BSG connector for Creatio

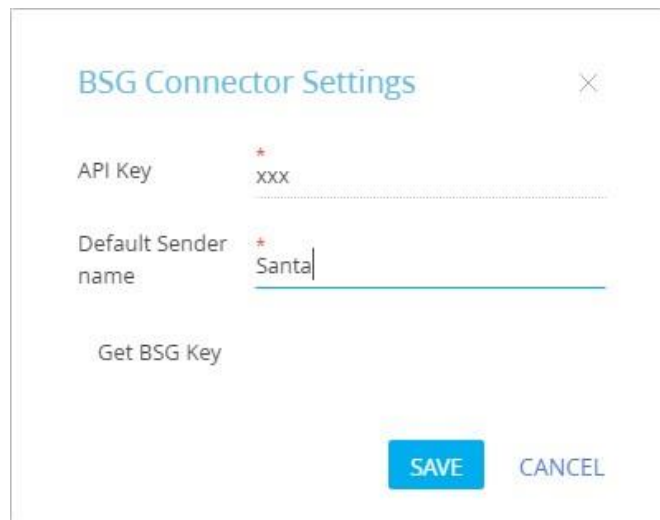User manual

1 CONTENTS

2 CONFIGURATION

You need to configure BSG Connector before usage: specify API key and default sender name.

To reach configuration page for BSG Connector please go to "System Designer":
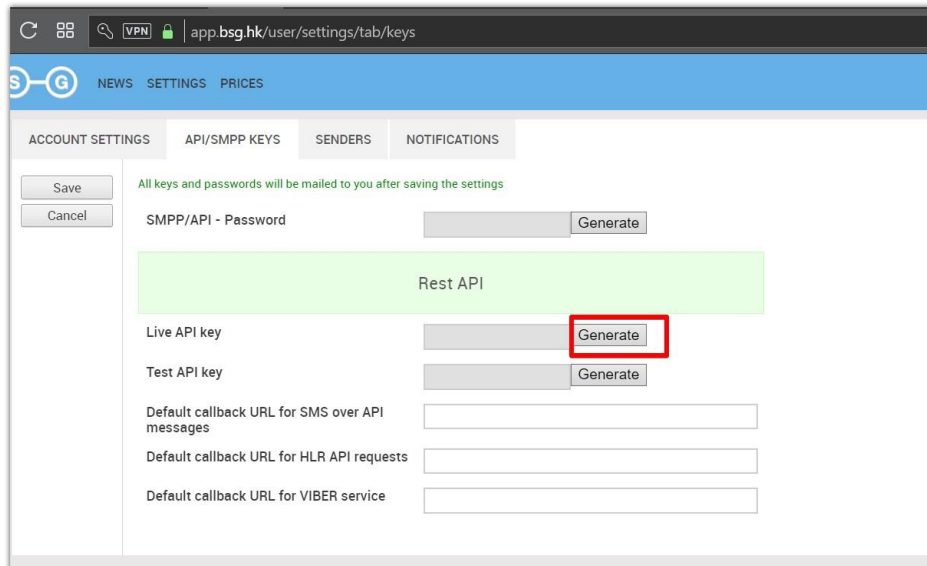
Then in 'Import and integration' section click on 'BSG Connector Settings' link:

The modal dialog with BSG Connector settings:

If you don't have API key for BSG service – please click on link 'Get BSG Key' and authenticate in BSG panel:

