INDUSTRIAL COBOTICS

iSee
Ui

iSee Ui Builder

User manual

v1.5

Original Instructions
Date: 2023, August

Contents

1. General information

1.1 Purpose

The purpose of the iSee Ui Builder is to provide an easy and user-friendly means for creating a custom iSee Ui user interface that can be loaded and used in the UR e-series robot by means of the iSee Ui URCap.

1.2 Company details

Industrial Cobotics BV
Atomveldstraat 10 - bus 2
9450 Haaltert
Belgium
Tel: +32 27 93 10 41
mail: info@industrialcobotics.be

1.3 Disclaimer

Industrial Cobotics continues to improve reliability and performance of its products, and therefore reserves the right to upgrade the product without warning. Industrial Cobotics takes every care that the contents of this manual are precise and correct, but takes no responsibility for any errors or missing information.

2. iSee Ui Builder

2.1 Intro

The iSee Ui Builder is a software intended for creating and editing iSee Ui interface files (.isui files). The software provides an easy-to-use builder, to create and design a custom user interaction dashboard. This created dashboard is then saved as a .isui file, which in turn can be loaded into the iSee Ui URCap on the UR e-series robot.

The Builder software does not provide a means to program the interaction once it is opened on the UR robot. The action the robot program must perform when, e.g. a push button is pressed, is programmed in the UR program itself. The Builder software, however, does allow specifying to display the live value of a variable or IO, or to write a new value to a variable or IO during program run. Chapter 5 will explain this in more detail.

2.2 Compatibility

This iSee Ui Builder version is compatible with the iSee Ui URCap version 1.4 and iSee Ui License Server version 1.1. The iSee Ui URCap is applicable to the UR e-series, version 5.6+.

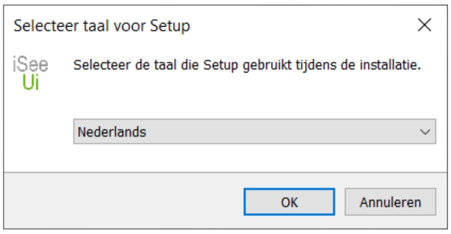
2.3 Requirements

To install and execute the iSee Ui Builder, the following requisites are needed:

- OS Windows: 10 or 11; or Linux or Mac
- UR e-series software version: 5.6 or higher

2.4 Installation

The iSee Ui Builder software can be installed on your computer by double-clicking the setup file, run as administrator if needed, then follow the instructions shown on your computer screen:

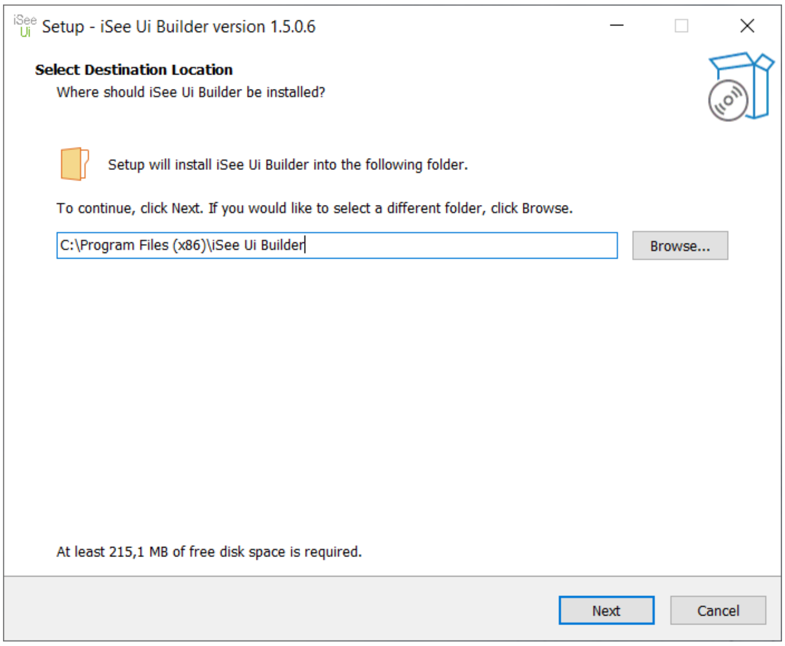


Selecteer taal voor Setup

Selecteer de taal die Setup gebruikt tijdens de installatie.

Nederlands

Select the setup language



Select Destination Location

Where should iSee Ui Builder be installed?

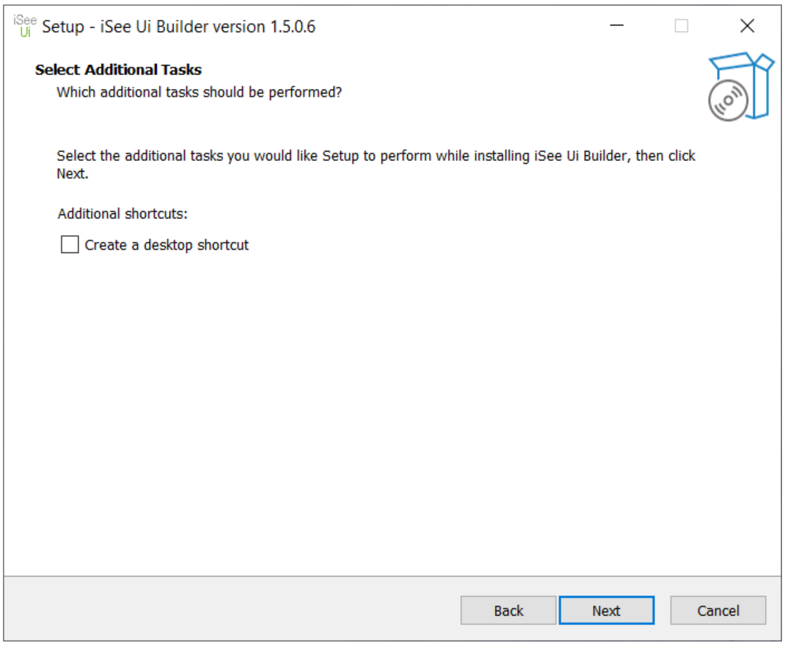
Setup will install iSee Ui Builder into the following folder.

To continue, click Next. If you would like to select a different folder, click Browse.

C:\Program Files (x86)\iSee Ui Builder

At least 215,1 MB of free disk space is required.

Define the installation directory for the software



Select Additional Tasks

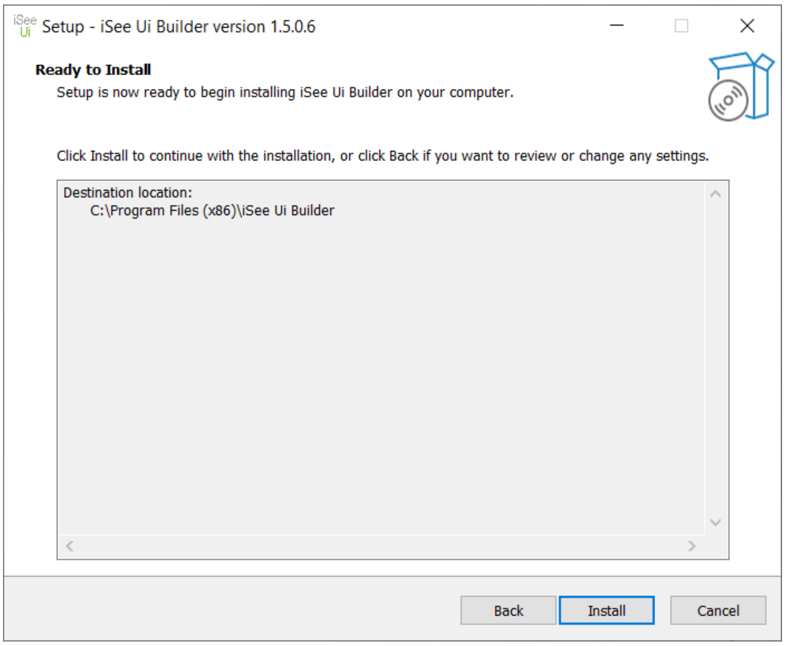
Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing iSee Ui Builder, then click Next.

Additional shortcuts:

□ Create a desktop shortcut

Choose if you want to create a desktop shortcut



Setup - iSee Ui Builder version 1.5.0.6

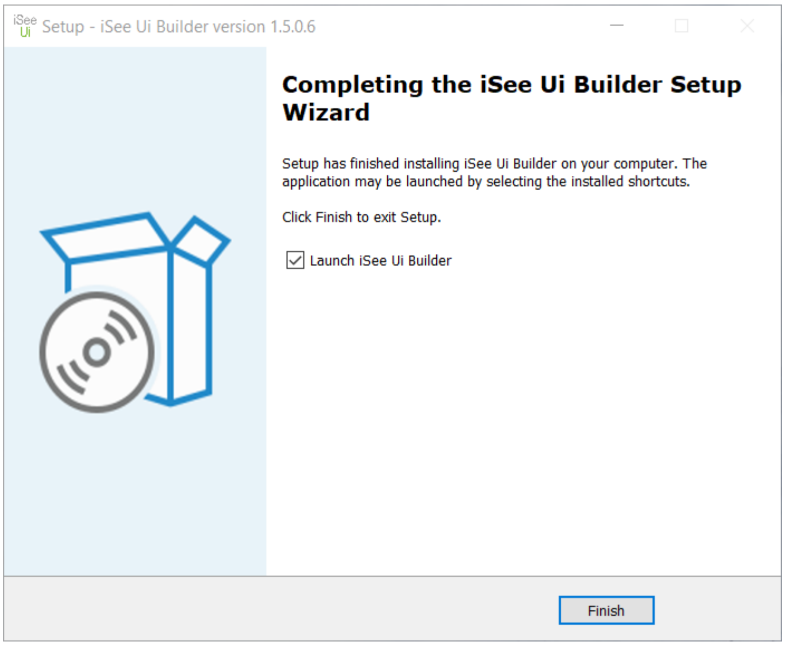
Ready to Install

Setup is now ready to begin installing iSee Ui Builder on your computer.

Click Install to continue with the installation, or click Back if you want to review or change any settings.

```
Destination location:
      C:\Program Files (x86)\iSee Ui Builder
```

Begin the installation



Setup - iSee Ui Builder version 1.5.0.6

Completing the iSee Ui Builder Setup Wizard

Setup has finished installing iSee Ui Builder on your computer. The application may be launched by selecting the installed shortcuts.

Click Finish to exit Setup.

☑ Launch iSee Ui Builder

Complete installation and run the software

2.5 Caution

Never make changes to the “License” folder in the installation directory!
Any changes to the folder or its content, by renaming, adding, deleting, copying, or any other action, will revoke the validity of all licenses and render them unavailable!

3. License

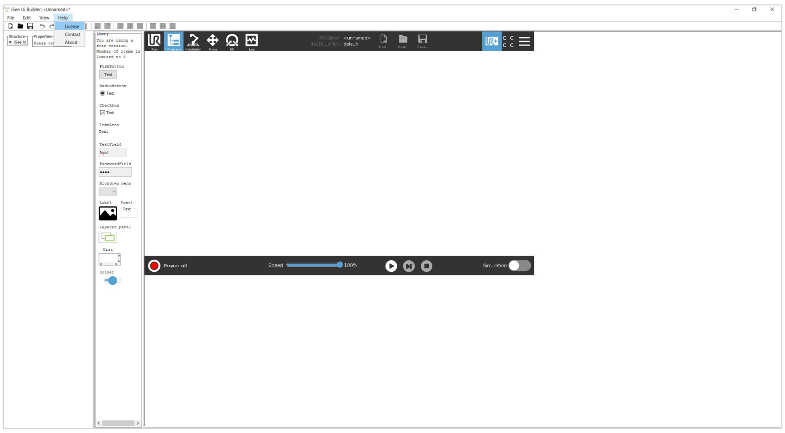
3.1 Free and trial version

In projects where an interface with the company logo, some buttons and/or text is needed, the software can be used for **free**, and no license is needed. This free version is limited to max 8 items.

If the desired interface requires more than 8 items, a license will have to be purchased or a trial license can be requested for 14 days.
The interface created in the Builder with a trial version can only be used with a trial license for the URCap. In order to use the interface in the URCap with a purchased license, the interface will need to be opened in the Builder with a purchased license as well.

3.2 License types

If more than 8 items are needed, a license needs to be activated. This can be done by opening the “Help” menu, and then pressing the “License” button (see figure 3.1). A window will open where you can select to either use a fixed or a floating license and view the current state of the license.



You are using a free version. Number of items is limited to 8

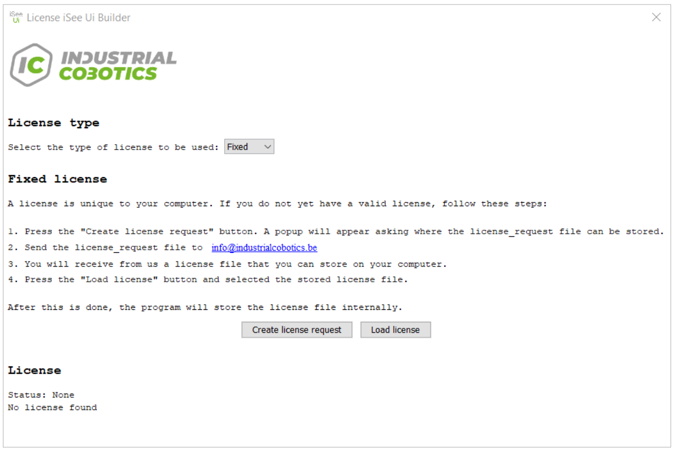
Figure 3.1: Opening the license window

3.2.1 Fixed license

When using a fixed license, the license will only be valid for this particular computer.

3.2.1.1 Activate a fixed license

To activate a fixed license, a step-by-step description is shown of how to activate a license, see figure 3.2.



License type

Select the type of license to be used: Fixed

Fixed license

A license is unique to your computer. If you do not yet have a valid license, follow these steps:

1. Press the "Create license request" button. A popup will appear asking where the license_request file can be stored.
2. Send the license_request file to info@industrialcobotics.be
3. You will receive from us a license file that you can store on your computer.
4. Press the "Load license" button and selected the stored license file.

After this is done, the program will store the license file internally.

