Release Notes Contact Version 2.8

Moscow

September, 14,2021

Together to the next level

Agenda

- Tech Stack Change
- Business features demo
- Technical features demo
- Global Demo Environment
- Document Collection about Contact

Tech Stack Change

- Upgrade Angular (web framework)version to 8.2.14 for website and Business Administration Console
- End of support for MS Internet explorer web browser

Just to be sure you know

Third-party licenses

- Aspose Java Total—document generation libraries (Developer licence with unlimited installation bought by CRIF) https://products.aspose.com/total/java/
- Ag-grid enterprise—UI component for GRIDs (Deployment License must be provided by CRIF) https://www.ag-grid.com/license-pricing.php

Browsers support policy

- Contact collection solution supports only desktop browsers Edge(Chromium-based), Chrome, Firefox, with version not older than 1 years

Strategy One support

- Contact collection solution currently support engine version 6.4.0

Business features demo

Together to the next level

New business feature

- Added the ability to customize the top block around the debtor's full name in the debtor card, including:
  - changing the background color
  - entering additional information
  - adding a colored marker
- Custom API processing is supported (url, urlParams, etc. are specified in the gui.json plugins in the config)

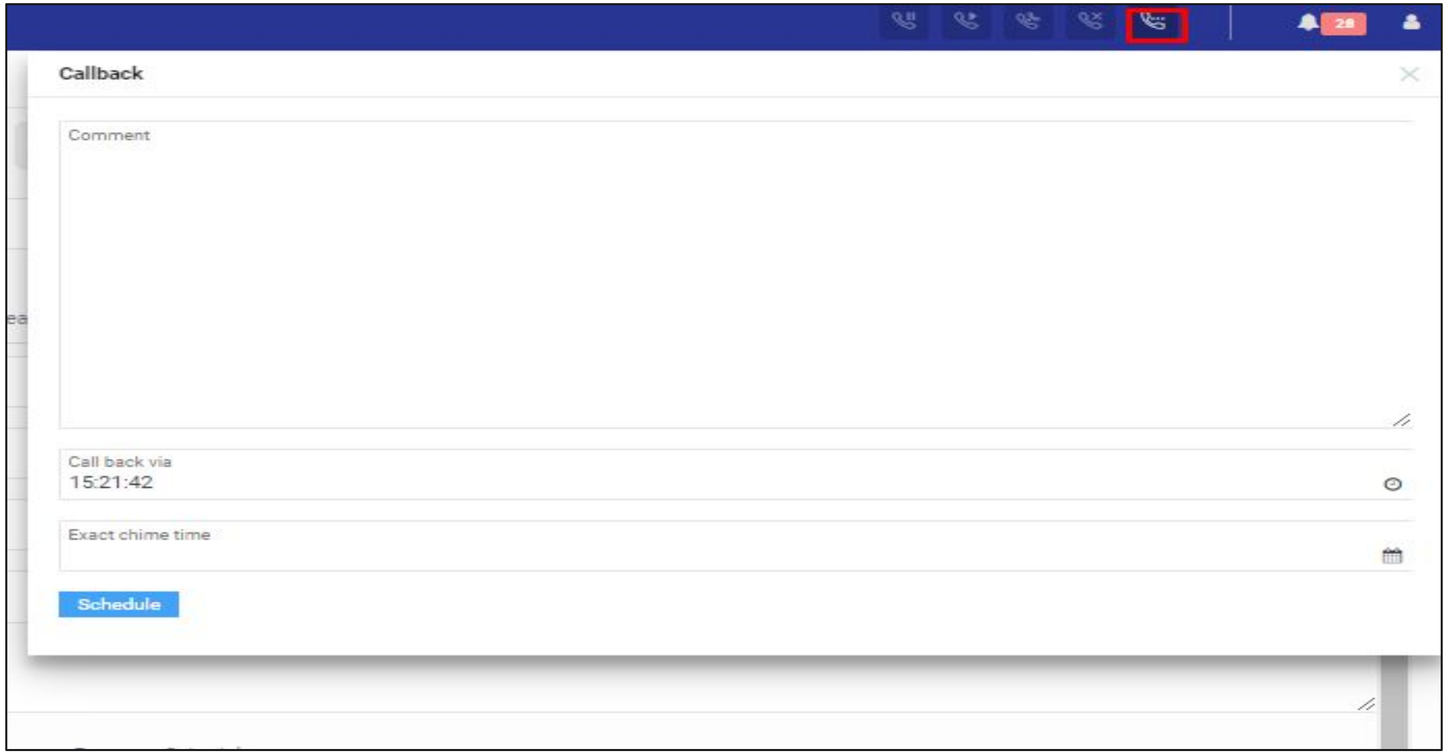
Data

```
{
  backgroundColor: "black",
  textColor: "white",
  indicatorColor: "red",
  additionalText: "debtor 4"
}
```