Press 'Generate' button and copy received API key:

live_10mLRXzXXXXXXXXXX – for production

or

test_10mLRXzXXXXXXXXXX – for testing

Put received key into 'API Key' input field:
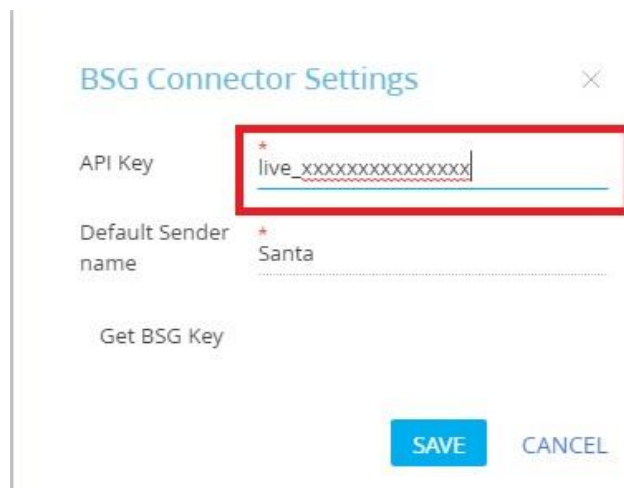
and press 'Save' button.

On success you'll see the confirmation window.

There is also the direct link to connector's configuration page right from main form, but only if current user is administrator:

BSG,2019

If connector was configured before – there will be a link to disconnect it from BSG Service instead:

3 USAGE

Our BSG Connector is designed for commons use cases and has key features:

- single SMS delivery,
- mass delivery,
- message templates,
- template tags
- balance check
- count SMS messages

If your business require more specific features or you need advanced customization – feel fee to contact our support team support@bsg.world

3.1 SINGLE SMS DELIVERY

Main form for SMS sending is on 'BSG' tab, top section of selected contact page:

Main form in details:

Control elements:

1. Selected contact and the phone that SMS message will be send to
2. Dropdown with allowed tags
3. Message text
4. Validation messages
5. Symbol counter with split for actual SMS messages
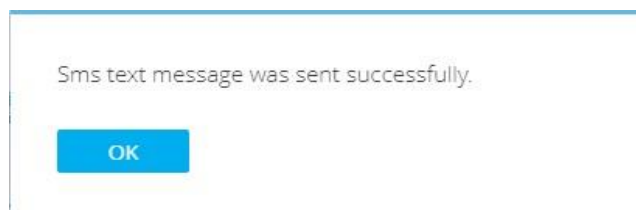6. Select pre-defined message template
7. Check balance in BSG Service (if connected)
8. Link/Unlink BSG Account from connector
9. Send SMS action button
10. Encode message into translit

On successful delivery you will see notification window:

3.2 MASS SMS DELIVERY

Our BSG connector allows mass delivery of SMS messages to multiple selected contacts (or to all). To do that, you need to go to 'Contacts' section and in 'Actions' menu selects 'Select multiple records' or 'Select all':