License

Status: None
No license found

Figure 3.2: Fixed license window, no license

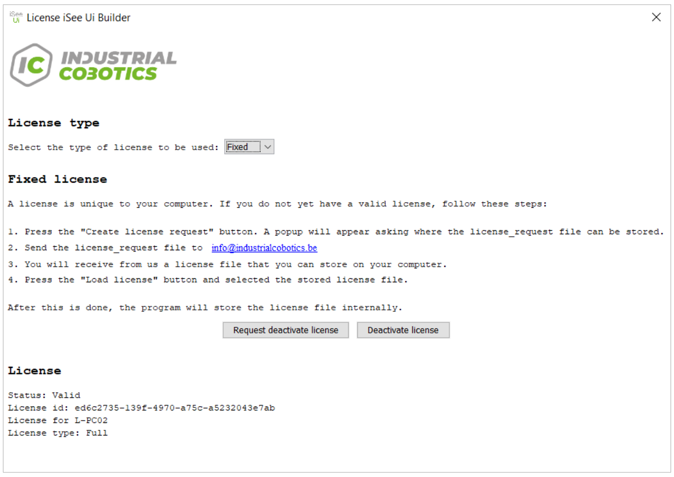
In this window, the "Create license request" button can be pressed. Next, choose a location for the license request file and press the "Save" button, after which the license request file will be generated. This license request file can then be mailed to info@industrialcobotics.be.

Remark: When sending the license request file, make sure it is the last license request file that was generated. Pressing the "Create license request" button a second time, will render the previously generated file invalid. Thus, once the license request file has been generated, and sent to Industrial Cobotics, don't press the button a second time. If this situation did occur, the newly generated license request file will also have to be sent in order to receive a valid license file.

A license file will then be sent back that is valid for that particular computer. Once the license file has been stored locally on the computer, the button "Load license" in the license window can be pressed to select the license file. The application will read the license content and store it in the installation directory of the application.

CAUTION: Do not alter, edit, add, delete or remove the "License" folder and all of its content, that has been created in the installation directory of the application. Doing so will invoke the validity of the license, rendering it useless.

When a correct license file is found, the license information will be visible in the license window, as can be seen in figure 3.3.



License type

Select the type of license to be used: Fixed

Fixed license

A license is unique to your computer. If you do not yet have a valid license, follow these steps:

1. Press the "Create license request" button. A popup will appear asking where the license_request file can be stored.
2. Send the license_request file to info@industrialcobotics.be
3. You will receive from us a license file that you can store on your computer.
4. Press the "Load license" button and selected the stored license file.

After this is done, the program will store the license file internally.

License

Status: Valid
License id: ed6c2735-139f-4970-a75c-a5232043e7ab
License for L-PC02
License type: Full

Figure 3.3: Fixed license window, valid license with license info

3.2.1.2 Deactivate a fixed license

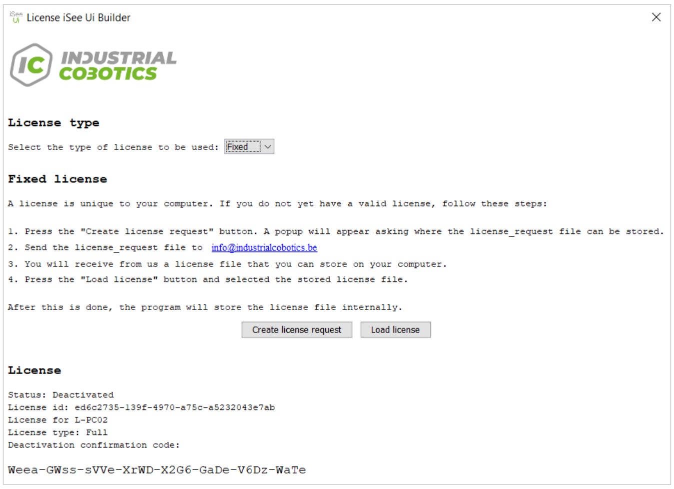
Since the activated license is linked to the computer on which it was activated, a deactivation procedure must be followed before the same license can be used on a different computer. If a license is needed that can be used on multiple computers, see section 3.2.2.

To deactivate a license, navigate to the “Help” >“License” and open the license window. Here, the currently active license can be deactivated by pressing the “Request deactivate license” button (see figure 3.3). Once pressed, a location has to be specified where the deactivation request file for the license can be stored. This will not deactivate the license just yet, allowing to keep using the license on the computer until the deactivation process is completed.

This deactivation request file can then be mailed to info@industrialcobotics.be. Afterward, you will receive a deactivation file which you can load by pressing the “Deactivate license” button. Once the deactivation file is selected, the license will be deactivated and a deactivation confirmation code will be generated in the license window (see figure 3.4). In order to complete the deactivation, this deactivation code needs to be sent to info@industrialcobotics.be such that the license will then be registered as deactivated for the specified computer. Now the license can be activated on another computer by means of creating and sending the license request file for that new computer. The license request file can either be sent together with the deactivation confirmation code, or at a later date.

3.2.2 Floating license

When a floating license is desired, the License server software can be installed on the designated server. Consult the License server manual on how to activate a license.



License iSee Ui Builder

License type

Select the type of license to be used: Fixed

Fixed license

A license is unique to your computer. If you do not yet have a valid license, follow these steps:

1. Press the "Create license request" button. A popup will appear asking where the license_request file can be stored.
2. Send the license_request file to info@industrialcobotics.be
3. You will receive from us a license file that you can store on your computer.
4. Press the "Load license" button and selected the stored license file.

After this is done, the program will store the license file internally.

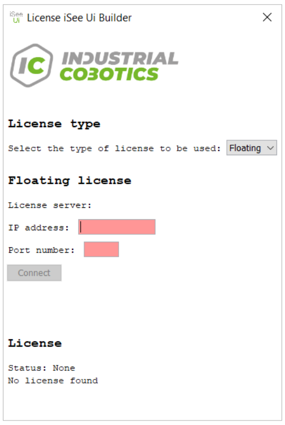
License

```
Status: Deactivated
License id: ed6c2735-139f-4970-a75c-a5232043e7ab
License for L-PC02
License type: Full
Deactivation confirmation code:

Weea-GWss-sVVe-XrWD-X2G6-GaDe-V6Dz-WaTe
```

Figure 3.4: Fixed license window, deactivated license

For a floating license, the computer needs to be able to communicate with the server. Once the server can be found by the computer and the license server is running, enter the server IP address and port number (see figure 3.5). If a license is available, and the computer can be granted a license by the server, a license will be made available as long as the computer is connected to the server.



License iSee Ui Builder

License type

Select the type of license to be used: Floating

Floating license

License server:

IP address:

Port number:

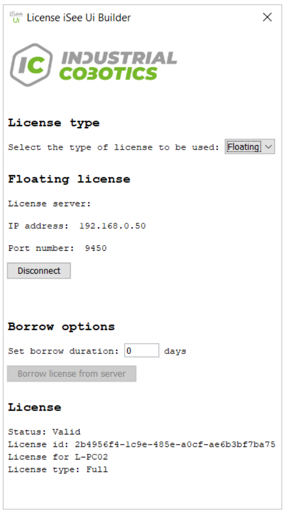
License

```
Status: None
No license found
```

Figure 3.5: Floating license window, no license

3.2.2.1 Commuting the floating license

When a floating license is needed offline, first connect to the license server in order to get a license. Once a license is granted, the license can be commuted for offline use. In order to do this, enter the amount of days that the license needs to be used offline.



License iSee Ui Builder

License type

Select the type of license to be used: Floating

Floating license

License server:

IP address: 192.168.0.50

Port number: 9450

Borrow options

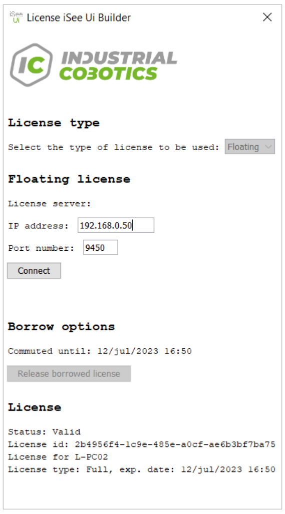
Set borrow duration: 0 days

License

Status: Valid
License id: 2b4956f4-1c9e-485e-a0cf-ae6b3bf7ba75
License for L-PC02
License type: Full

Figure 3.6: Floating license window, valid license

Figure 3.6 shows the borrow license options where the number of days can be entered. This number is not necessarily the amount of days that this commuted license will be valid. In the license server, each client can be configured to borrow/commute a license for a specified maximum amount of days.



License iSee Ui Builder

License type

Select the type of license to be used: Floating

Floating license

License server:

IP address: 192.168.0.50

Port number: 9450

Borrow options

Commuted until: 12/jul/2023 16:50

License

Status: Valid
License id: 2b4956f4-1c9e-485e-a0cf-ae6b3bf7ba75
License for L-PC02
License type: Full, exp. date: 12/jul/2023 16:50

Figure 3.7: Floating license window, commuted license

4. General use

In this chapter, the general interactions to create or edit an iSee Ui interface will be discussed.

4.1 iSee Ui file operations

4.1.1 Create a new interface

To create a new iSee Ui interface, open the "File" menu and press the "New" button, or use the shortcut key <ctrl> + <N>. This will create an empty iSee Ui interface as shown in figure 4.1.

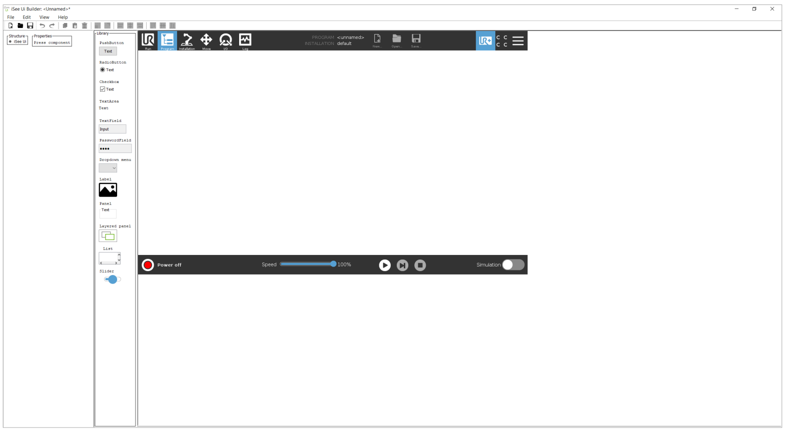


Figure 4.1: New iSee Ui dashboard

In the "Library" panel, all the available item types can be found. Every item type will be discussed in more detail in chapter 5.

4.1.2 Save an interface

If any changes are performed, e.g. adding or removing an item, editing an existing item or the overall structure, the software will denote that these changes have not been saved yet by adding a '*' symbol at the end of the file name in the window title. These changes can be saved by pressing the "Save" or "SaveAs" buttons in the "File" menu, or by using the shortcut key <ctrl> + <S>.

If a new iSee Ui interface was created, and the file hasn't been saved yet, the "Save" button will have the same effect as the "SaveAs", thus asking for the storage location and the name of the new file. The name of the iSee Ui interface file ends with the ".isui" extension. This extension should never be changed to something different, else both the iSee Ui URCap and the iSee Ui Builder software will not be able to find and load the file.

When changes to an iSee Ui interface haven't been saved and the "New" button is pressed to create a new interface, the software will denote that some changes haven't been saved yet and ask if these changes have to be saved first or not. If the user selects to first save the changes, and the interface has already been assigned a name, the software will save the changes to the same file. If a new iSee Ui interface has been created, the user will be prompted to select a storage location and a name for this new interface.

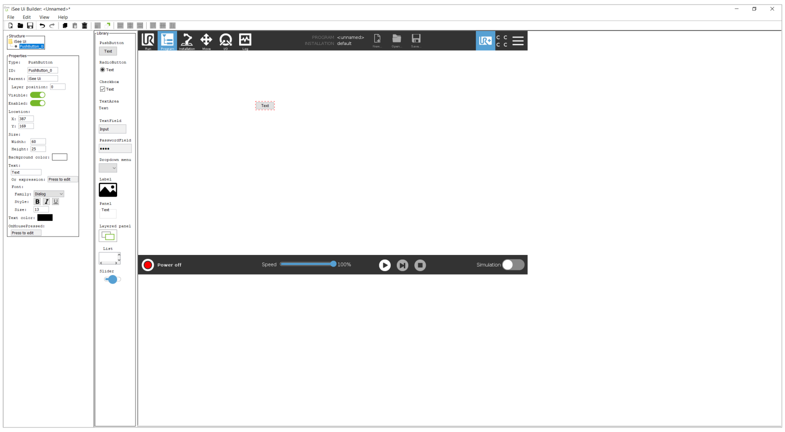
4.1.3 Load an interface

An existing iSee Ui interface file, of the ".isui" extension, can be loaded by pressing the "Load" button in the "File" menu. As mentioned in section 4.1.2, the file extension should never be altered, since the software won't be able to find and load the file if the extension is changed.

Furthermore, as mentioned in section 4.1.2, if a file is being opened while the previously loaded interface has changes that haven't been saved yet, the software will prompt the user whether these changes have to be stored before loading the new file or not.

4.2 Adding and removing an item

An item, of a certain item type, can be added to the iSee Ui interface by selecting the desired item type in the "Library" or by dragging the item from the library and placing it directly onto the interface. The item will be given an initial unique ID (or name). The item, with its unique name, will also be added to the structure tree, as can be seen in figure 4.2. More information regarding the structure tree will be discussed in section 4.5.



Type: PushButton
ID: PushButton_0
Parent: iSee Ui
Layer position: 0
Visible:
Enabled:
Location:
X: 387
Y: 169
Size:
Width: 60
Height: 25
Background color:
Text:
Text
Or expression: Press to edit
Font:
Family: Dialog
Style: B I U
Size: 13
Text color:
OnMousePressed:
Press to edit

PROGRAM <unnamed>
INSTALLATION default

Figure 4.2: Adding a new item to the interface

4.3 Item edit actions

Once an item has been added, the item can be edited by use of the properties panel (section 4.4), structure tree (section 4.5) or by simple mouse and/or keyboard interactions (section 4.6).

The result of any edit will immediately take effect on the item, and the window will display that a change has been made by the addition of the ‘ * ’ symbol to the window title.

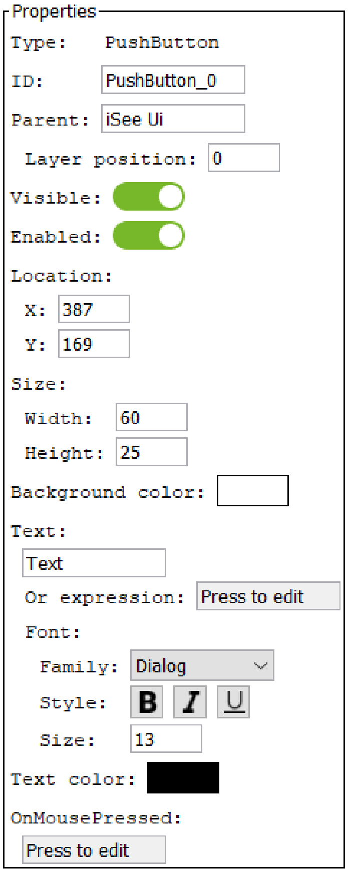
4.3.1 Undo redo

When an item property has changed, an item is added or removed, the change can be undone and redone by use of the toolbar icons ↶ ↷, pressing the “undo” and “redo” button in the menu under “Edit” or by using the shortcut keys <ctrl> + <Z> and <ctrl> + <Y> respectively.

