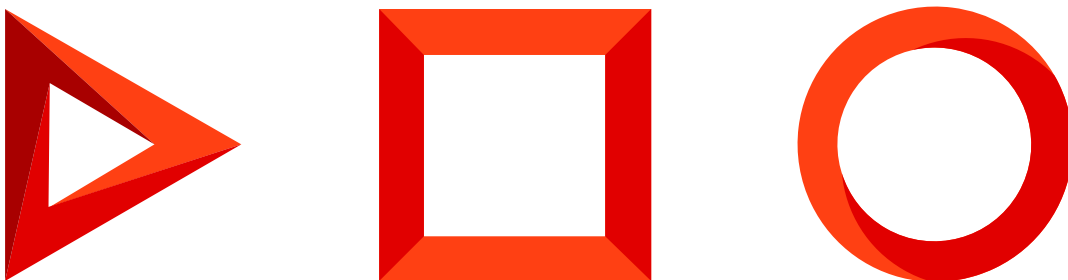


Freedom UI page customization

Display the value of a system variable

Version 8.0



This documentation is provided under restrictions on use and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this documentation, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Table of Contents

| | |
|---|----|
| Display the value of a system variable | 4 |
| Display the values of system variables on a page | 5 |
| 1. Set up the page UI | 5 |
| 2. Set up the retrieval of system variable values | 7 |
| Outcome of the example | 9 |
| System variables list in Freedom UI | 10 |

Display the value of a system variable

Beginner

Creatio 8 Atlas uses the `sdk.SysValuesService` service to manage system variables.

In Creatio 8 Atlas, you can access system variables differently from the previous versions. In this version, write the system variable name in lowercase without `_` delimiters and the `CURRENT` prefix. View the complete list of system variables in a separate article: [System variables list in Freedom UI](#).

To display the value of a system variable on a page:

1. **Add a page inscription** to display the values of system variables on step 1 of the [Freedom UI page customization procedure](#) if needed.
2. **Set up how to display the value of a system variable** on the page on step 2 of the [Freedom UI page customization procedure](#).
 - a. Enable the `sdk.SysValuesService` system variable service. To do this, add the `@creatio-devkit/common` dependency to the AMD module.

View an example that adds a dependency to the `UsrAppClientSchemaName` AMD module below.

AMD module dependencies

```
/* AMD module declaration. */
define("UsrAppClientSchemaName", /**SCHEMA_DEPS*/["@creatio-devkit/common"] /**SCHEMA_DEPS*/
    return {
        ...
    };
});
```

- b. Add an attribute that stores data to the `viewModelConfig` schema section. Add the attribute similarly to the [procedure for setting up the field display condition](#).
- c. Bind the `caption` property to the corresponding model attribute in the `viewConfigDiff` schema section. Bind the property similarly to the [setup procedure for the field display condition](#). Instead of the `visible` property, use the `caption` property that displays the text in the element.
- d. Add a custom implementation of the `crt.HandlerViewModelInitRequest` system query handler to the `handlers` schema section. The handler is executed when the `View` model is initialized. Depending on the value of the attribute (`true` or `false`), the handler executes different business logic.
 - a. Instantiate the system value service from `@creatio-devkit/common`.
 - b. Load system values.
 - c. Calculate the value and write the calculation result to the corresponding attribute if needed.

View an example of a `crt.HandlerViewModelInitRequest` query handler with `someVariable` calculation result

written in the `SomeAttributeName` attribute below.

handlers **schema section**

```
handlers: /**SCHEMA_HANDLERS*/[
  {
    request: "crt.HandleViewModelInitRequest",
    /* Custom implementation of a system query handler. */
    handler: async (request, next) => {
      /* Instantiate the system value service from @creatio-devkit/common. */
      const sysValuesService = new sdk.SysValuesService();
      /* Load system values. */
      const sysValues = await sysValuesService.loadSysValues();
      /* Calculate the value. */
      const someVariable = sysValues.systemVariable;
      ...;
      /* Write the result of someVariable calculation to the SomeAttributeName attri
      request.$context.SomeAttributeName = someVariable;
      /* Call the next handler if it exists and return its result. */
      return next?.handle(request);
    }
  }
]/**SCHEMA_HANDLERS*/,
```

