconnect I/O modules from the dashboard.

### I/O status

The **I/O status** widget shows the current status of the selected I/O module.

## Hook

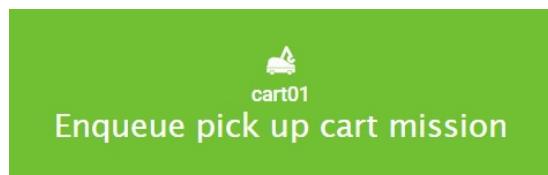


### Cart actions

This widget lets you queue the following missions: **Pick up cart**, **Place cart**.

Use the check boxes to define which missions are available in the widget. You must select at least one option (either **Pick up cart** or **Place cart**).

When you select the widget, the robot adds the mission shown in the widget to the mission queue. In the **Place cart** mission, the robot releases the gripper, lowers the hook, and leaves the cart in the current position. In the **Pick up cart** mission, the robot tries to find a cart within the hook camera's sight and pick it up. For the **Pick up cart** mission to work, it is necessary that the hook camera detects the QR code or AprilTag at the robot's current position.



## Hook brake

This widget lets you activate and deactivate the hook brake manually. The text in the widget shows the action that it executes when you select it and changes depending on the state of the hook brake. For example, if the brake is active (the arm is locked), the widget reads **Deactivate hook brake**, and selecting the widget deactivates the brake.

After you select the widget, it shows the current action and you have an option to undo the action until it is over. For example, if the brake is active and you select the widget, it shows **Deactivating... Click to undo**.



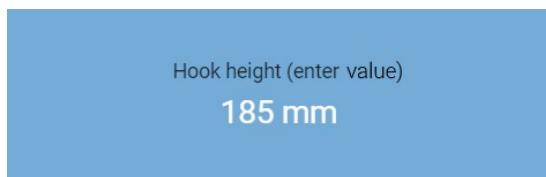
## Hook gripper control

This widget lets you open and close the hook gripper. This widget shows the current action (closing or opening) and lets you undo it during execution.



## Hook height

This widget lets you set the height of the hook manually. Use the arrows to change the value.



Selecting the height value opens the following dialog:

Enter target height

**Hook height**

OK Cancel

## Hook information

This widget shows the following information about the hook:

- Hook brake state
- Hook gripper state
- Hook height
- Hook angle

## HOOK INFORMATION

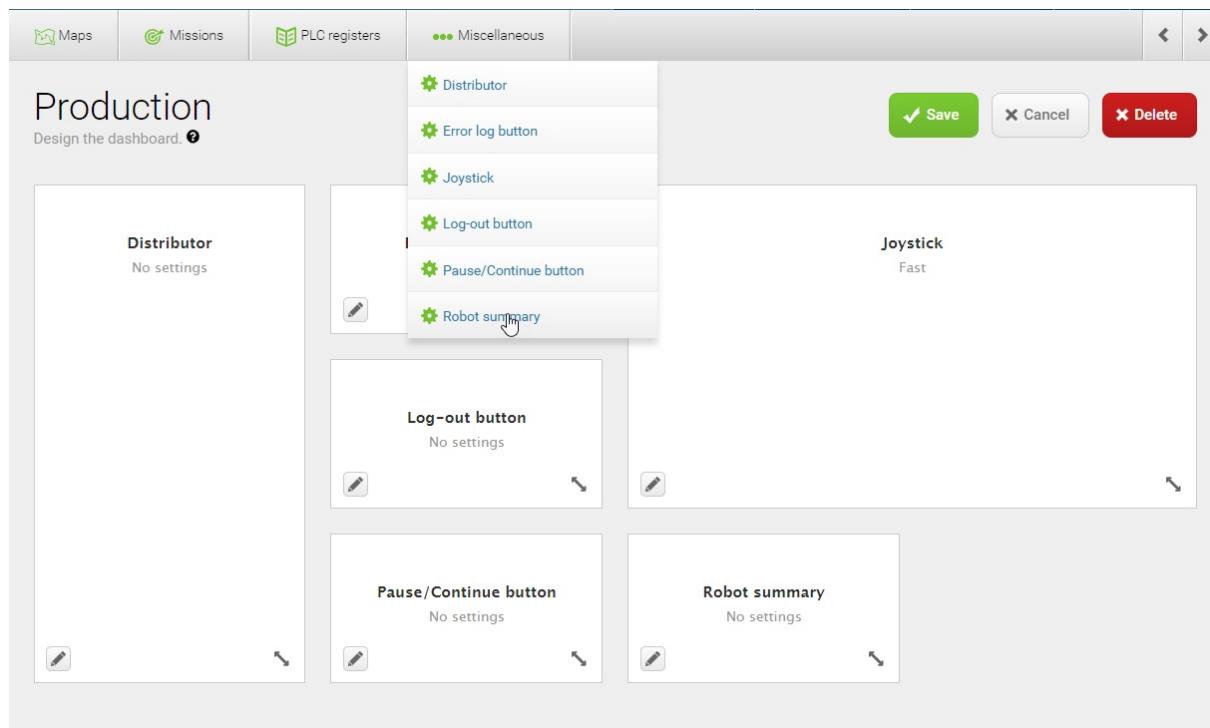
**Hook brake state**  
Active

**Hook gripper state**  
Open

**Hook height**  
185 mm

**Hook angle (horizontal)**  
0.0 degrees

## Miscellaneous



The screenshot shows the MiR Dashboard configuration interface. The top navigation bar includes tabs for Maps, Missions, PLC registers, and Miscellaneous (which is currently selected). Below the tabs, a dashboard titled "Production" is displayed with the sub-instruction "Design the dashboard." A context menu is open over a "Robot summary" button, listing several options: Distributor, Error log button, Joystick, Log-out button, Pause/Continue button, and Robot summary. At the top right of the interface are three buttons: "Save" (green), "Cancel" (grey), and "Delete" (red).

### Distributor

This widget shows information about the distributor if any distributor data has been entered in the Distributor data section under **System > Settings**.

### Error log button

The **Error log button** allows you to get an error log via the dashboard. The error log is a list of all detected system errors. Each entry is shown with a description, an indication of which module is affected, and the time when the error occurred.

### Joystick

Make one or more joysticks available directly on the dashboard. Different speeds can be selected for the joysticks; slow, medium or fast. The standard joystick in the top bar is fast, except when mapping where it runs medium speed.

## Log-out button

The **Log-out button** allows you to log off via the dashboard.

## Pause/Continue button

The **Pause/Continue button** functions the same way as the pause/continue icon on the top bar of the robot interface, but may be inserted as wizard if you want it in a larger size.

## Robot summary

The **Robot summary** widget makes it possible to have information about the robot on the dashboard: name, serial no., battery percentage, remaining battery time, uptime, and moved distance.

# 4. Setup

This section describes the items in the Setup menu.



The Setup menu contains the following items:

<b>4.1 Missions</b>	<b>27</b>
<b>4.2 Maps</b>	<b>63</b>
<b>4.3 Sounds</b>	<b>86</b>
<b>4.4 Transitions</b>	<b>88</b>
<b>4.5 I/O modules</b>	<b>91</b>
<b>4.6 Paths</b>	<b>92</b>
<b>4.7 Path guides</b>	<b>94</b>
<b>4.8 Marker types</b>	<b>97</b>
<b>4.9 Footprints</b>	<b>101</b>
<b>4.10 Users</b>	<b>105</b>
<b>4.11 User groups</b>	<b>108</b>

## 4.1 Missions

A mission is a predefined series of actions that the robot can be set to perform. A mission can be a simple transportation task between defined positions or a more complex job that includes both moving between positions and performing actions, such as unloading a pallet, moving to a charging station when the battery is low, or sending an email on arrival at a position.

Missions are started by adding a given mission to the mission queue. The robot will perform the missions in the order they are added, and an operator may rearrange the queued missions if needed.

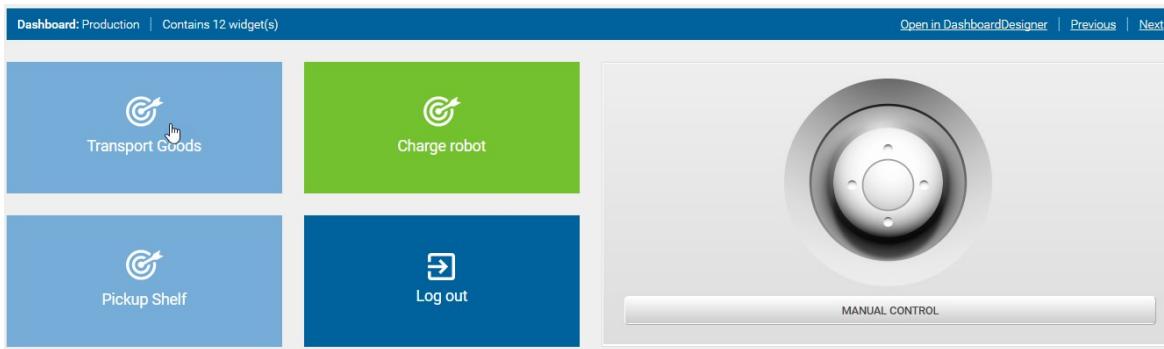
In MiR Fleet, missions are controlled in the **Scheduler**. The **Scheduler** makes it possible to prioritize missions in queue and to set a start time.

## Start mission

You can enqueue a mission in one of the following ways:

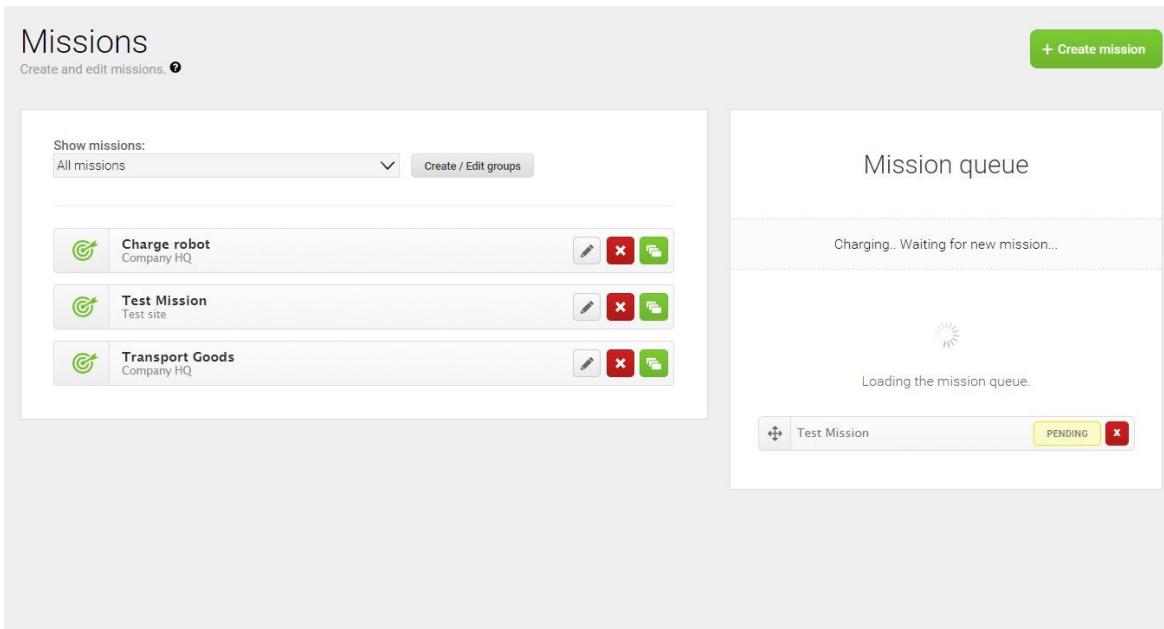
- **From a dashboard**

You can configure a Mission button widget on a dashboard.



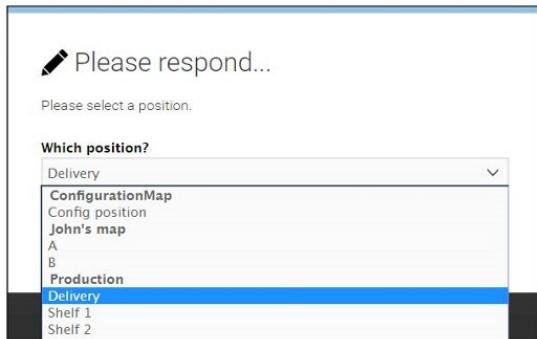
- **From the Missions menu**

To enqueue a mission in the Missions menu, select **Queue mission**  for the mission you want to add to the robot's mission queue.

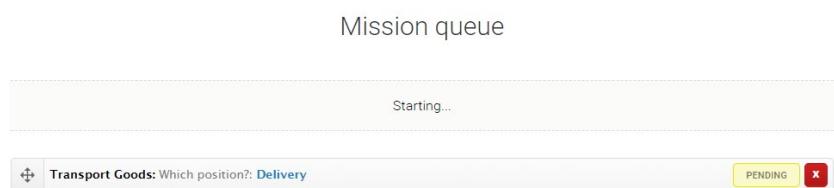


The screenshot shows the "Missions" interface. On the left, there is a list of missions under "Mission queue": "Charge robot" (Company HQ), "Test Mission" (Test site), and "Transport Goods" (Company HQ). Each mission has edit, delete, and copy icons. On the right, a "Mission queue" panel shows a message "Charging.. Waiting for new mission..." and a progress bar. Below it, a message "Loading the mission queue." is displayed. A single mission "Test Mission" is shown in the queue with a status bar indicating "PENDING".

If there are variable parameters in a mission, for example a variable position, you will be asked to select the position when adding the mission to the queue.



The selected parameters are shown in blue text.



## Creating a mission

MiR robots function through missions that you create. A mission is made up of actions, such as: move actions, logic actions, docking actions, and sounds, which can be put together to form a mission with as many actions as needed. Missions themselves can also be embedded into other missions.

Most actions have adjustable parameters, for example which position to go to. Most actions can also use variables, enabling the user to choose the value of a parameter each time the mission is used. This can be practical in cases where the robot performs the same series of actions in different areas of the site that require different parameter settings in the mission actions.

When you create a mission, you can save it in the default **Missions group**, or you can choose to save it in any of the available actions groups. The actions groups are found in the top bar of the mission editor window, and you can distinguish missions from actions by the small icons shown next to their names: missions have a target icon , and actions have a running-man icon .

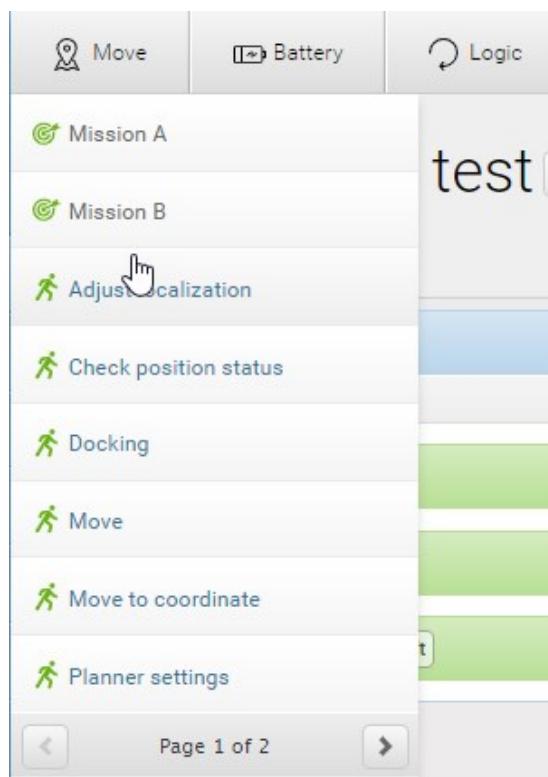
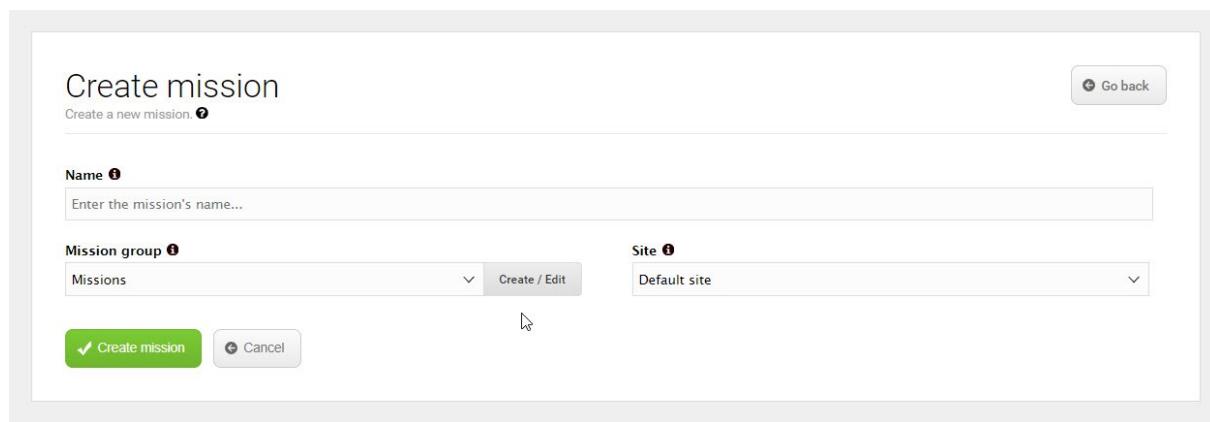


Figure 4.1. Different actions can be created and put together to make up a mission.

You can find more information about mission groups in [Mission groups](#) on the next page.

The **Mission** section also comes with a set of template missions that you can use and/or modify.



The screenshot shows a 'Create mission' dialog box. At the top, it says 'Create mission' and 'Create a new mission.' Below that is a 'Name' field with a placeholder 'Enter the mission's name...'. To the right is a 'Go back' button. Further down are 'Mission group' and 'Site' dropdown menus, both currently set to 'Missions' and 'Default site' respectively. At the bottom are two buttons: a green 'Create mission' button with a checkmark icon and a grey 'Cancel' button.

When you create a mission, you must fill in the following information:

- **Name**

The name must be unique and is used to identify the mission. For example, *Go to charging station*, *Deliver spare parts* or *Warehouse to production line 1*.

- **Mission group**

Select which group you want the mission to be part of.

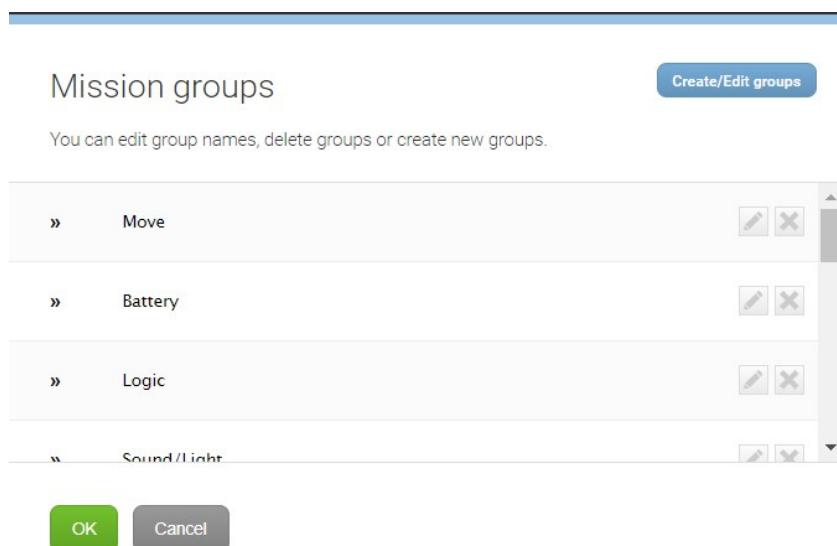
- **Site**

If you are using more than one site, select which site you want the mission to belong to.

Select **+ Create mission** to create a new mission that you can add actions to.

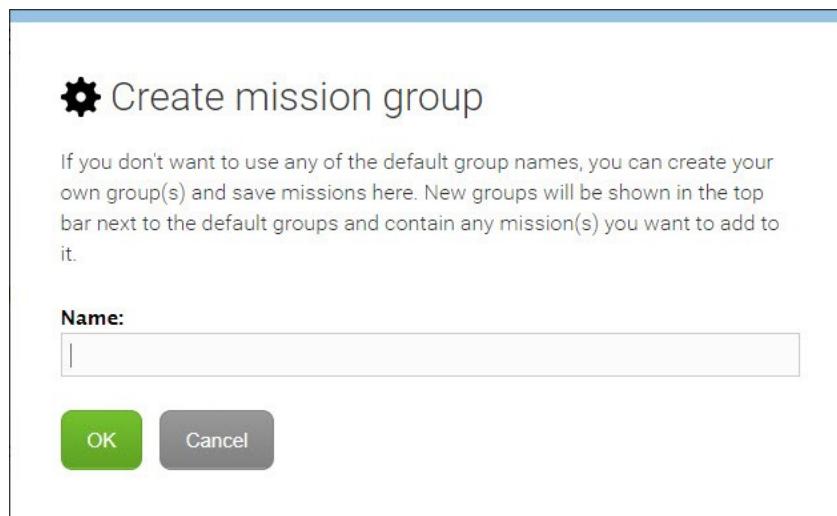
## Mission groups

Each mission group has a number of predefined actions that can be selected when you build the mission. One mission can contain actions from several groups. When you save the new mission, it will be placed in the selected group and can be used as a separate mission or as an embedded mission in other missions.



## Create mission group

If you don't want to use any of the default group names, you can create your own group(s) and save missions here. New groups will be shown in the top bar next to the default groups and contain any mission(s) you want to add to it.



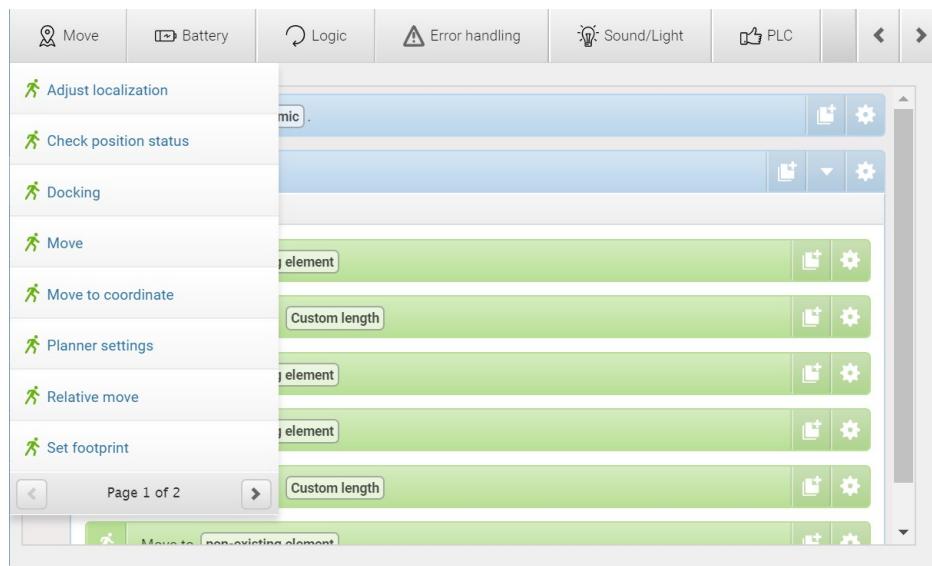
## Mission editor

A mission is built from actions that you pick from the menus in the top bar. You can also pick already created missions and embed them in new missions.

Actions and missions are grouped together in the top bar menus. All predefined actions are identified by a running-man icon. User created missions are placed together with actions in the group to which you append them and can be distinguished from actions by a target icon next to their names.

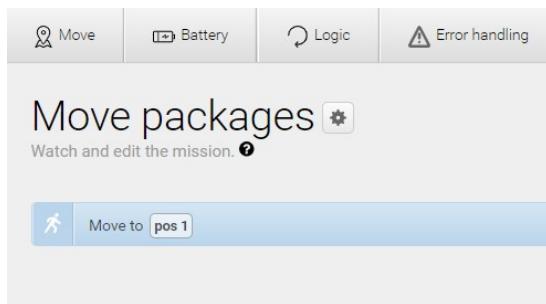
When you have picked the actions you want in your mission, do the following:

1. Drag the actions up or down with the four-headed arrow at the far left of the action line to sort them in the desired order. The actions are executed in a top-to-bottom order.
2. Set the parameters for the selected action by selecting the gear icon at the far right of the action line.



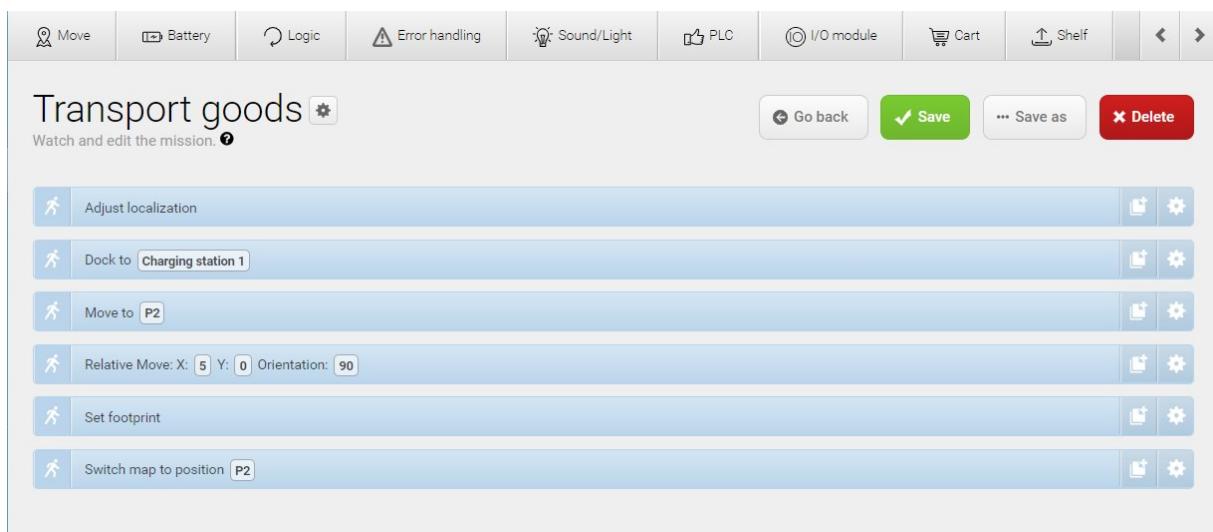
## Change mission settings

To change the name and mission group of a mission, select the gearwheel next to the name of the mission in the mission editor.



## Save mission

When you have completed the mission by adding all actions and sorted them in the desired order, select **Save** to save the mission.

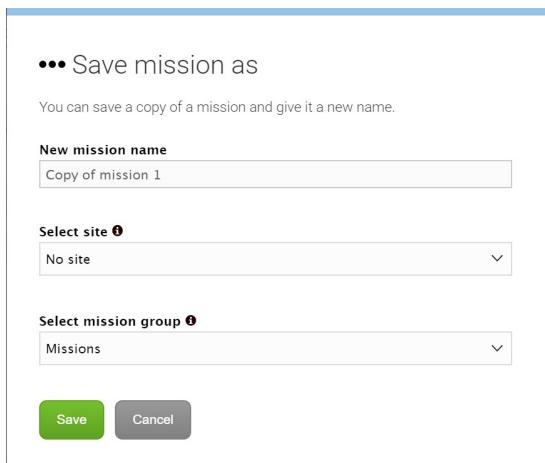


## Save mission as

You can duplicate a mission by saving it with a new name. That way it is easy to create a new mission based on an existing one.

In the Mission editor window, select **Save as**.

Name the copy of the mission, and select which site and mission group it should belong to.



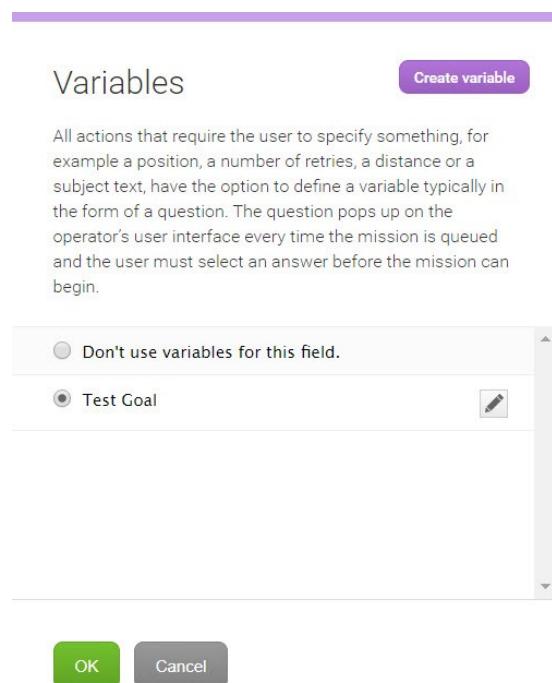
## Mission actions

Actions used in missions are in the Groups tool bar at the top of the window.



## Variables

All actions that require the user to specify something, for example, a position, a number of retries, a distance, or a subject text, have the option to define a variable. We recommend naming variables in the form of a question that describes what the value you are inserting should be used for. The question pops up on the operator's user interface every time the mission is queued or embedded in another mission, and the user must select an answer before the mission can begin.



## Create variables

In the **Name** field, enter a question that describes what the variable is used for, for example, "Which position should the robot move to?". In the **Default value** field, enter a default position.

