Avro UI forms

This guide explains how to work with Avro UI forms. Avro UI forms are GUI components in the Admin UI that allow you to create Kaa schemas and enter corresponding data records without using the Avro/JSON syntax.

This guide also explains how to use *Avro UI forms GWT Sandbox*, which is a testing environment for Avro UI forms.

Avro UI form types

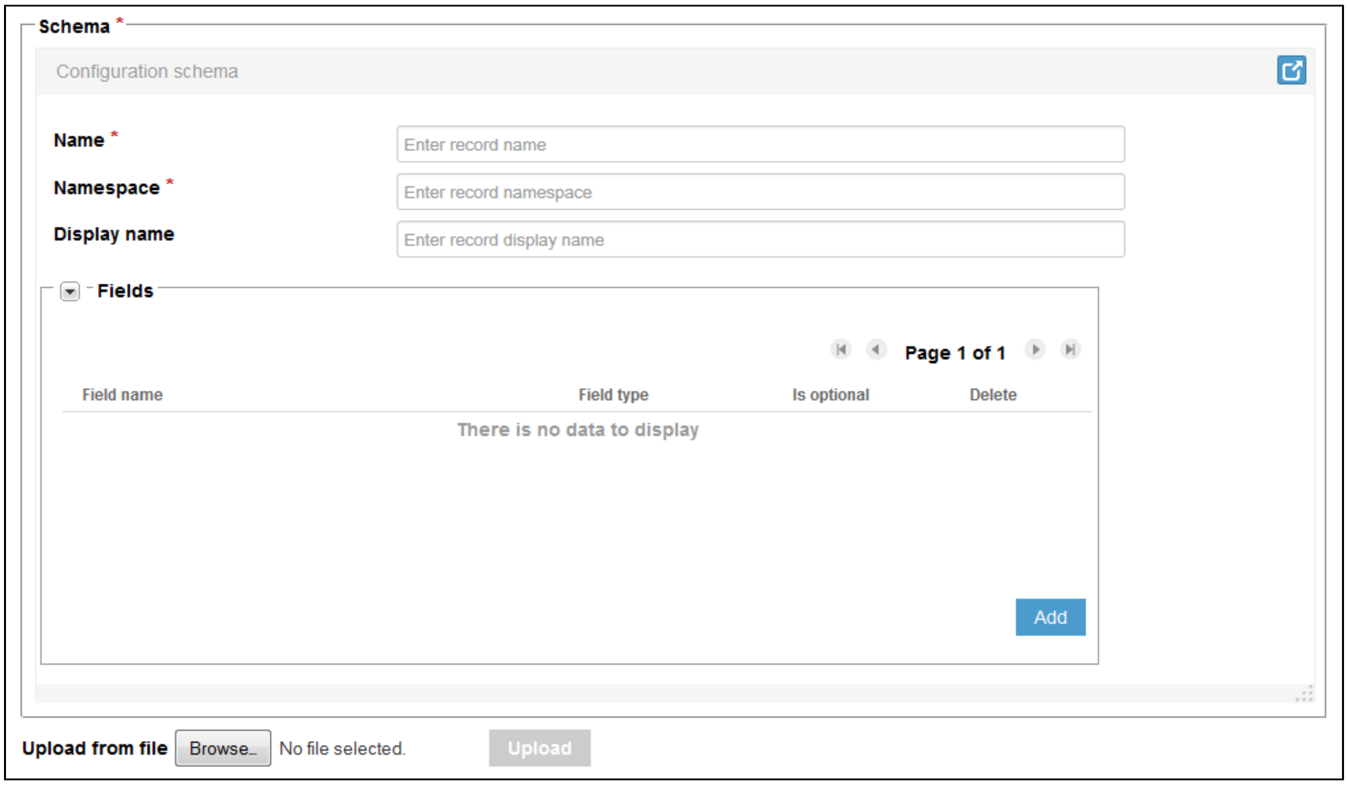
There are two Avro UI form types: a *schema form* and a *record form*.

Schema form

A schema form allows the user (tenant developer) to create Kaa schemas (profile, configuration, notification, and log schemas) for applications in Admin UI.

It consists of a header with the general schema info, such as the name, namespace, and display name, and the **Fields** section where you can add fields to your schema.

NOTE
You can also upload a schema from a file using the **Upload from file** option. In this case, the schema must be provided in the JSON format.