Detailed example: [Display the values of system variables on a page.](#)

Display the values of system variables on a page




Example. Display the following on the record page of the custom [*System variables*] section:

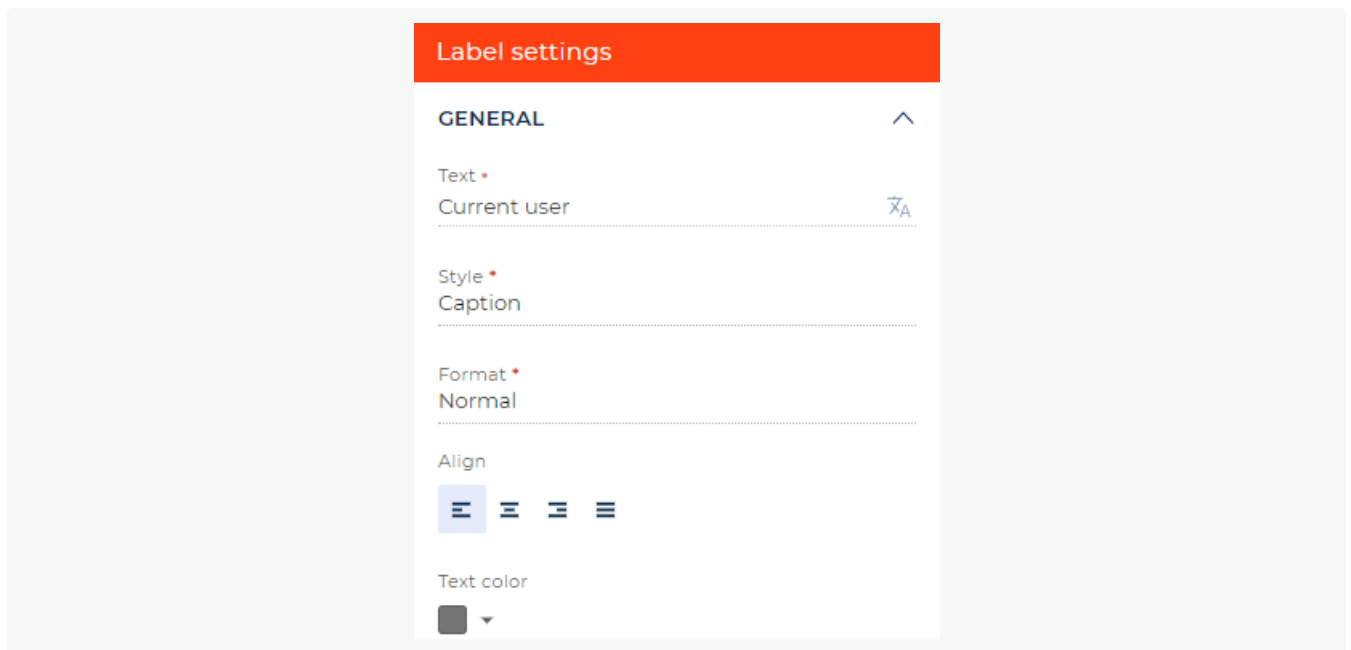
- the name of the current user
- the time offset value in hours between the time zone of the current contact and universal time (UTC)

Retrieve the values from the corresponding system variables.

1. Set up the page UI

1. Create a custom `System variables` app based on the [*Records & business processes*] template. To do this, follow the guide in the user documentation: [Create a custom app.](#)
2. Open the [*System variables form page*] page in the working area of the `System variables` app page.

3. Delete the [*Name*] field the [*Requests form page*] page includes by default.
4. Add the **label of the current contact**.
 - a. Add a [*Label*] type component to the working area of the Freedom UI Designer.
 - b. Click  in the action panel of the Freedom UI Designer and fill out the **label properties** in the setup area.
 - Set [*Title*] to "Current user."
 - Select "Caption" in the [*Style*] property.
 - Select gray in the [*Text color*] property.



5. Add the following **labels** in a similar way:
 - the value of the current contact name from the system variable
 - the time offset from UTC
 - the time offset value from UTC from the system variable

View the properties of the labels to add in the table below.