4.4 Properties panel

Every item has properties that define e.g. the name, location, size, content, ... These properties can be viewed, for a particular item, by pressing the item in question on the interface or by pressing its name in the structure tree (section 4.5).
In figure 4.3, the properties of a pushbutton item are shown. Once the properties for an item have been opened in the properties panel, the properties can be edited in the panel itself.

When selecting multiple items, the properties panel will display “shared” general properties (see section 5.1). When editing a property, the new value will be applied to all selected items. Figure 4.4 shows the properties panel for multiple items that have the same value for the X-location and parent, but different Y-location and size.



Properties

Type: PushButton

ID: PushButton_0

Parent: iSee Ui

Layer position: 0

Visible:

Enabled:

Location:

X: 387

Y: 169

Size:

Width: 60

Height: 25

Background color:

Text:

Text

Or expression: Press to edit

Font:

Family: Dialog

Style: B I U

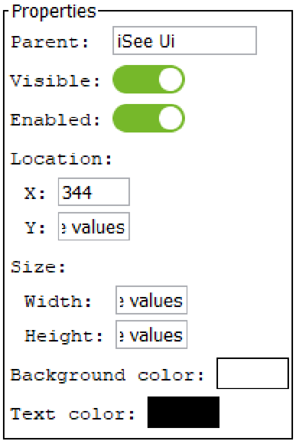
Size: 13

Text color:

OnMousePressed:

Press to edit

Figure 4.3: Example of properties for a pushbutton



Properties

Parent: iSee Ui

Visible:

Enabled:

Location:

X: 344

Y: values

Size:

Width: values

Height: values

Background color:

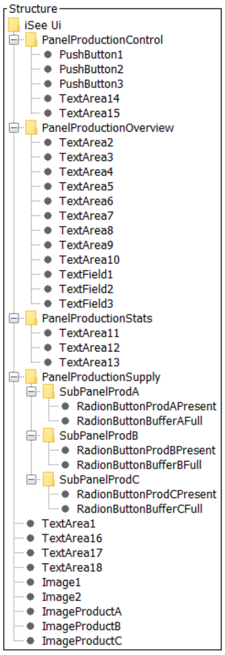
Text color:

Figure 4.4: Example of shared properties

In chapter 5, each item type will be discussed in more detail, as well as their properties.

4.5 Structure tree

The structure tree gives a detailed overview of the structure of the loaded interface. Moreover, it details which items are grouped by means of a "Panel" or "Layered panel" item. Figure 4.5 shows an example of an iSee Ui interface for which several panels were used. The advantage of using these panel items will be discussed in more detail in section 5.12 and 5.13.



```
Structure
iSee Ui
  PanelProductionControl
    PushButton1
    PushButton2
    PushButton3
    TextArea14
    TextArea15
  PanelProductionOverview
    TextArea2
    TextArea3
    TextArea4
    TextArea5
    TextArea6
    TextArea7
    TextArea8
    TextArea9
    TextArea10
    TextField1
    TextField2
    TextField3
  PanelProductionStats
    TextArea11
    TextArea12
    TextArea13
  PanelProductionSupply
    SubPanelProdA
      RadionButtonProdAPresent
      RadionButtonBufferAFull
    SubPanelProdB
      RadionButtonProdBPresent
      RadionButtonBufferBFull
    SubPanelProdC
      RadionButtonProdCPresent
      RadionButtonBufferCFull
  TextArea1
  TextArea16
  TextArea17
  TextArea18
  Image1
  Image2
  ImageProductA
  ImageProductB
  ImageProductC
```

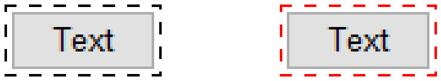
Figure 4.5: Example of a structure tree

From the structure tree, the properties of an item can be opened by selecting the item's ID in the tree. Thus, renaming an item's ID to a unique and descriptive name is a good practice for effectively creating an iSee Ui interface.

4.6 Mouse and/or keyboard interactions

4.6.1 Selecting and hovering

When hovering over an item in the result panel, a black dotted line is drawn around the component on which the mouse is positioned, as can be seen in figure 4.6a. When the left mouse button is pressed, it will be this item that will be selected.
When an item is selected, a red dotted line will be drawn around the component, as can be seen in figure 4.6b.



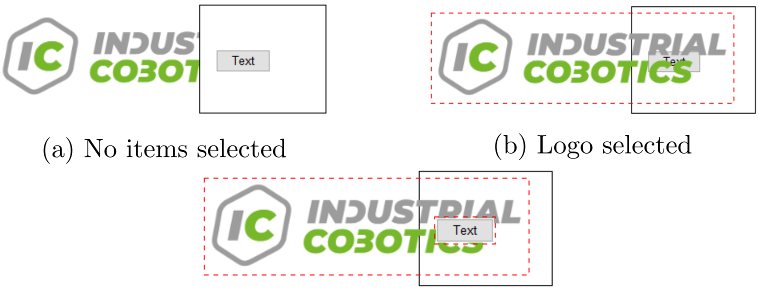
(a) Hovered item (b) Selected item

Figure 4.6: Hover and selection example

4.6.2 Selecting one or more items

Selecting one or more times can be done in several ways in both the interface result panel and the structure tree. Every item that is selected, will be brought in front of all the items that are not selected while maintaining the depth order between the selected items.

Figure 4.7 demonstrates this relation. In figure 4.7a the logo is placed behind the panel that contains the pushbutton. The moment the logo is selected, the logo will be brought in front of the panel and its content, the pushbutton. This can be seen in figure 4.7b. If the pushbutton is then selected as well, the pushbutton will be placed in front of the logo, as can be seen in figure 4.7c.



(a) No items selected

(b) Logo selected

(c) Logo and pushbutton selected

Figure 4.7: Selection example

Interface result panel

In the result panel, a single item can be selected by simply pressing the item on the panel with the left mouse button.

Multiple items can be selected by using the left mouse button in combination with either the <ctrl> or <shift> keys.

By pressing and holding the <ctrl> key, other items can be added to or removed from the selection when pressed with the left mouse button.

By pressing and holding the <shift> key and the left mouse button, a selection area can be drawn. Every item contained in the selection area will be selected, as can be seen in figure 4.8.

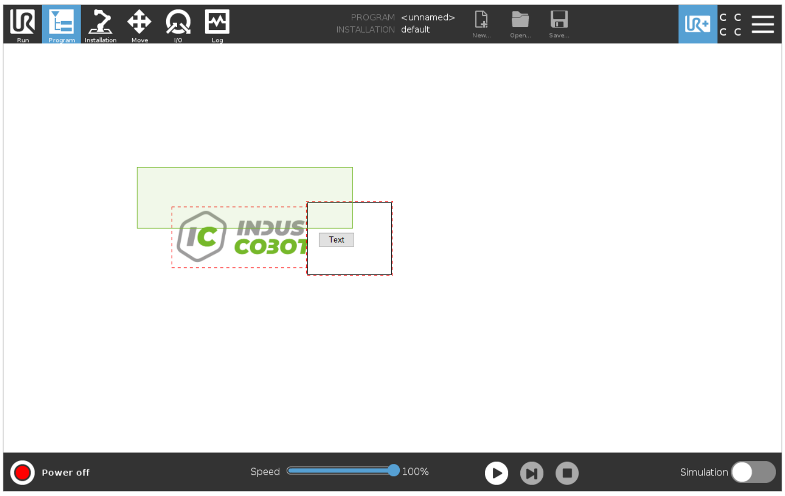


Figure 4.8: Using a selection area

Structure tree

In the structure tree, a single item can be selected by simply pressing the item name in the structure tree with the left mouse button.

Multiple items can be selected by using the left mouse button in combination with either the <ctrl> or <shift> keys.

By pressing and holding the <ctrl> key, other items can be added to or removed from the selection by pressing the item name in the tree with the left mouse button.

By pressing and holding the <shift> key, all item names in the tree between the currently selected name and the name selected with the left mouse button will be selected.

4.6.3 Assigning one or more items to a panel

As will be discussed in chapter 5, an item can be placed inside a panel. When an item is placed in a panel, the item is assigned to be a child of this panel, and the panel to be its parent.

This parent-child assigning can be accomplished by simple mouse interactions in the structure tree. This can be done in the structure tree by holding down the left mouse button on the name of the item that is to be the child of another item. The mouse can then be moved over the name of the other item, that is to be the parent. Releasing the left mouse button will then complete the parent-child assigning. In the structure tree, the parent item icon will change, if it isn't already, to a folder icon. The child item ID will then be added into this parent folder.

When performing this parent-child assigning in the structure tree, not every item type can be assigned to be the parent of another item. During the mouse movement, after pressing down the left mouse button on the child item, the mouse icon will indicate if the other item can be assigned to be a parent ⬉ or not 🛇.

Only the main iSee Ui interface, with ID/name “iSee Ui”, and a panel or layered panel item type can be assigned to be a parent of any item type.

4.6.4 Move and resize one or more items

The location of the item can be changed by changing both the “x” and “y” values in the properties panel. In chapter 5, this will be discussed in more detail. To resize an item, both the “width” and “height” values can be specified in the properties panel.

Another method to change the location or size of an item is by use of the mouse. When moving the mouse over an item, on the interface, the mouse icon will change to either an ✥ icon to move the item, or to an ↔ icon to resize the item. These mouse actions can be activated by holding down the left mouse button.

Furthermore, it is also possible to move or resize multiple items at the same time. To do so, simply select the items that need to be moved/resized, and use the left mouse button just like with a single item.

Finally, it is also possible for an item placed in a parent, to move it outside the parent boundaries. When an item is placed outside the boundaries of its parent, the item is in the “dead zone” of the parent, meaning that it will never be shown as long as it stays outside the boundaries.
This behaviour can be restricted such that an item always remains inside its parent boundaries. To do so, go into the menu to “View” > “Settings”. Here an option is given to allow items outside their parent boundaries or not.
When disabling this setting, the software will evaluate if any items are outside their parent boundaries or not. When there are items that are outside the boundaries, the user will be presented with a choice to relocate/resize the items or keep the items as they are.

4.7 Variables

Variables can be created in the “Variables” panel. This panel can be set viewable by navigating to the “View” menu and pressing the “Variables” button. This will open up the variables panel at the bottom of the window, as seen in figure 4.9.
Variables are used for the communication between the interface and the running program on the UR robot. Thus, a variable is used to communicate a value either originating from the robot program or from the operator through interaction with the dashboard.

In this panel, a new variable can be created by pressing the “Add” button. This will add a new variable with a unique name to the list as can be seen in figure 4.10.

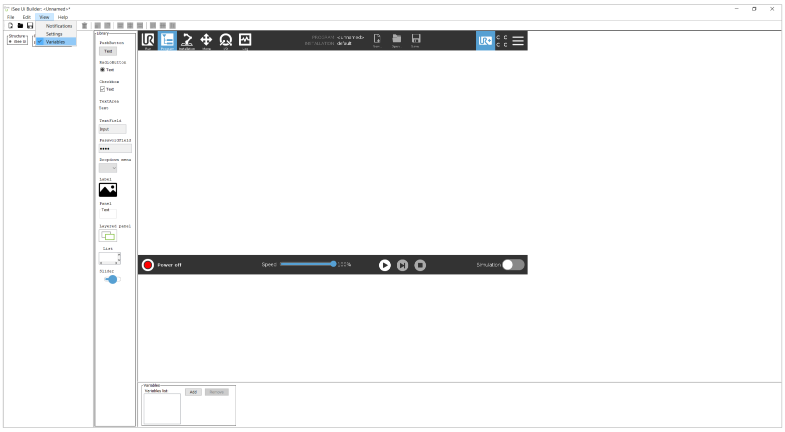
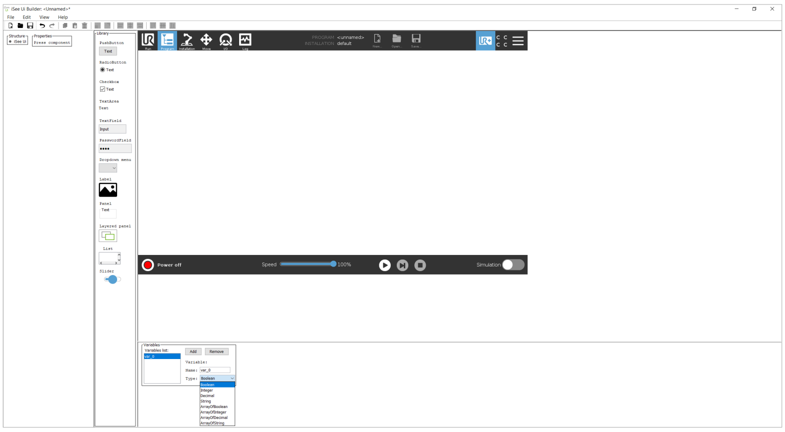


Figure 4.9: Open variables panel



var_0

Variable:

Name: var_0

Type: Boolean

Figure 4.10: Configuring a variable

Once the variable is added, a name and type can be given by first selecting the variable from the list and then assigning the name or selecting the type from the selection menu. The available types are:

- “Boolean”
- “Integer”
- “Decimal”
- “String”
- “ArrayOfBoolean”
- “ArrayOfInteger”
- “ArrayOfDecimal”
- “ArrayOfString”

Setting a correct type for the variable is of importance for the communication as well as for a correct operator interaction when the operator needs to enter a value on the robot display. The latter will be explained in section 5.3.

4.8 iSee Ui expression panel

In the iSee Ui expression panel, the operator interaction is configured for a particular item. To open this panel, first open the properties panel of an item for which an operator interaction needs to be assigned. For every item type, except the types "Panel" and "Layered panel", an expression can be configured.

Once the properties panel for the particular item is opened, a button "press to edit" can be found. When the button is pressed, the iSee Ui expression panel (figure 4.11), for that item's property is opened at the bottom of the Builder.

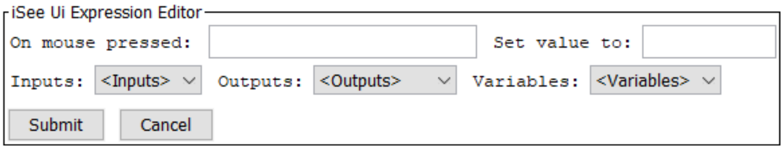


Figure 4.11: iSee Ui expression panel

As mentioned before, the iSee Ui interface is used to communicate the value of variables or IO's to the running robot program and vice versa.
An operator interaction thus sets a value to either a variable or an IO. In the iSee Ui expression panel, the specific variable or IO has to be entered in the field for which the action needs to occur.
For some properties, the expression also needs a value, being the value to be written to the configured variable/IO when the operator interaction takes place.
To complete the expression configuration, the "Submit" button can be pressed.
The iSee Ui expression for each item and property will be discussed in more detail in chapter 5.

