

# User Guide (en)

Date: 08/2021

Revision: v.3.2

# MiR 1000



## 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 © 2016-2021 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](mailto:support@mir-robots.com)

CVR: 35251235

# Table of contents

---

<b>1. About this document</b> .....	<b>7</b>
1.1 Where to find more information .....	7
1.2 Version history .....	8
<b>2. Product presentation</b> .....	<b>10</b>
2.1 Main features of MiR100 .....	11
2.2 External parts .....	12
2.3 Internal parts .....	18
<b>3. Safety</b> .....	<b>21</b>
3.1 Safety message types .....	21
3.2 General safety precautions .....	22
3.3 Intended use .....	26
3.4 Users .....	27
3.5 Foreseeable misuse .....	28
3.6 Warning label .....	29
3.7 Residual risks .....	29
<b>4. Getting started</b> .....	<b>31</b>
4.1 In the box .....	31
4.2 Unpacking MiR100 .....	32
4.3 Connecting the battery .....	34
4.4 Powering up the robot .....	38
4.5 Connecting to the robot interface .....	39
4.6 Connecting the robot to a WiFi network .....	42
4.7 Driving the robot in Manual mode .....	45

---

4.8 Checking the hardware status .....	47
4.9 Mounting the nameplate .....	47
4.10 Shutting down the robot .....	49
<b>5. Battery and charging .....</b>	<b>50</b>
5.1 Charging the robot .....	50
5.2 Disconnecting the battery .....	52
5.3 Battery storage .....	56
5.4 Battery disposal .....	56
<b>6. IT security .....</b>	<b>57</b>
6.1 Managing users and passwords .....	57
6.2 Software security patches .....	57
<b>7. Navigation and guidance system .....</b>	<b>59</b>
7.1 System overview .....	59
7.2 User input .....	61
7.3 Global planner .....	61
7.4 Local planner .....	63
7.5 Obstacle detection .....	64
7.6 Localization .....	71
7.7 Motor controller and motors .....	73
7.8 Brakes .....	73
<b>8. Safety-related functions and interfaces .....</b>	<b>75</b>
8.1 System overview .....	75
8.2 Personnel detection .....	79
8.3 Overspeed avoidance .....	84

---

8.4 Stability .....	84
8.5 Emergency stop circuit .....	84
8.6 Robot computer .....	86
8.7 Light indicators and speakers .....	86
<b>9. Commissioning .....</b>	<b>89</b>
9.1 Analysis of the work environment .....	89
9.2 Risk assessment .....	91
9.3 Creating and configuring maps .....	92
9.4 Markers .....	103
9.5 Positions .....	110
9.6 Creating missions .....	111
9.7 Creating a footprint .....	115
9.8 Making a brake test .....	119
9.9 Creating user groups and users .....	120
9.10 Creating dashboards .....	123
9.11 Updating MiR100 software .....	125
9.12 Creating backups .....	125
9.13 System settings .....	126
<b>10. Usage .....</b>	<b>134</b>
10.1 Creating markers .....	134
10.2 Creating positions .....	139
10.3 Creating the mission Prompt user .....	142
10.4 Creating the mission Try/Catch .....	146
10.5 Creating the mission Variable footprint .....	152

---

10.6 Testing a mission .....	155
<b>11. Applications .....</b>	<b>157</b>
11.1 Mounting a top module .....	157
<b>12. Maintenance .....</b>	<b>159</b>
12.1 Regular weekly checks and maintenance tasks .....	159
12.2 Regular checks and replacements .....	161
12.3 Battery maintenance .....	164
<b>13. Packing for transportation .....</b>	<b>165</b>
13.1 Original packaging .....	165
13.2 Packing the robot for transportation .....	166
13.3 Battery .....	166
<b>14. Disposal of robot .....</b>	<b>167</b>
<b>15. Payload distribution .....</b>	<b>168</b>
<b>15. Interface specifications .....</b>	<b>173</b>
15.1 Application interface .....	173
15.2 Emergency stop .....	174
<b>16. Error handling .....</b>	<b>176</b>
16.1 Software errors .....	176
16.2 Hardware errors .....	178
<b>17. Glossary .....</b>	<b>180</b>

# 1. About this document

This user guide explains how to set up and start operating your MiR100 robot and provides examples of simple missions you can expand to your purposes. This guide also contains information regarding the external and internal components of MiR100 along with instructions for proper maintenance of the robot. You will also find information regarding safety and specifications needed to commission a safe MiR100 robot application.



## NOTICE

Save this manual. It contains important safety and operating instructions.

## 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. It comes 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.
- **Operating guides** describe how to set up and use MiR accessories or supported functions that are mainly hardware-based, such as charging stations and shelf functions.
- **Getting started guides** describe how to set up MiR accessories 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.
- **Best practice guides** provide helpful information you can use when commissioning or operating your robot.
- **REST API references** for MiR robots, MiR hooks, and MiR Fleet. HTTP requests can be used to control robots, 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.

the robot			
			
Revision	Release date	Description	HW
3.2	2021-08-27	Updated for HW version 6.0 with new battery.	6.0
3.1	2020-10-30	Updates and improvements throughout the manual.	5.0
3.0	2020-10-01	<p>New manual structure.</p> <p>New chapters: Warning label, Mounting the nameplate, Battery and charging, IT security, Navigation and control system, Safety system, Usage, Disposal of robot, Error handling and Glossary.</p> <p>As a result of the new manual structure, the MiR100 User Guide has been aligned with the top module version history. Therefore, version 2.0 is skipped and is instead named 3.0.</p>	5.0
1.4	2019-08-20	<p>Updated for hardware release 5.0.</p> <p>Updates and improvements throughout the manual.</p>	5.0
1.3	2019-06-26	<p>Updated for hardware release 4.0.</p> <p>Updates and improvements throughout the manual.</p>	4.0
1.2	2019-01-04	<p>Updated for hardware release 3.0.</p> <p>Updates and improvements throughout the manual.</p>	3.0
1.1	2018-08-17	Updated for hardware release 2.1.	2.1

the robot			
			
Revision	Release date	Description	HW
		Updates and improvements throughout the manual.	
1.0	2017-11-27	First edition.	1.0

## 2. Product presentation

MiR100 is an autonomous mobile robot that can transport loads up to 100 kg indoors within production facilities, warehouses, and other industrial locations where access to the public is restricted.



Users operate MiR100 via a web-based user interface, which is accessed through a browser on a PC, smartphone, or tablet. Each robot has its own network—see [Connecting to the robot interface on page 39](#). The robot can be set up to run a fixed route, be called on demand, or perform more complex missions.

The robot interface of MiR100 can be accessed via Google Chrome, Google Chromium, Apple Safari, Mozilla Firefox, and Microsoft Edge browsers.

The robot uses a map of its work area to navigate and can move to any position on the map—see [Navigation and guidance system on page 59](#). The map can be created or imported the first time the robot is used. While operating, the robot avoids obstacles that are not mapped, like people and furniture.



Specifications for MiR100 are available on the MiR website.

## 2.1 Main features of MiR100

The main features of MiR100 are:

- **Driving in a populated workspace**  
The robot is designed to operate among people and maneuvers safely and efficiently in highly dynamic environments.
- **Overall route planning and local adjustments**  
The robot navigates autonomously to find the most efficient paths to its destinations. The robot adjusts the path when it encounters obstacles that are not on the map, like personnel and vehicles.
- **Efficient transportation of heavy loads**  
The robot is designed to automate transportation of loads up to 100 kg.
- **Sound and light signals**  
The robot continuously signals with light and sounds, indicating where it will drive and its current status, for example, waiting for a mission, driving to a destination, or destination reached.
- **User-friendly and flexible**  
The web-based user interface, accessed from a PC, tablet, or smartphone, gives easy access to operation and monitoring of the robot and can be programmed without any prior experience. Different user group levels and tailored dashboards can be set up to suit different users.
- **Alert for 'lost'**  
If the robot enters a situation where it is unable to find a path to its destination, it stops, turns on the yellow-purple running error light, and a custom defined Try/Catch action may be used to alert personnel or take other actions—see [Creating the mission Try/Catch on page 146](#).
- **Automatic deceleration for objects**  
The built-in sensors ensure that the robot is slowed down when obstacles are detected in front of it.

- **Internal map**

The robot can either use a floor plan from a CAD drawing, or a map can be created by manually driving the robot around the entire site in which the robot is going to operate. When the robot is mapping, the robot's sensors detect walls, doors, furniture, and other obstacles, and the robot then creates a map based on these input. After you've finished mapping, you can add positions and other features in the map editor—see [Creating and configuring maps on page 92](#).

## Top modules

The following top modules are available for MiR100:

- **MiR Hook 100**

A hook can be mounted on the robot enabling it to automate the internal transport of carts.



To learn more about the top modules and accessories, go to [the MiR website](#).

## Accessories

The following accessories are available for MiR100:

- **MiR Fleet**

MiR Fleet offers centralized control of robots throughout all your facility. MiR Fleet is available as an installation file you can install on a server of your choice or as a computer server where MiR Fleet comes pre-installed.

- **MiR AI Camera**

An intelligent camera that detects and recognizes different moving objects, which can trigger certain actions in MiR Fleet.

- **MiR Charge 24V**

These fully automatic chargers enable MiR robots to recharge their batteries. Together with MiR Fleet, charging of robots can be fully automated, ensuring that robots never run out of power during missions.

## 2.2 External parts

This section presents the parts of MiR100 that are visible on the outside.

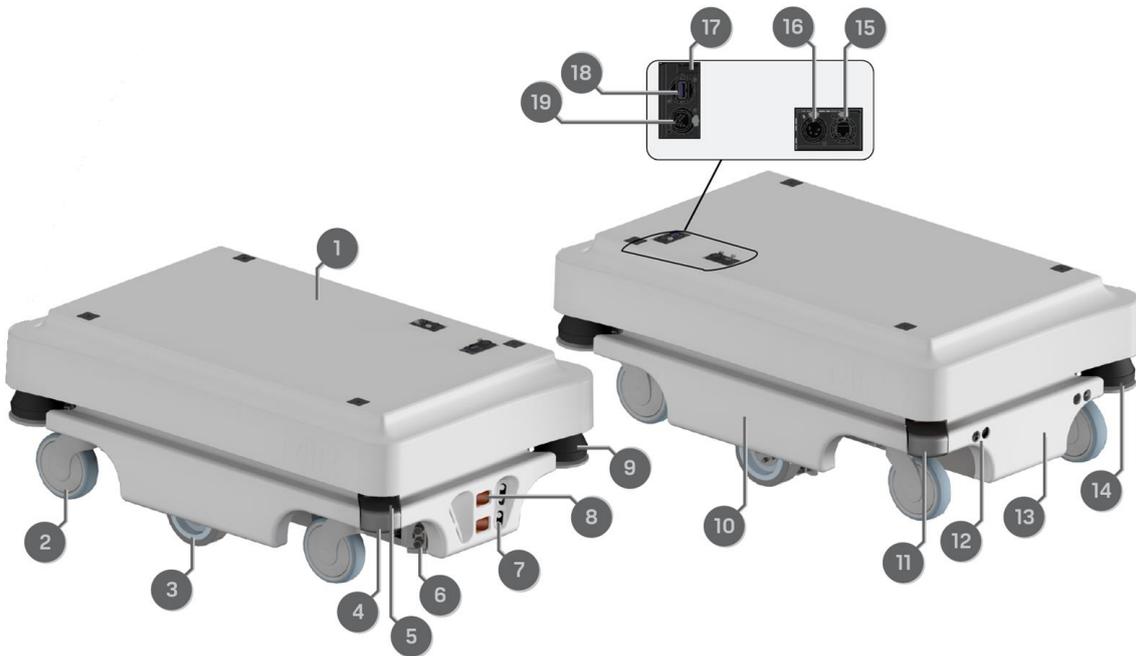


Figure 2.1. MiR100 external parts.

Table 2.1. Identification of the external parts in <i>Figure 2.1</i>			
Pos.	Description	Pos.	Description
1	Top cover: access to internal parts—see <a href="#">Internal parts on page 18</a>	2	Caster wheel: four pcs., one in each corner
3	Drive wheel: two pcs., differential control	4	Behind the removable front corner cover: : HDMI port and USB service port - connects to the robot computer
5	Scanner reset button (yellow) and On/Off button (blue)	6	Ultrasound sensors: two pcs., for detection of transparent objects (side)—see <a href="#">Obstacle detection on page 64</a>

Pos.	Description	Pos.	Description
7	3D depth camera: two pcs., both in the front—see <a href="#">Obstacle detection on page 64</a>	8	Pad connectors: two pcs., for connection to charging pins on MiR Charge 24V charging station
9	S300 safety laser scanner (front)—see <a href="#">Obstacle detection on page 64</a>	10	Side cover
11	Behind removable rear corner cover: Charging port with switch	12	Ultrasound sensors: two pcs., for detection of transparent objects (rear)—see <a href="#">Obstacle detection on page 64</a>
13	Rear cover	14	S300 safety laser scanner (rear)—see <a href="#">Obstacle detection on page 64</a>
15	RJ45 Ethernet—see <a href="#">Interface specifications on page 173</a>	16	Application power: for connection to top modules such as hooks—see <a href="#">Interface specifications on page 173</a>
17	Antenna socket	18	USB interface - connects to the robot's PC
19	Emergency stop interface: with added options for connection to small units and I5 input on SICK scanners—see <a href="#">Interface specifications on page 173</a>		

## Identification label

MiR100 is delivered with an identification label mounted to the product. The identification label identifies the product, the product serial number, and the hardware version of the product.

The identification label of MiR100 is located on the rear-left side of the chassis underneath the router.



Figure 2.2. Placement of the identification label.

Item number:	800127
Product serial number:	201703000
Hardware version:	V.5

Figure 2.3. Example of a MiR100 identification label.

## Nameplate

Every MiR application is delivered with a nameplate that must be mounted on the robot. The nameplate of MiR100 identifies the application model and serial number and includes the CE mark, technical specifications, and the address of Mobile Industrial Robots. The nameplate identifies the complete MiR application, for example, a robot with a top module.

It is the responsibility of the commissioner to mount the nameplate on the application—see [Mounting the nameplate on page 47](#).

Weight without payload:	70 kg	Nominal power:	2 × 188 W	Contains FCC ID:	TV7RB952-5AC2ND
Maximum speed:	1.5 m/s	Type:	Driverless truck	Contains FCC ID:	RYK-261ACNBT
Maximum incline/decline:	5%	Model:	MiR100		
Maximum payload:	100 kg	S/N:	100103000	This device complies with part 15 of the FCC rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.	
Footprint of payload:	Equal to robot footprint	Year of construction:	2021		
Lithium-ion battery		Made in Denmark			
Nominal voltage:	24 V	Mobile Industrial Robots A/S			
Maximum capacity:	39.6 Ah	Emil Neckelmanns Vej 15F			
Battery weight:	8.2 – 16.4 kg	DK-5220 Odense SØ			
Dimensions	163 × 169 × 212 mm				

Figure 2.4. Example of a MiR100 nameplate.

## Control panel

MiR100 has a control panel in the rear-left corner of the robot.

## The control panel buttons

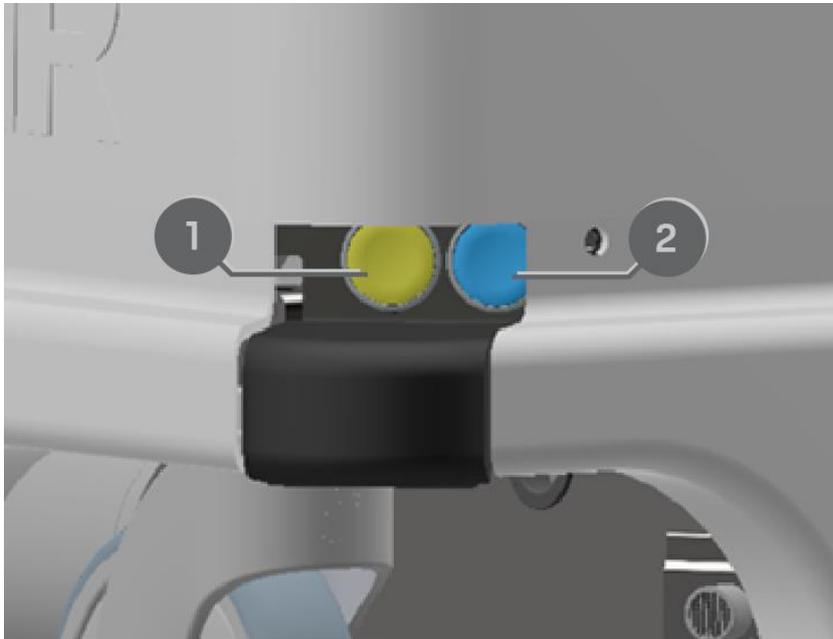


Figure 2.5. The MiR100 control panel.

Table 2.2. MiR100 control panel			
Pos.	Description	Pos.	Description
1	Scanner reset	2	Power

### Scanner reset

Pressing this button restarts the scanners after 5-7 seconds. This can be useful if you experience issues with the safety laser scanners.

### Power

Pressing this button for three seconds turns the robot on or shuts it off.

## Operating modes

MiR100 has two operating modes: Manual mode and Autonomous mode.

## Manual mode

In this mode, you can drive the robot manually using the joystick in the robot interface. Only one person can control the robot manually at a time. To ensure that nobody else takes control of the robot, the robot issues a token to the device on which you activate the Manual mode.

For information about activating this mode, see [Driving the robot in Manual mode on page 45](#).

## Autonomous mode

In this mode, the robot executes the programmed missions. The robot enters this mode automatically when you press **Continue** ► in the robot interface.

## 2.3 Internal parts

This section presents the parts of MiR100 that are visible on the inside after removing the top cover.



### CAUTION

Removing covers from the robot exposes parts connected to the power supply, risking damage to the robot from a short circuit and personnel getting burnt

- Before removing any covers, turn off the robot, and disconnect the battery—see [Disconnecting the battery on page 52](#).

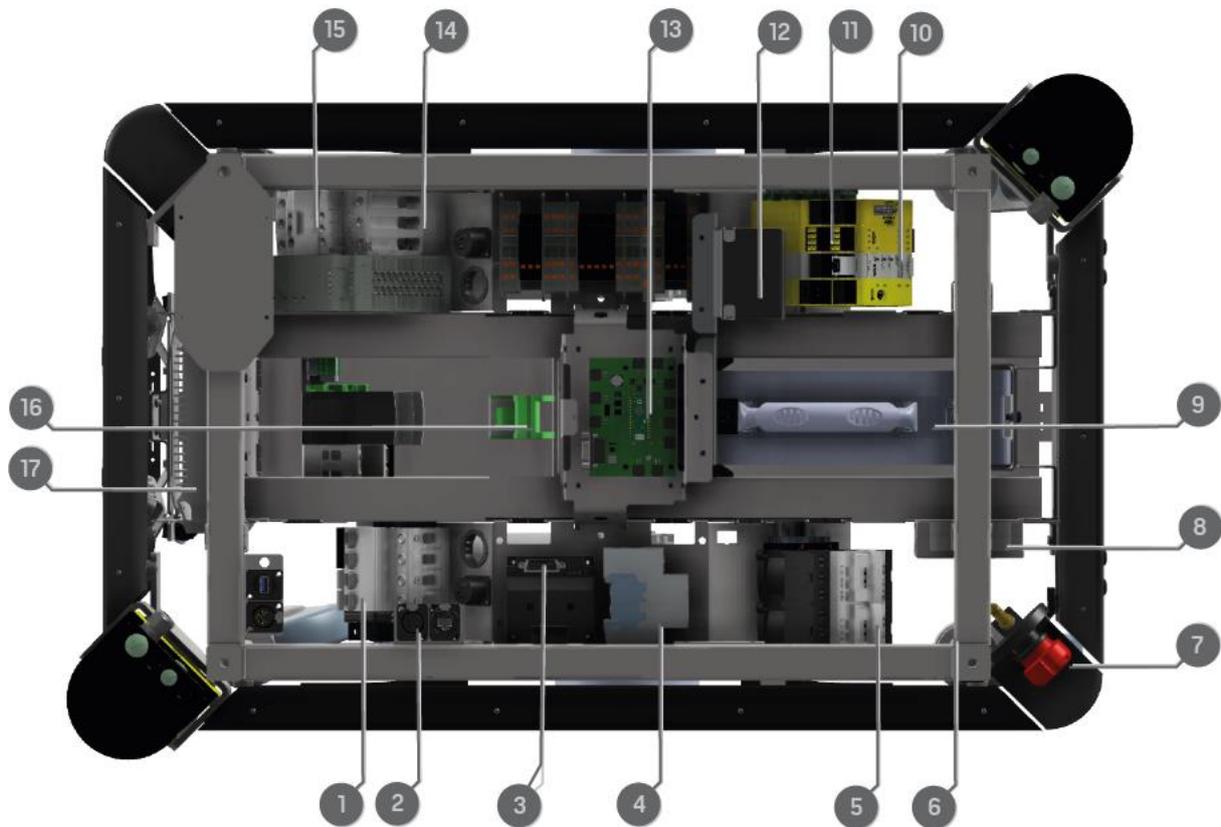


Figure 2.6. Internal parts of MiR100.

Table 2.3. Identification of internal parts in <i>Figure 2.6</i>			
Pos.	Description	Pos.	Description
1	Breaker: automatic fuse between battery and components	2	Robot power off relay: releases the latching relay (pos. 17) when the robot is shutting down.
3	Motor controller: manages the two motor drives	4	Brake relay: short circuits motor windings for faster braking
5	Safe torque off relay (controlled by Safety PLC)	6	CAN bus connection for Battery Management System, logging data such as number of charge cycles

Pos.	Description	Pos.	Description
7	Battery disconnect switch	8	Router
9	Battery with connector: main power to the robot	10	Safety PLC
11	Optocoupler: emergency stop signal to motor controller	12	Loudspeaker
13	MiR board: interface board for gyroscope, accelerometer, ultrasound, light, on/off circuit, and CAN bus communication	14	24 V power supply: secures stable voltage for the robot computer and PLC
15	Latching relay: activates the 24 V power supply turning on the robot	16	Transient protection module: provides circuit protection for the power supplies by absorbing voltage spikes from battery or top mounted applications
17	Robot computer		

## 3. Safety

Read the information in this section before powering up and operating MiR100.

Pay particular attention to the safety instructions and warnings.



### NOTICE

Mobile Industrial Robots disclaims any and all liability if MiR100 or its accessories are damaged, changed, or modified in any way. Mobile Industrial Robots cannot be held responsible for any damages caused to MiR100, accessories, or any other equipment due to programming errors or malfunctioning of MiR100.

### 3.1 Safety message types

This document uses the following safety message types.



### WARNING

Indicates a potentially hazardous situation that could result in death or serious injury. Carefully read the message that follows to prevent death or serious injury.



### CAUTION

Indicates a potentially hazardous situation that could result in minor or moderate injury. Alerts against unsafe practices. Carefully read the message that follows to prevent minor or moderate injury.



### NOTICE

Indicates important information, including situations that can result in damage to equipment or property.

## 3.2 General safety precautions

This section contains general safety precautions.



### WARNING

If the robot is not running the correct software and is therefore not functioning properly, the robot may collide with personnel or equipment causing injury or damage.

- Ensure that the robot is always running the correct software.



### WARNING

When the robot is in an operating hazard zone, there is a risk of injury to any personnel within the zone.

- Ensure that all personnel are instructed to stay clear of operating hazard zones when the robot is in the zone.



### WARNING

The robot may drive over the feet of personnel, causing injury.

- All personnel must be informed of the side Protective fields of the robot and be instructed to wear safety shoes near an operating robot—see [Personnel detection on page 79](#).



### WARNING

The robot may drive into a ladder, scaffold, or similar equipment that has a person standing on it. Personnel risk fall injuries, and equipment may be damaged.

- Don't place ladders, scaffolds, or similar equipment in the robot's work environment.



**WARNING**

The robot may drive down staircases or holes in the floor and cause serious injury to personnel and damage to the robot and to equipment.

- Mark all descending staircases and holes as Forbidden zones on maps.
- Install physical barriers around descending staircases and holes that are in the robot's operating area. If the hazard is not close to the robot's operating area, it may be sufficient to use a Forbidden zone on its own.
- Keep the maps up to date.
- Inform personnel that the robot cannot detect descending staircases and holes in the floor in time to stop.



**WARNING**

Contact with live electrical parts can cause electric shock.

- Do not touch any internal components of the robot while it is powered.



**WARNING**

Using a charging device different from the one supplied by the manufacturer can cause a fire and thereby burn injuries to nearby personnel and damage to the robot and equipment.

- Only use an original MiR charger.



## WARNING

Lithium battery packs may get hot, explode, or ignite and cause serious injury if they are misused electrically or mechanically.

Observe the following precautions when handling and using lithium-ion batteries:

- Do not short-circuit, recharge, or connect with false polarity.
- Do not expose to temperatures beyond the specified temperature range or incinerate the battery.
- Do not crush, puncture, or disassemble the battery. The battery contains safety and protection devices, which, if damaged, may cause the battery to generate heat, explode, or ignite.
- Do not allow the battery to get wet.
- In the event the battery leaks and the fluid gets into one's eye, do not rub the eye. Rinse well with water, and immediately seek medical care. If left untreated, the battery fluid could cause damage to the eye.
- In case of fire, use water to put out the fire. There is no need for special extinguishing media. An ambient fire can be fought with class D extinguishing media.
- Do not touch damaged batteries with bare hands. Only personnel using suitable Personal Protection Equipment (PPE) and tools should handle damaged batteries.
- Isolate the battery and keep clear if the following conditions are observed:
  - The battery exhibits abnormally high temperatures.
  - The battery emits abnormal odors.
  - The battery changes color.
  - The battery case is deformed or otherwise differs from the normal electrical or mechanical condition.
- Modifications or manipulations of the battery may lead to considerable safety risks and are therefore prohibited.
- Do not use the battery for anything other than MiR100.



**WARNING**

Load falling or robot overturning if the load on the robot is not positioned or fastened correctly can cause fall injuries to nearby personnel or damage to equipment.

- Ensure that the load is positioned according to the specifications and is fastened correctly—see [Payload distribution on page 168](#).



**CAUTION**

Load placed directly on top of the robot cover may cause damage to the cover of the robot.

- Ensure that the load is not placed directly on top of the robot cover.



**CAUTION**

Robot malfunctions can cause an electrical fire, causing damage and injury to equipment and personnel.

- Personnel operating near the robot must be informed on how to use an ABC fire extinguisher to put out an electrical fire should the robot malfunction and catch on fire.



**CAUTION**

Risk of trapping or injury to personnel if robots malfunction or if personnel enter operating hazard zones.

- Personnel operating near the robot must be informed on how to engage the robot's Emergency stop function in emergency situations.


**NOTICE**

Moving the robot by hand forcefully may cause damage to the top cover.

- If the robot is stuck, push or pull gently on the top cover corners to move the robot.

## 3.3 Intended use

MiR100 is intended to be commissioned and used in indoor industrial environments where access for the public is restricted. For details about the environmental conditions in which MiR100 should operate, see specifications for MiR100 on the MiR website.

MiR100 is intended to be commissioned according to the guidelines in [Commissioning on page 89](#). This is a prerequisite for safe usage of MiR100.

MiR100 is equipped with safety-related features that are purposely designed for collaborative operation where the robot operates without a safety enclosure or together with people—see [Safety-related functions and interfaces on page 75](#).

MiR100 is intended to be used with top modules supported by Mobile Industrial Robots or custom modules that:

- Do not have any moving parts.
- Do not extend the physical dimensions of the robot.
- Operate within the environmental conditions required for MiR100.
- Are within the weight distribution requirements—see [Payload distribution on page 168](#).

If used with custom modules, all obligations of a manufacturer apply to the individual who performs the modifications. All modifications must be in accordance with the machinery directive.

MiR100 is designed for and all risks are considered when used with one of the following types of top modules:

- MiR Hook 100 to tow carts.

MiR100 can be used as a partly complete machine as defined in the EU machinery directive with top modules that do not meet the above limitations. Those who design, manufacture,

or commission a system that does not meet the limitations of use of MiR100 carry the obligations of a manufacturer and shall ensure a safe design according to EN ISO 12100. Guidelines outlined in this manual are not sufficient.

The following list gives examples of custom modules that are foreseeable misuse of MiR100:

- Top modules (including the payload) that increase the physical dimensions of MiR100
- Conveyers (powered and non-powered)
- Industrial robot arms
- Devices that tow carts
- Customized load transfer stations



**NOTICE**

A safe machine does not guarantee a safe system. Follow the guidelines in [Commissioning on page 89](#) to ensure a safe system.

## 3.4 Users

MiR100 is only intended to be used by personnel that have received training in their required tasks.

There are three types of intended users for MiR100: commissioners, operators, and direct users.

### Commissioners

Commissioners have thorough knowledge of all aspects of commissioning, safety, use, and maintenance of MiR100 and have the following main tasks:

- Commissioning of the product. This includes creating maps and restricting the user interface for other users and making brake tests with a full payload.
- Conducting the risk assessment.
- Determining the payload limit, weight distribution, safe fastening methods, safe loading and unloading of loads on MiR100, and ergonomic loading and unloading methods if relevant.

- Ensuring the safety of nearby personnel when the robot is accelerating, braking, and maneuvering.
- Marking operating hazard zones.

## Operators

Operators have thorough knowledge of MiR100 and of the safety precautions presented in this user guide. Operators have the following main tasks:

- Servicing and maintaining MiR100.
- Creating and changing missions and map features in the robot interface.

## Direct users

Direct users are familiar with the safety precautions in this user guide and have the following main tasks:

- Assigning missions to MiR100.
- Fastening loads to MiR100 securely.
- Loading and unloading from a paused robot.

All other persons in the vicinity of MiR100 are considered indirect users and must know how to act when they are close to the robot. For example, they must be aware that visibly marked operating hazard zones must be respected.

## 3.5 Foreseeable misuse

Any use of MiR100 deviating from the intended use is deemed as misuse. This includes, but is not limited to:

- Using the robot to transport people
- Using the robot on inclines outside the robot's specifications
- Making changes to the SICK configuration
- Driving the robot on cross slopes
- Exceeding the maximum payload
- Positioning or fastening loads incorrectly according to the specifications—see [Payload distribution on page 168](#)
- Using Emergency stop buttons for anything other than emergency stops

- Using the robot in medical and life critical applications
- Operating the robot outside the permissible operating parameters and environmental specifications
- Using the robot in potentially explosive environments
- Using the robot outdoors
- Using the robot in hygiene zones

## 3.6 Warning label

MiR100 is supplied with a warning label that specifies that it is strictly prohibited to ride on the robot.

The label must be placed on the robot or top module so that it is clearly visible.



Figure 3.1. The warning label must be placed on the robot or top module.

## 3.7 Residual risks

Mobile Industrial Robots has identified the following potential hazards that commissioners must inform personnel about and take all precautions to avoid when working with MiR100:

- You risk being run over, drawn in, trapped, or struck if you stand in the path of the robot or walk towards the robot or its intended path while it is in motion.
- You risk being run over, drawn in, trapped, or struck if you stand in the path of the robot or walk towards it while it is driving in reverse. The robot only drives in reverse when undocking from a marker, such as a charging station or load transfer station.
- You risk being crushed or trapped if you touch the robot while it is in motion.

- You risk being crushed or trapped if the robot places a load outside a designated drop-off area due to faulty localization.
- You risk losing control of the robot if it is accessed by unauthorized users. Consider increasing the IT security of your product—see [IT security on page 57](#).



**NOTICE**

Other significant hazards may be present in a specific robot installation and must be identified during commissioning.

## 4. Getting started

This section describes how to get started with MiR100.



### NOTICE

Read [Safety on page 21](#) before powering up MiR100.

### 4.1 In the box

This section describes the contents of the MiR100 box.



Figure 4.1. The robot and accessories.

The box contains:

- The MiR100 robot
- The robot kit which contains:
  - One Emergency stop box, external antenna, and four pcs. M10x40 bolts
  - One charging cable
  - One external charger, 24 V DC, 10 A
- A MiR100 document folder containing a USB flash drive and the following printed documents:
  - *MiR100 Quick Start*
  - *The CE Declaration of Conformity for your robot*
  - *Getting the robot online*
  - *Passwords*
  - The unique nameplate for your robot
- The USB flash drive in the document folder has the following content:
  - *MiR100 User Guide*
  - *MiR100 Quick Start*
  - *MiR Network and WiFi Guide*
  - *MiR Robot Reference Guide*
  - *MiR Robot REST API Reference*
  - *Getting the robot online*
  - *CE Declaration of Conformity*

## 4.2 Unpacking MiR100

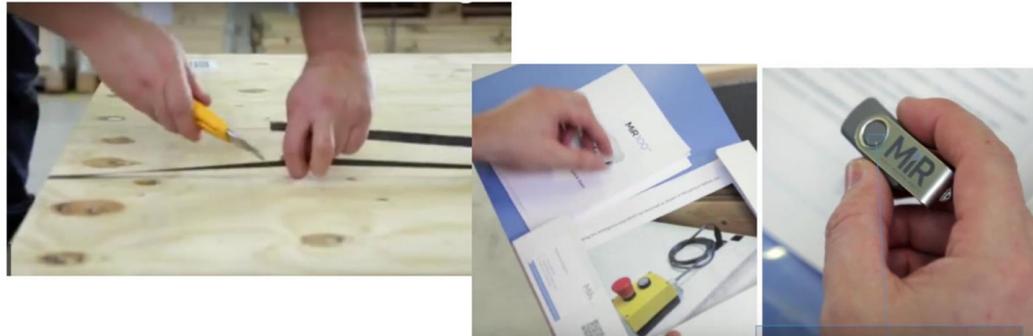
This section describes how to unpack the robot.



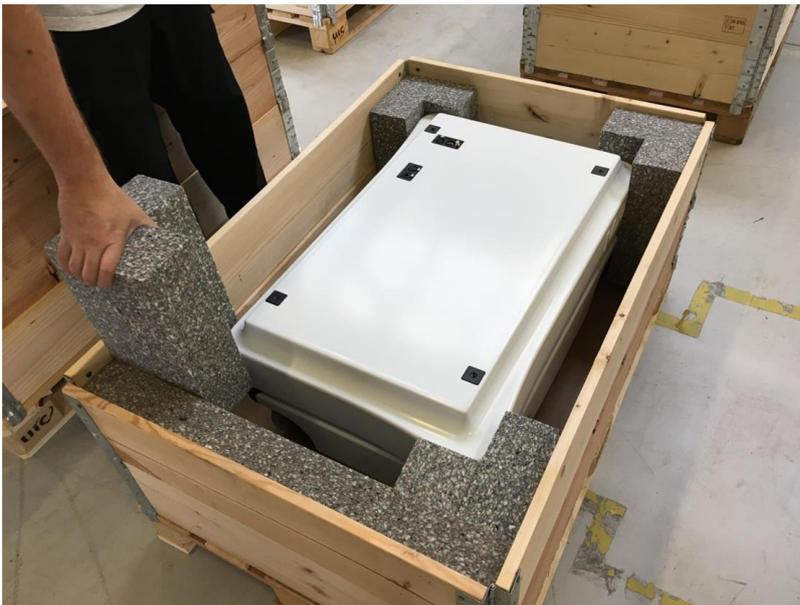
Keep the original packaging for future transportation of MiR100.

To unpack the robot, follow these steps:

1. Place the box with the robot so that there is three meters of free space at the front or the back of the box. This is necessary as the robot drives out of the box on a ramp.
2. Cut the protective straps surrounding the box.
3. Remove the lid from the box.
4. Take the folder with the printed documents and the USB flash drive out of the box.



5. Remove the walls of the box and the protective foam blocks.



6. Place the lid of the box so that you can use it as a ramp at the robot's front or rear end. Align the lid so that it is flush with the base of the box.



## 4.3 Connecting the battery

To connect the battery to the robot, follow these steps:

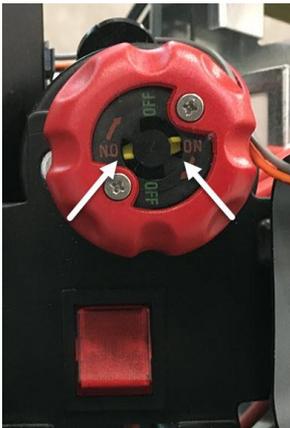
1. Grab the two rounded corners and carefully lift off the cover.



2. Switch on the three relays placed in the corner by the front laser scanner.



3. Ensure that the Battery disconnect switch, placed in the rear right corner, is on (the two yellow indicators pointing to On).



4. Put the cover back on and make sure to fit it correctly over the connector openings.



5. Mount and connect the Emergency stop box on top of the robot cover.





If a top module is going to be mounted on top of the robot, the Emergency stop must be placed in a position where it is easy to reach—see the mounting instructions for your top module.