Label property values

| Element | Property | Property value |
|---|-----------------------|-------------------------------|
| The label that contains the value of the current contact name from the system variable | [<i>Title</i>] | "Current user (value)" |
| | [<i>Style</i>] | Select "Body text" |
| The label of the time offset from UTC | [<i>Title</i>] | "Contact time offset" |
| | [<i>Style</i>] | Select "Caption" |
| | [<i>Text color</i>] | Select the gray color |
| The label that contains the time offset value from UTC from the system variable | [<i>Title</i>] | "Contact time offset (value)" |
| | [<i>Style</i>] | Select "Body text" |

6. Click  in the action panel of the Freedom UI Designer. After you save the page settings, Creatio opens the source code of the Freedom UI page.

2. Set up the retrieval of system variable values

Configure the business logic in the Client Module Designer. For this example, set up the retrieval of system variable values

1. Enable the `sdk.SysValuesService` system variable service. To do this, add `@creatio-devkit/common` to the AMD module as a dependency.

AMD module dependencies

```

/* Declare the AMD module. */
define("UsrAppSystemvariable_FormPage", /**SCHEMA_DEPS*/["@creatio-devkit/common"] /**SCHEMA_
    return {
        ...
    };
});

```

2. Add the following **attributes** to the `viewModelConfig` schema section:

- `CurrentUser`. Stores the name of the current contact.
- `ContactTimezone`. Stores the time offset in hours between the time zone of the current contact and UTC.

`viewModelConfig` schema section

```
viewModelConfig: /**SCHEMA_VIEW_MODEL_CONFIG*/{
  "attributes": {
    ...,
    /* The attribute that stores the name of the current contact. */
    "CurrentUser": {},
    /* The attribute that stores the time offset in hours between the time zone of the cu
    "ContactTimezone": {}
  }
}/**SCHEMA_VIEW_MODEL_CONFIG*/,
```

3. Change the `caption` **property value** in the `viewConfigDiff` schema section:

- `CurrentUser` for the `CurrentUserValue` element
- `ContactTimezone` for the `ContactTimeOffsetValue` element

The `caption` property handles the text contained in the element.

`viewConfigDiff` schema section

```
viewConfigDiff: /**SCHEMA_VIEW_CONFIG_DIFF*/[
  ...,
  {
    "operation": "insert",
    "name": "CurrentUserValue",
    "values": {
      ...,
      /* Bind the CurrentUser attribute to the caption property. */
      "caption": "$CurrentUser",
      ...
    },
  },
  ...,
  {
    "operation": "insert",
    "name": "ContactTimeOffsetValue",
    "values": {
      ...,
      /* Bind the ContactTimezone attribute to the caption property. */
      "caption": "$ContactTimezone",
```



```

        ...
    },
    ...
}
]/**SCHEMA_VIEW_CONFIG_DIFF*/,

```

4. Add a custom implementation of the `crt.HandlerViewModelInitRequest` system query handler to the `handlers` schema section. Execute the handler when Creatio initializes the `View` model.
 - a. Create an instance of the system value service from `@creatio-devkit/common`.
 - b. Upload the system values.
 - c. Specify the name of the current user contact in the `CurrentUser` attribute.
 - d. Convert the time zone offset value from minutes to hours and specify the converted value in the `ContactTimezone` attribute.

`handlers` schema section

```

handlers: /**SCHEMA_HANDLERS*/[
  {
    request: "crt.HandleViewModelInitRequest",
    /* The custom implementation of the system query handler. */
    handler: async (request, next) => {
      /* Create an instance of the system value service from @creatio-devkit/common. */
      const sysValuesService = new sdk.SysValuesService();
      /* Upload the system values. */
      const sysValues = await sysValuesService.loadSysValues();
      /* Specify the name of the current user contact in the CurrentUser attribute. */
      request.$context.CurrentUser = sysValues.userContact.displayValue;
      /* Convert the time zone offset value from minutes to hours and specify the converted value in the ContactTimezone attribute. */
      const offset = sysValues.userTimezoneOffset;
      const offsetDisplayValue = (offset > 0 ? '+' : '') + (offset / 60) + 'h';
      request.$context.ContactTimezone = offsetDisplayValue;
      /* Call the next handler if it exists and return its result. */
      return next?.handle(request);
    }
  }
] /**SCHEMA_HANDLERS*/,

