FORS

universalInterface Live

Live Universal Interface

LUI

Getting Started

Description of an example of creating application screen forms

Total pages - 48

Moscow 2019-10-17

Content

1. Terms and abbreviations

LUI - Live Universal Interface, is a tool environment for rapid development a visual user interface for web applications and interacting with DBMS based on meta-programming.

Meta-programming - is a type of programming associated with the creation of programs based on descriptive data about objects (metadata) to control user actions, processing and displaying data in screen forms.

Metadata - information about the features and properties of screen forms for the work of programs to control the display, data processing and valid user actions.

Application - root object of the metadata hierarchy of a set of screen forms to control the display of data, user actions, and interaction with the DBMS.

Architect - role or group of user rights to create an Application and grant permissions to modify form metadata.

List - tabular type of screen form for searching and displaying data retrieved from an SQL query, and for calling other forms and procedures to modify the data.

Blank – type of screen form with fields for adding or changing data.

Menu - type of tree screen form with a hierarchy of fields for selecting and calling forms or actions.

2. Brief description

This document is intended for the initial practice of working with LUI, which describes the steps to create users of the development team and examples of creating basic screen forms of the Application (List, Form, Menu).

The Application development team consists of a user with the Architect role and several users with the Developer role.

The Architect manages the creation and configuration of the Application, including managing the Developer accounts of the Application's screen forms, and can also participate in the development of forms

Developers create Application screen forms. By default, only the author - Developer who originally created the form record in the LUI database can make changes to the form metadata.

The process of getting started with LUI's Application and screen form creation technology consists of the following steps:

- Check compliance with technical requirements (see Chapter 3)

- Owner of the database schema LUI creates user with the Architect role

- Architect creates a new Application

- Architect creates a new user with the Developer role and grant him the right to developed the forms of the Application

- Developer creates a List to view the records in the table

- Developer creates a Blank to add or edit a record

- Architect creates a Menu for selecting and calling the Application forms

- Architect checks the work of the Application forms.

This document provides examples where the Postgres DBMS and the LUI web server are installed locally on the computer, only the basic screen forms are considered without configuring access rights, formats, valid values for selecting and verifying the entered data.

For a more detailed description of the form customization tunes in LUI, see the «Life Universal Interface. Application development and execution environment. Programmer manual». In the file - "LUI_Guide204.pdf" to the URL http://lui.fors.com/LUI_Guide204.pdf .

3. Specifications

Technical requirements for creating an Application in LUI according to this description:

- The PostgreSQL DBMS must be preinstalled locally or on a remote server with network access

- LUI must be preinstalled as described in the document - "LUI. Installation manual for Windows", URL - http://lui.fors.com/LUI_Setup4Windows.pdf

- Require access to the database lui@luidb, where user «lui» is the owner of the database «luidb»

- Postgres database must be pre-created table and granted rights according to the description in the section Appendix A of this document.

Only after checking the fulfillment of the listed technical requirements, you can begin to perform the introductory steps described in this document.

Examples of technical requirements checks:

- Verifying the installation of LUI and call a login form from a web browser at the URL http://localhost:8088/, where localhost:8088 – the name or IP address and port of the web server LUI

- Check in the database table app_t_country and its contents from the command line. If successful, created the file «psql.log» with records from the table:

```
psql -h localhost -d luidb -U lui -c "select * from app_t_country" -o psql.log
```

4. Create a user with the Architect role

Creating a user with Architect rights is performed by the LUI database schema owner user in the following sequence:

- Log in to LUI as the database owner user (see Figure 1)

- Open the form of creating a new user from the menu "Administration - > Access Control - > Users", enter the values of the fields, include the user in the group of rights Architect and click on "Save" (see Figure 2)

- Check that the list of users has a new user who is a member of the Architect group (see Figure 3).
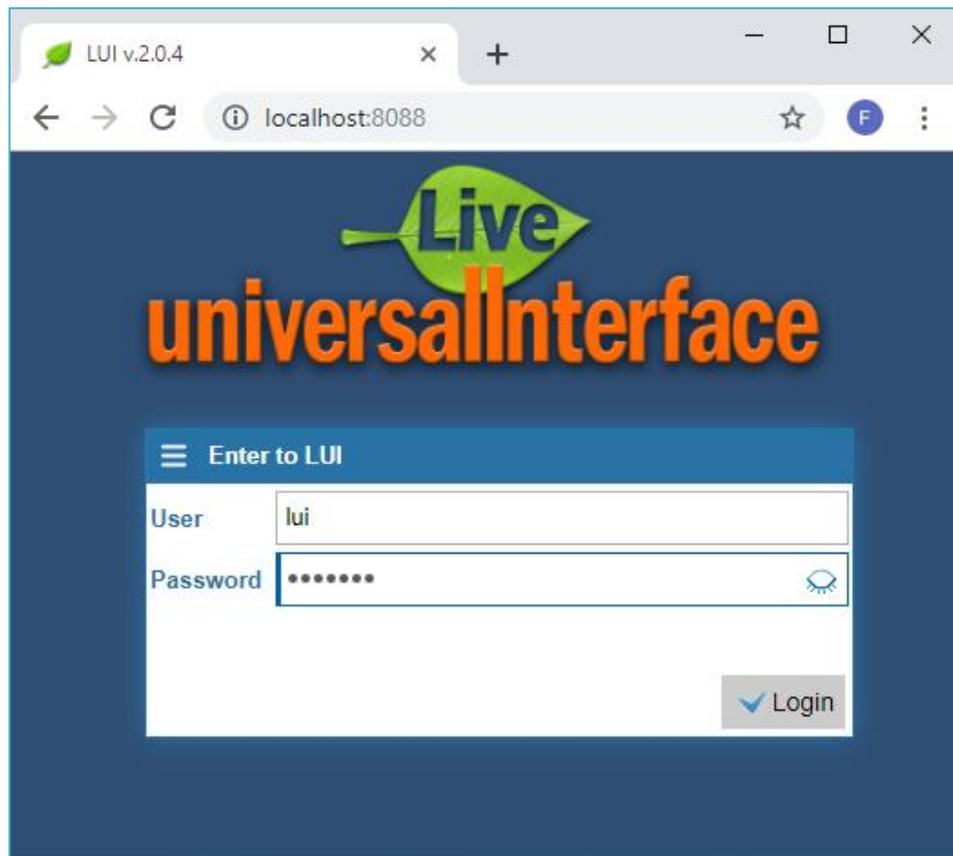
Figure 1. Login to LUI by the database schema owner to create the Architect

A user with the Architect role has rights to create a record of a new Application, create developer users, and assign them a form development Application.
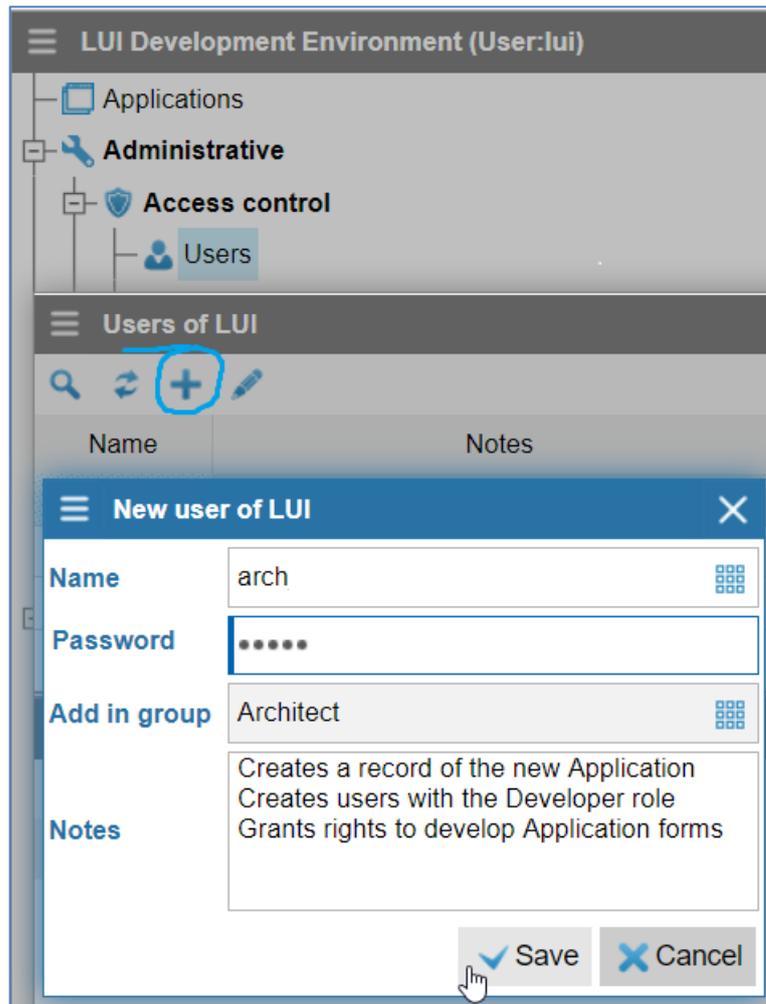
Figure 2. Form for creating a new user with the role Architect

Figure 3. A new user with the Architect role has been created and is listed

5. Add a new Application

Adding a new Application to LUI is done by the Architect in the following sequence:

- Log in to LUI as a user with the Architect role (see Figure 4)

- From the "Applications" menu, open the application list form and click on the "Add" icon (+). In the opened form "Adding of a new application system. Step1 " fill in the fields with the code and name of the new Application, and click on the "Next" button (see Figure 5)

- In the form "Adding of a new application system. Step2 " fill in the fields with database connection parameters and click on the "Test connection" button. If there are no errors after checking the connection to the database, click on the "Next" button (see Figure 6).

- In the form "Adding of a new application system. Step3" select all the proposed administrative functions to add to the new Application and click on the "Next" button (see Figure 7).

- In the form "Adding of a new application system. Finish" (Step4) enter a detailed description of the new application if necessary and click on the "Save" button (see Figure 8).

- Check that the new Application appears in the list for development, in the form from the Applications menu (see Figure 9).
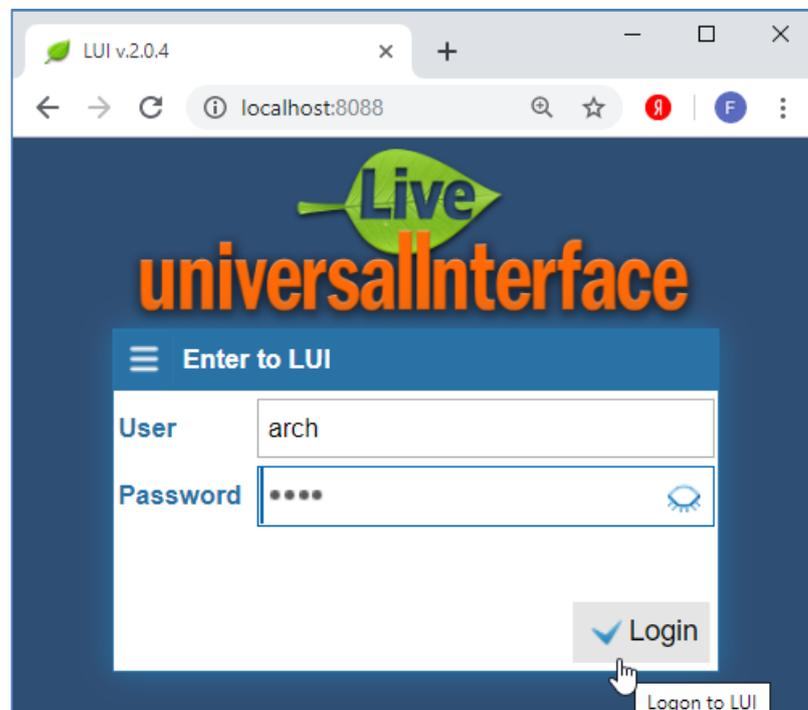
Figure 4. Login to LUI by Architect for add new Application and create new user with Developer role
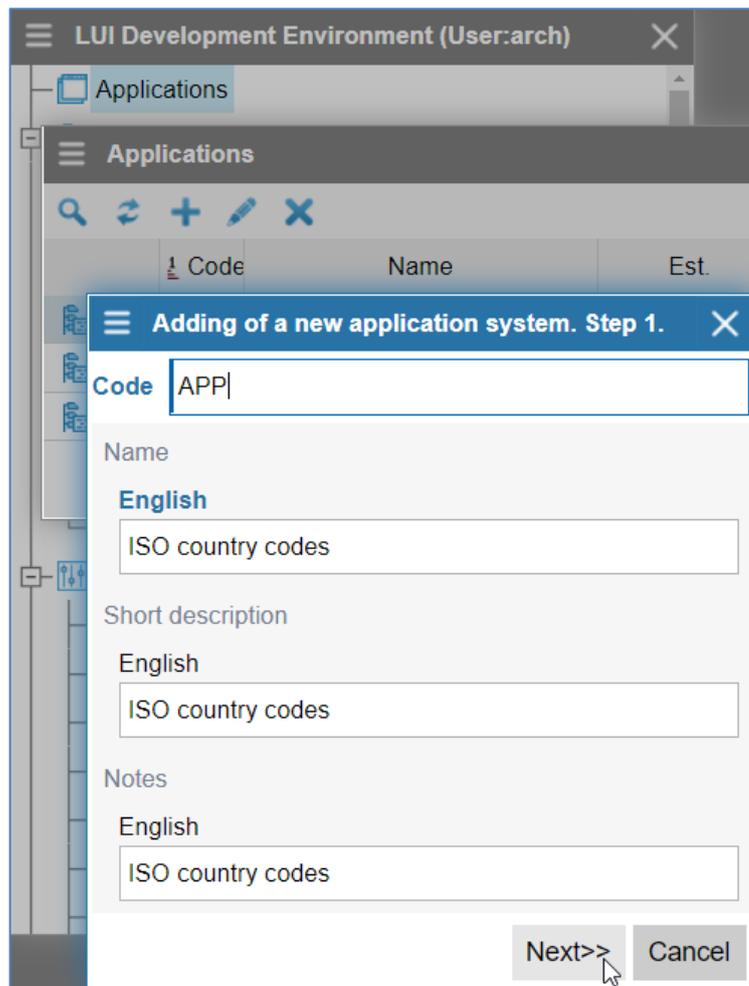
Figure 5. Adding of a new Application. Step 1. Enter code and name
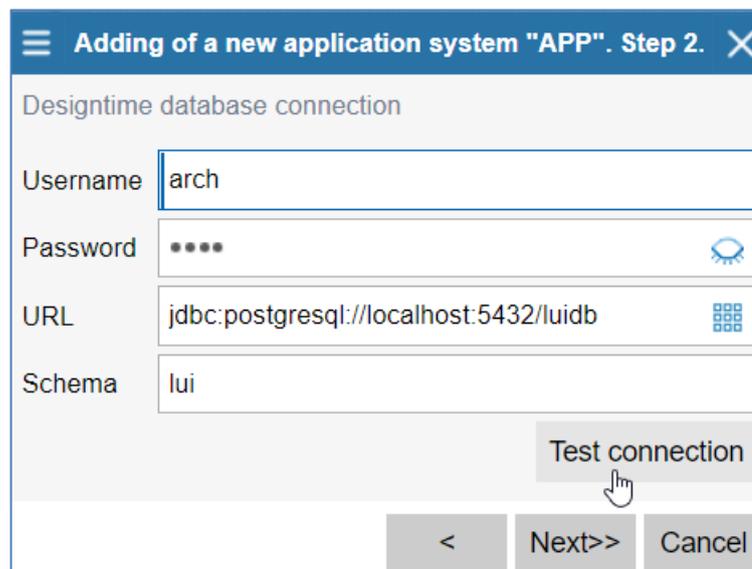
Figure 6. Adding of a new Application. Step 2. Entering and checking database connection parameters
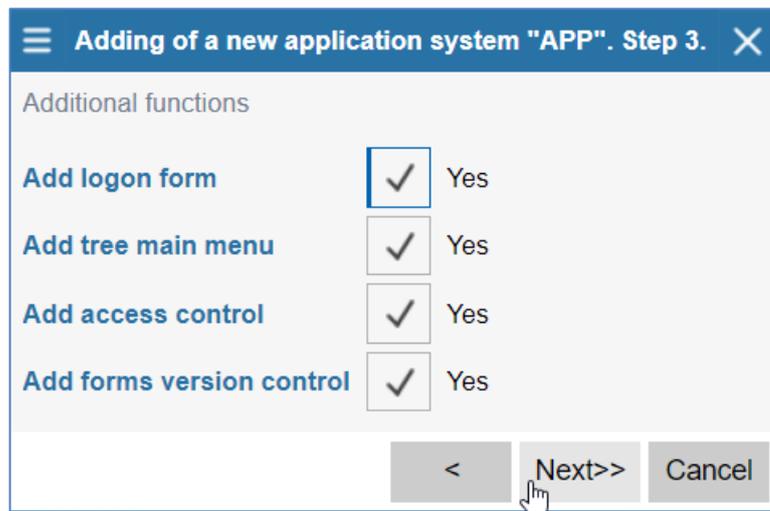
Figure 7. Adding of a new Application. Step 3. Adding of typical administrative functions