BSG,2019
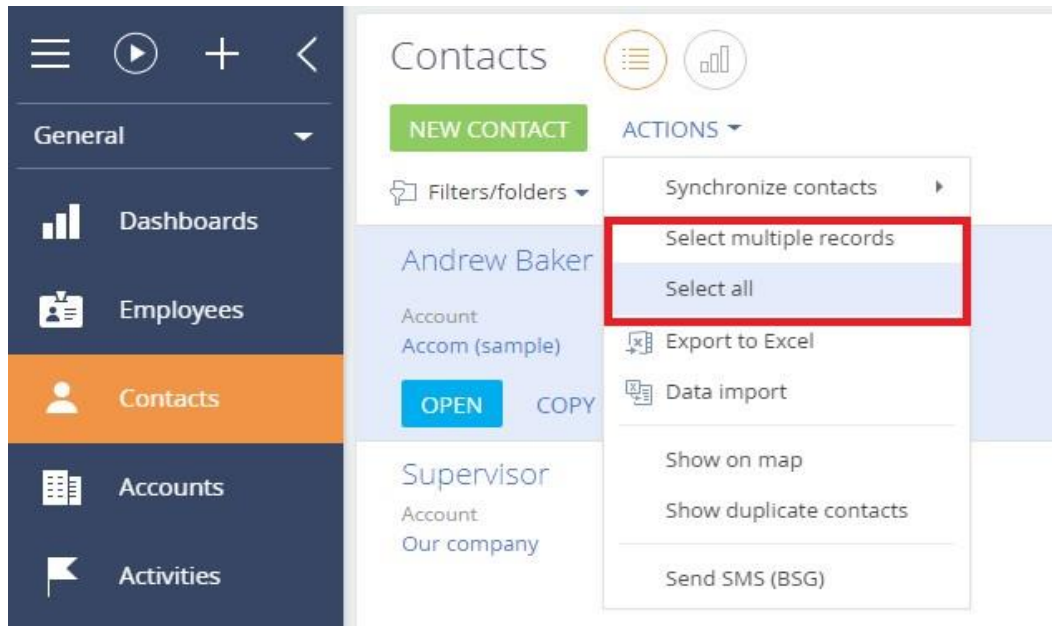
Next, after finish selecting of required contacts, you'll need to select 'Send SMS (BSG) ' option:
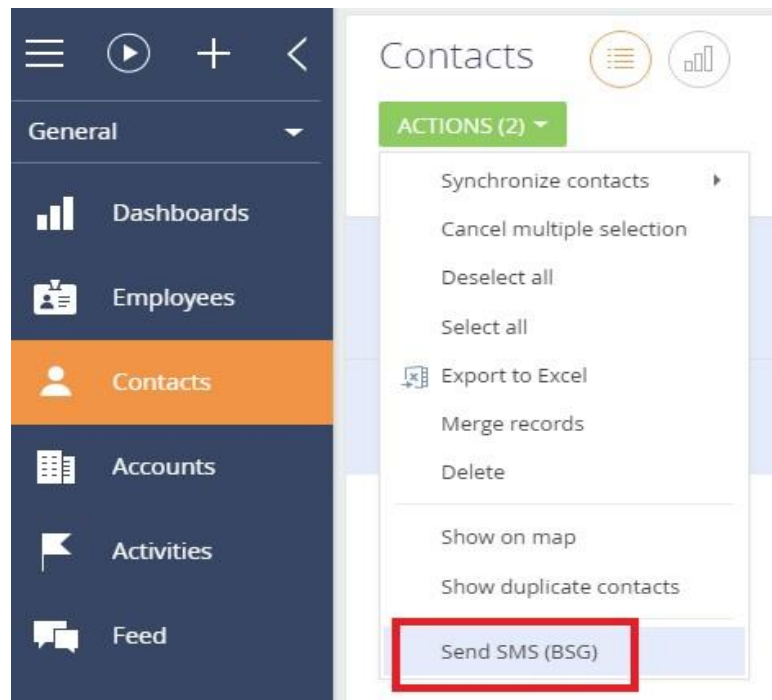
There will be a modal dialog with details for mass SMS delivery:

Control elements:

1. Number of selected contacts
2. SMS message text, supports tags input
3. Selection of pre-defined message template
4. Symbol counter with split for actual SMS messages
5. Action button to start mass delivery
6. Encode message to latin
7. Select tags for SMS message substitutions (see below)

3.3 CHECK BALANCE

This option will be available only after successful linking with BSG Account. To see your balance in BSG, click on 'Balance' link in main connector form:

There will be modal window with balance details:

3.4 SMS MESSAGE TEMPLATES

Our connector allows to create, use and manage SMS message templates. To use this feature please click on 'Template' link from main connector form (or from mass delivery form):

Main screen for SMS message templates:

Control elements:

1. Select current template from list and back to delivery form
2. Add new template
3. Template search

On template select, there are common dictionary actions allowed:
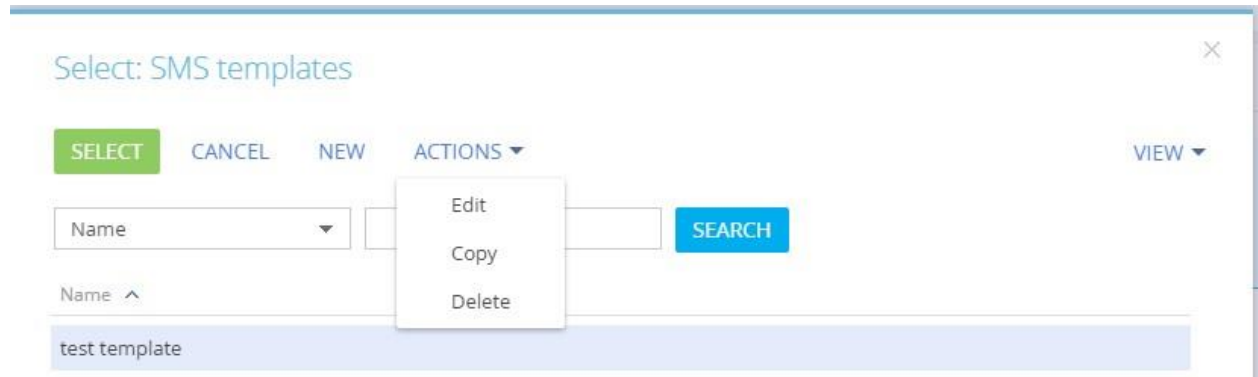
And default control of current view:
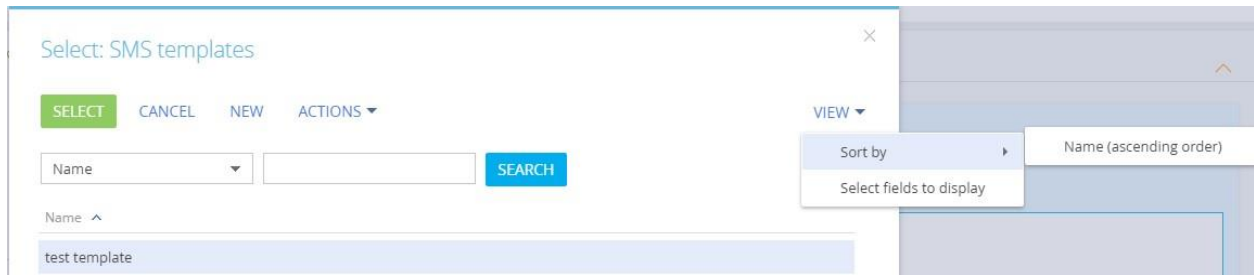
Filter options for template search:
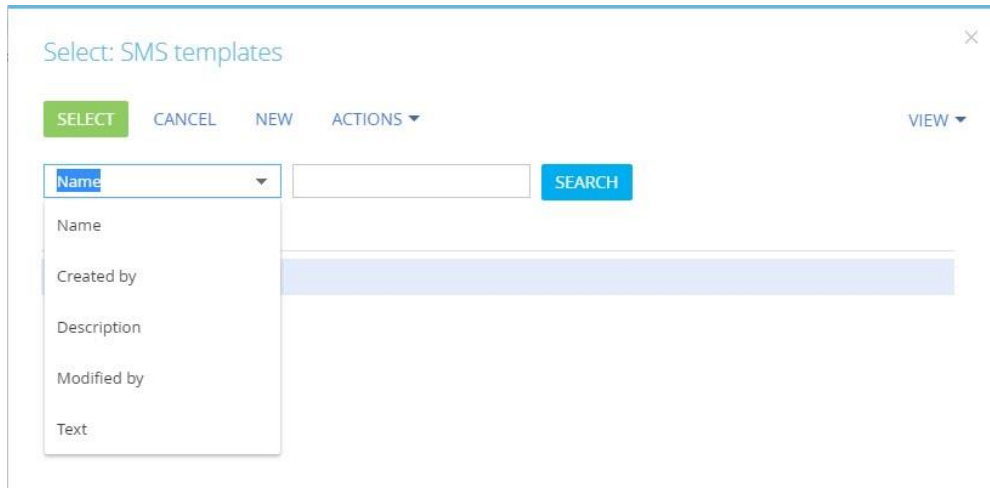
To create the new template, press 'New' button, there will be modal window:
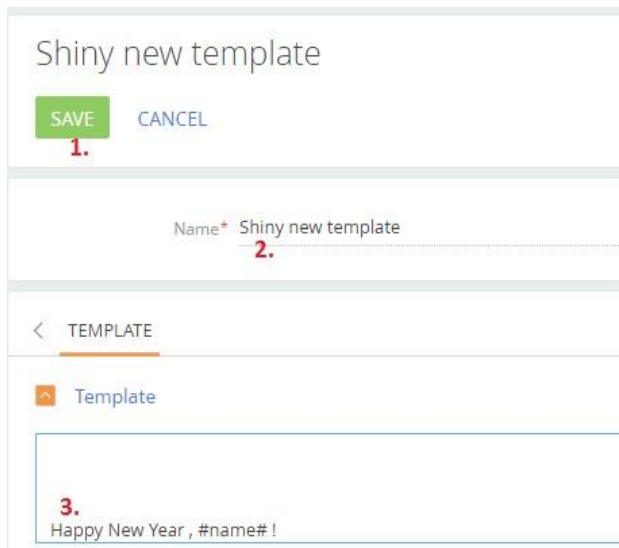
Control elements:

1. Action button to Save template

2. Template name
3. SMS message text

SMS message also allows tags input.

3.5 TEMPLATE TAGS

Our connector supports tags in SMS messages. To use them you need to put special keyword into message text and it will be automatically replaced to actual contact details.

Below is the list of supported tags:

Tag	Description
#name#	Full name of a contact
#GivenName#	First name
#MiddleName#	Middle name
#Surname#	Surname
#JobTitle#	Job Title
#BirthDate#	Date of birth
#MobilePhone#	Mobile phone
#Phone#	Work phone
#HomePhone#	Home phone
#Skype#	Skype
#Email#	Email
#Address#	Address
#Zip#	Zip

All tags are case insensitive, so all of them will work: #name# , #NAME# , #Name#

3.6 OVERRIDE SENDER name

Our connector allows to override default sender name for a value specified in associated Account.

There is specific field 'Sender name (BSG)' that our connector adds on install to Account entity, if this field is filled – the value will be used for all contacts associated with this Account entity.

To use this feature, go to Account page:

And fill 'Sender Name (BSG)' field, then press 'Save' button.

3.7 USAGE IN Creatio PROCESS

Our connector allows to use BSG service delivery as step in Creatio process. Below is step-by-step sample of how to add such step into new Creatio process.

Go to 'Process library'.

