e-library©

Functionality Project

Document version: **0.2**

Affect project version: **v2.0**

Author: **Jakub Rutkowski**

Date: **20.03.2020**

Table of contents

Change list:

Date	Version	Description	Author
02.04.2002	0.2	Adding new stories	Jakub Rutkowski

System data feeding (#135)

Description

System data feeding should be available for superuser (admin) as the option on settings form, below system data item. User will have access to the processes such as:

- Administrator data feeding;
- Messages and notifications data feeding.

Functionalities

System data feeding panel should consist of a table with following columns:

- Feed type;
- Data;
- Action.

Table should not be populated from data base. It is static one. Looking at the whole system, the table is unique in two things. First of all, last column – 'Actions' – stores buttons. These buttons are dedicated for each process – administrator data feeding process and messages and notifications dictionaries data process. By clicking 'Run' particular process should be executed and appropriate information dialog should be shown. By clicking 'Check', validation process should be ran and results should appear in the 'Data' column. After that information dialog should be displayed. These results are the data which is feeded by the process. For administrator data feeding process it will be just administrator data record (from com_user_data, usually, user with ID = 0), the data format should be like this:

[<ID_USER_DATA>, <LOGIN>, <PASSWORD>]

For messages and notification process the result shoud be messages types from mes_message_type, the data format should be like this:

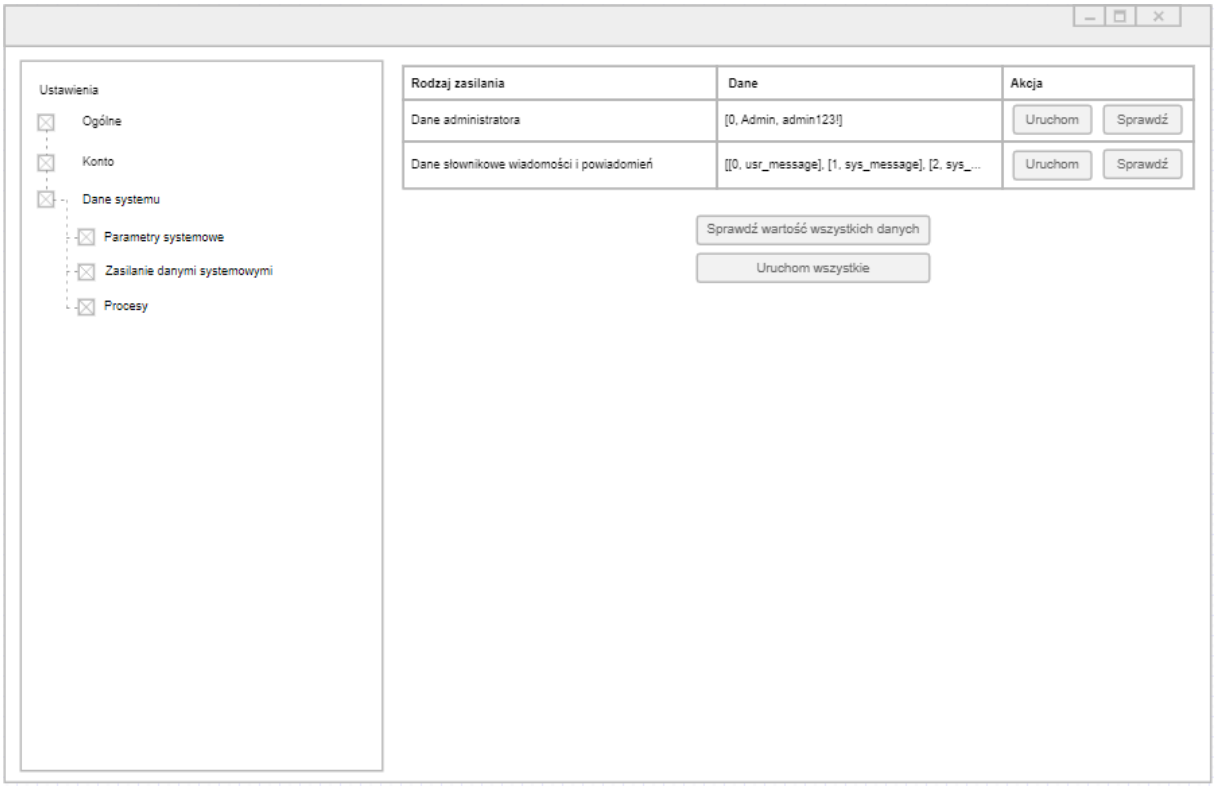
[[<ID_MESSAGE_TYPE>, <NAME>, <DESCRIPTION>],…,[<ID_MESSAGE_TYPE>, <NAME>, <DESCRIPTION>]]

Condition, if the data already exists in the system during the process execution should not be checked. If exists, nothing should be changed in data base.