Figure 8. Adding of a new Application. Finish (Step 4). Detailed description of the Application
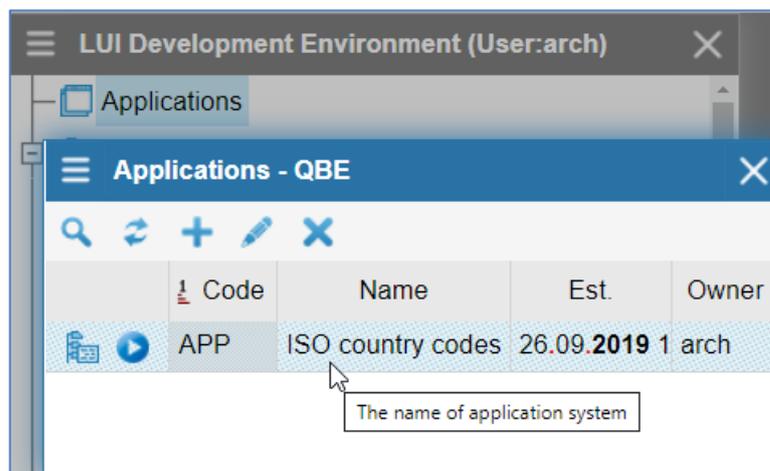
Figure 9. The new Application appeared in the list for development

6. Create a Developer and grant rights an Application

Creating a user with Developer privileges and granting rights an Application for development is performed by the user with the Architect role in the following sequence:

- Login to LUI as Architect (see Figure 4)

- Open the form for creating a new user from the menu "Administration -> Access Control -> Users", enter the name, password, include the user in the Developer rights group, specify the Application available for development and click on the "Save" button (see Figure 10)

- Check that the list of users has a new user with the role of Developer and a list of its Applications (see Figure 11).
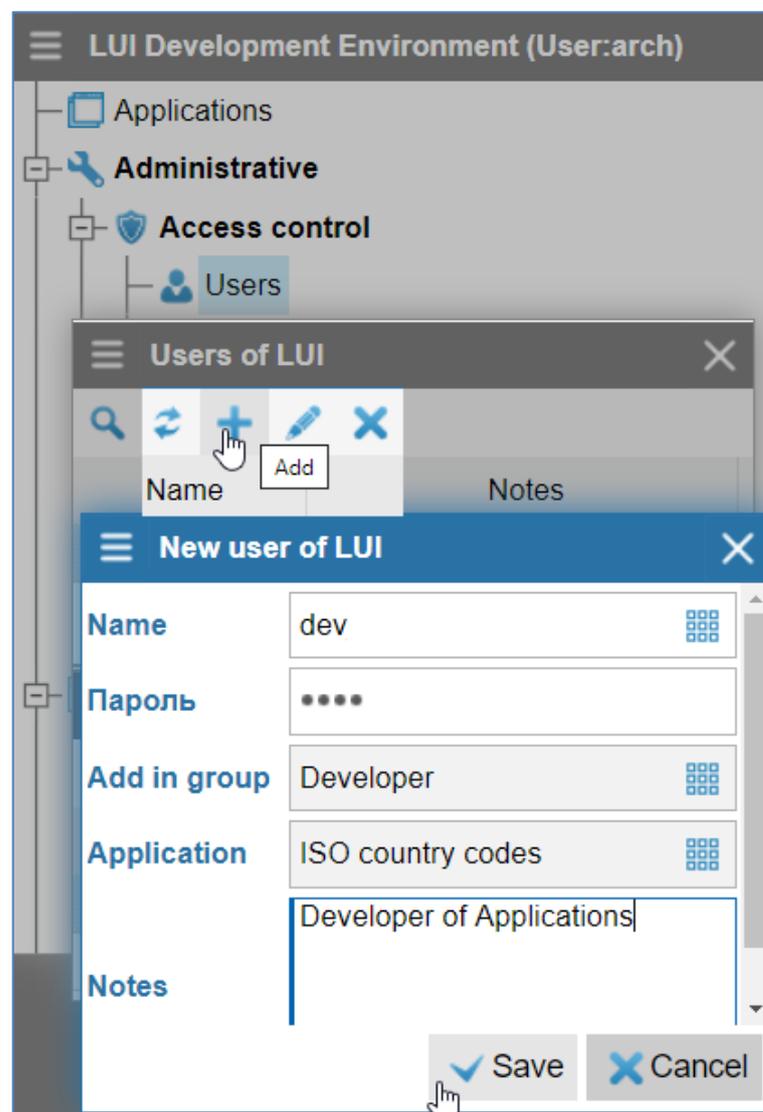
Figure 10. Architect creates a Developer and grants rights an Application
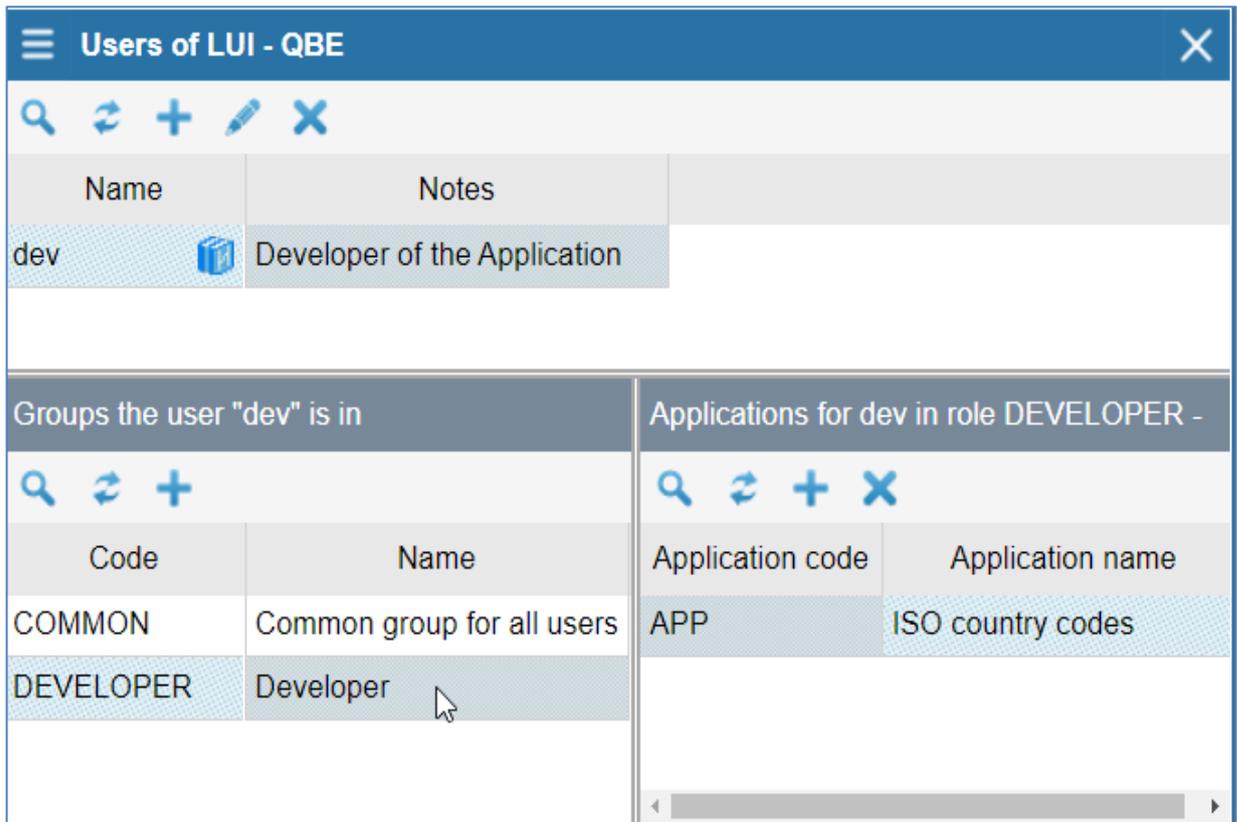
Figure 11. A new user with the Developer role and Applications list for development

After you create a user with Developer rights, you can change the list of Applications that are assigned to him for development.

To modify the Application list of Developer, navigate to form "Group the user ..." and put a pointer to the code string the group of DEVELOPER, then the right part will automatically show the form with the list of assigned Applications, which can be changed (see Figure 11).

Changing the list Applications of Developer is performed by actions "Delete" and "Add".

All active actions and changes in LUI are audited, and the database maintains a log of changes that can be viewed from LUI.

7. Call the form editor

Creating and modifying Application forms in LUI is done by the Developer using the metadata editor.

The Architect starts the application and validates the forms from It.

The form editor is called from the list "Applications", where from the menu on the right mouse click it is necessary to select and execute the action "Form designer", or click on the icon "Form designer" in the left part of the line (see Figure 12 and Figure 13).
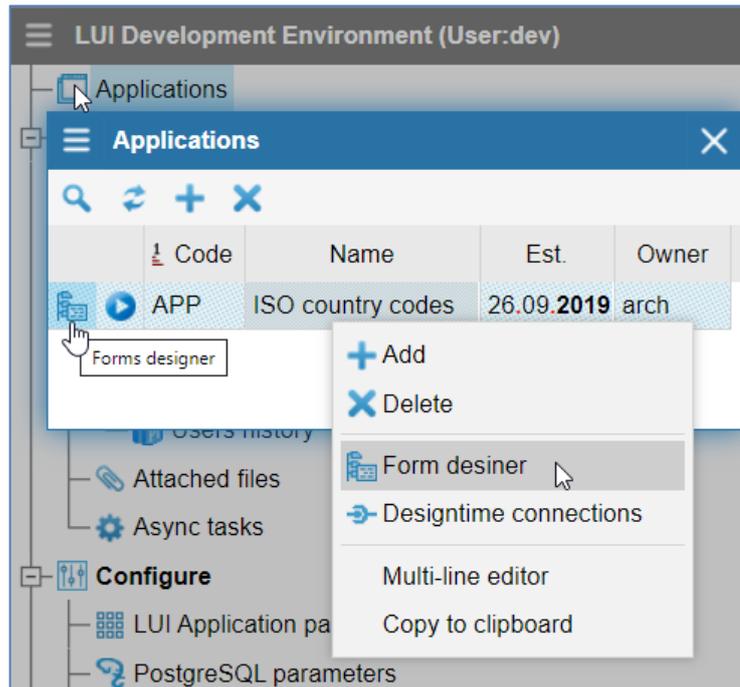
Figure 12. Calling the form metadata editor from the Application list
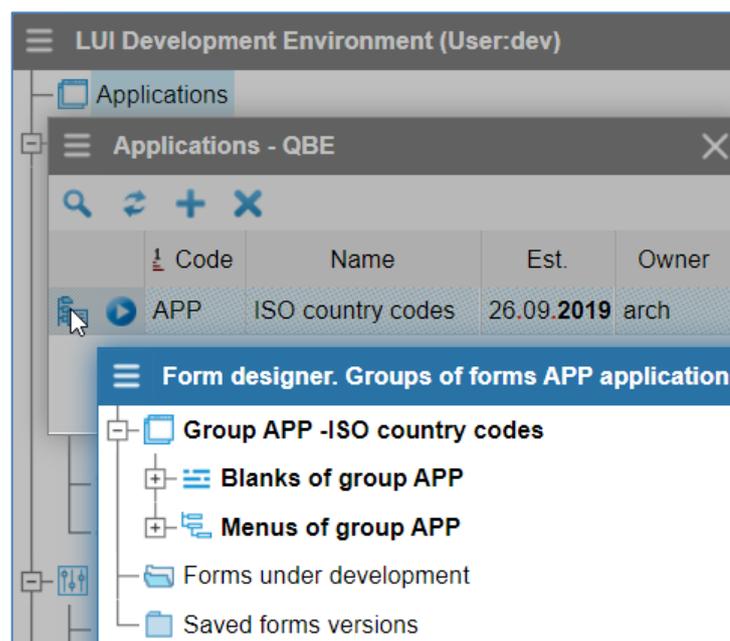
Figure 13. Metadata editor Application forms

8. Create a List form

Creating a form with the List type is performed by the Developer in the metadata editor (Figure 12 and Figure 13) in the following sequence:

- Login to LUI as Developer (see Figure 14)

- Create a new List form

- List configuration and SQL query input

- Create List elements: Field, Action and Output parameter

- Checking the operation of the new List.
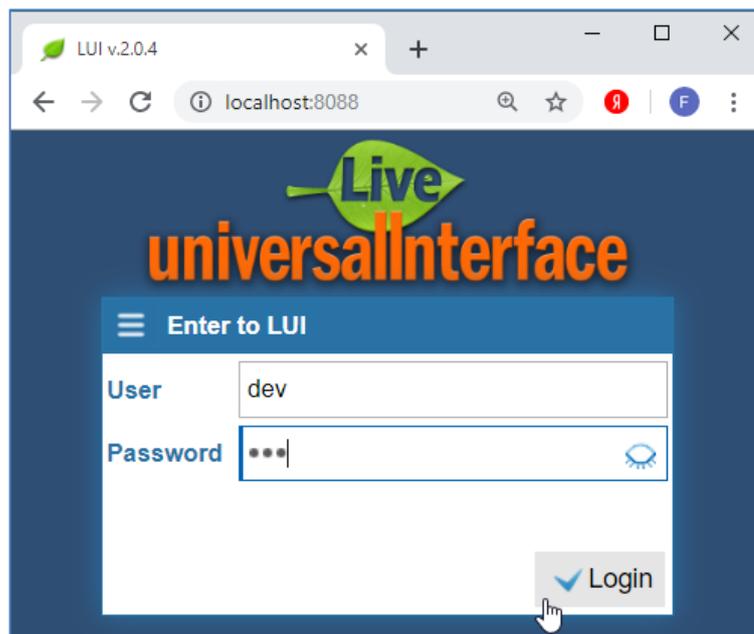
Figure 14. Login to LUI by Developer for creating and modifying Application forms

8.1. Create a new List form

To create a new List in the form editor:

- If there is no "Lists of group APP" node, when there is no List in the Application, go to the "Group APP – ISO country codes" root node. Right-click the menu and perform the action "Add List" (see Figure 15)

- If there is a node "List of group APP", expand it by clicking on the icon "+" in the left part of the menu item. Right-click the mouse to call and perform the action "+ Add"

- In the form "Create a new form group <…>", enter the value of the Code and Name of the basic configuration fields, and click on the "Save" button (see Figure 16).