Press 'New process' button:
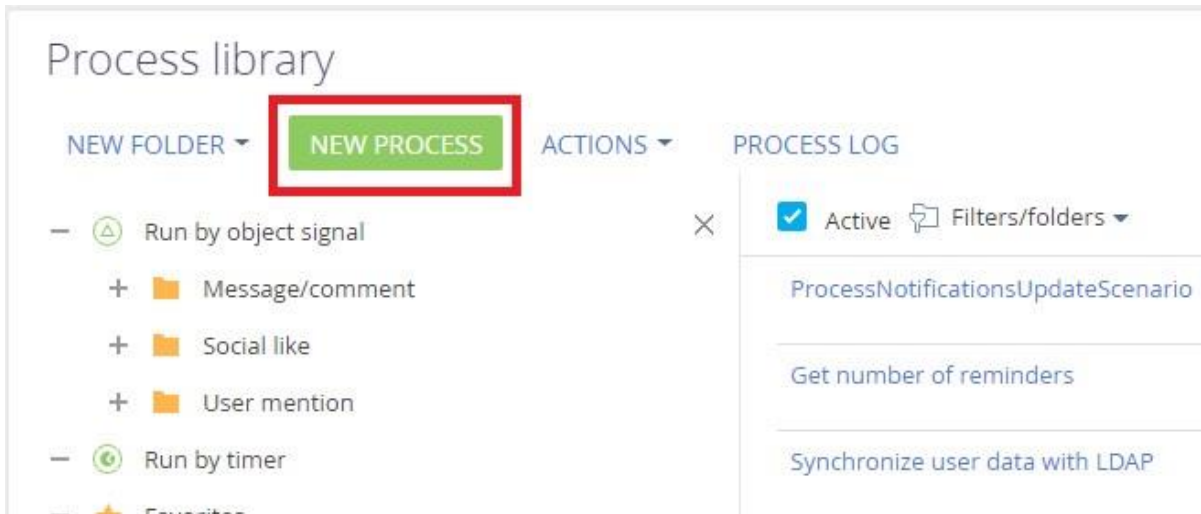
From the list on left (titled 'User actions') select 'Send SMS' option and drag it to Creatio diagram:
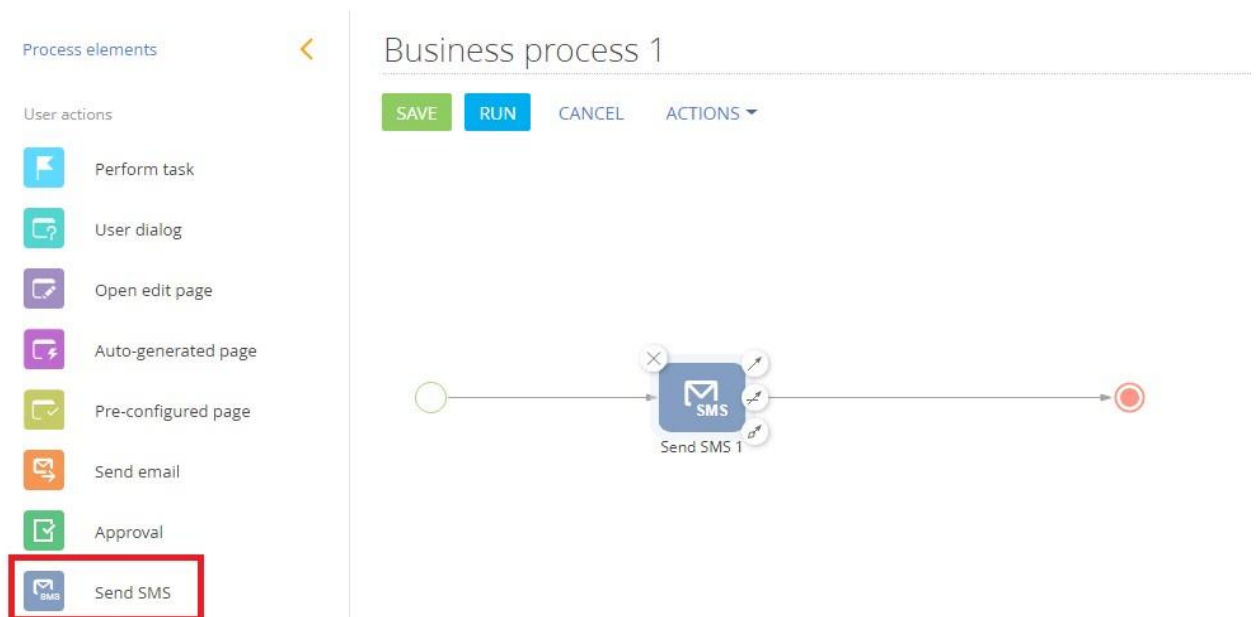
The line from start of process should be connected to our step, same as line from our step to 'end of process' point.

Now you'll need to fill the step parameters from (on right).