The validation for administator data should checks the administrator ID, login and password. These data should be same as these defined in application. In case of messages types, only names should be checed. All messages types should exists in the system. If any errors will be found, error should be displayed in the data column (error text should be red). Correct data should be green.

Interfaces & Integrations

Mockup



Data model

Table name	Column name	Column Data Type	Column Description	Description

Translations

Resource	PL	Default	GB
Tree menu item	Dane systemu	System data	System Data
Tree menu item	Zasilanie danymi systemowymi	System data feeding	System data feeding
Table header	Rodzaj zasilania	Feed type	Feed type
Teble content	Dane administratora	Administrator data	Administrator data
Table content	Dane słownikowe wiadomości i powiadomień	Messages and notifications dictonary data	Messages and notifications dictonary data
Table button	Uruchom	Run	Run
Table button	Sprawdź	Check	Check

Button	Sprawdź wartość wszystkich danych	Check value for each data	Check value for each data
Button	Uruchom wszystkie	Run all	Run all

System parameters (#135)

Description

System parameters should be available for superuser (admin) as the option on settings form, below system data item. User will have access to the configuration of some system parameters such as:

- Application penalty value;
- Application email address;
- Application email password;
- Application generated password length;
- Application templates directory.

Each parameters data should be stored in data base.

Functionalities

On the system parameters screen there should be a pair of some label and text fields/numeric field/password filed for each parameters. All value components should be populated with saved/default values while navigating into the screen. The parameters should be in data base and should be fed at the application start up same as administrator data and messages and notifications data. Above functionality is described as part of the next chapter (**System parameters feeding process**). The following parameters should be presented on the screen:

Application penalty value is the parameter which can be only the numeric, positive, real value. User could passed any value in range from 0.00 to 100.00, step 0.1. Penalty is used on 'Loan Service' tab in case of Client will not return book in the term. For each next month penalty will increase with defined penalty value.

Application email address and password are e-mail server authentication data which are used to make connection to the e-mail server. Passed password is masked by the system with dot character as it is f.eg. on login form. Below password text field there should be the check box, which will indicate if data validation should be performer after clicking button accept. If the check box is selected system will check if provided data is correct and system could initiate connection to the e-mail server. Described check box should be enabled only if internet connection is available (information about that is presented in the right, bottom corner of the main, tabbed form).

Application generated password length could be changed in case if administrator would like to change the length of generated password sent by system if the password is forgotten by the user.

Application templates directory will indicate the name of the main catalogue where the e-mail templates are saved by the system (option available on 'Templates' tab).

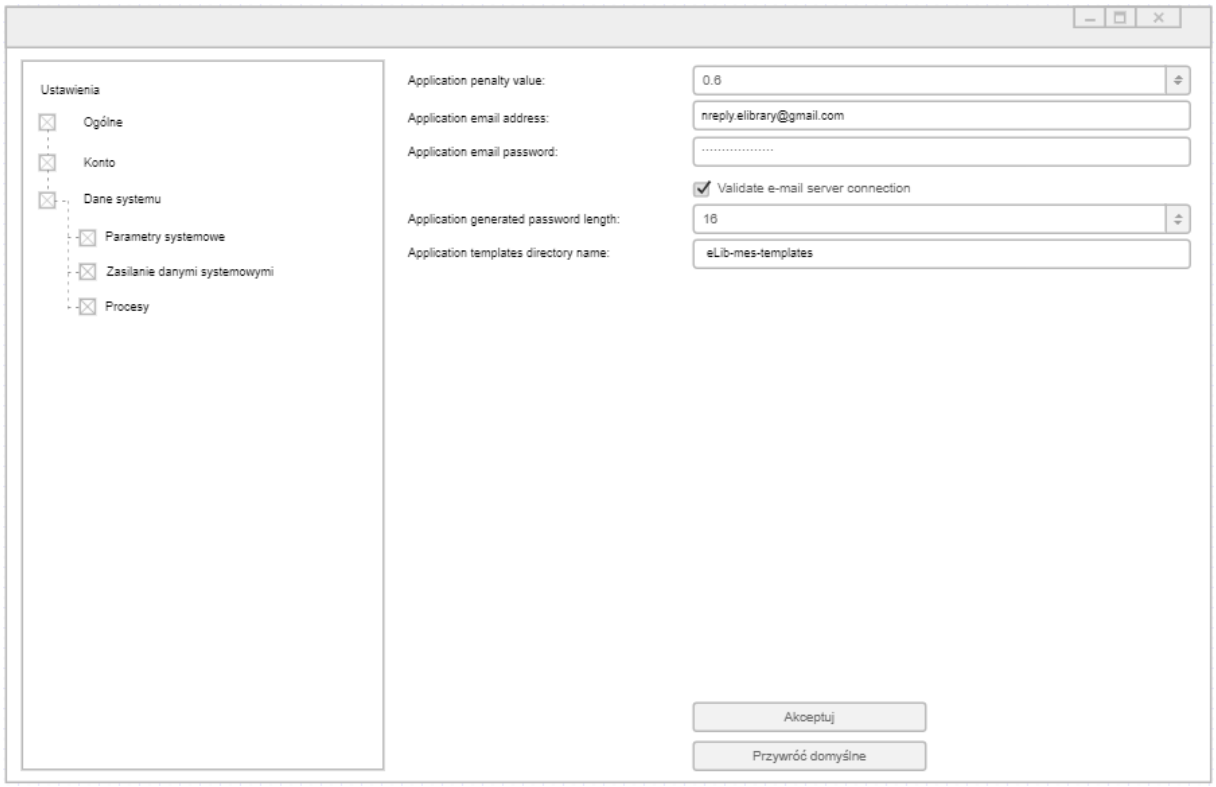
After then accept button – that is placed below parameters data form – will be clicked, system will display confirmation dialog and check if user is sure, that he want to make changes, next system will fetch passed data into the data base. From that point, new values of parameters will be used by the