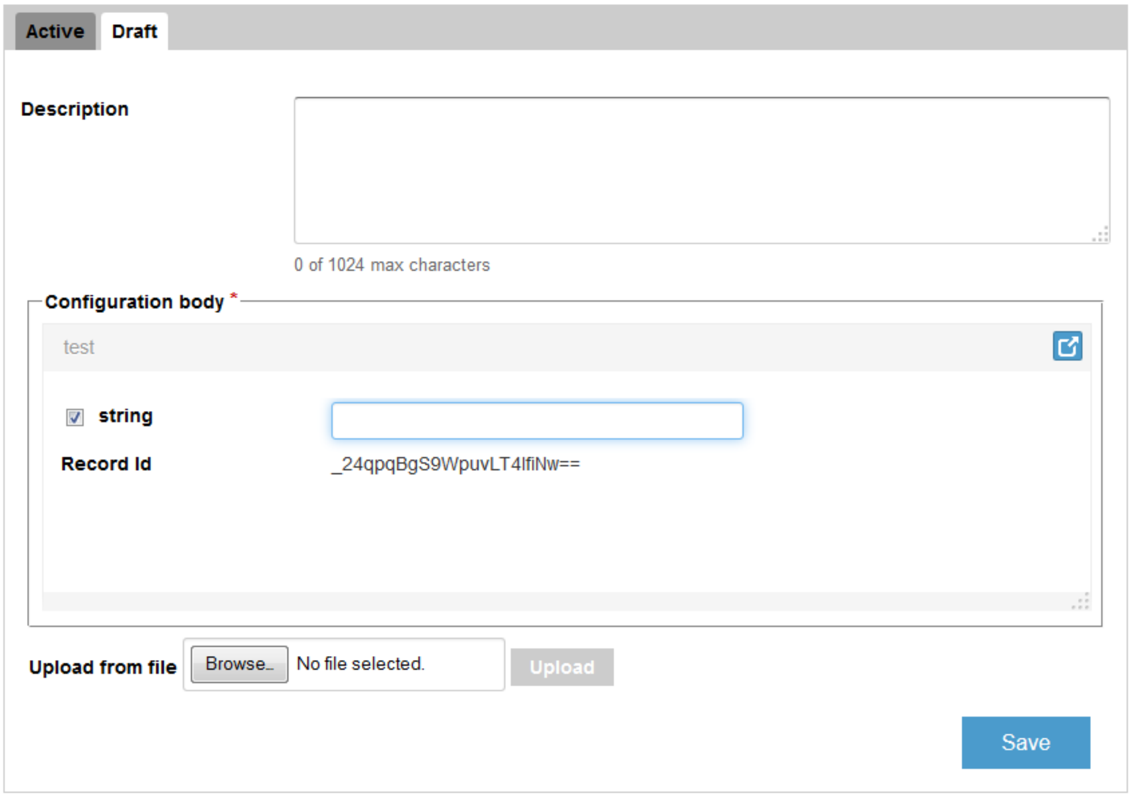


Record form

A record form allows the user (tenant developer) to enter data records according to corresponding Kaa schemas (configuration and notification schemas) for applications in Admin UI.

A record form contains the fields according to the corresponding schema.

> **NOTE**
> You can also upload data from a file using the **Upload from file** option. In this case, the data must be provided in the JSON format.

