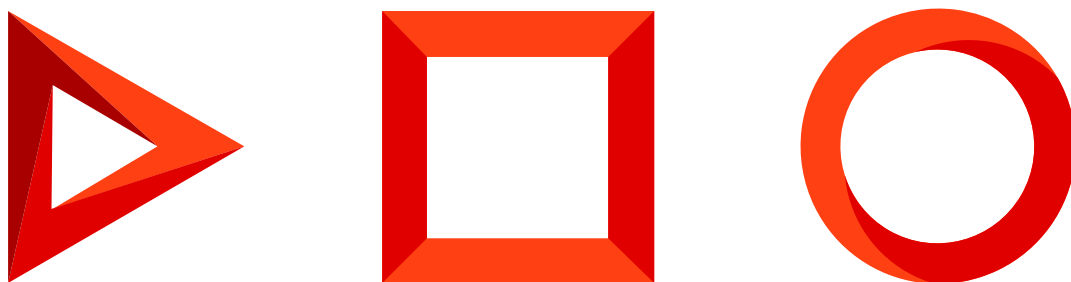


Marketplace app development

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

Marketplace app development	4
Creating of application packages	4
App UX recommendations	4
Integration with Creatio and public API	10
Binding data to package	10
Marketplace application licensing	14
Licensing methods	15
Migrating apps to the testing environment	17

Marketplace app development



Creating of application packages

Any Marketplace application is a specific set of packages, that are used to modify the configuration.

A [Creatio package](#) is a collection of configuration elements that implements particular block of functionality.

The dividing principle of the application functionality into packages is implemented using [dependencies](#).

When a customer [develops new functionality](#) and, therefore, creates new packages they have to use a revision control system (SVN). [Schemas](#) are the main content of the packages.

App UX recommendations

The negative user experience (in installing, configuring and working with the product) is the one of the main factors that significantly reduce the probability of purchasing a product after a trial. Potential customers often perceive quality and useful products negatively if they require complex and time-consuming setup.

We have come up with a number of recommendations to help avoid typical errors with UX of product setup and user's first impressions.

Clear navigation in the system

Convenient and logically organized navigation will help a new user to find the application, navigate to the necessary function and use it. When designing the navigation, consider following:

1. Displaying a section in the workplace


If the application functions include a new Creatio section, it should be displayed in at least one workplace. A workplace is a set of system sections available to a group of users. When adding a new section, take into account that Creatio workplaces are typically connected to specific functional and organizational user roles (i.e., sales, contact center, administration, etc.).

Case: "Adding the [Chats] section"

Incorrect	Create separate workplace and add the [Chats] section to it
Incorrect	Do not include section in any of the workplaces
Correct	Include the [Chats] section in the [Sales] and/or [Marketing] workplace

2. Accessing settings from the system designer

All custom sections and blocks used for setup and configuration should be available in the system designer. The

users can access the system designer by clicking the  button at the top right corner of the application window from anywhere in the system. The links to setup and configuration functions in the system designer are grouped into several blocks. The currently available blocks and their functions are listed in table.

Blocks of system designer settings

Settings block	Usage
Import and integration	Contains links to setup functions of different types of integrations, as well as access to Excel import. Add links to connector setup pages here.
System setup	Contains links to setup functions that affect general system behavior. Add links to custom application settings and vertical solution settings here.

Modify the system designer by adding links to your custom setup functions, such as:



- custom settings pages;
- custom setup sections;
- lookups used as setup medium.

3. Accessing settings from the section actions menu

The user experience is commonly formed as follows: user opens a section with new functions, starts studying it and only then understands that further work requires additional setup.

To help users navigate and perform necessary settings, add required action to the [*Actions*] button menu. Please note, that the section page (list of the records) and the page of specific record are the different pages. For each page, the set of actions can be different and the [*Actions*] button is configured separately for the list of records and the record page. The action added to the [*Actions*] menu of the section will not be added in the [*Actions*] menu of the record page.

Case: “Implement transition to the custom login authorization page in the X external system (login/password) that cooperates with the [*Integration*] section implemented in the application”

 Incorrect	Implement separate settings page accessible only by a direct URL address
 Correct	Add the “Configure the connection to the X external system” item in the [<i>Actions</i>] button menu of the [<i>Integration</i>] section

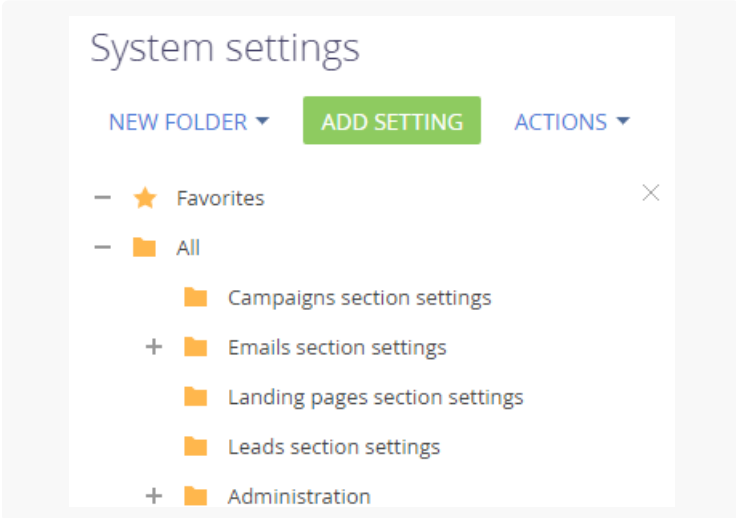
Settings are simple and easy to find

Many Marketplace applications require initial configuration by the user. To do this, the Creatio has a set of custom tools.