Interface

New business feature

- Several call scenarios have been implemented to call a client back when registering a contact:
  - Call back now
  - Call back at the scheduled time
  - Call back in a specified amount of time



New business feature

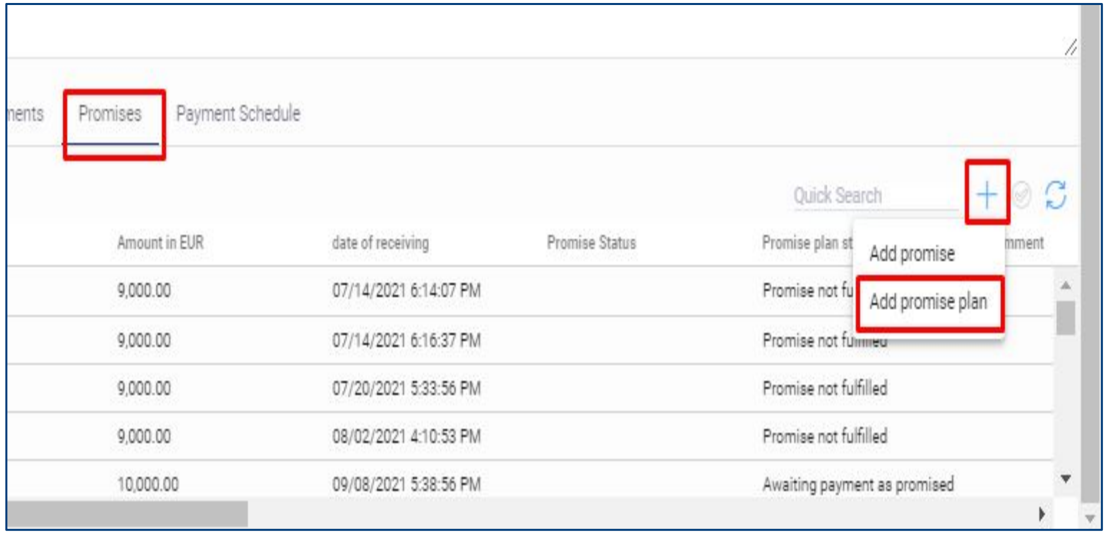
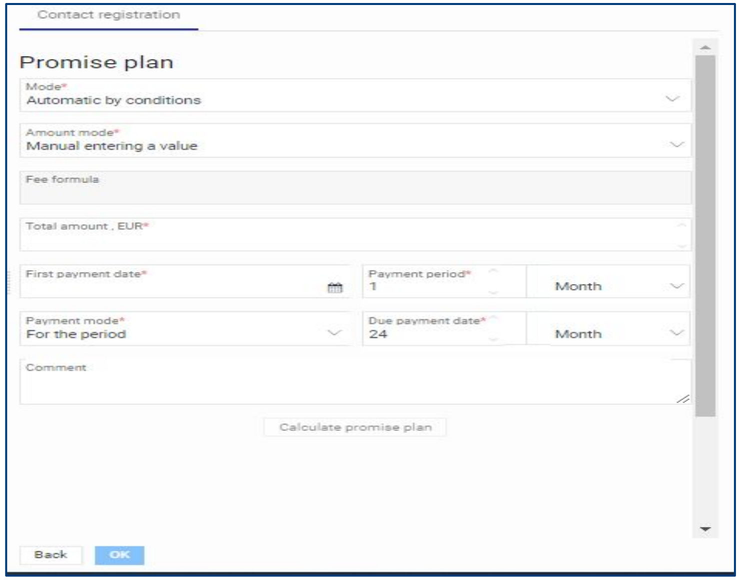
- When sending SMS, the fields "SMS Sender" and "SMS date" are not available for display / editing