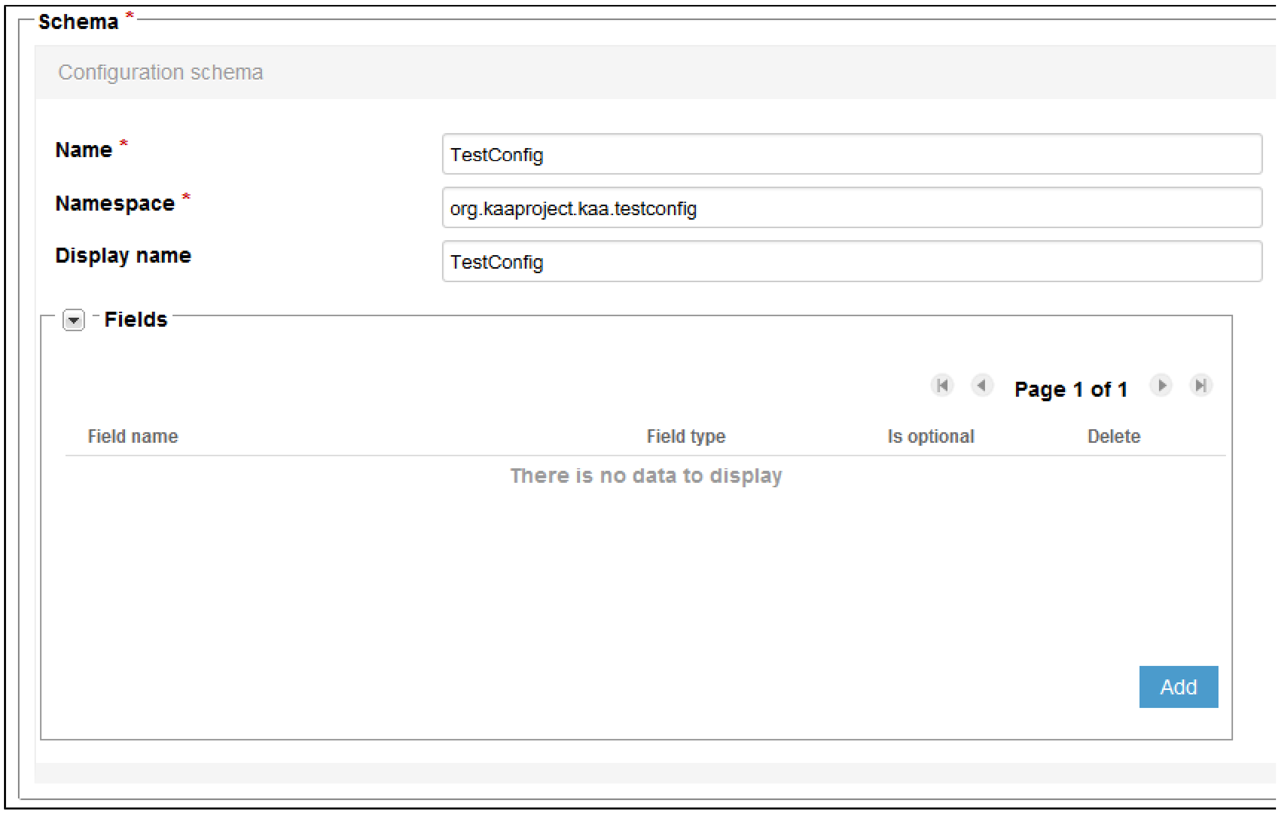


Description
0 of 1024 max characters
Configuration body *
test
☑ string
Record Id _24qpqBgS9WpuvLT4lfiNw==
Upload from file No file selected.

Working with schema form

An example of a schema form usage is creating a new configuration schema.

1. Open the **Configuration schemas** window for some application in Admin UI and then click **Add schema**.
2. In the **Configuration schema** schema form, specify the name, namespace, and display name of the record.



Schema *

Configuration schema

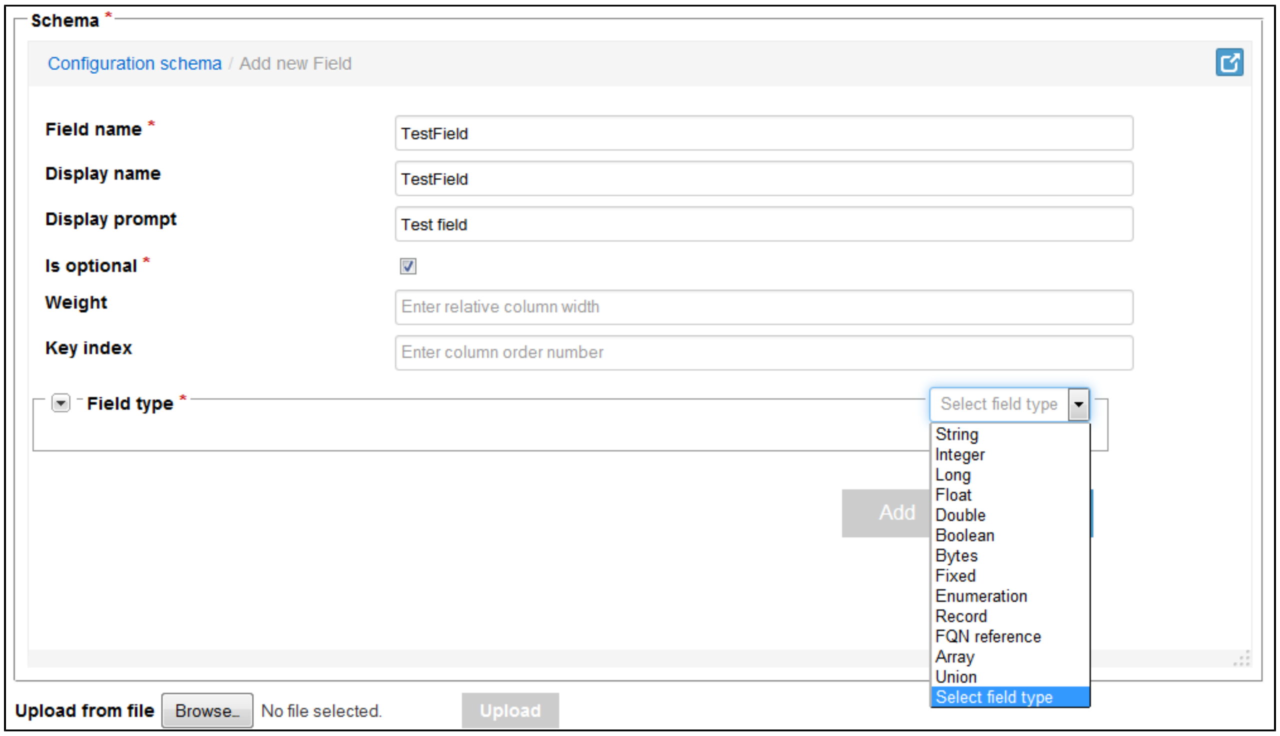
Name *	TestConfig
Namespace *	org.kaaproject.kaa.testconfig
Display name	TestConfig

Fields

Page 1 of 1

Field name	Field type	Is optional	Delete
There is no data to display			

3. To add a new field, click **Add** in the **Fields** section in the form.
4. On the **Add new Field** page of the form, specify the field parameters as required and select the field type.