If there are no errors, a form of List will appear in the "Lists of group <…>" node.
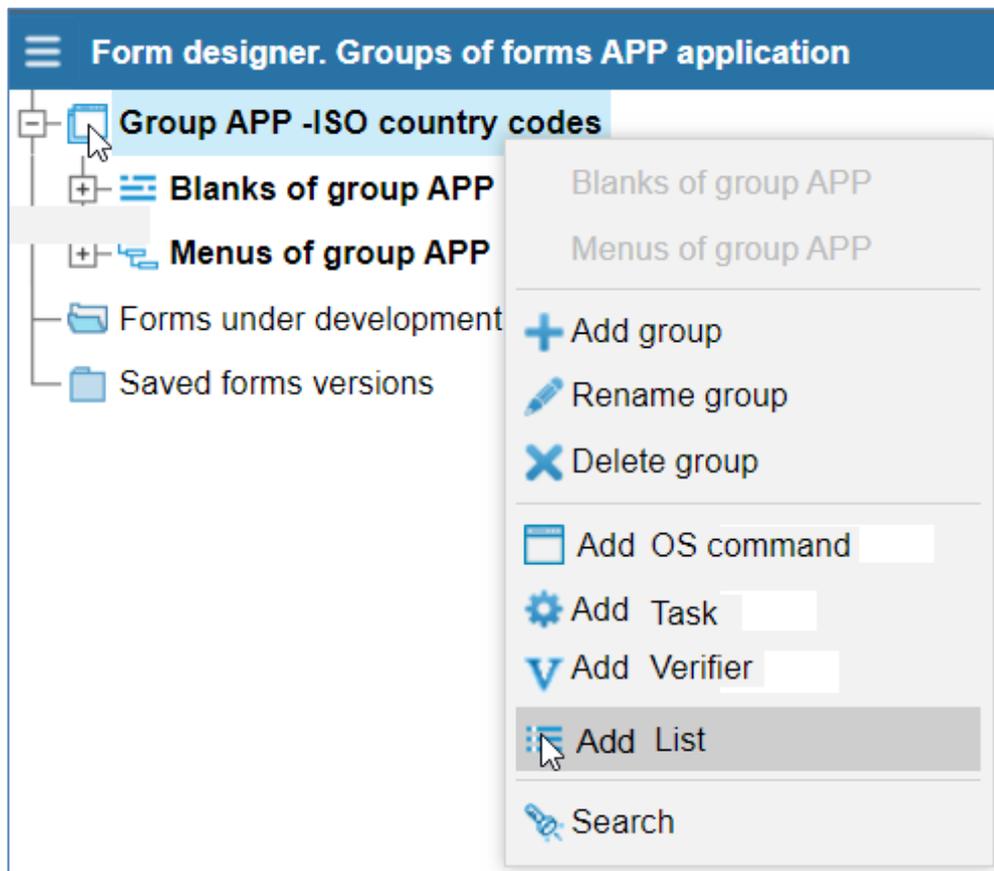
Figure 15. Call to add the first List in the Application

Figure 16. Create a new List form

8.2. List configuration and SQL query input

Configuration of the List and SQL query input is performed in the form that appears on the right, if you click on the node "Configuration DEFAULT – ISO country codes" (see Figure 17).
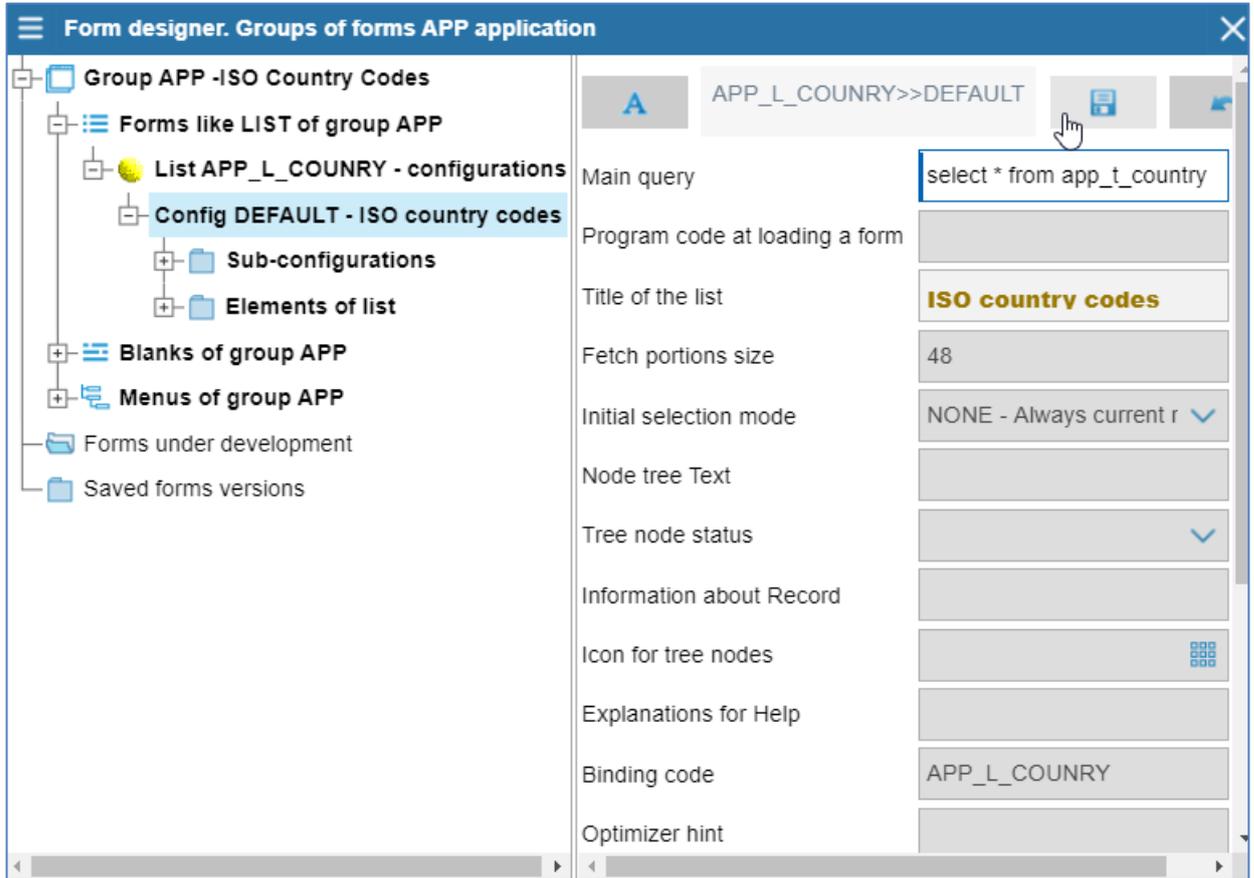
Figure 17. List configuration and SQL query input

Specify the SQL query to get the list and determine the composition of the table fields, the parameters of which are given in the Appendix A section of this document:

*SELECT * FROM app_t_country*

Specify the title of the List and click the "Save" button to save the settings.

8.3. Create List elements

This document describes how to create the following List elements: Column, Action (Add, Edit, Delete), and Output parameter.

To create List elements in the form editor, expand the "Elements of list APP_L_COUNTRY …" node by clicking the " + " icon in the left part of the menu item. Right-click and execute the action "+ Add element" (see Figure 18).
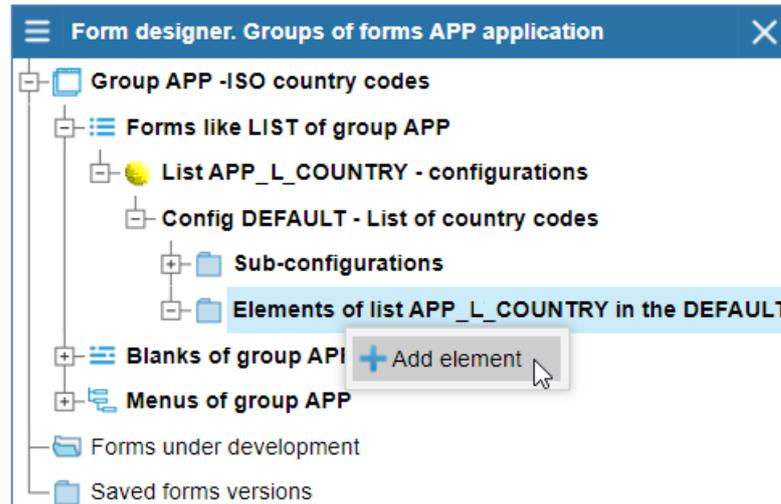
Figure 18. Adding new List elements

After executing the "Add element" action, the form is called to create a new List element (see Figure 19).

8.3.1. Create element of List - Column

To create a List element – Column, in the form "Create a new List element" in the field "Element type" select from the list the value "Column" (see Figure 19).
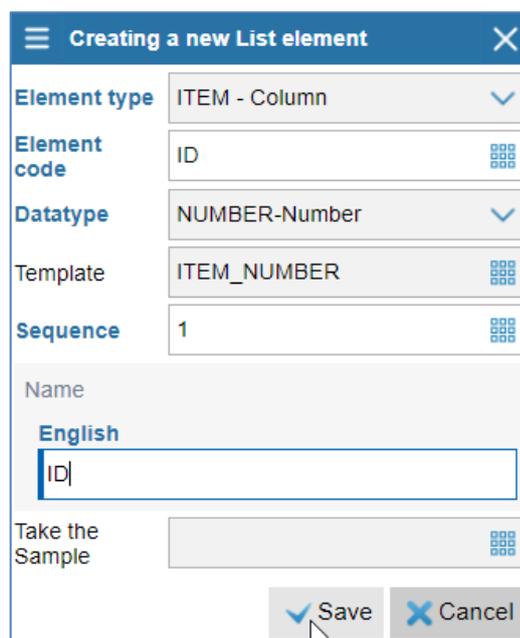
Figure 19. Creating a new List element - Column

In the field "Element code" select the code of the table field from the proposed list, which is obtained from the analysis of the SQL query in the List configuration. Specify the name of the List element and click on the "Save" button (see Figure 19).

If there are no errors, a new List item - Column will appear as part of the form elements in the "Elements of list APP_L_COUNTRY …" node of the metadata editor.

Change the properties of list items that are part of a unique row key, such as the List element ID (see Figure 20).
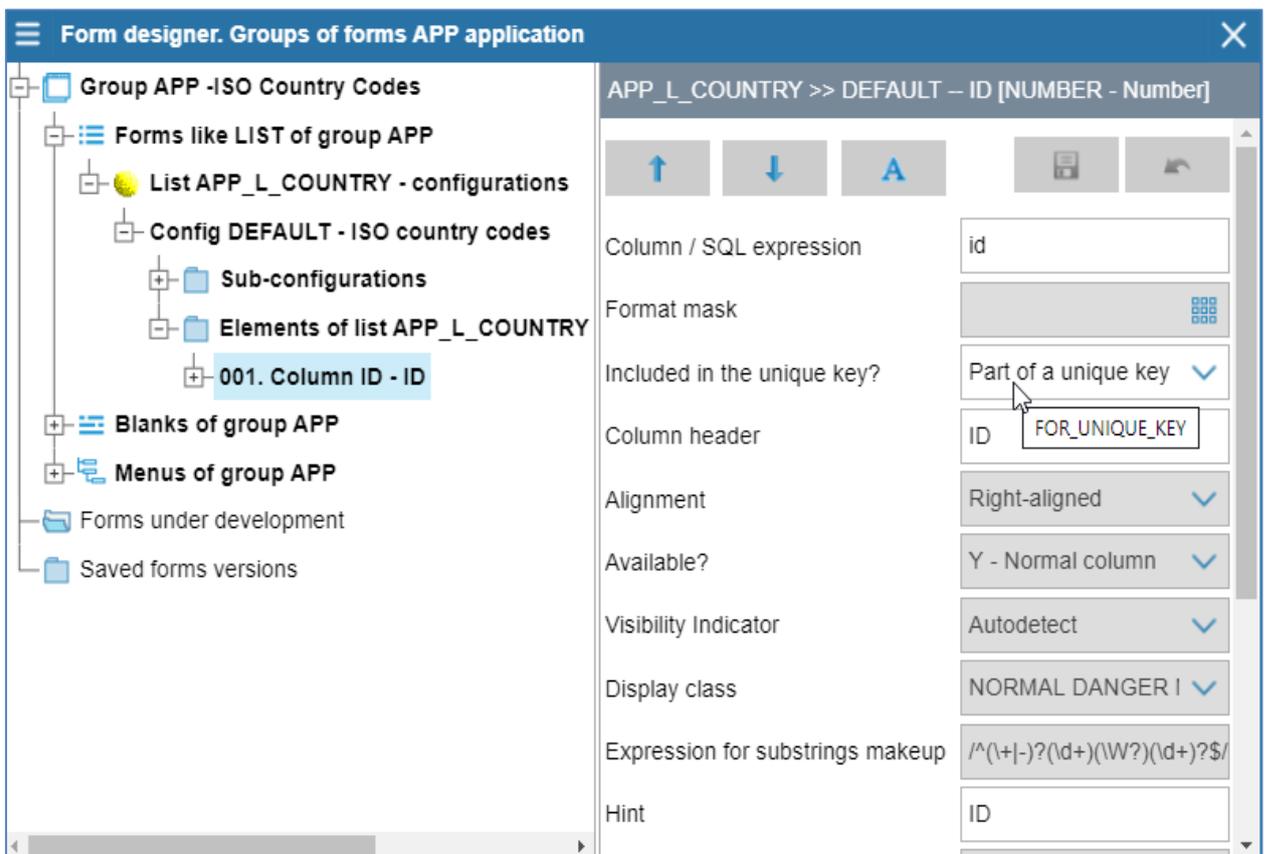
Figure 20. List element properties - Column that is included in a unique row key