  The list of system accesses has been expanded to include the following:
  - SMS_SENDER_EDIT*—enables to change the sender when generating SMS
  - SMS_SEND_DATETIME_EDIT—enables to change the date and time of sending SMS

*works, if value of constant SMS.Sender.Use is "True"

Business feature

❑ Option to add Promise Plan

From Debtor card	When registering a contact
	

Together to the next level

New business feature

- Enhanced the conversation scripts view window, which appears when registering a contact. Added:
  - vertical scrolling
  - horizontal scrolling
  - an optional white background
- Configured in root.json

New business feature

- The ability to cover more than one promise with one payment has been implemented for promise plans.

  Previously one payment could cover only one promise.

  Now it is possible to turn on covering several promises with one payment using

  new technical constant Payment.CoverSeveralPromise.Use

Together to the next level

New business feature

- The ability deleting personal data associated with a person and a debt

  When setting up these events, additional entities and attributes can be selected.

  After completing events, a check is performed for the presence of records in the audit tables, followed by their subsequent deletion.

Technical features demo

Together to the next level

Customize debtor card top block

- ❑ Added the ability to customize the top block around the debtor's full name in the debtor card, including:

1. Changing the background color
2. Entering additional information
3. Adding a colored marker

- ❑ Configuration

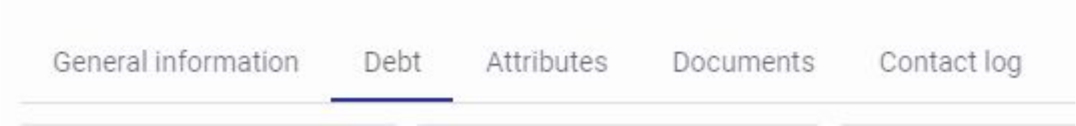

```
"plugins": [
  {
    "type": "personFullName",
    "paramKey": "debtorId",
    "loadingAdditionalParameters": {
      "url": "/testAdditionalParamsUrl/debt{id}",
      "urlMethod": "GET",
      "urlParams": {
        "id": ">> param(\"debtId\")"
      }
    }
  }
],
```

The work of this functionality is based on a separate request, which must be described in the ***gui.json*** file.

The structure of the request and a full description can be found at the link [Documentation]

Default tab in configurable cards

Implemented the ability to specify a tab that should open by default when you first open it.



Configuration

To set the default tab, you must specify the ***isDefault*** parameter for the tab (see example). If the parameter is not specified, the first tab will be opened when the card is opened for the first time. In the future, the tab selected by the user is saved in the storage.

[Documentation]()

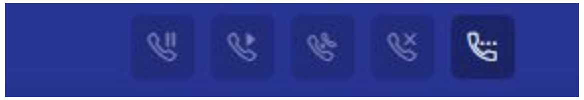
```
{
  "type": "tabs",
  "size": 70,
  "children": [
    {
      "label": "debtor.information.title",
      "type": "group",
      "groupType": "vertical",
      "size": 100,
      "display": ">> hasPermission(\"DEBT_FULL_INFO_ALL_VIEW\")",
      "children": [
    },
    {
      "label": "debtor.debtsTab.title",
      "type": "group",
      "groupType": "vertical",
      "size": 100,
      "isDefault": true,
      "children": [
    },
    {
      "label": "debtor.contactLogTab.title",
      "type": "group",
      "groupType": "vertical",
      "size": 100,
      "display": ">> hasPermission(\"CONTACT_LOG_VIEW\")",
      "children": [
    },
```