5. iSee Ui item types and their properties

This chapter will cover all the available item types, what they can be used for, and how these items can be customized. The available item types are:

- TextArea: to display text
- TextField: field for the operator to enter a value or text
- PasswordField: field for the operator to enter a value or text, except the entered value/text will be unreadable
- PushButton: button that can be pressed by the operator
- RadioButton: a circular button that displays whether a boolean signal is high or low. The state of the signal can also be changed by pressing the button
- Checkbox: same as "RadioButton" except now the icon is a squared checkmark
- Dropdown menu: allows for the operator to select an item from a list of items
- List: same as "Dropdown menu", except now all the items are visible without having to open a selection menu
- Slider: can be used to easily make a specific selection or an interpolated value
- Label: to display an image
- Panel: can be used to create structure by grouping items, without having the possibility to specify the depth of an item
- Layered panel: same as "Panel", except with a layered panel, each item inside the panel is given a depth

Each of these item types has a set of properties that can be customized. The general properties will be discussed in section 5.1. Afterward, each item type will be discussed in more detail, as well as the properties specific to that item type.

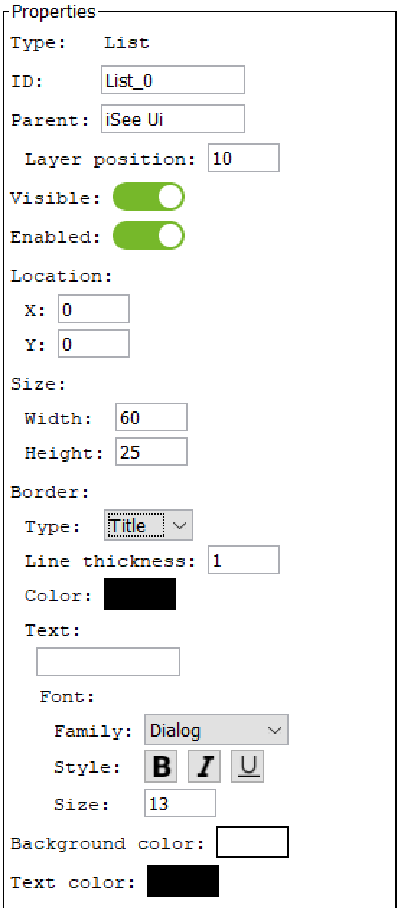
5.1 General properties

General properties that all item types have:

- ID
- Parent
- Layer position
- Visible
- Location
- Size

General properties that not all item types have:

- Enabled
- Border
- Background color
- Text color (foreground color)



Properties

Type: List

ID: List_0

Parent: iSee Ui

Layer position: 10

Visible: ●

Enabled: ●

Location:

X: 0

Y: 0

Size:

Width: 60

Height: 25

Border:

Type: Title

Line thickness: 1

Color: ■

Text:

Font:

Family: Dialog

Style: **B** *I* U

Size: 13

Background color: □

Text color: ■

Figure 5.1: General properties

5.1.1 General property: ID

The “ID” of an item is the name that is given to that item, thus the “ID” needs to be unique. It is also the name with which the item is visualized in the structure tree. Assigning a unique and descriptive name to an item will ensure that the structure tree is more readable and that the item can more easily be found in the structure tree.

Not only is a unique and descriptive name useful to effectively create an iSee Ui interface, but the “ID” of an item is also used in the configuration of the iSee Ui program node. With the iSee Ui program node, the robot can be programmed to alter an item’s property. Thus, assigning a unique and descriptive name will ensure that the item, for which a property needs to be altered during runtime, can be found more easily. For more detail, consult the iSee Ui URCap manual.

5.1.2 General property: Parent

The “Parent” property denotes to which parent item this child item belongs. The default and initial parent of each item is “iSee Ui” which denotes the iSee Ui main panel. A different parent can be assigned by entering the ID of the parent item or by using the drag and drop operation in the structure tree, as discussed in section 4.6.3.

Assigning a parent to an item will result in a folder-type structure in the structure tree. Additionally, the child item location will be positioned relative to its parent, as will be discussed in section 5.1.5.

5.1.3 General property: Layer position

The “Layer position” property of an item denotes the depth the item needs to be placed in its parent. This property is made available for items which parent is “iSee Ui” or a parent of the type “Layered panel”. Items in a parent of type “Panel” do not have a depth.

Items with a lower layer position are placed above items with a higher layer position.
An item can be placed at the top by entering 0 for its layer position.
An item can be placed at the bottom by entering a number larger than the number of items in the parent, or by entering -1.
To bring an item, or group of selected items, to the front or to the back, the buttons ⧉ ⧉ can also be used in the toolbar.

5.1.4 General property: Visible

With the “Visible” property, an item can be set visible or invisible. Setting the visible property off will render the item invisible in the iSee Ui Builder and when activating in the iSee Ui URCap on the UR robot.

This property can be changed by the UR robot program during runtime, thus allowing full control of which items need to be initially invisible and set visible at a certain moment during runtime.

By changing the visibility of an item, multiple screens can be created. This can be done by changing the visibility of panel items. This in turn allows for more advanced functionalities such as:

- Log-in procedure
- Wizard, step-by-step configuration, functionality
- Menu operation, for avoiding too much info on one page at a time

- Palletizer (builder)
- User accounts
- ... the possibilities are endless

5.1.5 General property: Location

An object's location on the iSee Ui interface can be altered by entering a value for the "x" and "y" coordinate for that particular item. The value denotes the relative position of the item towards its parent, as can be seen in figure 5.2.

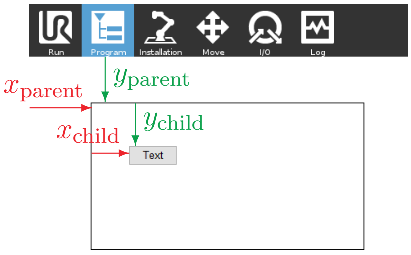


Figure 5.2: Location of an item

The "x" coordinate denotes the pixel distance from the left side of the item towards the left side of its parent.
The "y" coordinate denotes the pixel distance from the top side of the item towards the top side of its parent.

Thus, whenever the parent item location is changed, the absolute location (location on the screen) of the child item will be changed as well. As discussed in section 4.6.4, the location of an item can also be altered by holding down the left mouse button and dragging the item to the desired location.

5.1.6 General property: Size

The size of an object can be specified by entering the desired width and height for the item. The value for the width and height are also expressed in pixels.

In section 4.6.4, an alternative method to resize a component was discussed by holding down the left mouse button and dragging the mouse until the item has the desired size.

5.1.7 General property: Enabled

With the "Enabled" property, an item can be set enabled or disabled. Setting an item disabled will render it unresponsive to any interaction from the user, while still being visible on the interface. Any disabled item is greyed-out in order to visualize that the component is not enabled/active for interaction.

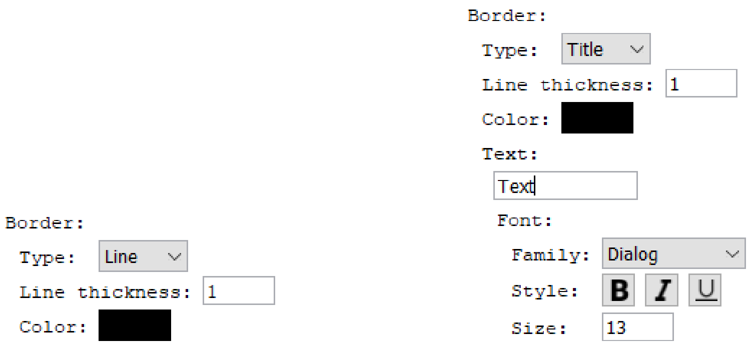
Not every item has an “Enabled” property. For instance, a text cannot be set disabled since it has no interaction. The items that do not have the “Enabled” property are:

- TextArea
- Panel
- Layered panel

5.1.8 General property: Border

With the “Border” property, the border of an item can be changed. The 3 available border types are:

- Empty
- Line
- Title



Border:
Type: Line
Line thickness: 1
Color: ■

(a) Line border

Border:
Type: Title
Line thickness: 1
Color: ■
Text:
Text
Font:
Family: Dialog
Style: **B** *I* U
Size: 13

(b) Title border

Figure 5.3: Properties of border types

Most of the items do not have a “Border” property, these are:

- PushButton
- RadioButton
- Checkbox
- TextField
- PasswordField
- Dropdown menu

Empty border
An item with an empty border will have no border.

Line
An item with a "Line" border, will have a squared line border, for which both the line thickness and line color can be changed, as can be seen in figure 5.3a.

Title
An item with a "Title" border, will have a border similar to the "Line" border except that a title is added to the line border if the title text is not empty. For the title of the border, the text font family, style and size can be changed, as can be seen in figure 5.3b.

5.1.9 General property: Background color

With the "Background color" property, the background color of an item can be changed by using the color chooser, as shown in figure 5.4.
The desired color can be selected from a set of predefined colors (figure 5.4a), or specified by its RGB[1] (figure 5.4b), HSV[2], HSL[3] or CMYK[4] value.

Another parameter used when specifying the HSV, HSL, RGB or CMYK value is the transparency or alpha value. HSV and HSL use transparency with a value from 0 (not transparent) to 100 (fully transparent). RGB and CMYK use an alpha value that ranges from 0 (fully transparent) to 255 (not transparent).

At the bottom of the color chooser, the item with the initial background color as well as a preview of the selected background color is shown.

Not every item has a "Background color" property. The items that do not have this property are:

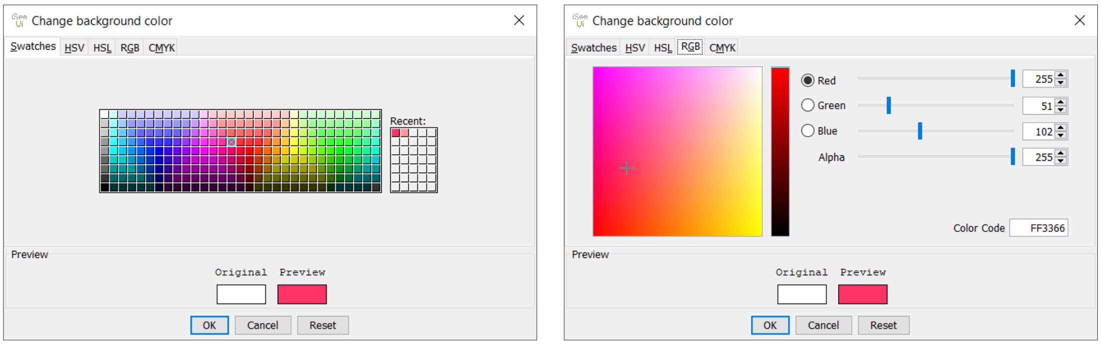
- Dropdown menu
- Label

[1] RGB: R = red, G = green, B = blue. The result is a combination of a value from 0 to 255 of red, green and blue.

[2] The HSV (which stands for Hue Saturation Value) scale provides a numerical readout of your image that corresponds to the color names contained therein. Hue is measured in degrees from 0 to 360. For instance, cyan falls between 181–240 degrees, and magenta falls between 301–360 degrees. The value and saturation of a color are both analyzed on a scale of 0 to 100 percent.

[3] HSL (hue, saturation, lightness). Hue is a degree on the color wheel from 0 to 360. 0 is red, 120 is green, and 240 is blue. Saturation is a percentage value. 0% means a shade of gray, and 100% is the full color. Lightness is also a percentage value. 0% is black, and 100% is white.

[4] CMYK: used with printers. C = Cyan, M = Magenta, Y = Yellow, K = Black. The result is a combination of a value from 0 to 255 of cyan, magenta, yellow and black.



● Red 255
○ Green 51
○ Blue 102
Alpha 255
Color Code FF3366

(a) Swatches

(b) RGB

Figure 5.4: Color chooser

5.1.10 General property: Text color

With the “Text color” property, the foreground or text color of an item can be changed using the color chooser, as described in section 5.1.9.

Not every item has a “Text color” property. The items that do not have this property are:

- Dropdown menu
- Label
- Panel
- Layered panel

5.2 TextArea

An item of the type “TextArea” is used to display a text or the live value of a variable or IO. A combination of both is not possible as of yet. For this, multiple “TextArea” items can be used. An example of a “TextArea” item as well as its properties panel is shown in figures 5.5 and 5.6.

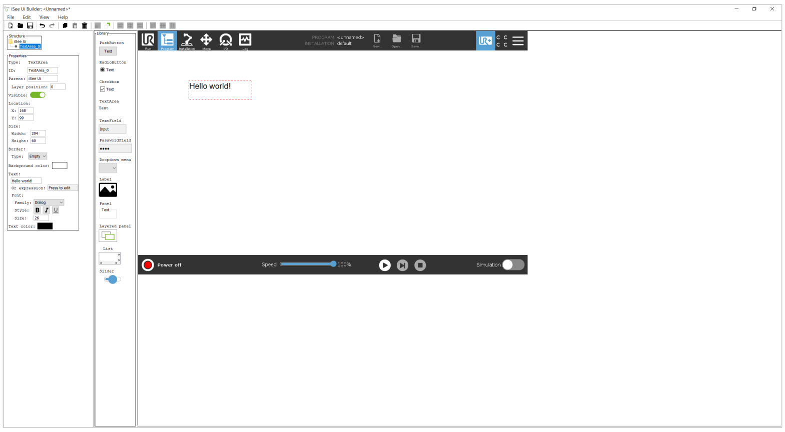
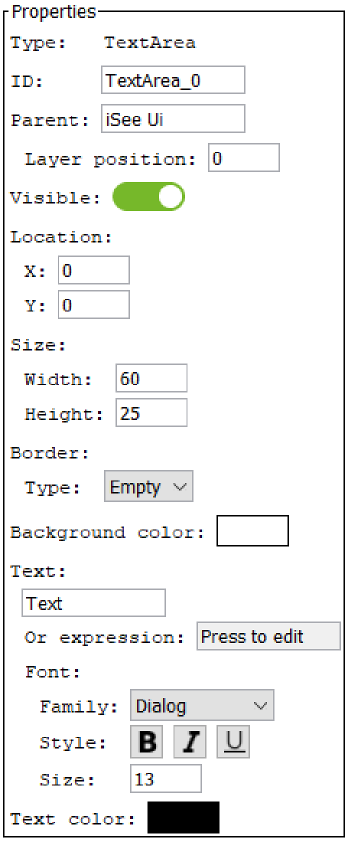


Figure 5.5: TextArea and its properties

For a “TextArea”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Text or expression
- Font



Properties
Type: TextArea
ID: TextArea_0
Parent: iSee Ui
Layer position: 0
Visible:
Location:
X: 0
Y: 0
Size:
Width: 60
Height: 25
Border:
Type: Empty
Background color: □
Text:
Text
Or expression: Press to edit
Font:
Family: Dialog
Style: **B** *I* U
Size: 13
Text color: ■

Figure 5.6: TextArea properties

5.2.1 TextArea property: Text

When the item has to display a fixed text, the text can be entered in the field provided for the "Text" property. The entered text will then be displayed in the item on the interface.