6. Connect the antenna to the connector on top of the robot cover. Remove the plastic cap from the connector before fixing the antenna.

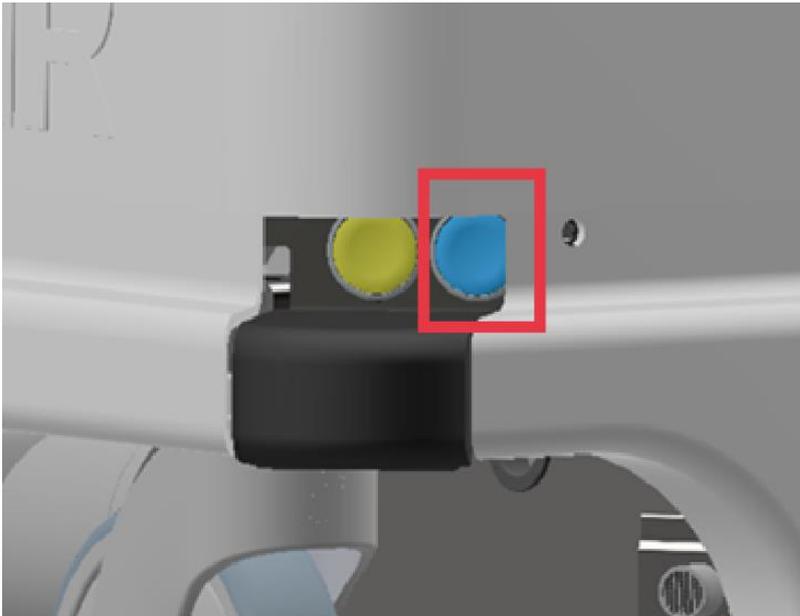


The antenna can be lowered and rotated in all directions to fit under a top module.

## 4.4 Powering up the robot

To power up the robot, follow these steps:

1. Press the Power button in the corner to turn on the robot. The status lights waver yellow, and the robot starts the software initialization process. When the initialization process ends, the robot goes into Protective stop.



2. Press the reset button on the Emergency stop when the button lights up. The status lights switches to yellow constant light, indicating that the robot is paused and ready to

operate.



## 4.5 Connecting to the robot interface

When the robot is turned on, it enables the connection to its WiFi access point. The name of the access point appears in the list of available connections on your PC, tablet, or phone.



### 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.

To connect to the robot interface, follow these steps:

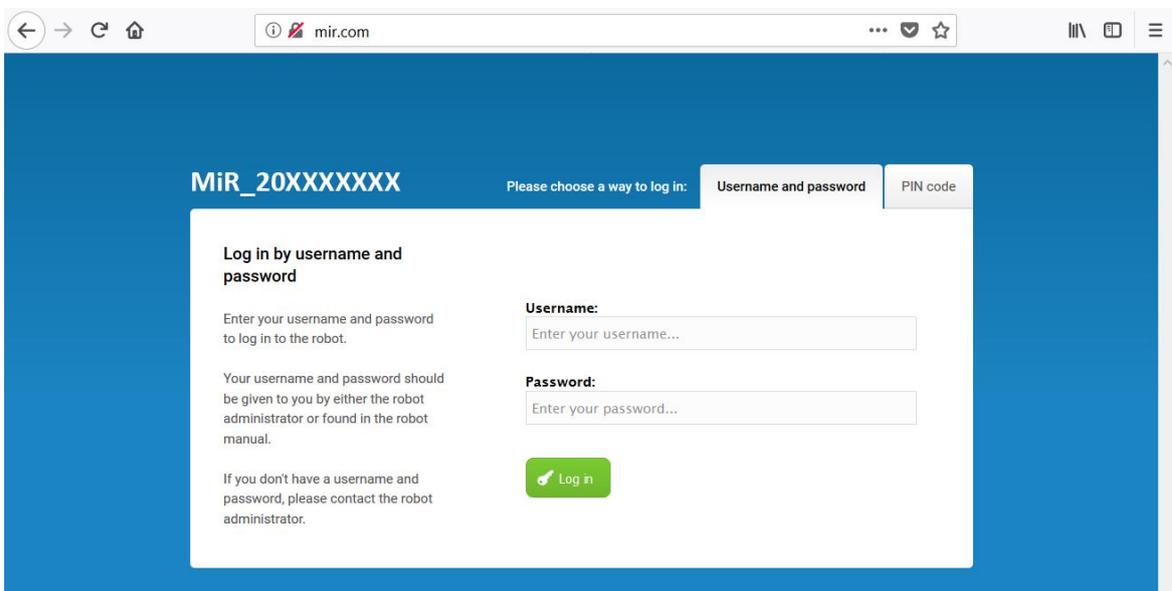
1. Using your PC, tablet, or phone, connect to the WiFi access point of the robot using the unique password for the WiFi access point. The access point name has the following format: **MiR\_20XXXXXXX**.



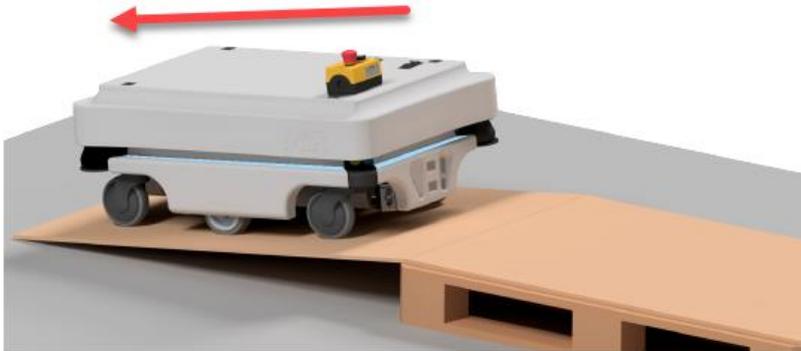
The access point name is derived from the robot application's model serial number.



2. In a browser, go to the address **mir.com** and sign in.



3. Switch to Manual mode, and drive the robot down the ramp—see [Driving the robot in Manual mode on page 45](#).



Always make sure your robot is running the latest recommended software. This also applies to newly supplied robots, as there may have been important software updates during the storage time and shipment of the robot. Contact your distributor for the latest recommended update file.

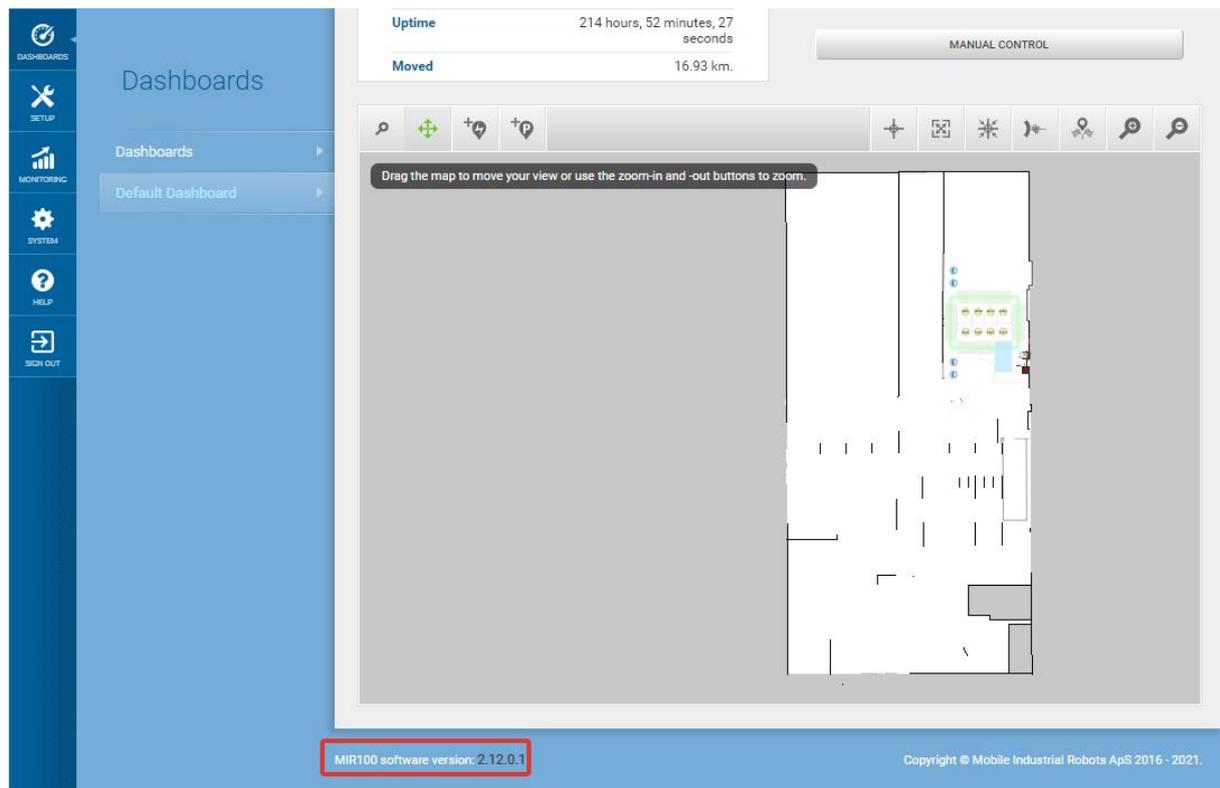


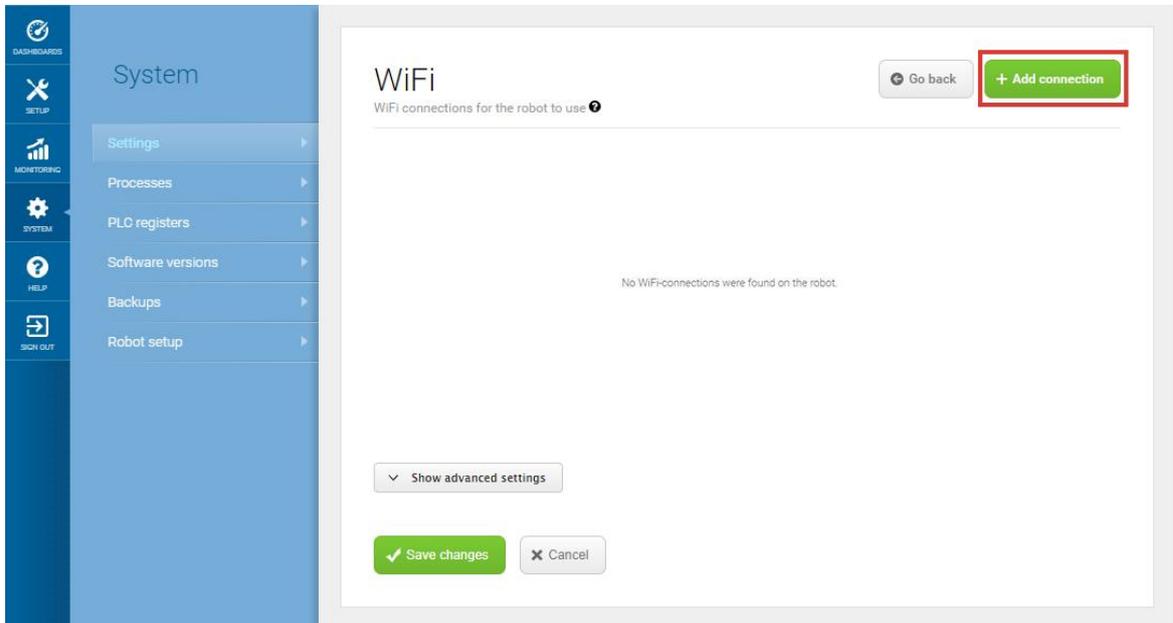
Figure 4.2. You can see the software version your robot is running in the bottom left corner of the robot interface.

## 4.6 Connecting the robot to a WiFi network

To communicate with the robot wirelessly without connecting an access point, you can connect the robot to your local WiFi network. This enables you to access the robot interface via your WiFi network by entering the assigned IP address of your robot in a web browser.

To connect the robot to a WiFi network, follow these steps:

1. Connect to your robot as described in [Connecting to the robot interface on page 39](#).
2. Go to **System > Settings > WiFi**, and select **+ Add connection**.



3. Select the network you want the robot to be connected to, and fill out the displayed fields.

**Add connection**

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

**Select a network:**

MIR

**Security type:**

WPA/WPA2 Enterprise TLS

**Username:**

**Password:**

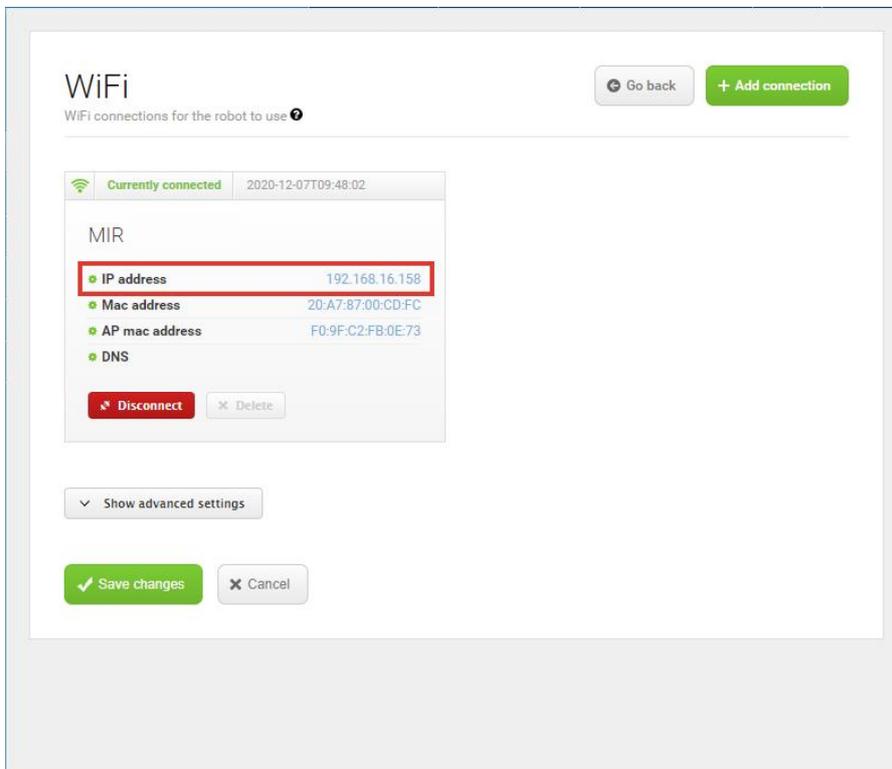
**DNS servers**

**Certificate:**

No file chosen

Use static IP

4. Select **+ Add connection** when you have finished. The robot is now connected to the network. When you are connected to the same network, you can access the robot's interface by entering the IP address displayed under the connection description into your internet browser.



## 4.7 Driving the robot in Manual mode



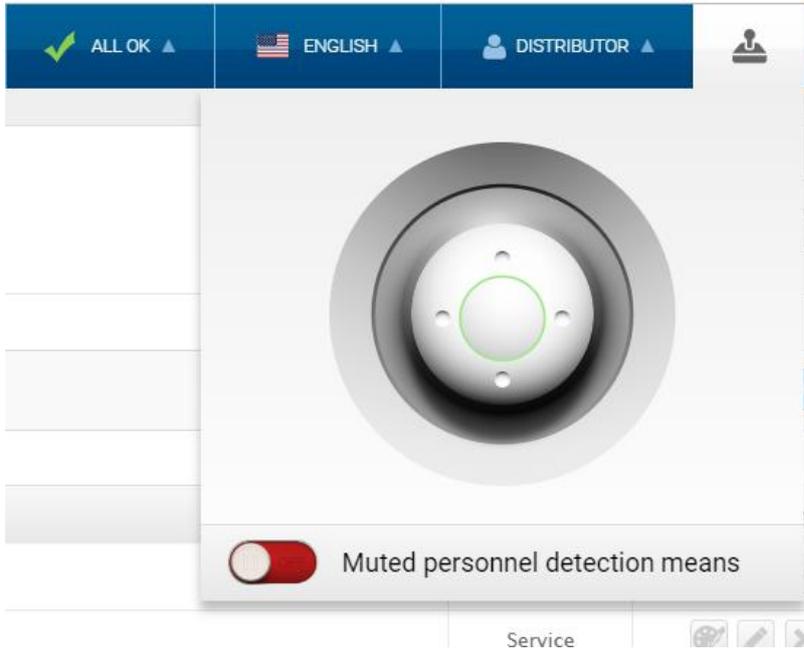
### CAUTION

When driving the robot in manual mode, it is possible to drive the robot into Forbidden zones and Unpreferred zones on the map. This can result in injury to personnel or damage to equipment if the robot is not driven carefully.

- Drive carefully to avoid collisions with any personnel or objects when driving the robot in Manual mode.
- Avoid driving the robot manually without a clear visual of the robot.

To drive the robot in Manual mode, follow these steps:

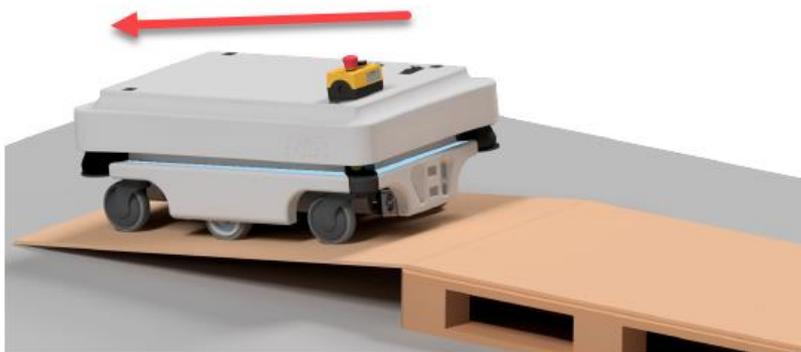
1. In the robot interface, select the joystick icon. The joystick control appears.



- 2.



Place your foot in front of the ramp while the robot drives on it to keep the ramp from slipping.



## 4.8 Checking the hardware status

To check that all hardware components work as intended, follow these steps:

1. Sign in to the robot interface—see [Connecting to the robot interface on page 39](#).
2. Go to **Monitoring > Hardware health**.
3. Check that all elements on the page have the **OK** status and that they have green dots on the left.

For more information, see Hardware health in *MiR Robot Reference Guide* on the MiR website.

## 4.9 Mounting the nameplate

Before using MiR100, you must mount its unique nameplate to it. The nameplate contains information specific to your MiR application—see [Nameplate on page 16](#).



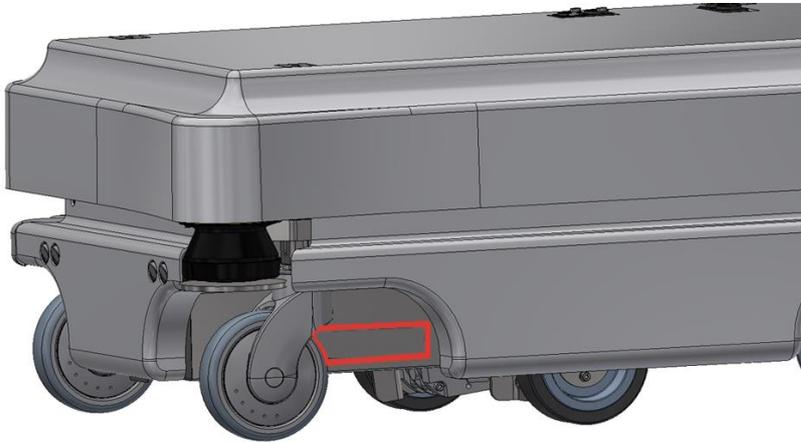
### NOTICE

The nameplate must be mounted as described in the following steps. If mounted incorrectly, the CE mark is invalid.

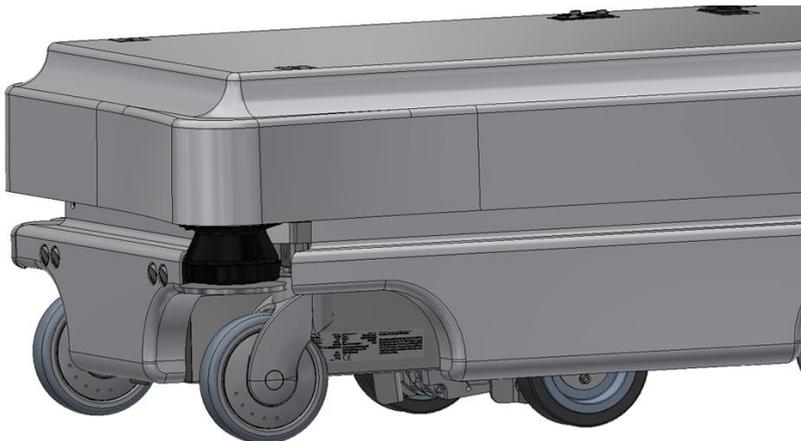
To mount the nameplate correctly, follow these steps:

1. Locate the area below the side cover near the swivel wheel at the rear end of the robot—see [External parts on page 12](#).

2. Clean the area marked in the image below with a degreasing agent. If you cannot access the area, either lift the robot to an appropriate height or remove the top cover to gain access.



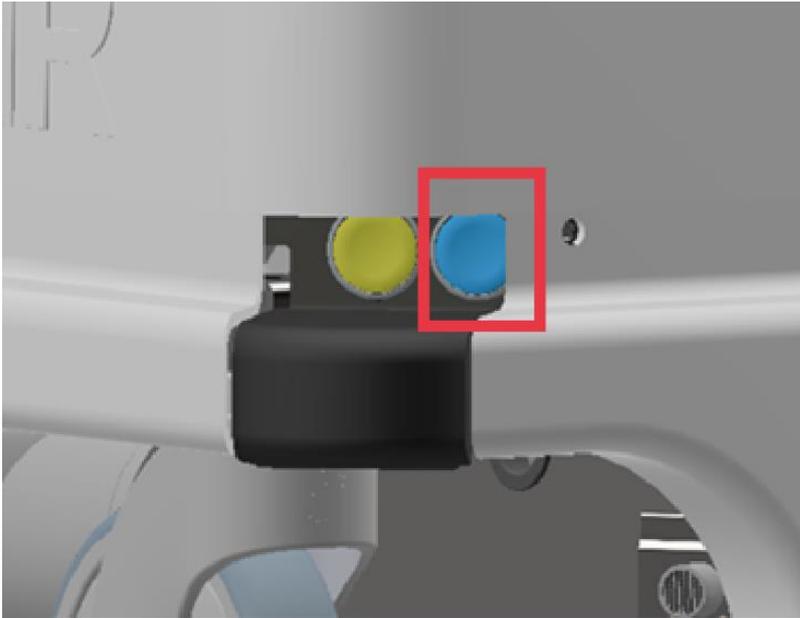
3. Mount the nameplate on the cleaned area.



## 4.10 Shutting down the robot

To shut down MiR100, follow these steps:

1. Ensure that the robot is not moving or executing an action.
2. Press the Power button for three seconds.



3. The robot starts the shutdown process. The status lights waver yellow.
4. When the robot finishes the shutdown process, the status lights are off.

When you shut down the robot for transportation, service, or repair, the battery must be disconnected—see [Disconnecting the battery on page 52](#).

## 5. Battery and charging

The robot is powered by a lithium battery that can be charged with a MiR cable charger or a MiR Charge 24V charging station.

### 5.1 Charging the robot

This section describes how to charge MiR100 using a MiR cable charger.



The robot is delivered 40-60% charged.



To avoid fast discharging and depleting the battery, we recommend that you turn off the robot while charging with a cable.



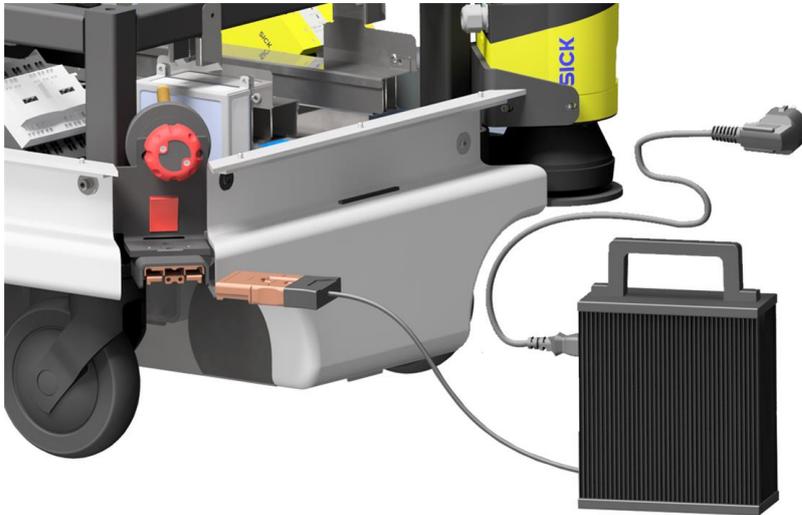
If you are charging two robots right after each other with a cable, wait approximately one minute between unplugging the first robot and plugging in the second. This will ensure that the charger registers that a new robot is being charged.

To charge MiR100 using the cable charger, connect the cable charger to the charging interface on the robot in the rear-left corner. To do this, follow these steps:

1. Remove the rear corner by pulling it towards you. You may have to apply a bit of force the first couple of times.



2. Attach the charger to the robot's charging socket and to a power outlet. Turn on the rocker switch on the robot to begin charging.



Use only an original MiR cable charger.

3. Look at the indicator lights on the charger to confirm that the robot starts charging. If the plug is not connected properly to the socket, the robot will not start the charging.
4. Turn off the rocker switch and disconnect the charging cable from the robot. Slide the corner cover back on.



The robot detects both cable and activated charging-button. If either the charging switch is turned on or the cable charger is connected, the robot enters Protective stop.

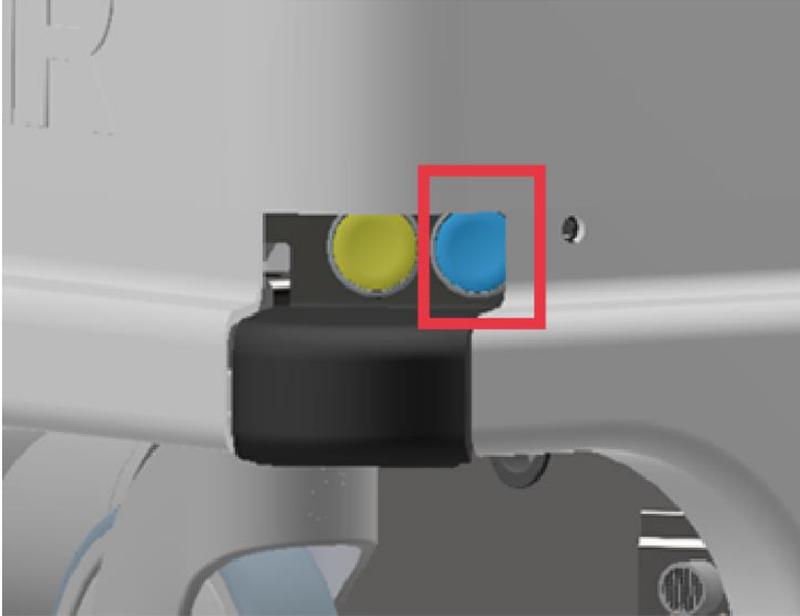
For information about the charging time, see specifications on the MiR website.

## 5.2 Disconnecting the battery

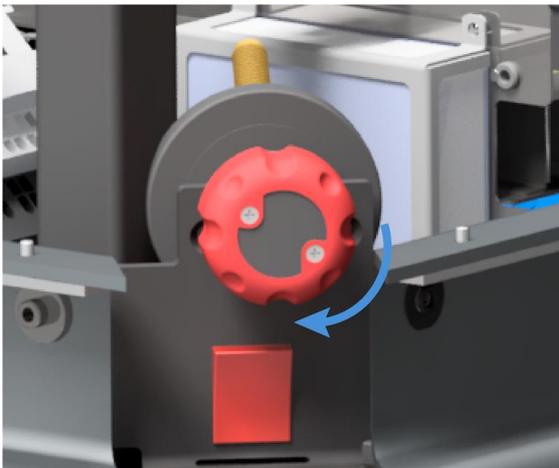
Whenever the robot is to be transported, undergo maintenance, or stored for over 24 hours, you should always disconnect the battery.

To disconnect the battery, follow these steps:

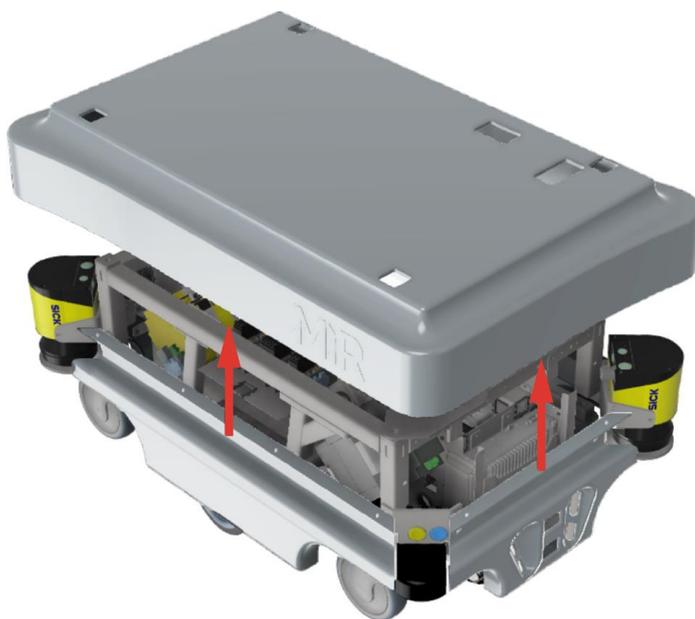
1. Turn off the robot.



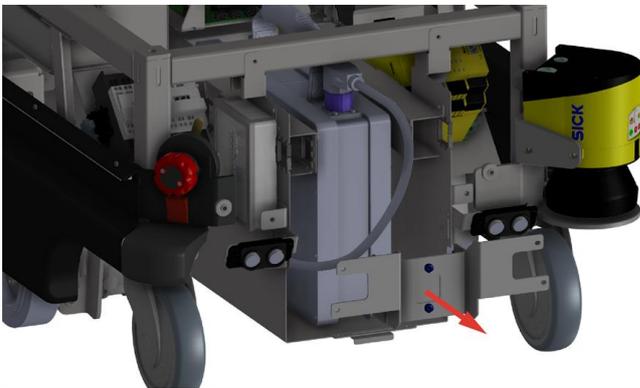
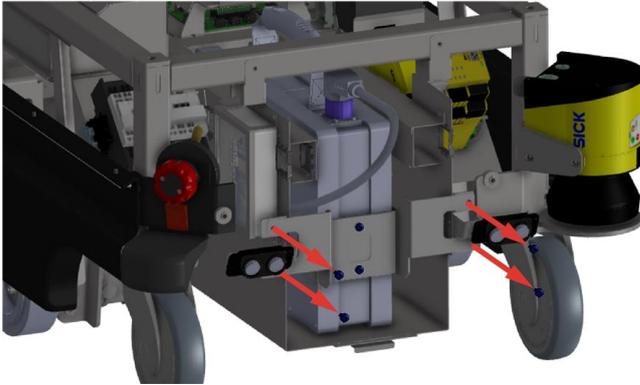
2. Turn the Battery disconnect switch to Off (the two yellow indicators pointing to Off).



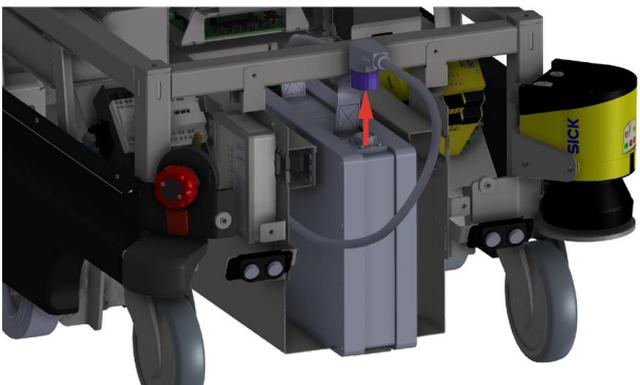
3. Remove the top cover.



4. With a screwdriver, remove the screws that fastens the battery bracket to the robot and remove the bracket.



5. Slide out the battery a bit, and disconnect the battery by unplugging the battery cable from the battery box.



6. Slide the battery back in place, fasten the battery bracket, and place the top cover back on the robot.

## 5.3 Battery storage

The battery should be stored in an area at room temperature with a non-condensing relative air humidity—see specifications on the MiR website. Temperatures and humidity below or above the specifications will shorten the service life of the battery.

The battery should not be exposed to nor submerged in any liquid as this may damage the battery.

Charge the battery before storage to preserve the service life of the battery.

To preserve the battery, disconnect the battery from the robot before storing the robot.

## 5.4 Battery disposal

Legal obligations apply to used batteries. Used batteries must be recycled.

Disposing used batteries in the household waste is prohibited.

A recycling label indicates that the battery needs to be recycled and not disposed as municipal waste—see **Figure 5.1**

Contact your distributor to get specific information about their take back service.



Figure 5.1. Battery disposal symbols.

## 6. IT security

IT security is a set of precautions you can take to prevent unauthorized personnel from accessing MiR100. This section describes the main IT-security related risks and how to minimize them when commissioning MiR100.

MiR100 communicates all data over the network that it is connected to. It is the responsibility of the commissioner to ensure that it is connected to a secure network. MiR recommends conducting an IT-security risk assessment before commissioning the robot.



Contact your distributor for a list of FAQs about IT security and the guide *How to set up MiR products to improve the IT security*.

### 6.1 Managing users and passwords

Managing your users and passwords is the main way you can control access to MiR100.

There are three default users with predefined passwords for you to start using. These are described in the *MiR Robot Reference Guide* along with instructions to create new users, user groups, and passwords. MiR advises you to:

- Change the default password for all predefined users if you choose to continue to use them. Make sure to choose a strong password since MiR100 does not enforce any password rules nor expire the password.
- Create new user groups if more levels of access are necessary.
- Create dedicated user accounts under the relevant user group for each person accessing MiR100, and ensure that the users change the password on their first sign-in. It is not recommended to have several users share the same account.
- Only enable users with a minimum level of access to use a pin code to sign in. Users with a higher level of access are recommended to use a strong password to sign in instead.

### 6.2 Software security patches

To improve the security of MiR100, MiR supplies security patches to the operating system in new MiR software update files. When you install a security patch, it takes approximately 10-15 minutes longer to update a MiR product.



Security patches are included from software version 2.8.3 and higher.

## Understanding MiR software versions

MiR uses the **Major.Minor.Patch.Hot fix** format to version software. For example, 2.8.1.1 means that the software is based on the second major release, the eighth minor release of the major version, the first patch release of the minor version, and, in this example, a single hot fix is included too.

- **Major releases** include the most significant changes that affect the entire robot software.
- **Minor releases** often include new features and smaller changes that only affect parts of the software.
- **Patch releases** focus on fixing small issues in the software and introducing quality improvements.
- **Hot fix releases** are only created when a patch release has introduced a critical issue that needs to be fixed immediately.

## Security patch policy

MiR applies the following policy when supplying security patches:

- New security patches are distributed per every minor release.
- All patch releases under a minor release include the previous security patches also. In other words, if you chose not to install the first software version in a minor release, such as version 2.9.0, the security patches will still be installed when you update to 2.9.1 or higher.

## 7. Navigation and guidance system

The navigation and guidance system is responsible for driving the robot to a goal position while avoiding obstacles. This section describes the processes and components involved in the robot's navigation and guidance system.

### 7.1 System overview

The purpose of the navigation and guidance system is to guide the robot from one position on a map to another position. The user provides the map and chooses the goal position the robot must move to. The diagram in **Figure 7.1** describes the processes in the system.

The main processes involved in the navigation system are:

- **Global planner**  
The navigation process starts with the global planner determining the best path for the robot to get from its current position to the goal position. It plans the route to avoid walls and structures on the map.
- **Local planner**  
While the robot is following the path made by the global planner, the local planner continuously guides the robot around detected obstacles that are not included on the map.
- **Obstacle detection**  
The safety laser scanners, 3D cameras, and ultrasound sensors are used to detect obstacles in the work environment. These are used to prevent the robot from colliding with obstacles.
- **Localization**  
This process determines the robot's current position on the map based on input from the motor encoders, inertial measurement unit (IMU), and safety laser scanners.
- **Motor controller and motors**  
The motor controller determines how much power each motor must receive to drive the robot along the intended path safely. Once the robot reaches the goal position, the brakes are engaged to stop the robot.

Each part of the process is described in greater detail in the following sections.