These parameters are typically filled from previous steps of Creatio process, so delivery step will take them from environment of current process.

To check that we created process correctly, press 'Run' button:

3.8 USING CONNECTOR FROM **Creatio** SCRIPTS

Our connector allows to been used from so-called 'Script Task' – the special step in Creatio process flow with custom script inside.

Below is step-by-step sample of how to use that for mass SMS delivery.

1. Navigate to 'Process library' section and create new Creatio process, add 'Read data' and 'Script Task' steps, as on screen below:

2. Click on 'Read data' step and configure parameters like on screen below:

Connector expects the collection of Ids from 'Contact' entity only, so all other fetched fields will not be used.

3. Click on 'Script Task' step and paste the code below:

```
var log = global::Common.Logging.LogManager.GetLogger("BSGConnector");

var userConnection = this.Get<UserConnection>("UserConnection");

log.Info("test process has been started " );

try {
```

```
var                                    colSource                                    =
Get<ICompositeObjectList<ICompositeObject>>("ReadDataUserTask1.ResultCompositeObjec
tList");

var listIds = new List<Guid>();

foreach (var parameter in colSource)
                              {
                                      Guid value;
                                      parameter.TryGetValue("Id",out value);
                      listIds.Add(value);
                                      log.Info(string.Format("parameter {0}",value));
                              }

log.Info(string.Format("total number of records: {0}",listIds.Count));

var p = new Terrasoft.Configuration.BSGConnector.BSGUserTaskParameters(); p.ContactIds
= listIds;
p.SmsMessage = "test message for mass delivery check, for #name# !";
p.SenderName = "Santa-Claus";

var  smsTask  =  new  Terrasoft.Configuration.BSGConnector.BSGUserTask(userConnection);
smsTask.SendSms(p);

} catch(Exception e) { log.Info("process error: "
        +e.Message);
}
return true;
```