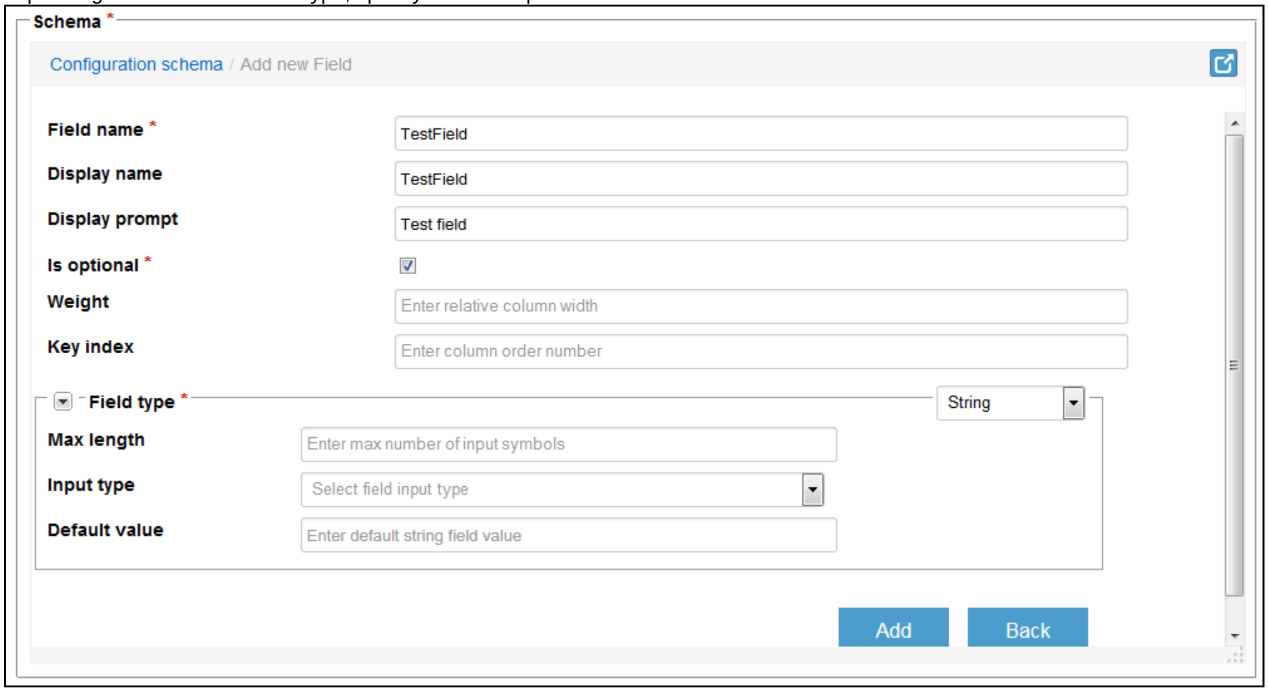
Schema *

Configuration schema / Add new Field

Field name *	TestField
Display name	TestField
Display prompt	Test field
Is optional *	☑
Weight	Enter relative column width
Key index	Enter column order number

Field type * Select field type

5. Depending on the selected field type, specify additional parameters for the field.



Schema *

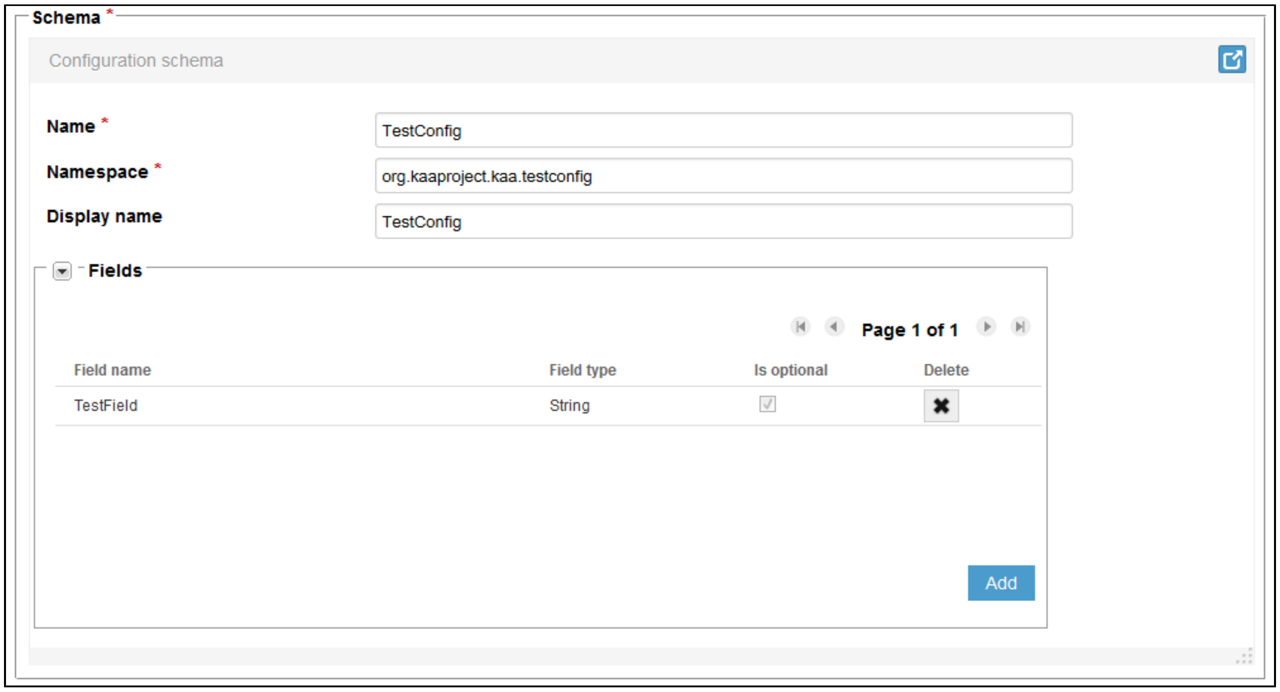
Configuration schema / Add new Field

Field name *	TestField
Display name	TestField
Display prompt	Test field
Is optional *	☑
Weight	Enter relative column width
Key index	Enter column order number