system. Moreother, there is the ‘Restore default values’ button which could be used for restoring default values of parameters. The default values are like below:

- Application penalty value – 0.6;
- Application email address – nrepley.elibrary@gmail.com;
- Application generated password length – 16;
- Application templates directory – eLib-mes-templates.

Interfaces & Integrations

Mockup



Data model

Table name	Column name	Column Data Type	Column Description	Description
com_system_parameter	id_system_properties	int4		PK, not null
	name	varchar(80)		unique
	value	varchar(80)		
	default_value	varchar(80)		

The names of the parameters should be capitals separated with ‘_’ character f.eg. for ‘Application penalty value’ it will be APPLICATION_PENALTY_VALUE.

Translations

Resource	PL	Default	GB
Tree menu item	Parametry systemowe	System parameters	System parameters
Label	Wartość kary:	Application penalty value:	Application penalty value:
Label	Adres e-mail serwera pocztowego:	Application e-mail address:	Application e-mail address:
Label	Hasło serwera pocztowego	Application e-mail password:	Application e-mail password:
Check box	Zwaliduj połączenie z serwerem pocztowym	Validate e-mail server connection	Validate e-mail server connection
Label	Długość wygenerowanego hasła:	Application generated password length:	Application generated password length:
Label	Nazwa głównego katalogu szablonów e-mail:	Application template directory name:	Application template directory name:
Button	Akceptuj	Accept	Accept
Button	Przywróć domyślne	Restore default values	Restore default values
Confiramtion dialog	Do you really want to continue?	Czy na pewno chcesz kontynuować?	Czy na pewno chcesz kontynuować?

System parameters feeding process (#135)

Description

System parameters data feeding process should be implemented and available for superuser (admin) for manual execution as part of the new process in feeding processes table in settings, in 'System data feeding'.

Functionalities

Described process should ran at the beginning, when application is started (same as f.eg. administrator data feeding process). Then – after clicking login page 'Accept' button – process should be executed and value of number_of_system_parameters in com_system_data (see chapter **System data feeding process**) should be updated with the actual value of system parameters amount. Below parameters data should be added into the com_system_parameter table if not already exists:

name	value	default_value
APPLICATION_PENALTY_VALUE	0.6	0.6
APPLICATION_EMAIL_ADDRESS	nreply@gmail.com	nreply@gmail.com
APPLICATION_EAMIL_PASSWORD		
APPLICATION_GENERATED_PASSWORD_LENGTH	16	16
APPLICATION_TEMPLATE_DIRECTORY_NAME	eLib-mes-templates	eLib-mes-templates

System parameters data feeding process could be ran manually and checked in feeding process table. The results of execution are the data which is fed by the process. For system parameters data feeding process the results are the data records (from com_system_parameters), the data format should be like this:

[[<NAME>, <VALUE>],…,[<NAME>,<VALUE>]]