### Create variable

In the Name field, enter a question that the operator must answer before the mission can begin, e.g. "Which battery level?"

In the Default value field, set a default percentage.

Variable name	Test Goal
Default value	Delivery
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

## Move

This action group contains the actions in the following table.

Action description	Parameter descriptions
<b>Adjust localization</b>	No adjustable parameters.
<b>Check position status</b>	<p>Position</p> <p>Select a position from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Option</p> <p>Select if the robot should check if a</p>

Action description	Parameter descriptions
<p>If the condition in the action is satisfied, the robot continues executing the mission. Otherwise, the robot reports an error.</p> <p>Example: Use this action for the following purposes:</p> <ul style="list-style-type: none"> <li>• Check whether the load is on the pallet rack before docking to the rack.</li> <li>• Check whether the cart is in position before picking it up with the hook.</li> <li>• Check whether the target position is free.</li> </ul>	<p>position is empty or occupied, or select the <b>XYZ</b> icon to define a variable</p> <p>Timeout (seconds)</p> <p>Enter the maximum time during which the robot checks the position status. If the position status does not match the option selected for this position and the time expires, the robot shows an error.</p>
<h3>Docking</h3> <p>A Docking action sends the robot to dock to a marker.</p>	<p>Marker</p> <p>Select a marker from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p>
<h3>Move</h3> <p>A Move action defines a map position the robot should move to.</p>	<p>Position</p> <p>Select a position from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Retries (Blocked Path)</p> <p>Set the number of times the robot should try to reach the position if the path is blocked, or select the <b>XYZ</b> icon to define a variable. If, after the set number of retries, the path is still blocked, the robot stops and produces an error message.</p> <p>Distance threshold</p> <p>Depending on how accurately the robot is required to position itself on the goal position, the threshold can be increased</p>

Action description	Parameter descriptions
	or decreased. The default is 0.1 m.
<b>Move to coordinate</b>	
A Move to coordinate action defines an X, Y position on the map the robot should move to. The map's origin (the 0,0 position with 0 orientation) is located at the point where the robot began mapping.  If you don't know the map's origin, you can create a fixed position with those values as a reference point.	<p>X</p> <p>Enter the X (horizontal) map position the robot should move to, or select the <b>XYZ</b> icon to define a variable.</p> <p>Y</p> <p>Enter the Y (vertical) map position the robot should move to, or select the <b>XYZ</b> icon to define a variable.</p> <p>Orientation</p> <p>Enter the orientation in degrees, that is the way the robot should turn relatively to the 0-orientation when arriving on the position, or select the <b>XYZ</b> icon to define a variable. A positive value rotates the robot counterclockwise, and a negative value rotates it clockwise.</p> <p>Retries (Blocked Path)</p> <p>Set the number of times the robot should try to reach the position if the path is blocked, or select the <b>XYZ</b> icon to define a variable. If, after the set number of retries, the path is still blocked, the robot stops and produces an error message.</p> <p>Distance threshold</p> <p>Depending on how accurately the robot is required to position itself on the goal position, the threshold can be increased or decreased. The default is 0.1 m.</p>

Action description	Parameter descriptions
<h2>Planner settings</h2> <p>A Planner settings action allows you to set the desired speed of the robot, to change the settings for how much the robot is allowed to deviate from its planned path, and to set how it should filter out obstacles when driving.</p> <p>Path deviation and obstacle clearing can be used, for example, if you want your robot to follow its path without it attempting to maneuver around any dynamic obstacles, the so-called Line-following mode.</p>	<p>Planner settings</p> <p><b>Desired speed:</b> sets the desired speed of the robot while it runs this mission.</p> <p><b>Path deviation:</b> sets the maximum distance the robot is allowed to deviate from its path before it generates a new path. Setting the value to 0 means no deviation is allowed.</p> <p><b>Path timeout</b> sets the amount of time the robot will wait for the path to clear before generating a new one. If you set the value to -1 the robot will wait indefinitely for obstacles to move out of its way instead of generating a new path.</p> <p><b>Obstacle history clearing</b> sets how the robot will clear its obstacle history during driving. The available options are, <b>No clearing</b>, <b>Clear all</b>, <b>Clear in front of robot</b>.</p>
<h2>Relative move</h2> <p>A Relative move action defines an X and a Y distance you want the robot to move and an orientation you want it to turn relative to its current position. A Relative move can be used, for example, to undock a robot from a marker.</p>	<p>X</p> <p>Enter a value in meters for how much the robot should move forward or backward from its current position. A positive value moves the robot forward, and a negative value moves it backward. Select the <b>XYZ</b> icon if you want to define a variable.</p> <p>Y</p> <p>Enter a value in meters for how much the robot should move left or right from its current position. A positive value moves the robot to the right and a negative</p>

Action description	Parameter descriptions
<p><b>⚠ CAUTION</b></p> <p>When using a Relative move action, be aware that the robot can move into Forbidden zones and through walls on the map. The robot will still drive with Collision avoidance and will not hit any obstacles it can detect.</p>	<p>value moves it to the left. Select the <b>XYZ</b> icon if you want to define a variable.</p>
	<p>Orientation</p> <p>Enter a value in degrees for how much the robot should turn (yaw) when finalizing the <b>Relative move</b>. A positive value moves it counterclockwise, and a negative value moves it clockwise. Select the <b>XYZ</b> icon if you want to define a variable.</p>
	<p>Maximum linear speed</p> <p>Enter a value in meters per second for the maximum forward or backward speed during the <b>Relative move</b>, or select the <b>XYZ</b> icon to define a variable.</p>
	<p>Maximum angular speed</p> <p>Enter a value in meters per second for the maximum turning speed during the <b>Relative move</b>, or select the <b>XYZ</b> icon to define a variable.</p>
	<p>Collision detection</p> <p>Select the check box to turn on automatic Collision detection.</p> <p>Collision detection makes the robot look for obstacles while it executes the action, and the robot will either stop or slow down to avoid colliding with the obstacle. Once the obstacle is no longer in the way, the robot will finish the action.</p>
	<p>In most situations we recommend</p>

Action description	Parameter descriptions
	<p>enabling Collision detection, but in cases where the robot needs to turn around its own center in tight spaces, it can be a good idea to disable it to prevent the robot from stopping when it gets too close to surrounding obstacles or walls.</p> <p><b>i</b> Collision detection is not a safety function. If a person enters the Protective field of the robot while Collision detection is disabled, the robot will still enter Protective stop.</p>
<h3>Set footprint</h3> <p>A Set footprint action makes it possible to change the robot's default footprint. This can be necessary, for example, if the robot carries a top module with larger proportions than the robot's own or you want to extend the footprint when the robot tows a cart. The footprint is shown as a shadow around the robot on the map.</p>	<p>Footprint</p> <p>Select a footprint, or select the <b>XYZ</b> icon to define a variable.</p>
<h3>Switch map</h3> <p>A Switch map action is required if the robot needs to switch automatically from one map to another within a mission. The maps must have overlapping areas where the robot can locate itself in the physical environment. Switch map actions are the basis for Transitions (<b>Setup &gt; Transitions</b>), which handle map switches automatically once they are set up.</p>	<p>Entry Position</p> <p>In the map you are switching to, select the position the robot should start from after the map transition, or select the <b>XYZ</b> icon to define a variable.</p> <p>The <b>Switch map</b> action must be preceded by a <b>Move</b> action to the position in the current map that physically overlaps the Entry position in the other map that the robot is switching to. The overlap of the these two positions in the physical area is</p>

Action description	Parameter descriptions
<p><b>i</b> Contact your distributor for the guide <i>How to set up transitions between maps.</i></p>	important for the robot to localize itself in the new map.

## Battery

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<h3>Charging</h3> <p>A Charging action is used to make the robot go to a charging station for automatic battery recharge. The action is defined by setting a minimum charging time and a minimum charging percentage. When the first of those are reached, the action is completed. For example, if you set the minimum time to 30 minutes and the minimum percentage to 80%, the robot will charge for minimum 30 minutes or until it reaches a battery level of 80%. You may also choose to ignore either time or percentage.</p> <p>A Charging action must be preceded by a Docking action where the robot moves to a previously defined charging marker in front of the charging station.</p>	<p>Minimum Time</p> <p>Set a minimum amount of time the robot should charge before it moves on, or select the <b>XYZ</b> icon to define a variable. The system will compare the set minimum time with the minimum percentage, and when the first of those two requirements is fulfilled, the mission continues.</p> <p>You may skip defining a minimum time by selecting the <b>Ignore value</b> check box. The robot will then charge until the minimum battery percentage level is reached.</p> <p>Minimum Percentage</p> <p>Enter the minimum battery percentage the robot should charge to before it moves on, or select the <b>XYZ</b> icon to define a variable. The system will compare the set minimum percentage with the minimum time, and when the first of those two requirements is fulfilled, the mission continues. You may skip defining a minimum percentage by selecting the</p>

Action descriptions	Parameter descriptions
	<b>Ignore value</b> check box. The robot will then charge until the minimum charge time is reached.
Charge until new mission in queue	Select this check box if you want the robot to continue charging until it receives a new mission. If selected, the robot stays in the charging station until it receives a new mission, but not until at least one of the criteria for minimum time or minimum percentage is reached.
If deselected, the robot leaves the charging station when either of the two charging criteria are reached regardless of queued missions.	

## Logic

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<b>Break</b>	A Break action is used to interrupt a Loop action.  <b>i</b> See Loop for more information.
<b>Continue</b>	No adjustable parameters.

Action descriptions	Parameter descriptions
<b>If</b>	Compare
An If action makes it possible to check battery level, number of pending missions, PLC registers, or input from I/O modules and then define which actions or missions should be performed if the conditions return either true or false. You may use one or more actions or missions to define both true and false conditions.	Select either <b>Battery Percentage, PLC Register, Pending Missions, or I/O input</b> , or select the <b>XYZ</b> icon to define a variable.
<b>Battery Percentage:</b> An If action on battery percentage checks if the battery percentage is below, above, or equal to a set limit and, depending on the result, either sends the robot to a charging station or continues the mission. The True action could be a previously defined charging mission. The False action could be any alternative actions or missions, but may also be left blank. In that case, the robot will continue to the next step in the mission.	Module
<b>Pending missions:</b> An If action on pending missions checks if the number of pending (queued) missions is below, above, or equal to a set number. You then set actions that define what the robot should do if the set condition returns true or false. An example could be to send the robot to a charging station if the number of queued missions exceeds a certain amount.	For I/O inputs, select an I/O module from the drop-down list, or select the <b>XYZ</b> icon to define a variable.
<b>PLC Register:</b> An If action on a PLC register checks if the register is set to a	Index
	For PLC registers: enter the required index number (Integer registers 1-100, Floating point registers 101-200), or select the <b>XYZ</b> icon to define a variable.
	Operator
	Select the arithmetic operator you want to use, or select the <b>XYZ</b> icon to define a variable.
	Operators are arithmetic operators used to specify the compare mission, for example, use the < operator to specify "If Battery percentage is below 50 percent".
	The available operators are:
	<ul style="list-style-type: none"> <li>• == 'equal to'</li> <li>• != 'not equal to'</li> <li>• &gt; 'greater than'</li> <li>• &gt;= 'greater than or equal to'</li> <li>• &lt; 'lesser than'</li> <li>• &lt;= 'lesser than or equal to'.</li> </ul>

Action descriptions	Parameter descriptions
<p>certain value, for example, register 6=1, indicating that a lift is lowered when the robot arrives at a shelf. The True action (the lift is lowered) could then be a Wait for PLC register action, for example, wait for register 6 to reset to 0.</p> <p><b>I/O input:</b> An If action on an I/O input checks if the register is set to a certain value, for example, register 6=1, indicating that a lift is lowered when the robot arrives at a shelf. The True action (the lift is lowered) could then be a Wait for PLC register action, for example, wait for register 6 to reset to 0.</p>	<p>Value</p> <p>Enter the value for the selected register, or select the <b>XYZ</b> icon to define a variable.</p>
<p><b>Loop</b></p> <p>A Loop action makes it possible to have the robot repeat a mission either a specified number of times or indefinitely (until stopped by an operator). Drag actions or predefined missions into the loop action to define the sequence of actions the robot will repeat. A loop can be interrupted with a Break action.</p>	<p>Iterations</p> <p>Set the number of times the robot should run the loop, or select the <b>XYZ</b> icon to define a variable.</p> <p>Content</p> <p>Insert the actions that should be performed in each loop iteration.</p>
<p><b>Pause</b></p> <p>A Pause action pauses the mission execution until an operator selects <b>Continue</b>.</p> <p>This can be used in missions where the robot should wait for an operator to do something, for example, placing items on the robot and manually sending the robot on to another position by selecting</p>	<p>No adjustable parameters.</p>

Action descriptions	Parameter descriptions
<b>Continue.</b>	
<b>Prompt user</b>	<p>Question</p> <p>Write a question which can be answered with a yes or a no, or select the <b>XYZ</b> icon to define a variable.</p>
A Prompt user action can be used when it is required to stop and ask the operator what the next step in the mission should be. The action consists of a Yes action, a No action, and a Time-out action. The operator will be asked, for example, "Do you want to go to position X?". If the operator answers Yes, the robot will go to position X. If the operator answers No, the robot will carry on to the defined No action, for example, move to an alternative position. If the operator does not answer yes or no within a given time, the Time-out action will be executed, for example, sending an email.	<p>User group</p> <p>Select which User group the mission is intended for, or select the <b>XYZ</b> icon to define a variable.</p> <p>Timeout (seconds)</p> <p>Set a timeout for when the robot should continue if the user does not answer the question. If the timeout is reached, the robot will execute the actions in the Timeout scope.</p>
<b>Return</b>	No adjustable parameters.
A Return action is used to abort a mission. It can be used, for example, as catch action in a <b>Try/Catch</b> action.	
<b>Wait</b>	<p>Time</p> <p>Set an amount of time the robot should wait before moving to the next action in the mission.</p>
A Wait action pauses the mission in a given period of time.	
<b>While</b>	<p>Compare</p> <p>Select either <b>Battery Percentage</b>, <b>PLC Register</b>, <b>Pending Missions</b>, or <b>I/O input</b>, or select the <b>XYZ</b> icon to define a variable.</p>
A While action makes it possible to check battery level, number of pending missions, PLC registers, or input from I/O modules and then define which actions or missions	

Action descriptions	Parameter descriptions
<p>should be performed while these conditions return true. You may use one or more actions or missions to define the while conditions.</p> <p><b>Battery Percentage:</b> A <b>While</b> action on battery percentage checks if the battery percentage is below or above a set limit and, depending on the result, either sends the robot to a charging station or continues the mission.</p> <p><b>PLC Register:</b> A <b>While</b> action on a PLC register checks if the register is set to a certain value, for example, register 6=1, indicating that a lift is lowered when the robot arrives at a shelf.</p> <p><b>Pending missions:</b> A <b>While</b> action on pending missions checks if the number of pending (queued) missions is below, above or equal to the set number. You then set an action that defines what the robot should do if the set condition returns true. An example could be to send the robot to a charging station if the number of queued missions exceeds a certain amount.</p> <p><b>I/O input:</b> A <b>While</b> action on an I/O input checks if the register is set to a certain value, for example, register 6=1, indicating that a lift is lowered when the robot arrives at a shelf. The <b>True</b> action (the lift is lowered) could then be a <b>Wait for PLC register</b> action, for example wait for register 6 to reset to 0.</p>	<p>Module</p> <p>For I/O inputs, select an I/O module from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Index</p> <p>For PLC registers: enter the required index number (Integer registers 1-100, Floating point registers 101-200), or select the <b>XYZ</b> icon to define a variable.</p> <p>Operator</p> <p>Select the arithmetic operator you want to use, or select the <b>XYZ</b> icon to define a variable.</p> <p>Operators are arithmetic operators used to specify the compare mission, for example, use the &lt; operator to specify "If Battery percentage is below 50 percent".</p> <p>The available operators are:</p> <ul style="list-style-type: none"> <li>• == 'equal to'</li> <li>• != 'not equal to'</li> <li>• &gt; 'greater than'</li> <li>• &gt;= 'greater than or equal to'</li> <li>• &lt; 'lesser than'</li> <li>• &lt;= 'lesser than or equal to'.</li> </ul> <p>Value</p> <p>Enter the value for the selected register, or select the <b>XYZ</b> icon to define a variable.</p>

## Error handling

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<b>Create log</b> <p>A Create log action is used to create user generated error logs. The error logs are stored under <b>Monitoring &gt; Error logs</b>. This is useful in, for example, a Try/Catch action where a log is created when catching an unsuccessful try.</p>	Description Enter a description for the log type you want to create, or select the <b>XYZ</b> icon to define a variable. An example of a description could be “Mission x fail log”
<b>Throw error</b> <p>A Throw error action is used to enter an error message that will be shown in the user interface when the mission is run. You can, for example, use this with a Try/Catch action, to make the robot report an error if it fails to execute its main mission.</p>	Message Enter the message you want displayed on the user interface when the mission is run, or select the <b>XYZ</b> icon to define a variable.
<b>Try/Catch</b> <p>A Try/Catch action is a way to reinforce missions by defining an alternative action if the first choice action fails. This will in many cases prevent a mission from discontinuing in case, for example, a position is blocked. A Try/Catch action consists of a Try action, which the robot should attempt to complete, and a Catch action that is used in case the Try action fails.</p>	Try Select the action(s) that should be attempted.  Catch Select the action that should be performed if the action(s) within <b>Try</b> fails.

## Sound/Light

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<p><b>Play sound</b></p> <p>A Play sound action sets a sound, for example, a beep, a horn, or a voice message that the robot will play at a given stage in the mission or for the whole duration of the mission. There is a selection of standard sound bites to choose from, or you can upload your own sounds to the robot in the section <b>Setup &gt; Sounds</b>.</p>	<p>Sound</p> <p>Select a sound from the list, or select the <b>XYZ</b> icon to define a variable.</p> <p>If you want to hear the sounds before selecting one, go to <b>Setup &gt; Sounds</b>. You can hear the sounds on your computer by selecting the headset icon.</p> <p>Volume</p> <p>Set the volume of the sound (0-100), or select the <b>XYZ</b> icon to define a variable. 100% is approximately 80 dB.</p> <p>Mode</p> <p>Select how the sound should be used in the mission:</p> <p><b>Full length</b> plays the sound from start to finish, starting at the point in the mission where it is inserted and ending when the sound file finishes.</p> <p><b>Loop</b> keeps repeating the sound file until the mission is completed.</p> <p><b>Custom length</b> plays the sound for the duration of time you set under <b>Duration</b>. If the set duration exceeds the duration of the sound file itself, the sound file will loop for the duration of the set time.</p>

Action descriptions	Parameter descriptions
	<p><b>i</b> You can insert a Stop sound action anywhere in the mission. This will stop the playing of the current sound no matter which mode you have selected.</p>
	<p>Duration</p> <p>Set an amount of time the sound should play, or select the <b>XYZ</b> icon to define a variable.</p>
Show light	<p>Effect</p> <p>Select a light effect from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Speed</p> <p>Select a fast or slow speed from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Color 1</p> <p>Select a color from the drop-down list, or select the <b>XYZ</b> icon to define a variable. If you select two different colors for Color 1 and 2, the robot will alternate between the two.</p> <p>Color 2</p> <p>Select a color from the drop-down list, or select the <b>XYZ</b> icon to define a variable. If you select two different colors for Color 1 and 2, the robot will alternate between the two.</p> <p>Intensity</p> <p>Set the intensity of the light, or select the <b>XYZ</b> icon to define a variable. The intensity is defined as a percentage where 100 is full intensity.</p>

Action descriptions	Parameter descriptions
	<p>Timeout (seconds)</p> <p>Set an amount of time the light should show, or select the <b>XYZ</b> icon to define a variable.</p>

## Stop sound

Stop playing the current sound. No adjustable parameters.

## PLC

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<p><b>Set PLC register</b></p> <p>A Set PLC register action is used to set a value in a register. The register can be set in three ways:</p> <ul style="list-style-type: none"> <li>• <b>Set</b>: sets a value every time the mission is executed.</li> <li>• <b>Add</b>: adds a value every time the mission is executed.</li> <li>• <b>Subtract</b>: subtracts a value every time the mission is executed.</li> </ul>	<p>Register</p> <p>Select a specific PLC register, or select the <b>XYZ</b> icon to define a variable. Registers 1 to 100 are reserved for integers and registers from 101-199 for floating point numbers.</p> <p>Action</p> <p>Select an action from the drop-down list, or select the <b>XYZ</b> icon to define a variable. The options are <b>Set</b>, <b>Add</b>, and <b>Subtract</b>.</p> <p>Value</p> <p>Enter a value for the selected register, or select the <b>XYZ</b> icon to define a variable. If the selected register is between 1 and 100, the value must be an integer. If the selected value is between 101 and 200, the value must be a floating point number.</p>

Action descriptions	Parameter descriptions
<h3>Set and reset PLC register</h3> <p>A Set and reset PLC register action is useful in missions where the robot is requested to set a value in a PLC register and reset the register to the original value when the action is finished.</p>	<p>Register</p> <p>Select a specific PLC register, or select the <b>XYZ</b> icon to define a variable. Registers 1 to 100 are reserved for integers and registers from 101-199 for floating point numbers.</p>
	<p>Value</p> <p>Enter a value for the selected register, or select the <b>XYZ</b> icon to define a variable. If the selected register is between 1 and 100, the value must be an integer. If the selected value is between 101 and 200, the value must be a floating point number.</p>
	<p>Reset value</p> <p>Enter a value for the selected register, or select the <b>XYZ</b> icon to define a variable. If the selected register is between 1 and 100, the value must be an integer. If the selected value is between 101 and 200, the value must be a floating point number.</p>
<h3>Wait for PLC register</h3> <p>A Wait for PLC register action is used to wait for a value and continue to the next action when the value is found in the set register.</p>	<p>Register</p> <p>Select a specific PLC register, or select the <b>XYZ</b> icon to define a variable. Registers 1 to 100 are reserved for integers and registers from 101-199 for floating point numbers.</p>
	<p>Value</p> <p>Enter a value for the selected register, or</p>

Action descriptions	Parameter descriptions
	<p>select the <b>XYZ</b> icon to define a variable. If the selected register is between 1 and 100, the value must be an integer. If the selected value is between 101 and 200, the value must be a floating point number.</p>
	<p>Timeout (seconds)</p> <p>Define how long the robot should wait for the value in the set register before giving an error.</p>

## Email address

This action group is only available after it has been enabled under **System > Settings > Features**.

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<b>Send email</b>	Recipient
A Send email action is used to send email messages to selected recipients as part of a mission, for example, to let an operator know that the robot has arrived at a specific location. Recipients must be set up in the Users section ( <b>Setup &gt; Users</b> ) with an email address. Furthermore, an email account must be set up in the robot under <b>System &gt; Settings &gt; Email configuration</b> .	Select a recipient from the drop-down list, or select the <b>XYZ</b> icon to define a variable. The recipients on the list come from the Users section.
	Subject
	Type a subject of the email, or select the <b>XYZ</b> icon to define a variable.
	Message
	Write the message that the robot should send to the selected email address when the mission is executed, or select the <b>XYZ</b> icon to define a variable.

## I/O module

This action group is only available after it has been enabled under **System > Settings > Features**.

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<b>Connect Bluetooth</b>  A Connect Bluetooth action is used when the robot must connect and stay connected to a Bluetooth module.	Module  Select a Bluetooth module from the drop-down list, or select the <b>XYZ</b> icon to define a variable. Bluetooth modules are set up in the I/O modules section ( <b>Setup &gt; I/O modules</b> ).
<b>Disconnect Bluetooth</b>  A Disconnect Bluetooth action is used when the robot must close the connection to a Bluetooth module.	No adjustable parameters.
<b>Set output</b>  A Set output action is used when the robot needs to send a command to an I/O module.	Module  Select an I/O module from the drop-down list, or select the <b>XYZ</b> icon to define a variable. I/O modules are set up in the section <b>Setup &gt; I/O modules</b> .  SMTP port  Enter which output port relay should be activated (1-4) , or select the <b>XYZ</b> icon to define a variable.  Operation  Set operation to <b>On</b> or <b>Off</b> , or select the <b>XYZ</b> icon to define a variable. For example, select <b>On</b> if the I/O module is used to open a door.

Action descriptions	Parameter descriptions
<b>Set and reset I/O</b>	<p>Timeout (seconds)</p> <p>Set an amount of time the relay should stay on, or select the <b>XYZ</b> icon to define a variable.</p>
	<p>Module</p> <p>Select an I/O module from the drop-down list, or select the <b>XYZ</b> icon to define a variable. I/O modules are set up in the section <b>Setup &gt; I/O modules</b>.</p>
	<p>Output</p> <p>Enter which output port relay should be activated (1-4) , or select the <b>XYZ</b> icon to define a variable.</p>
	<p>Operation</p> <p>Set operation to <b>On</b> or <b>Off</b>, or select the <b>XYZ</b> icon to define a variable. For example, select <b>On</b> if the I/O module is used to open a door.</p>
<b>Wait for input</b>	<p>Timeout (seconds)</p> <p>Set an amount of time the relay should stay on, or select the <b>XYZ</b> icon to define a variable.</p>
	<p>Module</p> <p>Select an I/O module from the drop-down list, or select the <b>XYZ</b> icon to define a variable. I/O modules are set up in the section <b>Setup &gt; I/O modules</b>.</p>

Action descriptions	Parameter descriptions
	Input
	Enter the input port number, or select the <b>XYZ</b> icon to define a variable.
	Value
	Set operation to <b>On</b> or <b>Off</b> , or select the <b>XYZ</b> icon to define a variable. For example, select <b>Off</b> if the <b>Wait for input</b> action is used to stop a conveyor belt.
	Timeout (seconds)
	Define how long the robot should wait for the input to match the state set in Value before giving an error.

## Safety system

MiR250, MiR500, and MiR1000 only

This action group is only available after it has been enabled under **System > Settings > Features**.

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<a href="#">Mute protective fields</a>	Sound
A Mute protective fields action makes the robot mute selected protective fields while it is executing a number of actions. Drag the actions you want the robot to perform into the Mute protective fields action.	Select a sound from the list, or select the <b>XYZ</b> icon to define a variable.  If you want to hear the sounds before selecting one, go to <b>Setup &gt; Sounds</b> . You can hear the sounds on your computer by selecting the headset icon.
	Volume

Action descriptions	Parameter descriptions
	<p>Set the volume of the sound (0-100), or select the <b>XYZ</b> icon to define a variable. 100% is approximately 80 dB.</p>
	<p>Front, Rear, and Sides</p> <p>Select whether to mute or unmute the protective fields to the front, rear, or sides of the robot.</p>

## Cart

MiR100 Hook and MiR200 Hook only

This action group is only available after it has been enabled under **System > Settings > Features**.

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<a href="#">Pick up cart</a>	<p>Position</p> <p>Select a position from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p>
<a href="#">Place cart</a>	<p>Cart</p> <p>Select either a specific cart or <b>Any valid cart</b> from the drop-down list. If a specific cart is chosen and another cart is at the position, the action will produce an error.</p>
	<p>Position</p> <p>Select a position from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p>
	<p>Release cart</p> <p>Choose whether or not to release the cart</p>

Action descriptions	Parameter descriptions
	<p>after arriving at the position.</p> <p>Reverse into place</p> <p>You can choose to allow the robot to reverse into place. <b>Yes, with collision check</b> means that the robot will scan the area and check for obstacles before moving the cart to the drop-off position. <b>Yes, without collision check</b> means that the robot will move the cart into place without scanning for obstacles. This can be necessary when the robot docks into alignment fixtures.</p>

## Shelf

This action group is only available after it has been enabled under **System > Settings > Features**.

This action group contains the following template missions that you can add to other missions and configure.