Should look like on screen below:

```
 2
 3    var userConnection = this.Get<UserConnection>("UserConnection");
 4
 5    log.Info("test process has been started " );
 6
 7▸   try {
 8
 9        var colSource = Get<ICompositeObjectList<ICompositeObject>>("ReadDataUser
10
11        var listIds = new List<Guid>();
12
13        foreach (var parameter in colSource)
14▸            {
15                    Guid value;
16                    parameter.TryGetValue("Id",out value);
17                    listIds.Add(value);
18                    log.Info(string.Format("parameter {0}",value));
19            }
20
21    log.Info(string.Format("total number of records: {0}",listIds.Count));
22
23    var p = new Terrasoft.Configuration.BSGConnector.BSGUserTaskParameters();
24    p.ContactIds = listIds;
25    p.SmsMessage = "test message for mass delivery check, for #name# !";
26    p.SenderName = "Santa-Claus";
27
28    var smsTask = new Terrasoft.Configuration.BSGConnector.BSGUserTask(userCor
29    smsTask.SendSms(p);
30
31▸   } catch(Exception e) {
32    log.Info("process error: " +e.Message);
33    }
34    return true;
35    |
```
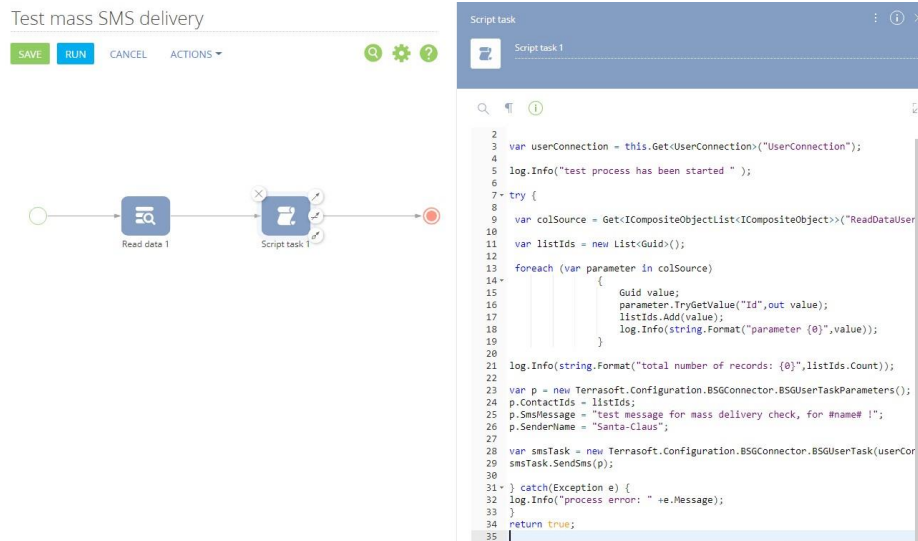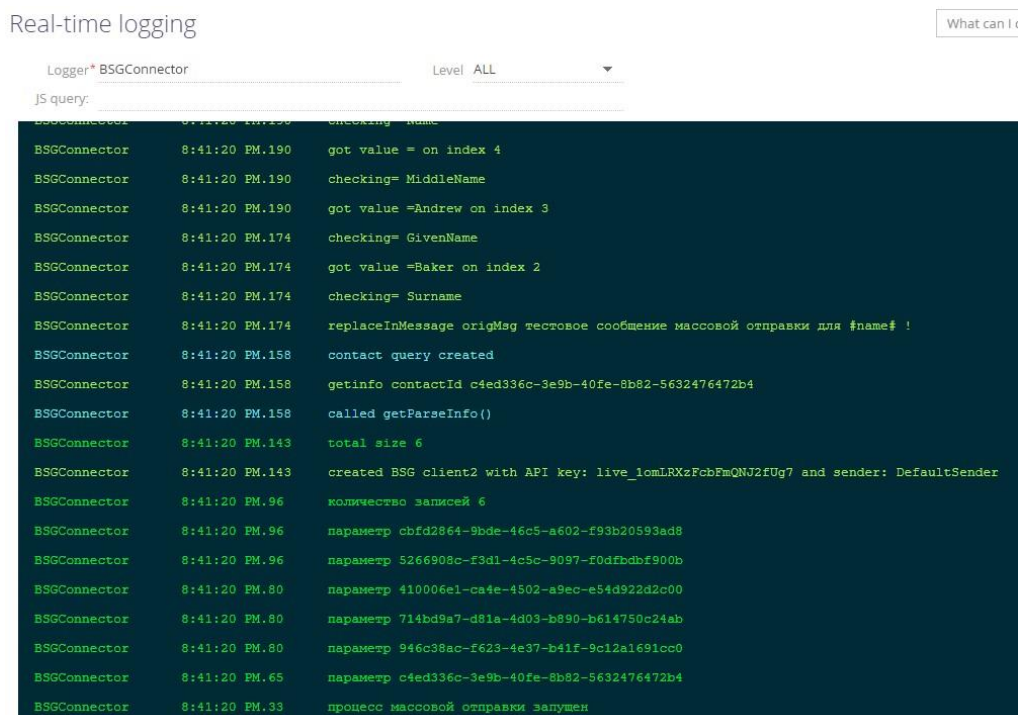
This test process will write log records to default log file (Common.log), also these records could be viewed in runtime with 'Telemetry extension' (see below):

4 TROUBLESHOOTING

Our connector supports multiple levels of logging, so to investigate source of problem you can adjust connector's log level to see more details of its internal work.

Logger name: BSGConnector

Detailed guide of how-to setup logging in Creatio offsite instance could be found in Creatio documentation.

For cloud instances, you can try this extension «Telemetry log for Creatio»:

https://marketplace.creatio.com/app/telemetry-log-creatio that will

allow to observe connector logs in real-time:

For any questions related to connector please contact our tech support: support@bsg.world