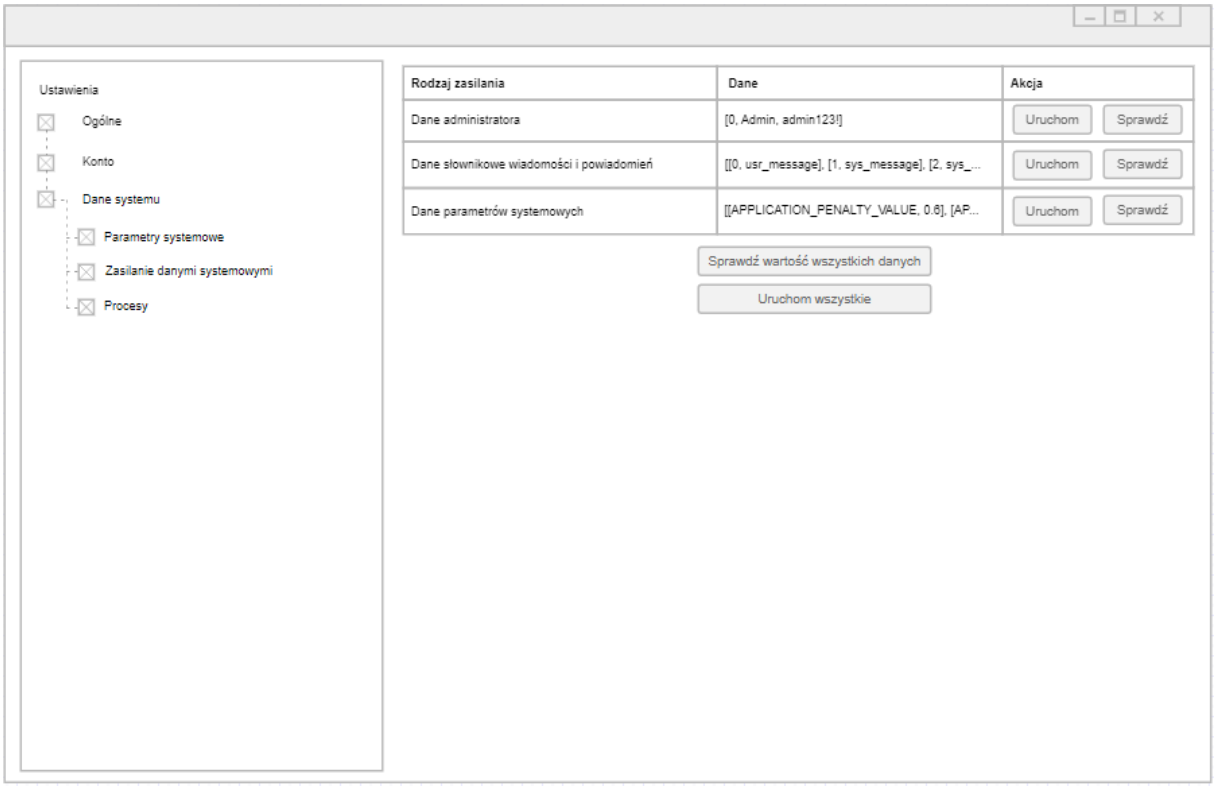
Condition, if the data already exists in the system during the process execution should not be checked. If exists, nothing should be changed in data base.

Whole business logic of f.eg. buttons actions, dialogs should be same as for other processes in the table. In case of system parameters data feeding process validation should be performed by checking if all parameters with appropriate name exists.

The validation for system parameters data should check the amount of system parameters which should be equal to this, which is saved in the com_system_data. If an error will be found, error should be displayed in the data column (error text should be red). Correct data should be green.

Interfaces & Integrations

Mockup



Data model

Table name	Column name	Column Data Type	Column Description	Description

Translations

Resource	PL	Default	GB
Table content	Dane parametrów systemowych	System parameters data	System parameters data

System data feeding process (#135)

Description

System data (for now library system data too) feeding process should be implemented and available for superuser (admin) for manual execution as part of the new process in feeding processes table in settings, in 'System data feeding'.

Functionalities

Described process should ran at the beginning, when application is started (same as f.eg. administrator data feeding process) as the first ran process. Then – after clicking login page 'Accept' button – process should be executed and below system data should be added into the com_system_data table if not already exists:

column name	value
system_data_initialization_date	current date
id_system_data	1

As well, reference ID should be added to the lib_system_data and newly created lib_system_data should be connected to the lib_library_system_data (see **Library data feeding process (#135)**

Library data feeding process (#135)).

column name	value
id_library_system_data	1 (FK)
id_library_data	1 (FK)

System data feeding process could be ran manually and checked in feeding process table. The results of execution are the data which is fed by the process. For system data feeding process the results are the data records (from com_system_data), the data format should be like this:

[<VERSION>,<BUILD_NUMBER>,<INSTALLATION_DATE>,<SYSTEM_DATA_INITIALIZATION_DATE>,<NUMBER_OF_SYSTEM_PARAMETERS>]

Condition, if the data already exists in the system during the process execution should not be checked. If exists, nothing should be changed in data base.