Action descriptions	Parameter descriptions
<p><a href="#">Pick up MiR500/MiR1000 shelf</a></p> <p>This template mission sends a MiR500/MiR1000 robot to a shelf position to pick up a shelf, changes the footprint of the robot, and then makes the robot move away from the shelf position.</p>	<p>Marker position</p> <p>Select a marker from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Marker type</p> <p>Select a marker type from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Shelf footprint</p> <p>Select a footprint, or select the <b>XYZ</b> icon</p>

Action descriptions	Parameter descriptions
	<p>to define a variable.</p>
Mute front	<p>Select <b>Muted</b> to mute the Protective field set in front of the robot.</p>
Mute rear	<p>Select <b>Muted</b> to mute the Protective field set behind the robot.</p>
Mute sides	<p>Select <b>Muted</b> to mute the Protective fields to the sides of the robot.</p>
Undocking distance	<p>Enter a value in meters for how much the robot should move forward or backward from its current position. A positive value moves the robot forward, and a negative value moves it backward. Select the <b>XYZ</b> icon if you want to define a variable.</p>
<hr/>	
Pick up Shelf I/O	<p>Module</p> <p>For I/O inputs, select an I/O module from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p>
This template mission sends a robot with a shelf lifting application controlled with I/O modules to a shelf position to pick up a shelf and changes the footprint of the robot.	<p>Marker position</p> <p>Select a marker from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p>
<hr/>	
	<p>Marker type</p> <p>Select a marker type from the drop-down list, or select the <b>XYZ</b> icon to define a</p>

Action descriptions	Parameter descriptions
<b>Pick up Shelf PLC</b>	<p>variable.</p> <p>Shelf footprint</p> <p>Select a footprint, or select the <b>XYZ</b> icon to define a variable.</p>
<b>Place MiR500/MiR1000 shelf</b>	<p>Marker position</p> <p>Select a marker from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Marker type</p> <p>Select a marker type from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p> <p>Shelf footprint</p> <p>Select a footprint, or select the <b>XYZ</b> icon to define a variable.</p>
	<p>Mute front</p> <p>Select <b>Muted</b> to mute the Protective field set in front of the robot.</p> <p>Mute rear</p> <p>Select <b>Muted</b> to mute the Protective field set behind of the robot.</p> <p>Undocking distance</p> <p>Enter a value in meters for how much the robot should move forward or backward from its current position. A positive value moves the robot forward, and a negative</p>

Action descriptions	Parameter descriptions
<h3>Place Shelf I/O</h3> <p>This template mission makes a robot with a shelf lifting application controlled with I/O modules place a shelf at the current position and changes the robot's footprint to the default value.</p>	<p>value moves it backward. Select the <b>XYZ</b> icon if you want to define a variable.</p>
<h3>Place Shelf PLC</h3> <p>This template mission makes a robot with a shelf lifting application controlled with PLC registers place a shelf at the current position and changes the robot's footprint to the default value.</p>	<p>Module</p> <p>For I/O inputs, select an I/O module from the drop-down list, or select the <b>XYZ</b> icon to define a variable.</p>

## UR

This action group is only available after it has been enabled under **System > Settings > Features**.

This action group contains the actions in the following table.

Action descriptions	Parameter descriptions
<h3>Run UR program</h3> <p>A Run UR action is used to communicate with a Universal Robots application. The action starts a .urp file saved on the Universal robot. The program name is [program name].urp. Leave out .urp when you type the name. The MiR robot will continue until the given UR program has</p>	<p>Program name</p> <p>Enter the name of the UR program (without the urp extension), or select the <b>XYZ</b> icon to define a variable.</p>

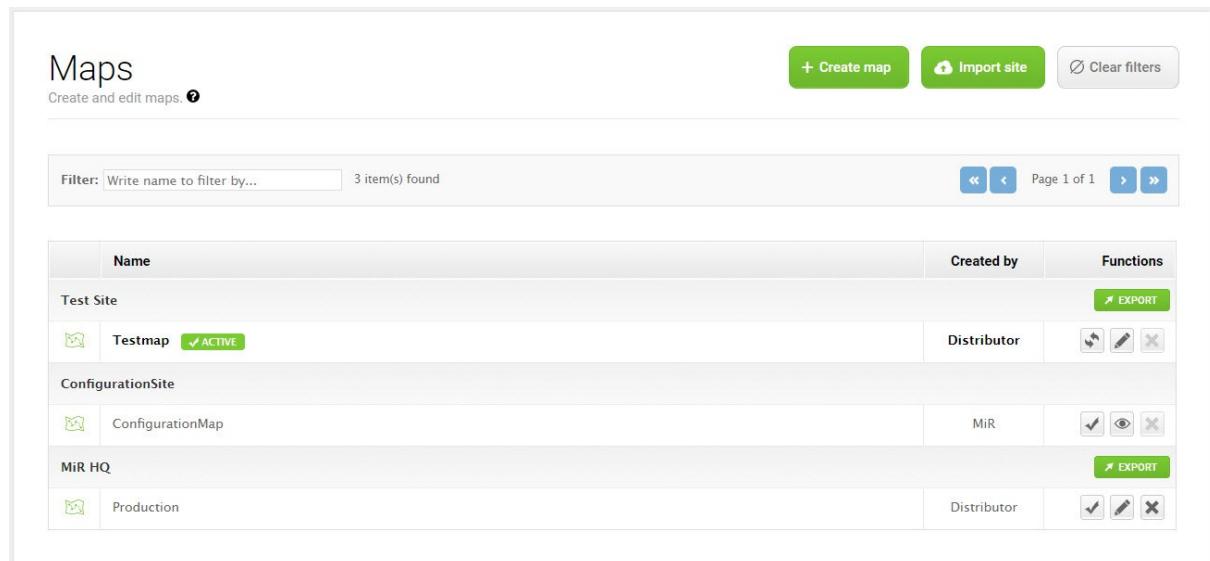
## Action descriptions

## Parameter descriptions

been executed.

## 4.2 Maps

In the **Maps** section, you create or edit the maps the robot uses to navigate by. All maps must belong to a site, which is the overall container for one or more maps used in the same facility. A site may, for example, have one map per floor or one per section of a large production hall. Maps must be contained in the same site for the robot to be able to move from one map to another.



Name	Created by	Functions
Test Site		
Testmap	Distributor	
ConfigurationSite		
ConfigurationMap	MiR	
MiR HQ		
Production	Distributor	

## Import and export sites

A site can be exported and imported into other robots.

A site contains the following information:

- Zones
- Cart calibrations
- Cart types
- Carts
- Dashboards

- Docking offsets (for the positions - not the global ones for the robot)
- I/O modules
- Maps
- Marker-types
- Missions
- Mission groups
- Path guides
- Path guides positions
- Paths
- Position transitions
- Positions/Markers
- Robot name
- Sessions (the site file itself)
- Sounds
- User group permissions
- User groups
- Users
- Widgets

To export a site, select **Export** next to the site you want to export. The exported file is named [Site name]\_[Robot name]\_[SW version]\_[Date].site

To import a site, select **Import site** and select the site file.



Site files must be imported to a robot with the same software version as the robot the site file was exported from. If you want to import a site file from another software version, you must update or downgrade your robot to that version first, import the file, and then update or downgrade back to the desired software version.

## Create map

To create a map, first enter a name for the map and select the site the map should belong to. When you select **Create map**, you are directed to the map editor where you find the tools to draw the map and add various features.

The screenshot shows a 'Create map' dialog box. At the top left is the title 'Create map' and a sub-instruction 'Create a new map.' with a help icon. At the top right is a 'Go back' button. Below the title is a 'Name' field containing 'Testmap'. Underneath it is a 'Site' dropdown menu set to 'Test Site', with a 'Create / Edit' button next to it. At the bottom are two buttons: a green 'Create map' button with a checkmark icon and a grey 'Cancel' button with a cancel icon.

### Name

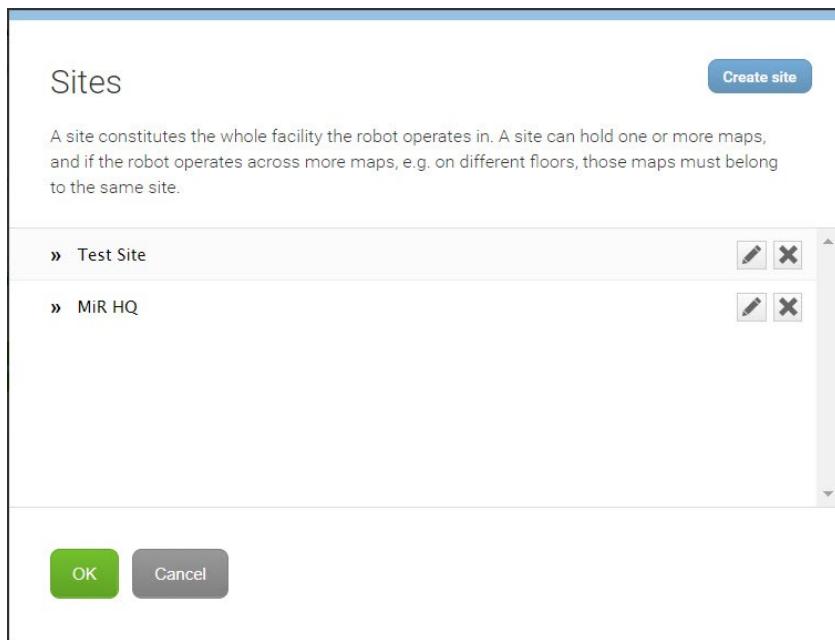
Enter a name that describes the map.

The name is used to identify a certain area of the site.

One way of naming maps is to select names that relate to the area of the map, for example, Ground floor or Hall A.

### Site

Select which site the map should be part of, or select **Create/Edit** to create a new site or edit the name of an existing site.



Select **Create site** to create a new site. Name the site and select **OK**.

Select **Create map** to save the map.

## Mapping tools

The built-in mapping functionality makes it possible to draw a map by manually driving the robot around the facility using the joystick while the robot uses the laser scanners to map the area.

### Two mapping methods

Recording of maps can be done using two different mapping methods: Cartographer (default method) and Hector.

- The Cartographer method records multiple smaller maps and then compiles those into one map after the recording session.
- The Hector method records and compiles data in one map during the mapping.

The tools used are the same for both methods, and when you start mapping you'll not be able to see which method is activated. However, to obtain the best result, there are different mapping patterns recommended for the two mapping methods.

- Cartographer: Mapping in a circular pattern and closing loops.
- Hector: Mapping in a branch pattern, while going back over your own tracks several times.

To change between the two mapping methods, go to **System > Settings > Mapping**.



For more information about creating maps, see your product's user guide or contact your distributor for the guide *How to create maps*.

The map recording and editing tools are all found on the icon tool bar, and the drop-down list contains all the features you can add to your map. Different tools are displayed on the icon toolbar depending on which feature you have selected from the drop-down list.



## Mapping tools



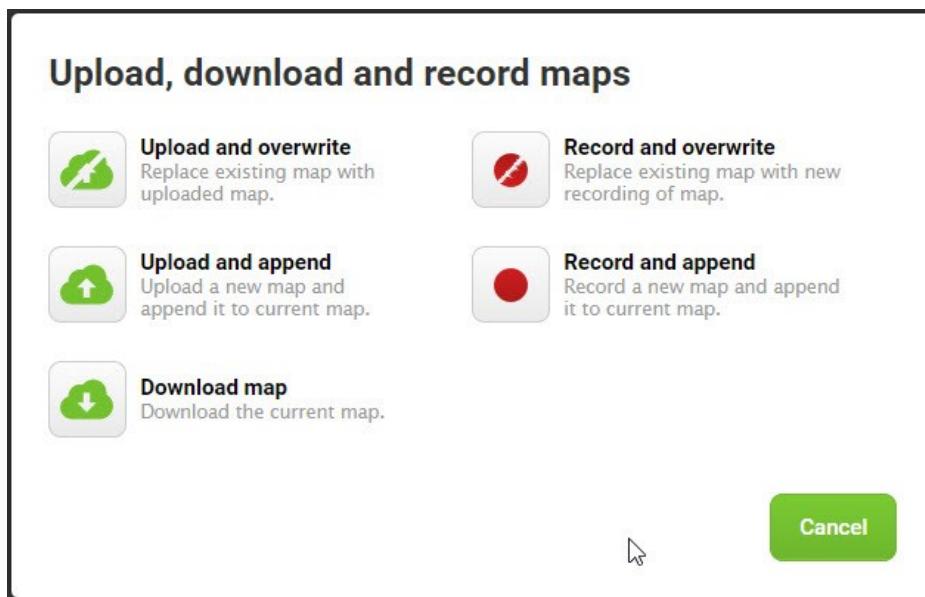
Press the 3-dots icon to open the **Upload, download and record map** dialog box. This toolbar has options for recording a new map or uploading/downloading existing ones.

You may upload a map from your computer in .png format. For example, if CAD drawings of the facility are available, it is possible to use those after converting them to .png instead of mapping the area with the robot, but you can also upload maps previously created with the robot and downloaded to your PC.



When you download a map, only the recorded map data is saved. Any added features such as positions and zones are not saved with the map file. If you want to save a map including all details, you can do so by exporting the whole site that the map belongs to.

The Download, upload and record map dialog has the following options.



- **Upload and overwrite**

The Upload and overwrite option erases the existing map and replaces it with the map you upload.

- **Upload and append**

The Upload and append option adds the uploaded map to the existing one.

- **Download map**

The Download map option saves the map to your PC as a PNG file.

- **Record and overwrite**

The Record and overwrite option erases the existing map and replaces it with the map you record.

- **Record and append**

The Record and append option adds the recorded map to the existing one.

## Recording a map

Select **Record and overwrite** or **Record and append** to start the mapping engine. If you select **Record and overwrite**, you will be asked if you want to overwrite current data.

A blinking icon indicates that the recording has started, and you can now activate the joystick and start moving the robot around the area.



The joystick switches to medium speed when used during mapping. This ensures better coverage of the mapped area.

As the robot moves, the laser scanners will detect physical obstacles, and those will be recorded in the map as walls. In the editing afterward, you can remove all obstacles that should not stay on the map, such as carts or boxes that were present at the time of recording but will not stay permanently.

During mapping, you may add positions based on the current position of the robot. If necessary, the positions can be edited after the mapping is finished.

To stop the mapping engine, press the record icon in the upper-left corner of the recording window.



After the mapping engine stops, you can rotate the map and align it using the grid.

Drag the map to rotate it. Route planning works better with horizontal or vertical paths. For example, if there is a long passage, route planning works best when the passage is either vertical or horizontal, that is when its walls on the map are parallel to the map borders.

On desktop computers, holding the **Shift** key when dragging the map rotates the map in increments of 11.25 degrees.

## Warehouse map

Edit and draw the map 

 Go back



When you have finished the editing, select the check mark in the upper left-hand side corner to save the map. You'll be asked if you want to activate the new map. If you do this, you'll be able to see the robot on the map and may create positions using the robot's live-view.



The grid and the rotation feature will only be present this one time. If you want to open the grid and rotation feature again, you must download the map, and then upload it again.

## Editing a map

When the map recording is done, modify the map by removing unwanted "noise", adding virtual walls, preferred or unpreferred drive zones, positions, and other features to get a reliable map that allows the robots to maneuver smoothly and efficiently in the area.

## Left-hand side tool bar

The left-hand side tool bar has the basic tools for saving, undoing, and navigating in the map you are working on. Furthermore, different tools appear on the tool bar depending on which map layer you select from the Object types drop-down list. These are presented on the following pages.

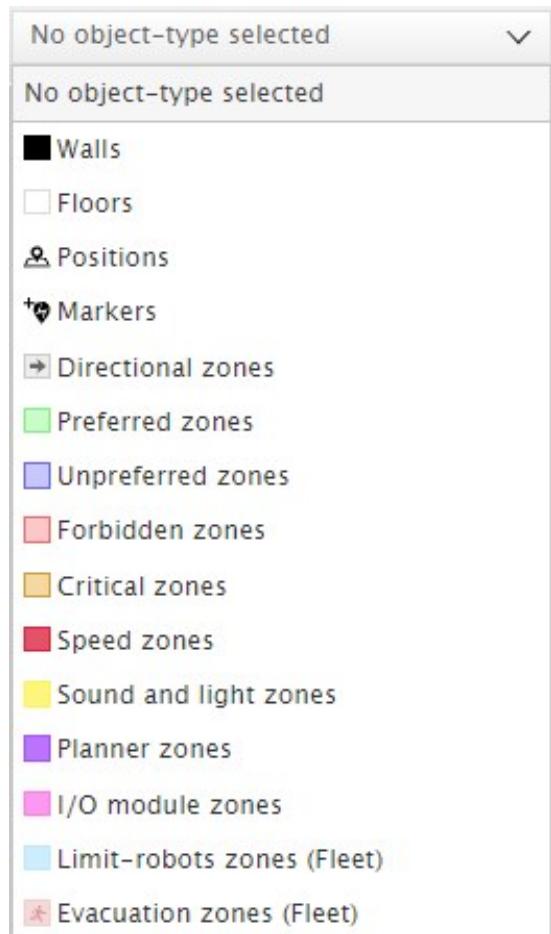


The toolbar contains the following elements:

- **Find position**  
Select the magnifying glass to search for a position on the map.
- **Download, upload and record map**  
Select to upload, download and record maps.
- **Undo icon**  
Select one or more times to undo your last operation(s). While you are drawing a shape or line in the map, the Undo tool is not available. But as soon as you finish by selecting the check mark, you can undo the whole shape or line.
- **Save icon**  
Select to save the changes to the map. For the changes to take effect, you'll need to reload the map.
- **Navigate icon**  
Select to view the map with all added details, and drag to move the view.

## Select object list

The Select object-list contains all features that you can add to the map, such as markers, positions, zones, walls, and floors. See detailed descriptions in [Object types on page 74](#).



## Right side tool bar

The right side tool bar has tools for controlling the map view and synchronizing the robot's position with the map view.



- **Send robot to target**

Select to move the robot to any map location. Use this tool to move the robot to a spot without having to create a position.

- **Show the whole map**

Select to zoom out to view the whole map.

- **Keep robot in center**

Select to view the robot in the center of the window. Otherwise, the map is centered.

- **Adjust robot position**

Select one or more times to adjust the robot on the map. When the red lines showing the scanners' live view align with the map lines (walls), the robot is adjusted. It may be necessary to first set the start position (see below).

- **Set robot's start position**

When you activate a map, the robot's actual position in the physical environment must be synchronized with the map. This is done by trying to align the robot's live sensors (shown as red flickering lines) with the black lines of the map. Select somewhere on the map that matches the robot's physical location as much as possible, and rotate the robot icon to turn the robot correctly, then select **Adjust robot position** a couple of times until the lines overlap. Setting the start position might require a couple of retries.

- **Zoom in**

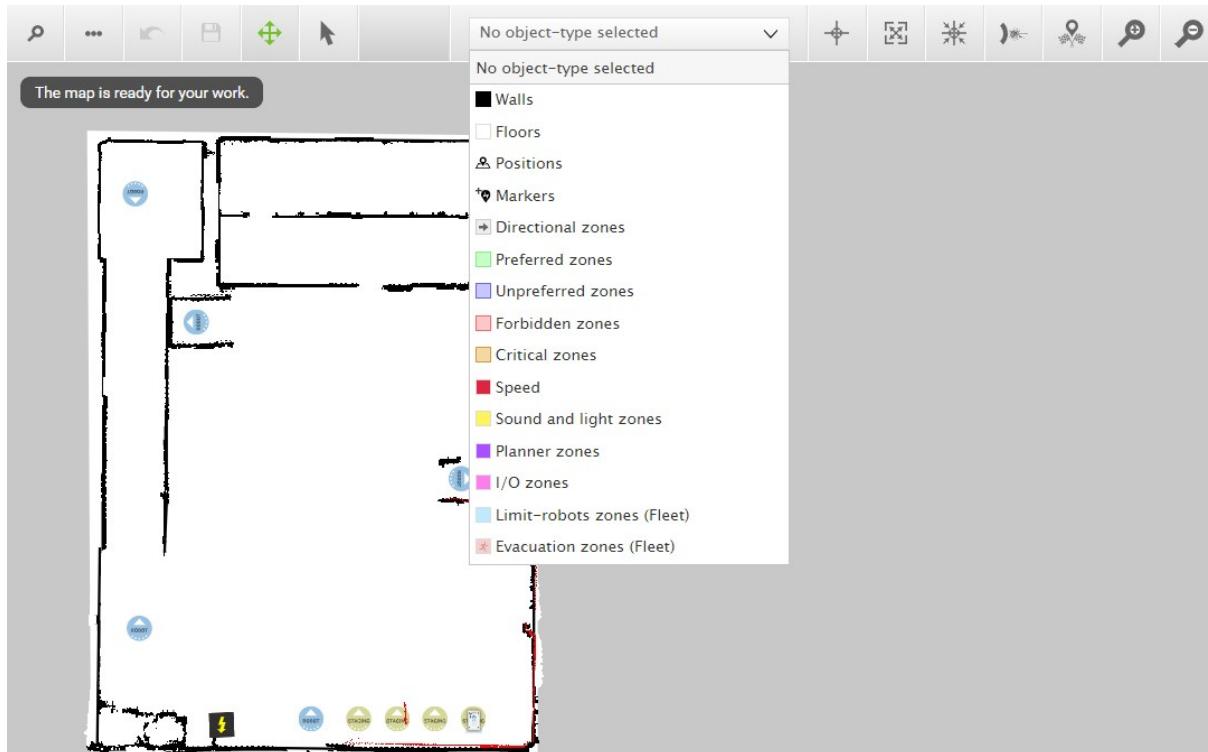
Zoom in on the map.

- **Zoom out**

Zoom out on the map.

## Object types

The Object types list lets you add elements such has walls, positions and zones to your map. **Walls** and **Floors** let you remove unwanted objects and add straight lines to create a more legible map. The other objects define the positions and markers robots can go to as well as different types of zones that set the rules for where and how the robots move.



### Walls

When mapping, physical objects detected by the scanners are recorded as "walls" in the system. These mark areas on the map were the robot cannot drive.

Apart from real walls, these objects could also be shelves, chairs, tables, and even people passing by. Some of these recordings are regarded as "noise" and will, if they are not removed, potentially send the robot on unnecessary detours during its path planning. It is therefore recommended to remove the objects that are not permanent.



Noise refers to recorded data that originates from interfering elements. This can be physical obstacles that make the robot record walls where there are none or more subtle interferences that can make recorded walls appear pixelated.

Use the **Eraser** tool to remove recorded walls, and the **Erase by selection** tool to remove walls you've drawn.

Use the **Draw new line** tool to add new walls to the map, and replace the coarse pixelated lines. The tool works by adding lines between each point you add to the map. Select the check mark when the line is done.

Use the **Select shape or line** tool to change an added object. You can add extra points or move the existing ones to change the shape. To erase a whole shape, select the **Erase shape or line** tool and select a shape to delete it.

## Floors

When mapping, the floor is created automatically. The floor on the map marks the areas where the robot is able to drive. You can use the **Floor** tool to touch up the existing floor if there are areas where the floor is missing.

Use the **Eraser** tool to remove unwanted floor from the map.

Use the **Draw a new shape** tool to add a new floor or patch up the exiting one. The tool works by filling the area between each point you add to the map with gray color. You may add as many points as needed and drag to where you want them on the map. Select the check mark when the shape is done. The gray shape will be converted to white indicating that it represents floor.

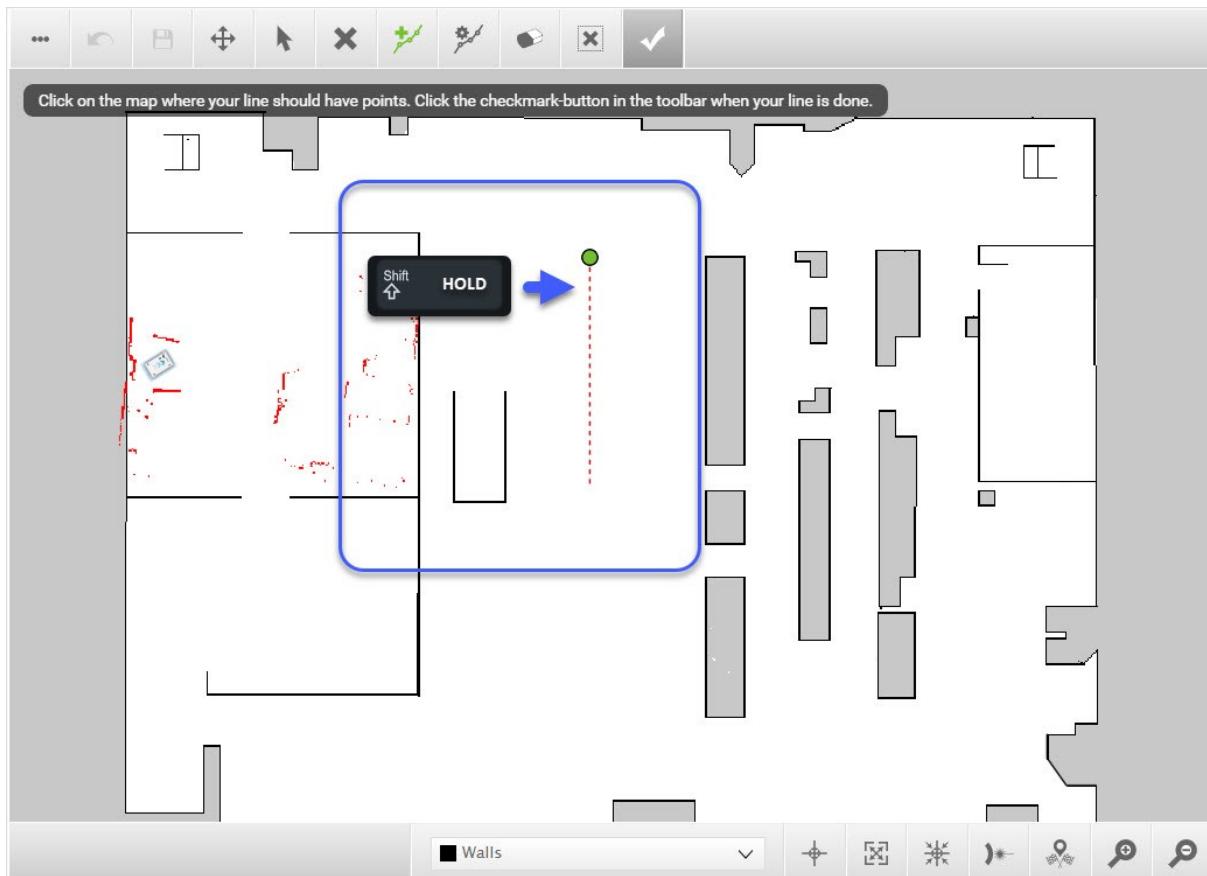


When you are using the floor tools, you cannot affect the walls on the map. If you make a shape that overlaps walls while you are using the **Draw a new shape** tool, the walls will not be removed from the map when the shape is converted to floor.

Use the **Select shape or line** tool to change an added object. You can add extra points or drag the existing ones to change the shape. To erase a whole shape, select the **Erase shape or line** tool and select a shape to delete it.



If you want straight lines, hold down the shift key while drawing a line or an area.



## Positions

Positions are defined as X-Y coordinates in the map and are used in missions.

Positions are used either as destination positions or as waypoints on a route. To define a position, select the **Position** tool, then select somewhere on the map, and rotate the position icon until the arrow points in the direction you want the robot to face at this position. In the dialog window that opens, it is then possible to adjust the position and the orientation manually or select **Use robot position** to use the current position of the robot.

When you select overlapping positions, a list of the overlapping positions is displayed. This enables you to easily select the position you want.

## Pick a position

 Cancel

- Position1 / Robot position
- Position2 / Robot position
- Position3 / Cart position
- Position4 / Shelf position



The **Use robot position** button is available only if you are editing the currently active map.

### Optional positions:

- Elevator and elevator entry positions are used by robots that are inside or waiting for an elevator. These should only be used if the robot is connected to MiR Fleet and MiR Fleet has an elevator set up.
- Emergency positions are positions that the robots go to when commanded to evacuate a zone from MiR Fleet.
- Cart positions for picking up and dropping off carts are available if a MiR hook is applied.
- Shelf positions for picking up and placing shelves are available if a shelf lift is applied.
- Staging positions used as waiting positions become available when the robot is part of a fleet.



Hook, Shelf, or Fleet must be enabled in the Features section under **System > Settings > Features** before the positions can be viewed.

## Markers

Markers are position types used by the robot to dock to physical V, VL, L, or straight bar-shaped objects. Markers are used when you want the robot to position itself accurately relative to a physical object.

To define a marker, first place the robot either facing front or rear to the marker, depending on which way you want the robot to dock. For charging station markers, the robot must always be placed facing front.

When you select the **Marker** tool, the quickest way to set the position is to use **Detect marker**. If the robot can detect the marker, the position, offset, and orientation fields will automatically fill. Otherwise, move the robot a bit closer, and try again. The values can be adjusted manually afterward if required.



The **Detect marker** button is available only if you are editing the currently active map.

You can see and edit the entry position of a marker by selecting the marker and then **Show entry position(s)**. The entry position stays visible until you select the marker again and select **Hide entry position(s)**.

## Zones

Zones are either used to influence the robot's global planner or to trigger actions automatically when a robot enters the zone—see *Figure 4.2* for the division of zone types. Zones that trigger an action always have zone settings that you must set when creating the zone. These zones apply both when the robot operates autonomously and when it is driven in Manual mode.

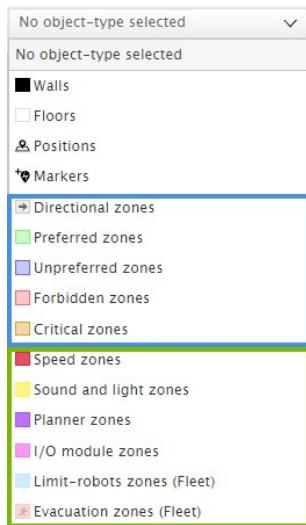


Figure 4.2. Zones that only affect the robot's global planner are in the blue square, and the zones that only take effect when a robot enters the zone are in the green square.

It is possible to create overlapping zones so that multiple events have affect at the same time, for example blinking and slowing down the speed of the robot when it drives in a certain zone.

Each zone has its own color on the map. To add a zone, select it on the drop-down list, then select the shape or line tool on the icon bar and draw the shape or line where you want it on the map.

### **Draw a new shape**

To create a zone shape:

- In the toolbar, select **Draw a new shape** and create a shape by placing the corners of the shape on the map.
- You can edit the shape by dragging the corners.