Field type * String

Max length	Enter max number of input symbols
Input type	Select field input type
Default value	Enter default string field value

6. Click **Add** to finish adding the field.


Schema *

Configuration schema

Name *	TestConfig
Namespace *	org.kaaproject.kaa.testconfig
Display name	TestConfig

Fields

Page 1 of 1

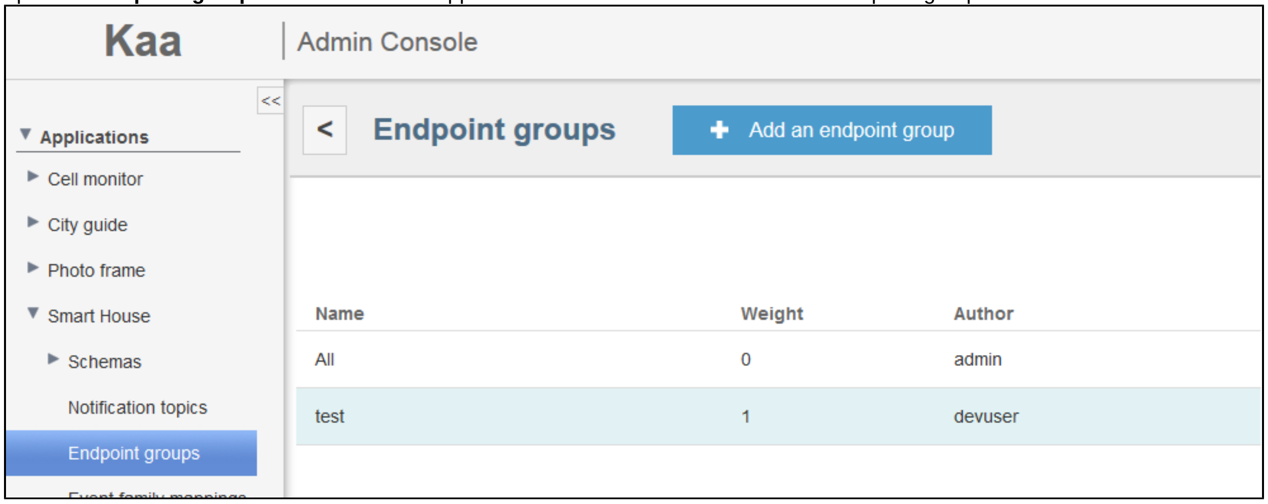
Field name	Field type	Is optional	Delete
TestField	String	✓	✖

7. In a similar way, add as many fields as necessary and then save the schema (see Administration UI guide for details).

Working with record form

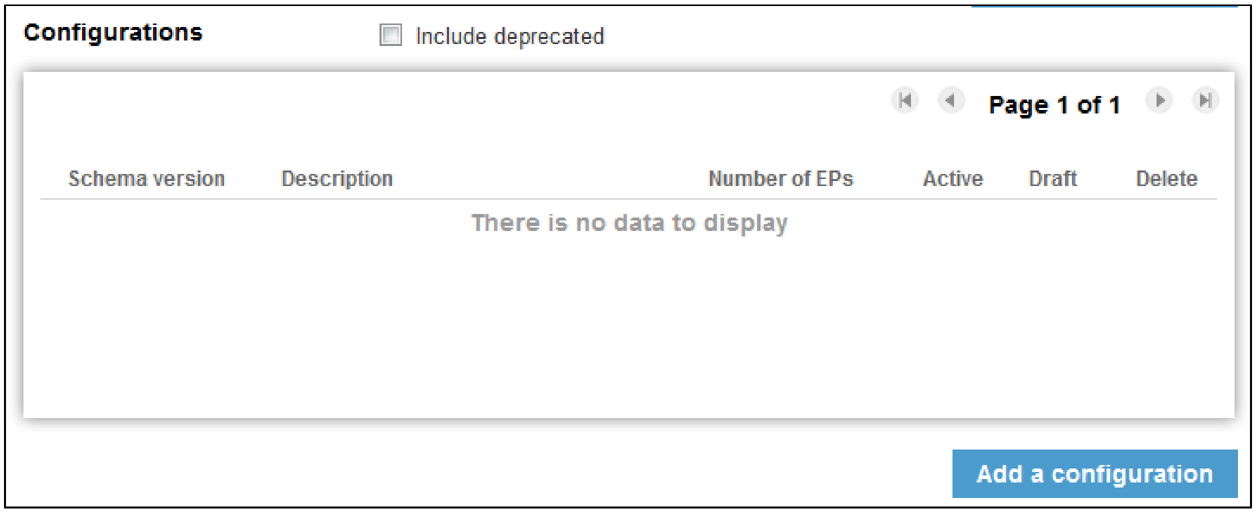
An example of a record form usage is adding new configuration data to an endpoint group.