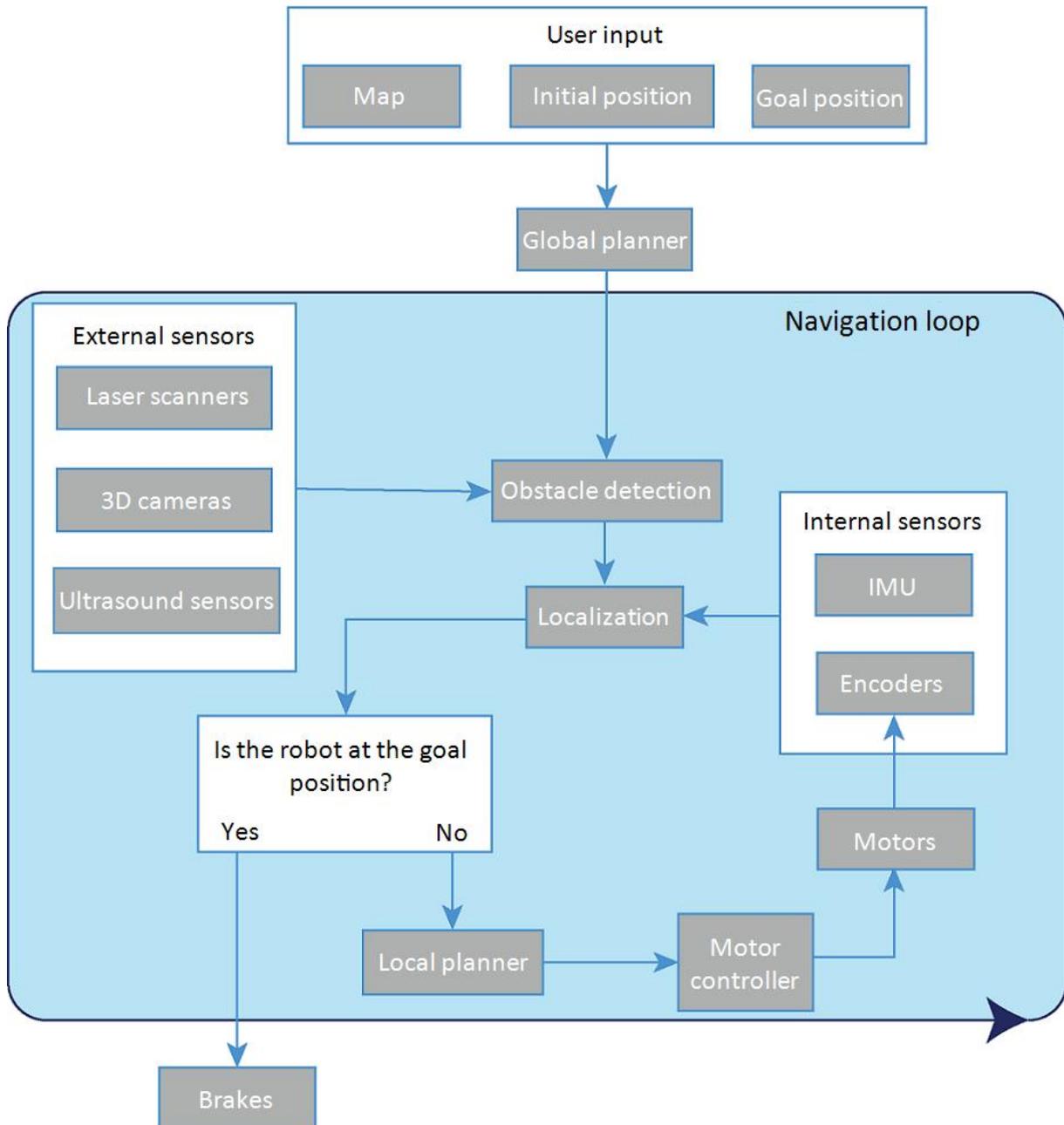


Figure 7.1. Flow chart of the navigation and control system. The user provides the necessary input for the robot to generate a path to the goal position. The robot executes the steps in the navigation loop until it reaches the goal position and stops by engaging the brakes.

## 7.2 User input

To enable the robot to navigate autonomously, you must provide the following:

- A map of the area, either from a .png file or created with the robot using the mapping function—see [Creating and configuring maps on page 92](#).
- A goal destination on that map—see [Markers on page 103](#).
- The current position of the robot on the map. This usually only needs to be provided when a new map is activated.

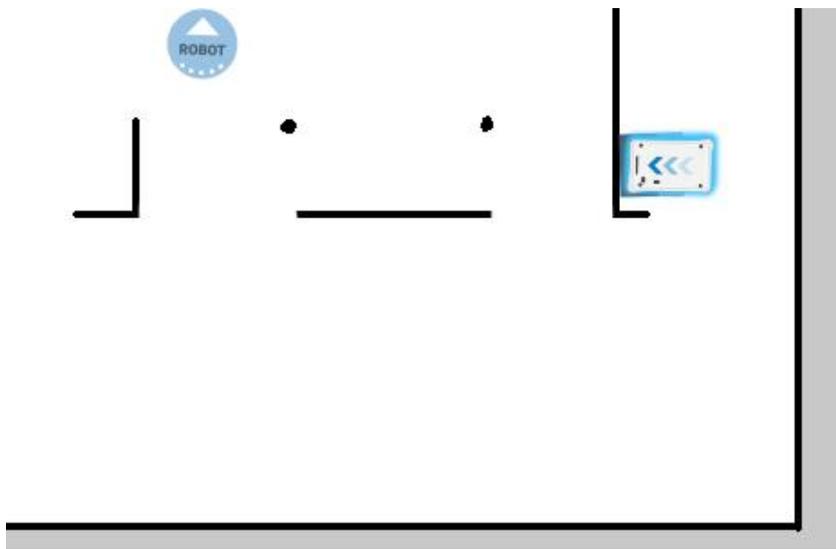


Figure 7.2. On the map, the current position of the robot is identified by the robot icon , and the goal destination in this example is the robot position . The robot computer now determines a path from the current position to the goal position.

Once the robot computer has a map with the robot's current position and a goal destination, it begins planning a route between the two positions on the map using the global planner.

## 7.3 Global planner

The global planner is an algorithm in the robot computer that generates a path to the goal position. This path is known as the global path.

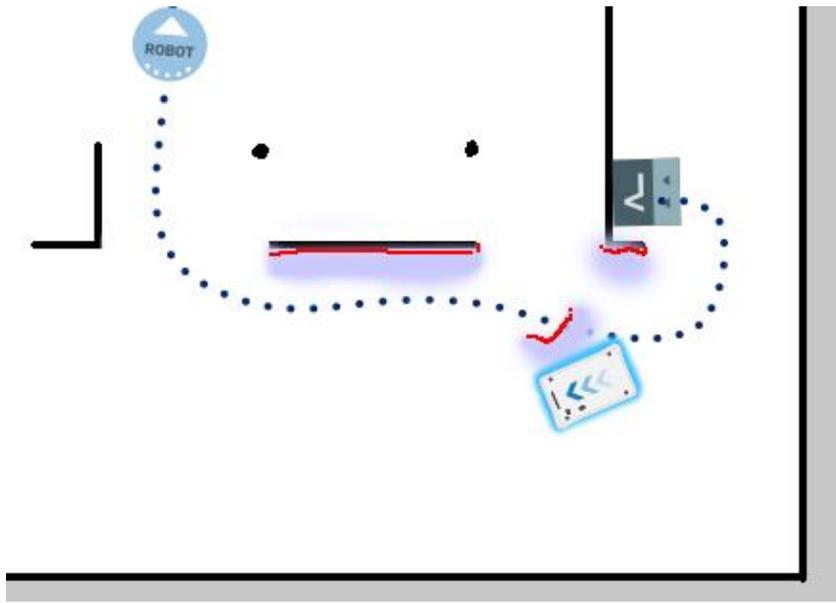


Figure 7.3. The global path is shown with the blue dotted line that leads from the start to the goal position.

The global path is created only at the start of a move action or if the robot has failed to reach the goal position and needs to create a new path. The generated path only avoids the obstacles the robot detected when the path was made and the obstacles marked on the map. The global path can be seen in the robot interface as a dotted line from the robot's start position to the goal position.

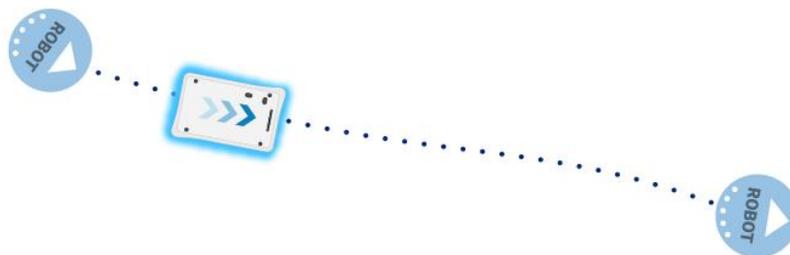


Figure 7.4. The dotted line from the start position of the robot to the goal position is the global path generated by the robot computer.

## 7.4 Local planner

The local planner is used continuously while the robot is driving to guide it around obstacles while still following the global path.

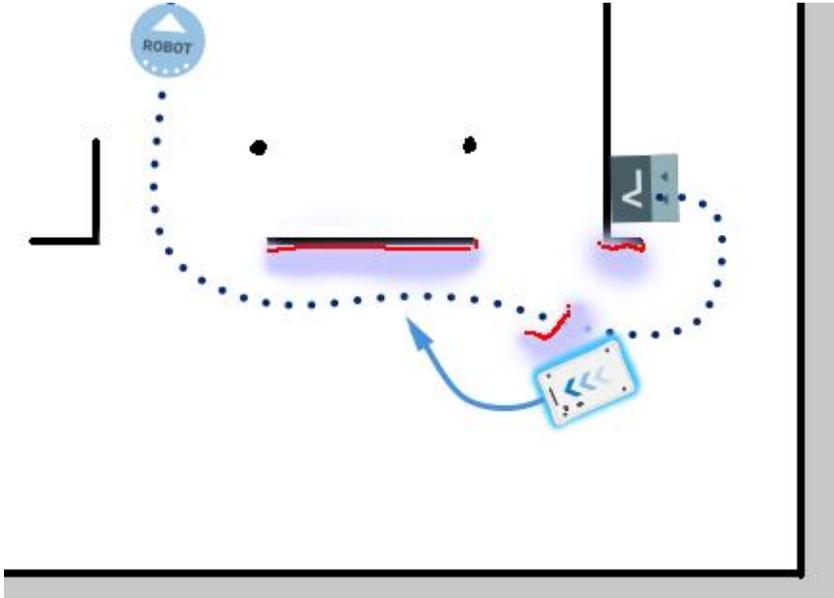


Figure 7.5. The global path is indicated with the dotted blue line and is visible on the map. The local path is indicated with the blue arrow, showing the robot driving around a dynamic obstacle.

Whereas the global planner creates a single path from start to finish, the local planner continues to create new paths that adapt to the current position of the robot and the obstacles around it. The local planner only processes the area that is immediately surrounding the robot, using input from the robot sensors to avoid obstacles.



The local path is not displayed in the robot interface. The arrows in the images here are visual aids used in this guide only.

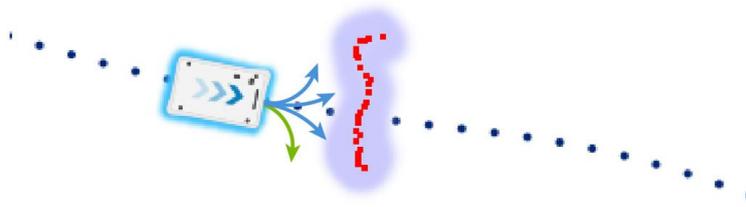


Figure 7.6. The local planner usually follows the global planner, but as soon as an obstacle gets in the way, the local planner determines which immediate path will get the robot around the obstacle. In this case, it will likely choose the path indicated with a green arrow.

Once the local path is determined, the robot computer derives the desired rotational velocity of each drive wheel to make the robot follow the local path and sends the desired velocities for each motor to the motor controllers—see [Motor controller and motors](#).

## 7.5 Obstacle detection

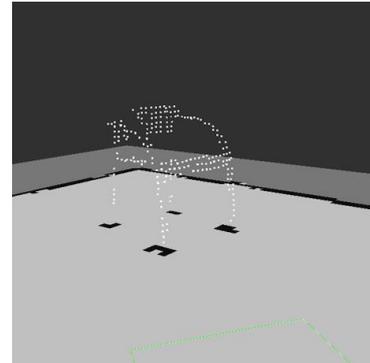
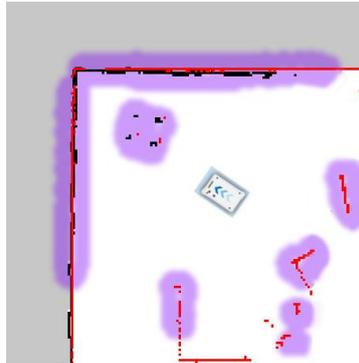
The robot detects obstacles continuously while driving. This enables the robot to use the local planner to drive around obstacles and to determine the robot's current position on the map.

Three sensor types are responsible for detecting obstacles:

- The safety laser scanners
- The 3D cameras
- The ultrasound sensors

**Table 7.1.**  
Description of how the robot sees obstacles with its sensors

What a human sees	What the laser scanners see	What the 3D cameras see
-------------------	-----------------------------	-------------------------



A chair placed in the corner of a room is detectable by the robot.

In the robot interface, the red lines on a map are obstacles detected by the laser scanners, and the purple clouds are an aggregate of the 3D camera and laser scanner data. The scanners only detect the four legs of the chair.

The 3D cameras detect more details of the chair when the robot gets close enough to it. This view cannot be seen in the robot interface.

## Safety laser scanners

Two safety laser scanners, diagonally placed on one front and one rear corner of the robot, scan their surroundings. Each safety laser scanner has a 270° field of view, overlapping and thus providing a full 360° visual protection around the robot.

When in motion, the safety laser scanners continuously scan the surroundings to detect objects.

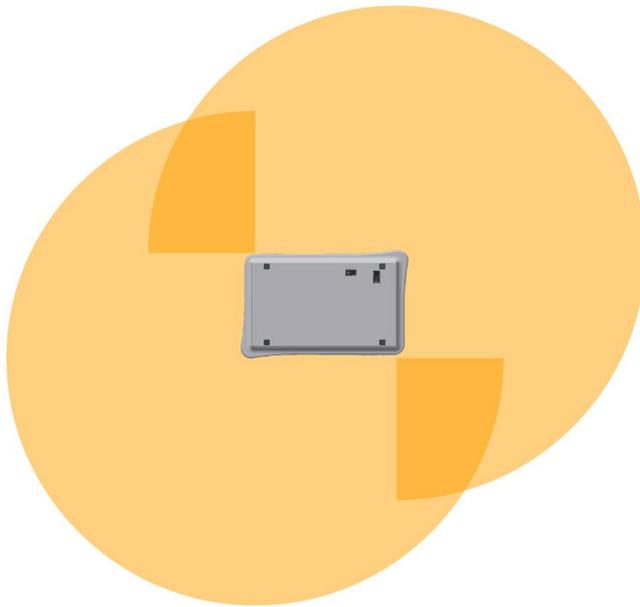


Figure 7.7. The two safety laser scanners together provide a full 360° view around the robot.

The laser scanners have the following limitations:

- They can only detect objects that intersect a plane at 200 mm height from the floor.
- They do not detect transparent obstacles well.
- The scanner data can be inaccurate when detecting reflective obstacles.
- The laser scanners may detect phantom obstacles if they are exposed to strong direct light.



If you are using the robot in an area with walls made of glass or reflective material, mark the walls as Forbidden zones on the map, not as walls—see [Creating and configuring maps on page 92](#). Walls on the map that the robot cannot detect will confuse the robot's navigation system.

## 3D cameras

Two 3D cameras positioned on the front of the robot detect objects in front of the robot. The 3D cameras detect objects:

- Vertically up to 1800 mm at a distance of 1950 mm in front of the robot.
- Horizontally in an angle of 118° and 180 mm to the first view of ground.

From the floor plane, the robot does not detect objects below 30 mm from the ground. This value increases by 10 mm per meter from the robot.

The 3D cameras are only used for navigation. They are not part of the robot's safety system.



### CAUTION

The 3D cameras cannot be used as a means for safety related risk reduction, for example, detecting low hanging obstacles. They cannot be used as an alternative to the Personnel detection safety function—see [Personnel detection on page 79](#). The robot may collide with and cause injury to personnel if you rely on the 3D cameras alone to detect personnel.

- Do not modify the safety laser scanner settings with the expectation that the 3D cameras will prevent collision with personnel.
- Do not include the 3D cameras as a risk reduction means in your risk assessment of the robot.



The camera readouts are used as 3D point cloud data. They are not recording recognizable objects or people.

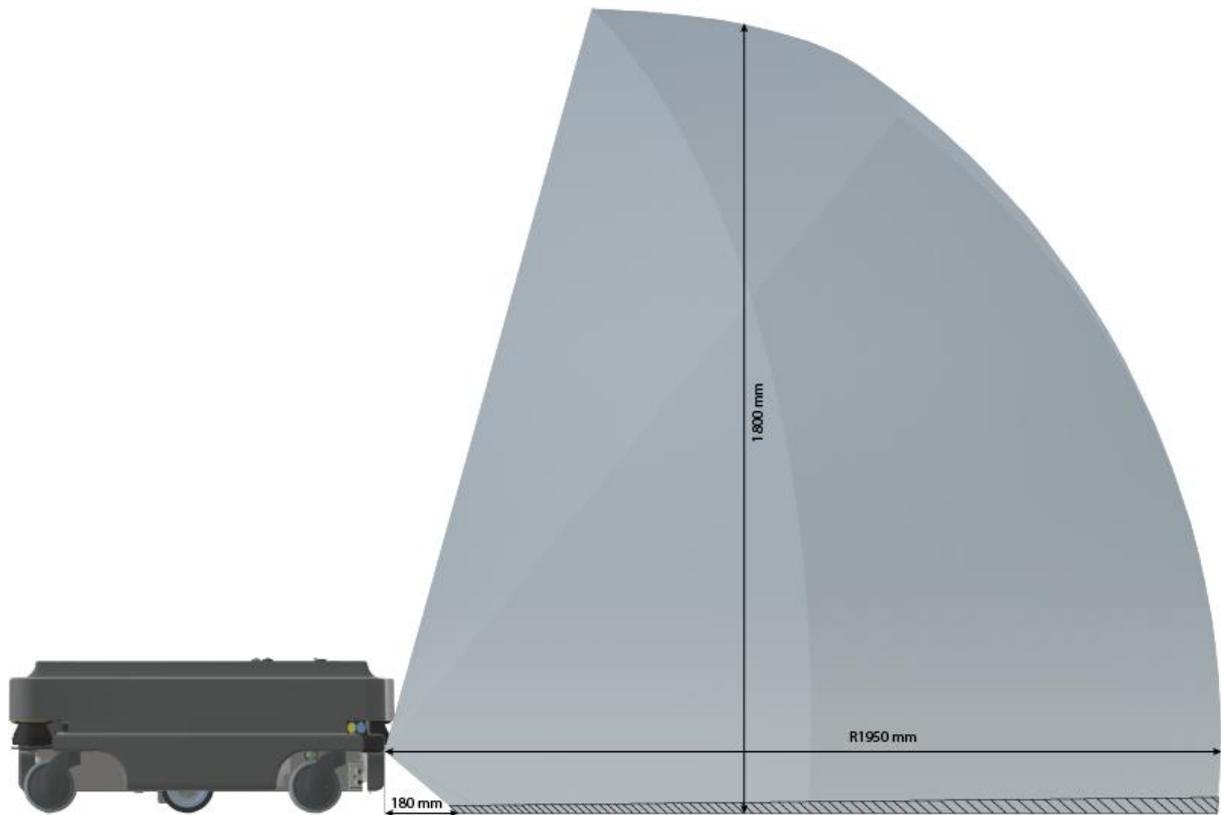


Figure 7.8. The two 3D cameras can see objects up to 1800 mm above floor height.



Figure 7.9. The two 3D cameras have a horizontal view of 118°.

The 3D cameras have the following limitations:

- They can only detect objects in front of the robot, unlike the full 360° view of the laser scanners.
- They do not detect transparent or reflective obstacles well.
- They do not detect holes or descending stairways.
- The cameras are not reliable at determining depth when viewing structures with repetitive patterns.
- The cameras may detect phantom obstacles if they are exposed to strong direct light.

## Ultrasound sensors

Four ultrasound sensors are placed on the robot: two at the front or rear and two at the front of the robot, but with an angle looking to the front sides.



Figure 7.10. Two ultrasound sensors are placed at the front of the robot (left) and two at the rear of the robot (right).

On robots where the last four digits of the 12-digit serial number are 1335 or lower, the ultrasound sensors are placed on the front and sides of the robot.

On robots where the last four digits of the 12-digit serial number are 1336 and up, the ultrasound sensors are placed on the rear and sides of the robot.

The ultrasound sensors are used to detect objects which cannot be seen by the camera or the laser scanners.

Table 7.2. Range of the ultrasound sensors		
Pos.	Minimum range	Maximum range
Front	10 mm	200 mm
Front side	200 mm	300 mm
Rear	10 mm	350 mm



Be aware that soft material such as foam or clothes can absorb sound and may not be detected by the sensors.

## 7.6 Localization

The goal of the localization process is for the robot to determine where it is currently located on its map. The robot has three inputs for determining where it is:

- The initial position of the robot. This is used as a reference point for the methods used to determine the robot position.
- The IMU and encoder data. This is used to determine how far and fast the robot has traveled from the initial position.
- The laser scanner data. This is used to determine the likely positions of the robot by comparing the data with nearby walls on the map.

This data is used by a particle filter to determine the most likely position of the robot on the map.

### IMU and motor encoders

Both the data from the IMU (Inertial Measurement Unit) and motor encoders is used to derive where and how fast the robot has traveled over time from its initial position. The combination of both sets of data makes the derived position more accurate.



If the drive wheels are worn down significantly—see [Maintenance on page 159](#)—or the robot is running with an incorrect gear ratio, the robot will miscalculate how far it has traveled based on the encoder data.

### Laser scanners and particle filtering

The robot computer compares the input from the laser scanners with the walls on the map to try and find the best match. This is done using a particle filter algorithm. The robot computer only compares input from the area where it expects the robot to be based on the encoder and IMU data. Therefore, it is important that the initial position of the robot is correct.

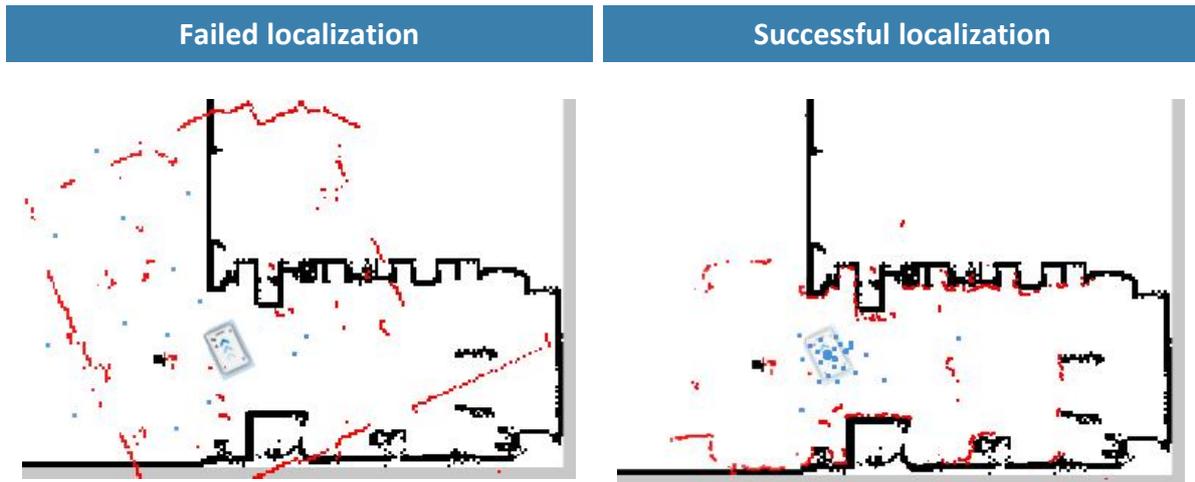
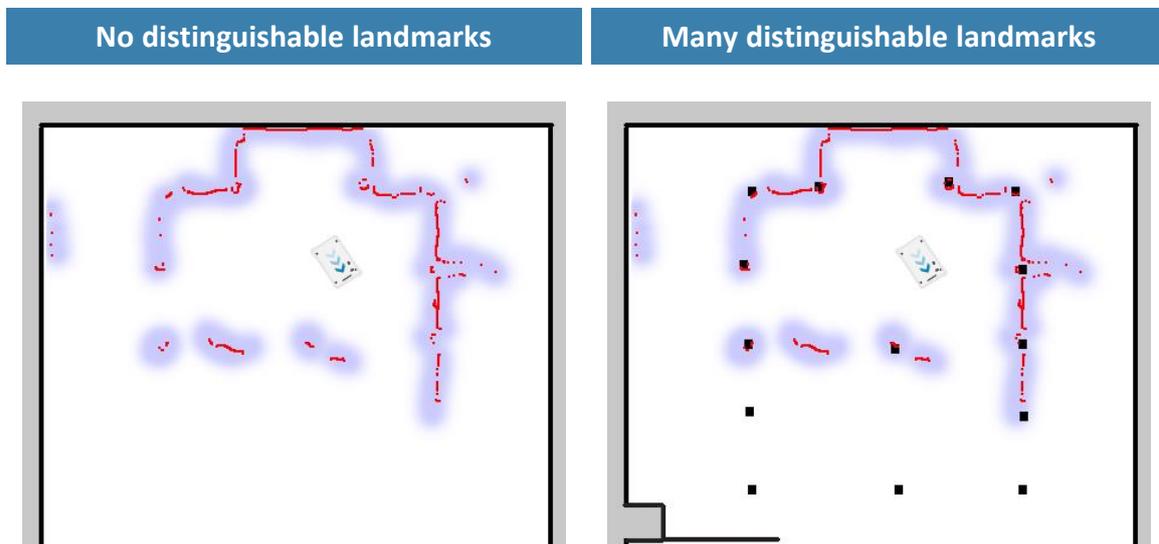


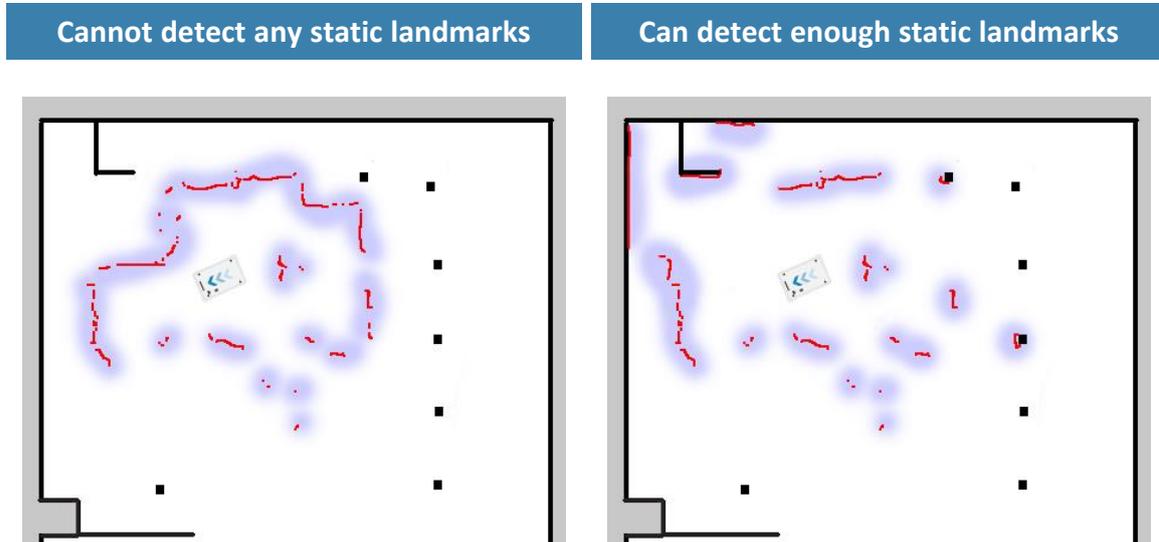
Figure 7.11. In a failed localization, the robot cannot determine a position where the red lines (laser scanner data) align with the black lines on the map. When the robot can localize itself, it determines a cluster of likely positions, indicated in the images above as blue dots.

To make sure the robot can localize itself well using particle filtering, consider the following when creating a map:

- There must be unique and distinguishable static landmarks on the map that are easily recognizable. A landmark is a permanent structure that the robot can use to orient itself, such as corners, doorways, columns, and shelves.



- The robot must be able to detect the static landmarks that are marked on the map to be able to approximate its current position. Make sure there are not too many dynamic obstacles around the robot so that it cannot detect any static landmarks.



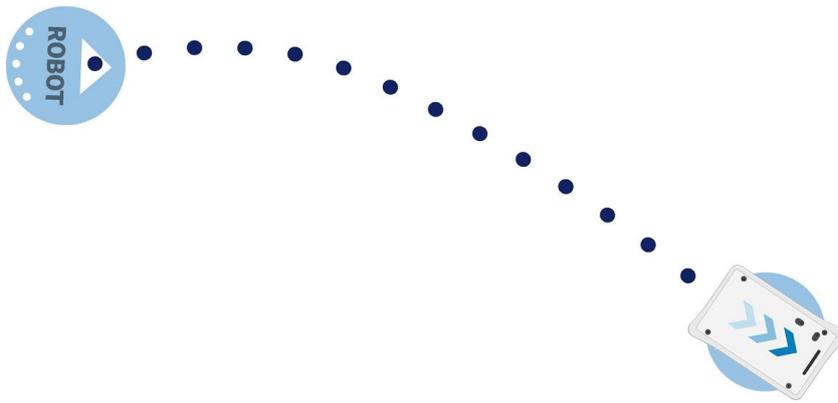
- The robot does not compare the laser scanner data with the entire map, but only around the area that it expects to be close to based on the IMU and encoder data and its initial position. This is why it is important that the initial position you place the robot at on the map is accurate.
- The robot can drive for a short distance without being correctly localized. As it drives, the estimated positions should converge to a small area, indicating the robot has determined an accurate estimate. If this does not occur within a set time limit, the robot reports a localization error.

## 7.7 Motor controller and motors

The robot keeps adjusting how much power is sent to each motor based on sensory input. This means that the robot can correct its speed when going up slopes or when carrying a heavier payload, and it can change its driving direction to avoid moving obstacles.

## 7.8 Brakes

Once the approximated position of the robot determined from localization is the same as the goal position calculated by the global planner, the robot stops by slowing down the motors.



# 8. Safety-related functions and interfaces

The robot's safety system is designed to mitigate significant hazards which could lead to injury, for example, stopping the robot if a person is in its path.

MiR100 is equipped with a range of built-in safety-related functions as well as safety-related electrical interfaces designed for integration with a top module. Each safety function and interface is designed according to the standard ISO 13849-1. The safety-related functions and interfaces are selected to support compliance with EN 1525 and ISO 3691-4.

## 8.1 System overview

The safety system is controlled mainly by the safety PLC. The PLC regulates inputs and outputs from safety-related functions or interfaces involved with ensuring the safety of personnel working nearby the robot.

If a safety function is triggered, the robot uses its relays to bring the robot to a category 0 stop (stopping by “immediate removal of power to the machine actuators” according to IEC 60204-1). This is known as bringing the robot into Emergency stop or Protective stop, depending on the function—see [Types of stop below](#).

There is also an Emergency stop electrical interface that enables you to connect any number of Emergency stop buttons to your robot. These Emergency stop buttons can, for example, be mounted to a top module. The Emergency stop buttons must be connected in series with two identical circuits to ensure redundancy.

### Types of stop

There are three different stopped states:

- Operational stop
- Protective stop
- Emergency stop

Protective stop and Emergency stop are monitored by the safety PLC.

## Operational stop

The robot is in Operational stop when it is stopped through the robot interface either through a mission action or by pausing the mission. The top module and all moving parts are still connected to a power supply.

## Protective stop

The robot enters Protective stop automatically to ensure the safety of nearby personnel. When the robot enters Protective stop, internal safety relays are switched so the robot's top module and all moving parts of the robot do not receive power. You can hear the safety relays emit audible clicks when they are switched.

When the robot is in Protective stop, the status lights of the robot turn red, and you are not able to move the robot or send it on missions until you bring the robot out of the Protective stop. The following cases describe the various Protective stops and how to bring the robot out of them:

- **A safety laser scanner detects an object in its active Protective field**  
Remove the object from the active Protective field—see [Personnel detection on page 79](#). The robot will resume its operating state after two seconds.
- **The robot finishes the startup process**  
The reset button on the Emergency stop will flash after startup. Press the reset button on the Emergency stop to bring the robot out of Protective stop.
- **The safety system detects a fault, or the motor control system detects a discrepancy**  
To bring the robot out of Protective stop, resolve the fault causing the error. Use information regarding the error from the robot interface to determine the fault. Go to [Monitoring > Hardware health](#) to find specific information on what caused the issue. For further guidance, contact your distributor for help troubleshooting the issue.

## Emergency stop

The robot enters Emergency stop when an Emergency stop button has been pressed physically. When you press the Emergency stop button, internal safety contactors are switched so the robot's top application and all moving parts of the robot do not receive power. You can hear the safety contactors emit audible clicks when they are switched.

When the robot is in Emergency stop, the status lights of the robot turn red, and you are not able to move the robot or send it on missions until you bring the robot out of the Emergency stop. To do this, you must press the flashing Emergency stop reset button. If the robot is in Emergency stop, it will immediately resume an operating state after you press the flashing Emergency stop reset button.



Figure 8.1. MiR100 has one Emergency stop button that must be connected through the electrical interface. You can also connect the interface to a series of additional Emergency stop buttons.



### CAUTION

Emergency stop buttons are not designed for frequent use. If a button has been used too many times, it may fail to stop the robot in an emergency situation, and nearby personnel may be injured by electrical hazards or collision with moving parts.

- Only press Emergency stop buttons in emergencies.
- Regularly check that all Emergency stop buttons are fully functional—see [Maintenance on page 159](#).
- Use the robot interface to stop the robot in non-emergency situations.

## Safety-related functions

The following functions are integrated within the robot itself and cannot be modified or used with other applications. The following list introduces the main safety-related functions integrated in MiR100:

- **Personnel detection**  
This function ensures that the robot stops before it collides with personnel or an object. If the laser scanners detect an object or person within a defined Protective field, the robot is brought to a stop. The function determines what the current speed of the robot is based on data from the motor encoders, and the function switches between predefined Protective fields accordingly. The faster the speed, the larger the Protective field is.
- **Overspeed avoidance**  
The safety system monitors if the motor encoder data indicates that the speed of each motor is above the limits for maximum speed. If a limit is exceeded, the robot enters Protective stop.
- **Stability**  
The safety system monitors if the motor encoder data indicates that the speed difference between the two motors are above predefined limits. If the limit is exceeded, the robot enters Protective stop.
- **Emergency stop circuit**  
The Emergency stop circuit goes through the Auxiliary emergency stop interface and connects to the top module. It is possible to connect multiple emergency stop buttons to the circuit. When the circuit is broken, the robot goes into Emergency stop.

These functions are described in further detail in the following sections.

The diagram in **Figure 8.2** shows the inputs to these functions and interfaces and how they are all connected and monitored by the safety PLC. The safety PLC is able to switch the safety contactors to cut off power to the robot motors and the top module whenever a Protective or Emergency stop is triggered. Also, the safety PLC sends information to the robot computer to be displayed in the robot interface (in **Monitoring > Hardware health**) and to indicate the robot's status through the status lights and the speaker.