1. The [*System settings*] section

The [*System settings*] section is designed for additional setup of system sections. For example, set the color of overdue activities, parameters for sending emails, external service connection parameters, etc.

Case: “Add a new system setting”

<p>Use short and clear names for system settings. The name should reflect the essence of the setting and point on functions to which it belongs.</p>	<ul style="list-style-type: none"> ✘ Incorrect (the name is too long): “The currency that is used in the system by default”. ✔ Correct: “Base currency”. ✘ Incorrect (the name does not reflect the essence) “Order status”. ✔ Correct “Order status by default”. ✘ Incorrect (the name does not provide sufficient information about configured functions) “Automatic launch of the process” ✔ Correct “Automatic start of the incident management process”
<p>Group the settings.</p>	<p>User has no chance to find a specific setting in the general settings list without knowing its exact name. Grouping of the settings is a simple and effective way to simplify application configuration.</p>  <p>The screenshot shows a 'System settings' window with a 'NEW FOLDER' dropdown, an 'ADD SETTING' button, and an 'ACTIONS' dropdown. Below these are several folders: 'Favorites', 'All', 'Campaigns section settings', 'Emails section settings', 'Landing pages section settings', 'Leads section settings', and 'Administration'. Each folder has a minus sign to collapse and a plus sign to expand.</p>
<p>Add the description</p>	<p>The description of the system behavior that the system setting affects will answer most of user’s questions.</p>

2. The [*Lookups*] section

The [*Lookups*] section is used to manage all data (settings and system data) that can be displayed as a simple list.

The examples of using the [*Lookups*] section for different system tasks are given in the table. It is highly recommended to use the lookups cases listed in the table.

Lookups use cases

Task	An example of a lookup used
A list of data for user selection	[<i>Document statuses</i>] The lookup contains a list of all statuses that a document can have, for example, "Active", "Inactive" or "Draft".
Field autofill during integration	[<i>Lead channels</i>] Lookup contains the list of all resource types that provided new leads, for example, "Social accounts", "Search-based advertising" or "Email".
Configured business logic	[<i>Case notification rule</i>] Lookup contains a list of rules that manage sending notifications to contacts about their case progress (for example, "Send immediate", "Send after a delay", "Disabled").
System list	[<i>Webitel users</i>]
Configuring the displaying of functionality in different sections	[<i>Gantt chart configuration</i>]

3. Settings in the user profile

Individual settings of a specific user (individual login/password or preferences for using certain product functions) can be added in the user profile.

Examples:

1. Integration connection settings:

- Id of the phone operator/phone number by which the user works with a phone integration connector.
- Settings for connecting the user to the mail service.

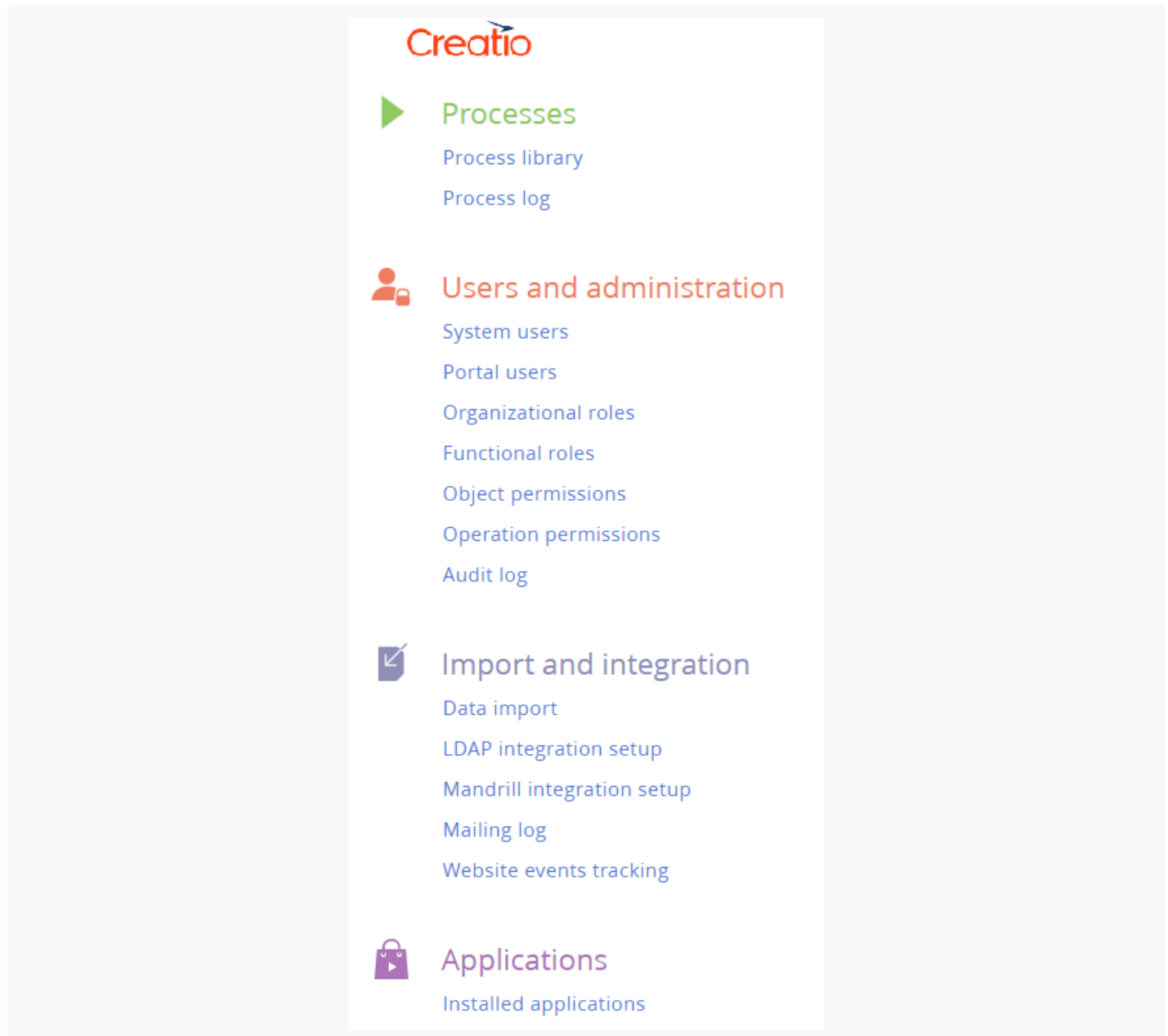
2. System behavior settings for the user:

- Notification settings.
- The default language settings that the user uses in the business card scanner.

4. Separate settings page

Putting all the settings on a separate page is the most convenient way for the user to manage the settings of the

application. However, separate settings pages are also harder to implement.



You can open settings page from corresponding section and/or system designer.

It is recommended to put all application settings on a separate page in the following cases:

- If the settings should be carried out in a specific sequence.
- If the setup requires both changes of settings and modification of lookups.
- If the setup requires changes of settings, lookups and additional actions.
- If a user will need to update the settings during the work with the application on a regular basis.

Development according to Creatio base logic

Correct filling of fields when creating leads

Creating leads when the Creatio is integrated with messengers, chats, landing pages or social networks is one of the typical integration tasks. When a lead is automatically created, it is necessary to keep correct logic of filling the fields of the lead page, including the information about lead generation channels.

When developing an application, make sure that Creatio users correlate each field name with its purpose and be able to use end-to-end analytics for the leads. An example of names of lead fields is given in the table.

Examples of lead page names

Name	Usage
[<i>Registration method</i>]	Lead registration method: <ul style="list-style-type: none"> • automatically created; • added manually; • incoming call/email; • landing page; • case.
[<i>Channel</i>]	Type of the resource from which the lead was received: <ul style="list-style-type: none"> • web; • social networks; • offline advertising; • event; • recommendation/personal contact.
[<i>Source</i>]	The name of the specific resource from which the lead was received (Twitter, Google, etc.).
[<i>Transition website</i>]	The site where the user clicked a link to the landing page, which resulted in creation of a new lead. The field is not editable and is filled in automatically when receiving a lead from the landing page.

UI minimum

Section icon

An icon should be added for each new section. The section icon requirements:

- *.PNG or *.SVG image
- 38 x 38px

- A flat white color image without small elements/lines, without background.

You can find suitable icons in the Marketplace template library ([Section icons for Creatio](#)). You can also use the free service to search and convert flat icons.

Integration with Creatio and public API

Creatio has a [wide range of integrations](#) with custom third-party applications.

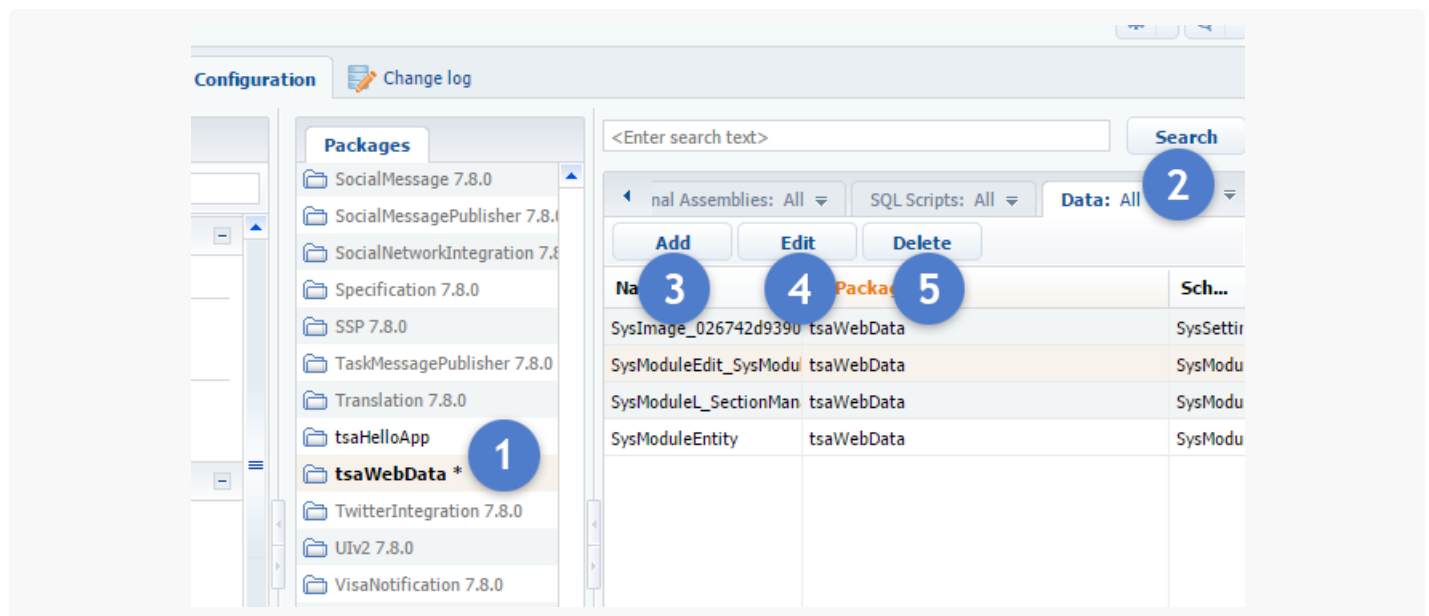
First, accessing Creatio by a third-party application requires [authentication](#). Starting with version 7.10, authentication is protected from CSRF attacks.