### **Draw a new line**

To create a zone line:

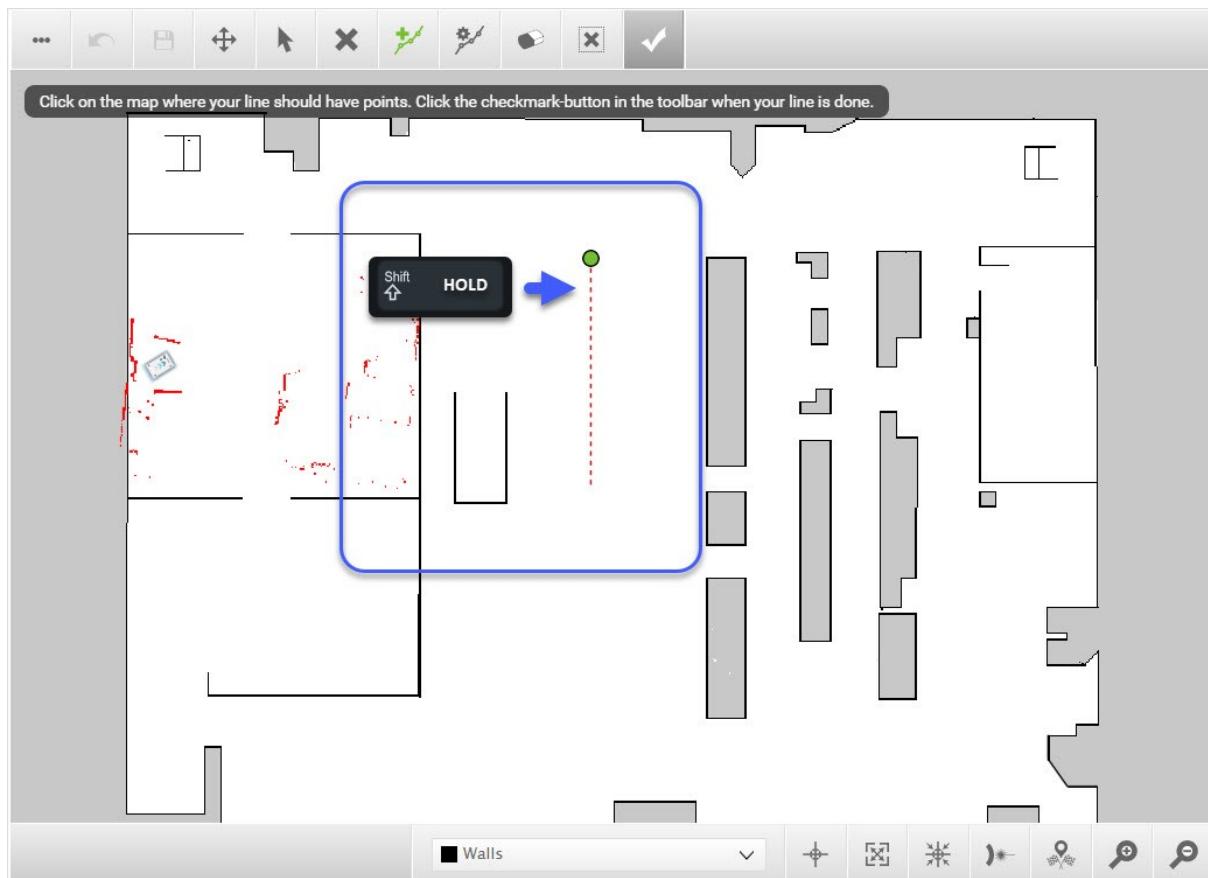
- In the toolbar, select **Draw a new line** and create a line by placing points on the map.
- Select **Line settings** to change the width of the line. Use one of the presets or enter a custom width. Select **Close** to save line settings.



You can only create zone lines for zones that affect the global planner.



Hold down the shift key while drawing a line or an area if you want straight lines.



Select the check mark on the tool bar to finish the shape or line.

To edit or remove a shape or line, select the zone type on the drop-down list. Then, to edit, select the **Select shape or line** tool and select the object. You can change a shape or line by pulling the points, adding or removing points, or by changing the thickness of a line. To add extra points, first select an existing point, then select where you want to add the point. Pull to change the shape if needed. To delete a line or shape, select the **Erase shape or line** tool and select the object to delete.

**Zone settings:** Zone settings allow the user to customize a certain zone to their needs. One or more actions can be set. While the robot is in the zone, it will perform the actions. When the robot leaves the zone, it will go back to the default settings.



Only zones that trigger an action have zone settings—see Speed, Sound and light zones, Planner zones, I/O module zones, Limit-robots zones (Fleet), and Evacuation zones (Fleet).

## Directional zones

Directional zones let you organize the motion of robots by specifying which direction the robot's global path can go. When you create a directional zone, you specify its direction, and the map shows the direction with arrows drawn on the zone.

The directional zones apply the following limitations to the robots' global planner:

- The global path is not allowed to move in the direction opposite to the direction of the arrow.
- The global path can move perpendicular to the direction of the arrow or at any angle less than 90° to the arrow.

There are two types of directional zones: directional shape and directional line. A directional shape is a shape on the map with a direction defined. The direction on a shape can have any value from 0° to 360° with an increment of 45°.

A directional line is a line with a direction defined. The direction of a line is from one of its ends to the other end. To reverse the direction of a line, select a line and select **Reverse direction** in the toolbar.

## Preferred zones

The robot tries to plan its global path to go through Preferred zones. It will drive out of the zone to avoid obstacles.

## Unpreferred zones

The robot tries to avoid planning its global path through Unpreferred zones but may go into it if there are no other possibilities.

## Forbidden zones

The robot's footprint never enters a Forbidden zone.

## Critical zones

The robot's global planner can plan a path that brings the robot closer than usual to detected obstacles when there is a Critical zone in that area. This is useful in narrow passages where the robot can just fit. The safety system and local planner are still active, meaning the robot will still avoid and stop for obstacles.

## Speed

The robot slows down or increases its speed when driving in a Speed zone. Slowing down may be used if the robot is driving in a zone with many people, and speeding may be used to traverse a zone free of people and obstacles quickly.

## Zone settings

- **Name**  
Enter a name for the zone.
- **Desired speed**  
Enter the speed (m/s) the robot should drive with in this zone. Default: 0.8, minimum: 0.1, maximum: 1.5 m/s

## Sound and light zones

The robot can play a sound and blink its status lights when driving in this zone. You can use this zone to warn people about the presence of the robot.

## Zone settings

- **Light**  
The robot drives with the status lights flashing.
- **Sound**  
Select the sound you want the robot to play.

## Planner zones

The robot can ignore data from its laser scanners and localize with encoders, decrease its field of view to run smoothly in populated areas, optimize the wait-time and deviation of paths, and ignore obstacles when in this zone.

## Zone settings

- **No localization**

The robot ignores laser scanner data and only uses its motor encoders to localize itself.

Useful for special driving, like ramps.

- **Look-ahead**

Look-ahead is used to define a decreased field of view. Maximum is 3 meters (default).

Minimum is 0.

- **Path timeout**

Maximum amount of time the robot keeps trying to complete the current path before generating a new global path. Default is 5 seconds. Minimum is -1, which means that the feature is disabled.

- **Path deviation**

Maximum allowed distance the robot can deviate from the path before generating a new global path. Default is 0.5 meters. Minimum is 0. Maximum is 3 meters.

- **Ignore obstacles**

With this action, The robot ignores all obstacles detected with the 3D cameras. This is useful, for example, if you experience problems with the robot stopping in front of windows because of sunlight.

- **Obstacle history clearing**

Select how the robot will clear its obstacle history while driving. **No clearing**: the robot remembers all obstacles and only clears those in the field of view of the cameras. **Clear in front of robot**: the robot disables obstacle history in a cone shape in front of the robot,

starting with the width of the footprint and increasing the width by 0.3 m per meter. **Clear all**: the robot disables obstacle history altogether and only avoids obstacles that it detects

with its sensors while driving.

## I/O module zones

The robot activates an I/O module when entering the zone. An I/O module zone may be used instead of controlling I/O activation through a mission.

### Zone settings

- **I/O module**

Select the I/O module you want to use.

- **PLC registers**

**Index:** Index is the register number and spans from 1–200. Registers 1–100 are reserved for integers and registers from 101–199 for floating point numbers.

**Entry action:** An Entry action is used to set a value in a register. The register can be set in three ways: **Set**: sets a value every time the mission is executed. **Add**: adds a value every time the mission is executed. **Subtract**: subtracts a value every time the mission is executed.

**Entry value:** Enter the value that will apply to the Entry action.

**Exit action:** An Exit action is used to set a value in a register. The register can be set in three ways: **Set**: sets a value every time the mission is executed. **Add**: adds a value every time the mission is executed. **Subtract**: subtracts a value every time the mission is executed.

**Exit value:** Enter the value that will apply to the Exit action.

## Limit-robots zones (Fleet)

Limit-robots zones apply only when robots are controlled by MiR Fleet. Only a defined number of robots may be in the zone at the same time. This zone is used to keep an area clear of too many robots, for example in areas where MiR Hook robots unload and pick up carts.

### Zone settings

- **Robot limit**

Enter the number of robots that are allowed to drive in the zone. Minimum is 1.

## Evacuation zones (Fleet)

**Evacuation zones** make it possible to evacuate all robots in case of an emergency situation.

One or more Evacuation zones can be created on the map and will appear on a list under **Evacuation zones**. It is possible to evacuate one certain zone or all zones at once.

Select **Evacuate all zones** to evacuate all zones (in the top bar or under **Evacuation zones**), or select the toggle switch under **Evacuate** next to a specific Evacuation zone to evacuate that zone. All robots will leave the selected Evacuation zones and go to the nearest Emergency positions.

To reset the Evacuation zones when the emergency is over, use the toggle switches under **Evacuated** to deactivate the zones. When the zones are reset, the robot(s) will wait at their Emergency position(s) for new missions.

Evacuation zones should only be used in case of an emergency as all missions are discontinued.



There must be at least one Emergency position per robot when Evacuation zones are applied.

## Zone settings

- **Evacuation zones (Fleet)**

Select whether or not the evacuation zone is active. 0 is inactive and 1 is active.

## Delete map

You can delete maps that are created by you or another member of the user group you belong to.

Select **Delete X** next to the map you want to delete. In the next dialog box, select **X Delete map** to delete the map.

### Delete map

Delete a map

You are about to delete the map with the following details.

» Name	Testmap
» X-position	0
» Y-position	0
» Theta	0

**X Delete map** **X Cancel**

## 4.3 Sounds

In the Sounds menu, you can upload new sounds to the robot or edit the volume and length of the sounds.

Sounds are used in missions and can be used as alerts or to attract people's attention, for example, when the robot has arrived at a position.

### Sounds

Upload and edit sounds. [?](#)

**Upload sound** **Clear filters**

**Filter:** Write name to filter by... **4 item(s) found** **Page 1 of 1**

	Name	Length	Note	Volume	Created by	Functions
	Beep	0:00:11		100	MiR	
	Horn	0:00:07		100	MiR	
	Foghorn	0:00:07		50	MiR	
	Step aside	0:00:03		100	Administrator	

## Edit sound

You can rename any of the user added sounds on the robot and adjust the volume.

Select the **Play** icon to listen to the sound on the robot itself.

Select the **Listen** icon to listen to the sound on your computer.



The screenshot shows the 'Edit sound' dialog box. At the top left is the title 'Edit sound'. To its right is a 'Go back' button. Below the title is a sub-instruction 'Edit data of an existing sound.' with a help icon. The main area contains three input fields: 'Name' (containing 'Step aside'), 'Volume (0 - 100)' (containing '100'), and 'Note' (containing 'Enter a note about the sound...'). At the bottom are four buttons: a green 'Save changes' button with a checkmark, a grey 'Play on robot' button with a play icon, a grey 'Listen' button with a speaker icon, and a red 'Delete' button with a delete icon. A 'Cancel' button is also present.

The **Edit sound** dialog box contains the following fields:

- **Name**  
You can change the names of user uploaded sounds. The names of the standard system sounds cannot be changed.
- **Volume (0-100)**  
The maximum of 100 is approximately 80 dB.
- **Note**  
You can write a small note about the selected sound (optional).
- **Delete**  
You can delete user uploaded sounds from the robot.

Select **Save changes** to save the settings.

## 4.4 Transitions

Transitions are used to handle changeovers from one map to another within the same site. Map transitions are used, for example, where two adjoining production halls have separate maps.

A transition entry consists of two robot positions: a start position and a goal position, one in each map at a physical point where the two maps overlap. Furthermore, it requires pre-defined missions that include **Switch map** actions.



Going from map A to map B and from map B to map A requires two different missions.

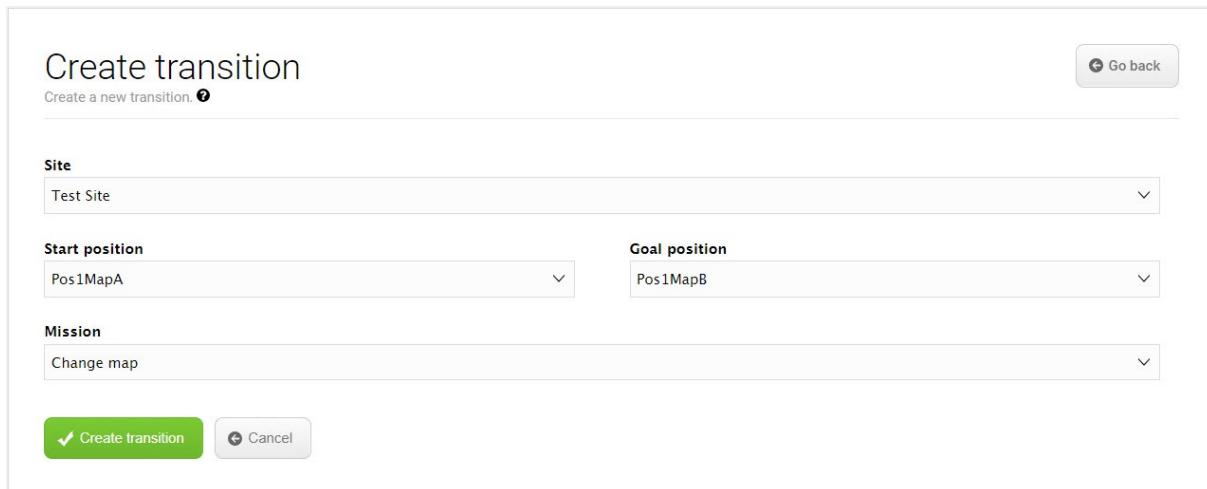
Once the transition is set up in the user interface, the robot handles switches from one map to another automatically. You just set up your mission as you would in a single map-environment, and the system will include the switch positions, the **Switch map** mission, and the transition action automatically. When the robot transitions to another map, it stops for a short while to position itself on the new map.

The screenshot shows a user interface for managing transitions. At the top, there's a header with the title 'Transitions' and a sub-header 'Create and edit transitions'. To the right are buttons for '+ Create transition' and 'Clear filters'. Below the header is a search bar with the placeholder '1 item(s) found' and navigation buttons for 'Page 1 of 1'. The main area is a table with the following data:

	Start	Goal	Mission	Created by	Functions
	Pos1MapA	Pos1MapB	Change map	Distributor	

## Create transition

To create a transition, select a start position and a goal position on two different maps at a point where the maps overlap. The positions must have been predefined as Robot positions on the two maps.



The screenshot shows the 'Create transition' dialog box. At the top left is the title 'Create transition'. Below it is a sub-instruction 'Create a new transition.' with a help icon. On the right is a 'Go back' button. The main area contains three dropdown menus: 'Site' (set to 'Test Site'), 'Start position' (set to 'Pos1MapA'), and 'Goal position' (set to 'Pos1MapB'). Below these is another dropdown menu for 'Mission' (set to 'Change map'). At the bottom are two buttons: a green 'Create transition' button with a checkmark icon and a grey 'Cancel' button with a cancel icon.

The **Create transition** dialog box contains the following fields:

- **Site**  
Select the site in which the two maps are represented.  
Both maps must be part of the same site for a map transition to be possible.
- **Start position**  
Select the start position of the transition.  
The start position must be a Robot position and must be created in advance in a place where the two maps overlap. Start and goal positions must be placed on the exact same spot physically but named differently, for example, "Map A\_posA" and "MapB\_posA" to indicate the relation between the two.
- **Goal position**  
Select the goal position of the transition.  
The goal position must be a Robot position and must be created in advance in a place where the two maps overlap. Start and goal positions must be placed on the exact same spot physically but named differently, for example, "Map A\_posA" and "MapB\_posA" to indicate the relation between the two.

- **Mission**

Select a mission that includes a Switch map action.

The Switch map mission must be created in advance and must include two Switch map actions: the first Switch map action must include the "from" map and the defined start position, and the second Switch map action must include the "to" map and the defined goal position.

Select **Create transition** to save the settings.

## Edit transitions

To edit a transition you must select a start position and a goal position in two different maps at a point where the maps overlap. The positions must have been predefined as Robot positions in the two maps.

### Edit transition

Edit an existing transition. [?](#)

**Site**  
Test Site

**Start position**  
Pos1MapA

**Goal position**  
Pos1MapB

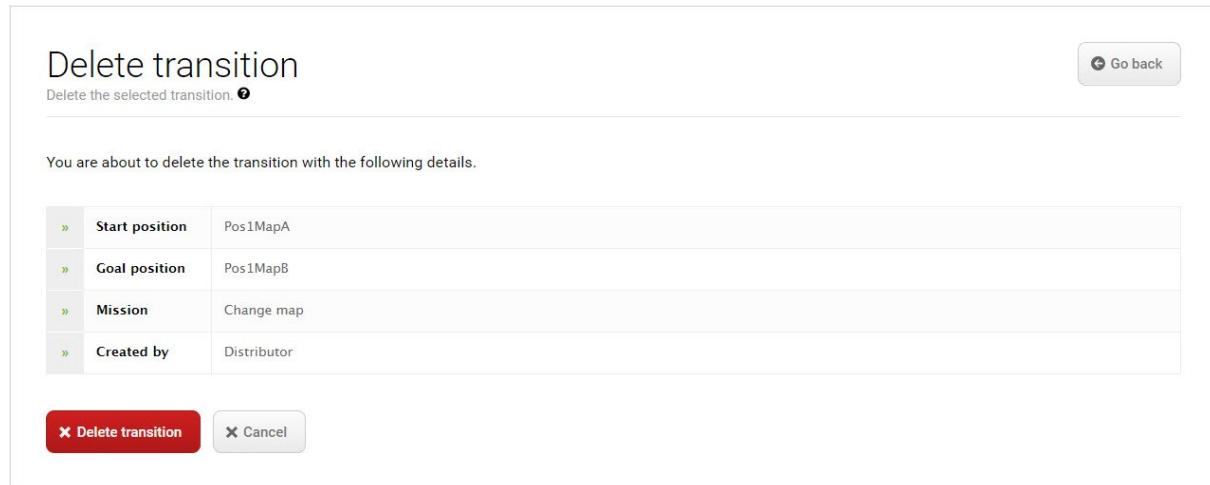
**Mission**  
Change map

✓ Save changes ✗ Delete Cancel

## Delete transitions

You can delete transitions that are created by you or another member of the user group you belong to.

If you delete a transition, the start and goal positions and attached mission are deleted as well.



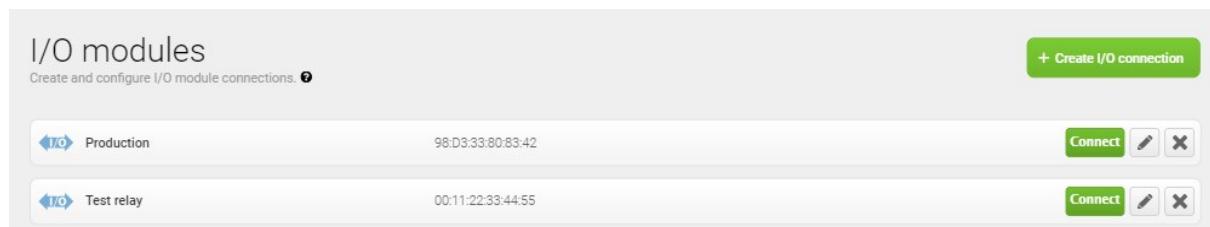
## 4.5 I/O modules

I/O modules (Bluetooth and WISE) are used for receiving and giving input and output to be able to communicate with external devices. Outputs are used to signal information from the robot to the external device, and inputs are used to receive signals from the device.

WISE modules work in the range of 0-3, Bluetooth modules work in the range of 1-4.

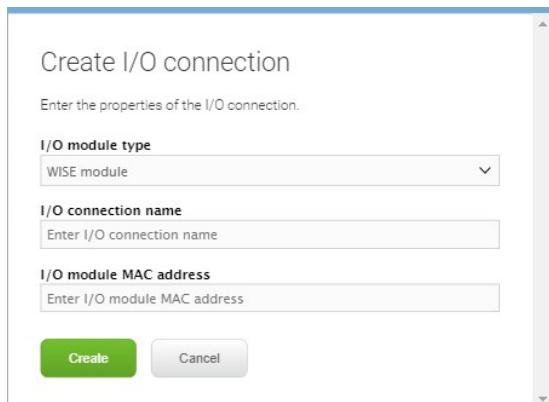


For more information about how to use WISE modules with MiR robots, contact your distributor for the guide *How to use WISE I/O modules*.



## Create I/O connection

To create an I/O connection, you must choose an **I/O module type** (Bluetooth or WISE) and enter a name and an IP address.



Select **Create** to save the settings.

## Delete I/O connection

You can delete I/O modules that are created by you or another member of the user group you belong to.

---

Delete I/O connection?

You are about to delete the I/O connection.

**Delete**

**Cancel**

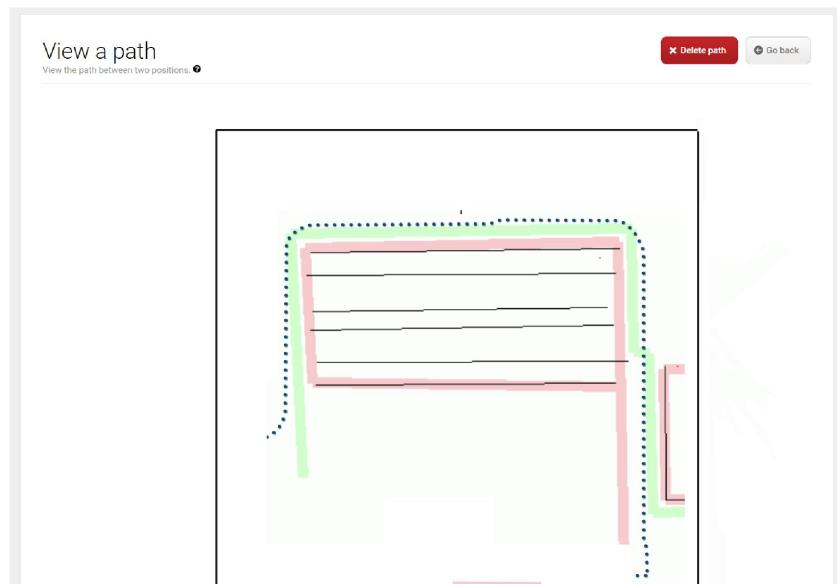
## 4.6 Paths

Paths are saved routes between two positions.

The first time the robot runs the route between two positions, the calculated path is saved and used every time the robot runs the same route, thereby saving time for route calculation. A path is automatically recalculated only in the event that one of its positions is modified.

If you find that an automatically calculated path is unnecessarily long, for example, if the robot had to go around a dynamic obstacle at the time it was created, you can delete it, and the robot will then calculate a new path the next time it runs between those two positions.

You can use Preferred zones and Unpreferred zones to influence where you want the robot to plan its path. After it has created the path using the zones, it will continue to use this path even if you remove the zones. Likewise, if the robot has already created paths between the affected positions, and you add zones afterward, the zones will not take effect until you remove the previous paths.



- **View a path**

The path is shown as a dotted line between two positions on the map. The view can be used to check if calculated paths look appropriate.

## Delete path

Paths can be deleted if you want the robot to recalculate the route between two positions.

Select **Delete X** next to the path you want to delete. In the next dialog, select **X Delete path** to delete the path.



## 4.7 Path guides

A path guide makes it possible to define paths that the robot should follow between two positions by defining waypoint positions between the start and goal. Path guides can be very useful in locations where you want the robot to follow a certain path, for example, along a wall.

In environments where multiple robots operate, an obvious application of path guides would be to create right-hand drive paths where two robots can pass each other without stopping to recalculate each time they meet. This is done by creating one path guide going from A to B and another one in the opposite lane going from B to A.

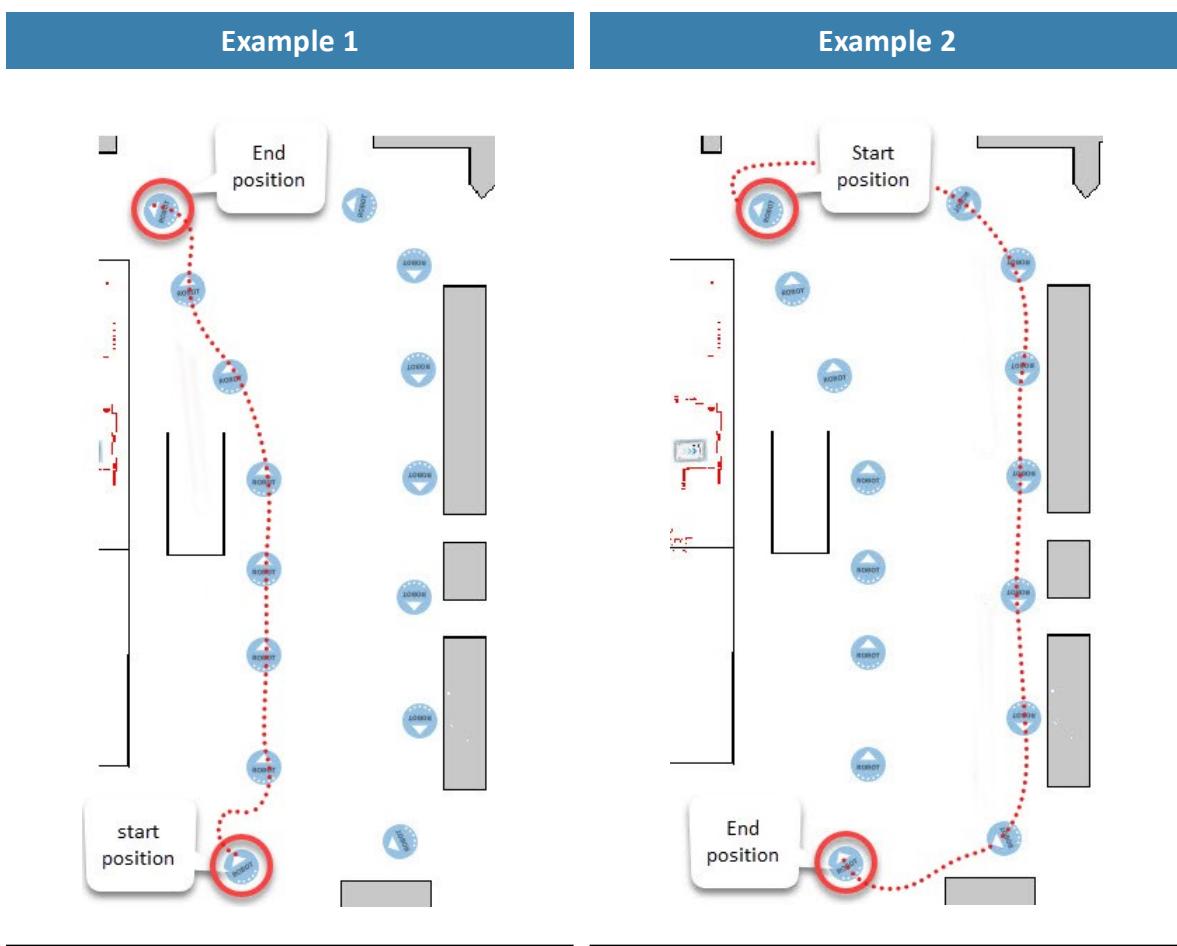
To create a path guide, you must first create a number of robot positions that act as waypoints on the map. The positions must be placed on the path in succession, and they must be oriented in the driving direction.

When the positions are made, you create the path guide. A path guide consists of one or more start positions, one or more goal positions, and a number of waypoints in between. You may use the same path to go between more start and goal positions.

When you set up missions that include positions used as start and end positions, the robot will automatically use the path guide.