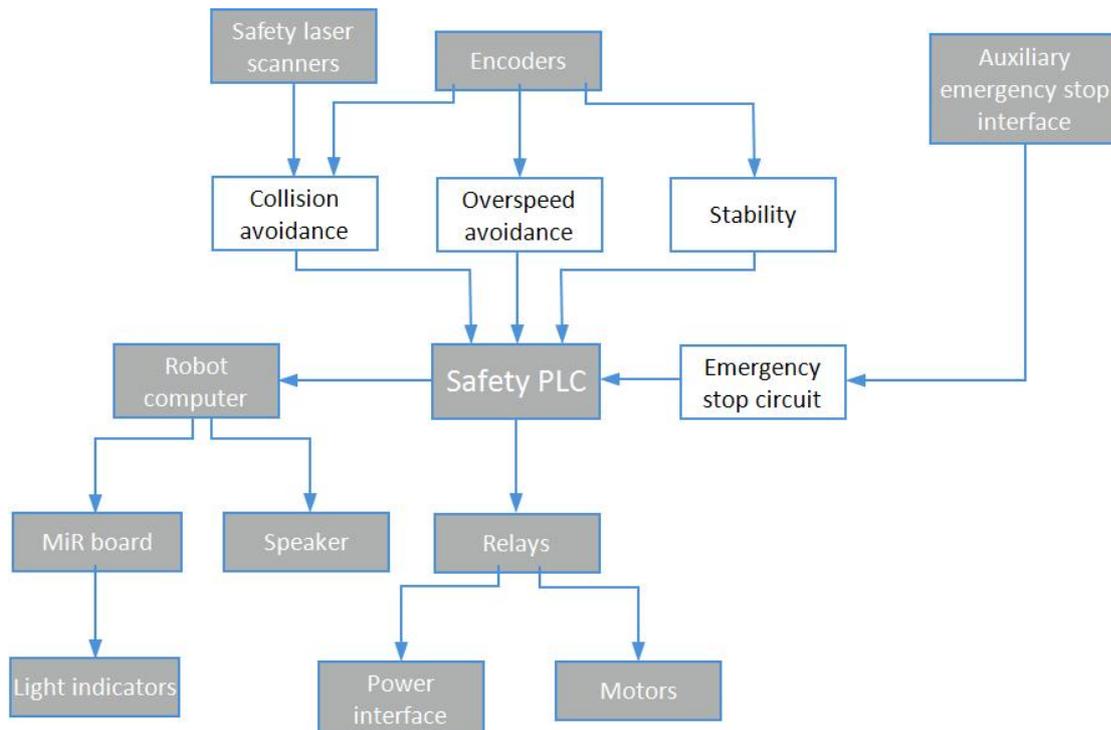


Figure 8.2. Overview of components involved in each safety function and interface. When a safety function is triggered, the safety PLC switches the STO and brake relays so the brakes, motors, and power supply to the top module (pin 3 in the Application interface) no longer receive power.

## 8.2 Personnel detection

The Personnel detection safety function prevents the robot from colliding with personnel or obstacles by stopping it before it collides with any detected obstacles. It does this using the safety laser scanners.

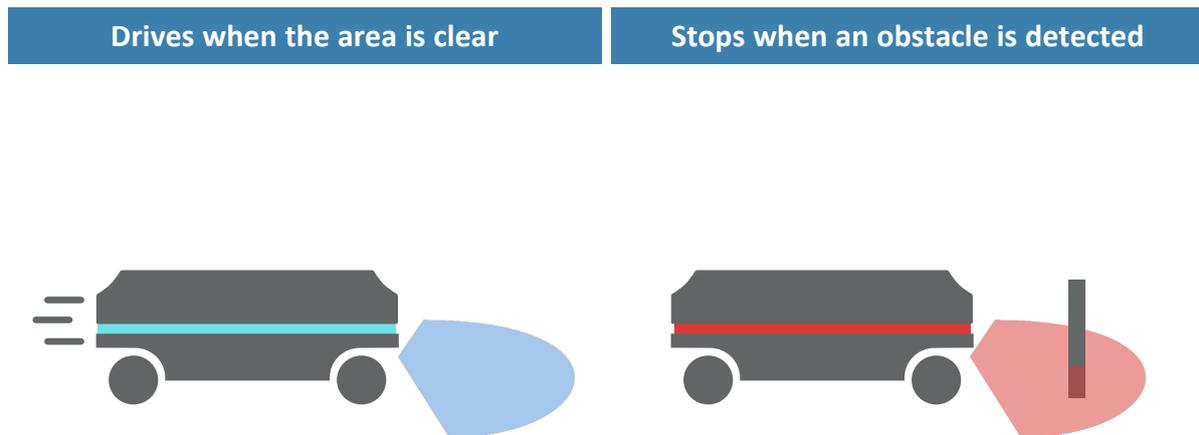


Figure 8.3. Personnel detection ensures that the robot drives when its path is clear and stops if an obstacle is detected within its Protective field.

The safety laser scanners are programmed with two sets of Protective fields. One field set is used when the robot is driving forward and the other when it is driving backward. The Protective field sets are part of the robot's Personnel detection safety function. Each Protective field in the sets is an individually configured contour around the robot. The robot activates the correct field based on the speed. If a person or object is detected within the active Protective field, the robot enters Protective stop until the Protective field is cleared of obstacles for at least two seconds.

The tables in the following sections show the sizes of the Protective fields at given speeds. The faster the robot moves, the larger the scanners' field is. The speed of the robot is determined based on the encoder data.

**WARNING**

The Protective field sets are configured to comply with the safety standards of MiR100. Modifications may prevent the robot from stopping in time to avoid collision with personnel and equipment. Any modifications of the SICK configuration requires a new CE certification of the robot and compliance to all safety standards listed in the specification of the application and in other way declared.

- Do not modify the safety system without a competent third party to evaluate the safety of the design and performance of the robot after the modifications are applied.

## Field set when driving forward

The following table shows speeds and the field range when driving forward. The table describes the length of the Protective field in front of the robot in different cases. Each case is defined by a speed interval that the robot may operate at. The colors and cases in **Table 8.1** correspond to the field set shown in **Figure 8.4**.

Table 8.1. Range of the robot's Protective fields within its forward speed interval cases.			
Case	Speed	Protective field range	Comments
1	-1.40 to 0.20 m/s	0-20 mm	Reversing, standstill, and slowly forwards
2	0.21 to 0.40 m/s	0-120 mm	
3	0.41 to 0.80 m/s	0-290 mm	
4	0.81 to 1.10 m/s	0-430 mm	
5	1.11 to 2.00 m/s	0-720 mm	Forwards at maximum speed

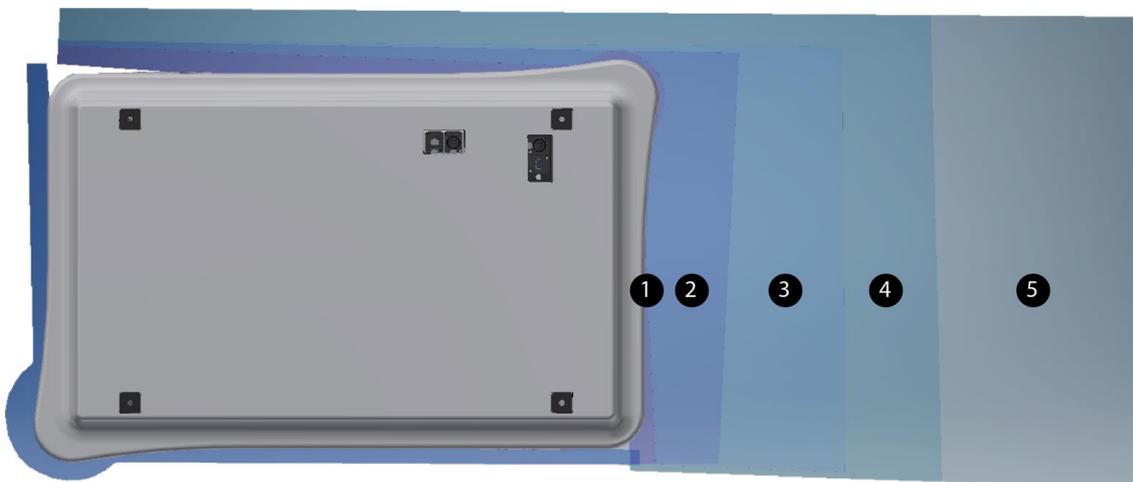


Figure 8.4. The illustration shows the field set contours when the robot drives forward. The range of the active field changes with the robot's speed.

## Field set when driving backward

The field set for driving backward is the same as the field set for driving forward. The colors and cases in **Table 8.2** correspond to the field set shown in **Figure 8.5**.

Table 8.2. Range of the robot's Protective fields within its backward speed interval cases.			
Case	Speed	Protective field range	Comments
1	-0.14 to 1.80 m/s	0-30 mm	Forward, standstill and slowly backwards
2	-0.14 to -0.20 m/s	0-120 mm	
3	-0.20 to -0.40 m/s	0-290 mm	
4	-1.40 to -1.50 m/s	0-430 mm	Backward at max. speed



Figure 8.5. The illustration shows the field set contours when driving backward. The range of the active field changes with the robot's speed. The illustration also shows how the front scanner reduces its Protective field to a minimum when the robot moves backward.



**NOTICE**

Scanners measure distances to diffuse reflections, which means that a tolerance is added to the Protective field sets to secure a safe detection of persons crossing the Protective field sets. The tolerance distance is 100 mm.

## 8.3 Overspeed avoidance

## 8.4 Stability

The stability function prevents the robot from driving if the motor encoders measure that the expected difference between how fast each wheel turns is outside the predefined safety limits. This indicates that the robot is not driving as intended, for example, if one of the wheels loses traction.

If the robot detects instability, it is immediately brought into Protective stop. This ensures that the robot cannot drive if it has lost control of the speed of each drive wheel.

## 8.5 Emergency stop circuit

The Emergency stop circuit goes through the Auxiliary emergency stop interface and uses external input to bring the robot into an Emergency stop. The interface uses two output pins to provide a 24 V signal and two input pins to bring the robot into Emergency stop.

It is intended that the circuit is set up so the 24 V signal delivered from the safety PLC outputs passes through all Emergency stop buttons of the top module and then continues to the two input pins. When the input pins both receive 24 V, the robot can operate. The connected Emergency stop buttons must break the circuit when you press them so both inputs receive a 0 V signal that will bring the robot into Emergency stop.

If the circuit or an Emergency stop button is installed incorrectly so the input signals are not the same, the robot enters Protective stop until the circuit is fixed.

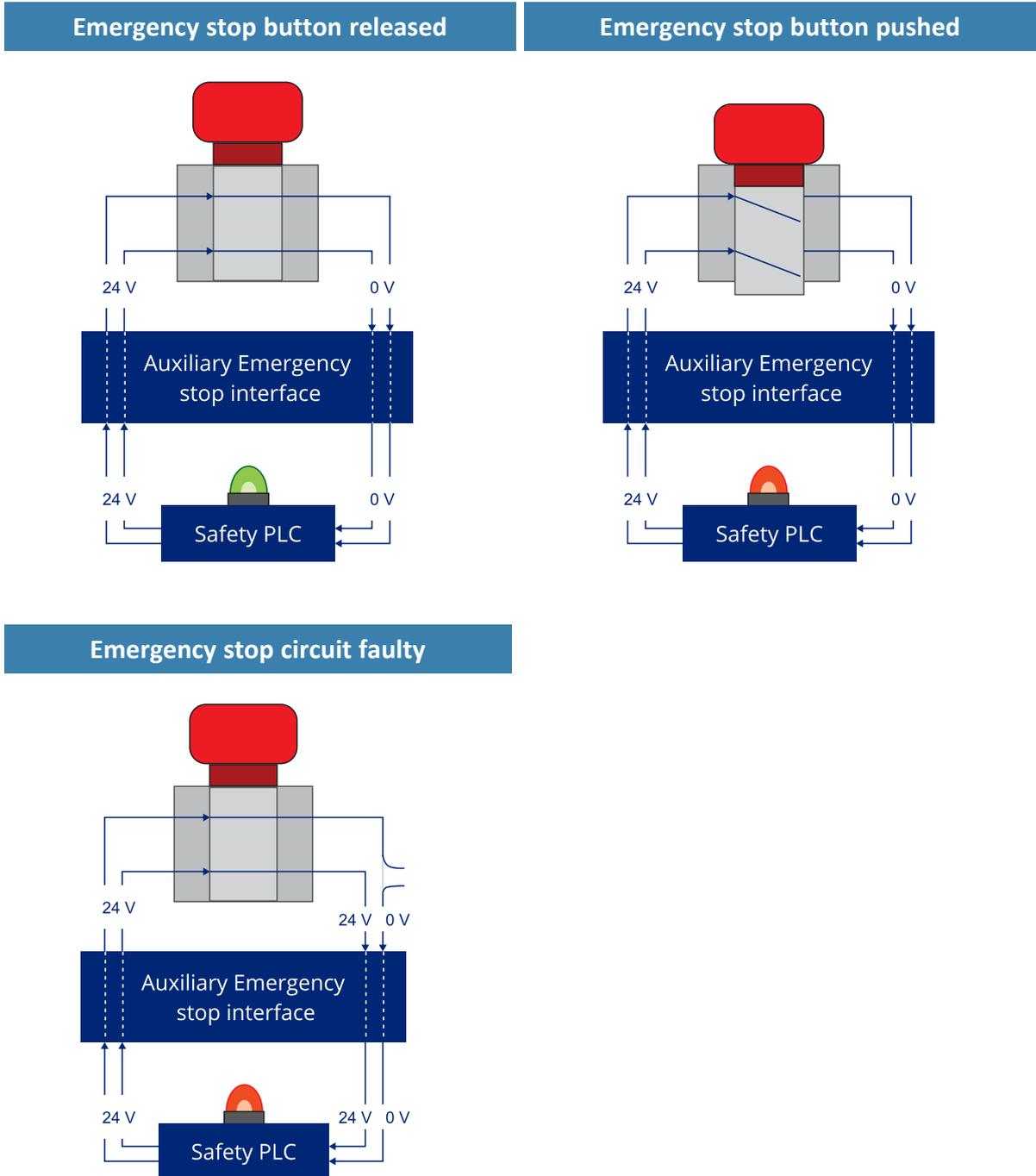


Figure 8.6. If the input pins deliver 24 V to the robot, it can operate. When you push a connected Emergency stop button, both pins deliver 0 V, and the robot enters Emergency stop. If the pins do not deliver the same input, the robot enters Protective stop until the circuits are fixed.

## 8.6 Robot computer

The robot computer is connected to the safety PLC via an Ethernet cable. The safety PLC sends all of the statuses of its various inputs to the robot computer so the information can be sent to the robot interface. This enables you to identify which part of the safety system may be causing a Protective or Emergency stop.

Additionally, the robot computer sends the current robot state to the MiR board, which regulates the status lights making them indicate which state the robot is in.

## 8.7 Light indicators and speakers

The robot uses its status lights and speaker to let people in the environment know what the robot is currently doing or planning to do.

### Status lights

The LED light bands running all the way around the robot indicate the robot's current operational state. Colors may also be used as part of missions, but as standard, status lights indicate the statuses described in **Table 8.3**.

Table 8.3. Status light colors	
Red	Emergency stop
Green	Ready for job
Cyan	Drives to destination
Purple	Goal/Path blocked
White	Planning/Calculating
Yellow	Mission paused
Yellow wavering	Startup signal before PC is active
Yellow fade	Shutting down robot
Yellow blinking	Relative move, ignoring obstacles

Purple - yellow	General error, for example hardware, localization
Blue	Manual drive
Blue wavering	Mapping
Contracting white	Charging at charging station
White wavering	Prompt user / Waiting for user's response
Cyan wavering (robots connected to MiR Fleet only)	Waiting for MiR Fleet resource or for another MiR robot to move



When the robot's battery reaches a critically low level of power (0-1%), the ends of the status lights flash red.



When the robot is charging in a charging station, the status lights on the side of the robot indicate the robot's battery percentage.

## Speakers

In **Setup > Sounds**, you can upload new sounds to the robot or edit the volume and length of the default sounds.

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

**CAUTION**

Unaware personnel may not see the robot in certain situations and risk colliding with the robot. This may result in injury to personnel or damage to equipment.

- Make sure to adjust the volume of the robot's warning sounds so they are audible in the robot's work environment.
- Implement warning sounds from the robot in missions and areas where it can reduce the risk of hazardous situations.

# 9. Commissioning

This section describes how to commission MiR100.

Commissioning should be done without any load on the robot, except when doing brake tests where the robot should have a load equaling the heaviest load it will be driving with.

Only persons assigned with the commissioning task should be present during commissioning.

It is the responsibility of the commissioner to:

- Analyze the work environment.
- Make a risk assessment of the full installation.
- Create and configure the site with maps.
- Create missions.
- Configure audio and light signals according to the environment.
- Change the robot's footprint or create new footprints depending on the top module or loads of the robot.
- Create operating hazard zones.
- Make a brake test.
- Create user groups and users.
- Create dashboards.
- Update robot software.
- Change the relevant system settings.

## 9.1 Analysis of the work environment

The work environment of the robot must fulfill a number of requirements for the robot to function properly and safely. This section describes the factors that must be considered when the robot is being commissioned to function in a work environment.

### Surfaces

The floor surface of the work environment must be dry. MiR100 functions on many different types of surfaces, but some materials can affect the performance and safety of the robot, such as very thick carpets or slippery floors.

It is the responsibility of the commissioner to test the performance and safety of the robot on the surfaces in the work environment—see [Making a brake test on page 119](#).

## Light, reflections, and materials

Bright sunlight and reflective or transparent objects can affect the performance of the robot's laser scanners and cameras. This can result in the robot detecting nonexistent objects or failing to detect real objects.

Likewise, docking to markers made in very high gloss or transparent materials can reduce the effectiveness of the robot's scanners, hindering a successful docking.

It is the responsibility of the commissioner to test if sunlight, reflections from high gloss materials, and transparent objects affect the robot's performance or safety.

## Temperature and humidity

Temperatures outside of the approved temperature range can affect the performance and durability of the robot—see specifications on the MiR website. This is especially relevant for the robot's battery—see [Battery storage on page 56](#).

## Inclines, doorways, gaps, and sills

The robot must operate within the approved specifications for driving on inclines, through doorways, and over gaps and sills—see specifications on the MiR website. Operating in areas that do not meet the specifications may result in the robot failing to complete the missions or losing control of its load.

## Space

The robot must have sufficient space to operate efficiently. Determine during commissioning if the robot has sufficient space to drive, dock, turn, and perform other tasks. Make sure to test each mission under the most likely operating conditions to determine if there is enough space for the robot to maneuver.



### CAUTION

Driving in environments with no escape routes can lead to hazardous situations. There is a risk of damage to equipment or injury to personnel.

- At all times there must be at least 0.5 m wide and 2.1 m high pedestrian escape routes on each side of the robot, also in operating hazard zones.

## Dust

Dusty environments can affect the performance and durability of the robot. Dust can get into the robot computer and mechanical parts, affecting their performance and durability, and it can obstruct the view of the robot's sensor system. Make sure the environment MiR100 operates in is suitable for its IP rating—see specifications on the MiR website.

## Static landmarks and dynamic obstacles

The robot uses static landmarks to navigate by. If it cannot detect enough distinguishing landmarks, it cannot navigate the map efficiently—see [Localization on page 71](#).

## 9.2 Risk assessment

To achieve a safe installation, it is necessary to conduct a risk assessment of MiR100 in the environment it will be used in. This is the responsibility of the commissioner.

The risk assessment must cover both MiR100 itself and also take into account carts, potential load transfer stations, work cells, and the work environment.



### NOTICE

Mobile Industrial Robots takes no responsibility for the creation and performance of the risk assessment, but we provide information and guidelines that may be used in this section.

It is recommended that the commissioner follows the guidelines in ISO 12100, EN ISO3691-4, EN 1525, ANSI B56.5, or other relevant standards to conduct the risk assessment.

In EN 1525 clause 4 there is a list of possible significant hazards and hazardous situations that the commissioner should consider.

A risk assessment of the application must be used to determine the adequate information for users. Special attention to at least the following Essential Health and Safety Requirements (EHSR) from Directive 2006/42/EC must be taken:

- 1.2.2 Control devices
- 1.3.7 Risk related to moving parts
- 1.7.1 Information and warning on the machinery
- 1.7.2 Warning of residual risks
- 1.7.3 Marking of the machinery
- 1.7.4 Instructions

The risk assessment will lead to new instructions that shall be written by the party who draw up the CE marking. The instructions must at least include:

- Intended use and foreseeable misuse.
- A list of residual risks.
- Training required for personnel.

## 9.3 Creating and configuring maps

The map is visible in the robot interface and is the basis for the robot's ability to navigate its surroundings safely and efficiently. The map illustrates the physical area in which the robot operates.

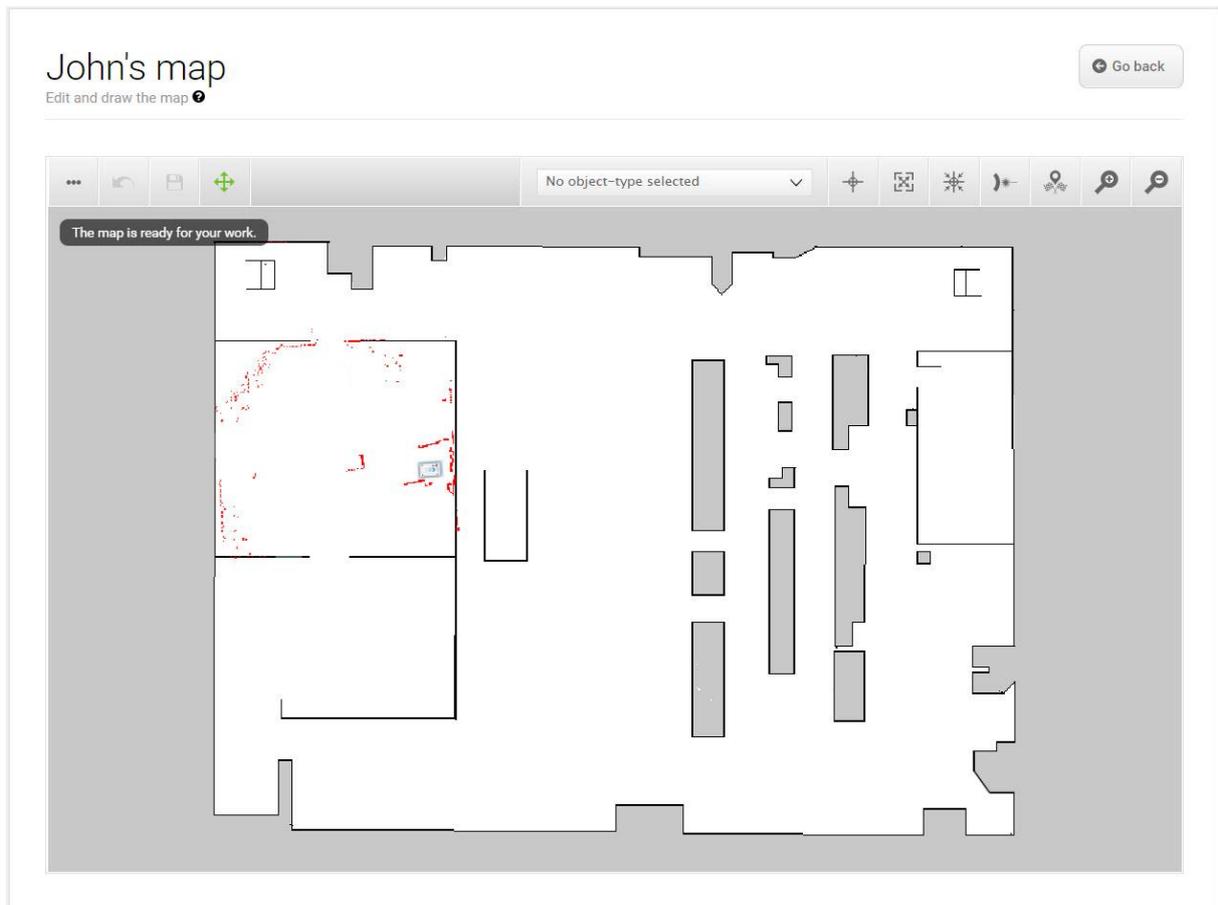


Figure 9.1. Example of a map without any added zones, positions, or markers.

The robot must have a map for every area that it operates in. It is important to create robust and reliable maps for the robot to perform effectively and safely.

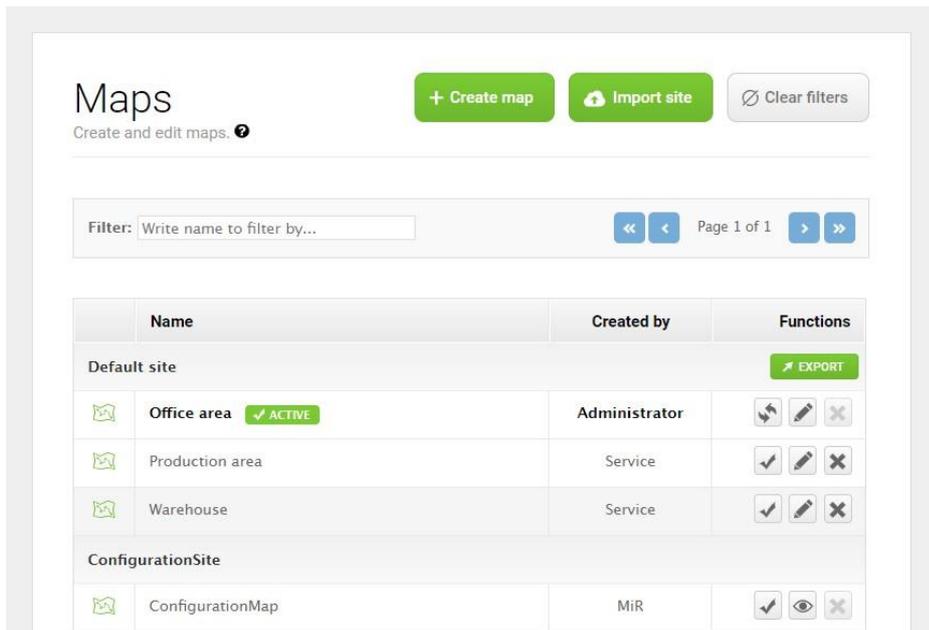


Figure 9.2. The **Default site** has three maps within it for the areas in the site: **Office area**, **Production area**, and **Warehouse**.

A site contains one or more maps that often connect to each other. The number of maps you need in a site depends on the work environment of the robot:

- If the operating area is very large, you may need to split the area into smaller maps.
  - You can tell that a map is too large if the robot takes a long time to plan its routes or often reports CPU errors.
  - In general, we recommend that maps should not exceed an area of 300 x 300 meters.
  - You can connect maps using map transitions—see *MiR Robot Reference Guide*, or ask your distributor for the guide *How to set up transitions between maps*.
- If the robot must operate on different floors connected with ramps or elevators, you must have a map for each floor.
  - If you are using an elevator, ask your distributor for the guide *How to set up elevators in MiR Fleet*.
  - If you are using ramps, connect the maps using transitions—see *MiR Robot Reference Guide*, or ask your distributor for the guide *How to set up transitions between maps*.



Each site also includes other elements in the interface, such as missions. For the full list of what is included in a site, see *MiR Robot Reference Guide* on the MiR website.

## Creating a map

To create a new map, you drive the robot around its intended work environment while its sensors gather data to generate a map from. This process is known as mapping.

As the robot moves during mapping, the laser scanners detect physical obstacles, which are recorded on the map as walls. In the editing afterward, you can remove all obstacles that should not stay on the map, for example carts or boxes that were present at the time of recording but will not stay permanently.

Before you map a new location, be sure to do the following preparations:

- Clear the area of dynamic obstacles, such as pallets and carts. Dynamic obstacles can also be deleted from the map later.
- Ensure that all doors and gates that the robot should be able to go through are opened before mapping.

Avoid doing the following:

- Starting the mapping with the robot in a very open space.
- Getting the robot stuck close to walls or objects as you will have to push it away manually.

To create a new map, see *MiR Robot Reference Guide* on the MiR website. When mapping, you should apply the following best practices:

- Focus on mapping in a circular pattern around the perimeter of the working environment.
- When reaching long corridors with few obstacles, let the robot stay in position for approximately five seconds before moving down the corridor.
- Walk behind the robot as you map.
- End the mapping in the same place you started it.



For more information on creating a map, see the *Creating your first map-* course in MiR Academy on the MiR website. Contact your distributor for access to MiR Academy.

## Cleaning up a map

The robot navigates best when using a clean map with as little noise as possible. **Figure 9.3** is an example of what a map can look like after the mapping process but where it still needs further editing.



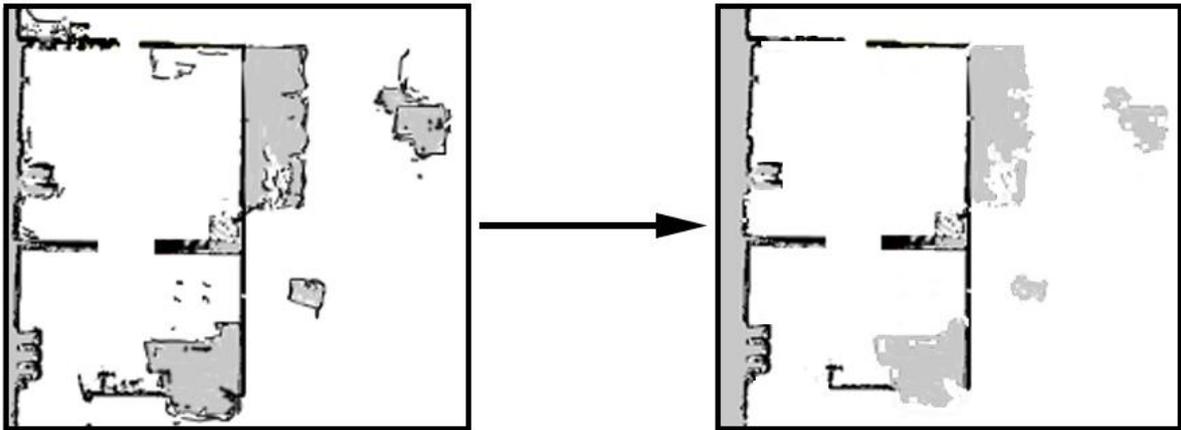
Figure 9.3. Example of a map that includes too much noise and dynamic obstacles.

There are several tools in the robot interface that you can use to improve your map:

- Use **Erase uploaded or recorded data**  when editing walls to remove walls that were created around dynamic obstacles and noise on the map.



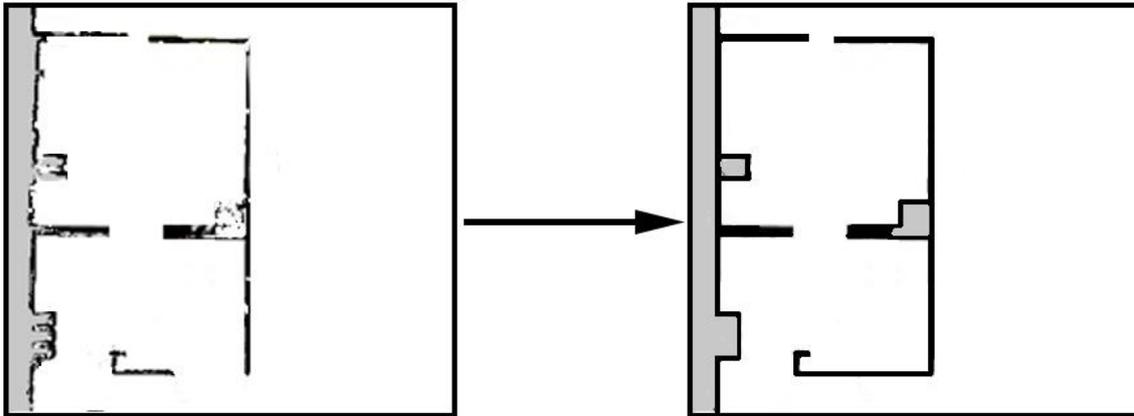
Noise refers to recorded data that originates from interfering elements. This can be physical obstacles that make the robot record walls where there are none or more subtle interferences that can make recorded walls appear pixelated.



- Use **Draw a new shape**  when editing floors to fill out the gray areas where there should be floor. When using this tool, you do not affect the walls on the map.



- Use **Draw a new line**  when editing walls to create solid and even walls.



## Adding zones to the map

Adding zones to the map helps organize robot traffic. There are several different zones that can optimize the preferred paths and driving behavior of the robot.



For more information about what each zone does, see *MiR Robot Reference Guide* on the MiR website, or ask your distributor for the guide *How to use zones on a map*.



### NOTICE

All zones are ignored when you drive the robot in Manual mode or when you use a Relative move action (except when using Relative move actions in Limit-robots zones).



### NOTICE

The robot prioritizes the instructions it receives in the following order:

1. Instructions from zones.
2. Instructions from mission actions.
3. Instructions from system settings (except from the setting Maximum allowed speed, which is never overruled by neither zones nor mission actions).

## Examples of when and how to use zones

The following sections describe examples of cases where certain zones can be used to improve the robot's operations.



For more examples, contact your distributor for the guide *How to use zones on a map*.

### Descending staircases

**Issue:** The robot sensors cannot detect descending staircases. Marking a staircase as a wall on the map will only confuse the robot as it will try to navigate from a wall that is not there.

**Solution:** Mark staircases and areas surrounding staircases or holes in the floor as Forbidden zones on the map.

### Low hanging fixtures

**Issue:** If a low hanging fixture is outside of the robot sensors' range, the robot may try to travel beneath it. This can be dangerous if the robot is carrying a tall top module or load that can collide with the fixture.

**Solution:** Mark the area where the low hanging fixture is located as a Forbidden zone.

## Highly dynamic areas

A highly dynamic area is an area where objects are moved frequently. This could be a production area where pallets and boxes are often moved back and forth.

**Issue:** The robot will stop if a person steps out in front of it. In a transient work flow area, the robot will stop and reassess its paths many times a day, thereby wasting valuable time.

**Solution:** Mark highly dynamic areas on the map with Unpreferred zones (blue) or Forbidden zones (red) depending on the environment. Directional zones can also be used here to guide the robot in a specific direction.

If the robot has trouble with localization in a highly dynamic area, place some static objects with three meters of distance between them and mark them as walls on the map. Remove the 'walls' created from dynamic obstacles in the area. Static objects make it easier for the robot to localize and navigate the area.

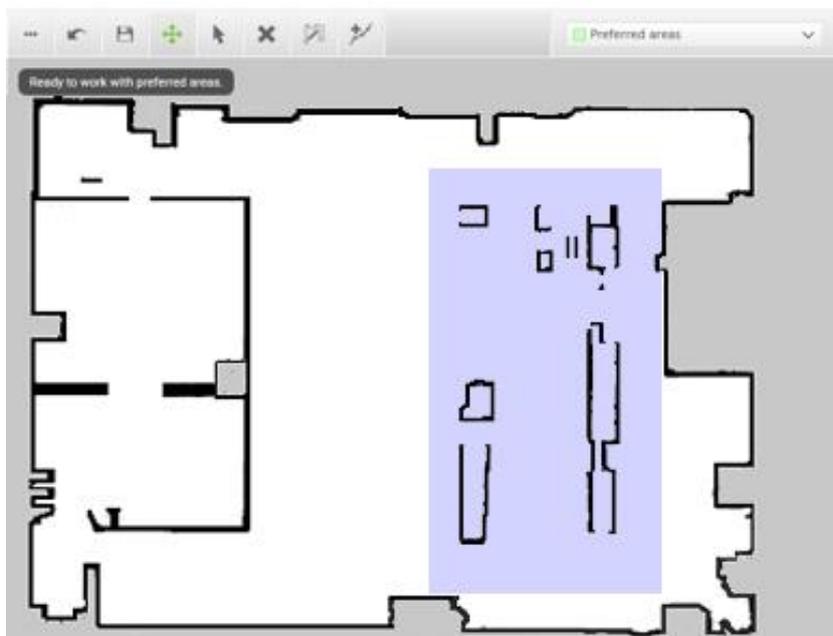


Figure 9.4. Unpreferred zones (marked with purple) can be used in highly dynamic areas to solve issues with replanning of paths.

## Doorways

Going through narrow doorways can cause problems for the robot's global planner since the robot must drive closer to wall edges than it usually would. It can also be hazardous for the people working near the robot, as they might not see the robot coming.

**Issue:** The robot does not plan its global path through narrow doorways, since this will bring the robot too close to a known obstacle.

**Solution:** Add a Critical zone (orange) in the narrow doorway to enable the global planner to make a path through the corridor. You only need to place the zone down the center of the doorway so the center of the robot is in the zone. Add Sound and light zones (yellow) in narrow doorways to warn people near the doorway that the robot is coming through.

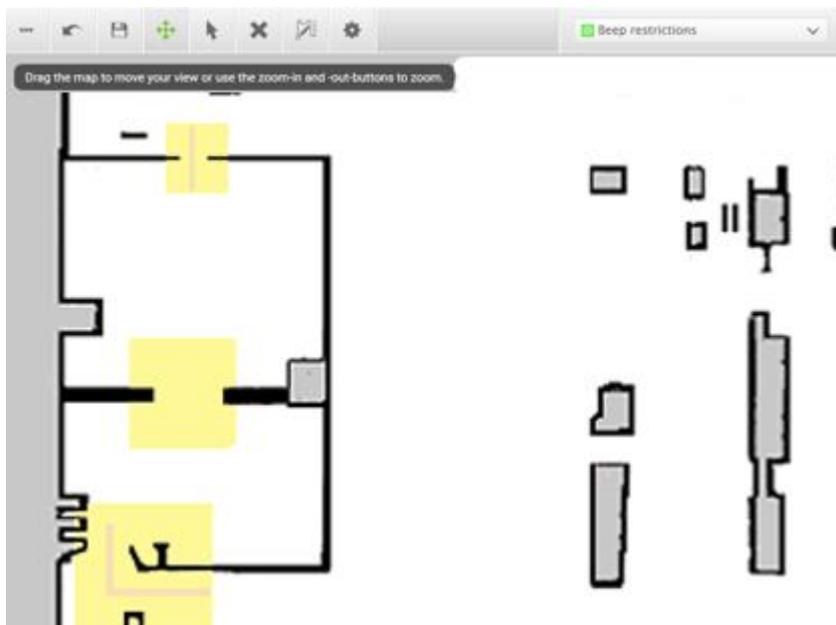


Figure 9.5. Narrow doorways can be marked with a Sound and light zone (marked with yellow) to warn people that a robot is coming through. A Critical zone (orange) can be placed in the narrow doorway to enable the global planner to make a path through the corridor.

