Validators

Last Modified on 17.09.2026

Ensure Data Quality in Your Applications

Introduction

Validators allow you to create custom validation rules for form fields within your target applications. With Validators, you can measure data quality problems first, then decide how to act: from silent monitoring, to real-time feedback, to submit-time intervention.

Every Validator runs in one of three modes: **Measure**, **Visible**, or **Warning**. A typical workflow starts in Measure mode to quantify compliance issues, then progresses to Visible or Warning mode once the data supports intervention.

Key concepts

Validators and Field Validations

A **Validator** represents a form in your target application. It bundles together everything needed to validate that form: **Field Validations** on the individual fields, a **Completion Button** on the form's submit or save button (required in every mode), and a mode that controls how the Validator behaves.

Each **Field Validation** targets a single field and defines what format the data should follow (either a regular expression or a "can't be empty" rule), along with hint text that explains the expected format to the user.

For example, a Validator for a patient registration form might contain three Field Validations (date of birth, insurance ID, and email address) with its Completion Button on the form's "Save" button. In Warning mode, clicking Save with non-compliant fields triggers a Warning message listing what needs attention.

Modes

Every Validator runs in one of three modes. You choose the mode when you create the Validator and can change it at any time. All three modes require a Completion Button (see below).

Mode	End-user UI	What it does
Measure	None	Silently evaluates field rules and collects compliance data. End users see no indication that validation is running. Use this mode to discover and quantify data quality problems before deciding to act.
Visible	Field-level feedback	Displays an input hint when the user focuses on a field, and shows a green or red border after the user leaves the field to indicate whether their input is valid. No intervention at submit time.
Warning	Field-level feedback + Warning message	Provides the same field-level feedback as Visible mode. Additionally, when the user clicks a Completion Button (such as Save or Submit) and one or more fields are non-compliant, a Warning message appears listing the fields that need attention. The user can correct the fields or choose to proceed anyway.

The modes form a progression. A common workflow is to deploy a Validator in Measure mode first to quantify the problem, then switch to Visible or Warning mode once you have data to support the decision.

Completion Buttons

A **Completion Button** is the submit, save, or next-step button in your target application that you select when configuring a Validator. Every Validator requires a primary Completion Button in all three modes (Measure, Visible, and Warning). You can optionally add a secondary button (for example, if the form has both "Save" and "Save and close").

Because a Completion Button is present in every mode, a compliance event is recorded whenever the user clicks it. You get submit-level compliance data in Measure and Visible modes as well as Warning, letting you understand how often non-compliant data reaches submission and decide how much intervention you need.

- **Warning mode:** When the user clicks the Completion Button and any Field Validations are non-compliant, the Warning message appears.
- **Measure and Visible modes:** The Completion Button records compliance data without any user-facing intervention.

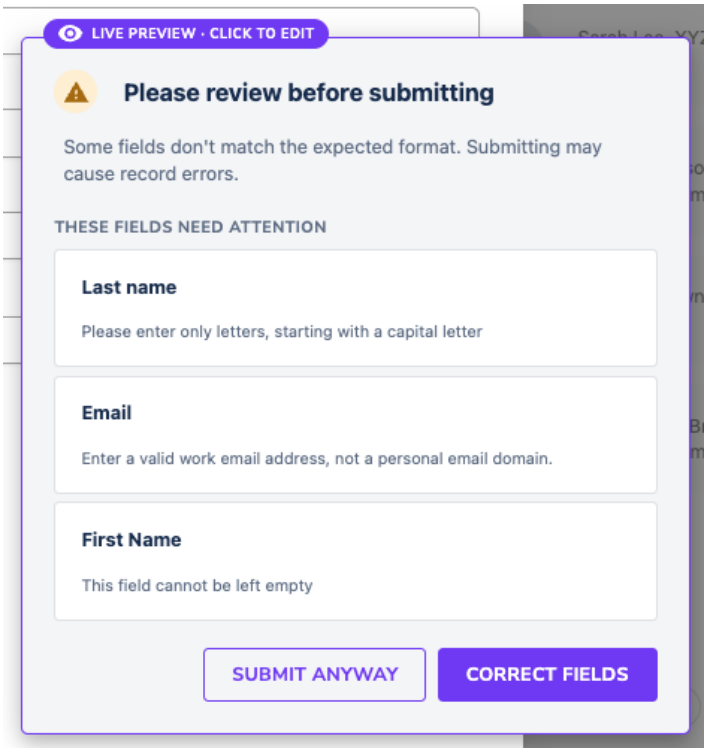
The primary Completion Button is recorded as part of the Validator creation flow. A secondary button can be added afterwards in the Validator settings. A maximum of two buttons can be configured per Validator. Save-as-draft buttons should not be used as Completion Buttons.

