



Input Validations

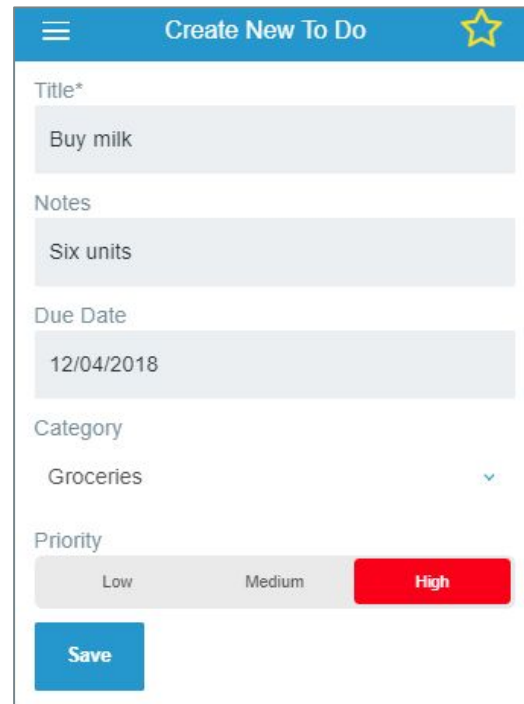


Topics

- Built-in Validations
- Custom Validations
- Validation Messages

Validating User Inputs

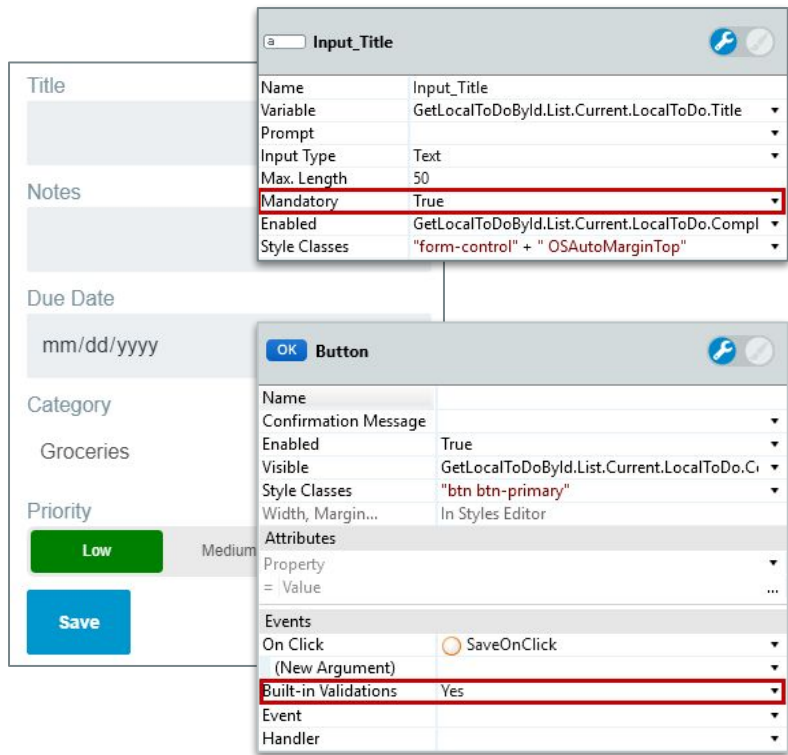
- Applications should limit user mistakes
 - ... but they are inevitable
- Input fields may...
 - Have different data types
 - Depend on other user input values
 - Require business-specific validations
- OutSystems provides mechanisms to help implementing input validations



The screenshot shows a mobile application interface for creating a new to-do item. The header is blue with a hamburger menu icon, the text 'Create New To Do', and a yellow star icon. The form contains several input fields: 'Title*' with the value 'Buy milk', 'Notes' with the value 'Six units', 'Due Date' with the value '12/04/2018', and 'Category' with a dropdown menu showing 'Groceries'. Below these is a 'Priority' section with three buttons: 'Low' (disabled), 'Medium' (disabled), and 'High' (active, highlighted in red). At the bottom is a blue 'Save' button.