## Shelves

Shelves are often placed in a certain height above the floor on four (or more) posts and will often appear as dots on a map for the robot. This may cause the robot to believe that there is enough space (if the posts are far enough apart) below the shelves to pass through. The robot will then plan a path underneath the shelves, but when it comes closer, the camera will see the obstacle. This could result in replanning paths several times a day.

**Issue:** The robot will only see shelves as dots on the map and believe that it can make a global plan underneath the shelves.

**Solution:** Add a Forbidden zone (red) around the shelves.

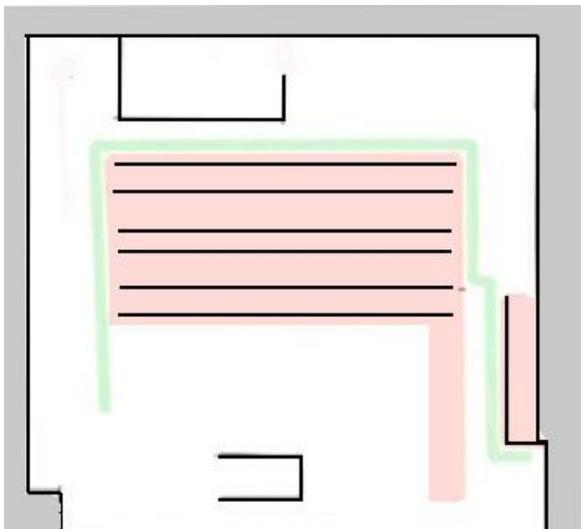


Figure 9.6. A Forbidden zone covering the shelf area.

## Glass

Highly transparent glass may not be detected by the safety laser scanners.

**Issue:** The robot will not stop before driving into a glass window, door, or other glass objects.

**Solution:** Make the glass visible to the safety laser scanners by gluing non-transparent window film on the glass in the scanner height, 150 to 250 mm, or mark the wall as a Forbidden zone. Edit the map afterwards in the robot interface and mark the glass as walls to help the robot localize.

## Directional lanes

**Issue:** In some areas, such as long corridors, robots driving towards each other may have a hard time passing each other efficiently.

**Solution:** If there is not enough space for the two robots to pass each other, you can create a two-way lane using Directional zones in combination with Forbidden or Unpreferred zones.

- Create a thin Forbidden zone (red) in the middle of the corridor parallel to the corridor walls. This is the lane separator.
- Create Directional zones (gray with arrows) on both sides of the Forbidden zone. Make the directions of the zones opposite.

With such a configuration, robots going in the opposite directions use different lanes and do not get in each others' way. Replacing the Forbidden zone with an Unpreferred zone gives robots more space for maneuvers, for example, if a robot needs to cross the lane separator to drive around an obstacle.



Figure 9.7. The robot drives down a two-way lane. The two Directional zone lanes are separated by a Forbidden zone.

If there isn't enough space for robots to pass each other, you can use a Limit-robots zone to specify that only one robot may drive down the corridor at a time.



To use Limit-robots zones, your robots must be connected to MiR Fleet.

## 9.4 Markers

Markers are defined as X-Y coordinates on a map that mark locations where you want the robot to travel to. Markers are points on the map that mark a physical entity, such as a charging station or a pallet rack, and enable the robot to position itself accurately relative to

this entity.

You should always use markers when it is important that the robot is positioned accurately relative to an object in the work environment, such as load transfer stations and work stations.

## Docking to markers

Markers require the robot to do a docking sequence. When the robot is docking, it uses its safety laser scanners to detect the marker and drives itself to the correct position relative to the detected marker. The robot begins docking to a marker from the marker's Entry position—see **Figure 9.8**. The Entry position is automatically created approximately one meter in front of the marker and can be moved in the map editor. The Entry position should not be move so far from the marker that the robot can no longer detect the marker.

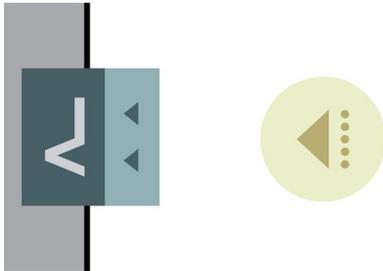


Figure 9.8. A VL-marker with its Entry position.



### CAUTION

When a robot docks to a marker (except L-markers), it mutes its Protective fields and disables Collision avoidance, enabling it to drive closer to the marker. If an object is placed or personnel walk in front of the robot while these functions are disabled, the robot will continue docking until it collides with the obstacle.

- Mark the area around the marker as an operating hazard zone, and inform personnel not to walk in front of the robot while it is docking—see [Using operating hazard zones](#)**REF-FAIL**.
- Do not place objects directly in front of a robot marker.
- By default, Collision avoidance is disabled 0.2 m from the final docking point. If necessary, you can modify the distance under **System > Settings > Docking > Distance to marker for disabling collision checks**.



L-markers are the only markers where the Protective fields are not muted automatically. If you want the robot to mute the Protective fields while docking to an L-marker, you must use a Mute Protective fields action—see [Personnel detection on page 79](#).

## Undocking from markers

A MiR robot can undock from markers automatically. When undocking, the robot reverses from the marker until it is outside of the undocking area—see **Figure 9.9**. The robot will keep muting its Protective fields while it undocks. Once the robot begins a new action that requires it to plan a new path, the Protective fields are activated again, and the robot continues normal operation.



The only case where the robot will not execute the undocking sequence is if you use a Relative move action right after the Dock to action. The robot will always execute the Relative move action first, and if the robot is outside of the undocking area after finishing the Relative move action, it will continue to the next action without undocking.

To enable automatic undocking from markers, go to **System > Settings > Docking** and enable the parameter **Undock from markers**.

You can also modify the size of the undocking area under **System > Settings > Docking** by using the parameters **Docked at marker reverse distance** and **Docked at marker side threshold**—see **Table 9.1**.

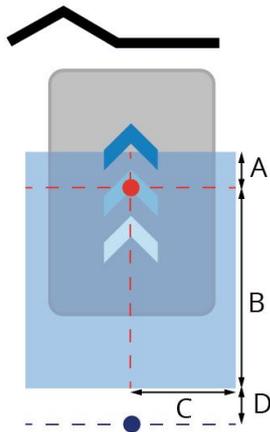


Figure 9.9. The undocking area initiates an undocking sequence for any robots with their center in the zone before beginning an action (except Relative move actions). The red dot represents the center of the robot when it is docked to the marker, the blue dot represents how far back the robot intends to move to undock from the marker, and the blue area represents the undocking area.

Table 9.1. Identification of the dimensions in <i>Figure 9.9</i>			
Pos.	Description	Pos.	Description
A	Always 10 cm. This is the distance in front of the robot's final position when docked to the marker. This ensures that even if the robot is pushed forward a bit, it will still undock when leaving the marker.	B	The distance entered under <b>Docked at marker reverse distance</b> . This is the distance from the center of the robot when it is docked to the edge of the undocking area furthest from the marker.
C	The distance entered under <b>Docked at marker side threshold</b> . This is the distance to either side of the center line	D	Always 10 cm. To make sure the robot's center has exited the undocking area, the robot aims to move back an additional 10

Pos.	Description	Pos.	Description
	from the robot's docked position.		cm.

The undocking sequence will always occur if a robot begins an action where it plans a new route while inside the undocking area, even if the robot was not previously docked to the marker. This means the sequence is also initiated on robots that have entered the area via a Relative move action or Manual control.

The robot will reverse from the marker as long as it's orientation to the marker is below 90°. If the orientation difference is greater than 90°, the robot will move forward to exit the undocking area instead. This means that if the robot reverses into the area or is perpendicular to the marker, it will undock by moving forward.

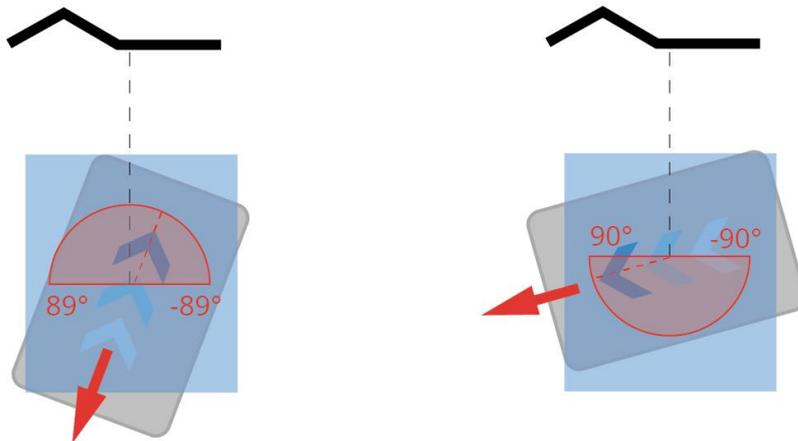


Figure 9.10. Robots that stop inside the docking area at an angle less than 90° to the marker undock by driving backward. Robots that stop with an angle greater than 90° undock by driving forward.



L-markers are the only markers that do not support automatic undocking. If you want a robot to undock from an L-marker, you must use a Relative move action.

## Types of markers

There are four standard marker types that all MiR robots can use: V, VL, L, and Bar-markers.

A **V-marker** is a small, V-shaped marker that is designed for the robot to either dock to so its front or its rear is facing the marker. The V-marker is the simplest marker available for the robot. It consists of a V shape with an interior angle of  $120^\circ$  and sides of 150 mm. The robot must dock straight toward the inside of the V.

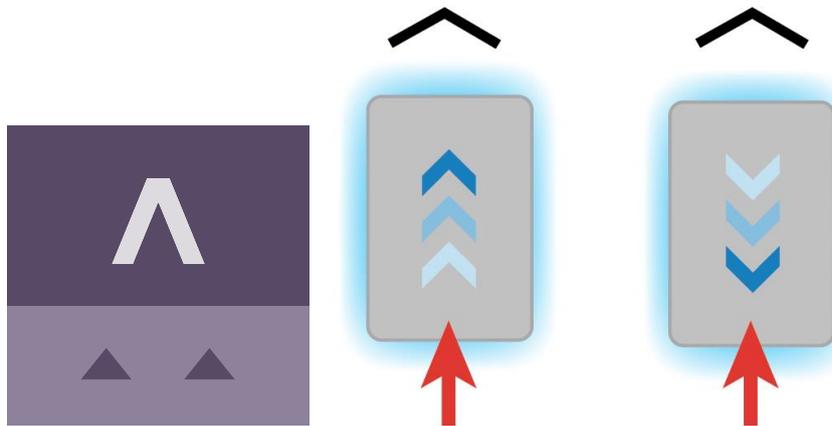


Figure 9.11. The icon used for V-markers in the interface and an illustration of how robots can dock to the marker.

A **VL-marker** is a larger marker that enables the robot to dock more accurately than V-markers. It consists of a V-marker with a 350 mm plate attached to the right of the V shape. Like V-markers, VL-markers are also designed for the robot to either dock to so its front or its rear is facing the marker. The robot must dock straight toward the inside of the V.

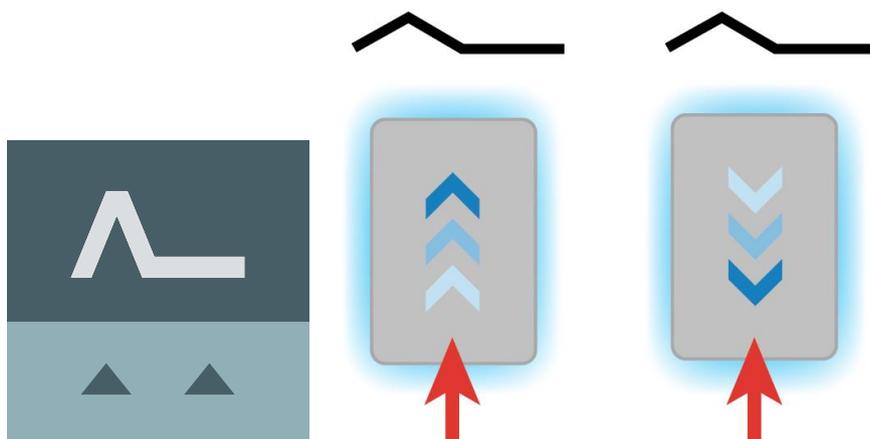


Figure 9.12. The icon used for VL-markers in the interface and an illustration of how robots can dock to the marker.

An **L-marker** makes it possible for the robot to dock in several different ways and orientations. Robots can both dock to the inside and outside of an L-marker, and the marker can be on any side of the robot. The marker is shaped like an L with a defined angle of 90° and the dimensions 400 mm × 600 mm.

The robot must dock so the longer side of the L-marker is parallel with the right and left sides of the robot. If the robot is docking outside of the marker, it must approach from the side with the 90° corner.

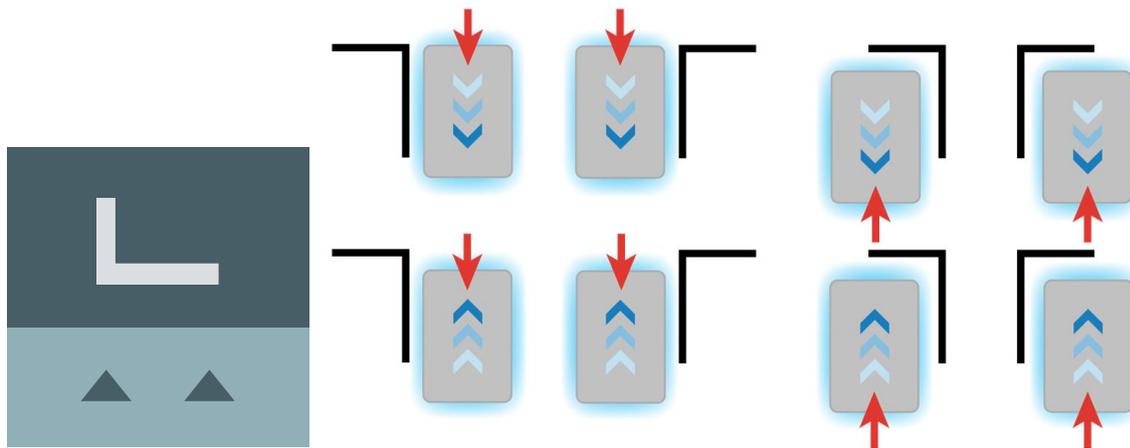


Figure 9.13. The icon used for L-markers in the interface and an illustration of how robots can dock to the marker.

A **Bar-marker** can be used for forward or reverse docking between two bars or plates, similar to pallet racks or shelves. Bar-markers must be between 400 mm and 2000 mm long, and the distance between the bars must be between 750 mm and 1500 mm. The robot can dock and undock from either end of the marker.



The distance between the bars must be larger than the physical dimensions of your robot.

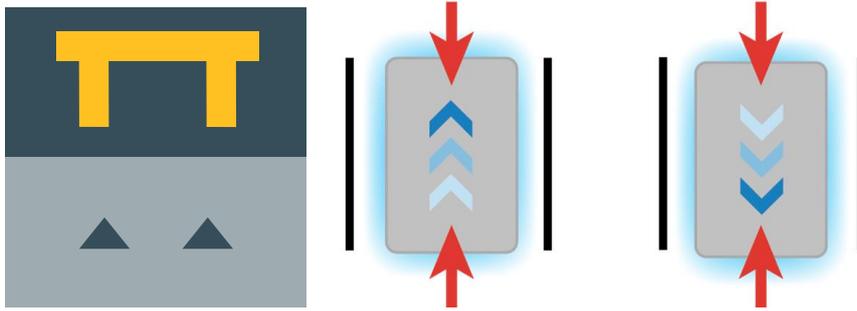


Figure 9.14. The icon used for Bar-markers in the interface and an illustration of how robots can dock to the marker.

A few centimeters between all the types of markers should make docking possible. Determine during commissioning if more space is required.



For further information on markers, contact your distributor for the guide *How to create and dock to V-markers, VL-markers, L-markers, and Bar-markers*.



To create a marker, see [Creating markers on page 134](#).

## 9.5 Positions

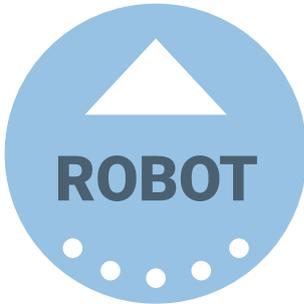
Positions are defined as X-Y coordinates on a map that mark locations where you want the robot to travel to. Positions mark a point on the map the robot travels to. To reach a position, the robot must be correctly localized on the map—see [Localization on page 71](#).

Positions are used either as destination positions or as waypoints on a route that you want to use in missions. With positions, the robot does not compare its position to a physical entity, making them less accurate than markers.

Generally, positions are used to mark where robots should wait when they are idle, which points robots must pass through along a route, or as destinations you often want to send the robots to.

The final orientation of the robot is indicated by the arrow on the position icon.

There are different types of positions depending on whether the robot is part of a fleet or drives with top modules, but the standard position that is available in all MiR applications is the Robot position. This position has no special features, it simply marks a location where you want to be able to send the robot to.



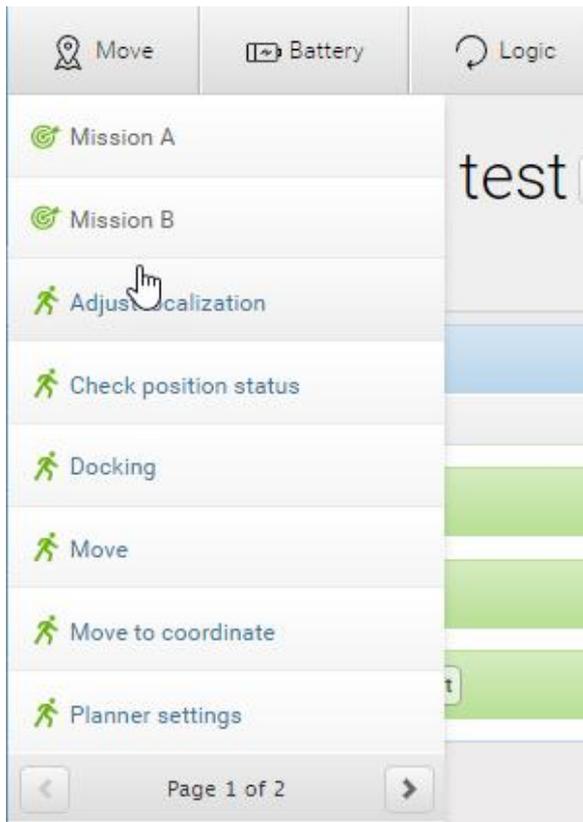
To create a position, see [Creating positions on page 139](#).

## 9.6 Creating missions

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 .



For more information on parameters and variables, contact your distributor for the guide *How to use variables in missions*.



### NOTICE

The robot prioritizes the instructions it receives in the following order:

1. Instructions from zones.
2. Instructions from mission actions.
3. Instructions from system settings (except from the setting Maximum allowed speed, which is never overruled by neither zones nor mission actions).

To create efficient missions, you should first familiarize yourself with the available actions in MiR Robot Interface—see the *MiR Robot Reference Guide*— and then consider:

- Which tasks do I want the robot to perform?
- Which actions are involved in this task and in which order are the actions executed?
- How much do each of these tasks differ? Are they similar enough that you can reuse the same mission but use variables for some of the parameters? If so, identify which of the parameters change in each mission—see **Figure 9.15**.

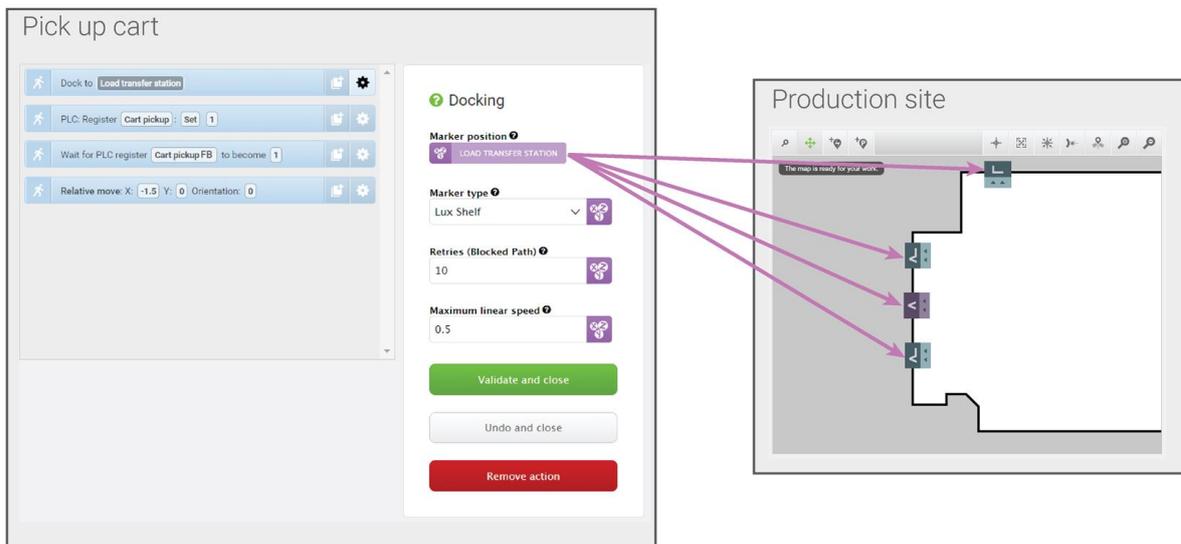


Figure 9.15. You can use variables to make a mission where you can set a parameter in one of the actions each time you use the mission (either when you add the mission to the mission queue or embed it in another mission). In this example, you can set the variable **Load transfer station** to any marker created on the map. This means that you can use the same mission for making the robot pick up a load from any of the markers on the map.

- Are there small parts of different missions that are the same each time where it would be worthwhile to make a mission for the repeated task and embed this mission into the larger missions? For an example, see **Figure 9.16**.

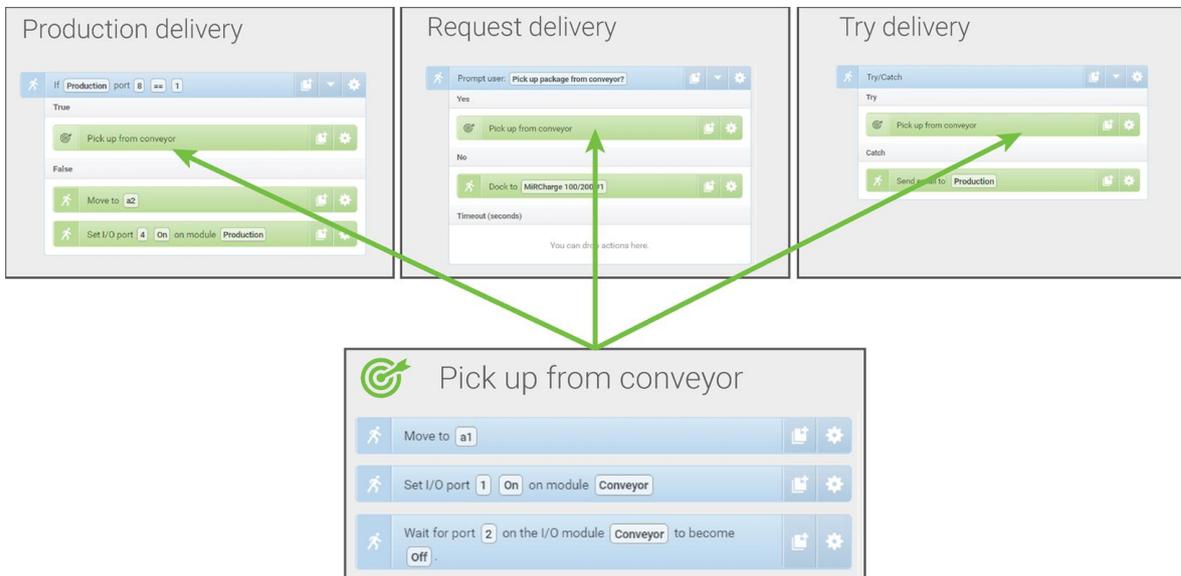


Figure 9.16. You can embed small missions into other missions. In this example, the mission *Pick up from conveyor* is used in three different missions. If you want to change how the robot picks up a package from the conveyor, you only need to change it once in the original mission instead of three times in each individual mission.



It is often a good idea to reuse the same missions if you know that any changes that may need to be applied to one of the tasks will also need to be applied to all other similar tasks.

When you make a mission, you should also consider all the possible outcomes from the mission and prepare it for the possibilities of error and what the robot should do if an error occurs—see an example of this in [Creating the mission Try/Catch on page 146](#).



For more information on building robust missions, see the *Mission robustness* videos in MiR Academy on the MiR website. Contact your distributor for access to MiR Academy.

When you have figured out which tasks you want the robot to perform and how many different missions you need to create, you should consider how you want to organize the missions in different mission groups. You can consider the following:

- Do you want to add the missions into existing action groups?
- Do you want to create new mission groups to organize your missions in? If so, consider how you want to divide your missions. For example, you can divide them based on function, location, priority, or responsible users.

The section [Usage on page 134](#) provides several examples of how to create simple missions with different types of mission actions and describes how you add a mission to the mission queue to test it. Whenever you create a mission, it is very important that you test it to ensure the robot performs as expected.



For more information on creating missions, see *MiR Robot Reference Guide* and the *Making your first missions*-course in MiR Academy on the MiR website. Contact your distributor for access to MiR Academy.

## 9.7 Creating a footprint

The footprint specifies how much space the robot occupies, including any loads or top modules. The footprint is defined by a number of points relative to the robot's center coordinate system and the total height of the robot application.

If your robot drives with loads or top modules that exceed the width or length of the robot, you must define new footprints for the robot to ensure that the robot plans its route correctly and avoids colliding with obstacles with its top module or load.

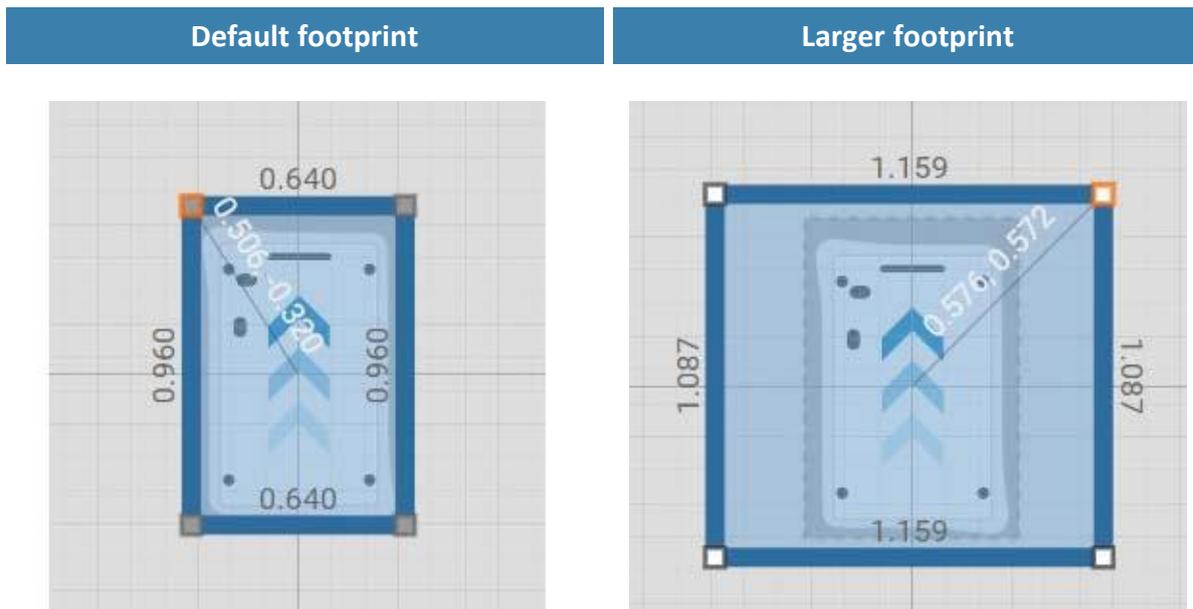


Figure 9.17. Examples of the default robot footprint and an extended footprint. The values displayed along each line is the length of the edge in meters.

The number of footprints you need to define depends on:

- The top modules you use with your robot.
  - If a robot's top module exceeds the width or length of the robot, you must define a new footprint for that top module.
  - If a top module has moving parts that can extend over the edges of the robot's footprint while the robot is moving, you must define a footprint that includes the moving parts when they are at their most extended positions.
- The loads the robot transports.
  - For each load the robot transports that exceeds the length or width of the robot, you must define a footprint for that load.
  - If you prefer to only have one footprint for the robot when it is carrying oversized loads, create a footprint that is suitable for the load that has the largest footprint.
- If there are low hanging fixtures that the robot can pass under only when it is not carrying certain loads or top modules, you must define new footprints for the various heights that the robot and its load can have to ensure that they don't collide with the low hanging fixtures.

**CAUTION**

The footprint is only used by the robot's global and local planner to avoid obstacles. The Personnel detection safety function—see [Personnel detection on page 79](#)—still uses the same Protective field sets. If your robot is carrying a load or top module that extends the footprint in front of or behind the robot, it may collide with personnel or equipment.

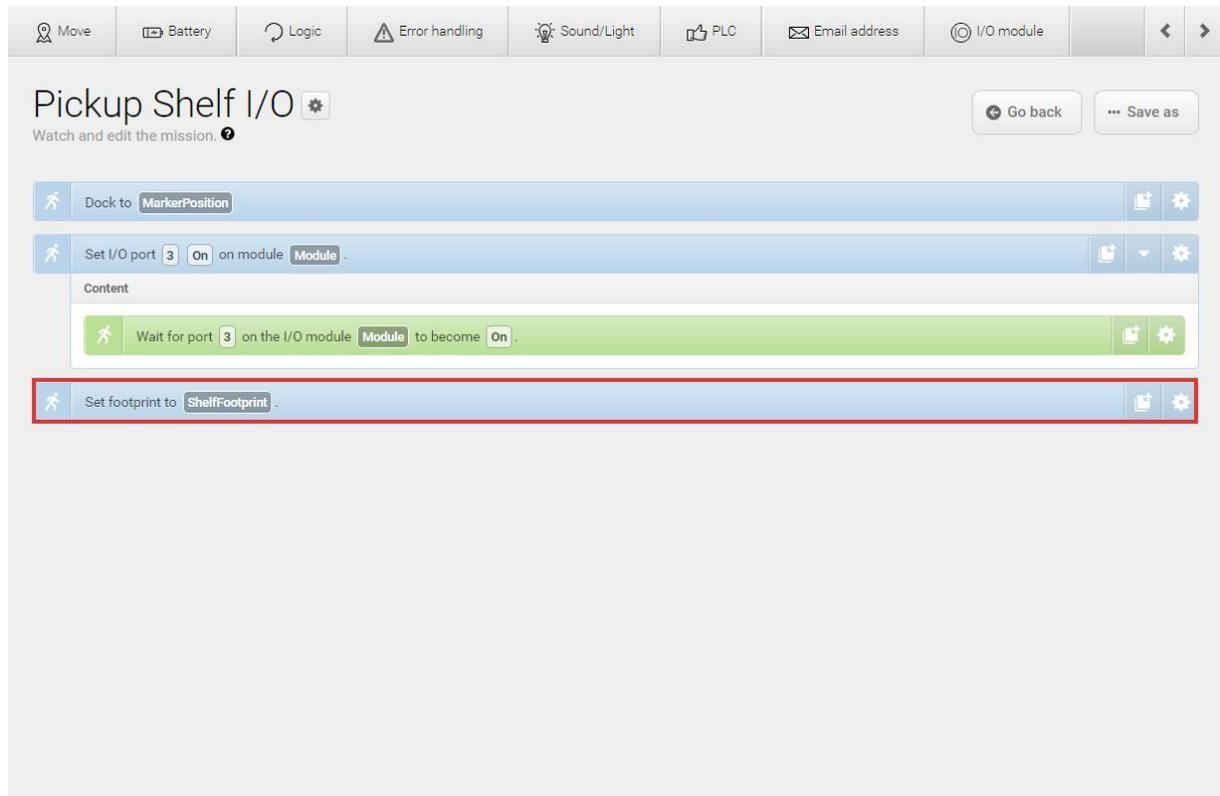
- Avoid extending the footprint in front of or behind the robot.
- Mark all areas where the robot drives with an unsafe load as operating hazard zones.
- Consider modifying the Protective field sets if necessary—see [Personnel detection on page 79](#).



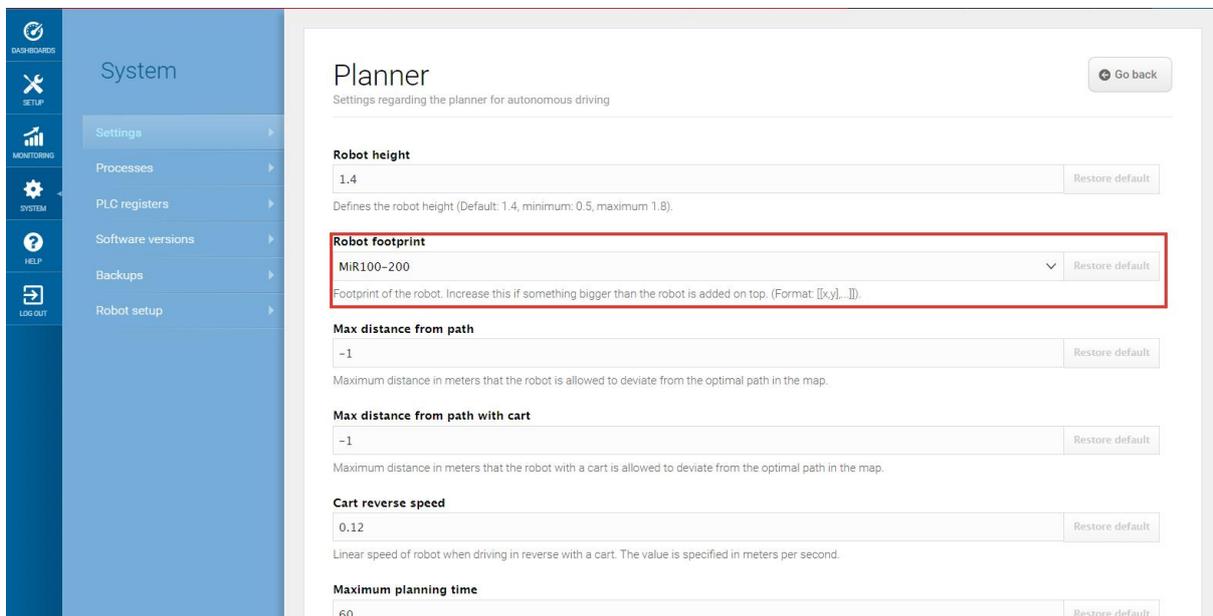
For a more thorough guide to creating footprints, contact your distributor for the guide *How to change the robot footprint*.

For more information about the footprint editor, see *MiR Robot Reference Guide* on the MiR website.

If you want to change the footprint in a mission, use the Set footprint action found under the Move action group. This is used to change the footprint when the robot picks up a load that extends the footprint or places a load and the footprint returns to the default.



If you want to edit the default footprint of the robot, for example if the mounted top module is larger than the robot, go to **System > Settings > Planner**, and select a new footprint under **Robot footprint**.



## 9.8 Making a brake test

It is the responsibility of the commissioner to perform an adequate test of the robot's braking capability.

The braking distance of MiR100 is particularly dependent upon four factors:

1. The speed of the robot
2. The payload of the robot
3. The surface the robot drives on
4. The decline of the surface the robot drives on

Because of this, it is not possible to predetermine the exact braking distance of MiR robots. The distance has to be determined in the environment and under the driving conditions the robot will be operating in.

The goal of the brake test is to ensure that the robot will brake in time to avoid a collision with a person or object when driving with maximum payload, with different field sets for different speeds, and at the steepest supported decline.

If the measured braking distance is too long, the Protective field sets of the robot should be made larger to ensure a safe installation. This can happen if the floor has low friction, for example, high gloss floors and similar. The Protective fields should always be at least somewhat larger than the braking distance at all speeds. To modify the field sets, contact your distributor for the guide *How to adjust the Protective field sets on MiR100 and MiR200*.