Wide range of integration capabilities is available through API provided by the [DataService](#) service. If the third-party system uses the [OData](#) protocol, it can also be used for integration with Creatio. Using the `<iframe>` HTML element for integration.

Binding data to package

Once the Section Wizard automatically attaches the data of a custom section to the custom package. However, for the section to show up in a workplace and with demo records, you need to attach additional data.

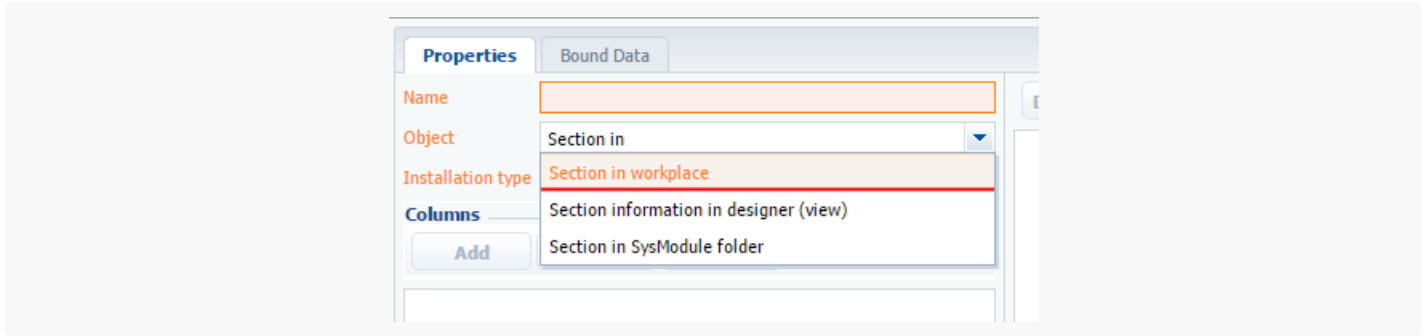
The data is attached in the [*Configuration*] section. To view the data attached to the package, select the necessary package (1) and select the [*Data*] tab (2). You can add (3), edit (4) and delete (5) the data bound to package.



The following data attachment examples are based on the [standard approach](#) to creating user sections.