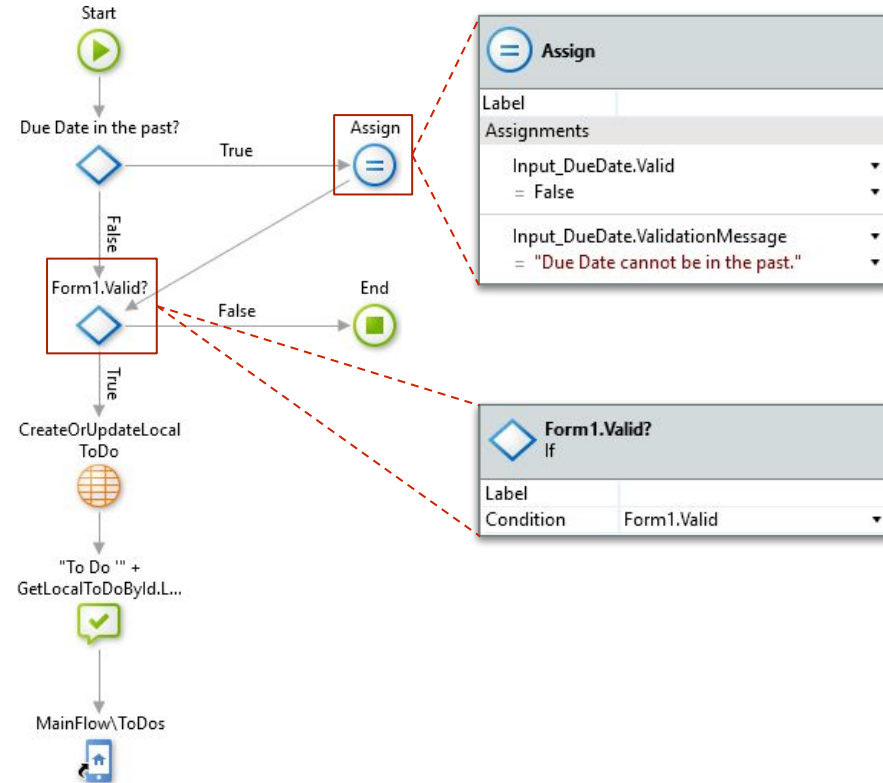
Built-in Validations

- OutSystems provides **built-in validations** for Input Widgets
 - **Mandatory fields** must be filled
 - User entered input values must **comply with the data types** of the Variables bound to the Widgets
- Validations are performed automatically
- Buttons inside a Form have a Built-Validations Property
 - Set to **Yes** to perform built-in validations



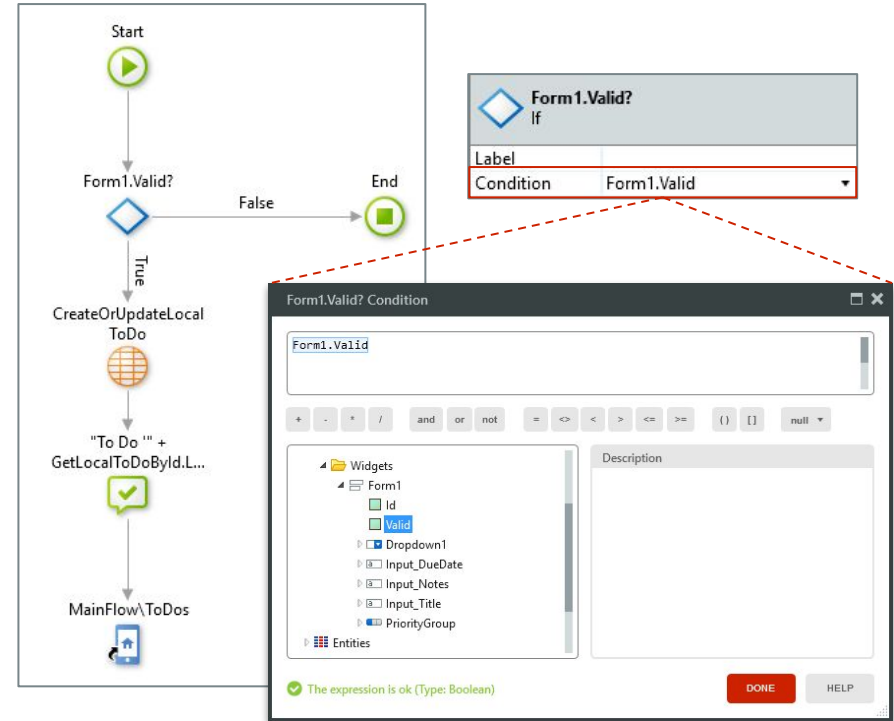
Custom Validations

- Forms inputs can have custom validations
 - Must be performed in the Action flow
 - For invalid Form Input fields
 - Set the **Valid** property to False
 - Set the **ValidationMessage** accordingly
- The **Form.Valid** property should be checked after the last custom validation
 - If one Input is not Valid, the Form is automatically not Valid
 - The **Form.Valid** property cannot be explicitly assigned



Checking Built-in Validations

- Form.Valid property should be checked even if there are no custom validations
 - To avoid storing / manipulating invalid data
- OutSystems automatically performs the validations
 - But it is up to the developer to use that in the logic



Validation Messages

- The Valid property of the Input widgets are checked when rendering
- If Valid property is set to True:
 - Displays the regular widget
- If Valid property is set to False:
 - Displays the regular widget
 - Applies the specific styling
 - e.g. a red border
 - Displays the Validation Message when the input gains focus

The screenshot shows a mobile application interface for creating a new to-do item. The title bar is blue with a hamburger menu icon, the text 'Create New To Do', and a yellow star icon. The form contains several fields: 'Title*' with the value 'Book Vacations', 'Notes' with the value '15 days', 'Due Date' with the value '12/03/2018', 'Category' with a dropdown menu showing 'Travel', and 'Priority' with three buttons: 'Low', 'Medium' (which is highlighted in orange), and 'High'. A red border is visible around the 'Due Date' input field, and a red error message 'Due Date cannot be in the past.' is displayed below it. At the bottom left, there is a blue 'Save' button.

Summary

- Built-in Validations
- Custom Validations
- Validation Messages

A city skyline at night, featuring several tall skyscrapers with illuminated windows. The foreground is dominated by long, horizontal light trails in white, yellow, and red, suggesting fast-moving traffic or light painting. The overall scene is a vibrant, high-energy urban environment.

Input Validations
Thank You!