```

[Complete source code of the page schema](#)

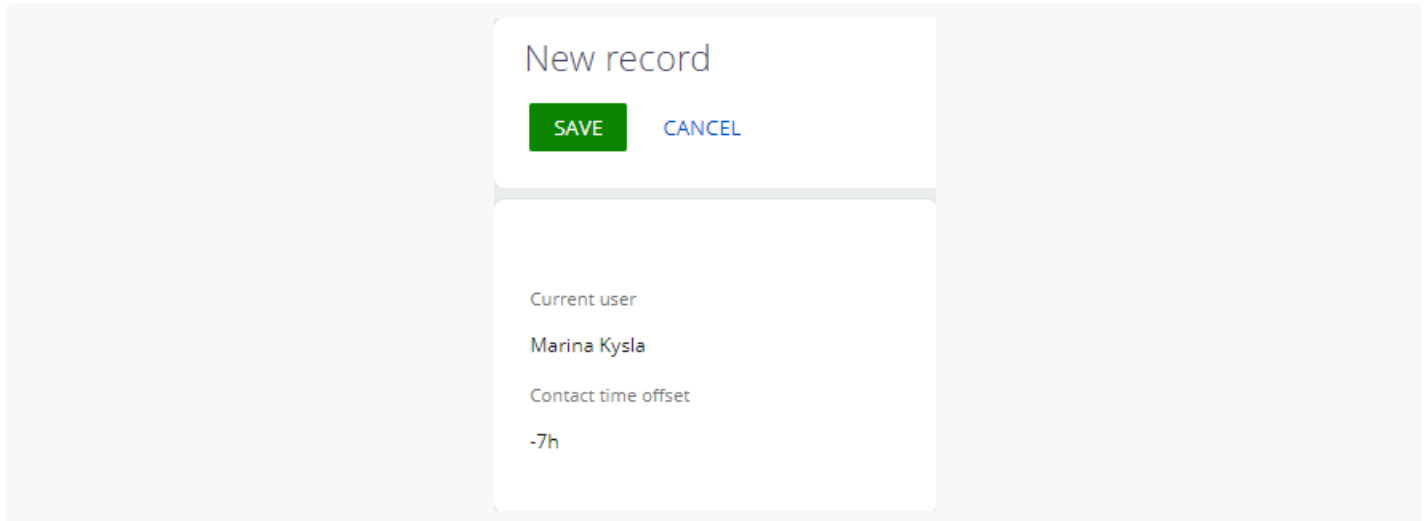
5. Click [Save] on the Client Module Designer's toolbar.

Outcome of the example

To **view the outcome of the example**:

1. Open the `System variables` app page and click [*Run app*].
2. Click [*New*] on the `System variables` app toolbar.

As a result, Creatio will display the name of the current user and the time offset value in hours between the time zone of the current contact and UTC on the record page of the custom [*System variables*] section. The values will be retrieved from the corresponding system variables.



System variables list in Freedom UI JS

 Beginner

In Creatio 8 Atlas, you can access system variables differently from the previous versions. In this version, write the system variable name in lowercase without `_` delimiters and the `CURRENT` prefix.

Full list of the system variables in Freedom UI

| System variable | Type | Description |
|--|------------------------|---|
| <code>maintainer</code> | Lookup Value | Maintainer. Use the [<i>Publisher</i>] (<code>Maintainer</code> code) system setting to set the value. Learn more: Configuration (user documentation). |
| <code>maxEntitySchemaNameLength</code> | number | Max length of entity schema name |
| <code>moneyDisplayPrecision</code> | number | Precision of decimals in the displayed multicurrency control |
| <code>primaryLanguage</code> | Lookup Value With Code | Primary language |
| <code>user</code> | string | Username |
| <code>userAccount</code> | Lookup Value | Account linked to the user |
| <code>userContact</code> | Lookup Value | Contact linked to the user |
| <code>userCulture</code> | Lookup Value | Creatio UI language of the current user |
| <code>userTimezoneCode</code> | string | Code of the user time zone |