## Two examples

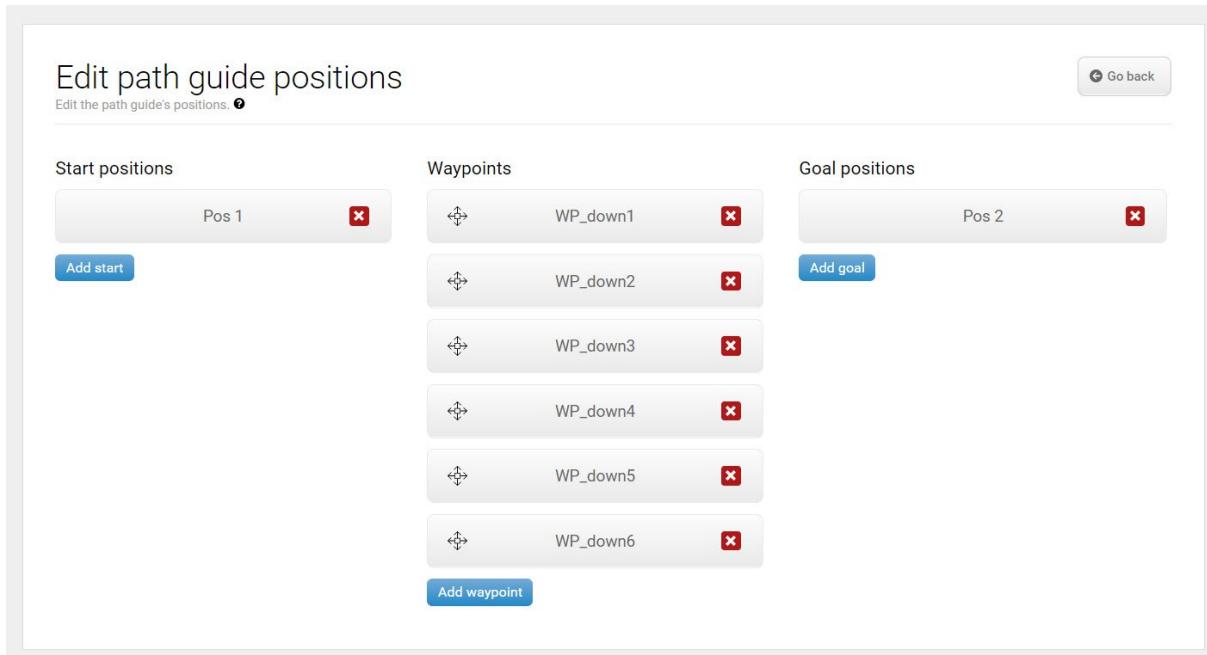
The following two examples illustrate how robots avoid planning around each other every time they pass while crossing the production hall.



Path guide 1 forces the robot to follow one lane going south to north, and Path guide 2 forces the robot to follow another lane going north to south. The same two positions are used for start and end positions but reversed in the two path guides.

## Create a path guide

To create a path guide, first enter a name for the path guide and select the map it should belong to. After selecting **Create path guide**, you are directed to the section where you select start and goal positions as well as the waypoints that make up the path guide between them.



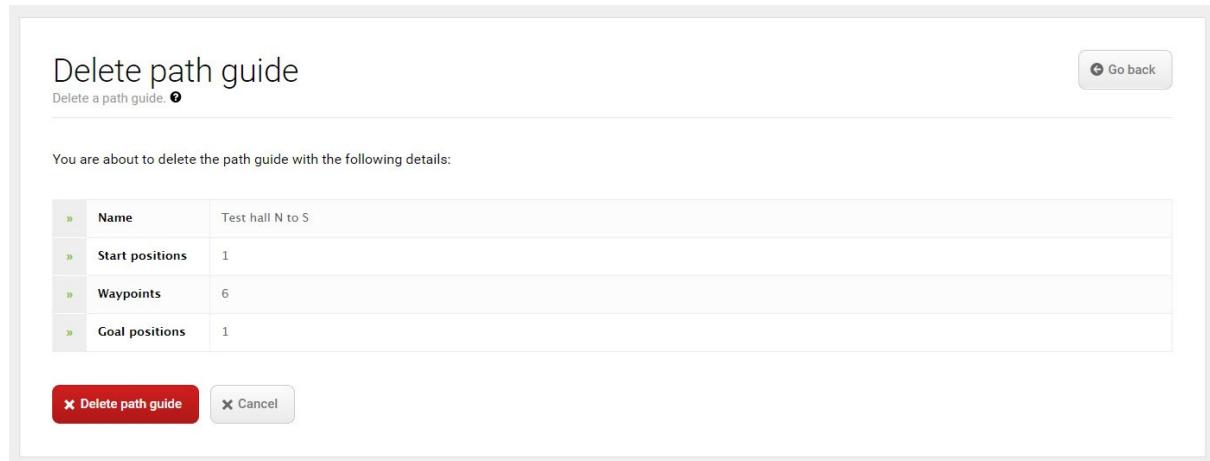
The screenshot shows the 'Edit path guide positions' interface. At the top, there is a title 'Edit path guide positions' and a note 'Edit the path guide's positions.' Below this, there are three main sections: 'Start positions', 'Waypoints', and 'Goal positions'. The 'Start positions' section contains one item, 'Pos 1', with a delete button. The 'Waypoints' section contains six items labeled 'WP\_down1' through 'WP\_down6', each with a delete button. The 'Goal positions' section contains one item, 'Pos 2', with a delete button. There are also 'Add start', 'Add waypoint', and 'Add goal' buttons. A 'Go back' button is located in the top right corner.

- **Add start**  
Select one or more start positions for this path guide.
- **Add waypoint**  
Select the waypoints created for this path guide.  
The waypoints must point in the driving direction.
- **Add goal**  
Select one or more end positions for this path guide.

## Delete path guide

You can delete path guides that are created by you or another member of the user group you belong to.

Select **Delete X** next to the path guide you want to delete. In the next dialog box, select **X Delete path guide** to delete the path guide.



## 4.8 Marker types

To set up the robot for lifting, moving, and placing shelves or tables, marker types with unique names and dimensions must be set up in the robot interface. This will enable the robot to dock and undock correctly.

### Create marker type

To create a marker type, you must first select if it is a **Bar shelf marker** or a **Leg shelf marker**.

**Bar shelf markers** are used for MiR100 and MiR200 robots.

**Leg shelf markers** are used for MiR250, MiR500, and MiR1000 robots.

After selecting the marker type, you must enter the dimensions of the shelf plus positional offsets, which the robot uses to fine-adjust its position when docking to the shelf.

## Create marker type

Create a new marker type. [?](#)

[Go back](#)

---

**Name [?](#)**

**Shelf type [?](#)**

---

**Bar length in meters [?](#)**

**Bar distance in meters [?](#)**

---

**Orientation offset in degrees [?](#)**

**Offset X in meters [?](#)**

---

**Offset Y in meters [?](#)**

[Create marker type](#)
[Cancel](#)

- **Name**

Enter a name for the marker type.

Marker types are used in missions to define pick up and place shelf actions.

The marker type name must be unique and is used to identify the marker type.

- **Shelf type**

Select which shelf marker type you want to create.

The selection must fit the type of shelf you are going to use:

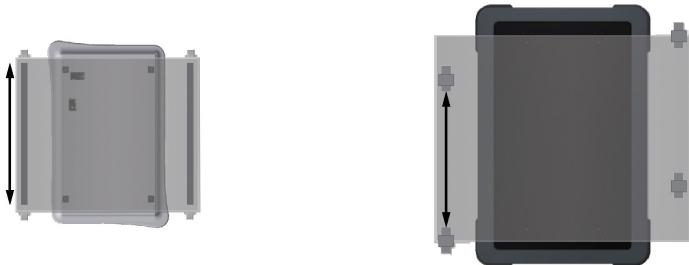
**Bar shelf markers** for MiR100 and MiR200 are for shelves with two side bars.

**Leg shelf markers** for MiR250, MiR500, and MiR1000 are for shelves with four legs.

- **Bar length in meters**

For Bar shelf markers: Enter the length of one of the side bars in meters with up to two decimals.

For Leg shelf markers: Enter the distance between a pair of legs on one side of the robot in meters with up to two decimals.

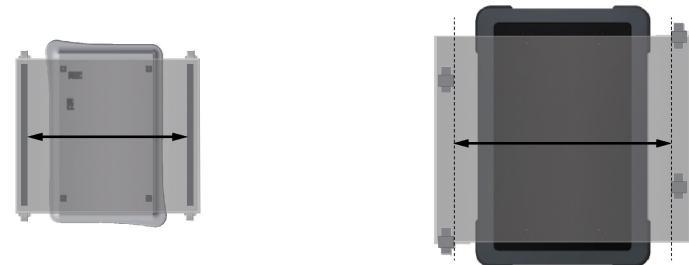


- **Bar distance in meters**

Enter the width of the area under the shelf where the robot can dock.

For Bar shelf markers: Enter the distance between the side bars in meters with up to two decimals.

For Leg shelf markers: Enter the distance between a pair of legs on opposite sides of the robot in meters with up to two decimals.



- **Orientation offset in degrees**

Enter the orientation offset in degrees.

An orientation offset is used to rotate the robot's final orientation when docking to the shelf. A positive rotates the robot counter-clockwise.

Minimum offset value is 0.

- **Offset X in meters**

Enter the marker type's X-offset in meters with up to two decimals.

An X-offset is used to move the robot further forward or backward when docking to the shelf. A positive value moves the robot forward.

Minimum offset value is 0.

- **Offset Y in meters**

Enter the marker type's Y-offset in meters with up to two decimals. A positive value will move the robot to the left.

A Y-offset is used to move the robot further to the left or right when docking to the shelf.

Minimum offset value is 0.

- **Leg asymmetry in meters** (Only for Leg shelf markers)

Enter the value in meters that defines the offset between the two front shelf legs.

The value must be measured on the shelf that you are going to use.

Minimum value: 0 (the legs are symmetric).

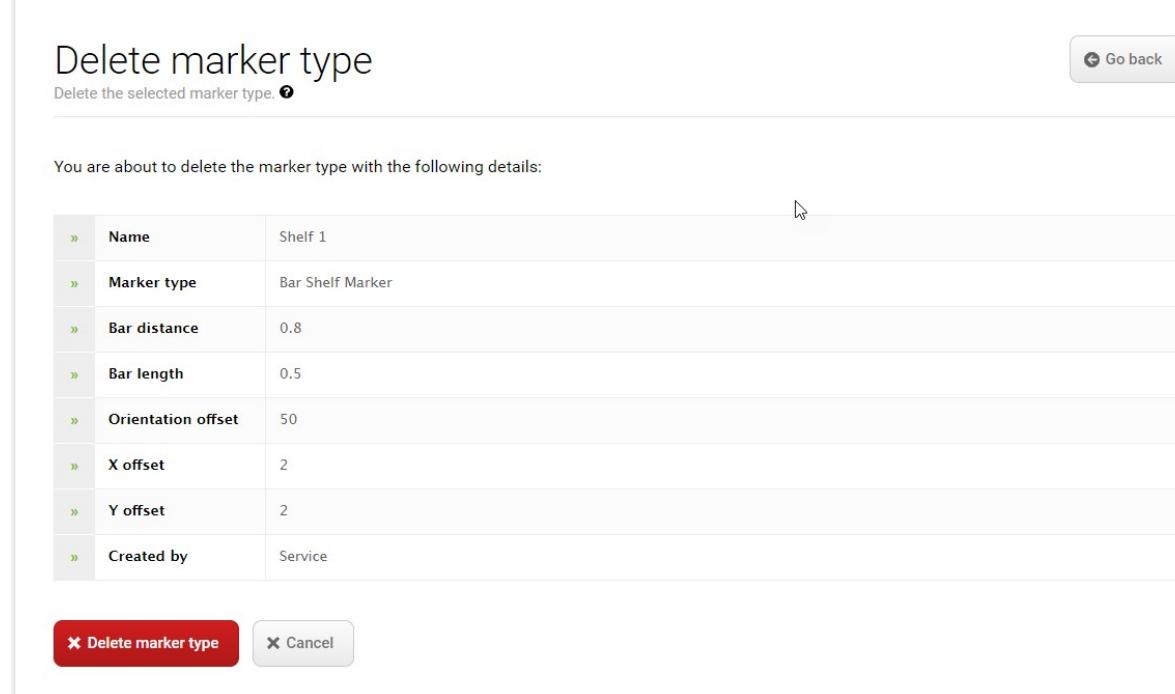
Maximum value: 0.5.

Select **Create marker type** to save the settings.

## Delete marker type

You can delete marker types that are created by you or another member of the user group you belong to.

Select **Delete** ✖ next to the marker type you want to delete. In the next dialog box, select ✖ **Delete marker type** to delete the marker type.



The screenshot shows a 'Delete marker type' dialog box. At the top left is the title 'Delete marker type'. To its right is a 'Go back' button. Below the title is a note: 'Delete the selected marker type. ⓘ'. The main area contains a table with the following data:

» Name	Shelf 1
» Marker type	Bar Shelf Marker
» Bar distance	0.8
» Bar length	0.5
» Orientation offset	50
» X offset	2
» Y offset	2
» Created by	Service

At the bottom of the dialog are two buttons: a red 'Delete marker type' button with a white icon and a grey 'Cancel' button with a white icon.

## 4.9 Footprints

A footprint is the amount of space the robot occupies including its top module and load. It consists of a horizontal shape around the robot, slightly bigger than the robot itself, and a maximum height. The horizontal shape is defined as coordinates relative to the robot's center.

If your robot carries a load with larger dimensions than the robot itself, you must change the footprint to fit the dimensions of the load to ensure that the robot plans its routes so the load does not collide with obstacles. You can choose from the list of default footprints or you can define your own.

## Create footprint

To create a new footprint, first enter a name, then select **Create footprint** to continue to the footprint editor.

The screenshot shows a dialog box titled 'Create footprint'. At the top right is a 'Go back' button. Below the title is a sub-header 'Create a new footprint ?'. The main area contains a 'Name' field with a question mark icon, which is currently empty. Below the field is a placeholder text 'Enter the name of the footprint ...'. At the bottom left is a green 'Create footprint' button with a checkmark icon, and at the bottom right is a 'Cancel' button with a circular arrow icon.

- **Name**

The name is used to identify the footprint.

Select **Create footprint** to save the settings and continue to the footprint editor.

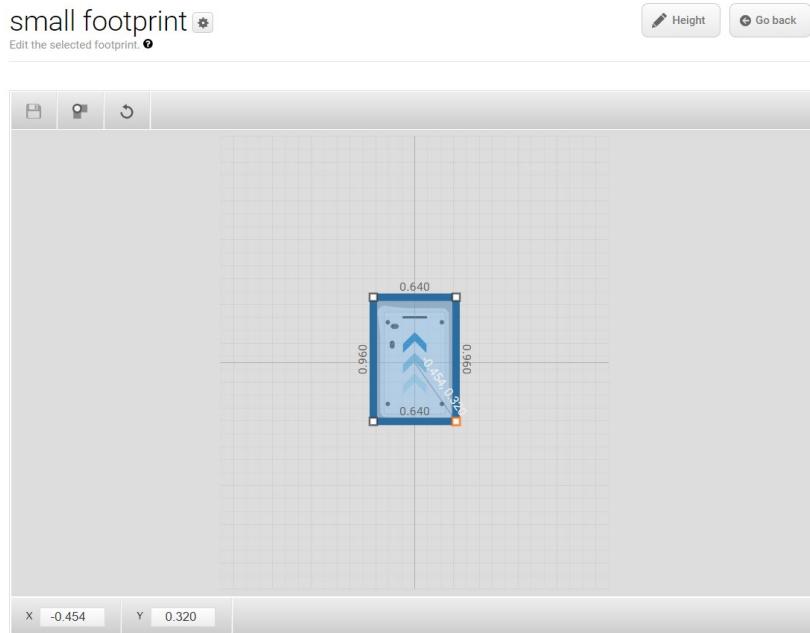
## Edit footprint

You can edit a footprint in either a simple or an advanced mode:

**Simple mode** lets you change the footprint length **X** and width **Y**

**Advanced mode** lets you add and remove points and reshape the footprint as long as it forms a convex shape.

- Use the **Toggle** icon to switch between simple and advanced edit modes.
- Drag the points to change the size and shape of the footprint, or select a point and enter the X and Y values at the bottom-left corner of the editor.
- Select the **Arrow+** icon to add extra points to the shape.
- Select the **Arrow-** icon to remove points from the shape.
- Select the **Edit height** button to open the footprint height editor.
- Select the **Save** icon to save the changes.



## Delete footprint

You can delete footprints that are created by you or another member of the user group you belong to.



If you delete a footprint, any missions where it is used will have an undefined footprint value instead, and the mission cannot be executed by robots.

Select **Delete**  next to the footprint you want to delete. In the next dialog box, select  **Delete footprint** to delete the footprint.

## Delete footprint

Delete the selected footprint. [?](#)

You are about to delete the footprint with the following details:

» Name	Lux Shelf
» Product	MIR100-200
» Created by	Distributor
» Footprint	[[0.45,-0.55],[0.45,0.55],[-0.45,0.55],[-0.45,-0.55]]
» Height	1.6

**X Delete footprint** **X Cancel**

## Migrate footprint

A migrated footprint comes from a site file with a software version that is lower than the version in which the footprint editor was introduced (SW 2.8.0).

You can edit name, robot type, and height of the migrated footprint.

## Migrate footprint

Edit migrated footprint [?](#)

**Name** [?](#)

**Select robot type** [?](#)

**Height** [?](#)

**✓ Migrate footprint** **X Cancel**

## 4.10 Users

All users of the robot, from daily operators to system administrators, must have a user profile in the system. Users are administered in the Users section where you set up, edit, and delete system users.

### Users

Create and edit users [?](#)

**+ Create user** **Clear filters**

Filter: Write name to filter by...		Group: Show all	5 item(s) found	Page 1 of 1
	Name	Username	Email	Functions
	Fleet	fleet		
	Service	service		
	Distributor	distributor		
	Administrator	admin		
	User	user		

## Create user

In **Create users** you set up new users by entering their data, such as name, email, user credentials, and access rights. Access rights are given by associating each user with a User group that delimits which sections of the user interface the user has access to.

**i** User groups should be defined prior to setting up Users.



The screenshot shows the 'Create user' dialog box. At the top left is the title 'Create user' and at the top right is a 'Go back' button. Below the title is a sub-instruction 'Create a new user.' with a help icon. The main form contains the following fields:

- Name:** Test (input field)
- Username:** Test (input field)
- Password:** \*\*\*\* (input field)
- Email address:** Test@test.com (input field)
- User group:** Users (dropdown menu)

Below the input fields are two checkboxes:

- This is a SingleDashboard user
- Allow this user to log in by PIN code

At the bottom of the dialog are two buttons: a green 'Create user' button with a checkmark icon and a grey 'Cancel' button with a cross icon.

The Create user dialog box has the following fields:

- **Name**

Enter the name of the user.

The name is shown in the upper right-hand corner of the web interface when the user is logged in and is not to be confused with the Username.

- **Username**

Enter the name that the user should use to sign in to the system.

- **Password**

Enter a password that the user should use to sign in to the system. Passwords are case sensitive.

Users can change their own password when logged in by selecting their login name in the upper right-hand corner of the window and changing the password in the window that pops up.

- **Email address**

Enter the user's email address. Email addresses can be used as part of a mission, for example, to notify a user about a completed mission.

- **User group**

Select a user group for the user. Each user must be attached to a pre-defined user group. The user group specifies which parts of the system the user has access to. User group permissions are defined for each system command or feature and are granted as read-only or write permissions.

- **SingleDashboard user**

Select the option if the user's only task is to control the robot(s) from a dashboard, for example, if the user's task is to start missions from a tablet attached to a top module.

SingleDashboard users do not have access to any other parts of the user interface.

Select a dashboard for the SingleDashboard user.

When the SingleDashboard user signs in, the selected dashboard will be the one that's available to this user.

- **PIN code**

Select this option if the user is allowed to enter the system using a PIN code. We only recommend this for users with minimum access.

Each PIN code user must have a unique code.

Select **Create user** to save the settings.

## Edit user

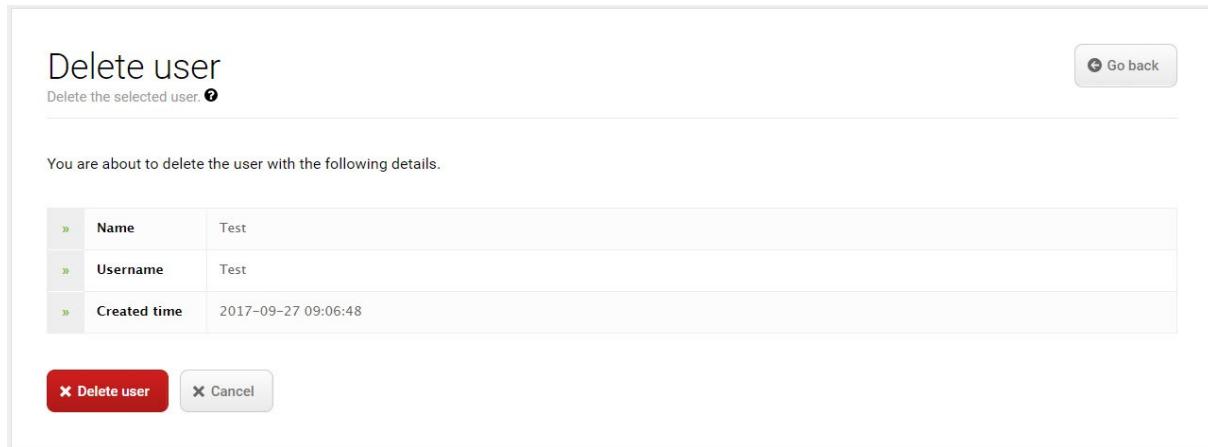
In **Edit user** you can change the settings of a user's profile.

Any of the settings can be changed, except for the password. Users can change their own passwords by selecting the user name in the upper right-hand corner of the window and changing the password in the **Edit user** dialog.

## Delete user

When you select **Delete user**, all settings and updates made in the system by the user in question stay unchanged. Only the user's personal data is removed.

Select **Delete X** next to the user you want to delete. In the next dialog box, select **X Delete user** to delete the user.



## 4.11 User groups

The User groups section is used to create user groups and assign permissions to each group.

A user group defines which sections of the user interface users have access to and whether the access rights should involve viewing only or give full write access. To edit permissions for a group, select the key icon next to the name of the user group to open the User group permissions section.

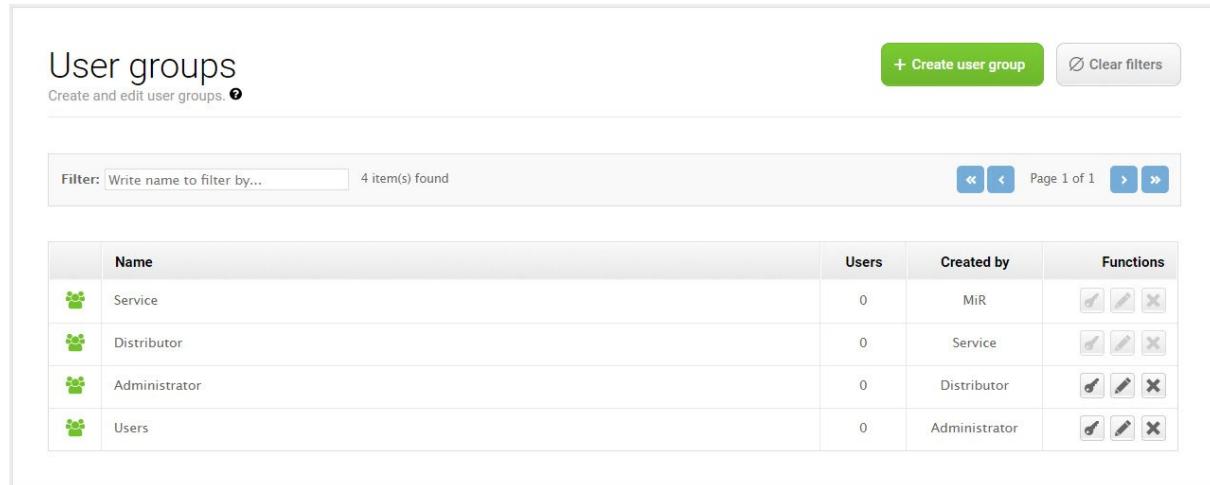
The MiR user interface comes with a number of default user groups:

- **Distributors** have full read/write access to the user interface and can administer the permissions of the Administrators and Users groups.
- **Administrators** per default have full read/write access to the user interface and can administer the permissions of the Users group.
- **Users** per default have access to view the whole user interface and permission to create and edit dashboards.

Users with write access to the User groups section can also create additional user groups.



When setting up users in the Users section, each user must be assigned to a user group.



User groups

Create and edit user groups. [?](#)

**+ Create user group** [Clear filters](#)

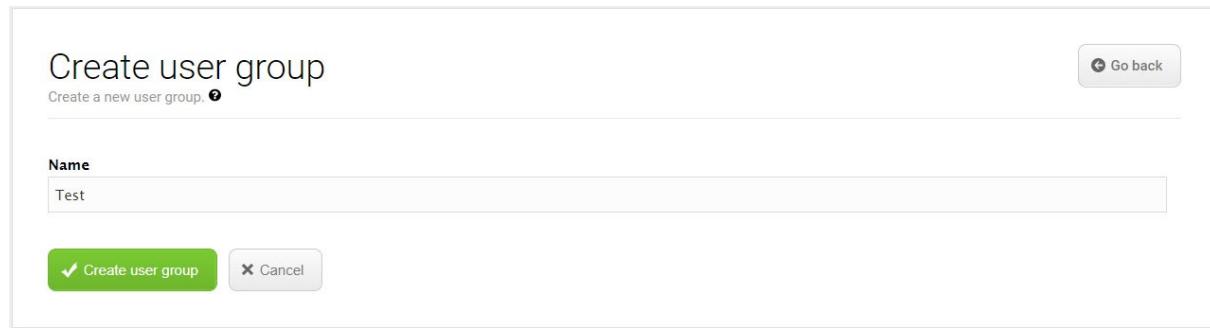
	Name	Users	Created by	Functions
	Service	0	MiR	
	Distributor	0	Service	
	Administrator	0	Distributor	
	Users	0	Administrator	

## Create user group

Besides the default user groups, you can create as many additional user groups as needed. The number of user groups needed depends on how many different tasks and permission levels are required. Several users carrying out the same tasks can belong to the same user group.

Fill out the name field to create a new user group.

You can give permissions to all sections of the user interface that you have access to.



Create user group

Create a new user group. [?](#)

**Name**

Test

**✓ Create user group** **✗ Cancel**

The **Create user group** dialog has the following field:

- **Name**

The name must be unique and is used to identify the group of users it represents.

One way of naming user groups is to select names that characterize the tasks of the users in the particular group. For example, a group of users operating the robot by starting and queuing missions could be named Operators.

Select **Create user group** to create the user group.

## User group permissions

You can assign permissions to all parts of the system that are available to the user group the creator belongs to.

Select which sections of the system the user group should have access to. User group permissions are divided into groups of related items, and you can select a whole group or individual items in a group.

The user group will have access to all the items you select for the group. All other items will be visible but not editable to the users of the group.

### User group permissions

[Go back](#)

 Set permissions for the user group. [?](#)

You are currently setting permissions for the user group **Users**.

General	
Section	Write
Control	<input checked="" type="checkbox"/> Write
Dashboards	<input checked="" type="checkbox"/> Write
Remote support	<input type="checkbox"/> Write
Sounds	<input type="checkbox"/> Write
PLC registers	<input type="checkbox"/> Write
Shelf types	<input type="checkbox"/> Write
Carts	<input type="checkbox"/> Write
I/O modules	<input type="checkbox"/> Write
Modbus	<input type="checkbox"/> Write

Maps & positions	
Section	Write
Sites	<input type="checkbox"/> Write
Maps	<input type="checkbox"/> Write

## Delete user group

To delete a user group, you must be signed in as user of the user group that created the group.



When you delete a user group, all users belonging to that group will be deleted as well. To avoid deleting one or more users of the group you are about to delete, go to the **Users** section and associate those users with a different user group.

Select **Delete** next to the user group you want to delete. In the next dialog, select **Delete user group** to delete the group.

### Delete user group

Delete the selected user group

You are about to delete the user group with the following details.

»	Name	Test
»	Created time	2017-09-27 09:10:05
»	Contained users	0

**Delete user group**    **Cancel**

# 5. Monitoring

This section describes the items in the Monitoring menu.