Attaching sections to a workplace

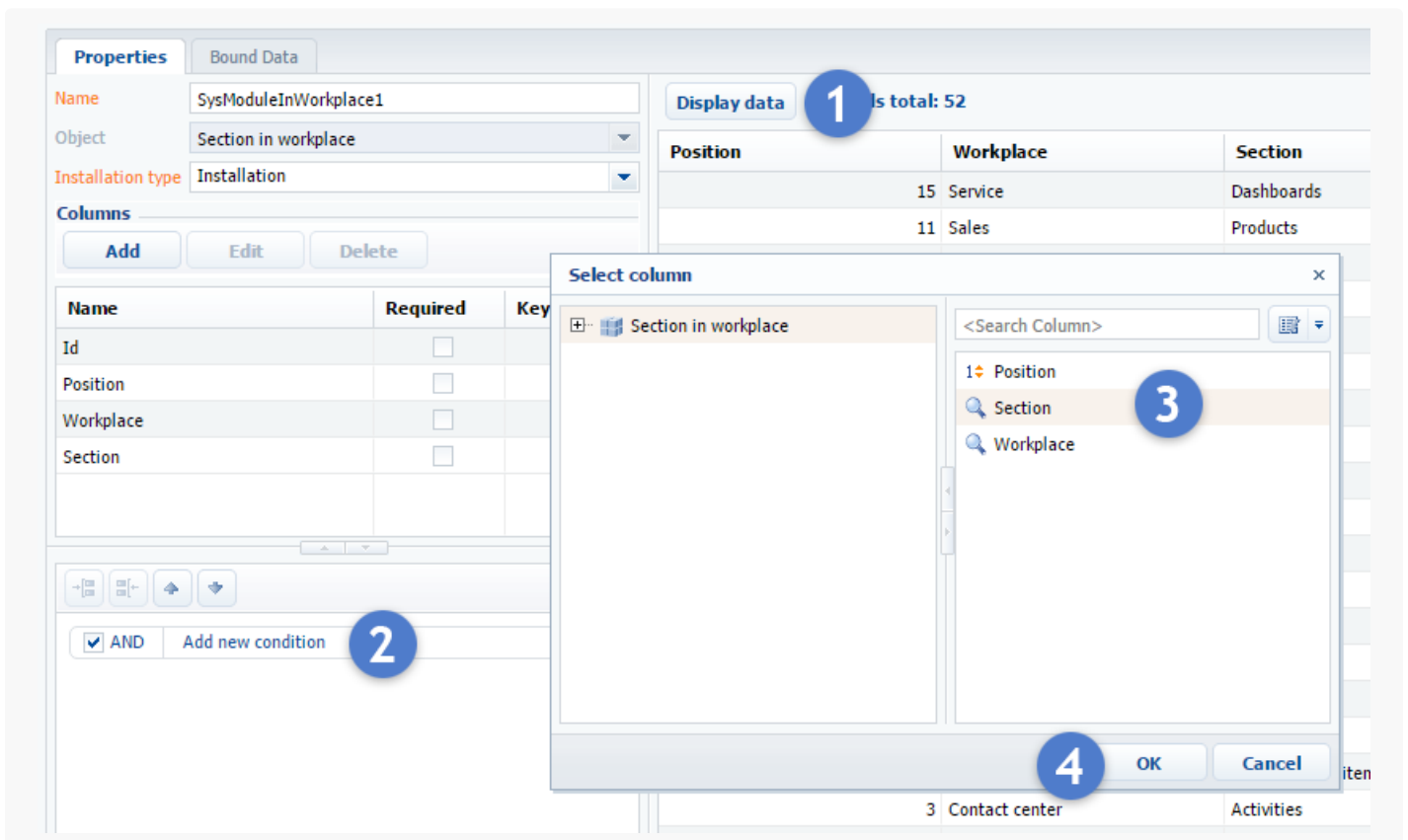
Bind the data of the [*Section in workplace*] for the section to show up in the [*Marketing*] workplace. Click [*Add*], and select the Section in the workplace value in the [*Object*] field of the package data attachment window.



Click [Yes] in the pop-up window.

Select the installation type. The universal option is "Installation". If you select "Installation", the data will be attached after the package installation and with subsequent updates.

Click [*Display data*] (1), to view information about all sections and workplaces currently in development.



Prior to attaching the data to the created section, you need apply a filter. In the filter area, add a new condition (2), select a column to filter by (3) and click [OK] (4).

Build the second part of the condition. Click on the condition sign (5), select the name of the section (6), and click [OK] (7).

Properties Bound Data

Name SysModuleInWorkplace1

Object Section in workplace

Installation type Installation

Display data Records total: 52

Position	Workplace	Section
15	Service	Dashboards

Columns

Add Edit Delete

Name	Required	Key
Id	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Position	<input type="checkbox"/>	<input type="checkbox"/>
Workplace	<input type="checkbox"/>	<input type="checkbox"/>
Section	<input type="checkbox"/>	<input type="checkbox"/>

AND Section that is equal to <?> Add new condition

Section - Google Chrome

tscore-dev-12:85/SaE_M_SeE_792ENU_Softkey_IStupak_1263706_0320/0/ViewPage.aspx?Id=33cc4a3

Title Web Search

Web data

Website events

OK Cancel

Save Cancel

The filter for selecting the created section is built as the result (1).

Properties Bound Data

Name SysModulIWorkplace

Object Section in workplace

Installation type Installation

Display data Records total: 0

Position	Workplace	Section
6	Marketing	Web Data

Columns

Add Edit Delete

Name	Required	Key
Id	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Position	<input type="checkbox"/>	<input type="checkbox"/>
Workplace	<input type="checkbox"/>	<input type="checkbox"/>
Section	<input type="checkbox"/>	<input type="checkbox"/>