Similarly, add all the elements of the List form with the Column type that are offered for selection and correspond to the fields of the table in the database (see figure. 21).
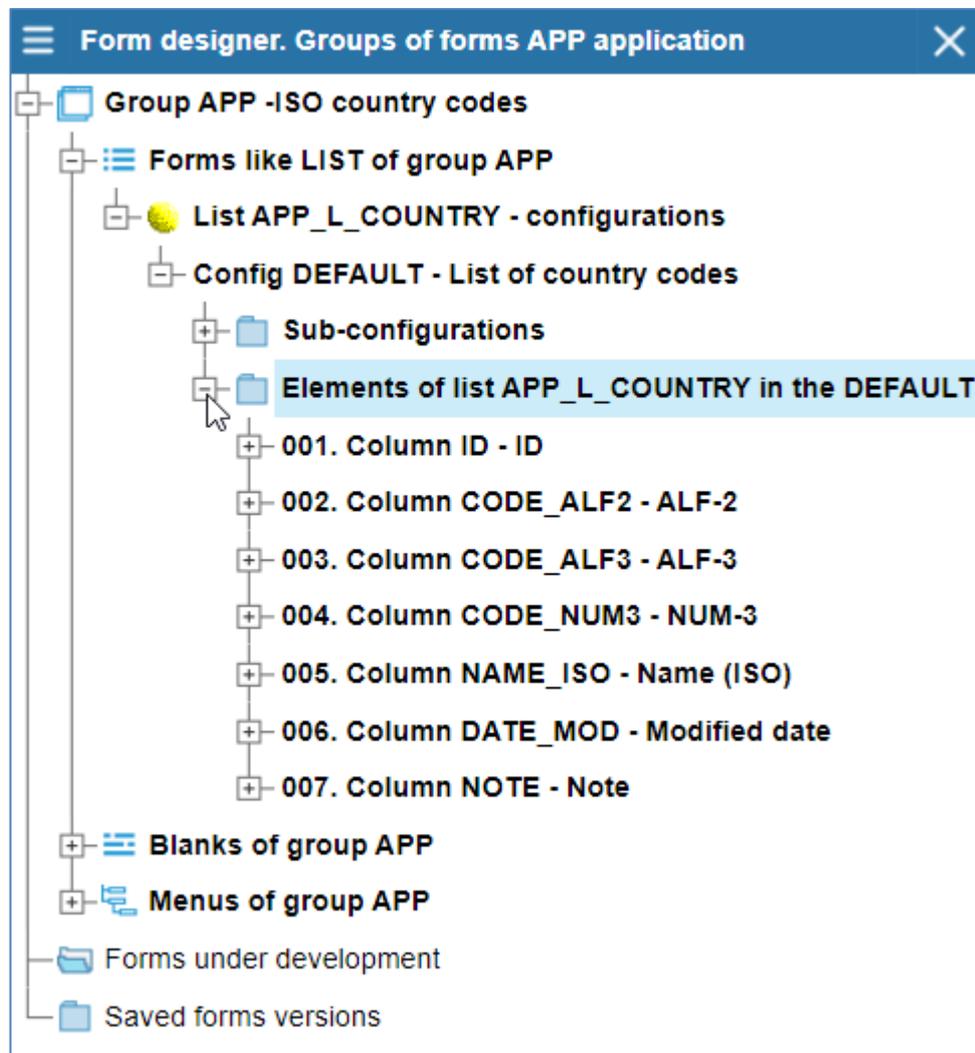
Figure 21. Added List elements with Column type

The value of the properties of the added columns of the List is left by default, except for the ID Column element, which is included in the unique row key (see Figure 19). Verify that the List form works (see chapter 8.4).

8.3.2.   Create a List element - Action

The form with the List type is intended mainly for viewing data in the database, calling other forms and procedures to perform the Actions: Add, Edit, Delete.

Creating List elements with the Action type is performed in the form editor from the expanded node "Elements of list APP_L_COUNTRY …" from the menu item "+Add element" by right mouse click (see chapter 8.3 and Figure 18).

8.3.2.1. Create List element - Action Delete

Creating List element - Action Delete is performed in the "Create a new List element" form, where you must select the following field values from the suggested lists (see figure 22):

- Element type: Action

- Element code: DEL

- Role: DELETE.

Figure 22. Creating a new List element - Action Delete

To save the element settings, click the "Save" button. If there are no errors, the Action Delete will appear in the list of elements in this form in the metadata editor (see Figure 23).
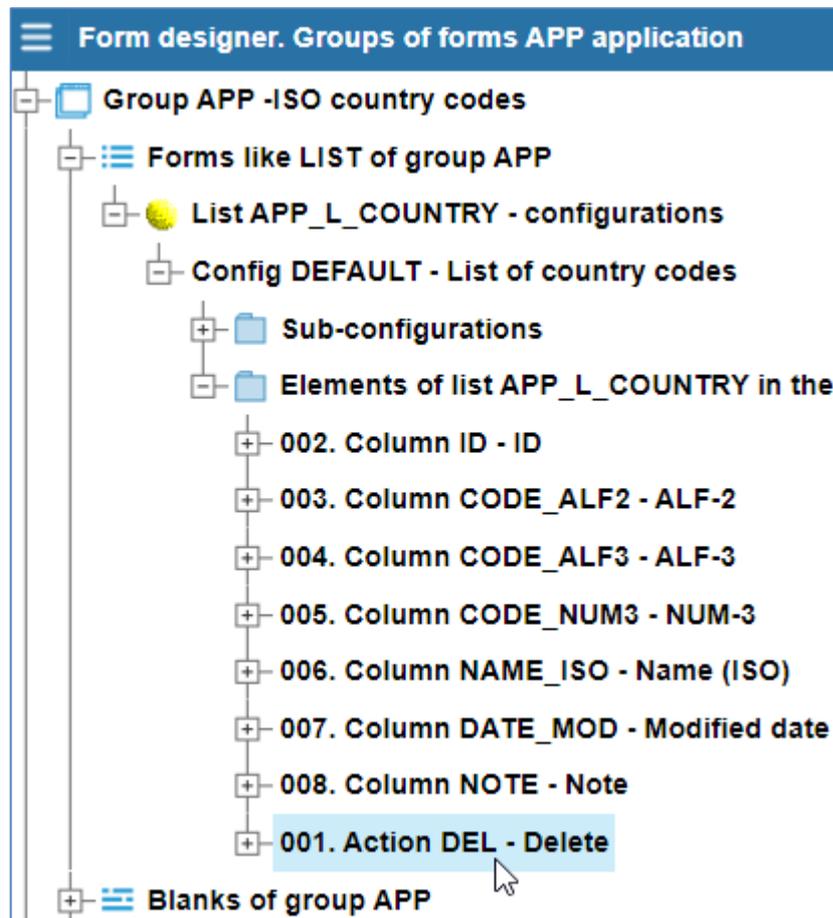
Figure 23. A new List element has been created - Action Delete

Edit the properties of the List element - Action Delete by clicking on its branch "001.Action Del - Delete" and enter SQL request to delete the entry in the LUI syntax (see Figure 24):

SQL:delete from app_t_country where ID={F:ID}
where,
       SQL: - execute string as SQL query
       {F:ID} - get the value of ID parameter.

To save element settings - Action Delete , click the "Save" button.

Figure 24. List element properties - Action Delete

8.3.2.2. Create List element - Action Add

Creating List element - Action Add is performed in the "Create a new List element" form (call see chapter 8.3 and figure 18) by selecting the following field values from the proposed lists (see Figure 25):
- Element type: Action
- Element code: ADD
- Role: ADD.

Figure 25. Creating a new List element – Action Add

To save the element settings, click the "Save" button. If there are no errors, the Action Add is appears in the List items in the metadata editor.

Figure 26. List element properties - Action Add

In the properties of the List element - Action Add the value of the parameter "Called object" is left empty for now, and fill in later, when we create a form with the type Blank for data entry (see Figure 26).

8.3.2.3.   List - Action Edit and Output parameter

To create a new List element - Action Edit, in the form "Create a new List element" (call see chapter 8.3 Fig.18) select from the list the following field values (see Figure 27):

- Element type: Action
- Element code: EDIT
- Role: DBLCLICK.

Figure 27. Creating a new List element – Action Edit

To save the element settings, click the "Save" button. If there are no errors, the Action Edit appears in the List items in the metadata editor.

Figure 28. List element properties - Action Edit

In the properties of the List element - Action Edit the value of the parameter "Called object" is left empty for now, and fill in later, when we create a form with the type Blank for data entry (see Figure 28).

In the current List form, it is assumed that for the Add and Edit actions, a call to another form with the Blank type will be used, to which the attribute of the current action will be passed - a parameter with a unique identifier of the row for changing it.

If the parameter contains a non-empty value, the Action Edit is performed, otherwise the Action Add is performed. To do this, create an **Output parameter in the List element – Action Edit**.

To call the form for creating the Output parameter of the Action Edit element, expand the node "003.Action EDIT-edit: Parameters ...", right-click and select the menu item "+ Add parameter " (see Figure 29).
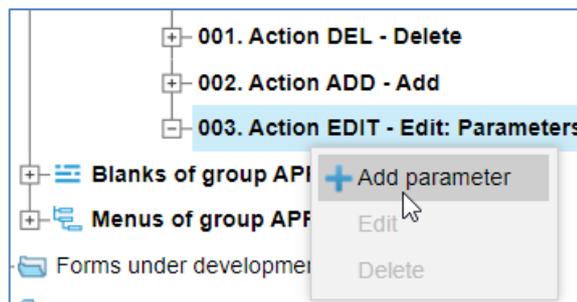
Figure 29. Call to create an Output parameter for the element - Action Edit

Creating an Output parameter of the Action Edit element is performed in the form "Creating a new List parameter for EDIT" by filling values in the fields (see Figure 30): Element code (ID), Name (ID).
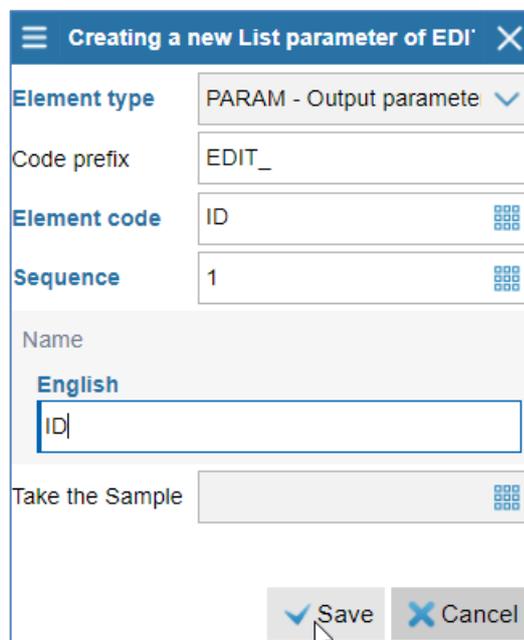
Figure 30. Create an Output parameter for the element – Action Edit

Save the Output parameter by clicking on the "Save" button. If there are no errors, the Input parameter will appear in the metadata editor in the branch "003. Action EDIT – Edit: Parameters …".

Enter an expression to calculate the parameter values of the properties "Output parameter EDIT_ID – ID" by filling in the fields (see figure 31):
- Parameter code:　　　　PCODE
- Default value:　　　　　{F:ID}
- Comments:　　　　　　 Unique identifier of the row.
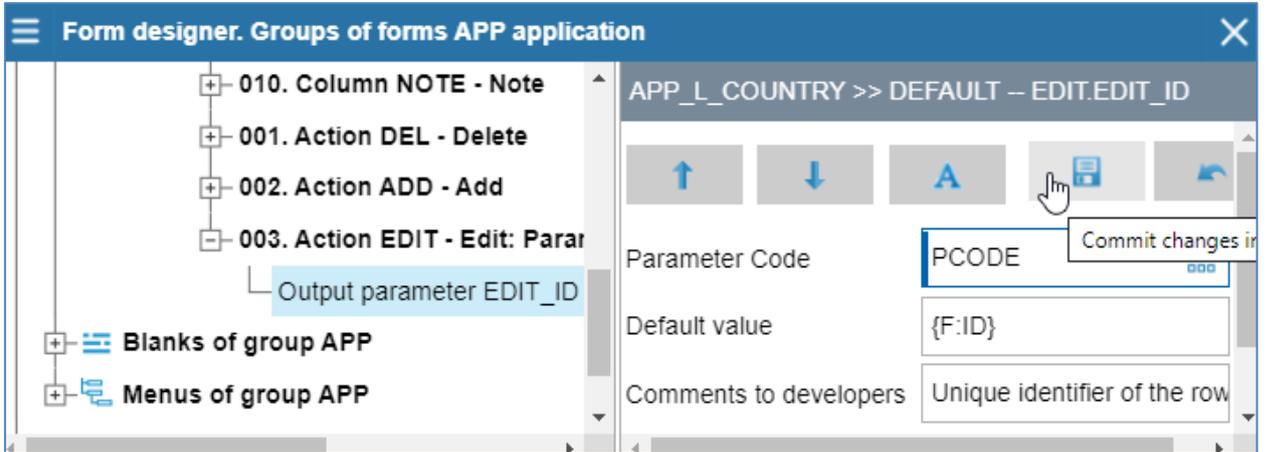
Figure 31. Properties of Output parameter of the element - Action Edit

After entering the values, save the properties of the Output parameter by clicking the "Save" button.