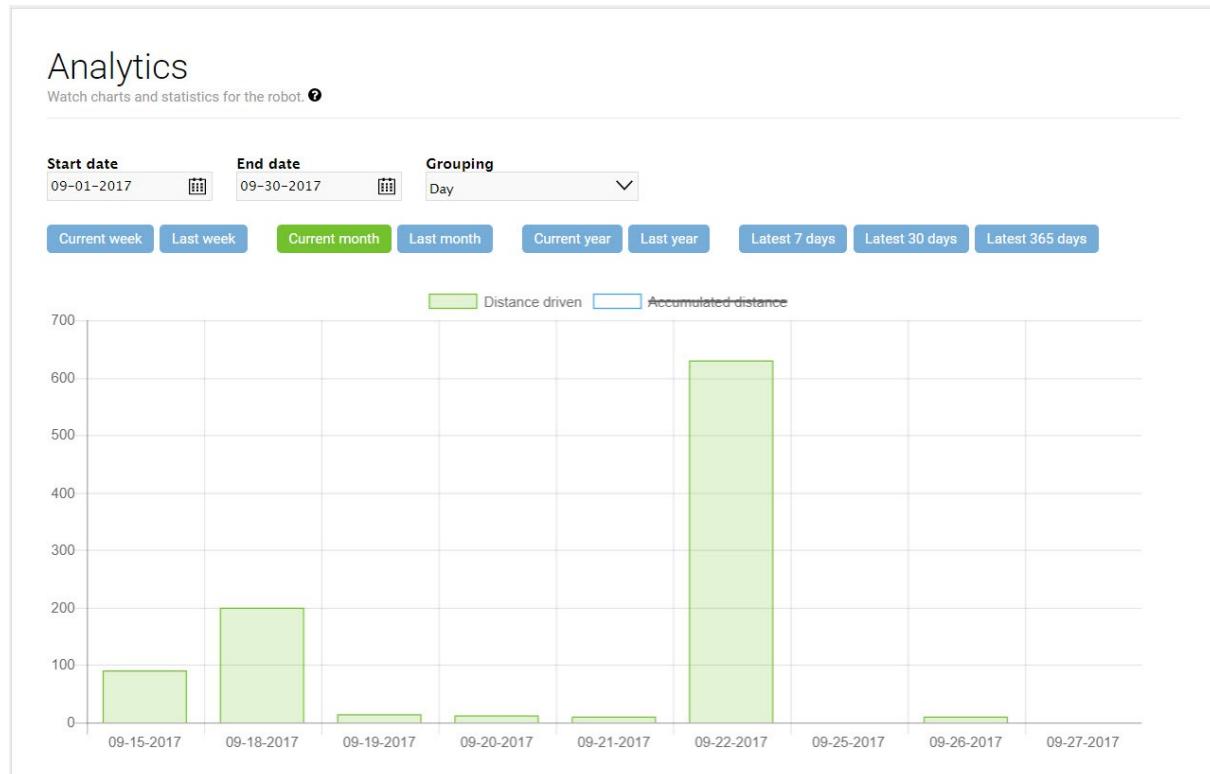
The Monitoring menu contains the following items:

<b>5.1 Analytics .....</b>	<b>114</b>
<b>5.2 System log .....</b>	<b>115</b>
<b>5.3 Error logs .....</b>	<b>116</b>
<b>5.4 Hardware health .....</b>	<b>117</b>
<b>5.5 Safety system .....</b>	<b>118</b>
<b>5.6 Mission log .....</b>	<b>120</b>

## 5.1 Analytics

Analytics gives a graphic overview of the robot's driven distance over a specified period of time.

You can select a period either by specifying a fixed start and end date or by selecting one of the buttons spanning from current week to the last 365 days. In addition, you can choose whether to see a chart per day or per month, and you can see a graph showing the accumulated distance for the selected time period in addition to the default bar graph view.



- **Start date**  
Select the first day of the period you want to see.
- **End date**  
Select the last day of the period you want to see.
- **Grouping**  
Select **Day** or **Month** to set how you want to view the graph.

## 5.2 System log

The system log contains events that are logged by the operating system components. The system log contains information about system state at a given time (shown by color-codes), the affected module, a short explanation, and a time stamp.

The system log is mainly used by technical supporters for troubleshooting.

State	Module	Message	Time
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:38
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:35
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:32
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:29
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:26
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:23
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:20
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:17
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:14
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:11
●	camera_floor_nodelet_manager	No devices connected.... waiting for devices to be connected	04:01:08
●	rosbridge_websocket	Could not process inbound connection: [/rosbridge_websocket] is not a publisher of [/mir_sound]. Topics are [/rosout', 'rosgraph_msgs/Log]]{'message_definition': 'string data\\n', 'callerid': '/mirSound', 'tcp_nodelay': '0', 'md5sum': '992ce8a1687cec8c8bd883ec73ca41d1', 'topic': '/mir_sound', 'type': 'std_msgs/String'}	04:01:07

The System log table has the following columns:

- **State**  
Shows the state of the system with a visual color-indication at the time of logging.
- **Module**  
Shows which module the logged event has taken place in.
- **Message**  
Describes the logged event.
- **Time**  
Records when the event happened in the following format: hh:mm:ss.

## 5.3 Error logs

**Error log** is a list of all detected system errors. Each entry is shown with a description, an indication of which module is affected, and the time when the error occurred.

Error logs are mainly used by MiR technical supporters for troubleshooting.

When further examination of a log entry is required, it can be downloaded in an encrypted file format and sent to MiR Support. Each file contains detailed information plus a recording of the last 30 seconds of robot action(s) before the error occurred.

It is also possible to create a user generated log with a recording of the last 30 seconds of the robot's actions.

Select **Generate log** to record the last 30 seconds of the robot's actions.

Select **Delete all** to delete the entire error log.

Error logs				
<a href="#">Download and delete error logs</a> 				
<input type="text" value="Filter: Write name to filter by..."/> <span>29 item(s) found</span>		<span></span> <span></span> <span>Page 1 of 3</span> <span></span> <span></span>		
Description	Module	Time	Functions	
 Missing	/Hook/Connection	2017-09-27T09:15:43	  	
 Missing – Last Message: No serial device configured'	/Serial Interface/Communication	2017-09-27T08:17:49	  	
 Generated from web interface by 'Service'	User	2017-09-26T11:35:31	  	
 Generated from web interface by 'Service'	User	2017-09-26T11:35:21	  	
 Calibration mission aborted – MoveAction failed (Current session ID and Goal session ID does not match!)	MissionController	2017-09-26T11:34:10	  	
 Missing	/Hook/Connection	2017-09-26T08:59:43	  	
 Missing	/Hook/Connection	2017-09-26T06:28:24	  	
 Missing	/Hook/Connection	2017-09-25T15:05:20	  	
 GoToPositionPrototype aborted – MoveAction failed (Failed to load position from DB)	MissionController	2017-09-22T15:24:39	  	
 No data for >15 sec	/Sensors/Laserscanner (Back)/Communication	2017-09-22T15:05:11	  	

The Error logs table has the following columns:

- **Description**

Describes why the error log was created.

- **Module**

Identifies which of the robot's modules has caused the error, for example /Hook/Connection.

- **Time**

Records when the error log was created in the following format: yyyy-MM-ddThh:mm:ss.

- **Functions**

The icons in this column allow you to download, view, or delete the error log.

- **Download**

Downloads the log entry to your device in an encrypted file format when you select it.

- **View**

Displays information for the selected log entry only.

- **Delete error log**

Deletes the log entry when you select it.

## 5.4 Hardware health

Hardware health allows you to check the condition of the robot's hardware components, such as motor controllers, lasers scanners, and cameras.

The components are collected under groups. Some groups are only displayed if they are enabled under **System > Settings > Features**. If all sub components are OK, the group will be marked with a green dot and OK, whereas if one or more components in a group are not in perfect condition, the group will be marked with a yellow or red dot and read Warning or Error respectively.

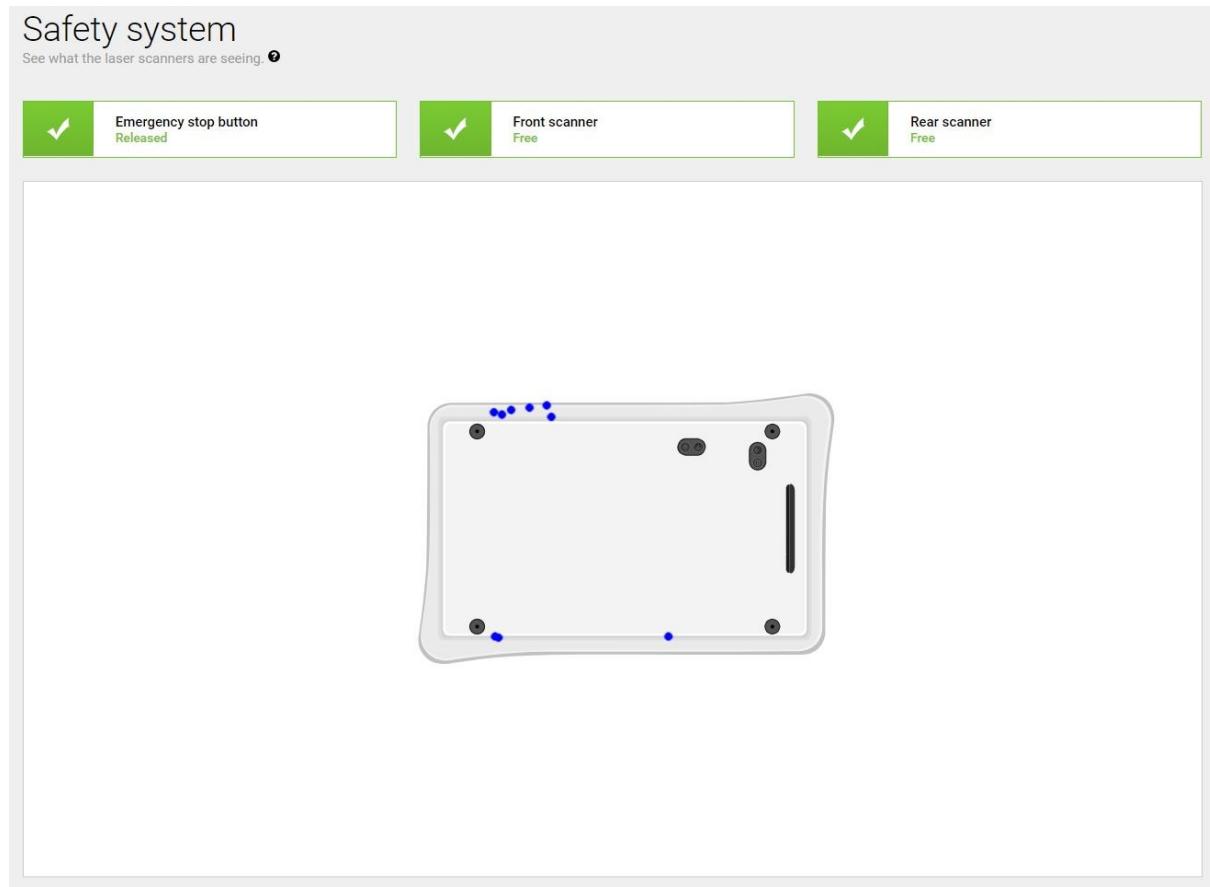
To find out more about the condition, you can expand the group by selecting the arrow next to the group name and see which components are not functioning correctly and why. Each sub component can be further expanded into one or more sub parts for further information on the condition.



## 5.5 Safety system

**Safety system** provides a live view of the input from the laser scanners and the state of the emergency stop button.

The purpose of the feature is mainly to be able to check if the robot has stopped unexpectedly due to a physical obstacle, or if someone has manually engaged the emergency stop button.



## Emergency stop

Released (green) indicates that the emergency stop button is not engaged.

Activated (red) indicates that the button has been manually engaged.

## Front scanner

Free (green) indicates that the laser scanner is not seeing any obstacles.

Blocked (red) indicates that a physical obstacle is blocking the scanner.

## Rear scanner

Free (green) indicates that the laser scanner is not seeing any obstacles.

Blocked (red) indicates that a physical obstacle is blocking the scanner.

## 5.6 Mission log

**Mission log** contains the list of all missions that the robot has executed and the mission that is running now. Select the eye icon in the **Functions** column to see the list of actions executed in a particular mission.

### Mission log

View the mission log.

Filter: <input type="text" value="Write name to filter by..."/>	320 item(s) found			Page 1 of 32	
---	-------------------	--	--	--------------	--

	Mission	State	Message	Start time	Ran for	Started by	Functions
	Transport goods	Done	ActionList was execu...	2018-05-15T14:05:14	0:1:5	Distributor	
	Transport goods	Done	ActionList was execu...	2018-05-15T14:04:02	0:1:12	Distributor	
	Move to parking place	Done	ActionList was execu...	2018-05-15T14:03:48	0:0:14	Distributor	
	Transport goods	Done	ActionList was execu...	2018-05-15T14:01:54	0:1:10	Distributor	
	Move to parking place	Done	ActionList was execu...	2018-05-15T14:01:41	0:0:13	Distributor	
	Transport goods	Done	ActionList was execu...	2018-05-15T13:59:05	0:2:36	Distributor	

The mission log contains the following columns:

- **Mission**

Identifies the name of the mission.

- **State**

Indicates the current state of the mission.

- **Message**

Displays a service message associated with the mission.

- **Start time**

Displays the start time of the mission.

- **Ran for**

Displays the duration of the mission.

- **Started by**

Identifies the user or service that put the mission into queue.

- **Functions**

Select **View** in the Functions column to open the Mission action log for a certain mission.



You can enter a text string in the Filter field to show the missions where either the mission name or the state contains the entered string.

## Mission action log

The mission action log contains the list of actions that the robot has executed within the selected mission and the action that the robot is executing now.

### Mission action log

[View the mission action log](#)

[Go back](#)

Filter:

4 item(s) found

Page 1 of 1

Action	State	Message	Start time	Ran for
move	Succeeded	Position 'Position A' reached..	2018-05-15T14:21:24	0:0:16
move	Succeeded	Position 'Position B' reached..	2018-05-15T14:21:40	0:0:17
move	Succeeded	Position 'Position A' reached..	2018-05-15T14:21:57	0:0:16
move	Succeeded	Position 'Position B' reached..	2018-05-15T14:22:13	0:0:16

The mission action log contains the following columns:

- **Action**  
Identifies the name of the action.
- **State**  
Indicates the current state of the action.
- **Message**  
Displays a service message associated with the action.
- **Start time**  
Displays the start time of the action.
- **Ran for**  
Displays the duration of the action.



Enter a text string in the Filter field to shows the actions where either the action name, the state, or the message contains the entered string.

# 6. System

This section describes the items in the System menu.



The System menu contains the following items:

<b>6.1 Settings .....</b>	<b>124</b>
<b>6.2 Processes .....</b>	<b>143</b>
<b>6.3 PLC registers .....</b>	<b>146</b>
<b>6.4 Software versions .....</b>	<b>150</b>
<b>6.5 Backups .....</b>	<b>151</b>
<b>6.6 Robot setup .....</b>	<b>152</b>
<b>6.7 Triggers .....</b>	<b>154</b>

## 6.1 Settings

**Settings** contains the robot's parameter settings.

The settings are divided into sub groups, and all parameters have context help texts.

Watch and edit settings for the robot. ⓘ

 <b>Mapping</b> Parameters for mapping configuration	 <b>Error handling</b> Settings regarding the handling of error scenarios	 <b>3D cameras</b> Configuration of the 3D cameras	 <b>Battery</b> Settings regarding the battery of the robot	 <b>Planner</b> Settings regarding the planner for autonomous driving
 <b>Localization</b> Settings regarding localization of the robot	 <b>Serial Interface</b> Parameters for configuration of the serial interface	 <b>Motorcontroller</b> Settings regarding the motor controller	 <b>UR interface</b> Settings regarding configuration for the Universal Robot arm	 <b>Docking</b> Parameters for docking to markers
 <b>Relative move</b> Configuration parameters for Relative move	 <b>Calibration</b> Calibration parameters for the individual robot	 <b>Features</b> Enable and disable features in the robot	 <b>Email configuration</b> Set up an email account enabling the robot to send emails	 <b>Distributor data</b> Edit data about the distributor selling the robot.
 <b>Safety system</b> Configures the robot's protective-fields system	 <b>Advanced</b> Advanced configuration parameters	 <b>WiFi</b> WiFi connections for the robot to use	 <b>Date &amp; time</b> Set the robots date and time	 <b>Modified defaults</b> See values different from defaults

## Mapping

In **Mapping** you can change the algorithm used for mapping. It is possible to choose between Cartographer and Hector, see [Mapping tools on page 66](#).

### Mapping

Parameters for mapping configuration

Go back

**Mapping algorithm** Cartographer

Restore default

Select the algorithm the robot uses for creating maps.

Save changes

Cancel

## Error handling

Change how often you are alerted of hardware errors and set timeouts for Protective stops and skid detections.

### Error handling

Settings regarding the handling of error scenarios

 Go back

#### Ignore hardware errors

False



 Restore default

Select **True** to make the robot continue to run even if a hardware error is detected. **Caution:** Only do this if you are very sure of what you are doing!

#### Hardware error timeout

10

 Restore default

Enter the number of minutes until a hardware error reappears after it has been cleared. Enter **0** to disable hardware errors from being cleared. Enter a negative value to disable hardware errors from reappearing until the robot is restarted.

#### Protective stop timeout

-1

 Restore default

Enter the number of minutes a robot may be in protective stop before it is considered an error. Enter a negative value to disable the feature.

#### Skid detection timeout

-1

 Restore default

Enter the number of minutes the robot should stop and wait if it detects a skid condition. Enter **0** to set an infinite waiting time. Enter a negative value to disable skid detection.

 Hide advanced settings

#### Skid detection angular speed threshold

0.3

 Restore default

Enter the threshold in rad/s for how big a difference in angular speed between the gyroscope and encoders is allowed before a skid condition is triggered

#### Skid detection orientation threshold

10

 Restore default

Enter the threshold in degrees for how big a difference in orientation between the gyroscope and encoders is allowed before a skid condition is triggered

## 3D cameras

Configure the robot settings of the robot's cameras including camera setup, camera type, serial number, and filter configuration.

There are two types of camera setups. The settings under Advanced settings differ depending on your robot's camera configuration. If your robot uses a Left and Right setup, or has a top camera, the settings also include the calibration transform values for each camera.

### 3D cameras

Configuration of the 3D cameras

 Go back

#### Camera setup

Top and floor



Restore default

Select the 3D camera setup of the robot. Select **Top and floor** if the robot has one built-in camera and possibly a top camera. Select **Left and right** if the robot has two built-in cameras next to each other.

#### Floor camera type

Intel Realsense R200



Restore default

Select the floor 3D camera mounted on the robot

#### Floor camera serial number

2501011336

Restore default

Floor camera serial number. Camera model: Intel RealSense R200.

#### Floor camera filter configuration

Default



Restore default

Select the filter configuration for the floor 3D camera. Camera model: Intel RealSense R200.

#### Top camera type

None



Restore default

Select the top 3D camera mounted on the robot (if applied)

Show advanced settings

## Battery

Define which battery type the robot is using, how many batteries, and for how long batteries should be recharged. Under advanced settings, you can configure how the battery should be charged once it has almost finished charging.

### Battery

Settings regarding the battery of the robot

[Go back](#)**Using lithium battery** True[Restore default](#)

Select **True** if the robot is equipped with a lithium battery and **False** if the robot is equipped with lead batteries.

**Using two lithium batteries** False[Restore default](#)

Select **True** if the robot is equipped with two lithium batteries.

**Minimum charge current** 1[Restore default](#)

Enter the minimum difference in charge current required to switch on the charging relay and begin charging the robot battery.

[^ Hide advanced settings](#)**Battery percentage threshold for topping up state** 95[Restore default](#)

Enter the battery percentage threshold where charging switches to topping up state.

**Battery topping up time** 3600[Restore default](#)

Enter the time in seconds for topping up the battery.

**Maximum charge cycle time** 18000[Restore default](#)

Enter the maximum time in seconds for charging before switching to topping up.

## Planner

Configure the maximum amount of time the robot should use to plan its route, the speed and acceleration it should drive with, the footprint and height of the robot, how far it may deviate from the planned path, and whether it should enable light indicators while driving.

Under advanced settings, you also have the option to modify how the robot should clear its obstacle history, charging station docking thresholds, and how the robot should drive when it has a hook.

### Planner

Settings regarding the planner for autonomous driving

[Go back](#)

---

<b>Robot height</b>	<input type="text" value="1.9"/> <span style="border: 1px solid #ccc; padding: 2px;">Restore default</span>
Defines the robot height (Default: 1.4, minimum: 0.5, maximum 1.8).	
<b>Robot footprint</b>	<input type="text" value="MiR100-200"/> <span style="border: 1px solid #ccc; padding: 2px;">Restore default</span>
Footprint of the robot. Increase this if something bigger than the robot is added on top. (Format: [[x,y]...]).	
<b>Max distance from path</b>	<input type="text" value="-1"/> <span style="border: 1px solid #ccc; padding: 2px;">Restore default</span>
Maximum distance in meters that the robot is allowed to deviate from the optimal path in the map.	
<b>Max distance from path with cart</b>	<input type="text" value="-1"/> <span style="border: 1px solid #ccc; padding: 2px;">Restore default</span>
Maximum distance in meters that the robot with a cart is allowed to deviate from the optimal path in the map.	
<b>Cart reverse speed</b>	<input type="text" value="0.12"/> <span style="border: 1px solid #ccc; padding: 2px;">Restore default</span>
Linear speed of robot when driving in reverse with a cart. The value is specified in meters per second.	
<b>Maximum planning time</b>	<input type="text" value="60"/> <span style="border: 1px solid #ccc; padding: 2px;">Restore default</span>
The maximum time allocated for planning a path. The value is specified in seconds.	
<b>Maximum allowed speed</b>	<input type="text" value="1.5"/> <span style="border: 1px solid #ccc; padding: 2px;">Restore default</span>
Overall maximum speed of the robot. This is the maximum speed the robot can move at, even if it is carrying a cart.	

## Localization

Define the thresholds for invalid localization adjustments and enable the use of odometry correction on slopes.

### Localization

Settings regarding localization of the robot

 Go back

#### Jump detection threshold (Linear movement)

0.2

 Restore default

Defines the linear movement threshold (in meters) for when a localization adjustment is considered an invalid jump. Enter a value below zero if you want to disable the function.

#### Jump detection threshold (Angular movement)

15

 Restore default

Defines the angular movement threshold (in degrees) for when a localization adjustment is considered an invalid jump. Enter a value below zero if you want to disable the function.

 Hide advanced settings

#### Enable odometry slope correction

False

  Restore default

Enables odometry correction on slopes. Note that the parameters 'Odometry slope correction inclination factor' and 'Odometry slope correction inclination threshold' need to be adjusted for the specific use case.

#### Odometry slope correction inclination factor

0.98

 Restore default

The correction factor for the odometry when a slope above 'Odometry slope correction inclination threshold' is encountered.

#### Odometry slope correction inclination threshold

3.5

 Restore default

The threshold in degrees for when to apply 'Odometry slope correction inclination factor' to the odometry.

## Serial interface

Set the baud rate, data bits, stop bits, and response delay of the serial interface. Register connected FTDI adapters for external interfaces.

### Serial Interface

Parameters for configuration of the serial interface

[Go back](#)**Serial port** [Restore default](#)

/dev/ttyUSB must be used when a serial adapter with a unique serial number is used. Otherwise the value may be changed to '/dev/ttyUSBx' for other USB-to-Serial adapters.

**Baud rate** [Restore default](#)

Baud rate of interface. Common values are 9600, 19200, 38400, 57600, 115200.

**Data bits** [Restore default](#)

Allowed options are: 7, 8.

**Stop bits** [Restore default](#)

Allowed options are: 0, 1, 2.

**Parity** [Restore default](#)

Allowed options are: 0 = none, 1 = odd, 2 = even

**Response delay** [Restore default](#)

Response delay for serial interface in microseconds. >10000us may be required for some devices

**External adapter serial number**[!\[\]\(3eb9fdadee47193a980c62ad0d74aa5a\_img.jpg\) Detect](#)

Serial number of FTDI adapter for external interface

## Motor controller

Set the gear ratio, ampere limit, and stall detection parameters of the motor controller. Update the motor encoder ticks for different hardware versions if the robot's database has been reset using the USB restore solution.



If your robot has a NUC7 robot computer, you can also use this page to detect the motor controller after USB restoring the robot.

### Motorcontroller

Settings regarding the motor controller

 Go back

 Hide advanced settings

#### Gear Ratio

12 : 1



Restore default

Gear ratio of robot

#### Ampere limit 10s average

12

Restore default

Ampere limit for 10 second average filter

#### Ampere limit 60s average

7.5

Restore default

Ampere limit for 60 second average filter

#### Stall ampere limit

15

Restore default

Ampere limit for stall detection

#### Stall timeout

7.0

Restore default

Timeout of stall detection in seconds

#### Motor encoder ticks

100 ticks/rev



Restore default

Set the correct number of encoder ticks per revolution of the motor. Select 128 ticks/rev if the robot has hardware version no. 1.3; MiR200 / 2.2; MiR100 or higher. Select 100 ticks/rev if the robot has a lower hardware version number. The setting is necessary only if the robot's database has been reset using the USB Restore solution.

#### Motor Controller Serial Number

 Detect

Serial number of FTDI adapter to motorcontroller. Note: This should only be changed if the robot uses a NUC7 computer.

## UR interface

Set the IP address of an attached Universal Robot arm.

### Docking

Disable undocking when the robot leaves a docking station, and set offsets and threshold for docking to charging stations, shelves and markers.

#### Docking

Parameters for docking to markers

 Go back

##### Undock from markers

True

▼

 Restore default

Select True to make the robot undock before starting move from docked position.

 Hide advanced settings

##### Relative move target when docking to markers.

0.5

 Restore default

Relative move target offset when docking to markers. The value is specified in meters.

##### Relative move target when docking to shelves.

1

 Restore default

Relative move target offset when docking to shelves. The value is specified in meters.

##### Distance to marker for disabling collision checks.

0.2

 Restore default

Distance to marker where collision detection is disabled.

##### Docked at marker depth threshold

0.2

 Restore default

Depth threshold for being docked to a marker

##### Docked at marker side threshold

0.4

 Restore default

Side threshold for being docked to a marker

##### Docked at marker angle threshold

20

 Restore default

Angular threshold for being docked to a marker

##### Docked at marker reverse distance

2

 Restore default

Distance to move backwards when being docked to a marker

## Relative move

Set the PID gains for the control system when performing a Relative move, or set the robot to use the method applied before software version 2.4.0.

### Relative Move

Configuration parameters for relative move

[Go back](#)[Hide advanced settings](#)**Angle to track: P-gain**

5

[Restore default](#)

Proportional gain for angle to track controller during relative move

**Angle to track: D-gain**

3

[Restore default](#)

Differential gain for angle to track controller during relative move

**Angle to track: I-gain**

0

[Restore default](#)

Integral gain for angle to track controller during relative move

**Use the relative move method from before software version 2.4.0**

False

[Restore default](#)

Select True to revert to the relative move method that was used before software version 2.4.0

## Ultrasound sensors

Enable the use of ultrasound sensors to detect objects right next to the robot at the start of any motion.

### Ultrasound sensors

Settings for the ultrasound sensors

[Go back](#)**Start at launch** False[Restore default](#)

Select **True** if the ultrasound sensors should start when the robot is turned on. If set to **False**, the additional sensor settings are ignored.

**Sensor configuration** Rear and side[Restore default](#)

Select the configuration of the ultrasound sensors. Select **Front and sides** on robots where the ultrasound sensors are placed in front and on the sides. Select **Rear and sides** on robots where the ultrasound sensors are placed at the back of the robot and on the sides.

**Turn on rear sensors** False[Restore default](#)

Select **True** to turn on the rear ultrasound sensors. This works only if **Start at launch** is set to **True**.

**Turn on side sensors** False[Restore default](#)

Select **True** to turn on the side ultrasound sensors. This works only if **Start at launch** is set to **True**.

## Calibration

Set global docking offsets that are applied every time the robot docks to markers, charging station, and shelves.

### Calibration

Calibration parameters for the individual robot



#### Global X offset for shelf docking



X offset for shelf docking. The value is in meters. A positive value will move the robot more in forwards direction. Example: 0.01 moves the robot 1 cm forwards.

#### Global Y offset for shelf docking



Y offset for shelf docking. The value is in meters. A positive value will move the robot more to the left. Example: 0.01 moves the robot 1 cm to the left.

#### Global offset orientation for shelf docking



Orientation offset for docking to shelves. The value is in degrees. A positive value will turn the robot counterclockwise.

#### Global X offset for V, L, and VL marker docking



X offset for V, L, and VL marker docking. The value is in meters. A positive value will move the robot more in forwards direction. Example: 0.01 moves the robot 1 cm forwards.

#### Global Y offset for V, L, and VL marker docking



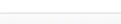
Y offset for V, L, and VL marker docking. The value is in meters. A positive value will move the robot more to the left. Example: 0.01 moves the robot 1 cm to the left.

#### Global orientation offset for V, L, and VL marker docking



Orientation offset for docking to V, VL and L markers. The value is in degrees. A positive value will turn the robot counterclockwise.

#### Global X offset for charging station docking



## Features

Enable the use of various external devices such as hooks, shelves, and I/O modules.

### Features

Enable and disable features in the robot

[Go back](#)

#### Hook

False	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
-------	----------------------------------	--

Select True if a hook is mounted on the robot.

#### Shelf

False	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
-------	----------------------------------	--

Select True if a shelf device is mounted on the robot.

#### I/O modules

True	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
------	----------------------------------	--

Select True to add actions for communicating with I/O modules from missions and zones

#### Email address

True	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
------	----------------------------------	--

Select True to add an action for sending emails from missions.

#### PLC registers

True	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
------	----------------------------------	--

Select True to add actions for setting PLC registers from missions and monitoring PLC register in the robot interface.

#### Universal Robots Interface

False	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
-------	----------------------------------	--

Select True to add an action for running UR-programs from missions.

#### Fleet

True	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
------	----------------------------------	--

Select True if the robot is part of a fleet

#### Modbus

True	<input type="button" value="▼"/>	<input type="button" value="Restore default"/>
------	----------------------------------	--

Select True if the robot uses Modbus

## Email configuration

Set up the email account of the robot, used when sending email prompts to users.