1. Open the **Endpoint groups** window for some application in Admin UI and click on some endpoint group.



Name	Weight	Author
All	0	admin
test	1	devuser

2. Scroll down to the **Configurations** form and then click **Add configuration**.


Configurations

Include deprecated

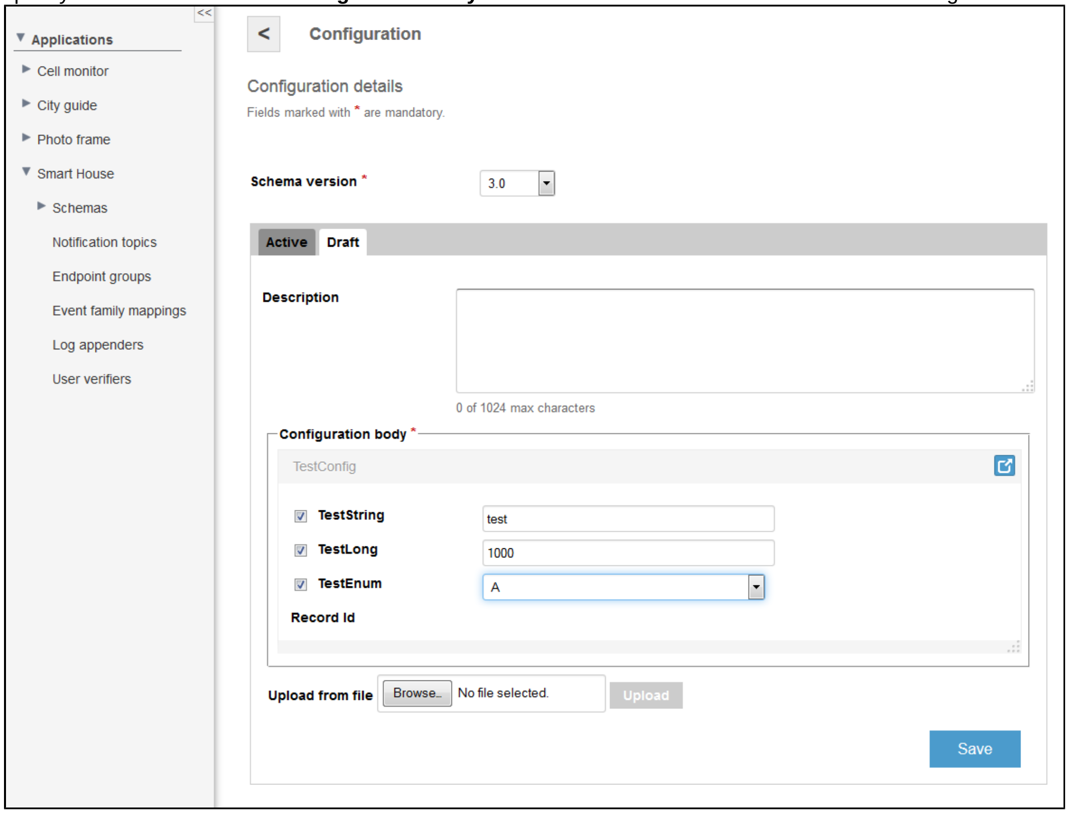
Page 1 of 1

Schema version	Description	Number of EPs	Active	Draft	Delete

There is no data to display

Add a configuration

3. Specify the field values in the **Configuration body** record form and then click **Save** to save the configuration.



Configuration

Configuration details

Fields marked with * are mandatory.