8.4. Check the operation of the List form

Checking the operation of the created List is performed in the form editor on the node "Config DEFAULT – List of country codes", where right-click the menu and select "Run" (see Fig.32), then start and display the List form (see Fig.33).
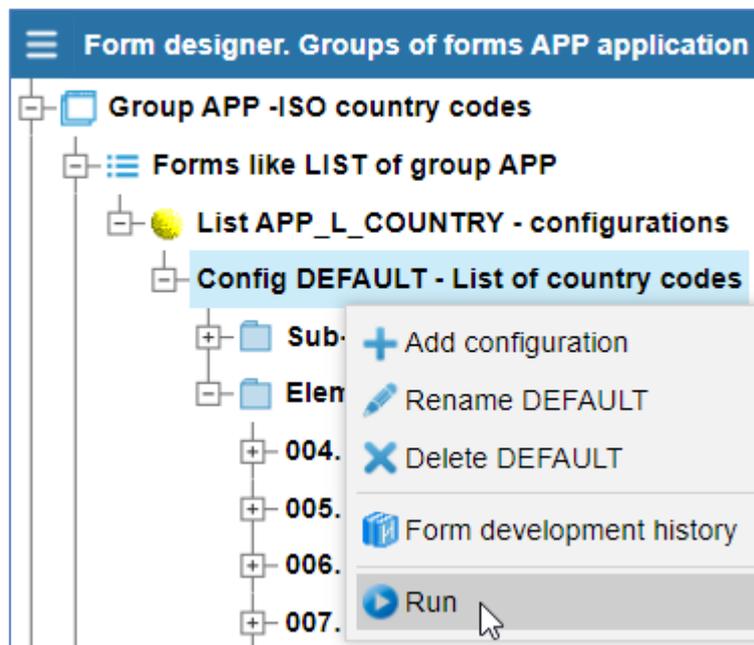
Figure 32. Call the execution of the List form to check from the metadata editor

Figure 33. Checking the operation of the created List in the form editor

Check the work and capabilities of the created List form at this stage (without the form for adding and editing records):

- Display and scroll through the list of records using the mouse and keyboard

- Presence of icons at the top of the List form: Add new record, Edit record, Delete row

- The presence in the menu, which is called from the current row of the record by right-clicking, the following items to call actions: Add, Edit, Delete.

Test the system-wide capabilities of the List form:

- Setup sorting of records by left click on the column header, where you can specify several columns in turn

- Setting the display of columns by right clicking on the column header

- Entering a query to search and select records by clicking on the magnifying glass icon "Enter query" at the top of the form

- Execution of the entered query by clicking on the icon "Execute query" at the top of the form

- Counting the number of rows in the list from the menu at the top left of the form, select item "Number of rows"

- Output records to a printable file from the top left menu, select item "Version for print".

9. Create a Blank form

Creating a form with the Blank type is performed by the Developer in the metadata editor in the following sequence:

- Login to LUI as Developer (see Figure 14)

- Create a new Blank form

- Blank configuration and SQL query input

- Create List elements: Output parameter, Field, Action (Add, Save)

- Checking the operation of the new Blank.

9.1. Create a new Blank form

To create a new Blank in the form editor, expand the node "Forms like Blank of group APP" by clicking on the " + " icon in the left part of the menu item. Right-click and execute the action "+ Add" (see Figure 34).
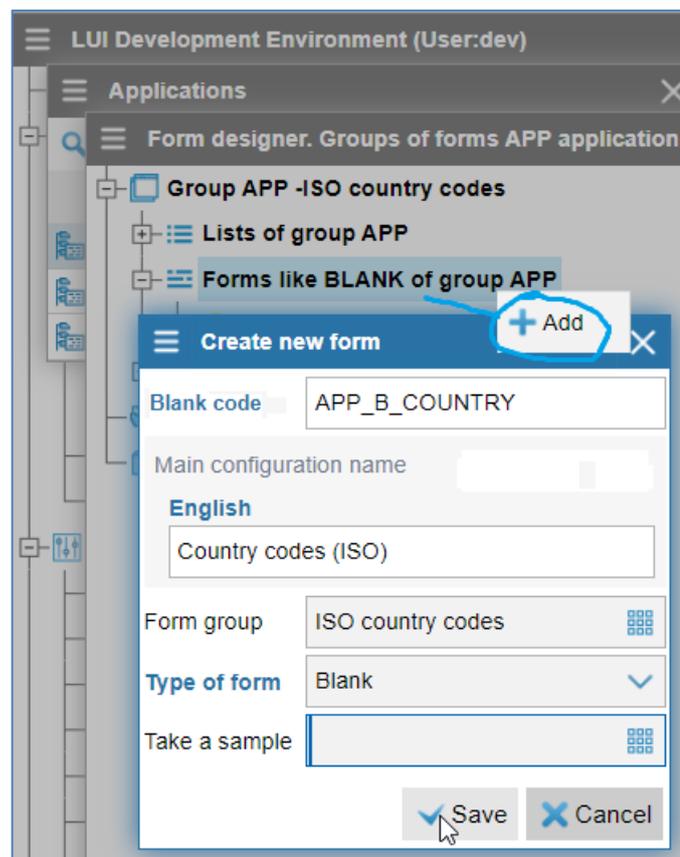
Figure 34. Create a new Blank form

In the form "Create new form", enter the value of the fields "Blank code" and "Main configuration Name", and click on the "Save" button (see Figure 34).

If there are no errors, the Blank will appear in the node "Forms like Blank of group APP" (see Figure 35).

9.2. Blank configuration and SQL query input

Blank configuration and SQL-query input is performed in the form that appears on the right, if you stand on the node " Configuration DEFAULT-County codes (ISO)". In the "Select fields" specify a SQL query to locate the record and determine the composition of the fields in the form, check the header of the form and click "Save" to save the settings (see Figure 35).
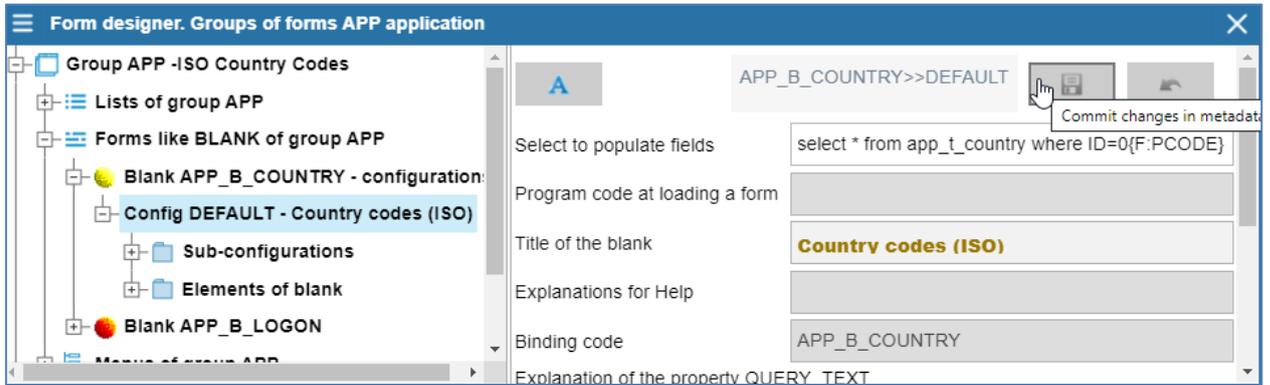
Figure 35. Blank configuration and SQL query input

Specify the SQL query to get the list and determine the composition of the table fields, the parameters of which are given in the Appendix A section of this document:

*SELECT * FROM APP_T_COUNTRY WHERE ID=0{F:PCODE}*

where,

PCODE - the Input parameter of the Blank (ID record), which is defined earlier in the List form when creating the Edit action as an Output parameter (see p.8.3.2.3 and Fig. 31).

0{F:PCODE} - this is the LUI syntax for getting the value of the PCODE parameter

0 - to exclude the error, when the value of {F:PCODE} is NULL.

If the PCODE value is not specified (NULL or 0), a new record is added, otherwise the current record is edited and saved.

9.3. Create Blank elements

This document describes the following elements of the Blank: Input parameter, Field, Action (Add, Edit, Cancel).

To create Blank elements in the form editor, expand the node "Elements of blank APP_B_COUNTRY …" by clicking the + icon on the left side of the menu item. Then right-click and execute the action "+Add element " (see Fig. 36).
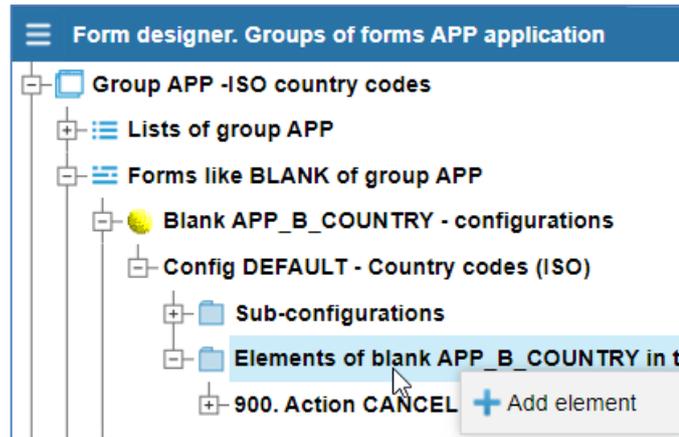
Figure 36. Call the form to add Blank elements

9.3.1.   Create Blank element - Input parameter

To create a new Blank element - Input parameter, in the form "Create a new Blank element" enter the following field values (see Figure 37):
- Element type: PARAM - Input parameter
- Element code: PCODE
- Name: Row ID.
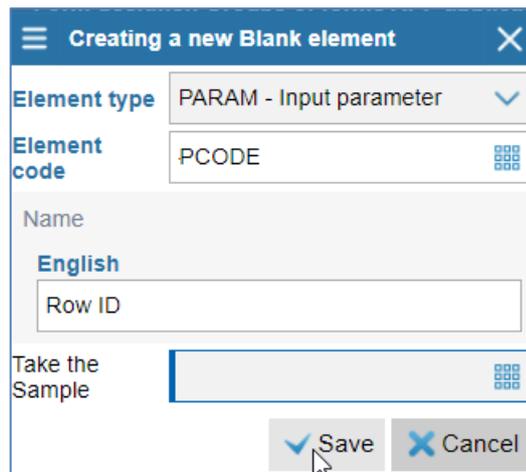
Figure 37. Create Blank element - Input parameter

Save the entered parameters by pressing the "Save" button. If there are no errors, the Input parameter appears in the list of elements of this form in the metadata editor. You do not have to enter the properties of the Input parameter.

9.3.2. Create Blank element - Field

To create a new Blank element - Field, in the form "Create a new Blank element" (call see p.9.3, Fig.36) enter the following field values (see Figure 37):
- Element type: ITEM - Field
- Element code: CODE_NUM (select from list of values)
- Name: Digital code, 3 digits.

Save the entered parameters by pressing the "Save" button. If there are no errors, the Input parameter appears in the list of elements of this form in the metadata editor.
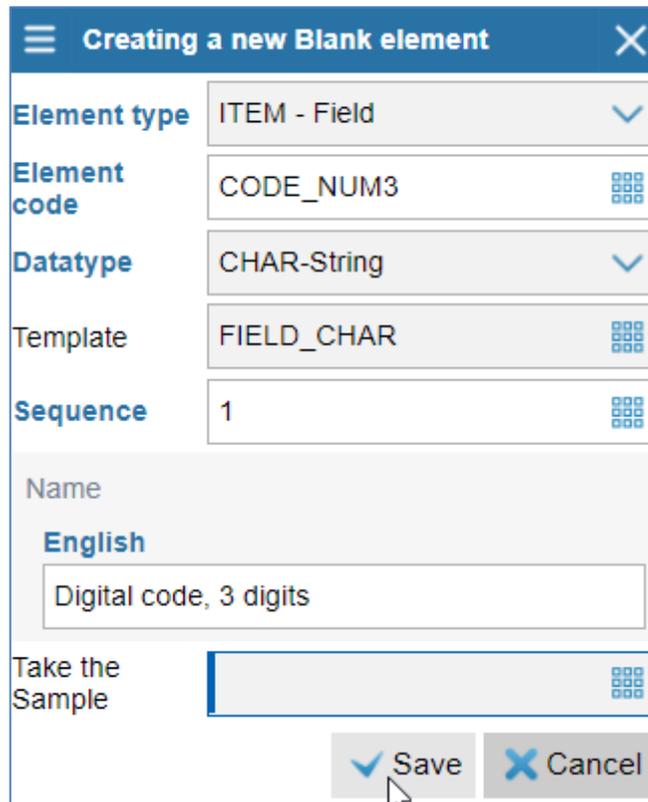
Figure 38. Create Blank element - Field

Similarly, add all the Blank elements with the Field type that are offered for selection and correspond to the fields of the table in the database (see Figure 39).

After you have entered all the fields of the Blank, change the properties of the two fields to make them unchanged, since they cannot be edited:
- "006. Field ID – Row ID" (see Figure 40):
   o Default Value (expression): {F:PCODE}
   o Change method: Read only (N)
- "007. Field DATE_MOD – Edit date":
   o Change method: Read only (N)
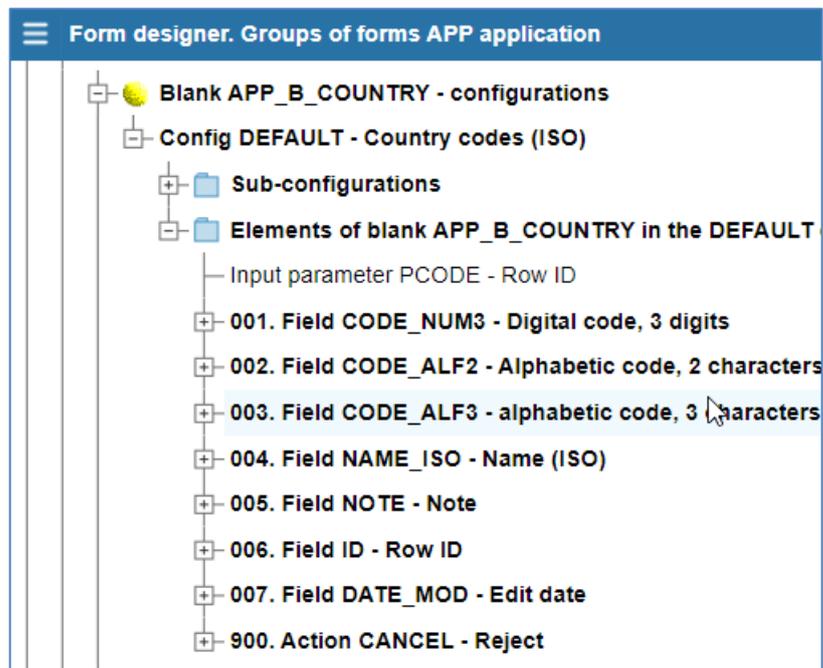
Figure 39. Added Blank elements: Input parameter and Field
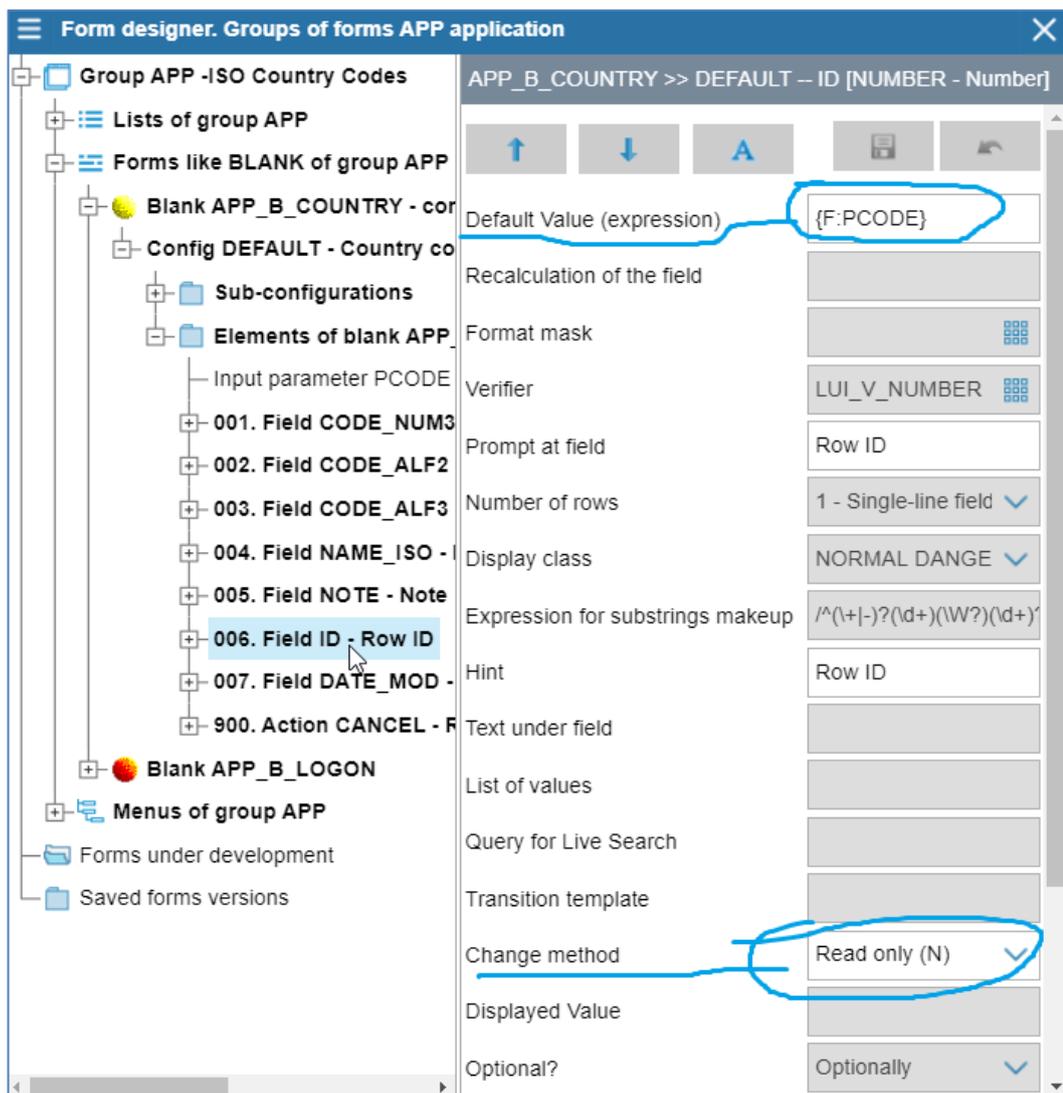
Figure 40. Properties of the Blank element- "006. Field ID. Row ID"