### Email configuration

Setup an email account for the robot enabling the robot to send emails

 Go back

#### Sender email

	Restore default
--	-----------------

The email address from which the emails from the robot will be sent

#### Sender name

	Restore default
--	-----------------

The sender name of emails sent from the robot

#### Username

	Restore default
--	-----------------

The username for logging into the email server

#### Password

	Restore default
--	-----------------

The password for logging into the email server

#### SMTP server

	Restore default
--	-----------------

The address of the SMTP server for sending emails

#### Output

0	Restore default
---	-----------------

The port for connecting to the SMTP server

#### Encryption type

None	▼	Restore default
------	---	-----------------

The encryption protocol for communicating with the SMTP server

#### Authentication required

True	▼	Restore default
------	---	-----------------

Active or deactivate authentication for the SMTP server

## WISE configuration

Set up your user account for connection to WISE modules.

WISE configuration

Set up your user account for connection to WISE modules.

[Go back](#)

[Hide advanced settings](#)

**Username**  
Root ▼ [Restore default](#)  
Enter the user name for connecting to WISE modules.

**Password**  
\*\*\*\*\* ▼ [Restore default](#)  
Enter the password for connecting to WISE modules.

[✓ Save changes](#) [✗ Cancel](#)

## Distributor data

Edit data about the distributor selling the robot.

### Distributor data

Edit data about the distributor selling the robot.

[Go back](#)**Name** Restore default

Enter the distributor's name...

**Address** Restore default

Enter the distributor's street address...

**City** Restore default

Enter the distributor's city...

**Zipcode** Restore default

Enter the distributor's zipcode...

**Country** Restore default

Enter the distributor's country...

**Phone number** Restore default

Enter the distributor's phone number...

**Support phone number** Restore default

Enter the distributor's support phone number...

**Email** Restore default

Enter the distributor's email address...

**Website** Restore default

Enter the distributor's website address...

## Safety system (MiR250, MiR500, and MiR1000 only)

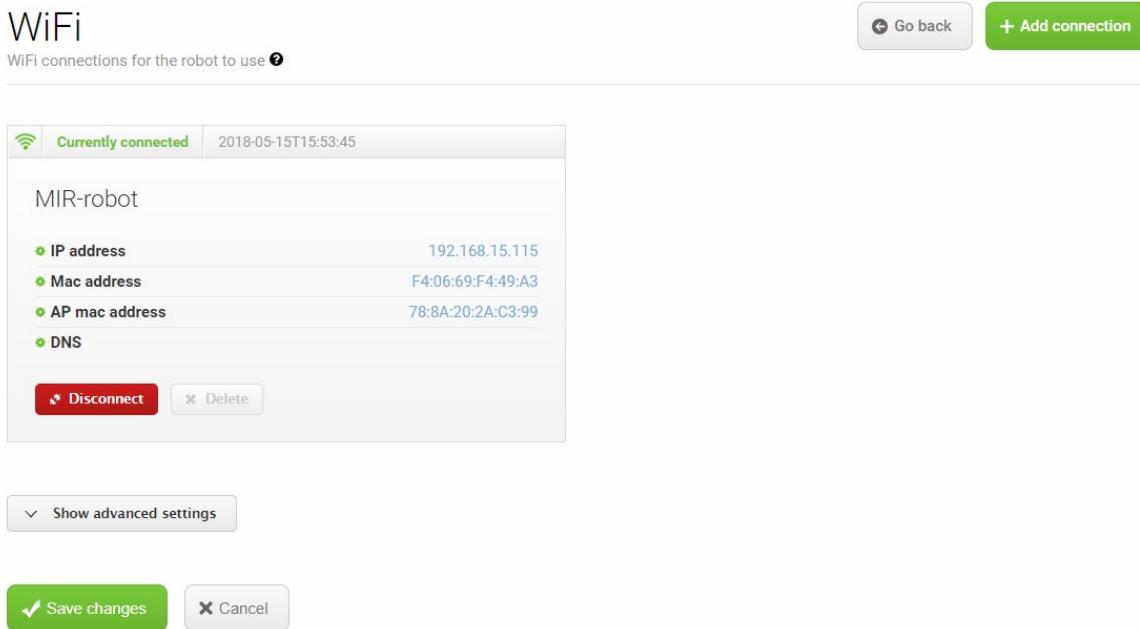
Configure the robot's personnel detection means system. You can modify which sounds the robot should play when its personnel detection means are muted and how loud the sounds should be.

## Advanced

The settings in this group are intended for technical supporters. In **Advanced** you can set the calibration of the front and back laser scanners, the distance between the robot's wheels, the diameter of towing wheels, the type of encoding used by the Modbus communication, and enable software lock.

## WiFi

In the WiFi section you can see current wireless networks on the robot, add new networks, and delete existing ones.



**WiFi**

WiFi connections for the robot to use ⓘ

Currently connected		2018-05-15T15:53:45
MIR-robot		
•	IP address	192.168.15.115
•	Mac address	F4:06:69:F4:49:A3
•	AP mac address	78:8A:20:2A:C3:99
•	DNS	
	Disconnect	

▼ Show advanced settings

## Add connection

You can set up new WiFi connections by selecting from a list of available wireless networks.

### Add connection

You can set up new WiFi connections by selecting from a list of available wireless networks.

**Select a network:**

**Security type:**  

**Password:**

**DNS servers**

Use static IP

The Add connection dialog contains the following fields:

- **Select a network**

Select the network you want to connect to from the list of available networks. If you cannot see the network you are looking for, try selecting **Reload**.

- **Security type**

Select a security protocol.

- **Password**

If required, enter a password to be used as log-in to the network.

- **DNS servers**

Enter DNS servers using the format: xxx.xxx.xxx.xxx. Use semicolon (;) as the delimiter.

- **Use static IP**

To use a static IP address, select this check box and fill in the following fields: IP address, Netmask, Gateway.

## Date & time

You can set the system date and time manually by entering values in the fields or automatically by selecting **Load from device**. The latter option sets the system time to the time of the computer connected to the robot.

The screenshot shows the 'Date & time' configuration page. At the top, there's a header 'Date & time' and a 'Go back' button. Below the header, it says 'Set date and time for the robot.' A question mark icon provides help. On the left, there are dropdown menus for 'Date' (September 27, 2017) and 'Time' (10:09:46). To the right, it displays 'Current robot date and time' as 'September 27 2017 10:09:55'. At the bottom, there are two buttons: a green 'Save changes' button with a checkmark icon, and a grey 'Load from device' button with a lightning bolt icon.

## Modified defaults

**Modified defaults** displays an overview of all values that have been modified from their default values.

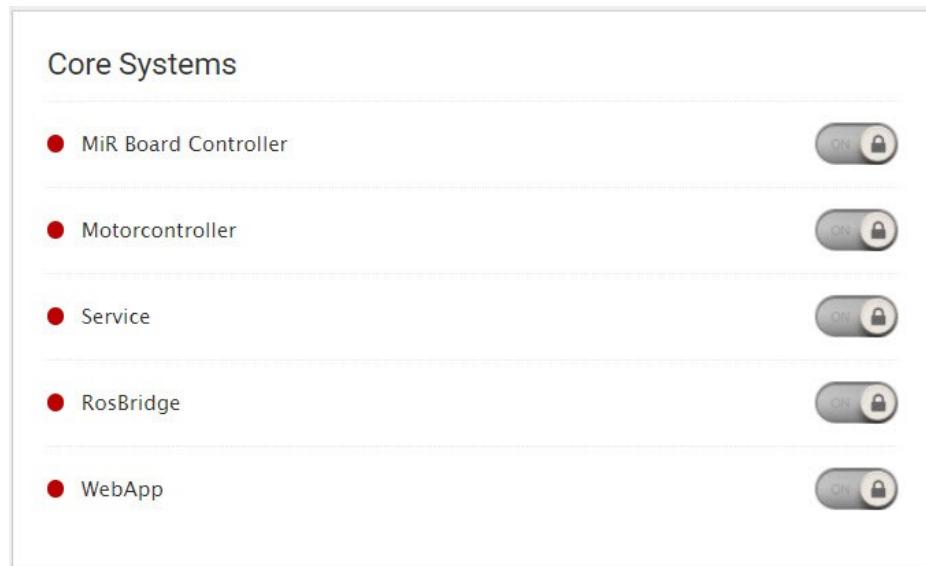
## 6.2 Processes

The **Processes** section displays the software modules that control the system processes on the robot. The modules are grouped according to functionality and, unless locked, each element in a group can be turned on and off individually.

As a rule, processes are turned on and off automatically as they apply. Therefore, the buttons should be used with care and only in the rare case that they are not turned on or off automatically or for test of a module, for example, the camera.

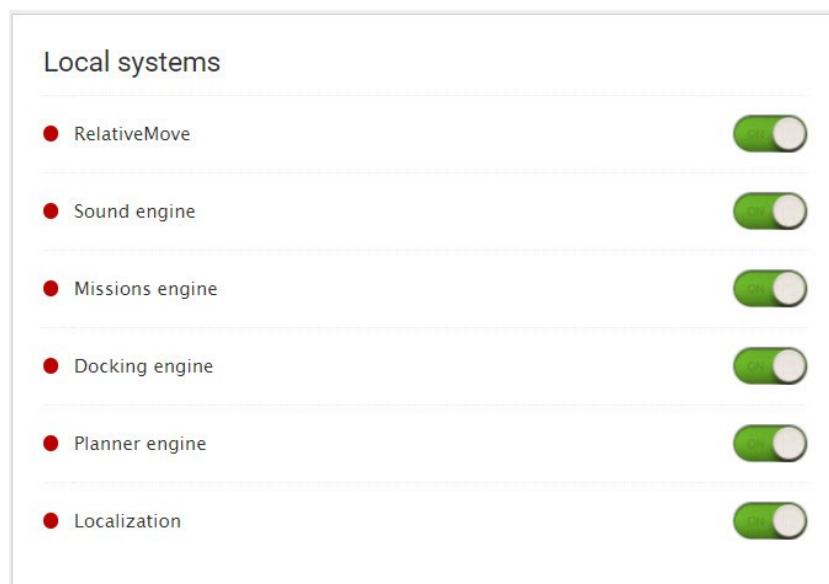
### Core systems

**Core systems** is a group of modules that control the core functions of the robot.



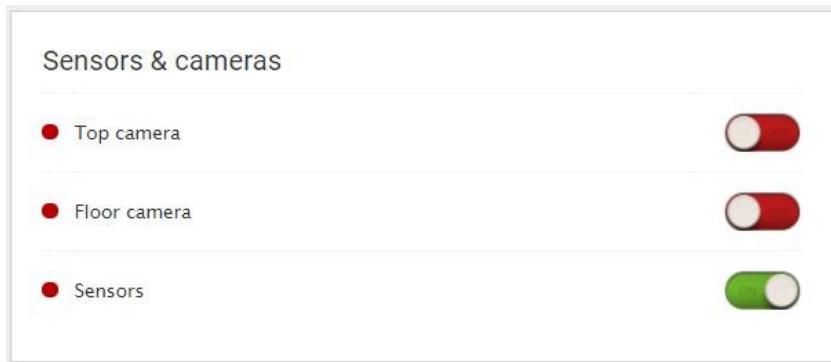
## Local systems

**Local systems** is a group of modules that the robot uses when in operation, for example, missions or local and global planners. **Localization** handles the AMCL navigational system and odometry. These two systems calculate the position of the robot on the map in which it is currently located.



## Sensors & cameras

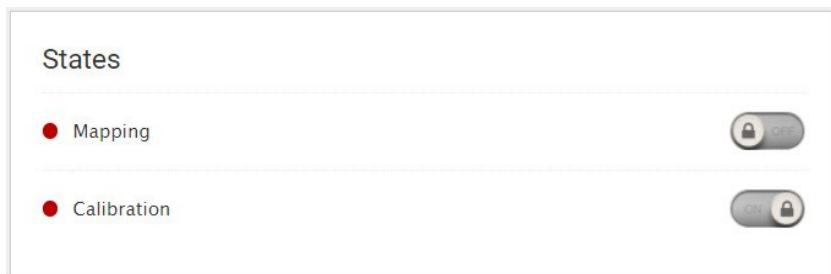
**Sensors & cameras** is a group of sensor modules. **Cameras** handle sensor data from the cameras. **Sensors** handles data from laser scanners



## States

**States** covers mapping and calibration modules.

- **Mapping** is turned on during mapping of a new site. It creates a map based on the manual driving of the robot, recording walls and obstacles in the mapped area.
- **Calibration** is turned on during IMU calibration of the robot's lasers and odometry.



## Options

**Options** is a group of modules that are turned on individually when an optional feature is applied, for example, a WISE module.



## 6.3 PLC registers

The PLC registers feature is only visible in the System menu if it has been enabled in **Features**.

PLC registers can be accessed through a serial interface using the robot's USB port (via RS232 adapter) or through a REST interface using the robot's Ethernet connection. Registers are used for handshake signals, for example, communication between a robot and a conveyor PLC.

In the PLC registers section, you can create and edit values for PLC-controlled devices. Registers are shared by missions in which PLC registers are used to either set a value or wait for a value.

You may change the default PLC register labels by selecting the edit icon next to the label and enter a text that describes what the particular register should be used for. For example, registers 8 and 9 could be labeled "Lower shelf" and "Raise shelf".



PLC registers can also be set through a dashboard widget.

### PLC registers

Read and set PLC registers on the robot. [?](#)

Integers (1 - 100)		
Registers 1-100 are 32 bit integers, i.e. whole positive or negative numbers.		
Registers 1-100 are used for example in shelf applications where registers 13 and 14 are used to lower and raise the shelf device respectively.		
If Modbus is enabled, the corresponding holding register addresses are displayed next to the PLC register. Notice! One PLC register uses two holding register addresses.		
Show registers from <input type="text" value="1"/> to <input type="text" value="20"/> Filter		
1	0	PLC register 1 <input checked="" type="checkbox"/>
2	0	PLC register 2 <input checked="" type="checkbox"/>
3	0	PLC register 3 <input checked="" type="checkbox"/>
4	0	PLC register 4 <input checked="" type="checkbox"/>
5	0	PLC register 5 <input checked="" type="checkbox"/>
6	0	PLC register 6 <input checked="" type="checkbox"/>
7	0	PLC register 7 <input checked="" type="checkbox"/>
8	0	PLC register 8 <input checked="" type="checkbox"/>
9	0	PLC register 9 <input checked="" type="checkbox"/>

Floats (101 - 200)		
Registers 101-200 are 64 bit floating point numbers, i.e. positive or negative decimal numbers.		
Registers 101-200 can be used where decimals are required for example to obtain precise information on distance travelled.		
If Modbus is enabled, the corresponding holding register addresses are displayed next to the PLC register. Notice! One PLC register uses two holding register addresses.		
Show registers from <input type="text" value="101"/> to <input type="text" value="120"/> Filter		
101	0	PLC register 101 <input checked="" type="checkbox"/> $\ast 42001 + 42002$ <input type="checkbox"/> <input type="checkbox"/>
102	0	PLC register 102 <input checked="" type="checkbox"/> $\ast 42003 + 42004$ <input type="checkbox"/> <input type="checkbox"/>
103	0	PLC register 103 <input checked="" type="checkbox"/> $\ast 42005 + 42006$ <input type="checkbox"/> <input type="checkbox"/>
104	0	PLC register 104 <input checked="" type="checkbox"/> $\ast 42007 + 42008$ <input type="checkbox"/> <input type="checkbox"/>
105	0	PLC register 105 <input checked="" type="checkbox"/> $\ast 42009 + 42010$ <input type="checkbox"/> <input type="checkbox"/>
106	0	PLC register 106 <input checked="" type="checkbox"/> $\ast 42011 + 42012$ <input type="checkbox"/> <input type="checkbox"/>
107	0	PLC register 107 <input checked="" type="checkbox"/> $\ast 42013 + 42014$ <input type="checkbox"/> <input type="checkbox"/>
108	0	PLC register 108 <input checked="" type="checkbox"/> $\ast 42015 + 42016$ <input type="checkbox"/> <input type="checkbox"/>
109	0	PLC register 109 <input checked="" type="checkbox"/> $\ast 42017 + 42018$ <input type="checkbox"/> <input type="checkbox"/>

The **PLC registers** section contains the following:

- **Integers**

Registers 1-100 are 32 bit integers, that is whole positive or negative numbers.

Registers 1-100 are used, for example, in shelf applications where registers 13 and 14 are used to lower and raise the shelf device respectively.

If Modbus is enabled, the corresponding holding register addresses are displayed next to the PLC register.

**NOTICE**

One PLC register uses two holding register addresses.

- **Floats**

Registers 101-200 are 64 bit floating point numbers, that is positive or negative decimal numbers.

Registers 101-200 can be used where decimals are required, for example, to obtain precise information on distance travelled.

If Modbus is enabled, the corresponding holding register addresses are displayed next to the PLC register.

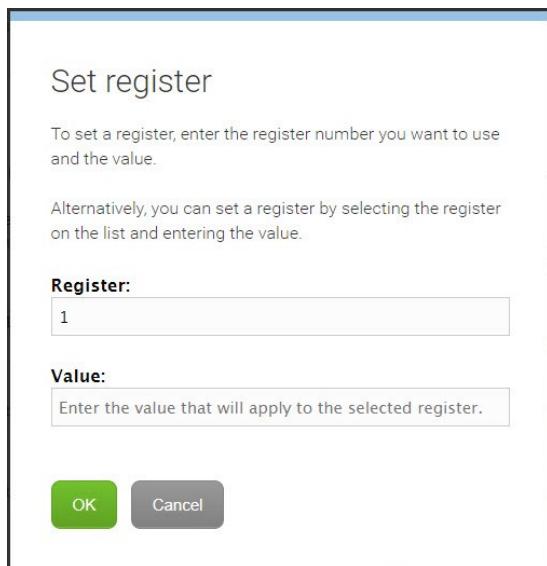
**NOTICE**

One PLC register uses two holding register addresses.

## Set register

To set a register, enter the register number you want to use and the value.

Alternatively, you can set a register by selecting the register on the list and entering the value.



## Delete register

You may delete a register by selecting **Delete**. Note that you delete only the register value and not the label.

## 6.4 Software versions

In the Software versions section, you can update the robot to run the newest software and see a list of all previous versions installed on the robot.

Select **Upload software**, and select the software file on your computer to start the upload. You can follow the update process on-screen. When finished, restart the robot, and sign in to the interface again. The robot is now ready to operate with the new software version.



If a hook is mounted on the robot, the hook must be updated to the same software version. Go to **Hook > Software versions** and follow the same procedure as for updating the robot. We recommend updating the hook before you update the robot.

### Software versions

Change the current software version.

**Upload software**

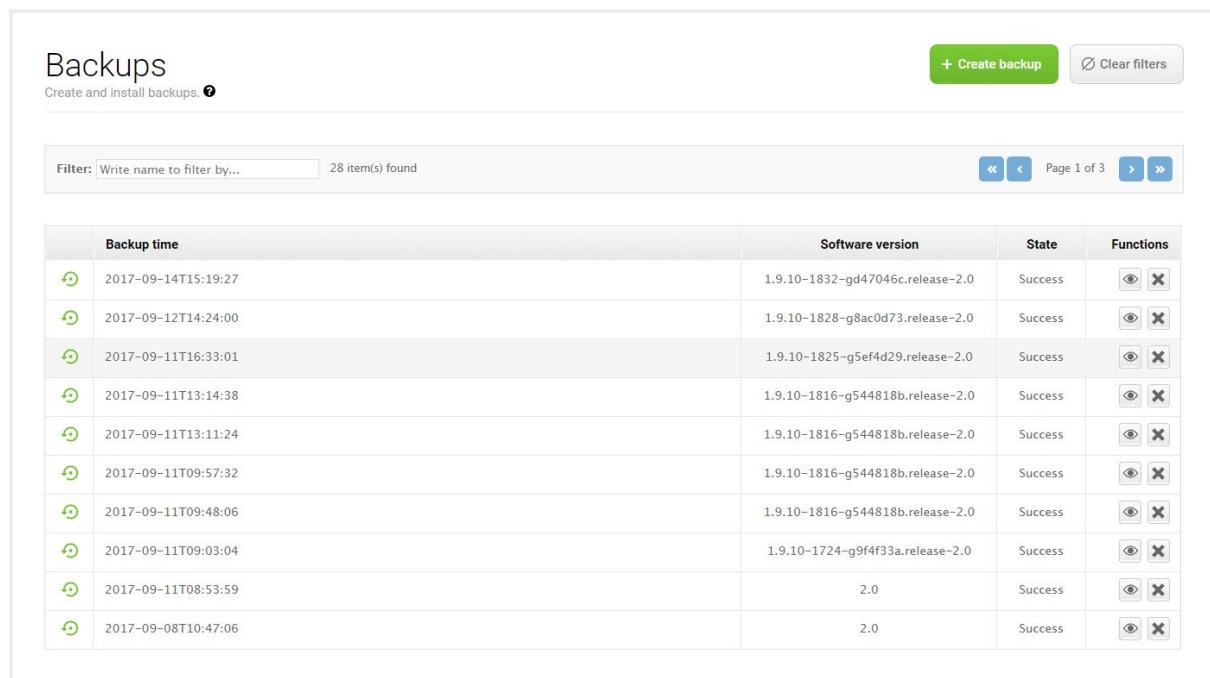
Version	Upgraded from	State	Started	Finished	Functions
1.9.12-1844-g5cb5113.feature-migration-circus-kh	1.9.10	Success	2017-09-15T10:39:45	2017-09-15T10:39:56	
1.9.12-1844-g5cb5113.feature-migration-circus-kh	1.9.10-1852-g24202f5.release-2.0	Failed	2017-09-15T10:38:44	2017-09-15T10:38:44	
1.9.12-1844-g5cb5113.feature-migration-circus-kh	1.9.10-1852-g24202f5.release-2.0	Failed	2017-09-15T10:34:36	2017-09-15T10:34:36	
1.9.10-1852-g24202f5.release-2.0	1.9.10-1832-gd47046c.release-2.0	Success	2017-09-14T15:19:27	2017-09-14T15:19:54	
1.9.10-1832-gd47046c.release-2.0	1.9.10-1828-g8ac0d73.release-2.0	Success	2017-09-12T14:23:59	2017-09-12T14:24:24	
1.9.10-1828-g8ac0d73.release-2.0	1.9.10-1825-g5ef4d29.release-2.0	Success	2017-09-11T16:33:01	2017-09-11T16:33:24	
1.9.10-1825-g5ef4d29.release-2.0	1.9.10-1816-g544818b.release-2.0	Success	2017-09-11T13:14:38	2017-09-11T13:15:01	
1.9.10-1816-g544818b.release-2.0	1.9.10-1724-g9f4f33a.release-2.0	Success	2017-09-11T09:03:04	2017-09-11T09:03:35	
1.9.10-1724-g9f4f33a.release-2.0	2.0	Success	2017-09-11T08:53:54	2017-09-11T08:54:23	
1.9.10-1812-gec11ad4.release-2.0	2.0	Failed	2017-09-08T16:57:31	2017-09-08T16:57:31	

## 6.5 Backups

In the Backups section, you can create a backup of the current system state and restore to a previous version of the software.

It can be useful to create a backup (snapshot) if you want to be able to revert to the exact state of the current software, including data such as settings, missions, and reports at a later stage.

Select **Backup** to create a backup of the current version.



	Backup time	Software version	State	Functions
⌚	2017-09-14T15:19:27	1.9.10-1832-gd47046c.release-2.0	Success	
⌚	2017-09-12T14:24:00	1.9.10-1828-g8ac0d73.release-2.0	Success	
⌚	2017-09-11T16:33:01	1.9.10-1825-g5ef4d29.release-2.0	Success	
⌚	2017-09-11T13:14:38	1.9.10-1816-g544818b.release-2.0	Success	
⌚	2017-09-11T13:11:24	1.9.10-1816-g544818b.release-2.0	Success	
⌚	2017-09-11T09:57:32	1.9.10-1816-g544818b.release-2.0	Success	
⌚	2017-09-11T09:48:06	1.9.10-1816-g544818b.release-2.0	Success	
⌚	2017-09-11T09:03:04	1.9.10-1724-g9f4f33a.release-2.0	Success	
⌚	2017-09-11T08:53:59	2.0	Success	
⌚	2017-09-08T10:47:06	2.0	Success	

## Delete backup

Select **Delete X** next to the backup you want to delete. In the next dialog box, select **Delete backup** to delete the backup.

## 6.6 Robot setup

This section contains the robot configuration and calibration properties.

### Configuration

You can edit the name of the robot in the **Name** field.

**Serial** shows the 9- or 15-digit serial number of the robot. The serial number is also found in the Help section under Robot information and on the product label on the robot.

**Security pack** shows the last time a new security pack was installed on your product.

Configuration	
Name	
Serial	
Security pack	2020-06-30

### Laser scanners (MiR100, MiR200, MiR500, and MiR1000 only)

In the Laser scanners group you find the serial numbers of the front and rear laser scanners and the functions for activating and swapping the two scanners.

The Detect button is used by the system to identify the two scanners. It should be used with great caution as it will make the robot move backwards when it should move forwards and vice versa.

The Swap button swaps the front and rear laser scanners. It should be used with great caution as it will make the robot move backwards when it should move forwards and vice versa.

### 3D cameras

The 3D cameras group shows the status of the cameras and their serial numbers.

## Serials

The Serials group lists the serial numbers of the robot's hardware components, for example NUC, BIOS, and SSD hard drive.

## Charging relay (MiR100 and MiR200 only)

The Charging relay buttons turn the robot's internal charging relay on or off. The charging relay is used when automatic charging is carried out by connecting the robot to a charging station.

## Inertial measurement unit

The IMU gain (Inertial Measurement Unit) is a calibration of the gyro's 360 degree rotation.

To calibrate the gyro, select **Start calibration** and make sure the robot has enough space to rotate around itself. The robot will start spinning on the spot while the progression of the calibration is shown in percentage. After a couple of minutes the calibration is finished and you get to decide if you want to keep the new calculated value.

If the value deviates significantly from the original one, it will show in red color and you can choose to discard the calibration and restore to the default value.



To perform the calibration, the robot must have an active map.

## Laser scanner calibration (MiR100 and MiR200 only)

The Laser scanner calibration is a calibration of the laser scanners to improve the robot's docking precision. To calibrate, place the robot approximately 2 m in front of a wall, and select the Calibrate front scanner or Calibrate rear scanner button. The robot now moves to a start position.

Measure the distance from the front of the laser scanners to the wall and enter the distance in the dialog box in robot the interface. Follow the instructions in the interface until the calibration is completed.

For detailed instructions on how to calibrate the scanners, see the how-to-article **Calibrate the laser scanners** in the How to section of the MiR website.

After a calibration, the robot must be restarted.

## Download SICK configuration for current robot software version

For MiR250, MiR500, and MiR1000 robots you can download MiR's supported SICK configurations for various robot applications. Select **Other SICK configurations** to see configurations for previous robot software versions.

## Encoder test

The encoder test runs the robot through a sequence where it moves forward at both a slow and a faster speed and then moves back again. While it is moving, stand behind the robot and observe whether the robot is drifting slightly to the left or right. You are prompted to enter your observations in the test dialog box.

When the test is finished, the results from the test are displayed in a table. The results include notes on what you observed and the number of recorded encoder ticks for each drive wheel. If the robot has a tendency to drift to either side, but the number of encoder ticks recorded for each drive wheel is almost the same, it indicates that there is an issue with the encoders. If the encoder ticks represent the robot's driving behavior accurately, there may be another issue with the robot. Contact your distributor for further assistance.

## 6.7 Triggers

The **Triggers** feature is visible in the System menu if **Modbus** is set to true under **System > Features**.

The robot can be set up to use Modbus TCP/IP communication. In the Triggers section you can set up links between robot missions and Modbus coil IDs, which will enable remote devices to add missions to the robot's mission queue.

Before you can establish a Modbus connection and create triggers, the Modbus feature must be activated under Features in the **System > Settings** section.

Triggers				
<a href="#">+ Create trigger</a> <span>(0)</span> <a href="#">Clear filters</a>				
Filter: <input type="text" value="Write name to filter by..."/> 1 item(s) found <span style="float: right;">Page 1 of 1</span>				
	Name	Assigned mission	Assigned coil ID	Created by
	Mission 1	Transport Goods	[ 1003 ]	Distributor
				<a href="#"></a> <a href="#"></a>

## Create trigger

To create a trigger, first enter a unique name, then enter a coil ID between 1001 and 2000, and finally assign the mission you want the robot to perform when the coil is activated.

### Create trigger

Create a new trigger. [?](#)

**Name !**

**Coil ID !**

**Mission !**

[✓ Save](#) [Go back](#)

The **Create trigger** dialog box contains the following fields

- **Name**  
Enter a name that describes the trigger.
- **Coil ID**  
Enter a coil ID that should be used for this trigger. The number must be an integer between 1001 and 2000.  
An integer is a whole positive or negative number.

- **Mission**

Select the mission that you want the robot to perform when the coil is activated.