Schema version * 3.0

Active | Draft

Description

0 of 1024 max characters

Configuration body *

TestConfig

✓ TestString	test
✓ TestLong	1000
✓ TestEnum	A
Record Id	

Upload from file Browse... No file selected. Upload

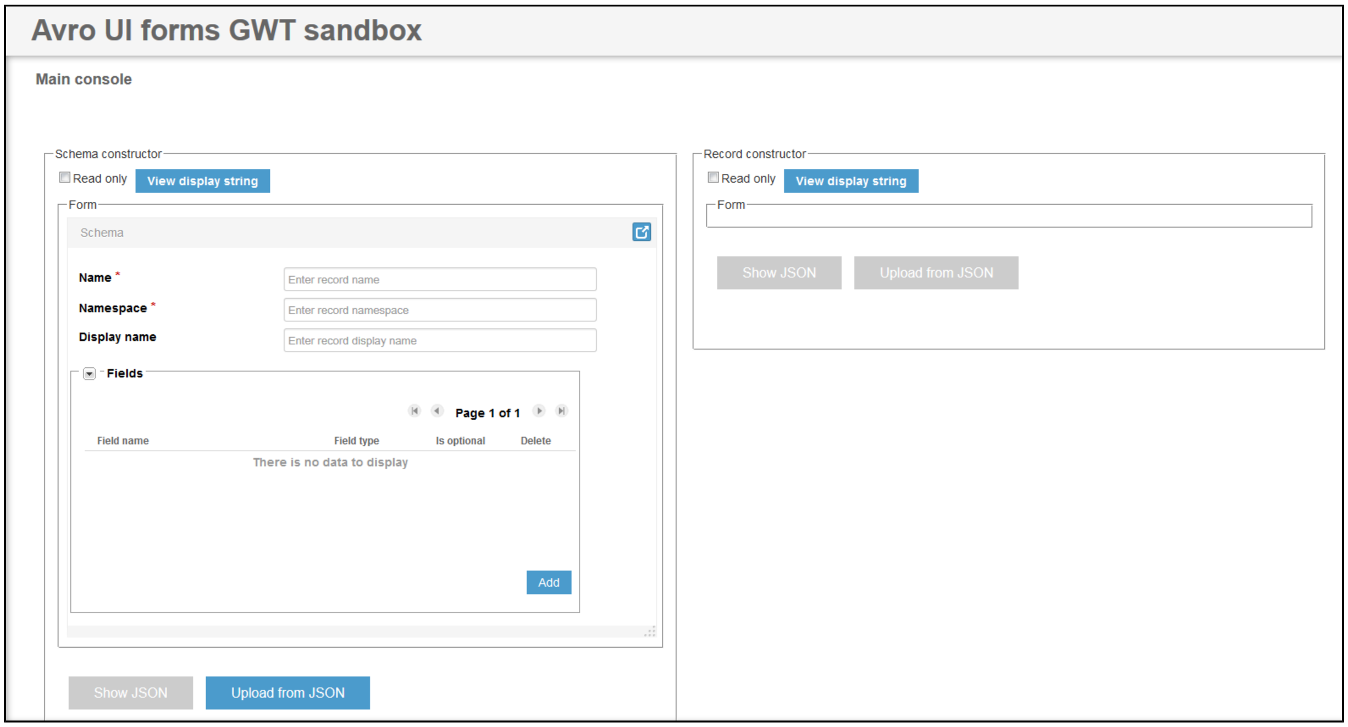
Save

4. To apply the new configuration, click **Activate**.

Avro UI forms GWT Sandbox

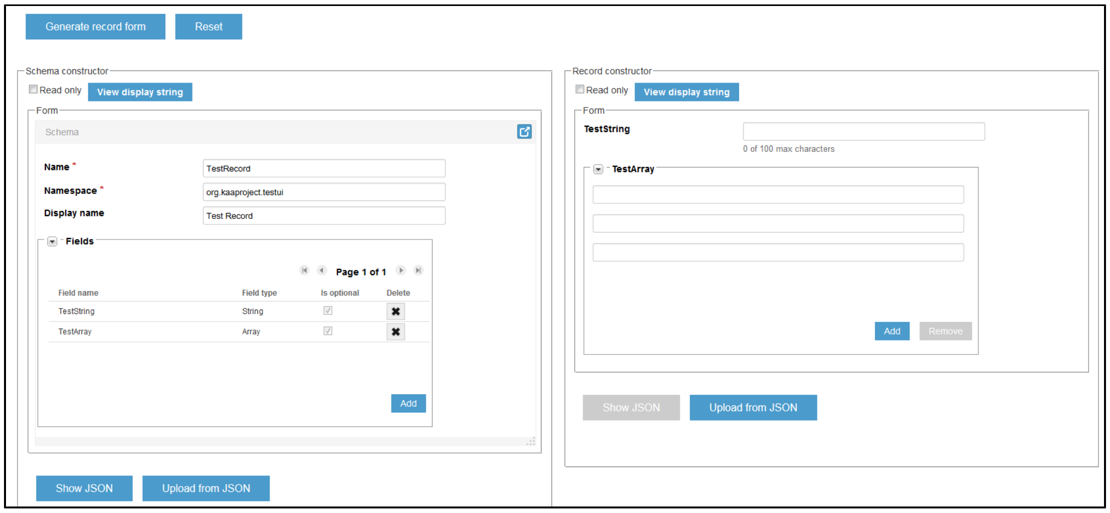
Avro UI forms GWT Sandbox is a tool for testing Avro UI forms. It allows you to create an Avro schema using the schema form (or adding a schema in the Avro format) and see what the corresponding record form will look like.

To start using this tool, install and run Kaa Sandbox at first and then go to the following URL (by default): http://127.0.0.1:9080/avroUiSandbox/.



To generate an Avro UI record form, proceed as follows:

1. Create an Avro schema in the schema form or, alternatively, click **Upload from JSON** and paste your schema into the text field.
2. Click **Generate form**.