The guide is written for MiR500 and MiR1000. There may be some differences in the top module. Until a new version of the guide is available, the current guide may function as a guideline.

## 9.9 Creating user groups and users

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

The user profiles are created during commissioning. By default, the robot has three user groups: User, Administrator, and Distributor. Parts of the user interface can be locked by the commissioner. The locked parts are typically related to the safety of the robot system, and changing these settings can violate the CE marking of the robot.

It is important to analyze and consider who is:

- Working directly with MiR100 as direct users or operators?
- Responsible for MiR100 as commissioner?

Furthermore, the following questions should be answered:

- How many different users are there?
- What tasks does each user have with MiR100?
- What permissions should the different users have?
- What functions or widgets should be available for the different users?



For more details on users and dashboards, see *MiR Robot Reference Guide* on the MiR website.

## Create user groups

In **Setup > User groups**, you can create specific user groups with specific access to different parts of the robot interface.

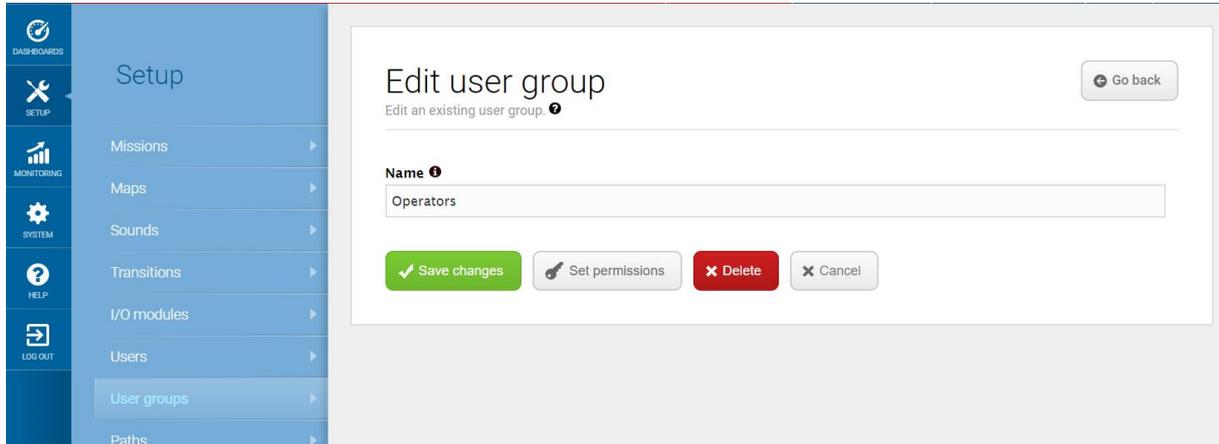


Figure 9.18. You can create specific user groups.

Under **Set permissions**, you can select the specific parts of the robot interface that the user group has access to.

Maps & positions		
	Section	<input type="checkbox"/> Write
	Sites	<input type="checkbox"/> Write
	Maps	<input type="checkbox"/> Write
	Zones	<input type="checkbox"/> Write
	Positions	<input checked="" type="checkbox"/> Write
	Paths	<input type="checkbox"/> Write
	Path guides	<input type="checkbox"/> Write
	Marker types	<input checked="" type="checkbox"/> Write

Figure 9.19. You can select the specific parts of the robot interface that the user group has access to.

## Create users

In **Setup > Users**, you can create new users and select:

- Which user group they belong in.
- If they are SingleDashboard users with no access to other parts of the interface than to control the robot from a dashboard.
- If they should be allowed quick access to the interface via a four digit PIN code. We only recommend PIN codes for users with no access to settings and safety system.

Figure 9.20. When you create a user, you must fill out the fields shown in this image.

Table 9.2. Examples of which users MiR recommends should be able to edit which features	
Feature	User group
Controlling the robot manually	Operator
Creating maps and positions	Commissioner
Creating and editing missions	Operator
Adjusting warning sounds	Commissioner
Creating new user groups	Commissioner

Feature	User group
Assigning missions	Direct user
Changing system settings	Commissioner

## 9.10 Creating dashboards

To make the user experience as easy and simple as possible, you can build a unique dashboard for each user. Dashboards are an easy way for different user groups to control the robot, giving direct access to the individual groups' key functions.



For more details on how to use and create dashboards, see *MiR Robot Reference Guide* on the MiR website.

A dashboard is made up of a number of widgets, each representing a feature in the system, for example a particular mission, the map the robot is operating on, or the current mission queue.

The system comes with a default dashboard—see **Figure 9.21**—, and users with access rights to create dashboards can create an unlimited number of additional customized dashboards.

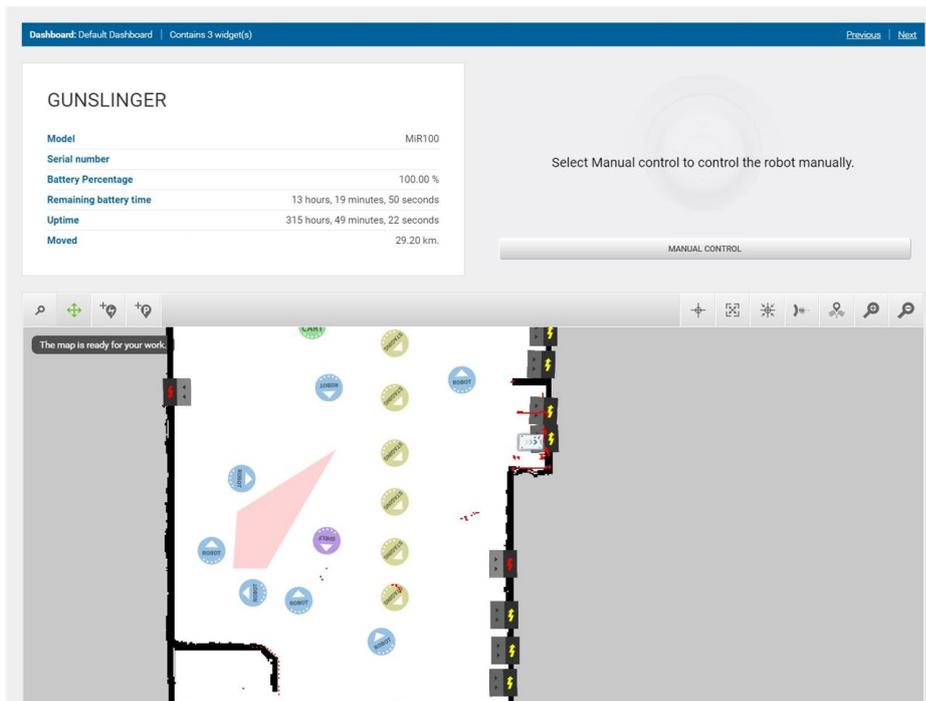


Figure 9.21. The default dashboard includes the robot information, a joystick for manual control, and the active map.

When creating new dashboards, you should consider the following:

- Who will be using the dashboards?
- Which functionalities will they need to use the most?
  - For example, if your robot uses many I/O modules, you may want to monitor them from the dashboard, or if there is a mission that the robot often has to execute on demand, you may want to add it to the dashboard.
- Will each user or user group need a different dashboard? If so, what should be included in each?
- Will some users need more than one dashboard?
  - Users that are responsible for both maintaining and operating the robot could have separate dashboards for the maintenance routine and another dashboard for operating the robot.

- If you have any SingleDashboard users, which functionalities will they need and which would be useful to include?
  - Often it is not a good idea to include too many widgets in the dashboard as this can slow down the interfacing to the robot. Try to include only the necessary widgets.

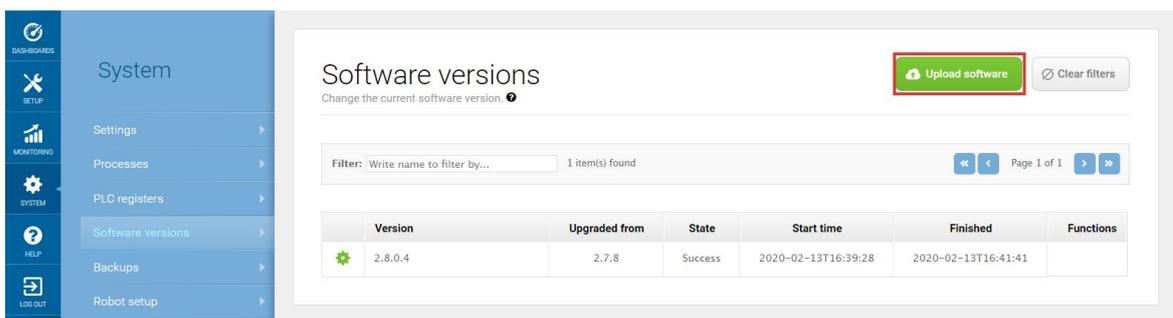
## 9.11 Updating MiR100 software

MiR continuously updates the software the robots use, either to fix issues, to improve existing features, or to introduce new features. Each software release is issued with a release note explaining the content of the update and its target audience.

Contact your distributor for the latest recommended update file.

Follow the steps below to update MiR100 software:

1. Connect your computer to the robot you want to update, and sign in to the robot interface.
2. Go to **System > Software versions** and select **Upload software**.



3. Locate and select the downloaded software package. It may take 10-20 minutes for the package to successfully upload depending on whether or not the software introduces new security patches.
4. Once the software is uploaded, turn the robot off and then on again.

## 9.12 Creating backups

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

We recommend to create a backup in the following cases:

- Before you update the robot software.
- Before making any large changes to your site.



Backups take up some of your robot's memory space. It is a good idea to remove any old backups you are certain you will not need in the future.

For more information on how to create, roll back, and delete backups, see *MiR Robot Reference Guide* on the website.

## 9.13 System settings

This section describes some of the commonly used system settings of MiR100 that the commissioner must be aware of.



Only the basic system settings are explained in this section—see *MiR Robot Reference Guide* on the MiR website for more information.



### NOTICE

The robot prioritizes the instructions it receives in the following order:

1. Instructions from zones.
2. Instructions from mission actions.
3. Instructions from system settings (except from the setting Maximum allowed speed, which is never overruled by neither zones nor mission actions).

In **System > Settings**, you can access the settings of the robot. Access to the settings must be restricted by the commissioner—see [Creating user groups and users on page 120](#).

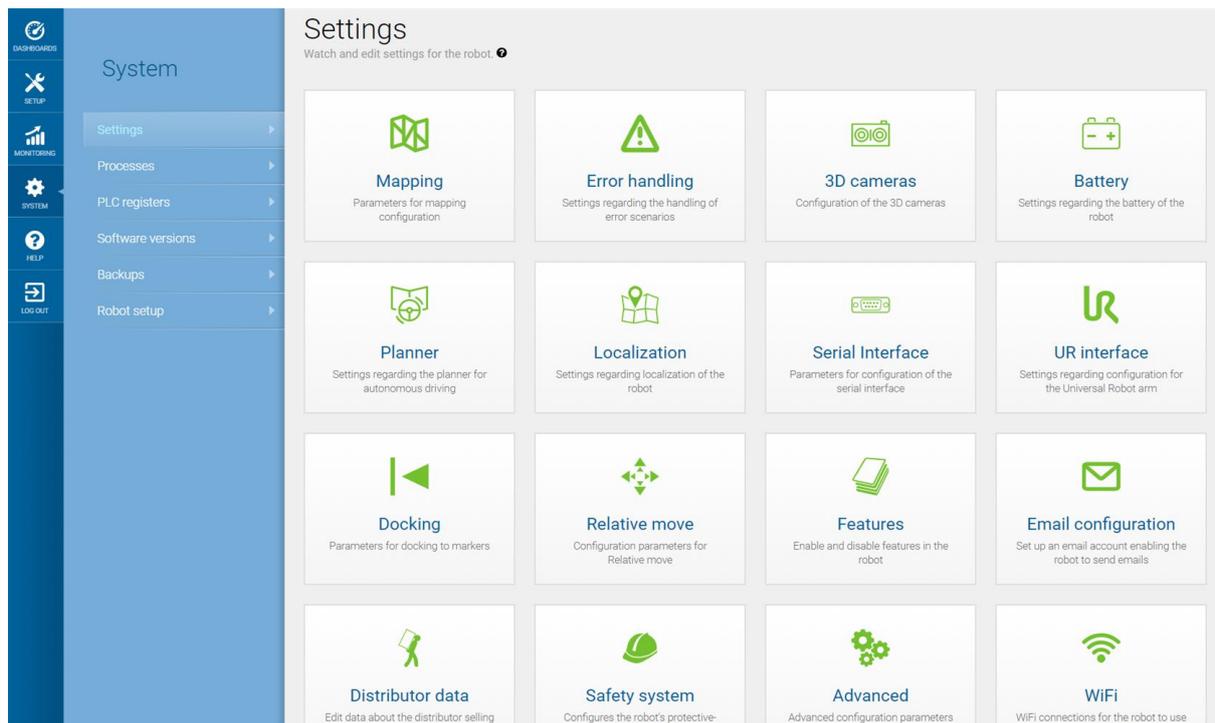


Figure 9.22. Under **System > Settings**, there are several menus where you can edit your robot's settings.



Remember to restart the robot if you have made any changes to the system settings.

## Planner

In the Planner section, you set the basic parameters for driving the robot.



This section refers to the local and global planner functions. For more information on the robot's path planners, see [Global planner on page 61](#) and [Local planner on page 63](#).

## Planner

Go back

Settings regarding the planner for autonomous driving

### Robot height

1.4

Restore default

Defines the robot height (Default: 1.4, minimum: 0.5, maximum 1.8).

### Max distance from path

-1

Restore default

Maximum distance in meters that the robot is allowed to deviate from the optimal path in the map.

### Max distance from path with cart

-1

Restore default

Maximum distance in meters that the robot with a cart is allowed to deviate from the optimal path in the map.

### Maximum planning time

60

Restore default

The maximum time allocated for planning a path. The value is specified in seconds.

### Path timeout

0

Restore default

Timeout before asking for a new global plan [in seconds]. Note! Negative value means infinite waiting time.

### Path deviation

-1

Restore default

Maximum distance in meters that the robot is allowed to deviate from the path planner in the map. Note! Negative value disables the feature.

### Indicator light

False

▼

Restore default

Set to True to enable indicator lights while driving autonomously.

Figure 9.23. You set basic parameters for driving the robot in the Planner section.

**Robot height** defines the height of the robot including top modules. Use this setting if your robot operates permanently with a top module that makes the combined robot application higher than the robot itself. This prevents the robot from colliding with obstacles from above.

**Max distance from path** defines the maximum allowed distance in meters that the generated global path is allowed to deviate from the most direct path on the map. By default, this parameter is disabled, meaning the robot will always make a global path and follow it to the goal position no matter how far the path is. If you want to avoid the robot traveling paths of a specific length and report an error instead, enter the maximum length that the global path may exceed the most direct path.

**Maximum planning time** defines the maximum time allowed for planning a path. By default, this parameter is disabled, meaning the robot will always try to finish planning a global path no matter how long it takes. If you want the robot to report an error after a set time period instead, enter the maximum amount of time in seconds that the robot can spend planning a path before it reports an error.

**Path timeout** defines the maximum time the robot's path can be blocked before the robot generates a new global path. By default, this value is 0, meaning the robot will not wait if its current global path is blocked by an obstacle it cannot navigate around using the local planner. If you want the robot to wait and see if the obstacle moves before planning a new path, enter the maximum waiting time.

**Path deviation** defines the maximum distance in meters that the local path is allowed to deviate from the global path before the robot makes a new global path. By default, this parameter is disabled, meaning the robot can deviate from the global path using the local planner to go around an obstacle as far as possible in the map.



Optimizing the timeout and deviation of paths is useful in situations where you want to configure how strictly the robot should follow the path it has planned. Making the robot follow the exact path it has planned with little or no deviation is known as Line-following mode. This can be useful, for example, in narrow corridors where there isn't enough space for the robot to go around dynamic obstacles—see **Figure 9.24**.

For more information on Line-following, contact your distributor for the guide *How to set up Line-following mode*.

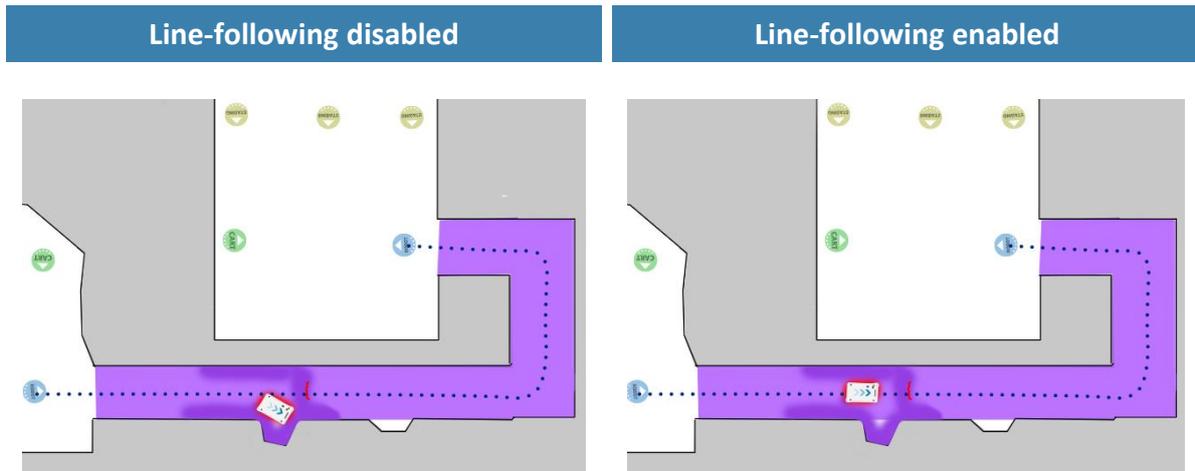


Figure 9.24. Example of where the robot might benefit from using a Line-following configuration. When there isn't enough space for the robot to go around an obstacle, it will often spend more time trying to maneuver around the obstacle and correct its trajectory afterward than it would have just waiting for the obstacle to move out of the way.

**Maximum allowed speed** defines the overall speed limit of the robot. The maximum allowed speed will never be exceeded no matter what is stated in a mission or Speed zone. This setting can be useful if, for example, the robot transports motion sensitive objects or if the work environment in other ways requires the robot to always stay below a certain speed threshold.

**Desired speed** sets the desired speed of the robot. This setting can be useful in the same way as maximum allowed speed, but with this setting, the robot will drive faster than the set desired speed in a Speed zone that requires it.

## Docking

In the Docking section, you can change the parameters regarding docking to and from 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

### Distance to marker for disabling collision checks.

0.2 Restore default

Distance to marker where collision detection is disabled.

### 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

### Parameter for driving more straight during docking

1.0 Restore default

Setting of maximum allowed change to marker estimate when the robot moves towards a marker (in meters per meter). Reducing the value of this parameter makes the robot move more straight during docking. However, it also reduces the precision of the docking.

### 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.

Figure 9.25. Change the parameters regarding docking to and from markers in the Docking section.

In **Undock from markers**, you can select if the robot should undock from a marker before it starts moving from a docked position. It is usually best to set this setting to **True** to prevent the robot from going into Protective stop when moving away from markers.



The robot cannot undock from L-markers automatically—see [Markers on page 103](#). You must use a Relative move action.

In the advanced settings, you can adjust the parameters for docking to markers. This can be useful in case of docking issues. To see the advanced docking settings, select **Show advanced settings**.

## Features

In the Features section, you can disable and enable robot features.

## Features

Enable and disable features in the robot

Go back

### Shelf

False Restore default

Select **True** if the robot is used to pick up and place shelves.

### Email address

True Restore default

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

### PLC registers

True 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 Restore default

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

### Fleet

False Restore default

Select **True** if the robot is part of a fleet

### Modbus

False Restore default

Select **True** if the robot uses Modbus

### I/O modules

True Restore default

Add actions for communicating with IO modules

Figure 9.26. Disable and enable robot features in the Features section.

**Shelf** enables the shelf feature. Enable this feature if the robot is used to pick up and place shelves.

**Hook** enables the hook feature. Enable this feature if your top module is a MiR Hook. This will enable the Hook menu in the robot interface.

**Email address** enables an action for sending emails from missions. You can configure the email account that the robot sends the emails from under **System > Settings > Email Configuration**.

**PLC registers** enables actions for setting PLC registers from missions and monitoring PLC registers in the robot interface. When enabled, you can access the page **System > PLC registers** to set up the registers.

**Universal Robots Interface** enables an action for running Universal Robots programs from missions. Enable this feature if the robot drives with an application from Universal Robots.

**Fleet** makes the robot visible for MiR Fleet. Enable this feature if the robot is part of a fleet.

**Modbus** enables Modbus communications. When enabled, you can access the page **System > Triggers** to set up the Modbus triggers.

**Precision docking** enables you to create precision docking markers on your map. Enable this feature if the robot has been installed with MiR Precision Docking.

# 10. Usage

The main way to use MiR100 is through missions that you create.

In the following sections you will find practical examples of how missions can be tailored to different tasks. The examples include:

- Setting markers and positions on the map.
- Creating a mission that uses a Prompt user action. The example mission is titled *Prompt user*.
- Creating a mission that uses a Try/Catch action. The example mission is titled *Try/Catch*.
- Creating a mission that uses variables. The example mission is titled *Variable footprint*.

## 10.1 Creating markers

Before creating a marker, you must ensure that the robot is localized correctly on an active map. If in doubt, you can check if the red lines representing the laser scanner view match the black lines on the map, as shown in **Figure 10.1**.

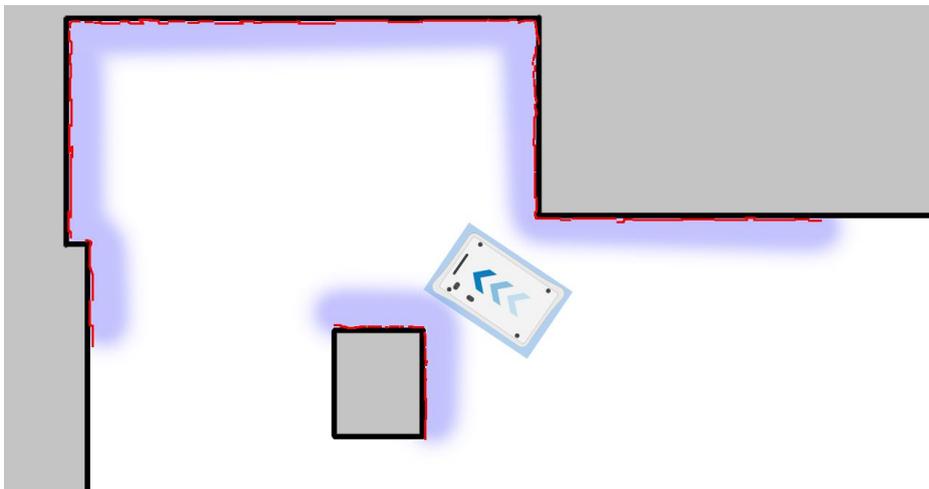
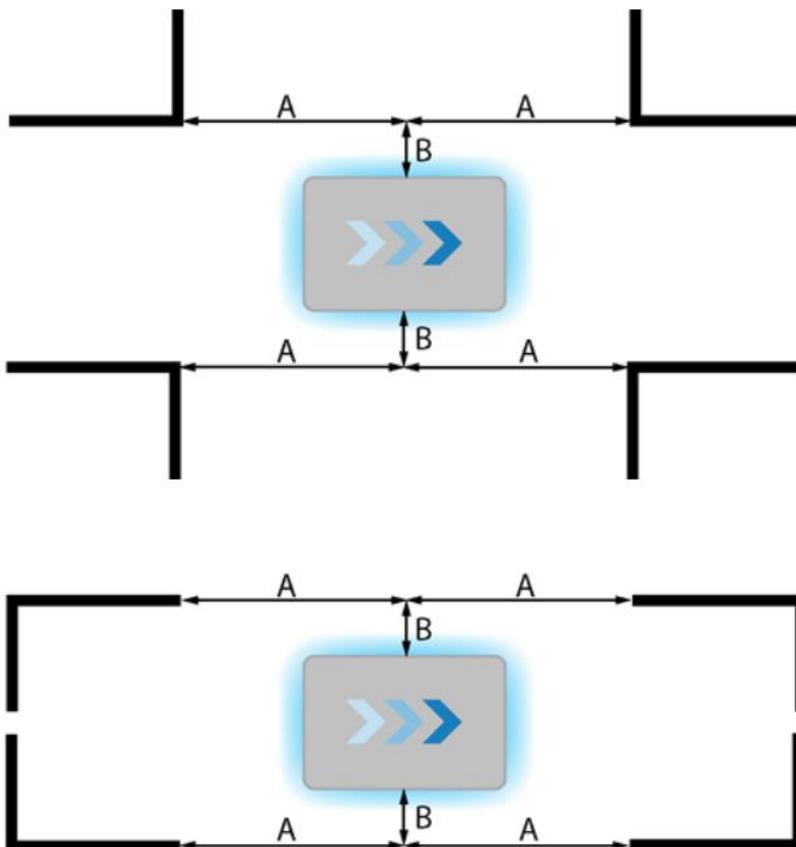


Figure 10.1. The red lines represent the obstacles the laser scanners detect. The robot is localized correctly when the red lines align with the black lines that represent walls.

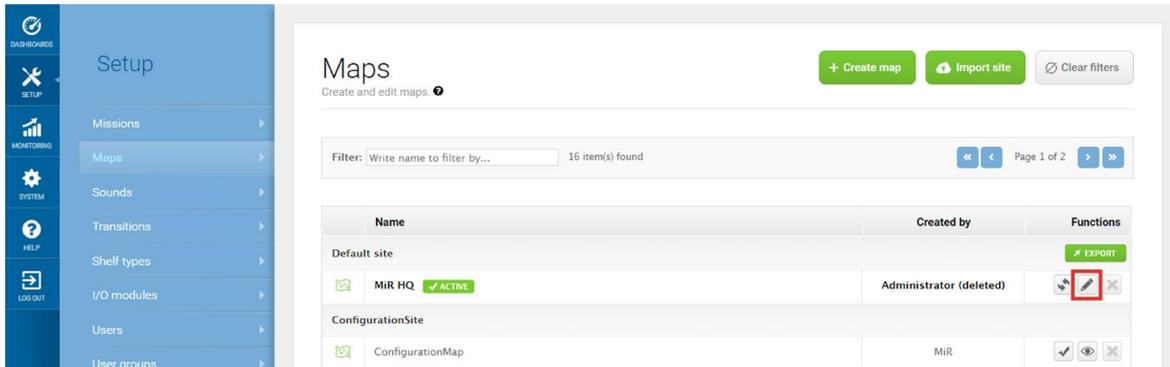
Once the robot is localized, you can insert a marker on the map. In this example, we are using a VL-marker . To create a marker, follow these steps:

1. Place your physical marker where you want the robot to dock.
2. Manually drive the robot to the marker so the robot is facing the marker. The correct distance from the marker differs depending on the marker type:
  - For L-markers, the following values apply:
    - A: 700 mm  $\pm$ 50 mm
    - B: 200 mm  $\pm$ 50 mm

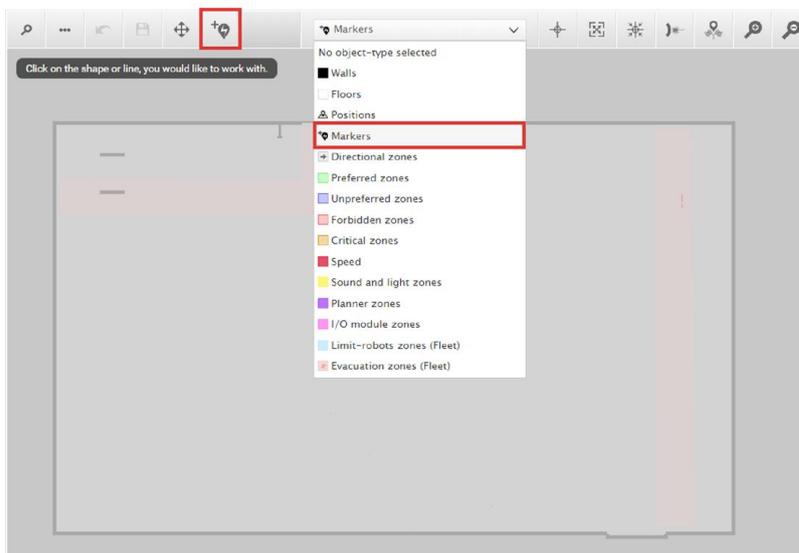


- For all other markers, the robot must be positioned approximately one meter directly in front of the marker.

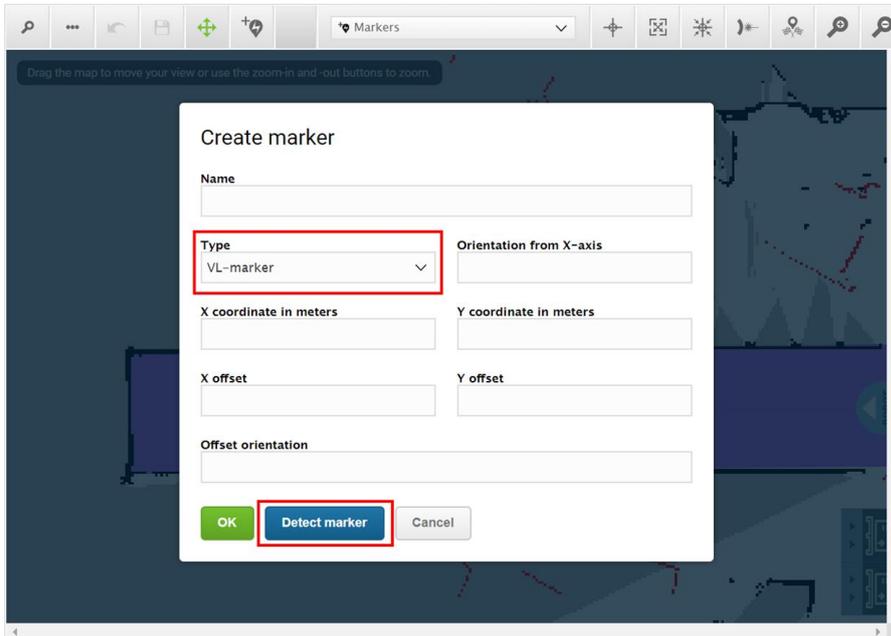
- Go to **Setup > Maps**, and select **Edit**  for the active map.



- Within the editor, select **Markers** in the Object-type drop-down menu, and then select **Draw new marker**  in the editor tools.



- In the **Create marker** dialog box, name the marker. Under **Type**, select your marker type. In this case, a **VL-marker** is used. Then select **Detect marker**.



The X, Y, and orientation values will automatically be filled out with the current position of the robot.

- If the robot cannot detect the marker, verify that the robot is correctly positioned and that the laser scanners can detect the marker in the active map by checking that red lines are displayed on the map where the marker is.
- If you are trying to make the robot detect an L-marker but it keeps detecting other objects with a 90° angle instead, shield the objects that the robot is not supposed to detect with a flat plate.
- If you want the robot to dock straight to the marker, set the **Offset orientation** to 0°. If you want the robot to reverse into the marker, set the **Offset orientation** to 180°.



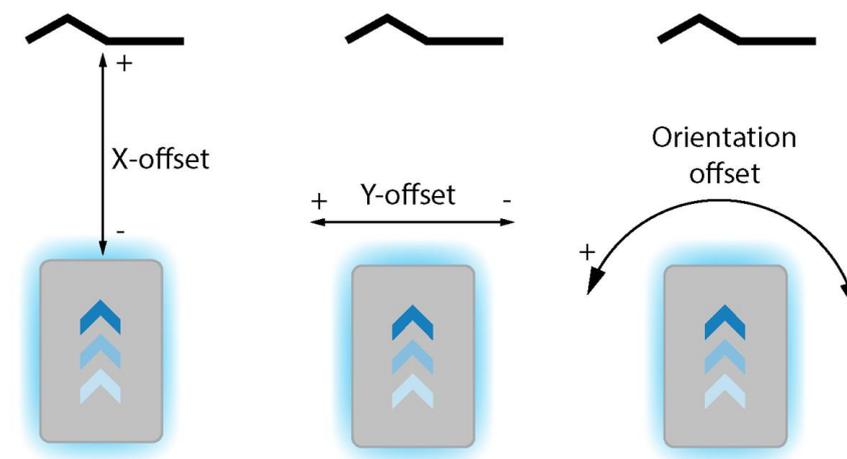
Detecting the marker with the rear scanner will automatically set the orientation offset to approximately 180° for a reverse docking.

- To change where the robot stops relative to the marker, you can adjust the offsets.

### Edit marker

Name ROEQ_Docking yoyo cart	
Type VL-marker	Orientation from X-axis -91.612
X coordinate in meters 26.152	Y coordinate in meters 18.196
X offset -0.51	Y offset 0.04
Offset orientation 0	
<input type="button" value="OK"/> <input type="button" value="Detect marker"/> <input type="button" value="Cancel"/>	

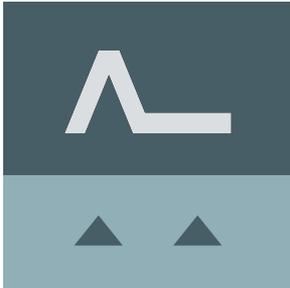
- The X-offset moves the robot closer to or further from the marker in meters.
- The Y-offset moves the robot further to the left or right of the marker in meters.
- The orientation offset changes the final orientation of the robot in degrees.





For more information about the marker offsets, see *How to create and dock to V-markers, VL-markers, L-markers, and Bar-markers.*

6. Select **OK** to create the marker. The marker is now visible on the map.



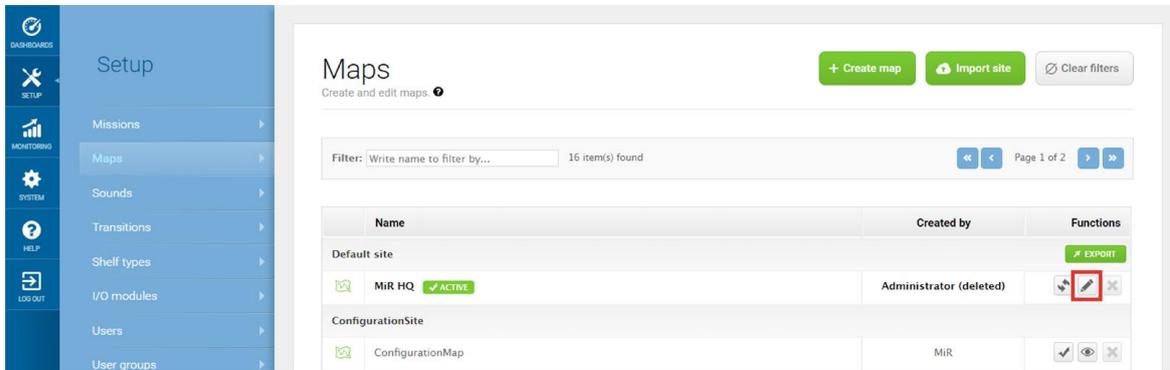
You can make the robot dock to the marker by selecting it on the map and selecting **Go to**. The marker can also be used in missions.



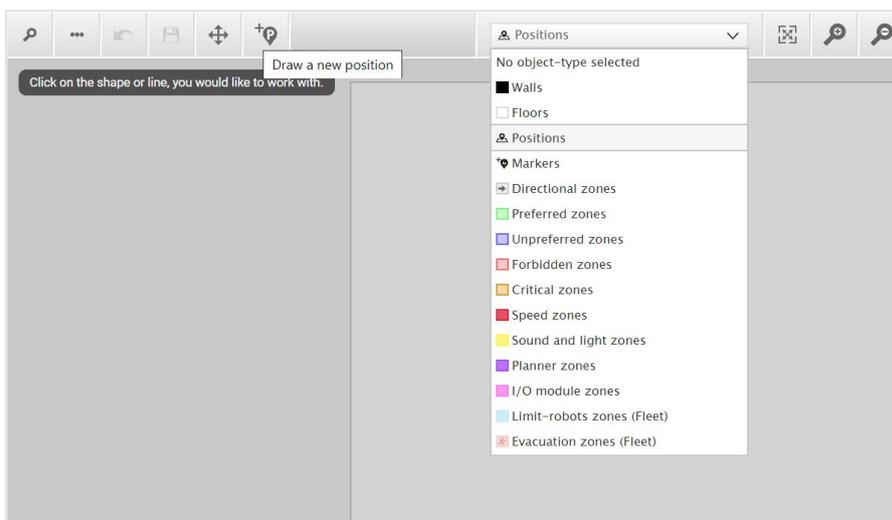
## 10.2 Creating positions

The following steps describe how to create a position on a map. In this example, we are creating a Robot position .

1. In the robot interface, go to the map editor of the map where you want to create a position. This is done by going to **Setup > Maps** and selecting **Edit**  next to the map you would like to work on.