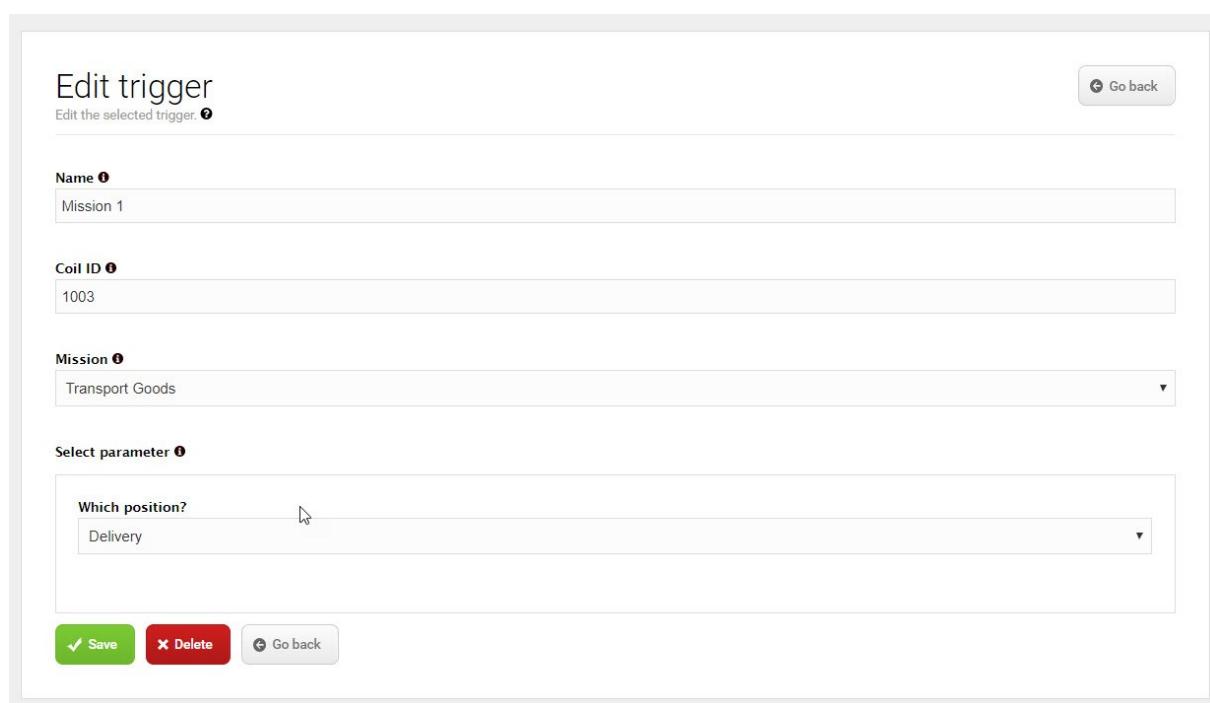
- **Select parameter (variables only)**

Select the parameter that should be used for this mission. Variable parameters are displayed if the selected mission was created with a variable parameter, for example, for positions.

Select **Save** to save the settings.

## Edit trigger

Edit details of the selected trigger. You can rename the trigger and change the coil ID and/or the mission that the robot will perform when the coil is activated.



The screenshot shows the 'Edit trigger' configuration page. At the top, it says 'Edit trigger' and 'Edit the selected trigger.' Below are input fields:

- Name**: Mission 1
- Coil ID**: 1003
- Mission**: Transport Goods
- Select parameter**: Which position? (Delivery)

At the bottom are three buttons: a green **Save** button, a red **Delete** button, and a **Go back** button.

## Delete trigger

You can delete triggers created by you or another member of the user group you belong to.

## Delete trigger

Delete the selected trigger. [?](#)

» title	Mission 1
» description	Transport Goods
» permissions	R/W
» data_type	boolean
» registers	[ 1003 ]
» Mission parameters	
» Which position?	Delivery

**X Delete** [Go back](#)

# 7. Help

This section describes the items in the Help menu.



The Help menu contains the following items:

<b>7.1 Robot information .....</b>	<b>158</b>
<b>7.2 API documentation .....</b>	<b>159</b>
<b>7.3 Remote access .....</b>	<b>160</b>
<b>7.4 Service book .....</b>	<b>161</b>
<b>7.5 Manual .....</b>	<b>162</b>

## 7.1 Robot information

This menu item contains the following information:

- **Robot name**  
This field shows the robot name.
- **Robot serial**  
This field shows the robot serial number.
- **Robot software version**  
This field shows the robot software version.

If you have a MiR hook mounted to the robot, the following information is also displayed:

- **Hook name**  
This field shows the hook name.
- **Hook serial**  
This field shows the hook serial number.

- **Hook software version**

This field shows the hook software version.

## Robot information

General information about the robot.

### Robot name

Johnny 5

### Robot serial

Robot serial no. not set.

### Robot software version

2.2.3-292-g8ab5ebe.release-2.3.0

## Hook information

### Hook name

I0082

### Hook serial

18050011100082

### Hook software version

2.2.3-292-g8ab5ebe.release-2.3.0

## 7.2 API documentation

All functionality found in the robot interface can also be accessed through the robot's REST API. The REST API is what the robot interface uses to communicate with the robot.

You can connect to the robot using either <https://mir.com:443/api> or <https://mir.com/api>. Alternatively, you can use the robot's IP address instead of [mir.com](#) if you are not connected to the robot's own WiFi.

For authorization, please refer to the example on this page, automatically generated when you enter your username and password.

If you want to try to use REST API, you can enter your username and password and then select [Launch API documentation](#). Each endpoint you can use is documented, and you can select [Try it out](#) to send an API request using the entered authorization.



If you are using a software version below 2.10.0, the format is  
<http://mir.com:8080> or <http://mir.com/api>

## API documentation

Get started with the REST API for the robot.

All functionality found in the robot interface can also be accessed through the robot's REST API. In fact, the REST API is what the robot interface uses to communicate with the robot - and so can your software.

You can connect to the robot using either <http://mir.com:8080> or <http://mir.com/api>. Alternatively you can use the robot's IP address if you are not connected to the robot's own WiFi.

For authorization, please refer to the given example, automatically generated when you enter your username and password.

**Username**   
**Password**   
**Language** ▼

```
GET /status HTTP/1.1
Content-Type: application/json
Accept-Language: en_US
Host: mir.com:8080
Authorization: Basic OmUzYjBjNDQyOThmYzFjMTQ5YWZizjRjODk5NmZlOTI0MjdhZTQxZTQ2NDliOTM0Y2E0OTU5OTFiNzg1MmI4NTU=
```

## 7.3 Remote access

MiR Remote enables the MiR Technical Support team to remote access the robot's software. In many cases, this helps to solve a software problem quickly.

You have command of the remote session, which means that you can retrieve access at any time by selecting the Disconnect button.

During the remote access session, you can continue using the robot if the problem you need solved allows it.



Port 8888 must be open on your network for MiR Remote to work.

## Remote access

Allow remote access to this robot.

MiR Remote makes it possible to give members of the MiR Support team remote access to the robot's software. This will in many cases help solving a software problem quickly and will save you, as customer, time on complex problem descriptions.

You are in charge of the amount of time MiR Support has access to the system, and you can withdraw the access at any time during the session or prolong it as needed.

During the remote access session, you can continue using the robot if the problem you need solved allows it.

**Connect**  
to MiR Remote™

 [Connect to MiR Remote™](#)

Give MiR Support access to the robot's software.

Time	Message
2017-11-09T10:51:49.179886	The connection was closed..
2017-11-09T10:51:49.078927	Disconnecting due to timeout..
2017-11-09T10:48:09.405708	* Successfully logged in..
2017-11-09T10:48:09.269405	* Connection successfully established on port '45370'
2017-11-09T10:48:08.502521	* Successfully retrieved port '45370'..

## 7.4 Service book

In the Service book you can enter notes about the robot, for example, about changes made in the robot. The notes can be read by all user groups and cannot be deleted.

### Service book

Read and create service notes. 

Filter: Write name to filter by... Group: Show all ▾ 3 item(s) found Page 1 of 1    

Owner group	Created	Created by	Note	Functions
 Distributor	2018-01-17 15:53:20	Distributor	The robot was updated to SW release 2.0.15 on February 9, 2018	
 Distributor	2018-01-17 16:07:54	Distributor	A new Bluetooth mission was implemented on February 6.	
 Distributor	2018-01-17 16:08:45	Distributor	Three dashboards were created for the operators on production line 4.	

Add a service note

|

 [Submit note](#)

## 7.5 Manual

A copy of this reference guide is available in the interface. To access the guide, go to: [Help > Manual](#).

# 8. Hook

MiR100 Hook and MiR200 Hook only

The Hook menu is visible only if you enable it in **System > Settings > Features**.

This section describes the items in the Hook menu.



The Hook menu contains the following items:

<b>8.1 Controlling MiR Hook .....</b>	<b>164</b>
<b>8.2 Carts .....</b>	<b>167</b>
<b>8.3 Settings .....</b>	<b>173</b>
<b>8.4 Setup .....</b>	<b>174</b>

## 8.1 Controlling MiR Hook

Preparing a MiR hooks to pick up and deliver carts accurately involves a number of setup and test procedures some of which are found in the **Hook** section others in different sections of the user interface. In brief, the steps are as follows:

1. Set **Hook to True** under **System > Settings > Features**. This will open the **Hook** section in the main menu bar of the user interface.
2. In the **Hook** menu, go to **Manual control** and perform a homing.
3. Create minimum one cart type and one cart in the **Carts** section.
4. Calibrate the relation between robot and cart in the **Carts** section.
5. Create a mission to test and confirm that the setup was done correctly.

### Manual control

The manual control section lets you perform the following actions:

- Open and close the gripper.
- Activate and deactivate the hook brake.
- Perform homing (maximum height calibration).
- Change the height of the hook manually.

**Hook state** provides information on the position of the hook and the status of the gripper and the brake. The **Live marker** section shows the ID code readout.

## Manual control

Manual control of hook functionality 

Control MiRHook™	» Open	» Close	» Activate brake	» Deactivate brake	» Home	» Change height	
------------------	--------	---------	------------------	--------------------	--------	-----------------	--

**Controlling your MiRHook™**

Preparation of the MiRHook to pick up and deliver carts accurately involves a number of setup and test procedures some of which are found in the Hook section others in different sections of the user interface. In brief the steps are as follows:

1. Activate the Hook under External features in the System settings section. This will open the Hook section in the main menu bar of the user interface.
2. Perform a homing (zero-point calibration) of the hook in the Manual control section.
3. Create minimum one cart type and one cart in the Carts section.
4. Calibrate the relation between robot and cart in the Carts section.
5. Create a mission to test and confirm that the setup was done correctly.

Hook state	
Status	Closed (Trolley attached)
Height above floor	185 mm
Hook angle	0.19 degrees
Brake status	Inactive

Live marker	
Marker name	cart01
X-value	0.02
Y-value	-0.00
Z-value	0.18



For more detailed instructions on setting up a MiR hook, see the MiR100 Hook or MiR200 Hook user guide. For instructions on mounting a hook mechanically, see the MiR Hook 100 or MiR Hook 200 Mounting guide.

## Open / Close

Use these buttons to open or close the hook gripper.

## Activate / Deactivate brake

Activate or deactivate the hook brake. The hook brake locks the hook arm and prevents it from swinging horizontally.

Activating the hook brake makes it easier to perform the cart calibration since even small hook movements may affect the calibration.

**NOTICE**

Do not drive the robot in manual mode with the brake activated as this may damage the motors or the brake.

Deactivate the brake if you need to move the hook arm manually.

## Home

Use the Home function to raise the hook to its highest position. The homing function is used to calibrate the hook height encoder. The robot raises the hook arm to its maximum height and resets the encoder.

## Change height

Use the Change height function to lower or raise the hook to a set position. The function should be used to find the entry, lock, and drive heights for a specific cart. Note down the heights and use them when you create a new cart in the system.

## Hook state

This group shows the following information:

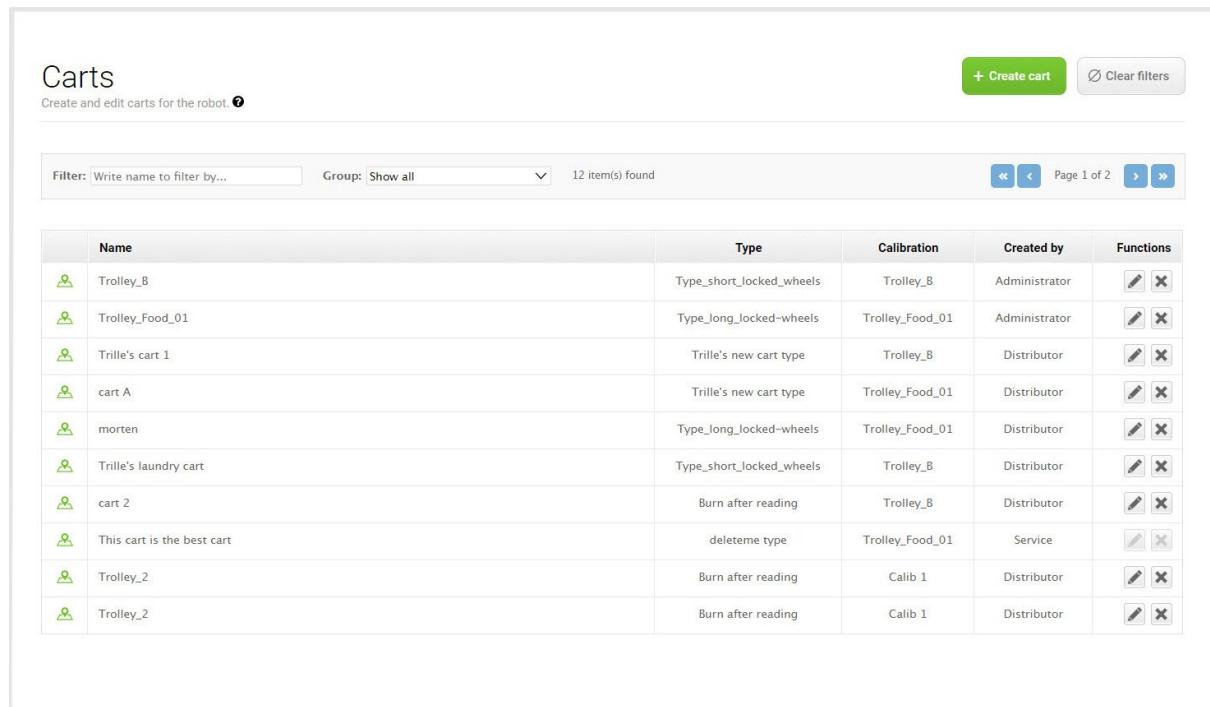
- **Status:** The status of the hook gripper.
- **Height above floor:** The height of the gripper above the floor.
- **Hook angle:** The angle of the hook. 0 degrees is the angle at which the hook arm is parallel to the robot. Use this readout to align the hook arm.
- **Brake status:** The status of the hook brake.

## Live marker

This group shows the information that the hook camera reads from the cart's ID tag.

## 8.2 Carts

All carts that are to be used with the MiR hooks must be set up in the robot interface.



Name	Type	Calibration	Created by	Functions
Trolley_B	Type_short_locked_wheels	Trolley_B	Administrator	
Trolley_Food_01	Type_long_locked-wheels	Trolley_Food_01	Administrator	
Trille's cart 1	Trille's new cart type	Trolley_B	Distributor	
cart A	Trille's new cart type	Trolley_Food_01	Distributor	
morten	Type_long_locked-wheels	Trolley_Food_01	Distributor	
Trille's laundry cart	Type_short_locked_wheels	Trolley_B	Distributor	
cart 2	Burn after reading	Trolley_B	Distributor	
This cart is the best cart	deleteme type	Trolley_Food_01	Service	
Trolley_2	Burn after reading	Calib 1	Distributor	
Trolley_2	Burn after reading	Calib 1	Distributor	

The setup includes carts, cart types, and calibration of the relation between cart and MiR Hook:

- Cart types are common types sharing the same length, width, and height.
- Carts are individual carts based on a cart type and specifications of entry, grip, and drive heights plus the ID of the QR code or AprilTag attached to the cart.
- Calibrations define the entry, lock, and drive heights of the hook when it picks up a particular cart.

### Create cart

Each cart to be used with MiR100 Hook or MiR200 Hook must be set up individually. The name must be the exact QR code or AprilTag name and all carts must be attached to a cart type specifying length, width, and height.

## Create cart

Create a new cart for the robot. [?](#)

[Back to the list](#)

Name (QR ID) <a href="#">?</a>	cart02	Detect
Type <a href="#">?</a>	Type_long_locked-wheels	Create / Edit
Calibration <a href="#">?</a>	Long cart calibration	Create / Edit
<a href="#">✓ Create cart</a> <a href="#">Cancel</a>		

The **Create cart** dialog contains the following fields:

- **Type**

Select the type of cart that fits the cart you are about to create, or select **Create/Edit** to create a new type.

The cart type must match the length, width, and height of the cart you are creating.

For more information, see [Type below](#).

- **Calibration**

Select an existing calibration or select **Create/Edit** to create a new one.

The calibrations list shows details of the calibrations that are already available in the system. You may edit or delete existing calibrations or create new ones.

A calibration consists of a name, the exact position of the robot in relation to the cart when picking up, and the entry, lock, and drive heights of the hook.

For more information, see [Calibration on page 170](#).

Select **Create cart** to save the settings.

## Type

All carts must be attached to a named cart type, specifying length, width, height, and locked wheels offset of the cart.

Select **Create/Edit** to open the Cart types window.

The cart types list shows details of the cart types that are already available in the system. You may edit or delete existing cart types or create new ones.

## Cart types

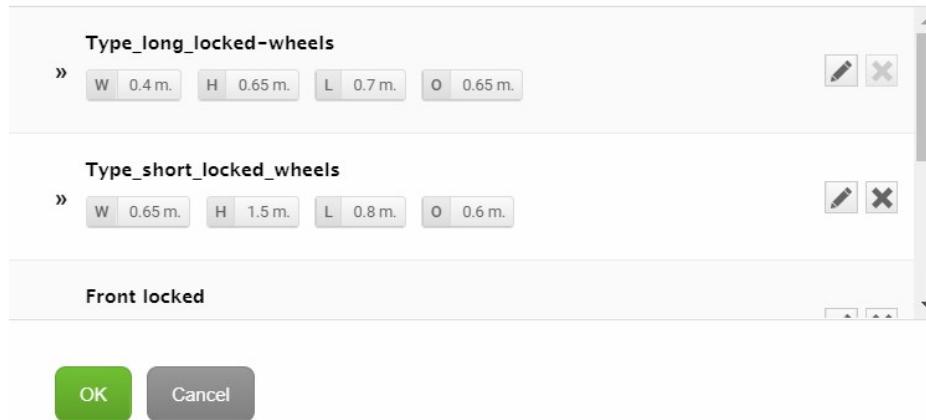
[Create type](#)

The cart types list shows details of the cart types that are already available in the system. You may edit or delete existing cart types or create new ones.

Type	W	H	L	O
Type_long_locked-wheels	0.4 m.	0.65 m.	0.7 m.	0.65 m.
Type_short_locked_wheels	0.65 m.	1.5 m.	0.8 m.	0.6 m.
Front locked				

[Edit](#) [Delete](#)

**OK** **Cancel**



## Create cart type

The **Create cart type** dialog contains the following fields:

- **Name**

Enter a name for the cart type you are about to create.

The name must be unique and is used to identify a group of carts.

One way of naming cart types is to select names that characterize the tasks of the carts in the particular group. For example, tall cage-type carts used for laundry transport could be named Laundry carts.

- **Width in meters**

Enter the width of the cart type in meters.

Measure from side to side of the cart.

- **Height in meters**

Enter the height of the cart type in meters.

Measure from floor to the highest point of the cart.

- **Length in meters**

Enter the length of the cart type in meters.

Measure the length from the rear end to the gripping bar.

- **Locked wheels offset in meters**

Enter the offset in meters.

Measure the distance from the line going through the centers of the locked wheels to the front of the gripping bar.



The locked wheels can be either at the front or the back of the cart. For information on towing different types of carts, see the user guide for your MiR hook.

## Calibration

The calibration specifies the entry, lock, and drive heights that the MiR hook should use to pick up a specific cart. As part of the calibration, the hook camera detects the ID tag's position on the cart. This information is used by MiR hook to position itself precisely every time it picks up a cart. So once calibrated, the ID tag should not be moved on the cart, and if it happens, the calibration must be redone.

Calibrations can be reused by multiple carts. This simply requires that the ID tags are placed in the exact same positions on the carts sharing the calibration and that the carts are of the same type.

**Calibrations**

**Create calibration**

The calibrations list shows details of the calibrations that are already available in the system.  
You may edit or delete existing calibrations or create new ones.

DFD - Trolley_A - 1
X 0.010 Y -0.012 Z 0.171 Entry height 150 mm.
Lock height 190 mm. Drive height 190 mm.

Cart calibration
X 0.019 Y -0.020 Z 0.171 Entry height 160 mm.
Lock height 185 mm. Drive height 185 mm.

**OK**   **Cancel**

## Create calibration

A calibration consists of a name, the exact position of the robot in relation to the cart when picking up, and the entry, lock, and drive heights of the hook.

### Create calibration

A calibration consists of a name, the exact position of MiR Hook 100 or MiR Hook 200 in relation to the cart when picking up, and the entry, lock, and drive heights of the hook.

<b>Name:</b>	<input type="text" value="Cart 1"/>		
<b>Depth offset</b>	<input type="text" value="0.2"/>	<b>Sideways offset</b>	<input type="text" value="0"/>
<b>Entry height in mm.:</b>	<input type="text" value="390"/>	<b>Lock height in mm.:</b>	<input type="text" value="400"/>
		<b>Drive height in mm.:</b>	<input type="text" value="400"/>

**OK**   **Detect**   **Set height**   **Cancel**

The **Create calibration** dialog contains the following fields:

- **Name**

Enter a name for the calibration you are about to create.

The name must be unique and is used to identify a specific calibration that can be used by the same cart types and with the ID tag attached in the exact same position.

- **Depth and Sideways offset**

The depth and sideways offsets describe the location of the 3D hook camera in relation to the ID tag.

Select **Detect** to live-update the positions every time you move the hook. The camera should point as closely to the middle of the ID tag as possible and have a distance of approximately 160 mm to the marker. This means that the offsets should be as close to 0 as possible.

- **Entry height in mm**

The hook's current height is automatically shown in the **Entry height in mm** field. To change the height, select **Set height** and enter a new height in mm.

Entry height is the height the hook will use to go under a cart before gripping it. The height should be measured as the distance between the floor and the bottom plate of the hook. It will typically be set to about 50 mm lower than the lock height.

- **Lock height in mm**

Set the height the hook should raise to, to grip and lock itself to the cart.

The lock height should be set so that the lower gripper hook is right under the gripping bar of the cart without lifting the wheels.

If you are unsure of the lock height, you may go to the **Manual control** section to find the correct height by using the **Change height**, **Close** and **Open** functions.

- **Drive height in mm**

The drive height can, in almost all cases, be set to the same value as the lock height.

Select **OK** to save the settings.

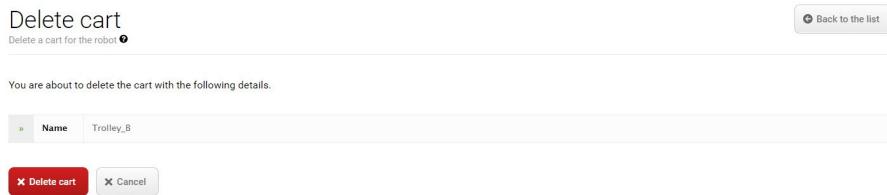
## Edit cart

If you change the name of the cart, make sure the name corresponds to a specific ID marker. Alternatively, use **Detect** to auto-fill the **Name** field using the camera live view.

## Delete cart

You can delete carts that are created by you or another member of the user group you belong to.

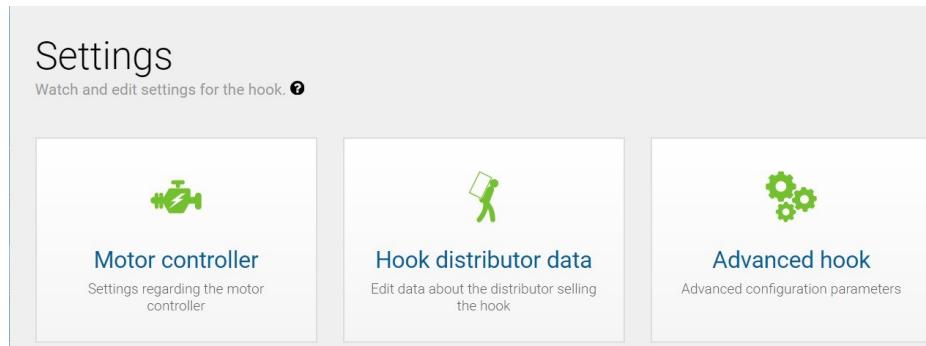
Select **Delete X** next to the cart you want to delete. In the next dialog, select **X Delete cart** to delete the cart.



The screenshot shows a 'Delete cart' dialog box. At the top, it says 'Delete cart' and 'Delete a cart for the robot'. A link 'Back to the list' is at the top right. Below that, a message says 'You are about to delete the cart with the following details.' A table shows a single row with 'Name' set to 'Trolley\_B'. At the bottom are two buttons: a red 'X Delete cart' button and a grey 'Cancel' button.

## 8.3 Settings

All settings of the robot are found and may be edited under **Hook > Settings**. The Robot setup section is divided into sub groups.



The screenshot shows the 'Settings' interface. It has a title 'Settings' and a subtitle 'Watch and edit settings for the hook.' Below are three cards:

- Motor controller**: Icon of a valve, description 'Settings regarding the motor controller'.
- Hook distributor data**: Icon of a hook, description 'Edit data about the distributor selling the hook'.
- Advanced hook**: Icon of gears, description 'Advanced configuration parameters'.

### Motor controller

The settings in this group are intended for technical supporters.

This group contains settings related to the motors in MiR Hook. The motors control the following parts:

- The hook brake
- The hook gripper
- The hook height actuator

### Hook distributor data

Edit data about the distributor selling the hook.

## Advanced hook

The settings in this group are intended for service technicians.

This group contains calibration parameters and limits.

## 8.4 Setup

In the **Setup** section you can change the name of the hook, and find the serial number and its integrated components. Also, the hook encoder can be zero calibrated, for example, if the encoder has been replaced.

### Configuration

In the Configuration group you may change the name of the hook and read the serial number.

#### Serial

Shows the 9- or 15-digit serial number of the hook. The serial number is also found in the Help section under **Robot information** and on the identification label on the MiR hook.

#### Motor controllers

The motor controllers group lists the serial numbers of the motor controllers for actuator and brake.

The magnifier icon is used to automatically find and enter the number of a motor controller, for example, if it has been replaced.

If both serial numbers must be re-entered, first disconnect one of the controllers and select the magnifier icon on the one that is not disconnected, then reconnect the controller and select the magnifier icon on the second controller.



Be sure to select the correct motor controller when you select the magnifier icon, as the system will not automatically detect if they are swapped.

#### Serials

The Serials group lists the serial numbers of hook hardware components, for example, NUC, BIOS, and SSD hard drive.

### Hook encoder offset

The hook encoder group shows the horizontal angle of the hook arm and makes it possible to make a new zero calibration of the hook by selecting the Set 0-value button.



The hook encoder has already been zero-set from the factory, and it should only be calibrated again if the encoder has been replaced.

### Hook angle

Hook angle shows a live view of the hook's horizontal position.

### Offset

Offset shows the hook angle offset value. Default is 0.

# 9. Modbus register reference

The tables in this chapter list the possibilities for the robot to communicate with external PLC devices through Modbus TCP/IP.

## 9.1 Status messages

Title	Description	Permission	Data type	Registers
Software version	Robot software version	R	int16array	[4001, 4002, 4003]
Mode	The current mode of the robot	R	int16	[4004]
State	The current state of the robot	R	int16	[4005]
Error code	The last error registered on the robot. 0 if no errors were detected.	R	int16	[4006]
Battery level	Remaining charge [%]	R	int16	[4008]
Uptime	The robot's uptime	R	int32	[4009, 40010]
Distance run	The distance run by the robot from the beginning of time.	R	float32	[40011, 40012]
Position X	Position X in global coordinates	R	float32	[40013, 40014]
Position Y	Position Y in global coordinates	R	float32	[40015, 40016]

Title	Description	Permission	Data type	Registers
Position Orientation	Orientation of the robot in global coordinates [degrees]	R	float32	[40017, 40018]
Length of mission queue	Number of missions pending or executing	R	int16	[40019]

## 9.2 PLC triggers

Title	Description	Permission	Data type	Registers
PLC integer	Value of PLC register	R/W	int32	[41001,41002]...[41199, 41200]
PLC float	Value of PLC register	R/W	float32	[42001,42002]...[42199, 42200]

## 9.3 Mission triggers

Title	Description	Permission	Data type	Coils
Trigger name	Description of trigger	R/W	Boolean	[1001]...[2000]

## 9.4 Action commands

Title	Description	Permission	Data type	Coils
Continue robot	Address of the coil used to trigger the Continue action on the robot	R/W	Boolean	[1]
Pause robot	Address of the coil used to trigger the Pause action on	R/W	Boolean	[2]

Title	Description	Permission	Data type	Coils
	the robot			
Cancel current mission	Address of the coil used to cancel the ongoing mission, if any	R/W	Boolean	[3]
Clear mission queue	Address of the coil used to clear the entire mission queue	R/W	Boolean	[4]
Clear error	Address of the coil used to clear the errors on the robot	R/W	Boolean	[5]
Continue robot	Address of the coil used to trigger the Continue action on the robot	R/W	Boolean	[6]