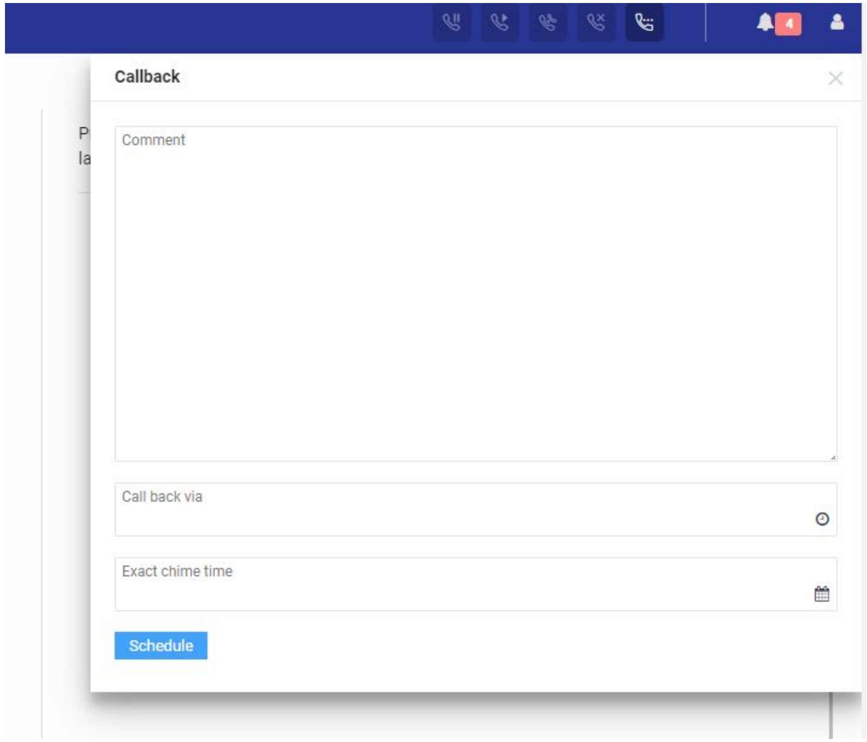

Together to the next level

Dynamic Window Configuration and Additional telephony controls

- Additional telephony controls

The new buttons support the use of any product operations.





- Dynamic Window

A dynamic window is an everlasting frame that can be opened using a product operation.

Its setting and configuration is done in the ***root.json.***

Additional telephony controls configuration

- Example

```
"additionalTelephonyControls": [
  {
    "icon": "co-call-settings",
    "title": "RECALL",
    "action": {
      "messageType": "DYNAMIC_WINDOW_OPERATION",
      "payload": {
        "operationType": "DYNAMIC_WINDOW_OPEN",
        "params": {
          "key": "recall"
        }
      }
    },
    "display": ">> true()",
    "enabled": ">> true()"
  }
],
```

- [Definition documentation]()
- [Product operations documentation]()

Dynamic Window Configuration

- Configuration example

```
"dynamicWindowsConfigurations": [
  {
    "url": "http://localhost:4210/recall-example",
    "key": "recall",
    "title": "Callback",
    "styles": {
      "height": "490px",
      "padding": "20px"
    }
  }
]
```

- [Definition documentation]
- [Open/Close operations documentation]
- [How to get info from pbx WS in iframe]

Together to the next level

Technical features demo

- For templates and formulas, the ability to format dates and numbers depending on the system locale has been developed. New system constant *System.DefaultFormat.LocaleId*.