AND Section that is equal to Web data Add new condition

OK Cancel

Save Cancel

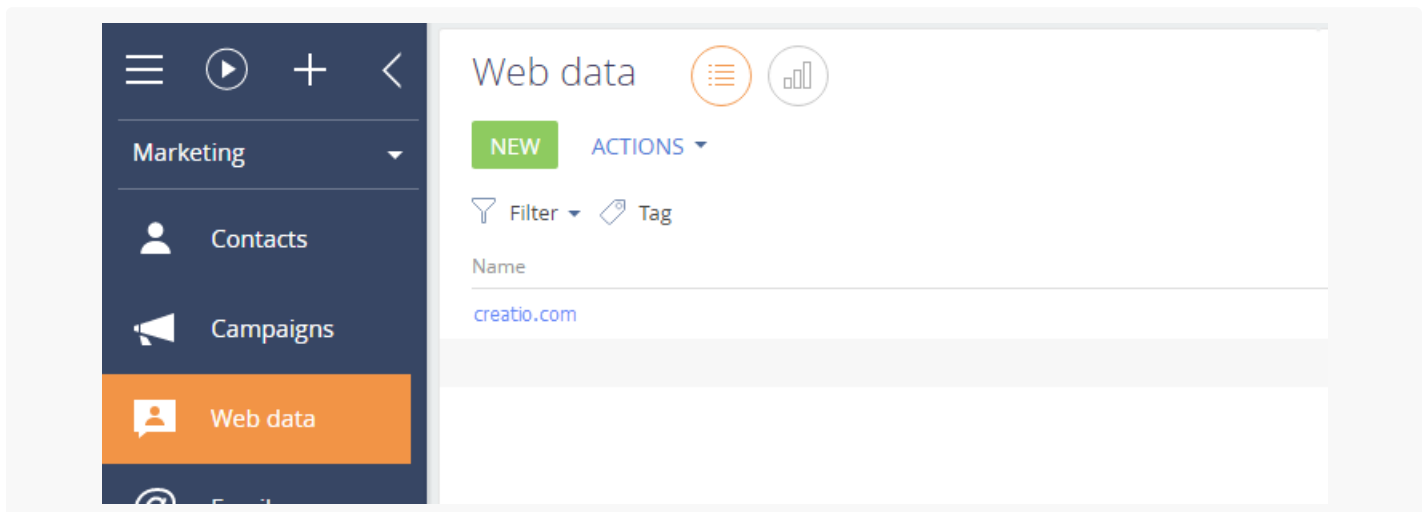
A new entry will show up in the [Configuration] section of the [Data] tab after saving the attached data (2).

Name	Package	Schema
SysModuleInWorkplace	tsaWebData	SysModuleInWorkplace
SysImage_026742d9390c	tsaWebData	SysSettings
SysModuleEdit_SysModule	tsaWebData	SysModuleActionLczOld
SysModuleL_SectionManager	tsaWebData	SysModuleLczOld
SysModuleIWorkplace	tsaWebData	SysModuleInWorkplace
SysModuleEntity	tsaWebData	SysModuleEditLczOld

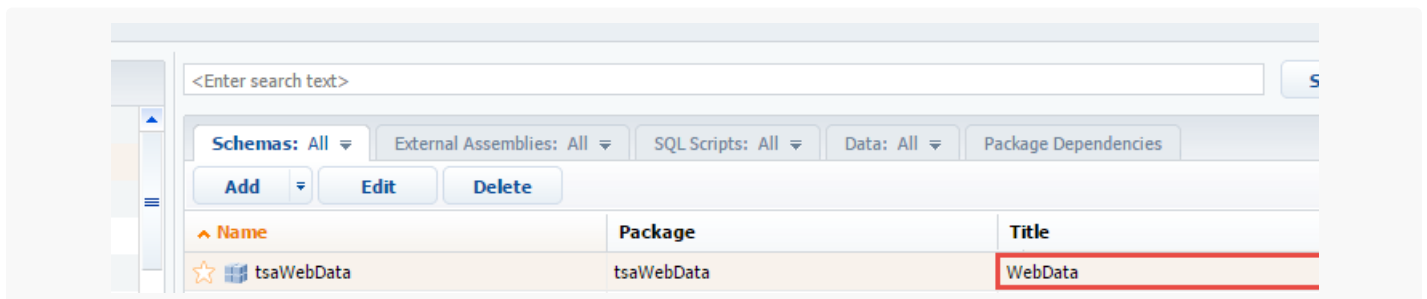
Section data attachment

To attach the demo content of the section, you must first determine the title of the main section object schema.