9.3.3. Create Blank element - Action

The form of type Blank is intended mainly for changing data in the database and there should be Actions: Add, Edit, Cancel.

Action Cancel is created automatically when you add a new Blank, so you must to create the Actions Add and Edit.

To create a new Blank element in the form editor, expand the node "Elements of blank APP_B_COUNTRY …" by clicking the + icon on the left side of the menu item. Then right-click and execute the action "+Add element " (see Figure 36 in chapter 9.3).

9.3.3.1. Create Blank element – Action Add

To create a new Blank element – Action Add, in the form "Creating a new Blank element" enter the following field values (see Figure 41):
- Element type: ACTION
- Element code: ADD
- Name: Add
- Role: DBLCLICK - Default action (ENTER).

Save the entered parameters by pressing the "Save" button. If there are no errors, the Input parameter appears in the list of elements of this form in the metadata editor.

Figure 41. Create Blank element – Action Add

Edit the properties of the Blank element - Action Add by staying on its node and filling in the property fields with the specified values (see Figure 42):

- Action is active?   - {F:PCODE}Y   - if the PCODE parameter has no value (empty, NULL), the action is active and can be selected from the menu or by the button

- Button activation?   - {F:PCODE}Y - if the PCODE has no value (empty, NULL), the button is displayed in the Blank

- Program code before action - (in the multirow editor):

```
PLPGSQL:declare t app_t_country%rowtype;
begin
 t.code_num3 := {F2SQL:CODE_NUM3};
 t.code_alf2 := {F2SQL:CODE_ALF2};
 t.code_alf3 := {F2SQL:CODE_ALF3};
 t.note      := {F2SQL:NOTE};
 t.name_iso  := {F2SQL:NAME_ISO};
   insert into
   app_t_country(code_num3,code_alf2,code_alf3,name_iso,note)
       values (t.code_num3,t.code_alf2,t.code_alf3,t.name_iso,t.note);
end;
```

where, PLPGSQL: sign of the program on PL/pgSQL - add fields from the Blank to the database

- Program code after action   - SQL:commit

Save the item properties settings by clicking the "Save" button.
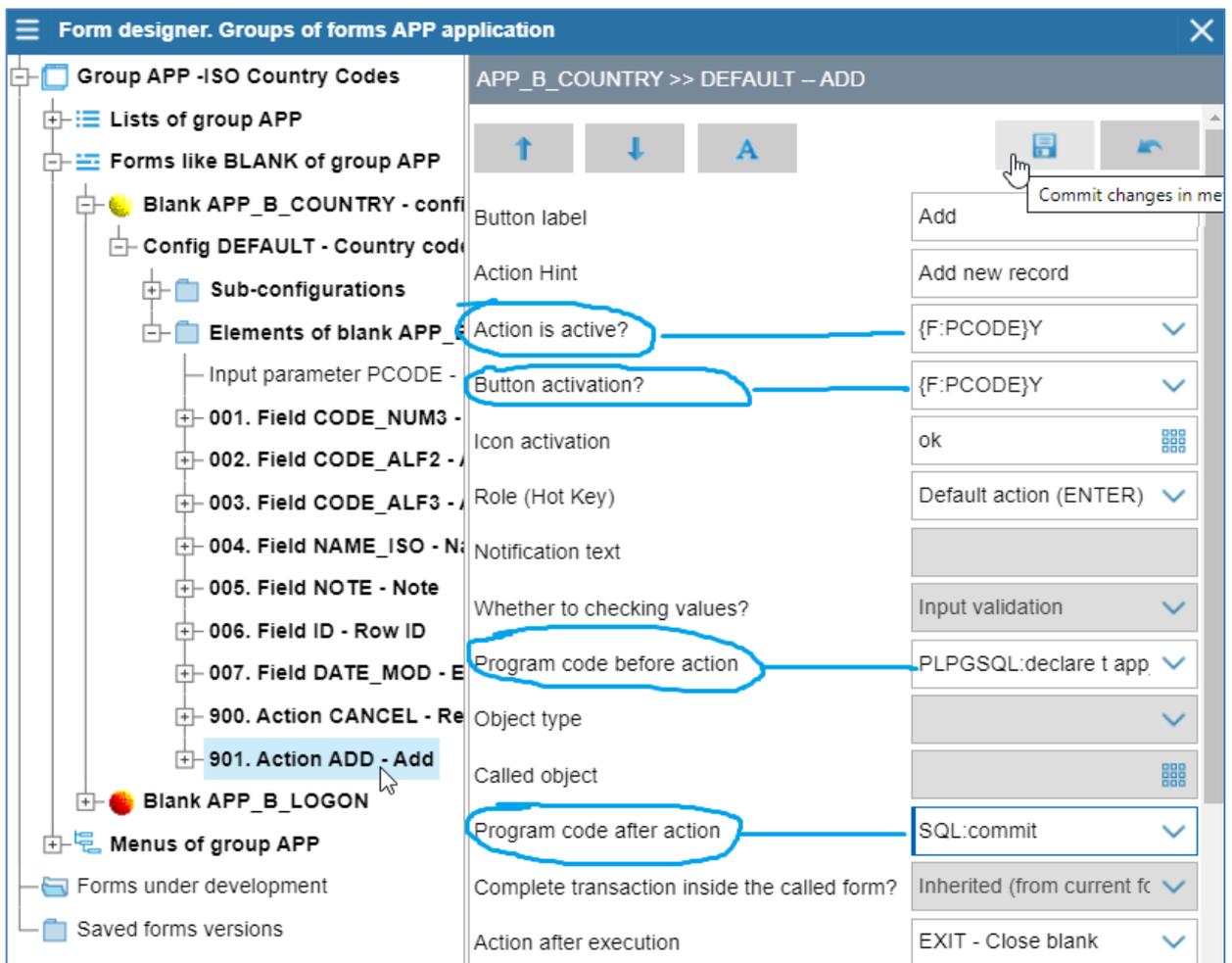
Figure 42. Properties of the Blank element- "901. Action ADD"

9.3.3.2. Create Blank element – Action Save

To create a new Blank element – Action Save, in the form "Creating a new Blank element" (call see Figure 36 in chapter 9.3) enter the following field values (see Figure 43):

- Element type: ACTION
- Element code: OK (Save)
- Name: Save
- Role: DBLCLICK - Default action (ENTER).

Save the entered parameters by pressing the "Save" button. If there are no errors, the Input parameter appears in the list of elements of this form in the metadata editor.

Figure 43. Create Blank element – Action Save

Figure 44. Properties of the Blank element- "902. Action OK - Save"

Edit the properties of the Blank element - Action Save by staying on its node and filling in the property fields with the specified values (see Figure 44):

- Action is active?

```
{JS:if ("{F:PCODE}" =="")"N"; else "Y";}
```

where, JS: - a sign of a JavaScript program - the action Save is active and can be performed from the action menu or button if the PCODE parameter does not have a non-empty value

- Program code before action - (in the multirow editor):

```
PLPGSQL:declare t app_t_country%rowtype;
begin
 t.code_num3 := {F2SQL:CODE_NUM3};   t.code_alf2 := {F2SQL:CODE_ALF2};
 t.code_alf3 := {F2SQL:CODE_ALF3};   t.note      := {F2SQL:NOTE};
 t.name_iso  := {F2SQL:NAME_ISO};
 update app_t_country set
        code_num3 = t.code_num3,  code_alf2 = t.code_alf2,
        code_alf3 = t.code_alf3,  name_iso  = t.name_iso,
        note      = t.note
          where id= {F2SQL:PCODE};
      end;
```

where, PLPGSQL: sign of the program on PL/pgSQL - update to the database fields from the Blank

- Program code after action - SQL:commit

Save the item properties settings by clicking the "Save" button (see Fig. 44).

The creation of the Blank is completed and now its code can be specified in the form of a List for calling the Add and Save actions (see chapter 8. 3. 2. 2 and Figure 26, chapter. 8. 3. 2. 3 and Figure 28), then create menu items to call this List form.

10. Create Menu

Create a Menu for the Application with the following hierarchy of items:

APP - APPLICATION MENU

-> APP – Directories

-> APP - Directory Code Countries -

to call form APP_L_COUNTRY

The previously created List form with APP_L_COUNTRY code (see chapter 8) will be called from the menu item "APP - Directory Code Countries ".

The Application Menu items are created by the Architect.

10.1. Create the Application's root Menu item

To call the form for creating the root Menu item of the Application, expand the node "Elements of menu APP_M_MAIN_MENU …", then right-click the mouse to call and then perform the action "Add element" (see Figure 45).
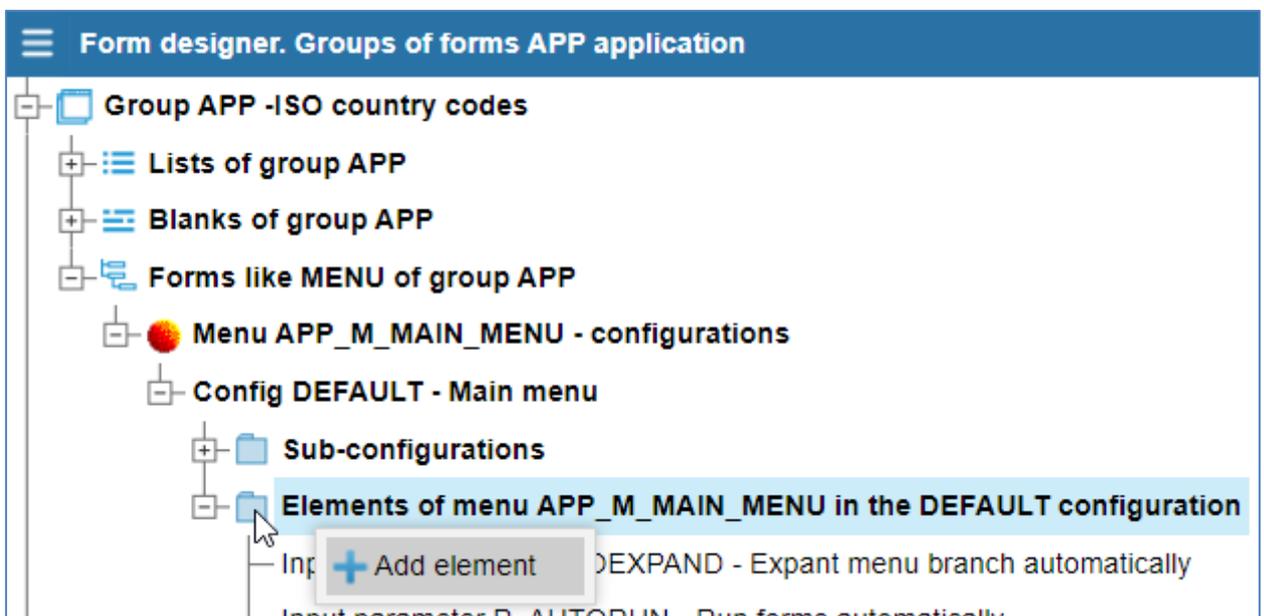
Figure 45. Call up a form to add a root Menu item

To create a new Menu element in the form "Creating a new Menu element" enter the following field values (see Figure 46):
- Element type: ACTION
- Element code: APP   (Enter manually)
- Name: APP - APPLICATION MENU   (Enter manually)

Save the entered parameters by pressing the "Save" button. If there are no errors, the Input parameter appears in the list of elements of this Menu in the metadata editor.
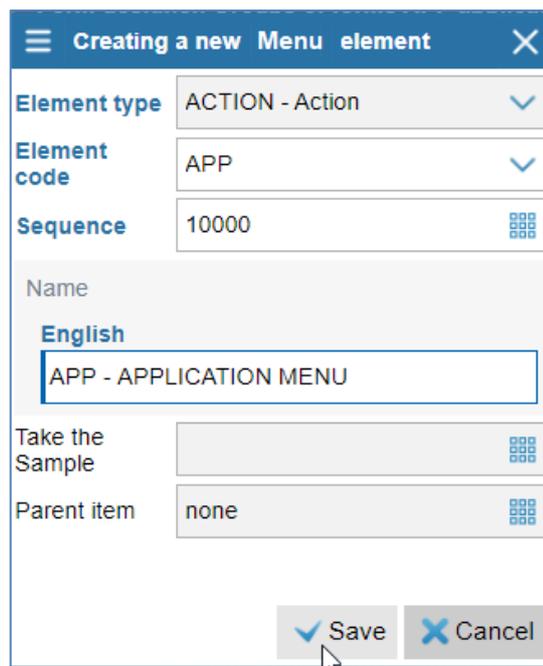
Figure 46. Create the Application's root Menu item