To have a multi-line text, "\n" can be used.
For instance, when entering "Hello\nWorld!", the result will be:
"Hello
World!"

If the text is too long, and the entire text is not visible, consider enlarging the width of the item or dividing the text into multiple lines by using "\n" or multiple "TextArea's".

5.2.2 TextArea property: Text as expression

As mentioned, the text of a "TextArea" can either display a fixed text, by entering the text in the "Text" field property or display the live value of a variable or IO. To configure the "TextArea" to display such a live value, an expression needs to be set by pressing the "Press to edit" button. For more information about the expression panel, see section 4.8.

After configuring an expression, the "TextArea" item on the result panel will display a text of the format "valueOf(<object name>)".

5.2.3 TextArea property: Font

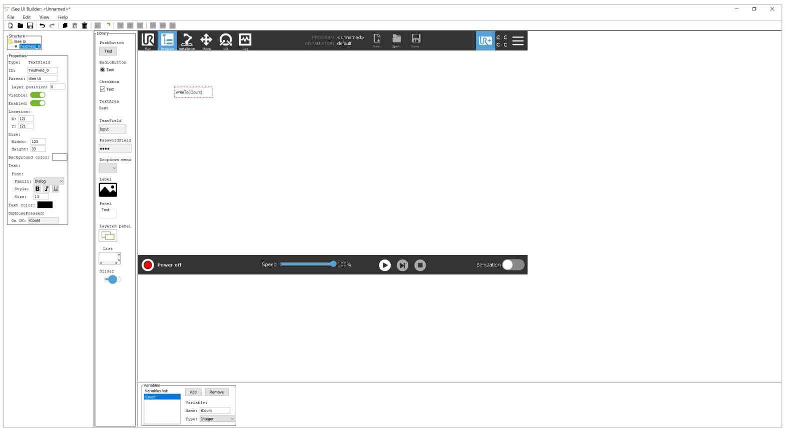
With the "Font" property, the text appearance can be changed. By changing the font family, style and size, a clear and structured interface can be created for the operator to interact with.

The font family offers the choice between some predefined letter types. As of now, other different letter types are not possible.
With the font style, a text can be set as bold, italic, and/or underlined.
The font size can be changed by entering the desired size in the provided field.

5.3 TextField

An item of the type "TextField" can be used when the operator needs to enter a value, which can be numerical or text. This value can either be written to a variable or an output IO signal, an input cannot be written to. Furthermore, the output signal or the variable can only be of the type "Integer", "Decimal" and "String". "Boolean" types cannot be used. For setting a boolean value, consider using either a "PushButton" (section 5.5), a "RadioButton" (section 5.6), a "Checkbox" (section 5.7) or even a "Dropdown menu" (section 5.8).

Once this object is pressed by the operator, an additional interface in the form of a keyboard or a numpad will be displayed. The type of interface depends on the type of variable or output to which a new value needs to be written to. An example of a "TextField" item as well as its properties panel is shown in figures 5.7 and 5.8.



Properties
Type: TextField
ID: TextField_0
Parent: iSee Ui
Layer position: 0
Visible: ●
Enabled: ●
Location:
X: 121
Y: 121
Size:
Width: 123
Height: 33
Background color:
Text:
Font:
Family: Dialog
Style: B I U
Size: 13
Text color: ■
OnMousePressed:
On OK: iCount

PROGRAM <unnamed>
INSTALLATION default

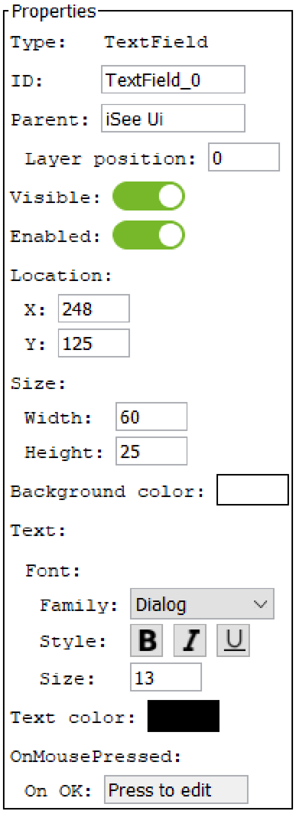
writeTo(iCount)

Variables
Variables list: iCount
Variable:
Name: iCount
Type: Integer

Figure 5.7: TextField and its properties

For a "TextField", the following properties, additional to the ones explained in section 5.1, can be customized:

- Font
- On Mouse Pressed



Properties
Type: TextField
ID: TextField_0
Parent: iSee Ui
Layer position: 0
Visible:
Enabled:
Location:
X: 248
Y: 125
Size:
Width: 60
Height: 25
Background color:
Text:
Font:
Family: Dialog
Style: B I U
Size: 13
Text color:
OnMousePressed:
On OK: Press to edit

Figure 5.8: TextField properties

5.3.1 TextField property: Font

The “Font” property is the same as discussed in section 5.2.3. In this case, the text will be the value that is set in the corresponding variable or IO.

5.3.2 TextField property: On Mouse Pressed

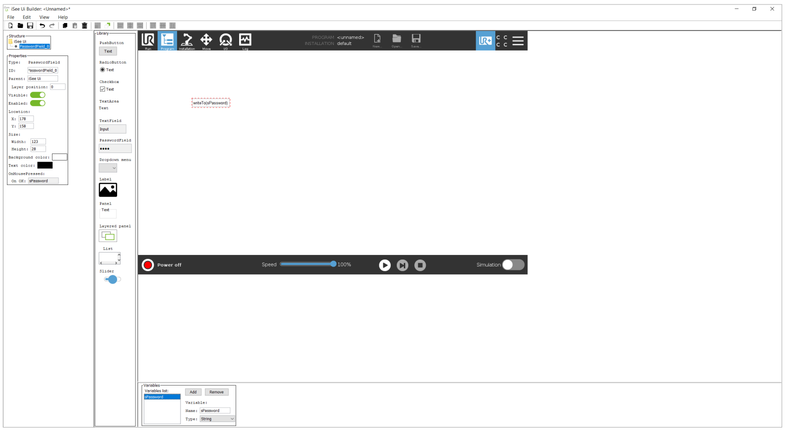
As mentioned, the “TextField” can either set a new value to a variable or an output when the operator presses the item. To configure the “TextField”, an iSee Ui expression needs to be set by pressing the “Press to edit” button. Once pressed, the iSee Ui expression panel will open as discussed in section 4.8. In this panel, the object name, for which the new value needs to be set, can be entered in the field next to “On mouse pressed”.

Once the name for the object is entered, the configuration of the iSee Ui expression can be completed by pressing the “Submit” button. After completion, the “TextField” item will display a text of the format “writeTo(<object name>)”.

Depending on the type of the configured object, the operator will either get a numpad (in case of an integer or decimal object) or a keyboard (in case of a string object).

5.4 PasswordField

A "PasswordField" is the same as a "TextField" (see section 5.3), except the entered text will be unreadable, and a "PasswordField" can only be used with a variable of type "Integer" and "String". An example of a "PasswordField" item as well as its properties panel is shown in figures 5.9 and 5.10.



iSee Ui Builder: <Unnamed>*

Structure
iSee Ui
PasswordField_0

Properties
Type: PasswordField
ID: PasswordField_0
Parent: iSee Ui
Layer position: 0
Visible:
Enabled:
Location:
X: 178
Y: 158
Size:
Width: 123
Height: 28
Background color:
Text color:
OnMousePressed:
On OK: sPassword

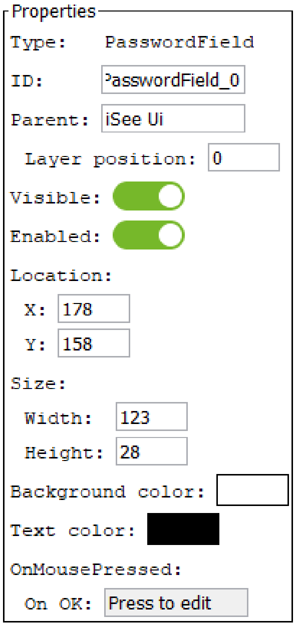
writeTo(sPassword)

Variables
Variables list:
sPassword
Variable:
Name: sPassword
Type: String

Figure 5.9: PasswordField and its properties

For a "PasswordField", the following properties, additional to the ones explained in section 5.1, can be customized:

- Font
- On Mouse Pressed



Properties

Type: PasswordField

ID: PasswordField_0

Parent: iSee Ui

Layer position: 0

Visible: ●

Enabled: ●

Location:

X: 178

Y: 158

Size:

Width: 123

Height: 28

Background color: □

Text color: ■

OnMousePressed:

On OK: Press to edit

Figure 5.10: PasswordField properties

These properties are configured in the same way as that for a "TextField" item, as discussed in sections 5.3.1 and 5.3.2.

During runtime, the unreadable content of a "PasswordField" item can be made readable/shown by the use of the iSee Ui Program node. For more detail, consult the iSee Ui URCap manual.

5.5 PushButton

A "PushButton" item can be used to configure an action when the operator presses the button. The action that can be configured is the writing of a predefined value to a variable or an output signal.

As with "TextField" items, a value cannot be written to an input signal. In contrast to a "TextField" item, with a "PushButton" all types of values can be written to either a variable or an output signal. Furthermore, the operator will not be given the choice to enter the desired value when the button is pressed. With this item type, the value that is written, when the button is pressed, is preconfigured when configuring the iSee Ui expression. An example of a "PushButton" item as well as its properties panel is shown in figures 5.11 and 5.12.

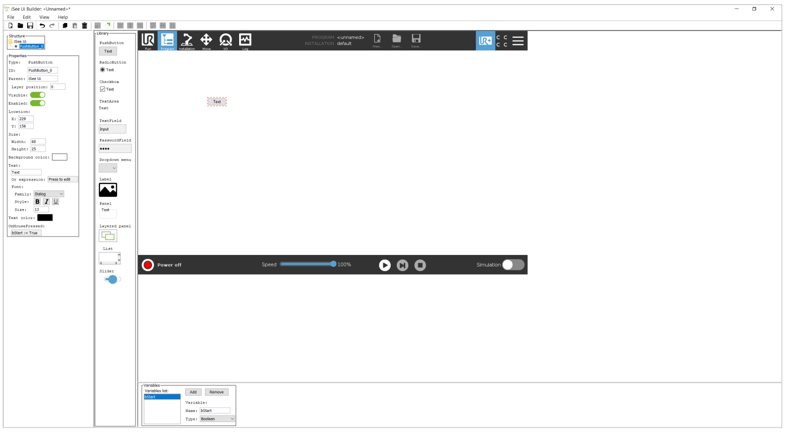
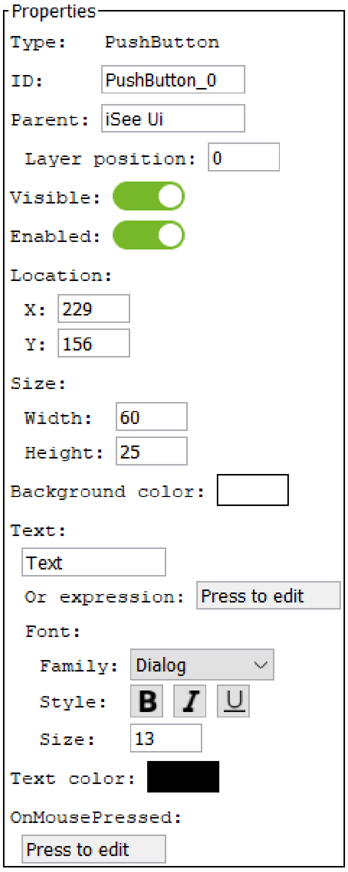


Figure 5.11: PushButton and its properties

For a “PushButton”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Text or expression
- Font
- On Mouse Pressed



Properties

Type: PushButton

ID: PushButton_0

Parent: iSee Ui

Layer position: 0

Visible:

Enabled:

Location:

X: 229

Y: 156

Size:

Width: 60

Height: 25

Background color:

Text:

Text

Or expression: Press to edit

Font:

Family: Dialog

Style: B I U

Size: 13

Text color:

OnMousePressed:

Press to edit

Figure 5.12: PushButton properties

5.5.1 PushButton property: Text or expression

The “Text” property is the same as discussed in section 5.2.1 and section 5.2.2 in case of an expression. The entered text will be the text that is displayed in the center of the button. Thus, specifying a descriptive text, mentioning the function of the button, is advised.

If a button with an image instead of text is desired, a “Label” item can be used, as will be discussed in section 5.11.

5.5.2 PushButton property: Font

The “Font” property is the same as discussed in section 5.2.3.

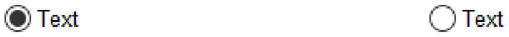
5.5.3 PushButton property: On Mouse Pressed

As mentioned, the “PushButton” item can set a new value to a variable or an output signal when the operator presses the item. To configure the “PushButton”, an iSee Ui expression needs to be set by pressing the “Press to edit” button. Once pressed, the iSee Ui expression panel will open as discussed in section 4.8. For this item, a value needs to be given in the adjacent field next to “Set value to:”. It will be this value that is written to the specified object each time the item is pressed.

5.6 RadioButton

A “RadioButton” item can be used to visualize or change the state of a boolean variable or digital IO signal. The “RadioButton” has to be configured if it either displays a value or writes a value. If the “RadioButton” is configured to write a value, the operator will be able to change the state by pressing the button.

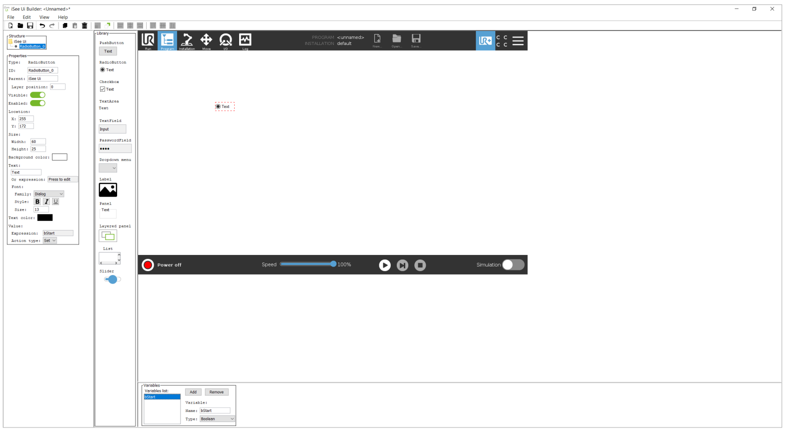
With a “RadioButton”, a circular button is added in front of the text. This circular button then indicates whether the boolean value is either high or low, as can be seen in figure 5.13.