Section demo content

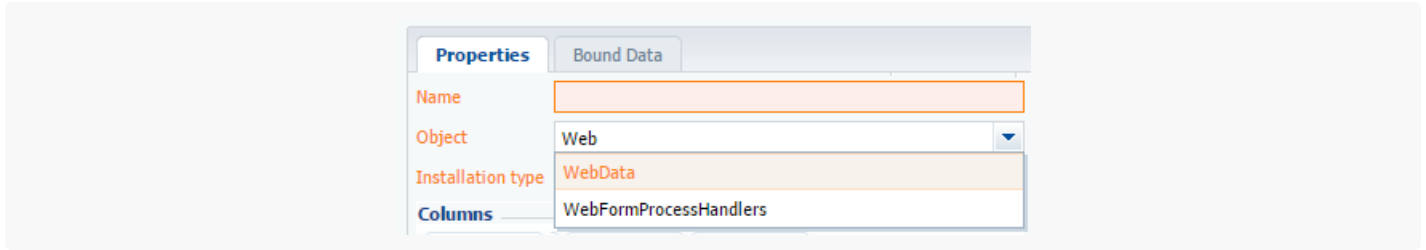


Section object schema header



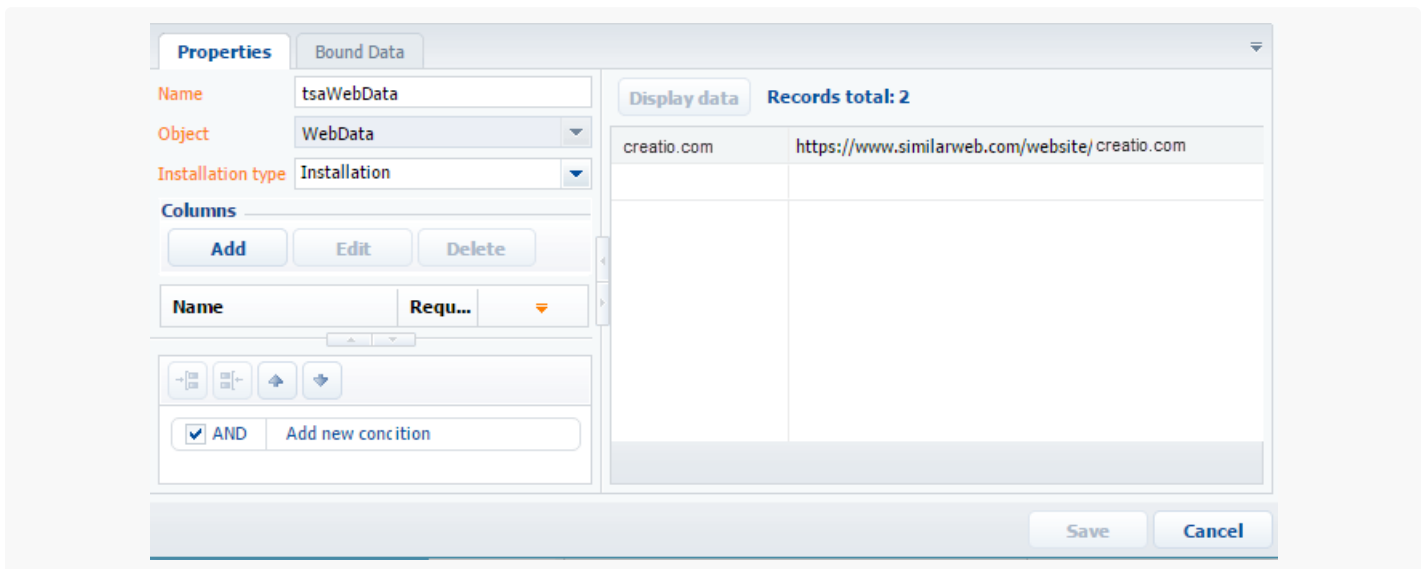
This schema is created automatically by the Section Wizard and the title of the schema should match the section header.

Click [Add], and select the Section in the workplace value in the [Object] field of the package data attachment window.



Click [Yes] in the pop-up window.

Select the installation type "Installation". Clicking the [*Display data*] button to view the attached data. Use filter to select specific records.



Click the [Save] button to save and close the data attachment window.

Marketplace application licensing

Paid Marketplace applications must be licensed to monitor their use. The main license types are:

- **Named licenses** - grant application access for specific users. These licenses are linked to user profiles and can not be utilized by other users. The system administrator can [redistribute registered licenses](#) at any time.
- **Server licenses** - do not provide for the need to license individual users. All Creatio users with the appropriate access rights will be able to access the licensed functionality.

Regardless of license type, it may include:

- **Licensing objects** - the names of custom objects added to the Marketplace application. This list can include any custom objects, such as a section, detail or lookup objects.
- **Licensing operations** - the names of operations added to the application logic to see whether a license is available for the use of certain functionality. For example, an additional action that requires the connection to the licensing operation has been added to one of the base system sections. When this action is triggered in the application, the license must be checked first, and based on that the functionality should either be continued or interrupted.

A Marketplace developer may learn about the amount of purchased and distributed user licenses by contacting Marketplace support (marketplace@creatio.com).

Licensing methods

Possible methods of licensing depend on the license type and the subject of licensing.

Possible licensing methods

License type	The subject of licensing	Business cases
Nominal	Objects	The Marketplace application is a new Creatio section. The developer would like to receive a fee for every user working in their section.
Nominal	Operations	The Marketplace application is a connector to a third-party service. The developer would like to receive a fee for every user with the access to the service. For example, a telephony connector with a single user subscription model.
Server-based	Objects	The Marketplace application is a new Creatio section. The developer would like to receive a fixed fee, regardless of how many users work with the section.
Server-based	Operations	The Marketplace application is a connector to a third-party service. The developer would like to receive a fixed fee, regardless of how many users work with the section. For example, a web-chat connector, used to register leads / cases in Creatio. The client purchases license regardless of the quantity of users.
Nominal	Objects and / or operations	The Marketplace application is a section with records that are created based on the data from third-party connectors to external services. The developer would like to receive a fee for every user with the access to the service. Additionally, the developer would like to receive a fee for every connector to the service. For example, the [Cases] section is modified in the vertical solution framework. It integrates with the external system used to register cases. The client pays for access to the section functionality for each user.

Attention. It is prohibited to use both nominal and server-based licenses that encompass the same objects and operations.

Validity period of a license

Every license includes the period of its validity. The Creatio platform monitors the validity of licenses. If the date has expired, the work of the licensed object or operation will be blocked.

Depending on the validity period, the licenses can be of two types:

- **Annual** - designed for Marketplace applications distributed with an annual subscription. The license is valid for one year. Upon expiration, the client must renew the annual subscription and get a new license.
- **Unlimited** - designed for Marketplace applications with a single payment method. The license is unlimited.

The process of issuing a license for a new product

1. When creating a Marketplace application, the developer must decide:

- What will be included in the license - licensing objects or operations (both options are possible). They must also create a list of licensed objects and/or operations.
- [Determine the type of license](#) - nominal or server-based.
- Determine the license validity period - the license is annual or unlimited.