2. In the Object-type drop-down menu, select **Positions**, and then select **Draw a new position** .



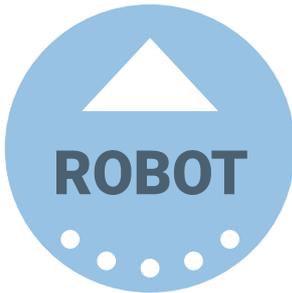
3. Select where on the map you want the position to be, and choose in which direction you want it to face.

- Name the position. Under **Type**, select which type of position you want to make. In this example we are making a Robot position.

### Create position

<b>Name</b> Test	
<b>Type</b> Robot position	<b>Orientation from X-axis</b> -74.604
<b>X coordinate in meters</b> 6.950	<b>Y coordinate in meters</b> 11.250
<b>OK</b>	Cancel

- Select **OK** to create the position. The position is now visible on the map.



You can send the robot to the position by selecting it on the map and selecting **Go to**. The position can also be used in missions.

<b>GO TO</b>	Send robot to this position
EDIT	Edit this position's name and parameters.
<b>DELETE</b>	Delete this position from the map.
<b>SHOW ENTRY POSITION(S)</b>	Show entry position(s) for this position.
CANCEL	Close this dialogue.

## 10.3 Creating the mission *Prompt user*

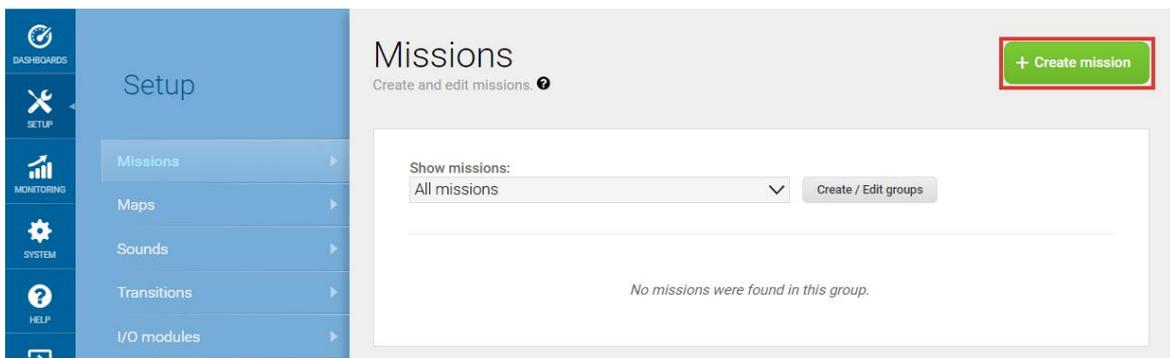
Prompt user actions are used for prompting the user with a question on how the robot should proceed. *Prompt user* is an example mission that uses a Prompt user action that lets you choose whether to send the robot to one position or another.

Before you create the mission *Prompt user*, it is assumed that you have completed the following:

- Created two robot positions named *p1* and *p2* as described in [Creating positions on page 139](#).
- Defined a user group named *Users*.

To create the mission, follow the steps below:

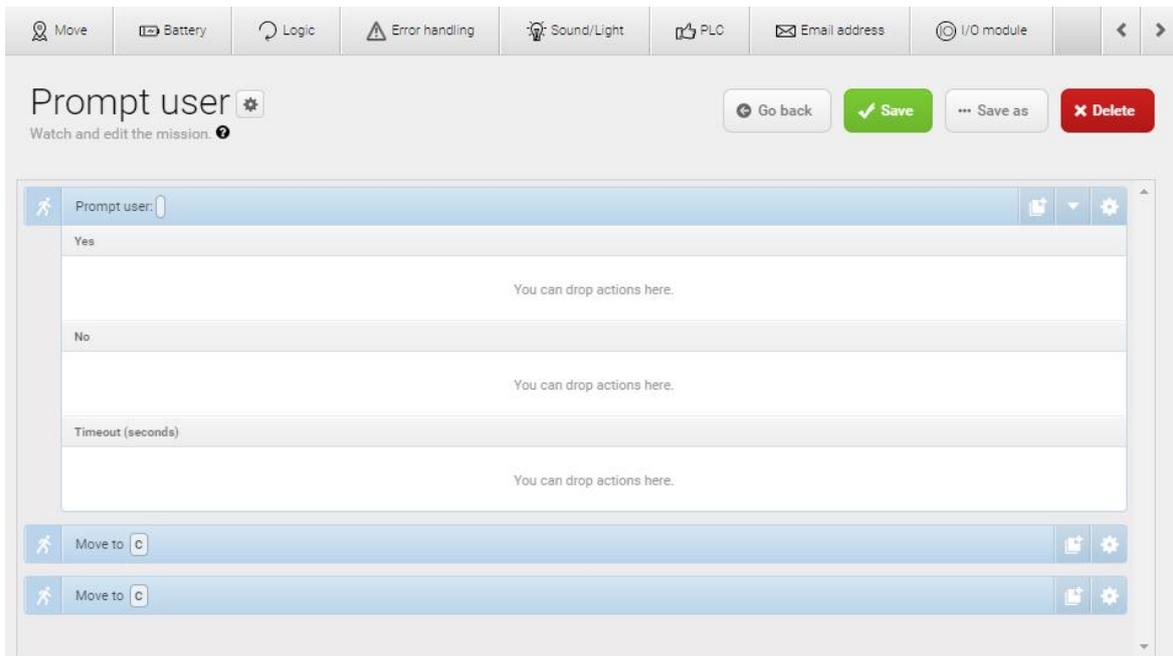
1. Go to **Setup > Missions**. Select **Create Mission**.



2. Name the mission *Prompt user*. Select the group and site you want it to belong to. Select **Create mission**.

3. Select the following actions:

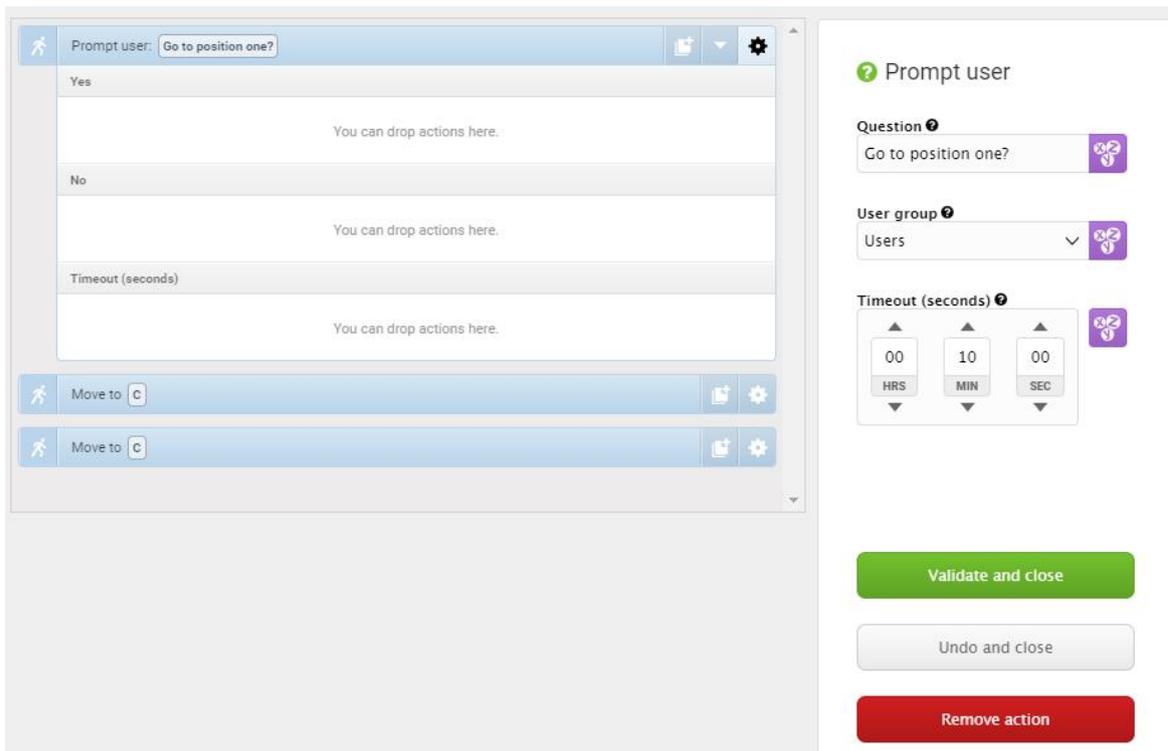
- In the Logic menu, select **Prompt user**.
- In the Move menu, select **Move**.
- In the Move menu, select **Move**.



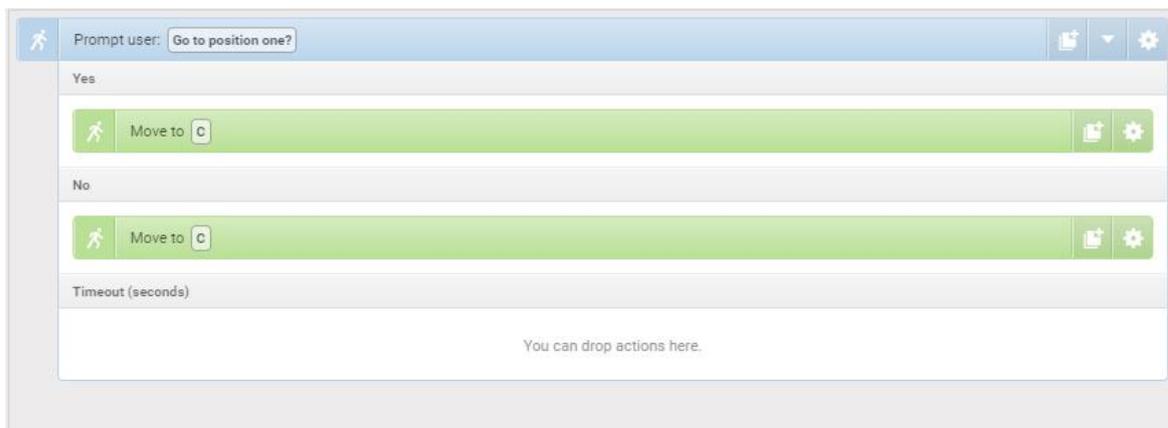
The following steps describe which parameters each action should be set to. To modify the parameters, select the gearwheel  at the right end of the action line to open the action dialog box. When you have set the parameters, select **Validate and close**.

4. In the Prompt user action, set the parameters as follows:

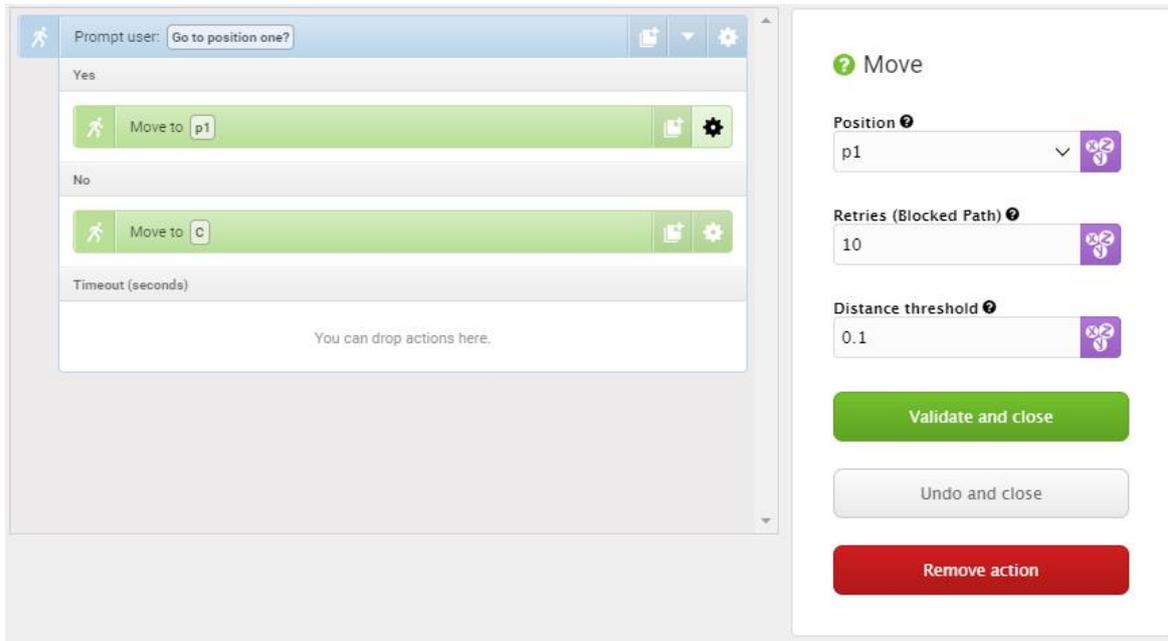
- **Question:** Enter the question **Go to position one?**.
- **User group:** Select **Users**.
- **Timeout:** Set the timeout to 10 minutes.



5. In the Prompt user action, drag a Move to action under the **Yes** box and a Move to action under the **No** box.

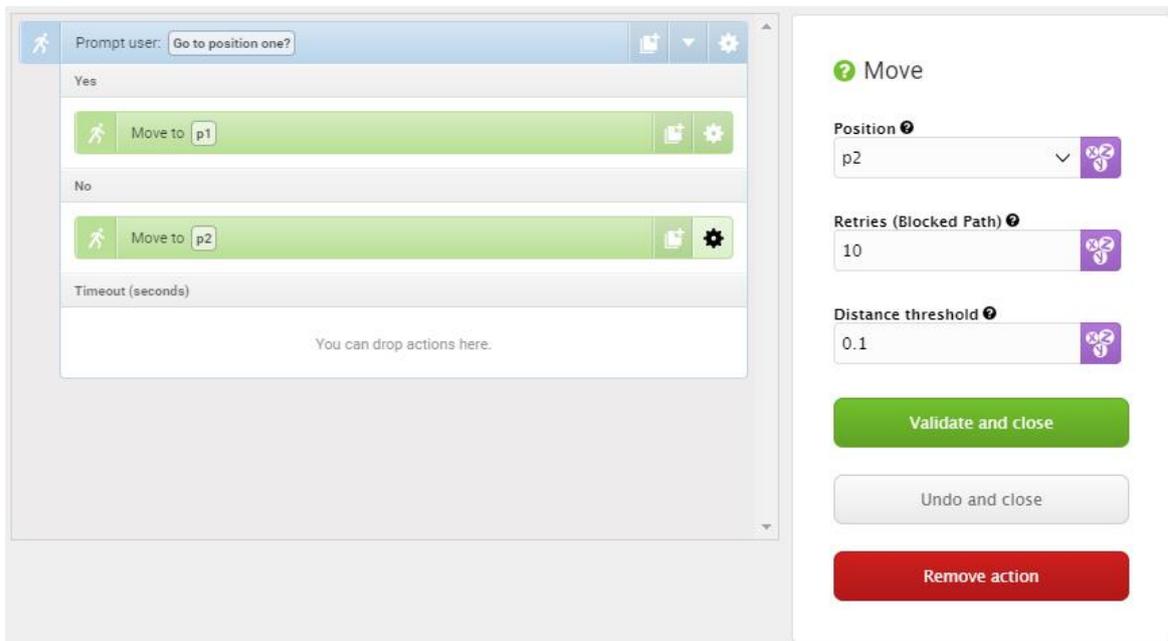


6. In the first Move to action, under **Position**, select **p1**.

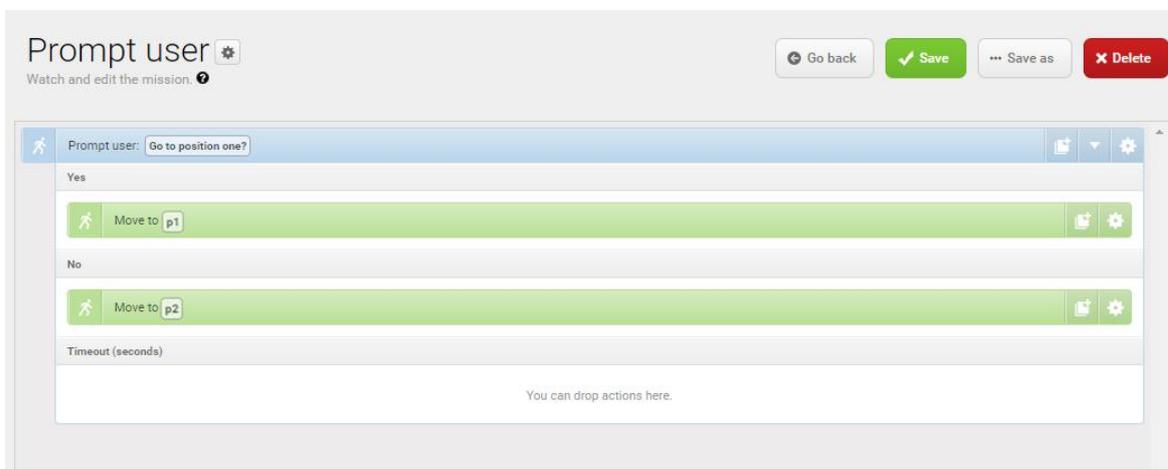


The screenshot displays the MiR software interface. On the left, a workspace contains a prompt bar at the top with the text "Prompt user: Go to position one?". Below this, there are two "Move to" actions. The first action is "Move to p1" and the second is "Move to c". A "Timeout (seconds)" section is visible below the actions, with the text "You can drop actions here." On the right side, a configuration panel for the selected "Move" action is shown. It includes a "Position" dropdown menu set to "p1", a "Retries (Blocked Path)" field set to "10", and a "Distance threshold" field set to "0.1". At the bottom of the configuration panel, there are three buttons: "Validate and close" (green), "Undo and close" (grey), and "Remove action" (red).

- In the second Move to action, under **Position**, select **p2**.



The mission should look like this:



- Select **Save** to save the mission.

## 10.4 Creating the mission *Try/Catch*

Try/Catch actions are used to handle mission errors. When you use a Try/Catch action, you can define what the robot should do if, at any point, it fails to execute its main mission. This prevents the robot from going into an error state and stopping in the middle of a mission by

providing an alternative course of action if the main mission fails.

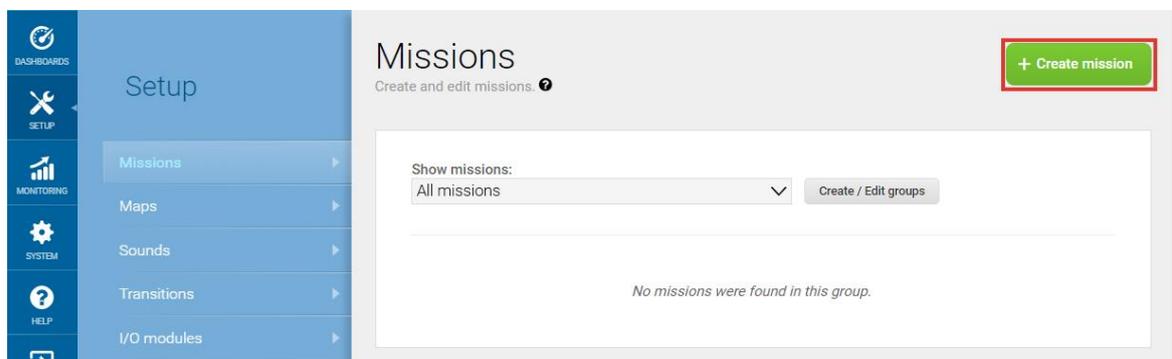
*Try/Catch* is a mission example where the robot runs the mission *Prompt user* created in [Creating the mission Prompt user on page 142](#), and if the robot for some reason fails to complete the mission, the robot plays a sound.

To create the mission *Try/Catch*, it is assumed you have completed the following:

- Created the mission *Prompt user* as described in [Creating the mission Prompt user on page 142](#).

To create the *Try/Catch* mission, follow the steps below:

1. Go to **Setup > Missions**. Select **Create Mission**.



2. Name the mission *Try/Catch*. Select the group and site you want it to belong to. Select **Create mission**.

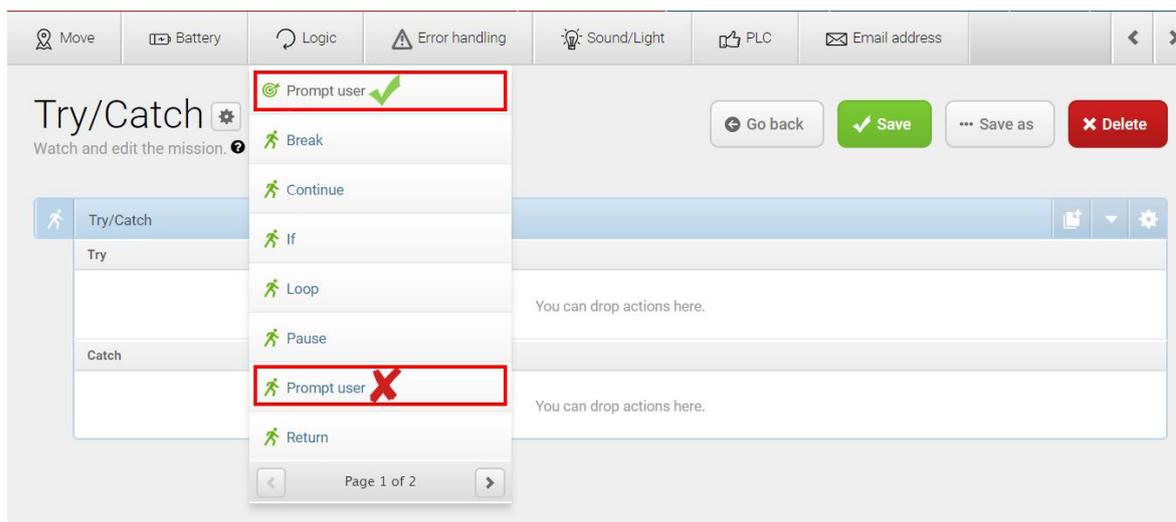
3. Select the following actions:

- In the Error handling menu, select **Try/Catch**.
- Select the Prompt user mission you have made.

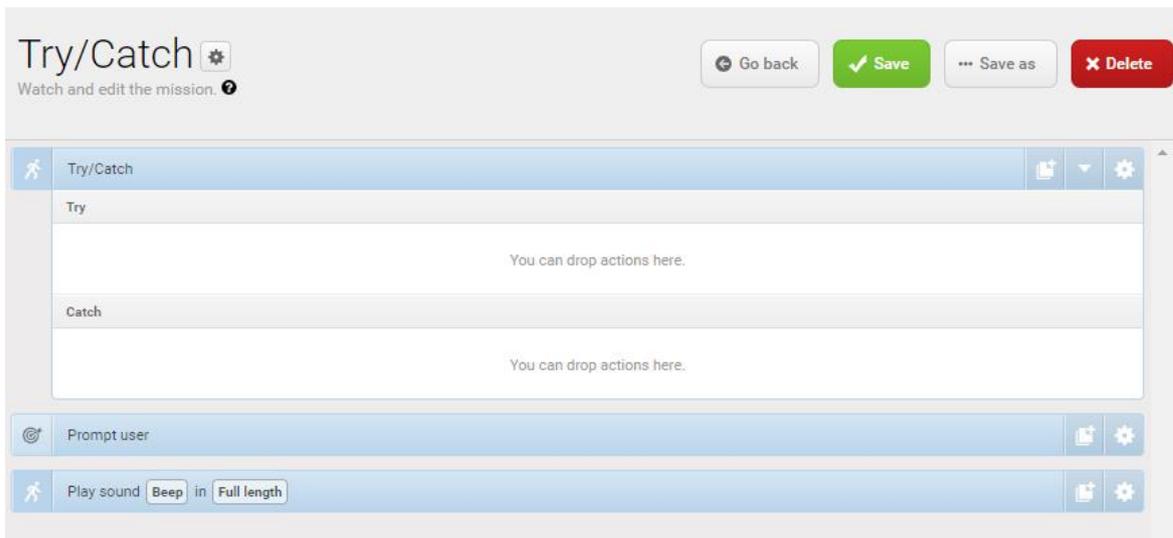
The mission menu you have saved the mission under will figure as a menu in the mission editor. The menus contain both missions and actions.



Missions have this icon and actions have this icon . In this example, the mission is saved under the Logic menu that also includes the Prompt user action . Be sure to select the Prompt user mission .

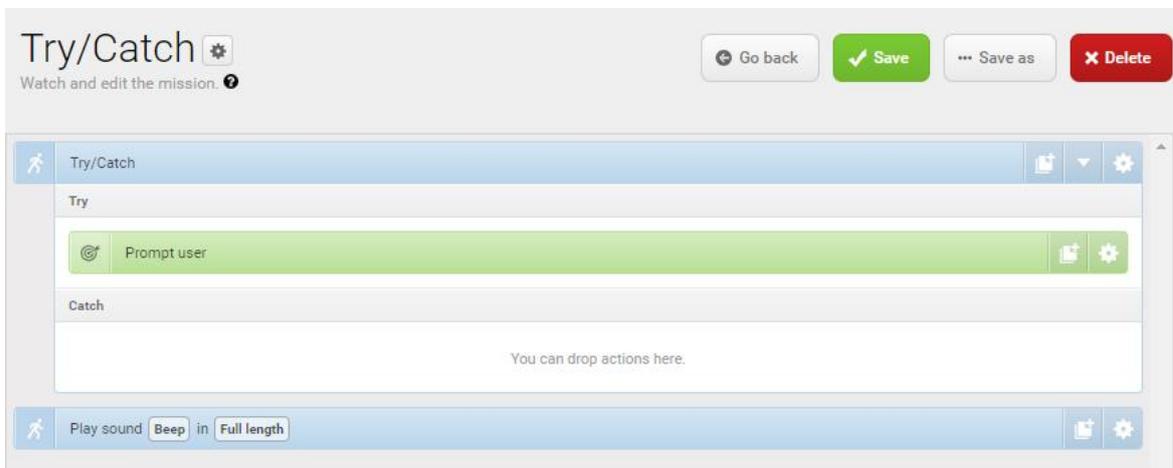


- In the Sound/Light menu, select **Play sound**.

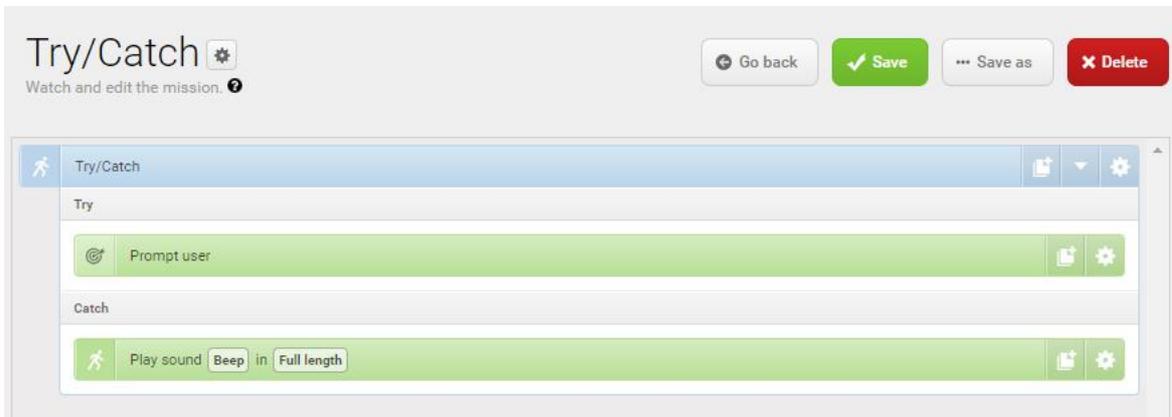


The following steps describe which parameters each action should be set to. To modify the parameters, select the gearwheel  at the right end of the action line to open the action dialog box. When you have set the parameters, select **Validate and close**.

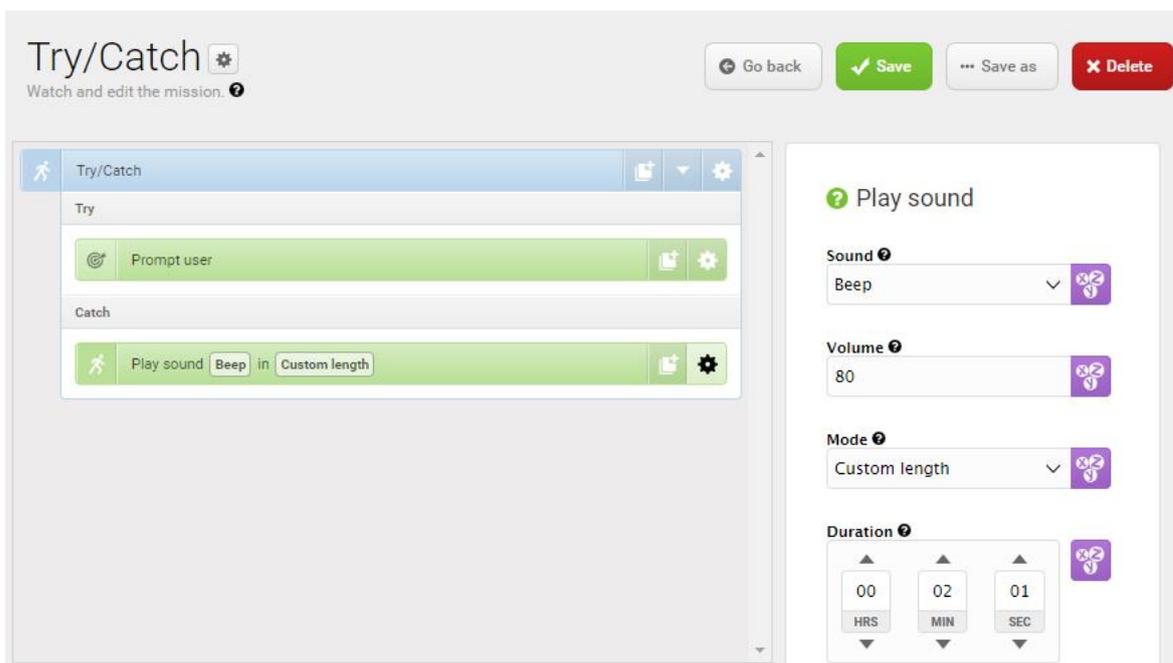
4. Drag the Prompt user mission into the **Try** box under **Try/Catch**.



5. Drag the Play sound action under the **Catch** box under **Try/Catch**.



6. In the Play sound action, set the parameters as follows:
  - **Sound:** Select **Beep**.
  - **Volume:** Enter the value **80**. This is approximately 64 dB.
  - **Mode:** Select **Custom length** so you can enter the duration of time the sound is played.
  - **Duration:** Set the duration to two minutes.



The mission should look like this:



7. Select ✓ **Save** to save the mission.

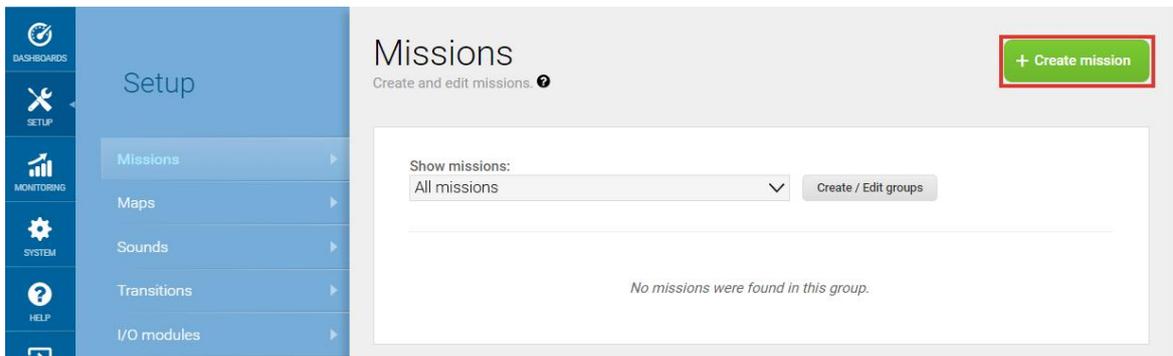
## 10.5 Creating the mission *Variable footprint*

All mission actions that require the user to specify the value of a parameter when they choose to use the mission have the option of defining a variable. If you use a variable in a mission, then when you add the mission to the mission queue or embed it inside another mission, you must select a value for the parameter where the variable is used. This allows you to reuse the same mission for different but still similar tasks.

*Variable footprint* is a mission example that enables you to select which footprint the robot should use each time the mission is run.

To create the mission, follow the steps below:

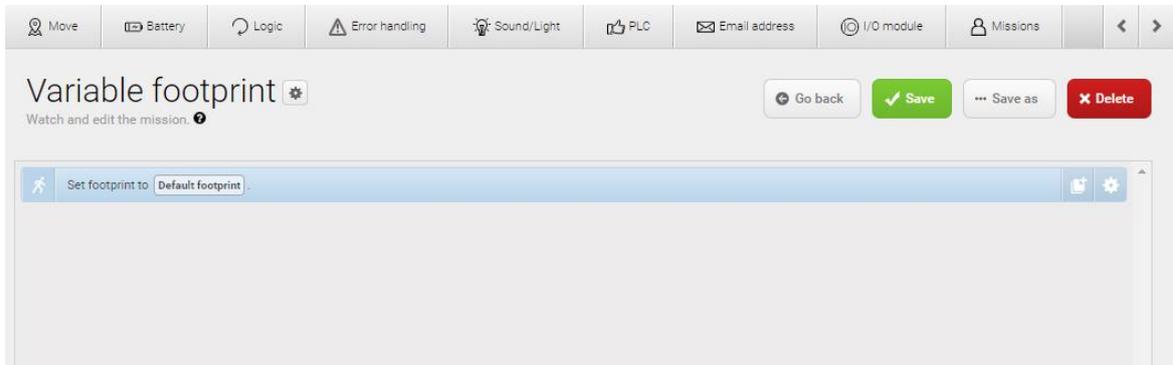
1. Go to **Setup > Missions**. Select **Create Mission**.



2. Name the mission *Variable footprint*. Select the group and site you want it to belong to. Select **Create mission**.

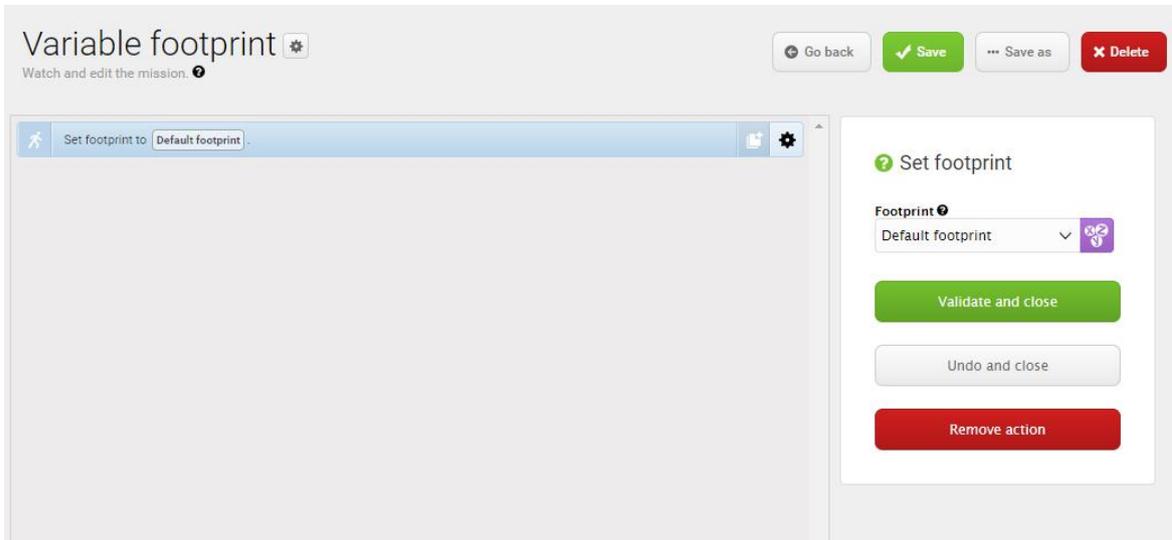
3. Select the following actions:

- In the **Move** menu, select **Set footprint**.



The following steps describe which parameters each action should be set to. To modify the parameters, select the gearwheel  at the right end of the action line to open the action dialog box. When you have set the parameters, select **Validate and close**.

4. In the **Set footprint** action, make the parameter **Footprint** a variable that can be set each time you use the mission. The following steps describe how to create a variable:
  - Under **Footprint**, select **Variables** .



- Select **Create variable** in the upper-right corner.
- Name the variable *Use default footprint or narrow footprint?*. Select **OK**.