(a) High value: “True” (b) Low value: “False”

Figure 5.13: RadioButton states

An example of a “RadioButton” item as well as its properties panel is shown in figures 5.14 and 5.15.



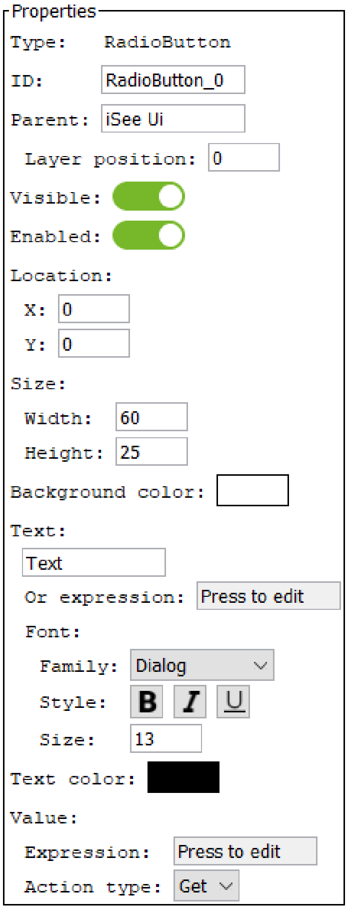
Properties
Type: RadioButton
ID: RadioButton_0
Parent: iSee Ui
Layer position: 0
Visible:
Enabled:
Location:
X: 255
Y: 172
Size:
Width: 60
Height: 25
Background color:
Text:
Text
Or expression: Press to edit
Font:
Family: Dialog
Style: B I U
Size: 13
Text color:
Value:
Expression: bStart
Action type: Set

Variables
Variables list:
bStart
Variable:
Name: bStart
Type: Boolean

Figure 5.14: RadioButton and its properties

For a “RadioButton”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Text or expression
- Font
- Value expression and action type



Properties

Type: RadioButton

ID: RadioButton_0

Parent: iSee Ui

Layer position: 0

Visible: ●

Enabled: ●

Location:

X: 0

Y: 0

Size:

Width: 60

Height: 25

Background color: □

Text:

Text

Or expression: Press to edit

Font:

Family: Dialog

Style: B I U

Size: 13

Text color: ■

Value:

Expression: Press to edit

Action type: Get

Figure 5.15: RadioButton properties

5.6.1 RadioButton property: Text or expression

The “Text” property is the same as discussed in section 5.2.1 and section 5.2.2 in case of an expression. The entered text will be the text that is displayed after the circular button. Thus, specifying a descriptive text, mentioning the function of the button or what the value of the button represents, is advised.

5.6.2 RadioButton property: Font

The “Font” property is the same as discussed in section 5.2.3.

5.6.3 RadioButton property: Value expression

As mentioned, the “RadioButton” item can either display the live value or set a new value to a variable or a digital IO signal. To configure the “RadioButton”, an iSee Ui expression needs to be set by pressing the “Press to edit” button. Once pressed, the iSee Ui expression panel will open as discussed in section 4.8.

5.6.4 RadioButton property: Value action type

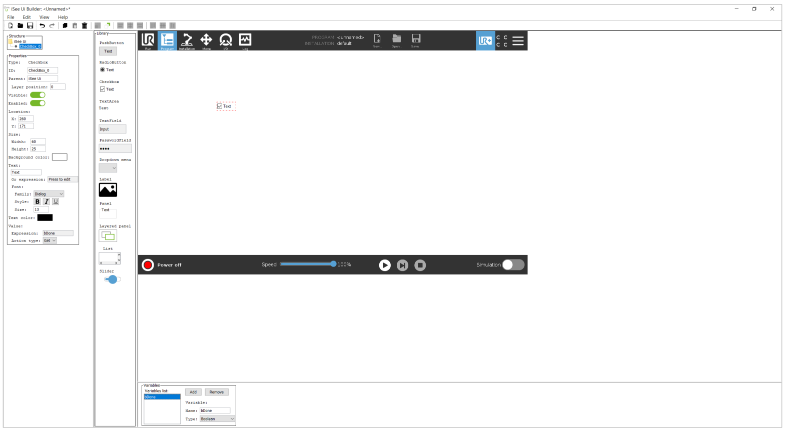
The “Action type” property allows configuring whether the “RadioButton” displays the live value, or if the operator can also change the value by pressing it.

If the type is set to “Get”, the “RadioButton” will only display the live value of the object configured in the iSee Ui expression. For this type, all digital IO signals and boolean variables can be selected.

If the type is set to “Set”, the “RadioButton” will write its state to the object configured in the iSee Ui expression. For this type, digital input signals cannot be selected since they cannot be written to. Only boolean variables and digital output signals can be selected.

5.7 Checkbox

A “Checkbox” item is the same as a “RadioButton” item (see section 5.6) except the icon in front of the text will be a squared checkmark. An example of a “Checkbox” item as well as its properties panel is shown in figures 5.16 and 5.17.

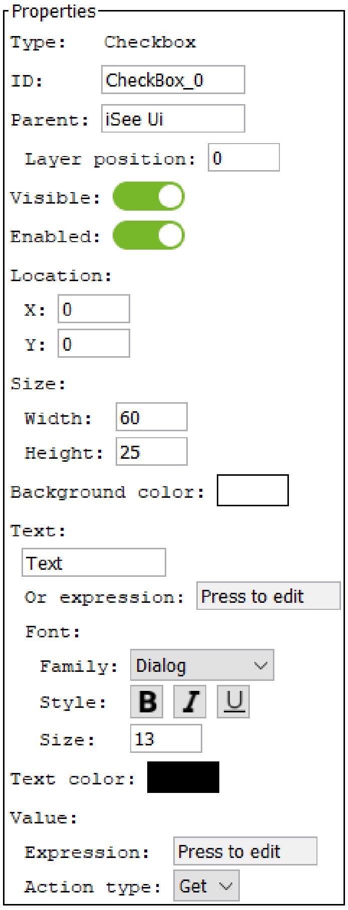


Properties
Type: Checkbox
ID: CheckBox_0
Parent: iSee Ui
Layer position: 0
Visible:
Enabled:
Location:
X: 260
Y: 171
Size:
Width: 60
Height: 25
Background color:
Text:
Text
Or expression: Press to edit
Font:
Family: Dialog
Style: B I U
Size: 13
Text color:
Value:
Expression: bDone
Action type: Get

Figure 5.16: Checkbox and its properties

For a “Checkbox”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Text or expression
- Font
- Value expression and action type



Properties

Type: Checkbox

ID: CheckBox_0

Parent: iSee Ui

Layer position: 0

Visible: ●

Enabled: ●

Location:

X: 0

Y: 0

Size:

Width: 60

Height: 25

Background color: □

Text:

Text

Or expression: Press to edit

Font:

Family: Dialog

Style: **B** *I* U

Size: 13

Text color: ■

Value:

Expression: Press to edit

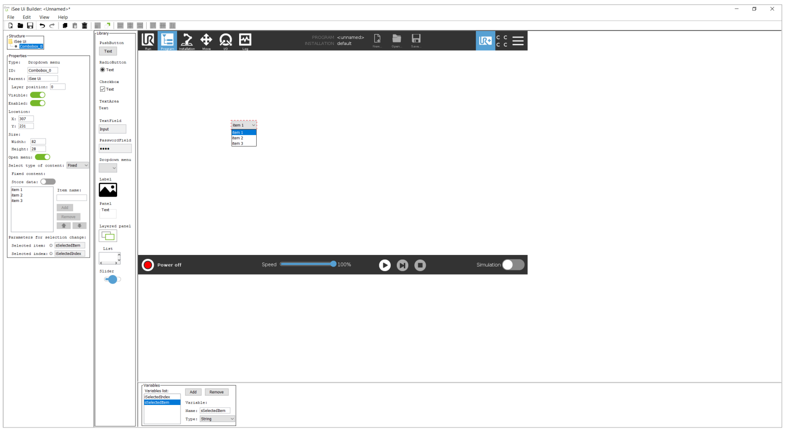
Action type: Get

Figure 5.17: Checkbox properties

These properties are configured in the same way as that for a “RadioButton” item, which are discussed respectively in sections 5.6.1, 5.6.2, 5.6.3 and 5.6.4.

5.8 Dropdown menu

A “Dropdown menu” item can be used for the operator to make a selection of a list of items. When the operator presses the item, a menu will be made visible listing the items from which the operator can select an item. The content of this menu can be fixed or variable, as will be discussed in section 5.8.2. An example of a “Dropdown menu” item as well as its properties panel is shown in figures 5.18 and 5.19.



iSee Ui Builder: <Unnamed>*

Structure: iSee Ui / Combobox_0

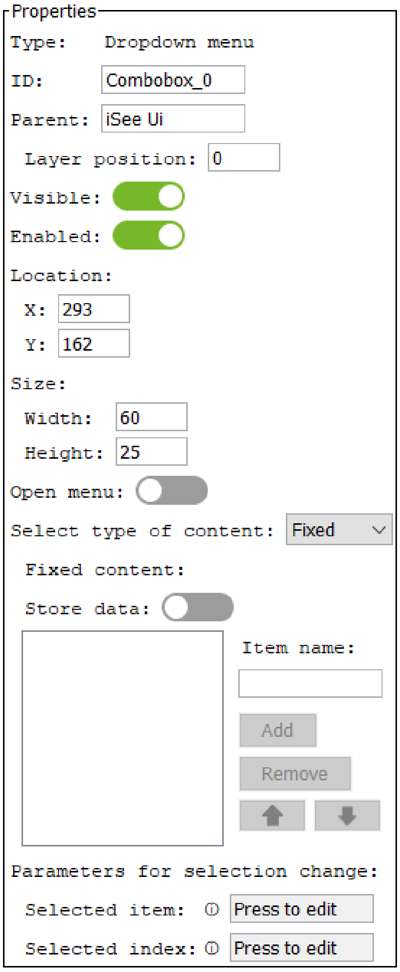
Properties
Type: Dropdown menu
ID: Combobox_0
Parent: iSee Ui
Layer position: 0
Visible: on
Enabled: on
Location:
X: 307
Y: 231
Size:
Width: 82
Height: 28
Open menu: on
Select type of content: Fixed
Fixed content:
Store data: off
item 1
item 2
item 3
Item name:
Parameters for selection change:
Selected item: sSelectedItem
Selected index: iSelectedIndex

Variables
Variables list:
iSelectedIndex
sSelectedItem
Variable:
Name: sSelectedItem
Type: String

Figure 5.18: Dropdown menu and its properties

For a “Dropdown menu”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Open menu
- Content type
- Selected item, selected index



Properties

Type: Dropdown menu

ID: Combobox_0

Parent: iSee Ui

Layer position: 0

Visible:

Enabled:

Location:

X: 293

Y: 162

Size:

Width: 60

Height: 25

Open menu:

Select type of content: Fixed

Fixed content:

Store data:

Item name:

Parameters for selection change:

Selected item: Press to edit

Selected index: Press to edit

Figure 5.19: Dropdown menu properties

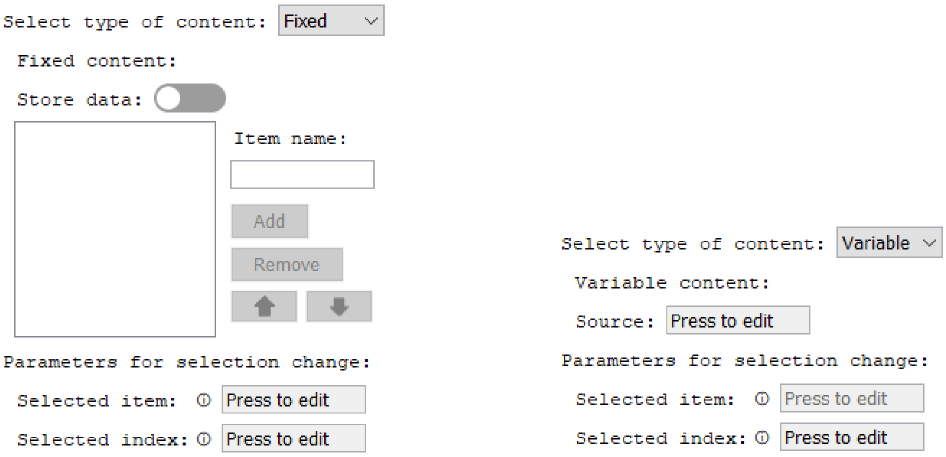
5.8.1 Dropdown menu property: Open menu

The "Open menu" property can be used to open the menu and preview the content. This will allow the user to preview whether the size of the object is sufficient in order for the entire menu content to be readable.
This property is only enabled if the "Content type" is of type "Fixed". For a "Variable" type, the content is unknown to the iSee Ui Builder, and thus it cannot be previewed.

5.8.2 Dropdown menu property: Content type

The "Select type of content" property can be used to configure the menu content of the "Dropdown menu" item. Figure 5.20 shows the properties for both a "Fixed" and a

"Variable" menu content type.



Select type of content: Fixed

Fixed content:

Store data:

Item name:

Parameters for selection change:

Selected item: Press to edit

Selected index: Press to edit

(a) Fixed

Select type of content: Variable

Variable content:

Source: Press to edit

Parameters for selection change:

Selected item: Press to edit

Selected index: Press to edit

(b) Variable

Figure 5.20: Menu content types

Fixed
A menu of type "Fixed" uses fixed, predefined, items of type "String". On the left side, a list is displayed showing the items already contained in the menu of the item.
An item can be added to this list by entering the name in the provided field and pressing the "Add" button, or by pressing the "enter" keyboard key.
Items can be removed from the list by selecting the item in the list and pressing the "Remove" button.
The order of the items in the list will also be the order of the items in the menu of the "Dropdown menu" item during runtime. To change the order of the items, simply select an item and use the ↑ and ↓ buttons.
An item can be changed by selecting the item in the list and changing the name in the provided field. Once the name is changed, press the "enter" keyboard key to change the item name.

As discussed in section 5.8.1, the menu of the "Dropdown menu" item can be previewed by switching on the "Open menu" property.

Another property made available, if the content type is of type "Fixed", is the "Store data". The functionality of this property will be discussed in detail in the iSee Ui URCap manual. For a "Fixed" content type, the iSee Ui program nodes (see URCap manual) will have additional functions to "Add" and "Remove" items from the menu during runtime. The "Store data" property can be switched on if these runtime changes have to be saved.

If the property is switched off, the next time the interface is reset, the changes to the menu during runtime will be lost, and the menu will be reset to the content configured in the iSee Ui Builder.
If the property is switched on, the changes to the menu during runtime will be stored.