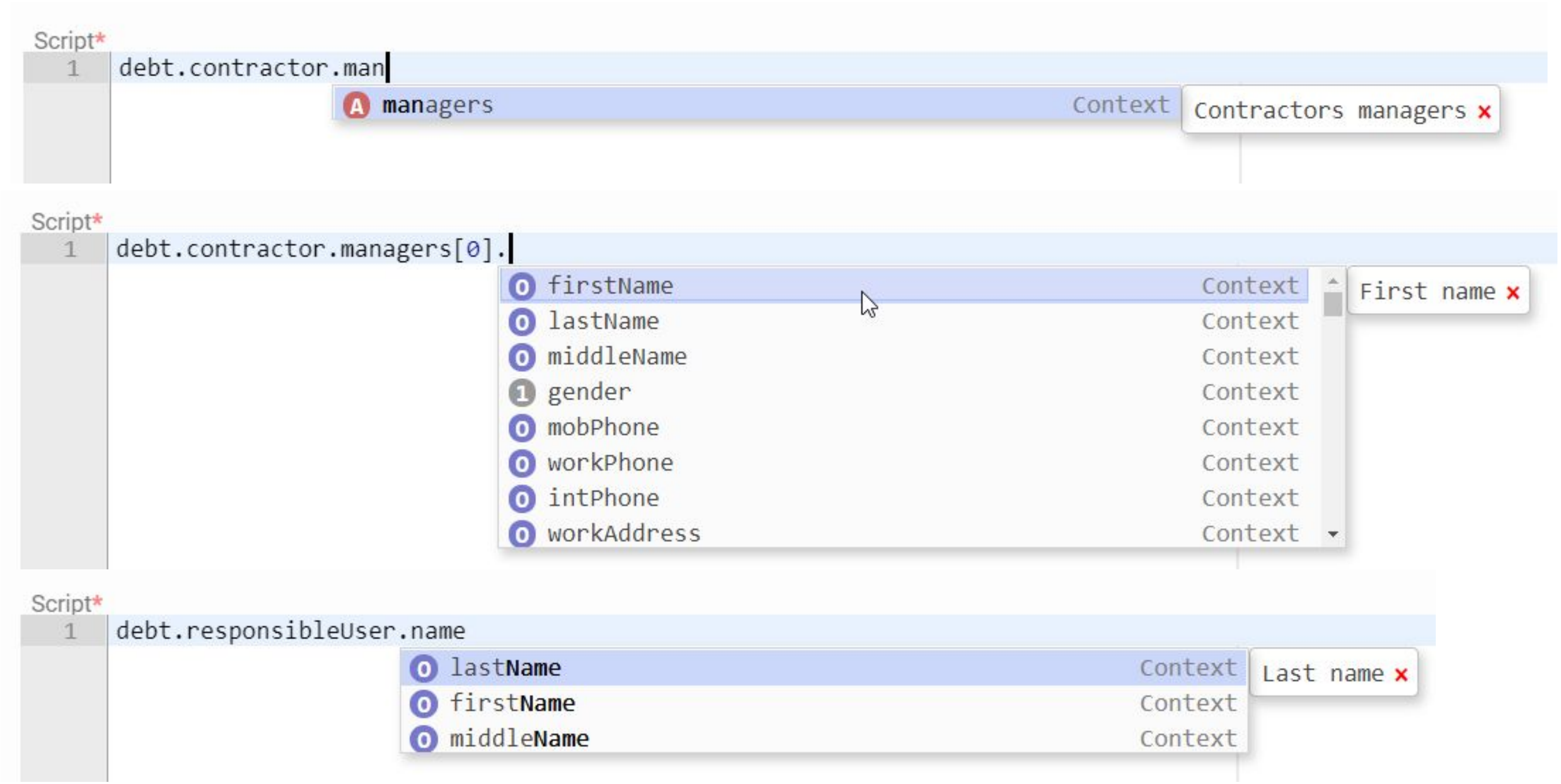
- A ‘Contractor manager’ model has been added to formulas.

  - first name, last name, middle name
  - mobile phone, work phone, internal phone
  - work address, position, branch

- The ‘User’ model for formulas has been extended with the same fields.

Formula models - http://confluence.luxbase.int:8090/pages/viewpage.action?pageId=167182349

Technical features



Technical features

Script*

```
debt.createDt
```

Type
Datetime

Result
2017-08-02T17:23:47+0300

Script*

```
formatDateByDefaultLocale(debt.createDt, true)
```

Type
String

Result
02.08.2017 17:23:47

Script*

```
formatDateByDefaultLocale(debt.createDt, false)
```

Type
String

Result
02.08.2017

Technical features

- The following EPS points have been supported:

  - PromisePlanCreatedEvent — called when creating a promise plan promiseCoverAmount
  - PromisePlanUpdatedEvent — called when updating a promise plan

  EPS points - http://confluence.luxbase.int:8090/pages/viewpage.action?pageId=219742239

Technical features

- Constants associated with calculating the status of promises by formulas have been hidden from the user interface. Instead, custom grids now support tags that are used for calculating status of promises and status of promise plans.