Edit the properties of the root Menu element by specifying the value of the "Menu item text" field and other (see Fig.47).
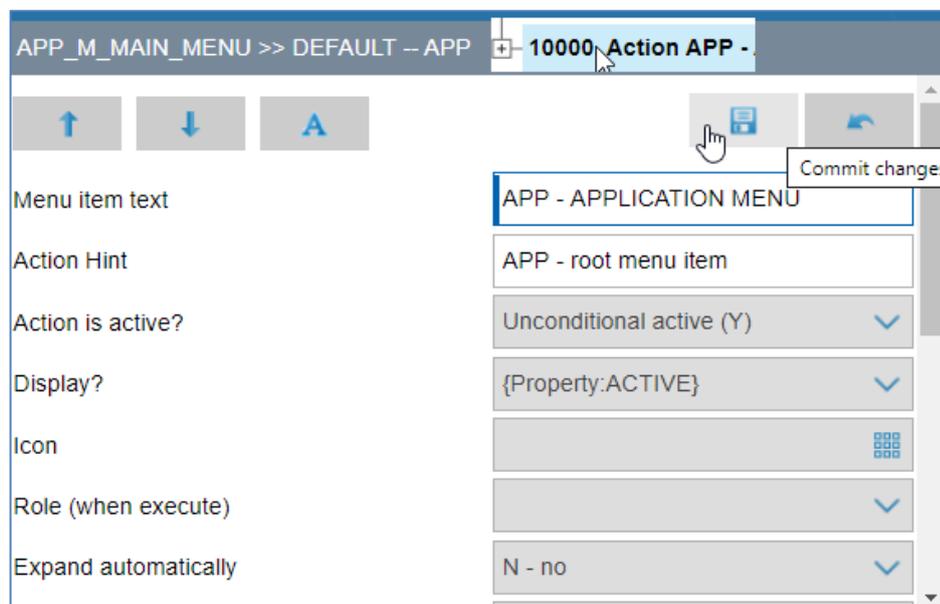
Figure 47. The properties root element the Application menu

10.2. Create level 2 Menu item

Creating a Menu item of the 2nd level is performed from the expanded node of the root Menu item, where right-click to call and then perform the action "Add element", after which the Form "Creating a new Menu item" will appear (see Figure 48).
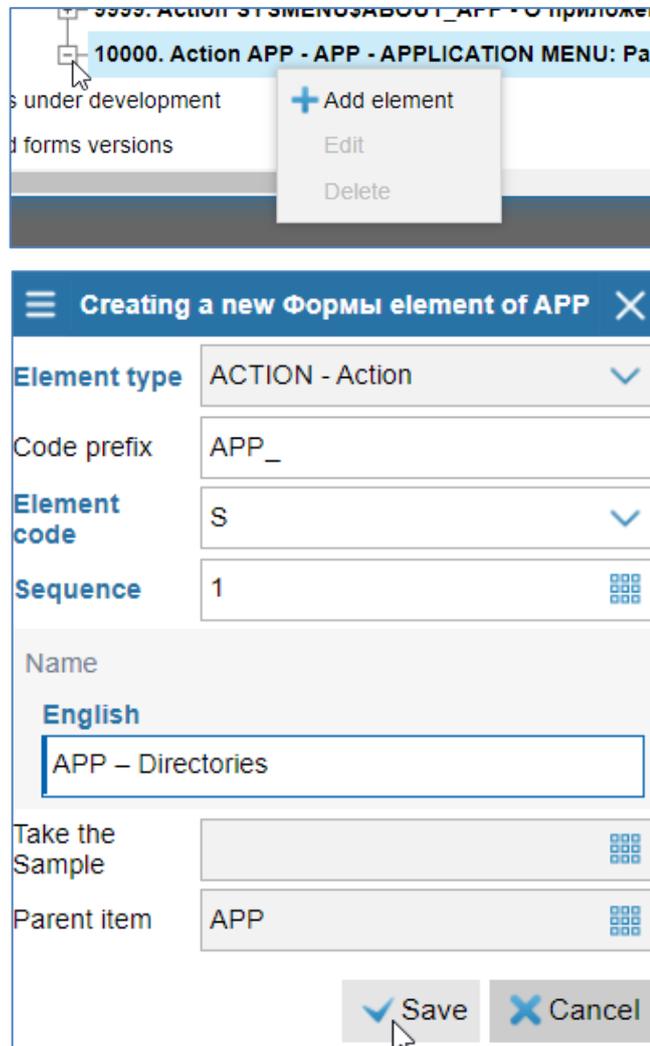
Figure 48. Creating a level 2 Menu item

To create a Menu element in the form "Creating a new Menu element" enter the following field values (see Figure 46):
- Element type: ACTION
- Element code: S (Enter manually)
- Name: APP - Directories (Enter manually)

Save the entered parameters by pressing the "Save" button. If there are no errors, the Input parameter appears in the list of elements of this Menu in the metadata editor.

Edit the properties of the Menu item of the 2nd level, specifying the value of the "Menu item text" field.

10.3. Create level 3 Menu item

Creating a Menu item of the 3rd level is performed from the expanded node of the menu item of the 2nd level, where the right mouse button is called and then the action "Add element" is executed, after which the form "Creating a new Menu element" appears (see figure 49).
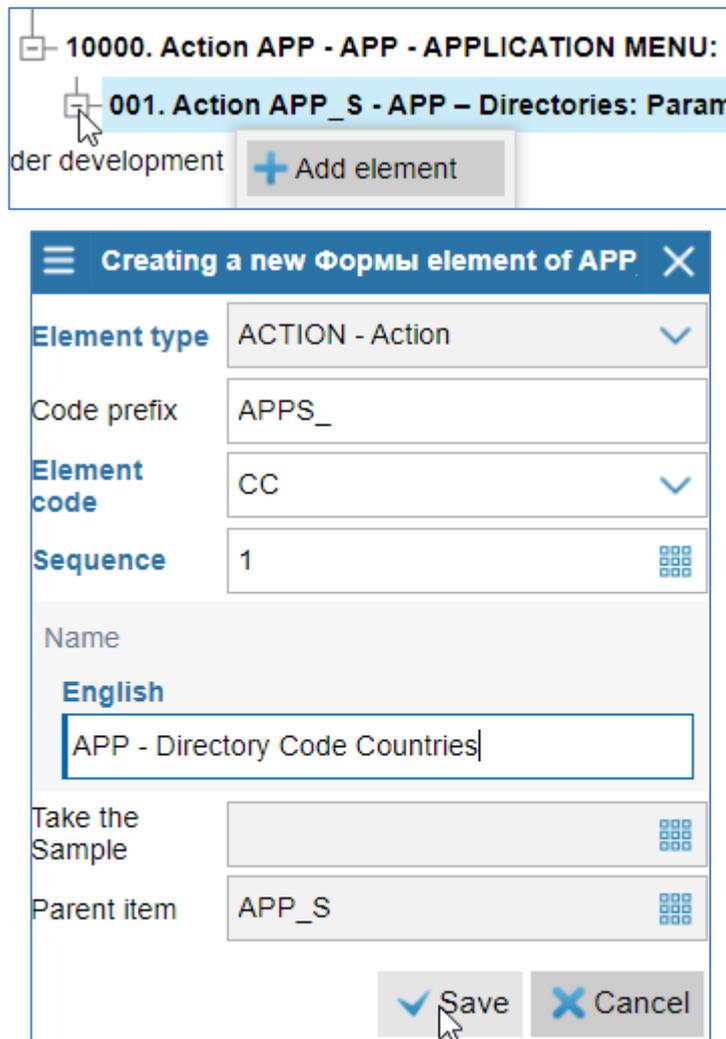
Figure 49. Create a menu item of the 3rd level

To create a Menu element of the 3rd level in the form "Creating a new Menu element" enter the following field values (see Figure 49):

- Element type: ACTION
- Element code: CC   (Enter manually)
- Name: APP – Directory Code Countries   (Enter manually)

Save the entered parameters by pressing the "Save" button.

If there are no errors, the Input parameter appears in the list of elements of this Menu in the metadata editor (see Figure 50).

Edit the properties of the Menu item of the 3rd level, specifying the value of the fields (see Figure 50):

- Menu item text: APP – Directory Code Countries
- Object type   :   LIST
- Called object   :   APP_L_COUNTRY

Figure 50. Property of the 3rd level Menu item with List call

11. Check the operation of the Application

To check the work created in LUI of a set of forms (Menu, List, Blank) run the Application from LUI, from the menu "Applications" open the form with a list of applications and to the left of the row click on the icon "Run the application system in a new browser tab" (see Figure 51).
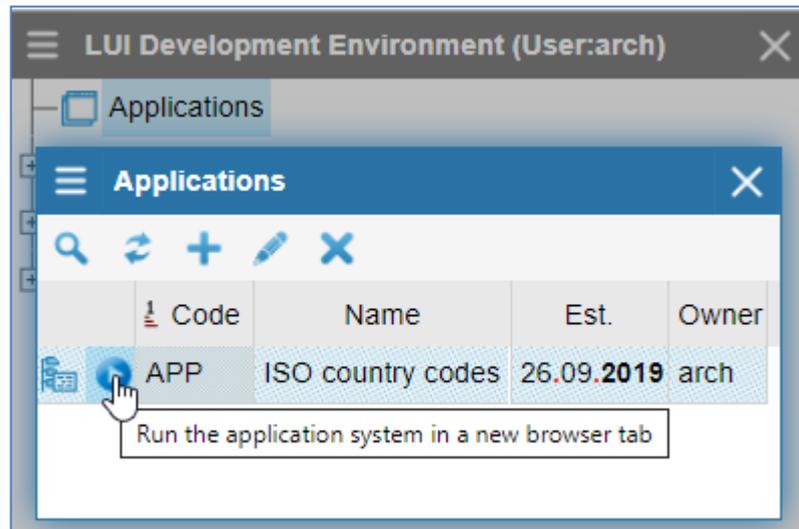
Figure 51. Running an Application from LUI

After the initial application login form appears in a new browser tab, perform login authorization, where specify the name and password of the user with the rights of the Architect and click on the "Login" button (see Figure 52).
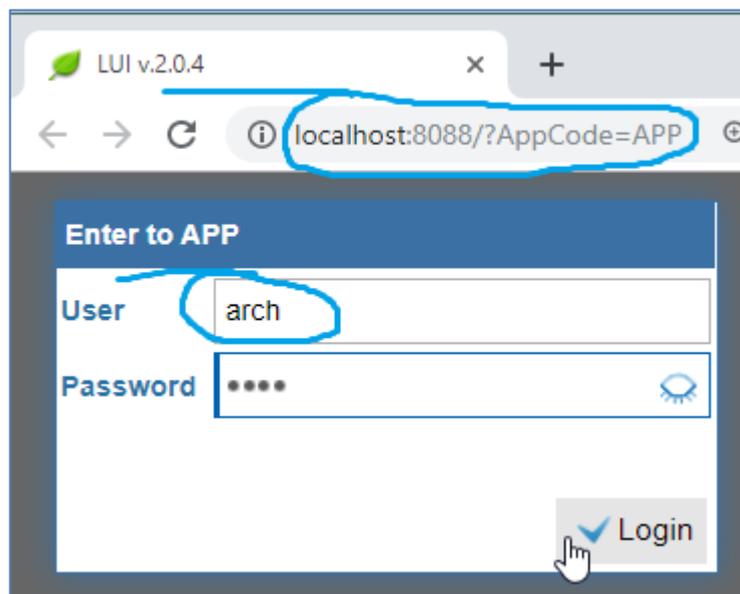
Figure 52. Start URL and Application form for login

After successful login authorization, the main application Menu will open. Open the List from the "APP – Directory Countries Codes" menu item, where you can check the operation of the Application (see Figure 53).
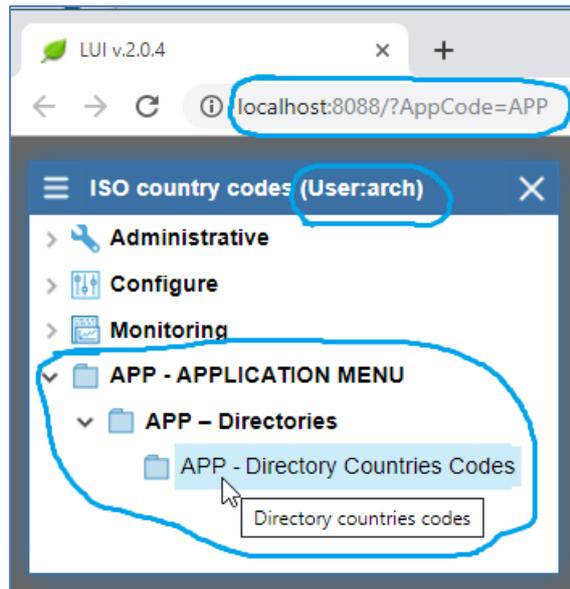
Figure 53. The main Menu of the Application with code APP

Check the operation of the form with the list of country code (see Fig.54):

- Presence and activity of icons and menus for actions: Add, Edit, Delete
- Performing the Delete action
- Performing Add and Edit actions.
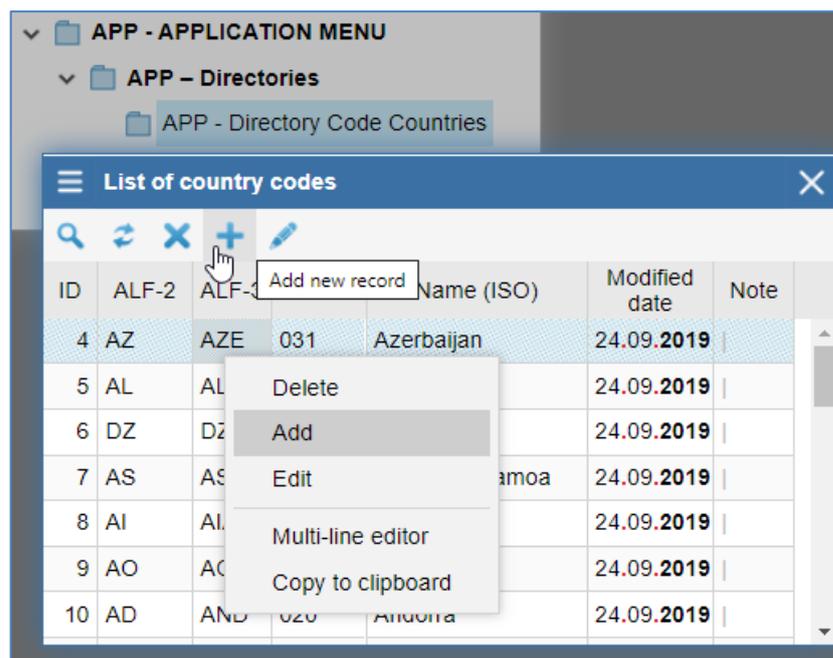
Figure 54. List called from the menu item "Directory countries codes"

The data editing Form should have different button names depending on the actions (see Figure 52):

- For the action Edit - Cancel and Save buttons

- For the action Add - Cancel and Add buttons

- "Row ID" and "Edit date" fields are viewable only, without the ability to enter or modify data.