Variable
A menu of type "Variable" does not have predefined items and will show the items contained in an array variable. Therefore, a "Source" needs to be configured for this type in the corresponding iSee Ui expression, as can be seen in figure 5.20b. The expression will only allow variables of types "ArrayOfBoolean", "ArrayOfInteger" and "ArrayOfDecimal". "ArrayOfString" is not allowed since PolyScope does not recognize a variable of this type.

With a "Variable" menu, the property "Open menu" is disabled because the content is unknown to the Builder and thus cannot be previewed.
The property "Store data" is not given for this type due to the fact that the content depends entirely on the content of the array variable. When a change, during runtime, of the content is desired, the content of the variable itself can be changed, and the menu content will change automatically.

5.8.3 Dropdown menu property: Selected item and index

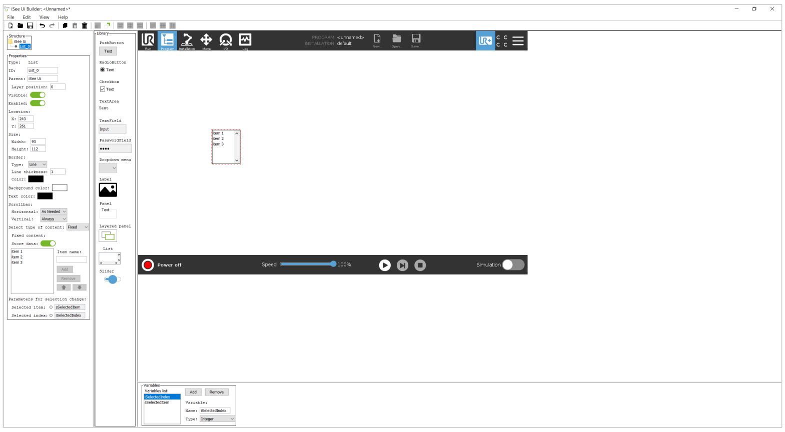
In order to detect an operator interaction when a new item is selected from the menu, an iSee Ui expression can be configured for the selected item and/or selected index. The item or index selected by the operator will only be written to the configured variable in the iSee Ui expression if an expression for respectively the item or index is configured.

Selected item
The object type allowed for the iSee Ui expression depends on the source of the content. If a fixed content type is selected, the menu will list items of type "String", thus object configured for the "Selected item" needs to be of type "String" as well.
For a variable content type, the content depends on the configured source. The object configured for the "Selected item" will thus also depend on the type of array configured for the source. If the source is not yet configured, the "Selected item" cannot be configured as well and will remain disabled until a source is given.

Selected index
The iSee Ui expression will only allow a variable of type "Integer". When an operator makes a selection change, the index of the selected item in the menu will be written to this variable, if it is configured. The indexing starts at '0', meaning that the index of the first item is '0' and the index of the last item is '$\text{size}_{\text{menu}} - 1$'.

5.9 List

A “List” item can be used for the operator to make a selection of, or to visualize a list of items. This item is similar to the “Dropdown menu” item discussed in section 5.8, the difference being that the items the operator can select are visible without having to first open a menu. As with the “Dropdown menu”, the content of the list can be fixed or variable. An example of a “List” item as well as its properties panel is shown in figures 5.21 and 5.22.



iSee Ui Builder: <Unnamed>*

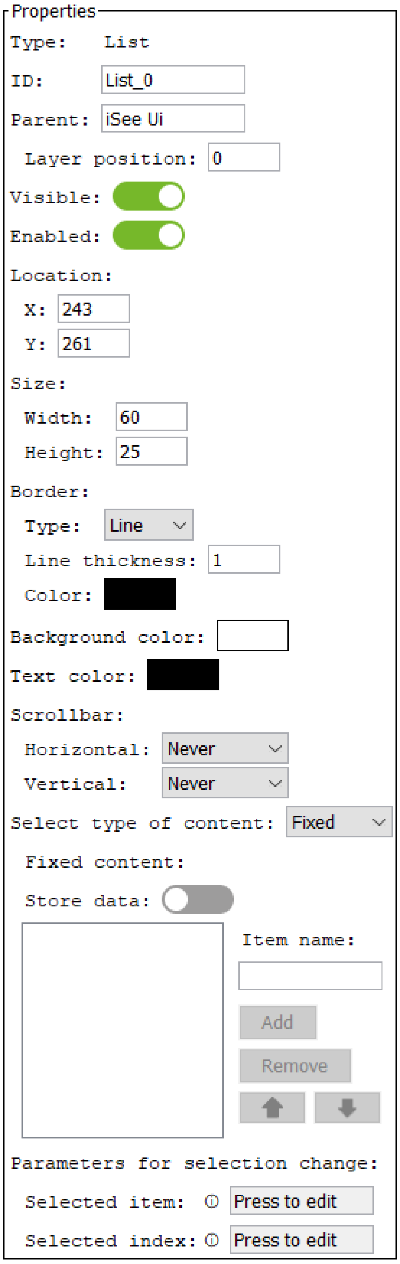
Properties
Type: List
ID: List_0
Parent: iSee Ui
Layer position: 0
Location: X: 243 Y: 261
Size: Width: 93 Height: 112
Border: Type: Line Line thickness: 1
Scrollbar: Horizontal: As Needed Vertical: Always
Select type of content: Fixed
Fixed content: item 1, item 2, item 3
Parameters for selection change:
Selected item: sSelectedItem
Selected index: iSelectedIndex

Variables
Variables list: iSelectedIndex, sSelectedItem
Variable: Name: iSelectedIndex Type: Integer

Figure 5.21: List and its properties

For a “List”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Scrollbars
- Content type
- Selected item, selected index



Properties

Type: List

ID: List_0

Parent: iSee Ui

Layer position: 0

Visible: ●

Enabled: ●

Location:

X: 243

Y: 261

Size:

Width: 60

Height: 25

Border:

Type: Line

Line thickness: 1

Color: ■

Background color: □

Text color: ■

Scrollbar:

Horizontal: Never

Vertical: Never

Select type of content: Fixed

Fixed content:

Store data: ○

Item name:

Parameters for selection change:

Selected item: ⓘ Press to edit

Selected index: ⓘ Press to edit

Figure 5.22: List properties

5.9.1 List property: Scrollbars

Both a vertical and horizontal scrollbar can be set for the list, such that if the list is too long, or entry lines are too long, they can be made viewable by the operator by use of

the scrollbars.
Both vertical and horizontal scrollbars can be set to:

- Never: the particular scrollbar will never be added.
- As Needed: if the entire content in that particular direction is viewable, then the scrollbar will not be added. From the moment the entire content is no longer viewable, a scrollbar will be added.
- Always: a scrollbar, for that particular direction, will always be present.

5.9.2 List property: Content type

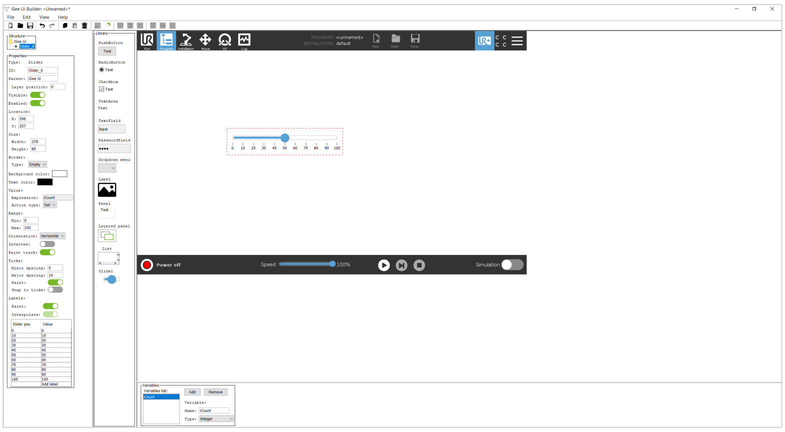
The "Content type" property is the same as the one for "Dropdown menu" item as discussed in section 5.8.2.

5.9.3 List property: Selected item and index

The "Selected item" and "Selected index" properties are the same as the ones for "Dropdown menu" item as discussed in section 5.8.3.

5.10 Slider

A "Slider" item can be used to visualize the value of a variable or analog IO, as well as its range of possible values. Similar to the "RadioButton" (section 5.6), the "Slider" can be configured to only read and display the value, or to write when the operator slides the slider button. The "Slider" can also be used similar to a "List" (section 5.9). That is, to give the operator a choice between items that can be selected by dragging the button to the desired item. One could even use a "Slider" item as a loading bar. An example of a "Slider" item as well as its properties panel is shown in figures 5.23 and 5.24.



Properties
Type: Slider
ID: Slider_0
Parent: iSee Ui
Layer position: 0
Visible: ●
Enabled: ●
Location:
X: 296
Y: 257
Size:
Width: 378
Height: 85
Border:
Type: Empty
Background color:
Text color:
Value:
Expression: iCount
Action type: Set
Range:
Min: 0
Max: 100
Orientation: Horizontal
Inverted: ○
Paint track: ●
Ticks:
Minor spacing: 0
Major spacing: 10
Paint: ●
Snap to ticks: ○
Labels:
Paint: ●
Interpolate: ●

Slider pos	Value
0	0
10	10
20	20
30	30
40	40
50	50
60	60
70	70
80	80
90	90
100	100
	Add label

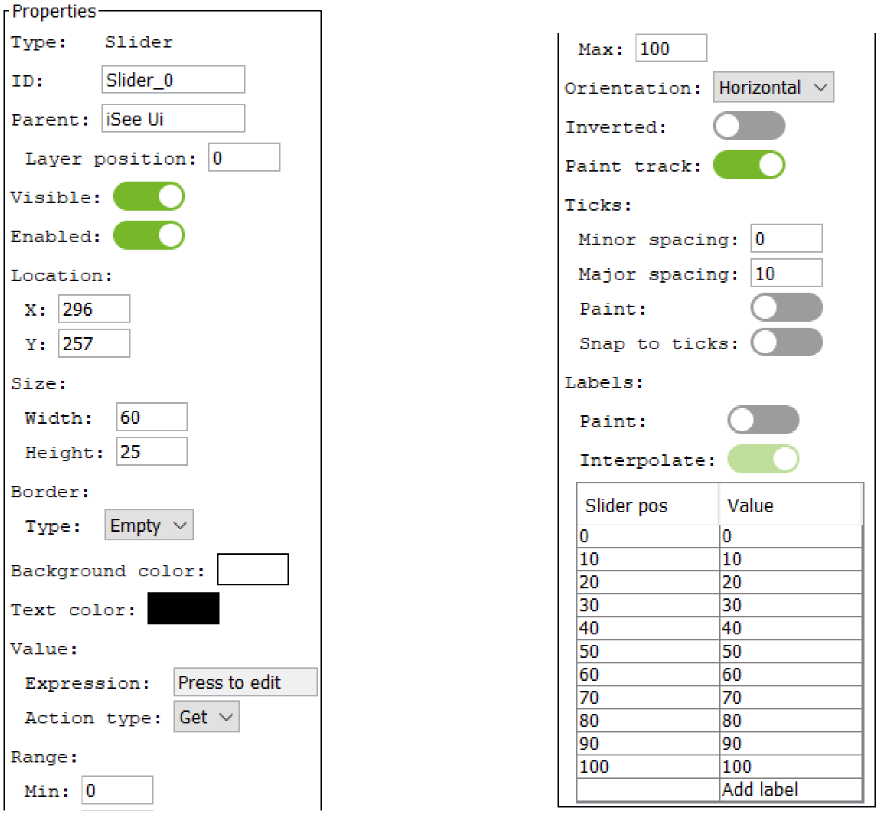
PROGRAM <unnamed>
INSTALLATION default

Variables list: iCount
Variable:
Name: iCount
Type: Integer

Figure 5.23: Slider and its properties

For a “Slider”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Value expression and action type
- Range
- Orientation
- Inverted
- Paint track
- Ticks
- Labels



Properties

Type: Slider

ID: Slider_0

Parent: iSee Ui

Layer position: 0

Visible: ☑

Enabled: ☑

Location:

X: 296

Y: 257

Size:

Width: 60

Height: 25

Border:

Type: Empty

Background color: □

Text color: ■

Value:

Expression: Press to edit

Action type: Get

Range:

Min: 0

Max: 100

Orientation: Horizontal

Inverted: ☐

Paint track: ☑

Ticks:

Minor spacing: 0

Major spacing: 10

Paint: ☐

Snap to ticks: ☐

Labels:

Paint: ☐

Interpolate: ☑

Slider pos	Value
0	0
10	10
20	20
30	30
40	40
50	50
60	60
70	70
80	80
90	90
100	100
	Add label

Figure 5.24: Slider properties

5.10.1 Slider property: Value expression and action type

Similar to the value expression and action type property of the “RadioButton”, discussed in section 5.6.3 and 5.6.4, the “Slider” can write a value to a variable or analog IO signal, or display the value.

To configure the “Slider”, an iSee Ui expression needs to be set for the “Value expression” property by pressing the “Press to edit” button. Once pressed, the iSee Ui expression panel will open as discussed in section 4.8.
For the value of the “Slider”, analog IO signals (integer values) as well as variables of types “Integer”, “Decimal” and “String” can be used. Depending on the type of the selected object, other properties are automatically set and/or disabled. This will be discussed in detail in the sections of these properties.

The property “Action type” can be used to configure the “Slider” to only display the value, or to write a value when the operator drags the button.
If the type is set to “Get”, any change of the slider button position done by the operator will be undone instantly. The slider will only display the actual value of the configured object in the value expression.
If the type is set to “Set”, the operator will be able to change the value of the configured value object by dragging the slider button to the desired value. In section 5.10.6 we will discuss the possibility of more accurate positioning of the slider button.

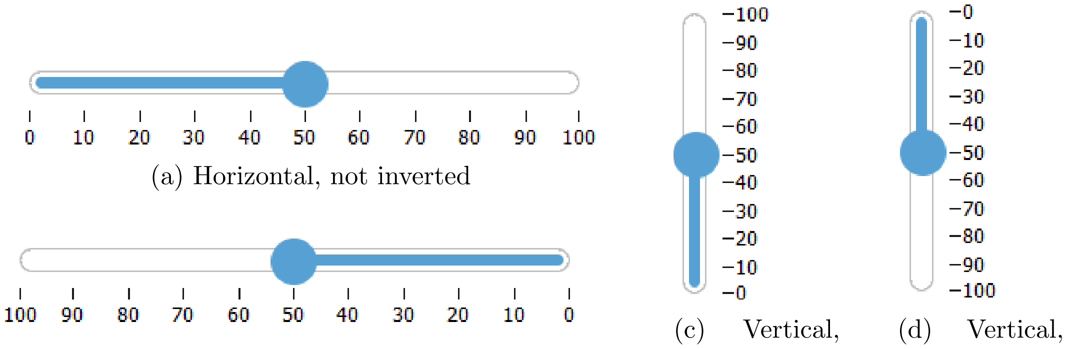
5.10.2 Slider property: Range

As mentioned, the “Slider” item allows one to visualize the value of an object in the range of possible values. The amount of possible values, slider positions, is determined by the range.
The “Range” property does not necessarily restrict the possible values that the operator can choose from. In section 5.10.7 we will discuss the possibility to map a label to a slider position. The “Range” restricts the possible slider positions.