Constants

ID	Type	Name
5	Float	Promise.CoveragePercent
6	Number	Promise.GraceDays
7	Number	Promise.ExpireDays

Example of custom grid fromClause:

promise p
inner join debt_promise dp on p.id = dp.r_promise_id
$PROMISE_STATUS_FROM_CLAUSE_TAG{p.id}

For more information see
http://confluence.luxbase.int:8090/display/WEB20/Tags

Technical features

- Updated functionality for configuring email senders. Now, it is possible to specify connection settings for each sending server.

OLD

```
mail.smtp.host=smtp.gmail.com
mail.smtp.port=587
mail.smtp.auth=true
mail.smtp.starttls.enable=true
mail.debug=true
mail.smtp.auth.username=myTestGmail
mail.smtp.auth.password=myPassword
```

NEW

```
contact.mail.sender.1.mail.debug=true
contact.mail.sender.1.mail.smtp.host=smtp.gmail.com
contact.mail.sender.1.mail.smtp.port=587
contact.mail.sender.1.mail.smtp.auth=true
contact.mail.sender.1.mail.smtp.starttls.enable=true
contact.mail.sender.1.mail.smtp.ssl.trust=*
contact.mail.sender.1.mail.smtp.auth.username=myTestGmail
contact.mail.sender.1.mail.smtp.auth.password=myPassword

contact.mail.sender.2.mail.debug=true
contact.mail.sender.2.mail.smtp.host=smtp.yandex.ru
contact.mail.sender.2.mail.smtp.port=465
contact.mail.sender.2.mail.smtp.auth=true
contact.mail.sender.2.mail.smtp.auth.username=myTestYandexMail
contact.mail.sender.2.mail.smtp.auth.password=myPassword
contact.mail.sender.2.mail.smtp.ssl.enable=true
```

Technical features

- Strategy One

1. S1 output for promises extended with following parameters:
   - promiseIsUnconfirmed
   - promiseCoverAmount
   - comment
   - isCanceled

2. The S1 input model for Promises has been extended with ‘isCanceled’ parameter, which indicates whether or not a promise is cancelled. Also, the Promise Debt Link entity, which identifies the debts linked to a promise, has been supported.

3. The S1 output model for emails has been extended with a 'sender ID' parameter. When the parameter is provided, an email is sent from the specified sender, otherwise—from the default sender.

Input - http://confluence.luxbase.int:8090/display/WEB20/S1+Input+data
Output - http://confluence.luxbase.int:8090/pages/viewpage.action?pageId=299335798

Technical features

- ❑ Added formula whitelist property: *formula.whitelist.classes*

```
formula.whitelist.classes= org.springframework.util.StringUtils,org.slf4j.*
```

```
import org.springframework.util.StringUtils
def myMap = StringUtils.capitalize(debt.contract);
```

Technical features

- Delivery team can now integrate their external java module into product module in some cases:

Integration case	Required product dependency	Confluence page
Custom excel upload format	contact-back-end-excel-load-api	http://confluence.luxbase.int:8090/pages/viewpage.action?pageId=482967588
Custom formula methods and models	contact-back-end-formula-api	http://confluence.luxbase.int:8090/pages/viewpage.action?pageId=519536667
Custom report/printable generation library	report-generator-api	http://confluence.luxbase.int:8090/pages/viewpage.action?pageId=581894214

Global Demo Environment

Global Demo Environment

We created the global demo environment for functional education of CRIF Employees.

The Global Demo Environment has only core product-settings.

The main purposes:

- to learn the functionality and capabilities of Contact
- to self-conduct demonstrations for clients

About using the Global Demo Environment

- Demo address

The Global demo environment is available at http://demo.contact.crif.com/ without VPN

- Support

The global demo environment is available 24/7 for any market

Support group - demo.contact@crif.com

- Access

To access the global demo environment, you need to send a request to demo.contact@crif.com

- Demo for Clients

To conduct the demo with clients or employees you need to book in the calendar demo.contact@crif.com to make sure it is stable during this time

Document Collection about Contact

CRIF
Together to the next level

Document Collection about Contact

On our portal https://project.crifnet.com/prj/365/default.aspx you can find all the necessary information about Contact:

- The product presentation
- User's guides, release notes
- The product road map
- Video lectures
- Technical documentation

Road Show Feedback