Warning message

The Warning message appears in Warning mode when the user clicks a Completion Button and one or more Field Validations are non-compliant. It contains:

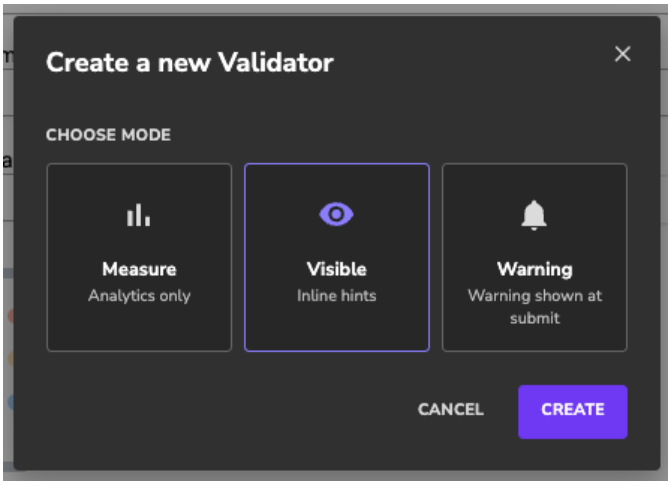
1. A header (default: "These fields need attention")
2. A title you can edit (default: "Please review before submitting")
3. A subtitle you can edit (default: "Fields don't match the expected format. Submitting may cause record errors.")
4. An auto-generated list of all non-compliant fields with their hint text
5. Two buttons: **"Submit anyway"** (lets the user proceed without correcting) and **"Correct fields"** (dismisses the message so the user can fix the issues)

You can edit the title and subtitle per Validator in the Editor. The header text and button labels can be customized globally in the Userlane Portal under the customize settings.



How to create a Validator

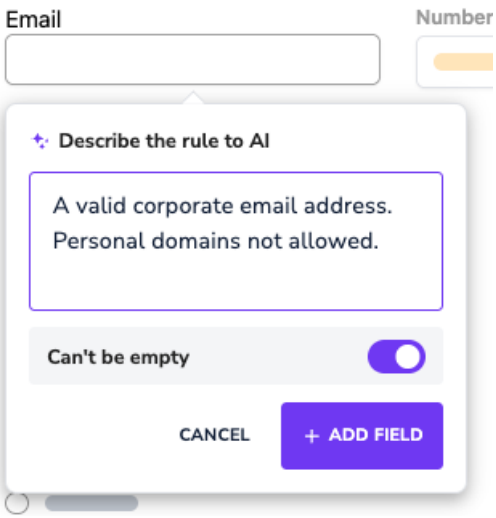
1. **Choose the mode.** In the Userlane Editor, select "Create new" from the Validators section. A modal prompts you to choose the mode (Measure, Visible, or Warning) before you begin recording. You can change the mode later without losing historical compliance data.



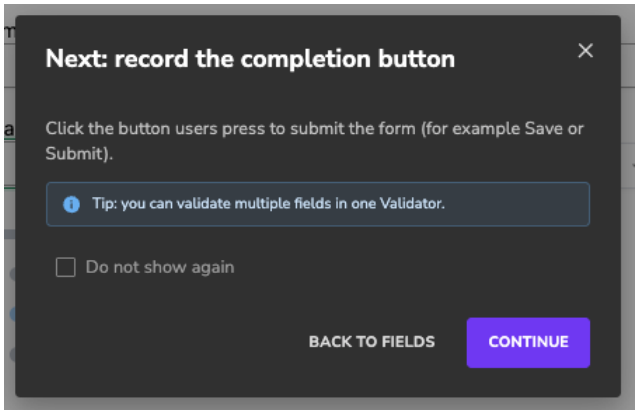
2. **Record your fields.** After selecting a mode, you enter recording mode on the target application. Click on a form field to add it as a Field Validation. For each field, a prompt appears where you can:

    - **Describe the rule to the AI:** Write a plain-language description of the expected format (for example, "A valid email address" or "Must be all capital letters, no numbers allowed"). The AI uses this description to generate the regular expression, draft the inline hint for end users, and auto-name the Field Validation, so a single description does all three.
    - **Toggle "Can't be empty":** Enable this if the field must not be left blank. You can combine it with an AI-generated rule.

    Click **"+ Add field"** to continue selecting more fields. You can record as many fields as needed in a single flow.

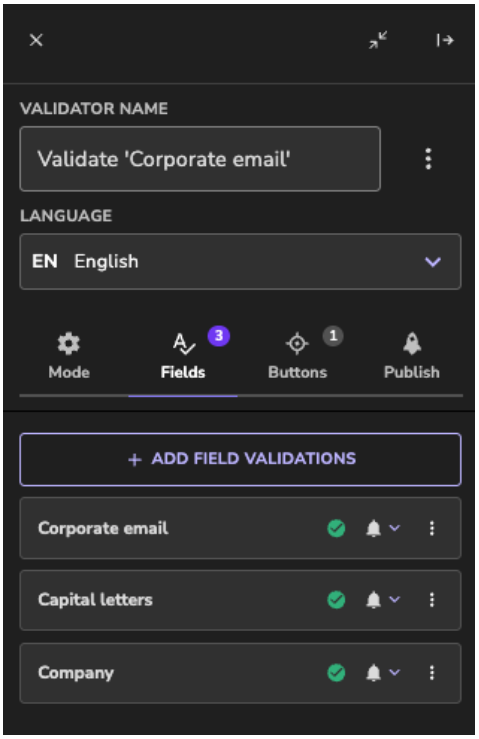


3. **Record the Completion Button.** After you finish adding fields, a prompt asks you to click the button that users press to submit the form (for example, Save or Submit). This step is required for every Validator, in all three modes, and records the primary Completion Button. A secondary button can optionally be added later in the Validator settings.