Note. If the license includes licensing operations, it is necessary to implement a license validation in the program code of the application. The verification logic for object licensing is already implemented in the base version of Creatio.

Attention. The names of the licensed objects must contain the prefix specified in the [developer profile settings](#). Developer prefix - unique ID added to the names of custom schemas, packages and custom columns in the objects inherited from the base objects. The prefix identifies the functions added or modified by the developer. The prefix must be at least 3 characters long. The developer prefix cannot be changed after the registration.

2. Upon successful verification of the application by Marketplace employees, the developer must provide a list of licensed objects and operations, as well as the information about the type and duration of the license to the Marketplace support team (marketplace@creatio.com).
3. Based on the provided list, the Marketplace support service generates licenses for this application.
4. In the future, after ordering and paying for the Marketplace application, the user will have the necessary licenses installed by the Marketplace support service.

Implement the license checking mechanism in a product

Checking licensing objects

If the license application includes only licensing objects, then no improvements in the Marketplace application are required. The license for licensing objects is checked by the base Creatio tools.

Checking licensing operations

If the application license includes a licensing operation, provide a license verification mechanism in the source code of the application to perform this operation.

Attention. We do not recommend verifying the existence of an operation license directly in custom schemas since the source code of a custom schema configuration packages is read/replace-only. Therefore, we recommend using licensing objects.

Application demo version

If the Creatio application does not have any downloaded licenses for any of the products, the application's demo mode is enabled by default. In demo mode, all Creatio functionality is available to the user, but not more than 1000 records may be added to each database table.

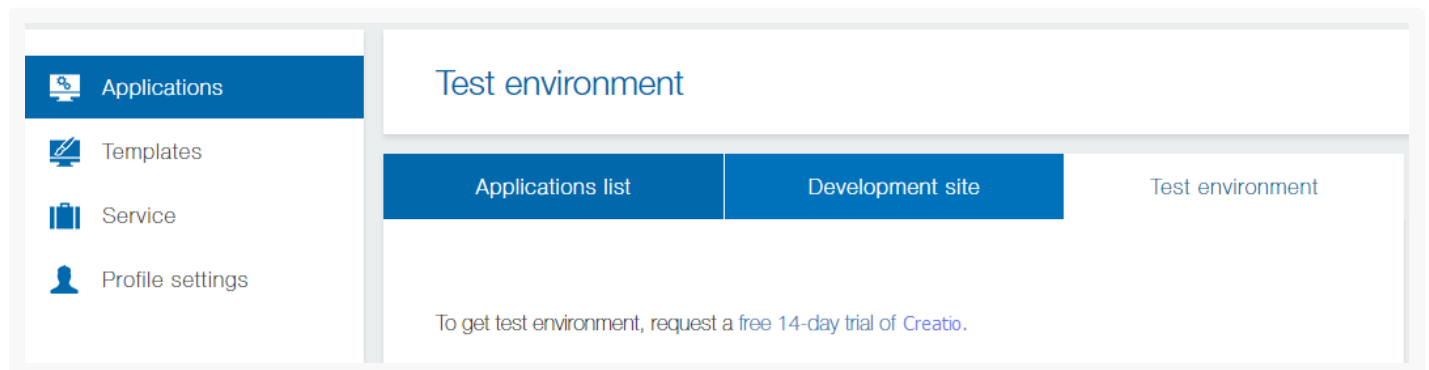
Migrating apps to the testing environment

Development of complex functionality requires proper [organization of the development processes](#).


We recommend using the free 14-day trial of Creatio to test the newly developed functions.

Test site registration

You can request a Creatio free 14-day trial for testing purposes directly from the Developer workspace. Go to the [*Applications*] section and select the [*Test Environment*] tab and use the link that will take you directly to the trial request form.



Click the link, fill out the form, and click [*Get Started*].



14-DAY FREE TRIAL

Experience unlimited functionality

Try the 14-day free trial for an unlimited number of users and start using bpm'online completely free of charge. Upon completion of the 14-day free trial period, users can prolong their subscription and continue working with their data and settings.

Free 14-day trial

First and Last Name*

Create a password*

Company*

Job Role*

Email*

Country*

City*

Code* Phone*

Fill in the system with demo data

[Start my free trial](#)

[Log in if you have already registered an account](#)

Receive Newsletter

The trial is available immediately. The [*Test Environment*] will display the activation and deactivation dates.

Note. You will receive a notification of the readiness to use the test application on the email address specified at registration.

Applications list	Development site	Test environment
<p>Trial site: 017153-crm-bundle</p> <p>Activation date: Wed, 09/27/2017 - 15:10</p> <p>Deactivation date: Wed, 10/11/2017 - 15:10</p>		

Transferring solutions to the testing environment

Before you publish your product in the Creatio Marketplace, you need to make sure that it works on the test site. To set up the integration:

1. [Export packages](#) with the developed functionality from the development environment.
2. Create a `*.ZIP` archive from the exported `*.GZ` package archives. If you exported a single package, there is no need to add it to a `*.ZIP` archive.
3. [Import](#) the `*.ZIP` archive (or the `*.GZ` file, if you exported a single package) to the test Creatio application.

After the application has been successfully installed in the testing environment, please check your developed functions and make sure they work as intended. If any errors occur, please make necessary changes in your development environment, then repeat the export/install process again.