Figure 55. Data entry form for actions Edit and Add

For more detailed configuration of access rights and Application forms, you need to read the document in the file " LUI_Guide204.pdf", located by URL http://lui.fors.com/LUI_Guide204.pdf: "Live universal Interface application development and execution Environment". Programmer manual».

Appendix A. Description of the database table

Creating screen forms in LUI is performed on the example of the following database table with country codes.

```sql
-- Drop table

-- DROP TABLE public.app_t_country;

CREATE TABLE public.app_t_country (
	id serial NOT NULL,
	code_num3 varchar(3) NOT NULL,
	code_alf2 varchar(2) NOT NULL,
	code_alf3 varchar(3) NOT NULL,
	name_iso varchar(255) NULL,
	date_mod timestamp NOT NULL DEFAULT now(),
	note text NULL,
	CONSTRAINT app_t_country_code_alf2_key UNIQUE (code_alf2),
	CONSTRAINT app_t_country_code_alf3_key UNIQUE (code_alf3),
	CONSTRAINT app_t_country_code_num3_key UNIQUE (code_num3),
	CONSTRAINT app_t_country_name_iso_key UNIQUE (name_iso),
	CONSTRAINT app_t_country_pkey PRIMARY KEY (id)
);

-- Permissions

GRANT SELECT ON TABLE public.app_t_country TO lui_common;
GRANT ALL ON TABLE public.app_t_country TO lui_adm;
GRANT ALL ON TABLE public.app_t_country TO lui_des;
GRANT ALL ON TABLE public.app_t_country TO lui_dev;


GRANT ALL ON SEQUENCE public.app_t_country_id_seq TO lui_adm;
GRANT ALL ON SEQUENCE public.app_t_country_id_seq TO lui_des;

GRANT ALL ON SEQUENCE public.app_t_country_id_seq TO lui_dev;

-- End DDL
-- Load app_t_country - country codes
--
insert into app_t_country(name_iso,code_alf2,code_alf3,code_num3,note) values
('Abkhazia','AB','ABH','895',''),
('Australia','AU','AUS','036',''),
('Austria','AT','AUT','040',''),
('Azerbaijan','AZ','AZE','031',''),
('Albania','AL','ALB','008',''),
('Algeria','DZ','DZA','012',''),
('American Samoa','AS','ASM','016',''),
('Anguilla','AI','AIA','660',''),
('Angola','AO','AGO','024',''),
('Andorra','AD','AND','020',''),
('Antarctica','AQ','ATA','010',''),
('Antigua and Barbuda','AG','ATG','028',''),
('Argentina','AR','ARG','032',''),
('Armenia','AM','ARM','051',''),
('Aruba','AW','ABW','533',''),
('Afghanistan','AF','AFG','004',''),
('Bahamas','BS','BHS','044',''),
('Bangladesh','BD','BGD','050',''),
('Barbados','BB','BRB','052',''),
('Bahrain','BH','BHR','048',''),
('Belarus','BY','BLR','112',''),
```

```
('Belize','BZ','BLZ','084',''),
('Belgium','BE','BEL','056',''),
('Benin','BJ','BEN','204',''),
('Bermuda','BM','BMU','060',''),
('Bulgaria','BG','BGR','100',''),
('Bolivia, plurinational state of','BO','BOL','068',''),
('Bonaire, Sint Eustatius and Saba','BQ','BES','535',''),
('Bosnia and Herzegovina','BA','BIH','070',''),
('Botswana','BW','BWA','072',''),
('Brazil','BR','BRA','076',''),
('British Indian Ocean Territory','IO','IOT','086',''),
('Brunei Darussalam','BN','BRN','096',''),
('Burkina Faso','BF','BFA','854',''),
('Burundi','BI','BDI','108',''),
('Bhutan','BT','BTN','064',''),
('Vanuatu','VU','VUT','548',''),
('Hungary','HU','HUN','348',''),
('Venezuela','VE','VEN','862',''),
('Virgin Islands, British','VG','VGB','092',''),
('Virgin Islands, U.S.','VI','VIR','850',''),
('Vietnam','VN','VNM','704',''),
('Gabon','GA','GAB','266',''),
('Haiti','HT','HTI','332',''),
('Guyana','GY','GUY','328',''),
('Gambia','GM','GMB','270',''),
('Ghana','GH','GHA','288',''),
('Guadeloupe','GP','GLP','312',''),
('Guatemala','GT','GTM','320',''),
('Guinea','GN','GIN','324',''),
('Guinea-Bissau','GW','GNB','624',''),
('Germany','DE','DEU','276',''),
('Guernsey','GG','GGY','831',''),
('Gibraltar','GI','GIB','292',''),
('Honduras','HN','HND','340',''),
('Hong Kong','HK','HKG','344',''),
('Grenada','GD','GRD','308',''),
('Greenland','GL','GRL','304',''),
('Greece','GR','GRC','300',''),
('Georgia','GE','GEO','268',''),
('Guam','GU','GUM','316',''),
('Denmark','DK','DNK','208',''),
('Jersey','JE','JEY','832',''),
('Djibouti','DJ','DJI','262',''),
('Dominica','DM','DMA','212',''),
('Dominican Republic','DO','DOM','214',''),
('Egypt','EG','EGY','818',''),
('Zambia','ZM','ZMB','894',''),
('Western Sahara','EH','ESH','732',''),
('Zimbabwe','ZW','ZWE','716',''),
('Israel','IL','ISR','376',''),
('India','IN','IND','356',''),
('Indonesia','ID','IDN','360',''),
('Jordan','JO','JOR','400',''),
('Iraq','IQ','IRQ','368',''),
('Iran, Islamic Republic of','IR','IRN','364',''),
('Ireland','IE','IRL','372',''),
('Iceland','IS','ISL','352',''),
('Spain','ES','ESP','724',''),
('Italy','IT','ITA','380',''),
('Yemen','YE','YEM','887',''),
('Cape Verde','CV','CPV','132',''),
('Kazakhstan','KZ','KAZ','398',''),
```

```
('Cambodia','KH','KHM','116',''),
('Cameroon','CM','CMR','120',''),
('Canada','CA','CAN','124',''),
('Qatar','QA','QAT','634',''),
('Kenya','KE','KEN','404',''),
('Cyprus','CY','CYP','196',''),
('Kyrgyzstan','KG','KGZ','417',''),
('Kiribati','KI','KIR','296',''),
('China','CN','CHN','156',''),
('Cocos (Keeling) Islands','CC','CCK','166',''),
('Colombia','CO','COL','170',''),
('Comoros','KM','COM','174',''),
('Congo','CG','COG','178',''),
('Congo, Democratic Republic of the','CD','COD','180',''),
('Korea, Democratic People`s republic of','KP','PRK','408',''),
('Korea, Republic of','KR','KOR','410',''),
('Costa Rica','CR','CRI','188',''),
('Cote d`Ivoire','CI','CIV','384',''),
('Cuba','CU','CUB','192',''),
('Kuwait','KW','KWT','414',''),
('Curasao','CW','CUW','531',''),
('Lao People`s Democratic Republic','LA','LAO','418',''),
('Latvia','LV','LVA','428',''),
('Lesotho','LS','LSO','426',''),
('Lebanon','LB','LBN','422',''),
('Libyan Arab Jamahiriya','LY','LBY','434',''),
('Liberia','LR','LBR','430',''),
('Liechtenstein','LI','LIE','438',''),
('Lithuania','LT','LTU','440',''),
('Luxembourg','LU','LUX','442',''),
('Mauritius','MU','MUS','480',''),
('Mauritania','MR','MRT','478',''),
('Madagascar','MG','MDG','450',''),
('Mayotte','YT','MYT','175',''),
('Macao','MO','MAC','446',''),
('Malawi','MW','MWI','454',''),
('Malaysia','MY','MYS','458',''),
('Mali','ML','MLI','466',''),
('United States Minor Outlying Islands','UM','UMI','581',''),
('Maldives','MV','MDV','462',''),
('Malta','MT','MLT','470',''),
('Morocco','MA','MAR','504',''),
('Martinique','MQ','MTQ','474',''),
('Marshall Islands','MH','MHL','584',''),
('Mexico','MX','MEX','484',''),
('Micronesia, Federated States of','FM','FSM','583',''),
('Mozambique','MZ','MOZ','508',''),
('Moldova','MD','MDA','498',''),
('Monaco','MC','MCO','492',''),
('Mongolia','MN','MNG','496',''),
('Montserrat','MS','MSR','500',''),
('Burma','MM','MMR','104',''),
('Namibia','NA','NAM','516',''),
('Nauru','NR','NRU','520',''),
('Nepal','NP','NPL','524',''),
('Niger','NE','NER','562',''),
('Nigeria','NG','NGA','566',''),
('Netherlands','NL','NLD','528',''),
('Nicaragua','NI','NIC','558',''),
('Niue','NU','NIU','570',''),
('New Zealand','NZ','NZL','554',''),
('New Caledonia','NC','NCL','540',''),
```

```
('Norway','NO','NOR','578',''),
('United Arab Emirates','AE','ARE','784',''),
('Oman','OM','OMN','512',''),
('Bouvet Island','BV','BVT','74',''),
('Isle of Man','IM','IMN','833',''),
('Norfolk Island','NF','NFK','574',''),
('Christmas Island','CX','CXR','162',''),
('Heard Island and McDonald Islands','HM','HMD','334',''),
('Cayman Islands','KY','CYM','136',''),
('Cook Islands','CK','COK','184',''),
('Turks and Caicos Islands','TC','TCA','796',''),
('Pakistan','PK','PAK','586',''),
('Palau','PW','PLW','585',''),
('Palestinian Territory, Occupied','PS','PSE','275',''),
('Panama','PA','PAN','591',''),
('Holy See (Vatican City State)','VA','VAT','336',''),
('Papua New Guinea','PG','PNG','598',''),
('Paraguay','PY','PRY','600',''),
('Peru','PE','PER','604',''),
('Pitcairn','PN','PCN','612',''),
('Poland','PL','POL','616',''),
('Portugal','PT','PRT','620',''),
('Puerto Rico','PR','PRI','630',''),
('Macedonia, The Former Yugoslav Republic Of','MK','MKD','807',''),
('Reunion','RE','REU','638',''),
('Russian Federation','RU','RUS','643',''),
('Rwanda','RW','RWA','646',''),
('Romania','RO','ROU','642',''),
('Samoa','WS','WSM','882',''),
('San Marino','SM','SMR','674',''),
('Sao Tome and Principe','ST','STP','678',''),
('Saudi Arabia','SA','SAU','682',''),
('Swaziland','SZ','SWZ','748',''),
('Saint Helena, Ascension And Tristan Da Cunha','SH','SHN','654',''),
('Northern Mariana Islands','MP','MNP','580',''),
('Saint Barthelemy','BL','BLM','652',''),
('Saint Martin (French Part)','MF','MAF','663',''),
('Senegal','SN','SEN','686',''),
('Saint Vincent and the Grenadines','VC','VCT','670',''),
('Saint Kitts and Nevis','KN','KNA','659',''),
('Saint Lucia','LC','LCA','662',''),
('Saint Pierre and Miquelon','PM','SPM','666',''),
('Serbia','RS','SRB','688',''),
('Seychelles','SC','SYC','690',''),
('Singapore','SG','SGP','702',''),
('Sint Maarten','SX','SXM','534',''),
('Syrian Arab Republic','SY','SYR','760',''),
('Slovakia','SK','SVK','703',''),
('Slovenia','SI','SVN','705',''),
('United Kingdom','GB','GBR','826',''),
('United States','US','USA','840',''),
('Solomon Islands','SB','SLB','90',''),
('Somalia','SO','SOM','706',''),
('Sudan','SD','SDN','729',''),
('Suriname','SR','SUR','740',''),
('Sierra Leone','SL','SLE','694',''),
('Tajikistan','TJ','TJK','762',''),
('Thailand','TH','THA','764',''),
('Taiwan, Province of China','TW','TWN','158',''),
('Tanzania, United Republic Of','TZ','TZA','834',''),
('Timor-Leste','TL','TLS','626',''),
('Togo','TG','TGO','768',''),
```

```
('Tokelau','TK','TKL','772',''),
('Tonga','TO','TON','776',''),
('Trinidad and Tobago','TT','TTO','780',''),
('Tuvalu','TV','TUV','798',''),
('Tunisia','TN','TUN','788',''),
('Turkmenistan','TM','TKM','795',''),
('Turkey','TR','TUR','792',''),
('Uganda','UG','UGA','800',''),
('Uzbekistan','UZ','UZB','860',''),
('Ukraine','UA','UKR','804',''),
('Wallis and Futuna','WF','WLF','876',''),
('Uruguay','UY','URY','858',''),
('Faroe Islands','FO','FRO','234',''),
('Fiji','FJ','FJI','242',''),
('Philippines','PH','PHL','608',''),
('Finland','FI','FIN','246',''),
('Falkland Islands (Malvinas)','FK','FLK','238',''),
('France','FR','FRA','250',''),
('French Guiana','GF','GUF','254',''),
('French Polynesia','PF','PYF','258',''),
('French Southern Territories','TF','ATF','260',''),
('Croatia','HR','HRV','191',''),
('Central African Republic','CF','CAF','140',''),
('Chad','TD','TCD','148',''),
('Montenegro','ME','MNE','499',''),
('Czech Republic','CZ','CZE','203',''),
('Chile','CL','CHL','152',''),
('Switzerland','CH','CHE','756',''),
('Sweden','SE','SWE','752',''),
('Svalbard and Jan Mayen','SJ','SJM','744',''),
('Sri Lanka','LK','LKA','144',''),
('Ecuador','EC','ECU','218',''),
('Equatorial Guinea','GQ','GNQ','226',''),
('Aland Islands','AX','ALA','248',''),
('El Salvador','SV','SLV','222',''),
('Eritrea','ER','ERI','232',''),
('Estonia','EE','EST','233',''),
('Ethiopia','ET','ETH','231',''),
('South Africa','ZA','ZAF','710',''),
('South Georgia and the South Sandwich Islands','GS','SGS','239',''),
('South Ossetia','OS','OST','896',''),
('South Sudan','SS','SSD','728',''),
('Jamaica','JM','JAM','388',''),
('Japan','JP','JPN','392','');
commit;

-- End load app_t_country
```