## Create variable

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

### Variable name

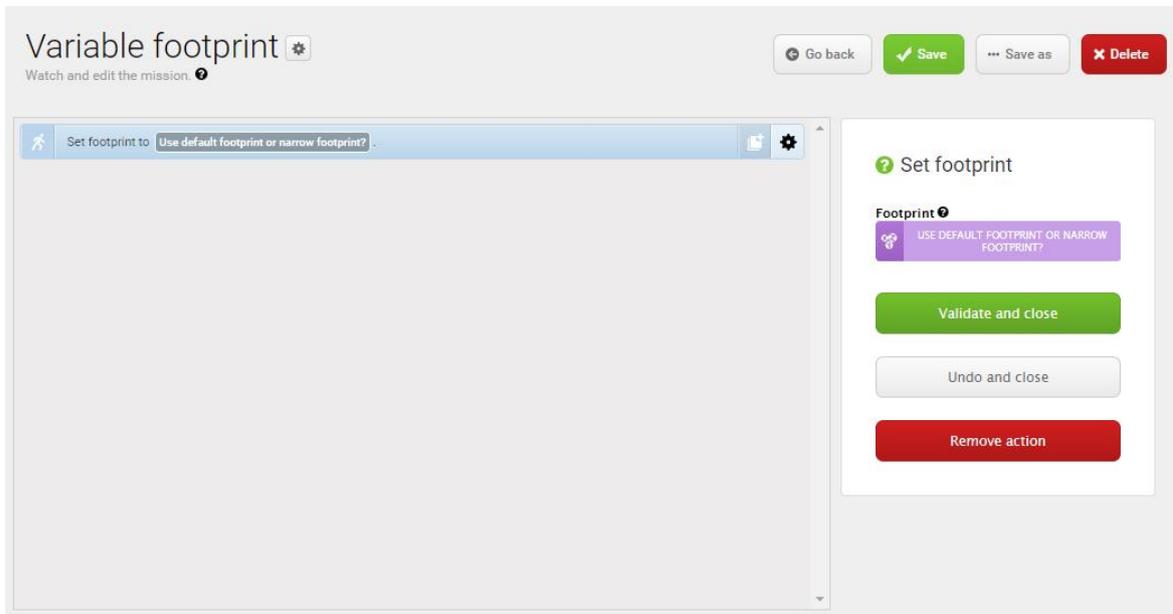
Use default footprint or narrow footprint?

### Default value

Default footprint



The mission should look like this:



5. Select ✓ **Save** to save the mission.

## 10.6 Testing a mission

After you create a mission, always run the mission to test that the robot executes it correctly.



### NOTICE

Always test missions without load to minimize potential hazards.

To run a mission, follow these steps:

1. Go to **Setup > Missions**.
2. Select **Queue mission**  next to the mission you want to run. The mission is now added to the mission queue.
3. Select **Continue**  to start the mission.
4. Watch the robot execute the mission, and verify that it performs as expected.



We recommend running the mission 5-10 times to ensure that it runs smoothly. If something interrupts the mission, use a Try/Catch action in that step of the mission and decide what the robot has to do if a mission action fails.

# 11. Applications

You can install top modules on top of MiR100 for specific applications.

For more information about top modules from MiR, see [MiR Top Modules](#) on the MiR website.

For more information about top modules from third parties, see [MiRGo](#) on the MiR website.

Top modules from MiR are delivered with instructions on how to mount and operate them with the robot.

For instructions on how to mount third party top modules and accessories, contact your distributor.

## 11.1 Mounting a top module

Top modules must be fastened using the self-tightening conically shaped mounting holes in each corner of the robot and should be mounted with a tightening torque of 47 Nm.



Figure 11.1. Top modules are fastened through the mounting holes in the top cover.



**CAUTION**

Personnel operating the robot risk injury if they cannot stop the robot in an emergency situation.

- If a top module prevents you from connecting the Emergency stop button delivered with your robot, make sure to install a new button on the top module, and to perform a risk assessment according to standard ISO 12100.



**CAUTION**

Certain top modules may lead to new hazards and increased risks that cannot be eliminated or reduced by the risk reduction measures applied by Mobile Industrial Robots.

- Perform a risk assessment according to standard ISO 12100 when mounting a top module—see [Risk assessment on page 91](#).



**CAUTION**

MiR100 may tip over if weight and payload specifications are not met, risking damage to equipment or injury to nearby personnel.

- Stay within the specifications for weight and the total payload's center of gravity—see [Payload distribution on page 168](#).

# 12. Maintenance

The following maintenance schedules give an overview of regular cleaning and parts replacement procedures.

It is the responsibility of the operator to perform all maintenance tasks on the robot.



The stated intervals are meant as guidelines and depend on the operating environment and frequency of usage of the robot.



It is recommended to make a maintenance plan to make sure that all maintenance tasks are done and that the responsible(s) are aware of their tasks.



See the sections [External parts on page 12](#), [Internal parts on page 18](#), and [Accessing the internal parts](#) to see where the parts mentioned in the maintenance schedules are located and how to access them.



## NOTICE

Only use approved spare parts. Contact your distributor for the list of spare parts and the appropriate how-to guides.

Mobile Industrial Robots disclaims any and all liability if unapproved spare parts are used. Mobile Industrial Robots cannot be held responsible for any damages caused to the robot, accessories, or any other equipment due to use of unapproved spare parts.

## 12.1 Regular weekly checks and maintenance tasks

Once a week, carry out the maintenance tasks in **Table 12.1**.

Table 12.1. Regular weekly checks and maintenance tasks	
Parts	Maintenance tasks
Robot cover and sides	<p>Clean the robot on the outside with a damp cloth.</p> <p><b>i</b> Do not use compressed air to clean the robot.</p>
Laser scanners	<p>Clean the optics covers of the scanners for optimum performance. Use anti-static cleaning products, and avoid aggressive or abrasive cleaning agents.</p> <p><b>i</b> Make sure to also clean the top part of the optics cover. The contamination sensors scan the entire optics cover glass piece.</p> <p>In the robot interface under <b>Monitoring &gt; Hardware health &gt; Emergency stop</b>, see if <b>Front scanner cover</b> and <b>Back scanner cover</b> is <b>Clean</b>.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p><b>⚠ NOTICE</b></p> <p>Static charges cause dust particles to be attracted to the optics cover. You can diminish this effect by using the anti-static plastic cleaner (SICK part no. 5600006) and the SICK lens cloth (part no. 4003353). See the manufacturer's own documentation.</p> </div>
Caster wheels (the four corner wheels)	<p>Remove dirt with a damp cloth, and make sure nothing is entangled in the wheels.</p>
Drive wheels (the two middle wheels)	<p>Remove dirt with a damp cloth, and make sure nothing is entangled in the wheels.</p> <p>Wipe away an grease that may have leaked from the roller bearings.</p>
Status lights	<p>Check if the LED light band is intact. Ensure the light shows all the way around the robot. Clean with a soft cloth to ensure even lighting around the robot.</p>

## 12.2 Regular checks and replacements

Before starting replacement tasks that involve removal of the top cover:

- Shut down the robot—see [Shutting down the robot on page 49](#).
- Disconnect the battery—see [Disconnecting the battery on page 52](#).

**Table 12.2** contains the parts that you should check and how often you should do that.

Table 12.2. Regular checks and replacements		
Part	Maintenance	Interval
Robot cover	<p>Check for cracks.</p> <p>Check mounting. Does the cover sit evenly on top of the robot with all connections accessible?</p>	Check monthly and replace as needed.
Safety PLC	In the robot interface under <b>Monitoring &gt; Hardware health &gt; Communication</b> , see if the robot is running with the correct SICK configuration or if the warning <b>The SICK Safety PLC is running a non-standard configuration</b> is shown.	Check monthly and after commissioning or if you make any changes to the robot setup.
Robot hardware	In the robot interface under <b>Monitoring &gt; Hardware health</b> , check if there are any warnings (marked with yellow).	Check monthly and after commissioning or if you make any changes to the robot setup.
Loudspeaker	Check that all auditory warnings function.	Check monthly, and replace as needed.
Caster wheels (the four corner wheels)	Check bearings and tighten, and check the wheels for wear and tear.	<p>Check weekly, and replace as needed.</p> <p>Replace all four caster wheels together. See the guide <i>How to</i></p>

Part	Maintenance	Interval
		<i>replace the caster wheels on MiR100 and MiR200.</i>
Drive wheels (the two middle wheels)	Check wheel surfaces for wear.	<p>Check every six months, and replace as needed.</p> <p>Replace in pairs. See the guide <i>How to replace the drive wheels on MiR100/MiR200.</i></p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p><b>⚠ NOTICE</b></p> <p>The robot's IMU must be calibrated after replacement of the wheels—see <i>How to calibrate the IMU.</i></p> </div>
Drive wheels (the two middle wheels)	Lubricate the roller bearings.	Lubricate the roller bearings yearly.
Safety laser scanners	Check for visual defects, for example cracks and scratches.	<p>Replace as needed. See the guide <i>How to replace a safety laser scanner on MiR100 and MiR200.</i></p> <div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p><b>⚠ NOTICE</b></p> <p>The robot must be calibrated after replacement of the scanners—see <i>How to calibrate the safety laser scanners.</i></p> </div>
Emergency stop	Check that the Emergency stop button works. Push down the	Every three to four months / according to <i>EN/ISO 13850 Safety</i>

Part	Maintenance	Interval
	red button, and check that the Emergency stop reset button lights up and that the status lights turns red.	<i>of machinery - Emergency stop function. See the guide <a href="#">How to replace the emergency stop interface on MiR100/MiR200</a>.</i>
Pad connectors	Check for dust or dirt, and clean with a soft cloth.	Check monthly.  <b>i</b> Over time the pad connectors may become discolored due to corrosion. This is merely a cosmetic issue and has no effect on the conductivity of the pad connectors. You can remove the corrosion using common methods of corrosion removal for silver.
3D cameras	Check for visual defects, for example cracks and scratches.  To test the cameras, see the guide <a href="#">How to test if the 3D cameras are working on MiR robots</a>	Check monthly, and calibrate or replace as needed— see the guide <a href="#">How to calibrate a D435 3D camera</a> .
Safety stickers and nameplate	Check if the safety stickers, identification label, and nameplate on the robot are still intact and visible.	Check every six months, and replace as needed.

**CAUTION**

If the robot has been impacted, it may be structurally damaged, causing a risk of malfunction and injury to personnel.

- If you suspect the robot has suffered any damage, you need to conduct a thorough inspection to ensure that the robot's strength and structure is not compromised.

## 12.3 Battery maintenance

The battery is generally maintenance-free but should be cleaned if it gets very dirty—see *How to check the battery on MiR robots*. Before cleaning, the battery must be removed from any power source. Only use a dry and soft cloth to clean the housing of the battery, and do not use abrasives or solvents. See the guide *How to replace the lithium battery in MiR100/MiR200*.

For storage of the battery, see [Battery storage on page 56](#).

For disposal of the battery, see [Battery disposal on page 56](#).

# 13. Packing for transportation

This section describes how to pack the robot for transportation.

## 13.1 Original packaging

Use the original packaging materials when transporting the robot.



Figure 13.1. The packing materials.

The packaging materials are:

- The bottom of the box (the pallet)
- The lid of the box (the ramp)
- The walls of the box
- Protective foam blocks: Side blocks and the top layer
- Protective corner braces. The braces prevent the robot from being damaged by the transport straps

## 13.2 Packing the robot for transportation

Before packing the robot for transportation:

- Shut down the robot—see [Shutting down the robot on page 49](#).
- Disconnect the battery—see [Disconnecting the battery on page 52](#).

To pack the robot, repeat the steps in [Unpacking MiR100 on page 32](#) in the reverse order.



### NOTICE

Pack and transport the robot in an upright position. Packing and transporting the robot in any other position voids the warranty.

## 13.3 Battery

The lithium-ion battery is subject to transport regulations. Make sure that you follow the safety precautions in this section and the instructions in [Packing for transportation on the previous page](#). Different regulations apply depending on the mode of transportation: land, sea, or air.

Contact your distributor for more information.



### CAUTION

Lithium-ion batteries are subject to special transportation regulations according to United Nations Regulation of Dangerous Goods, UN 3171. Special transport documentation is required to comply with these regulations. This may influence both transport time and costs.

## 14. Disposal of robot

According to the European directive 2012/19/EU, Article 2, paragraph 4) d) and 4) e) (WEEE directive), MiR's robots are not in scope of the directive.

The robot is therefore not classified as WEEE and can be disposed of in accordance with the applicable local regulations.



### NOTICE

It is important to disconnect and remove the battery before disposal of the robot.

For information on how to disconnect the battery, see [Disconnecting the battery on page 52](#).

For battery disposal, see [Battery disposal on page 56](#).

# 15. Payload distribution

The following drawings illustrate where the center of mass (CoM) of payloads must be located for safe operation with different payloads.



## CAUTION

Load placed directly on top of the robot cover may cause damage to the cover of the robot.

- Ensure that the load is not placed directly on top of the robot cover.



## WARNING

Load falling or robot overturning if the load on MiR100 is not positioned or fastened correctly can cause damage to equipment and injury to personnel.

- Ensure that the load is positioned according to the specifications and is fastened correctly.



## CAUTION

Bumps and holes can cause loads to fall off of the robot, causing damage to equipment and injury to personnel.

- The floor the robot drives on must be even without bumps and holes for the payload specifications to be valid. If bumps and holes are present, the commissioner must take additional measures to ensure a safe operation.



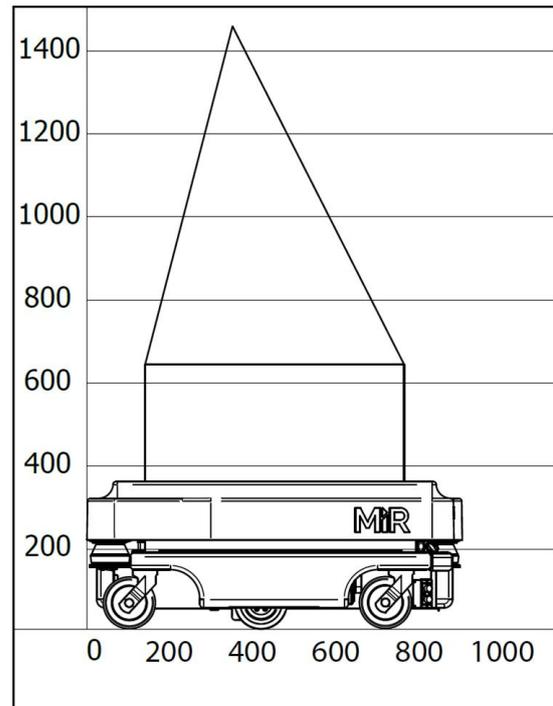
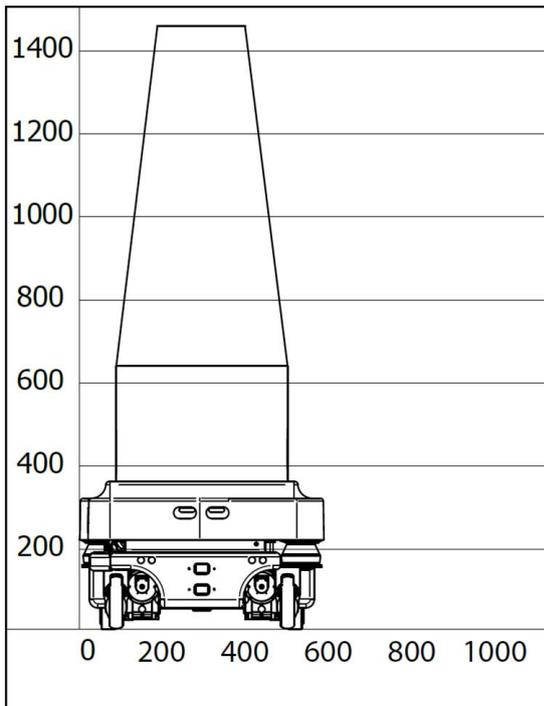
**CAUTION**

A slippery surface can cause instability when driving with payloads, causing damage to equipment and injury to personnel.

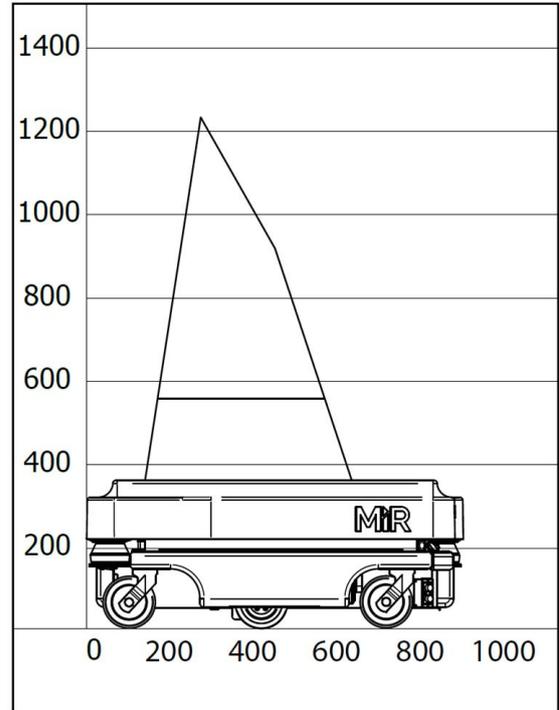
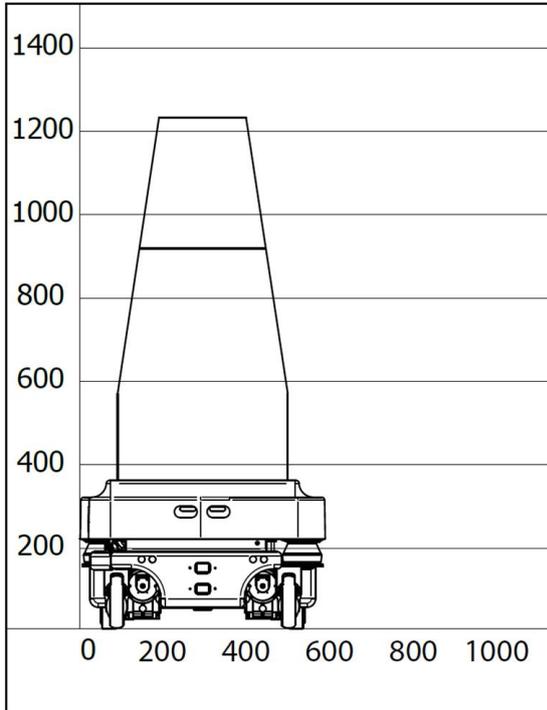
- The friction coefficient between the floor and the drive wheels on the robot must be in the range of 0.60 to 0.80 for the payload specifications to be valid. If the friction coefficient is not in this range, the commissioner must take additional measures to ensure a safe operation.

The specifications apply to total payloads of up to 100 kg.

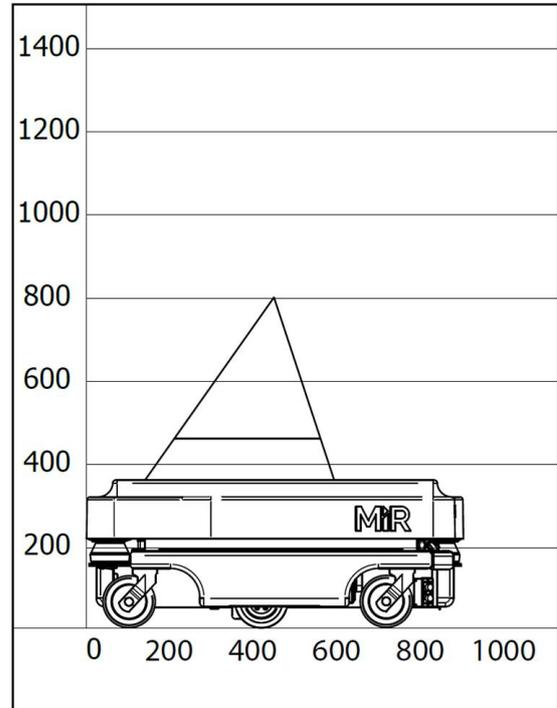
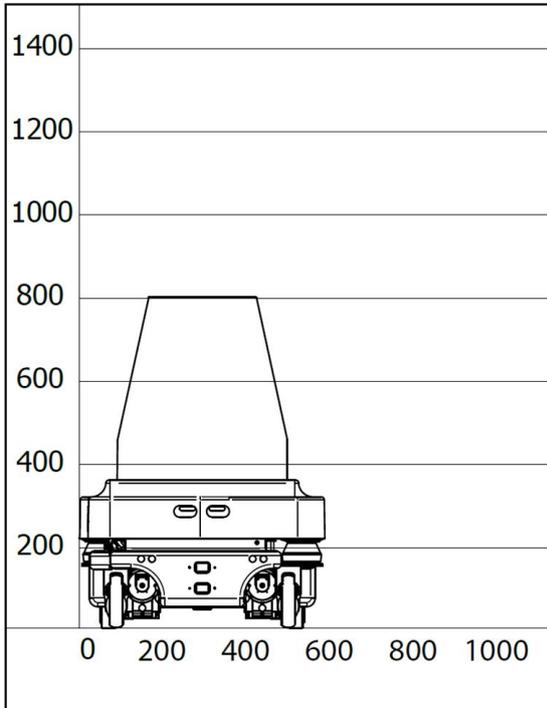
Payload: 25 kg



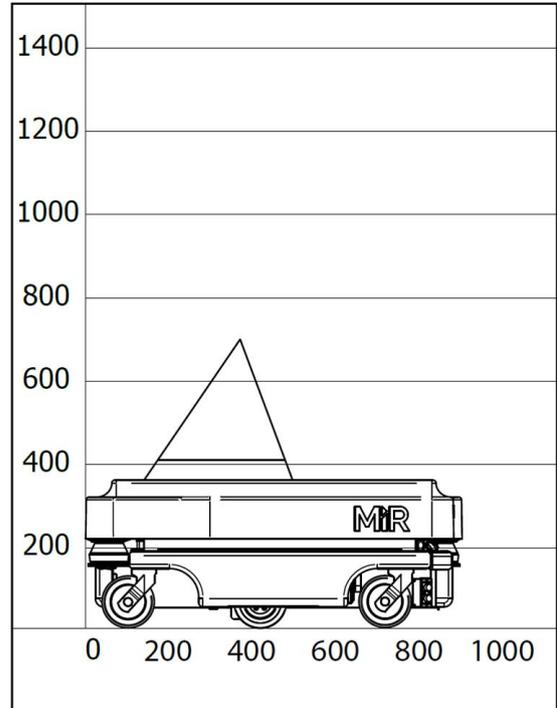
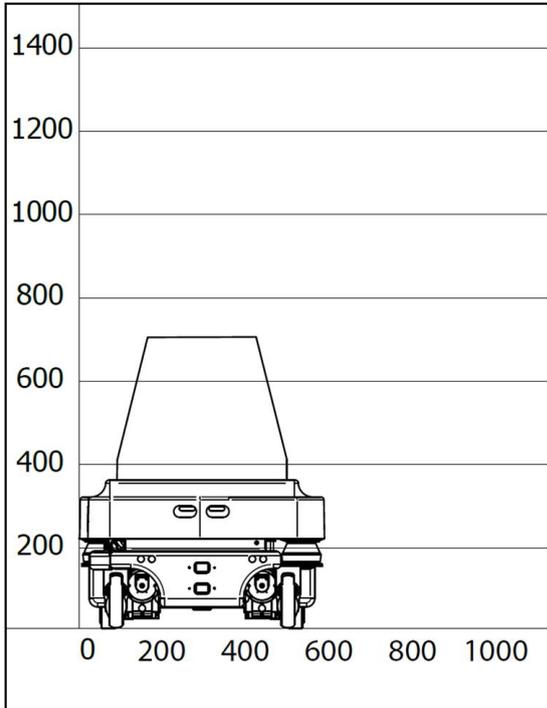
Payload: 50 kg



Payload: 75 kg



Payload: 100 kg



# 15. Interface specifications

This section describes the specifications of the top application interfaces.



**NOTICE**

Read [Safety on page 21](#) before using the electrical interface.

## 15.1 Application interface

The application interface plug is a NEUTRIK XLR panel-mount connector with 4 contacts (receptable).

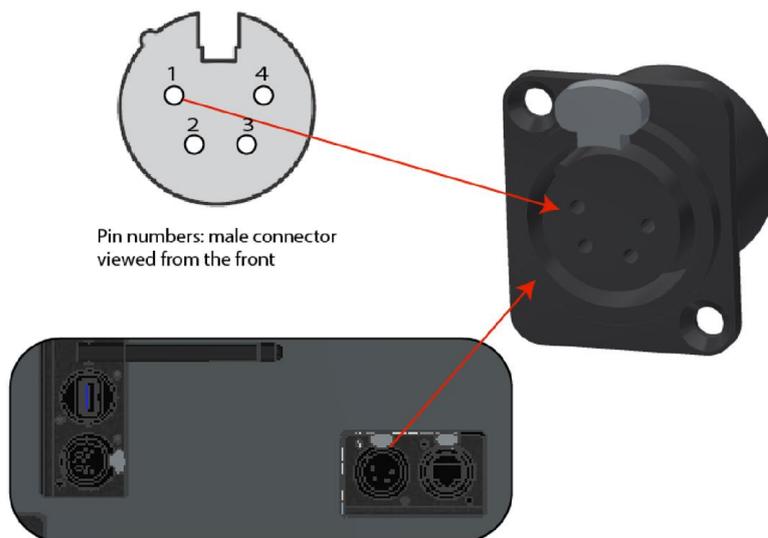


Figure 15.1. Application interface.

Table 15.1. Description of the pins in <i>Figure 15.1</i>			
Pin no.	Voltage	Max. current	Description
1	Battery voltage (24 V)	3 A	Starts with the robot.

Pin no.	Voltage	Max. current	Description
2	Battery voltage (24 V)	3 A	Starts with the robot.
3	Battery voltage (24 V)	10 A	Stops by Emergency stop.
4	GND	10 A	Ground.

## 15.2 Emergency stop

The emergency stop plug is a NEUTRIK XLR panel-mount connector with 10 contacts (receptable).

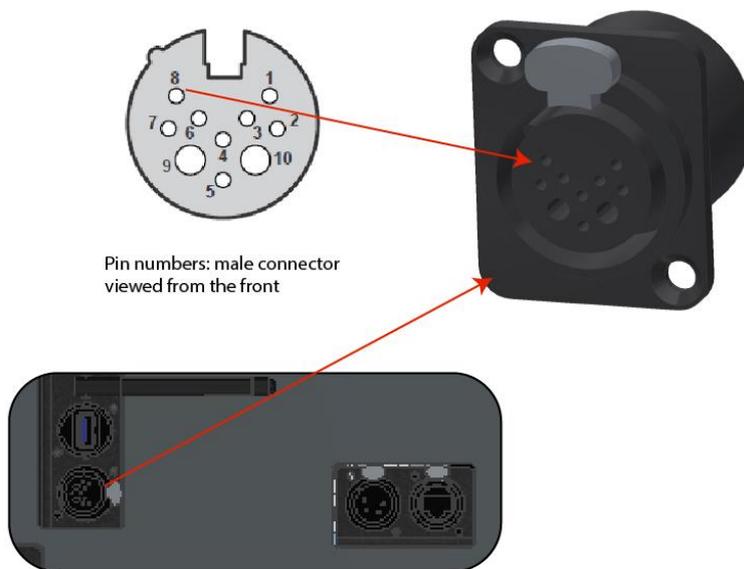


Figure 15.2. Emergency stop interface.

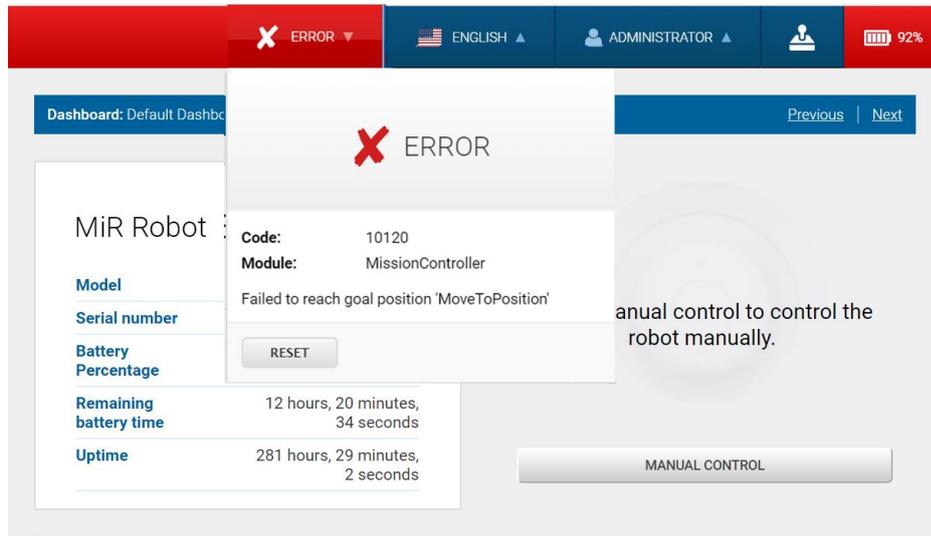
**Table 15.2.**  
Description of the pins in *Figure 15.2*

Pin no.	Signal name	Description
1	GND 24 V	GND for lamp in Scanner reset button.
2	X1 SICK	Test output from the robot safety PLC.
3	X2 SICK	Test output from the robot safety PLC.
4	Emergency stop, 1 GN	Safety input on robot safety PLC monitoring through Emergency stop button.
5	Emergency stop, GN-WH/RD	Safety input on robot safety PLC monitoring through Emergency stop button.
6	Reset button, BWN-WH	Safety input on robot safety PLC monitoring state of Scanner reset button used to release robot from Emergency stop.
7	RESET LAMP	Safety output from the robot safety PLC used to turn the lamp in the reset button On/Off.
8	I5	Connects to safety input I5 on robot safety PLC.
9	GND	24 V/1 A DC supply - NEG terminal. To be used as GND for 24V supply from pin 10.
10	24 E-STOP	24 V/1 A DC supply - POS terminal. Can be used to power small external units using up to 1 A such as tablets and PLC interfaces.

To see the locations of the electrical interfaces on the robot, see [Internal parts on page 18](#).

# 16. Error handling

The robot enters an error state when it can't solve a problem on its own.



Errors include:

- Hardware faults
- Failed localization
- Failure to reach destination
- Unexpected events in the system

An error triggers a Protective stop. The robot is paused until a user acknowledges the error and clears it.

## 16.1 Software errors

Software errors such as localization and failure to reach the goal destination can be prevented with the proper setup of maps and missions:

- Always test your missions under full observation and normal work environment conditions before leaving the robot to execute the missions autonomously—see [Testing a mission on page 155](#).

- Use Try/Catch actions to make the robot react in a specific way if it fails to execute certain actions—see [Creating the mission Try/Catch on page 146](#).
- Use Prompt user actions in missions that require intervention from users—see [Creating the mission Prompt user on page 142](#).
- Define forbidden areas with Forbidden or Unpreferred zones on the map—see [Creating and configuring maps on page 92](#).
- Remove noise from maps—see [Creating and configuring maps on page 92](#).
- Create Directional or Preferred zones to guide the robot around areas that are difficult for the robot to travel through—see [Creating and configuring maps on page 92](#).

To clear an error, select the red warning indicator in the interface, and select **Reset**.

For more details on setting up missions and error handling, see *MiR Robot Reference Guide* on the MiR website.

## 16.2 Hardware errors

If the error is a fault in the hardware, either you will not be able to clear it, or the error will return until the fault is fixed. If this occurs, you can try to fix the issue with these actions:

- Turn your robot off and then on again. This resets the robot components and may resolve the issue.
- Check that the Emergency stop button is released.
- Check your robot for any physical damage such as cracks, dents, or severe scratches or contamination such as dust, dirt, and grease. Pay special attention to the 3D cameras, safety laser scanners, and drive wheels.
- Sign in to the robot interface and go to **Monitoring > Hardware health**. The interface displays which component is failing and often for what reason. This can help identify the source of the error. If an internal component is failing, turn off the robot, disconnect the battery, and have the commissioner or operator visually check the internal component for obvious faults.



Figure 16.1. The interface in Hardware health displays which component is failing and often for what reason.

- For further troubleshooting, contact your distributor for specific MiR troubleshooting guides or assistance from MiR Technical Support.



For a full list of MiR error codes, contact your distributor for the document *Error codes and solutions*.

# 17. Glossary

---

## A

---

### Autonomous mode

Mode in which the robot drives autonomously based on the missions you assign to it.

## C

---

### Cart

A cart can be towed by a MiR robot with a MiR hook mounted to it.

### Commissioner

Commissioners have thorough knowledge of all aspects of commissioning, safety, use, and maintenance of MiR100 and have the following main tasks: commissioning the product, including creating maps and restricting the user interface for other users; making the risk assessment; determining the payload limit, weight distribution, and safe methods of fastening of loads to MiR100; and ensuring the safety of nearby personnel when a MiR robot is accelerating, braking, and maneuvering.

## D

---

### Direct user

Direct users are familiar with the safety precautions in the user guide and have the following main tasks: assigning missions to MiR100, fastening loads to MiR100 securely, and loading and unloading from a paused robot.

### Dynamic obstacle

Dynamic obstacles are obstacles that are moved around, such as pallets, crates, and carts. These should not be included when creating a map.

## E

---

### Emergency stop

Emergency stop is a state the robot enters when an Emergency stop button has physically been pressed. When the robot is in Emergency stop, the status light of the

robot turns red, and you are not able to move the robot or send it on missions until you bring the robot out of Emergency stop. To do this, you must release the Emergency stop button and then press the Resume button.

---

## F

### Footprint

A footprint set in the robot interface defines the size of the robot and its top module or load. It consists of a horizontal shape around the robot that should be slightly bigger than the robot itself to ensure that there is enough space along the calculated path, and a maximum height that ensures the robot does not try to drive under low obstacles. The horizontal shape is defined as coordinates relative to the robot's center.

---

## G

### Global path

The global path is the route the robot calculates that leads it to its goal position.

---

## I

### Identification label

The identification label is the label that is mounted to the product in production. The label is used to identify the components in your MiR application. It identifies the product model, the hardware version, and the product serial number.

---

## L

### Light indicators

Signal light and/or audial warnings. The signal lights are by default blinking yellow when the protective fields are muted. Audial warnings can be added in System settings.

### Local path

The local path is the route the robot creates within its immediate vicinity that guides it around obstacles while still following the global path.

### Localization

The method used by the robot to determine its position on the map relative to where it is in the work environment.

## M

---

### Manual mode

The mode in which you can drive the robot manually using the joystick in the robot interface.

### Mapping

Mapping is the process of manually driving the robot around to create a new map. The robot uses its sensors to detect walls and obstacles and generates a map of the area based on this input.

### Marker

A marker of a physical entity that the robot can dock to. This enables the robot to position itself accurately relative to the marker.

### MiR application

A MiR application is either a single MiR product or a combination of MiR products that is able to execute certain tasks. A MiR application is often a MiR base robot combined with a MiR top module. If a custom top module is used, the CE mark on the nameplate of the base robot does not extend to the top module.

### MiR robot interface

The MiR robot interface is the web-based interface that enables you to control your MiR robot. It is accessed by connecting to the same network as the robot and entering the robot's IP address in a browser.

## N

---

### Noise

With MiR robots, noise in maps 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.

---

## O

---

### Operating hazard zone

Operating hazard zones are areas with inadequate clearance for personnel to work close by the robot.

### Operator

Operators have thorough knowledge of MiR100 and of the safety precautions presented in the User guide of MiR100. Operators have the following main tasks: servicing, maintaining, and creating and changing missions and map positions in the robot interface.

---

## P

---

### Payload

The payload is the weight the robot carries. The maximum payload is the maximum weight the robot can carry, including the weight of any top modules, shelves, carts, or other devices.

### Position

A position is a set of X-Y coordinates on the map that you can send the robot to.

### Protective field sets

The Protective fields sets are a part of the Personnel detection safety function. They are contours surrounding the robot that change size depending on the speed of the robot. When a safety laser scanner detects a person or object within the active field, the robot enters Protective stop until the field is clear.

### Protective stop

Protective stop is a state the robot enters automatically to ensure the safety of nearby personnel. When the robot enters Protective stop, the status light of the robot turns red, and you are not able to move the robot or send it on missions until it is brought out of Protective stop. The robot goes into Protective stop in a number of situations: if a safety laser scanner detects an object in its active protective field, when the robot finishes the startup process, when the robot has switched between Manual mode and Autonomous mode, if the safety system detects a fault, or if the motor control system detects a discrepancy.

---

## R

---

### REST API

REST API is used by MiR Fleet and MiR robots to communicate status data and orders.

## S

---

### Site

A site is the overall container of any number of maps and all data relevant to the map, such as missions, features, robot groups, and transitions.

### Static landmark

Static landmarks are obstacles that cannot be moved, such as walls, columns, and fixed structures. These must be included on the map and are used by the robot to localize itself.