4. **Review and configure.** The Validator settings panel opens with your auto-named Validator and all recorded Field Validations. From here you can:

    - Review and edit Field Validation rules, hint text, or names
    - Add more Field Validations
    - Add a secondary Completion Button
    - Customize the Warning message title and subtitle (Warning mode only)
    - Set page or user segments



5. **Publish.** Publishing the Validator activates all Field Validations within it.

End-user experience

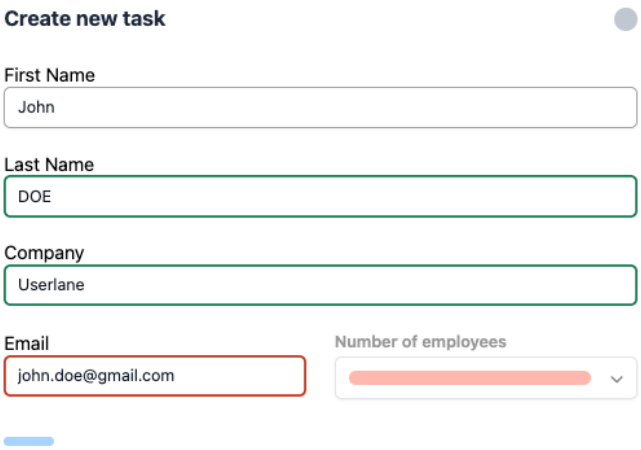
Measure mode

End users see no UI. Validation runs silently in the background and compliance data is collected for analytics.

Visible mode

When the user focuses on a validated field, the hint text appears in a prompt near the field. After the user leaves the field:

- If the input matches the rule, the field is framed in **green**.
- If the input does not match, the field is framed in **red**.



Warning mode

Users see the same field-level feedback as Visible mode. Additionally, when the user clicks a Completion

Button and one or more fields are non-compliant, the Warning message appears.

The user has two options:

- **"Correct fields":** Dismisses the Warning message so they can fix the flagged fields.
- **"Submit anyway":** Proceeds with the submission despite the non-compliant fields. This action is tracked as a separate analytics event.

Data privacy

Validators evaluate data locally in the user's browser. The validation rule (regular expression) runs against the input, and only a pass or fail result is recorded. **The actual values that users type are never sent to Userlane.** This applies to all modes, including Measure mode.

Best practices

- **Start with Measure mode.** Deploy a Validator in Measure mode first to understand where data quality issues exist. Use the compliance data to build a case for intervention before switching to Visible or Warning mode.
- **Review the AI-generated hint text.** The hint message is auto-drafted from your rule description, but it is the primary guidance your end users will see. Check that it is specific about the expected format and includes an example where possible (for example, "Enter a date in MM/DD/YYYY format, e.g. 03/15/2026").
- **Test before publishing.** Validate your regex rules with test inputs, both valid and invalid, before publishing to end users.
- **Keep the Warning message actionable.** The title and subtitle should tell the user what went wrong and what to do next. Avoid vague language.

Frequently asked questions

Q: What field types do Validators support?
Validators work with text fields, number fields, email fields, and other standard input elements (`<input>` and `<textarea>` HTML elements). Dropdown menus, checkboxes, radio buttons, and other non-text field types are not supported at this time.

Q: Can a single Validator cover multiple fields?
Yes. Each Validator can contain multiple Field Validations, each targeting a different field on the page. This is how you validate an entire form with a single Validator.

Q: Can I change the mode after publishing?
Yes. You can switch between Measure, Visible, and Warning modes at any time. Your historical compliance data is preserved, and the mode change is marked in your analytics timeline.

Q: Does Userlane see the data my users enter?
No. Validation runs locally in the browser. Userlane records whether the input matched the validation rule (pass or fail) but never receives the actual input values.

Q: What happens if a user clicks "Submit anyway"?
The form is submitted normally. The override is recorded as a separate analytics event, so you can track how often users choose to proceed with non-compliant data.

Q: Can I use Validators without showing anything to end users?
Yes. Set the Validator to Measure mode. Validation rules run and compliance data is collected, but end users see no UI at all.

Q: What happens when I add a new Field Validation to an existing Validator?
After you publish the updated Validator, the new field is included in validation. If the Validator is in Warning mode, the Warning message will reflect the new field's compliance state.

Troubleshooting

If you encounter any issues with Validators, please contact our support team for assistance.