The lowest slider position can be restricted by specifying the “Min” property of the “Range”.
The highest slider position can be restricted by specifying the “Max” property of the “Range”.

5.10.3 Slider property: Orientation

The “Orientation” property allows to specify if the “Slider” item needs to be placed horizontally or vertically. For some labels, especially text labels, a vertical orientation might be preferred to avoid the labels overlapping. Figure 5.25 shows the result of a horizontal or vertical orientation.



(a) Horizontal, not inverted

(b) Horizontal, inverted

(c) Vertical, not inverted

(d) Vertical, inverted

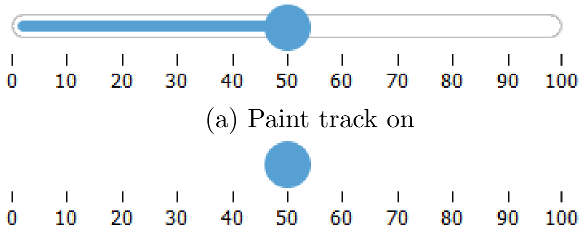
Figure 5.25: Slider examples of orientation and inverted

5.10.4 Slider property: Inverted

The “Inverted” property can be used to invert the direction of the value change of the slider.
For a horizontal slider, the value is increased from left to right. Inverting the slider will result in the value increasing from right to left.
For a vertical slider, the value is increased from bottom to top. Inverting the slider will result in the value increasing from top to bottom.
Figure 5.25 shows the result of inverting the slider for both horizontal and vertical orientation of the sliders.

5.10.5 Slider property: Paint track

The property “Paint track” can be used to configure if the track, with the blue progress line, needs to be visible or not. Switching this property off will result in only the slider button remaining visible, and the ticks and labels depending on their own “Paint” property. Figure 5.26 shows both a slider with “Paint track” switched on and off.



(a) Paint track on

(b) Paint track off

Figure 5.26: Slider examples of paint track

5.10.6 Slider property: Ticks

The property “Ticks” allows for the editing of the vertical lines (“thicks”), using the properties “Minor spacing”, “Major spacing”, “Paint” and “Snap to ticks”.

Tick spacing
Two types of ticks can be configured: minor and major ticks. The major ticks will have a longer vertical line than the minor ticks. The spacing between ticks can be configured for both, using the provided fields for “Minor spacing” and “Major spacing”.
When both spacings are set to ‘0’, the properties “Paint” and “Snap to ticks” will automatically be switched off and disabled until at least one spacing is higher than ‘0’.

Paint
With the “Paint” property, the user can configure whether the ticks have to be painted, visible, or not. Switching off this property will have no effect on the other “Ticks” properties.
The property will automatically be switched off and disabled if both “Tick spacing” fields have a ‘0’ value.

Snap to ticks
The property “Snap to ticks” can be used if the slider button needs to be “snapped” to the closest tick. When the operator drags the slider button, and this property is switched on, upon release of the button, the slider button will automatically move to the closest tick.

This property can be used even if the “Ticks Paint” property is switched off.
The property will automatically be switched off and disabled if both “Tick spacing” fields have a ‘0’ value.

When the operator displaces the button, the slider position is used with the labels (discussed in section 5.10.7) to determine the value that needs to be sent to the configured object in the value expression (discussed in section 5.10.1).
When the “Snap to ticks” is switched off, an interpolation of the label values has to be performed. For an object of type “String”, interpolation isn’t possible thus when using an object of this type, the “Snap to ticks” property will automatically be switched on and disabled.

5.10.7 Slider property: Labels

The property “Labels” allows for the editing of the slider labels, which in turn are the actual values to be used by the object configured in the value expression (discussed in section 5.10.1). The labels can be edited by use of the properties “Paint”, “Interpolate” and the labels table.

Paint
With the “Paint” property, the user can configure whether the labels have to be painted, visible, or not. Evidently, a good practice is to always switch this property on in order for the operator to have an idea what the slider is representing, and what the values are. In some cases, the labels are not necessary, for instance for creating a progress bar, and then the property can be switched off.

Interpolate
The property “Interpolate” can be used to minimize the number of Labels that are needed.
Any position the slider button can be placed at has to have a valid label value. Interpolating thus allows the user to avoid specifying a label for every slider position.

This property can only be used if the “Snap to ticks”, discussed in section 5.10.6, is switched on. When “Snap to ticks” is switched off, the slider can be placed at any slider position in the specified range (discussed in section 5.10.2), even at positions not represented by tick marks. Thus, providing a label for every position is impossible, and the “Interpolate” property will be automatically switched on and disabled.
The label value is then determined by interpolating the label values corresponding to the lowest and highest slider position, respectively the “Min” and “Max” values of the range.

When the “Interpolate” property is switched off, the label values can be chosen to be any value.

Labels table
The labels table allows the user to edit the labels used by the slider. The slider position is translated by use of this table to determine the value to be used by the object configured in the value expression (section 5.10.1).
When the value expression is changed, the labels table will be reevaluated, and automatically corrected in the event the label is of an incorrect type for the new expression.

Labels can be added by pressing the “Add label” button at the bottom of the table.
Labels can be removed by selecting the label and pressing the “delete” keyboard key.
Labels and slider positions can be changed by double-clicking the corresponding field with the left mouse button.

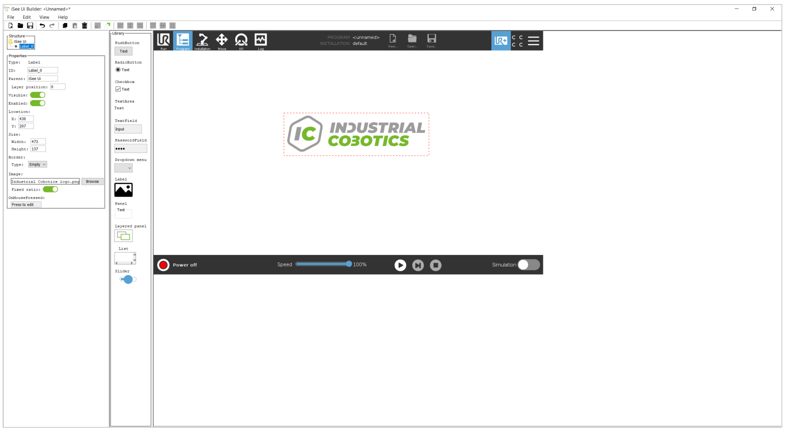
When the property “Interpolate” is switched on, only the labels corresponding to the “Min” and “Max” slider positions can be changed. When changing the value of one of these labels, the value of the other labels will be reevaluated.

When the property “Interpolate” is switched off, the table is reevaluated for any missing slider positions. A slider position, with a corresponding label, needs to be present for every tick mark. Any missing slider positions will be automatically added to the table with a default label value for that position. Slider positions, corresponding to tick marks, will not be able to be removed from the table. Pressing the “delete” keyboard key will only result in resetting the label value to its default value for that position.

Labels can always be added for slider positions that do not correspond to a tick mark. These labels will also be shown in the interface for the “Slider” item.

5.11 Label

A “Label” item can be used to display an image. An example of a “Label” item as well as its properties panel is shown in figures 5.27 and 5.28.

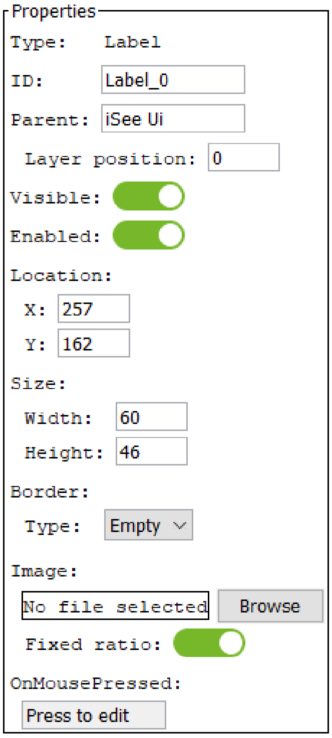


Properties
Type: Label
ID: Label_0
Parent: iSee Ui
Layer position: 0
Visible:
Enabled:
Location:
X: 430
Y: 207
Size:
Width: 473
Height: 137
Border:
Type: Empty
Image:
Industrial Cobotics logo.png Browse
Fixed ratio:
OnMousePressed:
Press to edit

Figure 5.27: Label and its properties

For a “Label”, the following properties, additional to the ones explained in section 5.1, can be customized:

- Image
- On Mouse Pressed



Properties

Type: Label

ID: Label_0

Parent: iSee Ui

Layer position: 0

Visible: ●

Enabled: ●

Location:

X: 257

Y: 162

Size:

Width: 60

Height: 46

Border:

Type: Empty

Image:

No file selected

Fixed ratio: ●

OnMousePressed:

Figure 5.28: Label properties

5.11.1 Label property: Image

The “Image” property can be used to specify the image file to be used by the “Label” item.
Furthermore, the image ratio can be maintained when resizing by switching on the “Fixed ratio”. When this option is switched on, any resizing, either with mouse interaction or changing the value of the width or height in the properties panel, will result in a proportional change for the other size dimension.

5.11.2 Label property: On Mouse Pressed

An “On Mouse Pressed” property can be configured, similar to the property of a “PushButton” discussed in section 5.5.3. This property allows for the creation of a button item with an image instead of text.

5.12 Panel

A “Panel” item can be used to group items together. The “Panel” item can thus provide structure. Furthermore, items placed inside a “Panel” item will be placed relative to the item's position, as discussed in section 5.1.5. Changing the location of the "Panel" item will thus also change the location, on the screen, of each of the items that are contained in the "Panel". An example of a "Panel" item as well as its properties panel is shown in figures 5.29 and 5.30.

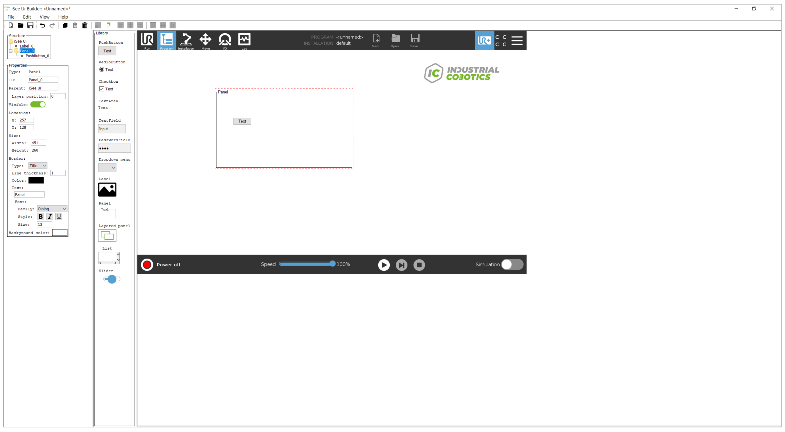
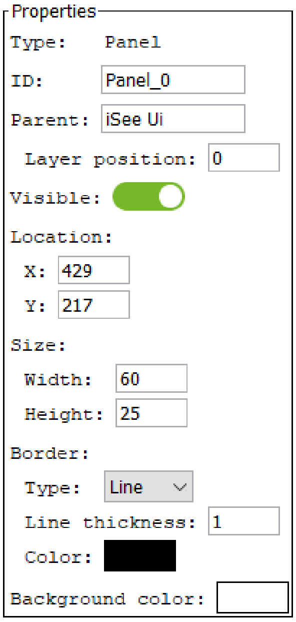


Figure 5.29: Panel and its properties

For a "Panel", the following properties, being the general properties explained in section 5.1, can be customized:



Properties
Type: Panel
ID: Panel_0
Parent: iSee Ui
Layer position: 0
Visible:
Location:
X: 429
Y: 217
Size:
Width: 60
Height: 25
Border:
Type: Line
Line thickness: 1
Color: ■
Background color: □

Figure 5.30: Panel properties

5.13 Layered panel

A “Layered panel” item is similar to a “Panel” item (discussed in section 5.12), it can also be used to group items together. The difference is that with a “Layered panel”, the items grouped by it can be given a depth, as discussed in section 5.1.3. An example of a “Layered panel” item as well as its properties panel is shown in figures 5.31 and 5.32.

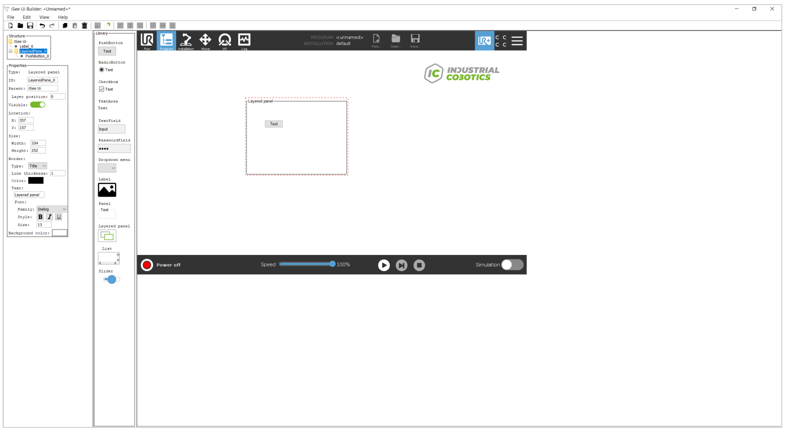
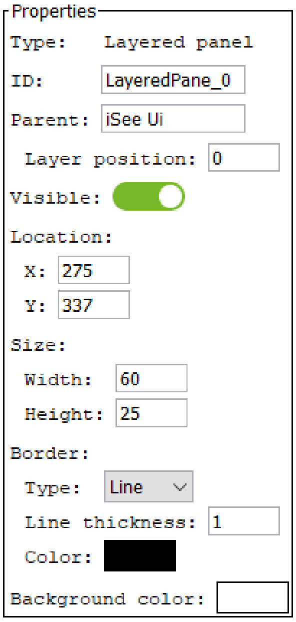


Figure 5.31: Layered panel and its properties

For a “Layered panel”, the following properties, being the general properties explained in section 5.1, can be customized:



Properties

Type: Layered panel

ID: LayeredPane_0

Parent: iSee Ui

Layer position: 0

Visible:

Location:

X: 275

Y: 337

Size:

Width: 60

Height: 25

Border:

Type: Line

Line thickness: 1

Color: ■

Background color: □

Figure 5.32: Layered panel properties