Whole business logic of f.eg. buttons actions, dialogs should be same as for other processes in the table. In case of system data feeding process validation should be performed by checking (*temporary solution which should be implemented till system configuration file functionality will not be deployed*) if any records in com_system_data exists. If an error will be found, error should be displayed in the data column (error text should be red). Correct data should be green.

Interfaces & Integrations

Mockup

Data model

Table name	Column name	Column Data Type	Column Description	Description
com_system_data	number_of_system_parameters	int4		
	system_data_initialization_date	date		

Translations

Resource	PL	Default	GB
Table content	Dane systemowe	System data	System data

Library data feeding process (#135)

Description

As system data should be connected to library data, library data should be fed in process which should be available for superuser (admin) for manual execution as part of the new process in feeding processes table in settings, in 'Library data feeding' too.

Functionalities

Described process should ran at the beginning, when application is started (same as f.eg. administrator data feeding process) as the first ran process. Then – after clicking login page 'Accept' button – process should be executed and below system data should be added into the lib_library_data table if not already exists:

column name	value
id_library_data	1
name	Library
branch	Main branch

Library data feeding process could be ran manually and checked in feeding process table. The results of execution are the data which is fed by the process. For system data feeding process the results are the data records (from lib_library_data), the data format should be like this:

[<NAME>,[<ADDRESS>,<NAME>,<CITY>],…,[<ADDRESS>,<NAME>,<CITY>]]

Condition, if the data already exists in the system during the process execution should not be checked. If exists, nothing should be changed in data base.

Whole business logic of f.eg. buttons actions, dialogs should be same as for other processes in the table. In case of library data feeding process validation should be performed by checking (*temporary solution which should be implemented till system configuration file functionality will not be deployed*) if any records in lib_library_data exists. If an error will be found, error should be displayed in the data column (error text should be red). Correct data should be green.

Interfaces & Integrations

Mockup

Data model

Table name	Column name	Column Data Type	Column Description	Description
lib_library_system_data	lib_library_data	int4		FK to com_system_data

Translations

Resource	PL	Default	GB
Table content	Dane biblioteki	Library data	Library data

System processes (#135)

Description

System processes should be available for superuser (admin) as the option on settings form. User will have an access to the configuration of some processes such as:

- Network process;
- Directory process.

Functionalities

On the system processes screen there should be two check boxes and labels and two buttons. All check boxex should be unselected by default and labels should be empty. The buttons are connected to below actions:

- Stopping process;
- Checking process health.

By executing 'Stop', selected processes should be stopped if was already ran, then information dialog should be displayed and processes health labels should be reloaded for selected processes. Second button should allow user to checking processes state – health. The result could be – ran or stopped. Result of fist case should be presented as green 'tick' label and second as red 'cross'. For unselected processes 'health label' should be set to blank. After clicking 'Check health' information dialog should be displayed.

Interfaces & Integrations

Mockup

Data model

Table name	Column name	Column Data Type	Column Description	Description

Translations

Resource	PL	Default	GB
Table content	Procesy	Processes	Processes
Information dialog – 'Stop'	Zatrzymanie wybranych procesów	Selected processes stopped successfully.	Selected processes stopped successfully.

	zakończyła się pomyślnie.		
Information dialog title – ‘Stop’	Informacja	Information	Information
Information dialog – ‘Check health’	Sprawdzenie statusu wybranych procesów zakończyło się pomyślnie.	Health of selected processes checked successfully.	Health of selected processes checked successfully.
Information dialog title– ‘Check health’	Informacja	Information	Information
Button	Uruchom	Run	Run
Button	Sprawdź status	Check health	Check health

Library system properties (#XXX)

Description

Functionalities

Interfaces & Integrations

Mockup

Data model

Table name	Column name	Column Data Type	Column Description	Description
lib_library_system_parameters				

Translations

Resource	PL	Default	GB
Table content	Dane bilbioteki	Library data	Library data

System processes – run button (#XXX)

Description

Functionalities

Interfaces & Integrations

Mockup

Data model

Table name	Column name	Column Data Type	Column Description	Description
lib_library_system_parameters				

Translations

Resource	PL	Default	GB
Table content	Dane bilbioteki	Library data	Library data