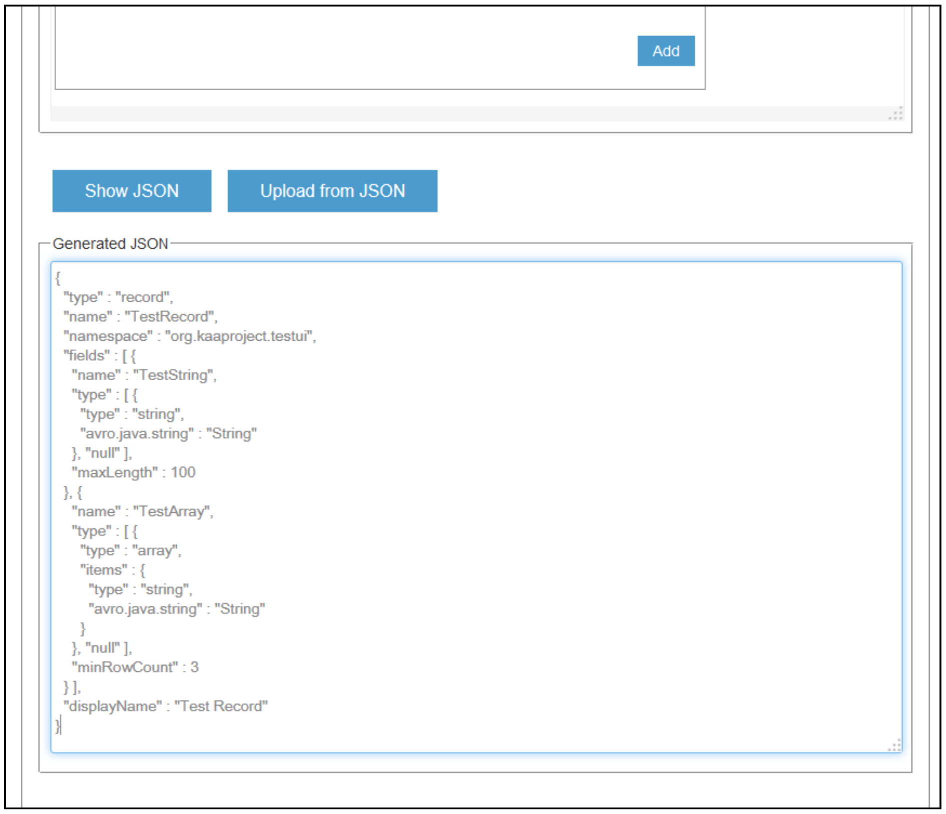
Name * TestRecord
Namespace * org.kaaproject.testui
Display name Test Record

Fields
Page 1 of 1

Field name	Field type	Is optional	Delete
TestString	String	☑	✖
TestArray	Array	☑	✖

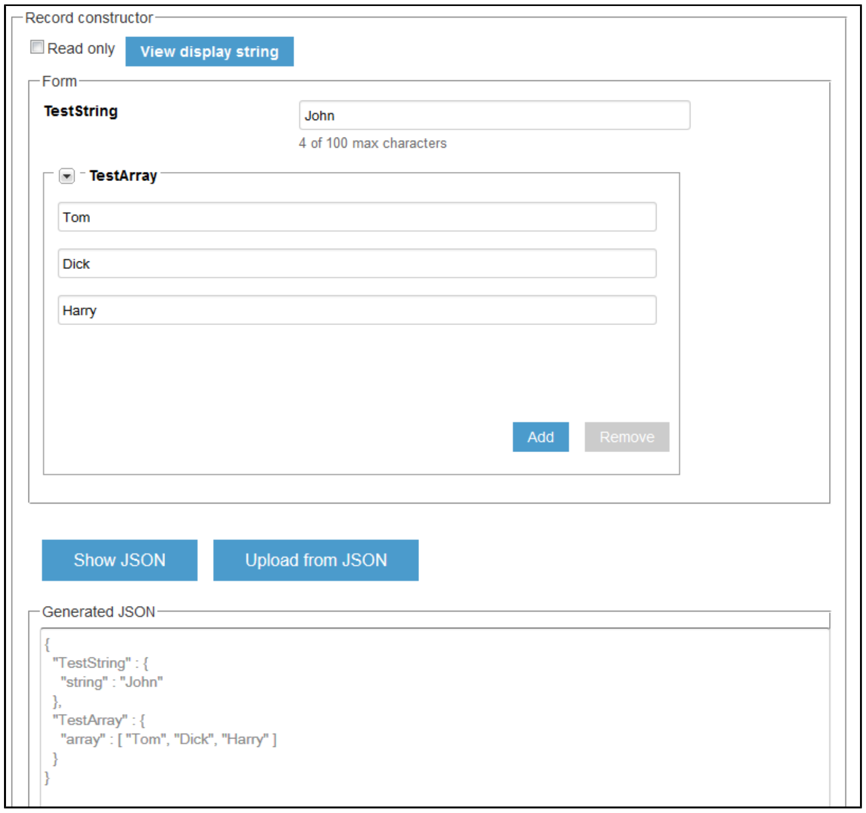
TestString
0 of 100 max characters
TestArray

3. (Optional) To view the generated schema in the JSON format, click **Show JSON** under the schema form.



```
{
  "type" : "record",
  "name" : "TestRecord",
  "namespace" : "org.kaaproject.testui",
  "fields" : [ {
    "name" : "TestString",
    "type" : [ {
      "type" : "string",
      "avro.java.string" : "String"
    }, "null" ],
    "maxLength" : 100
  }, {
    "name" : "TestArray",
    "type" : [ {
      "type" : "array",
      "items" : {
        "type" : "string",
        "avro.java.string" : "String"
      }
    }, "null" ],
    "minRowCount" : 3
  } ],
  "displayName" : "Test Record"
}
```

To view the JSON record for the data from the record form, enter some data into the field(s) and click **Show JSON**.



TestString

John

4 of 100 max characters

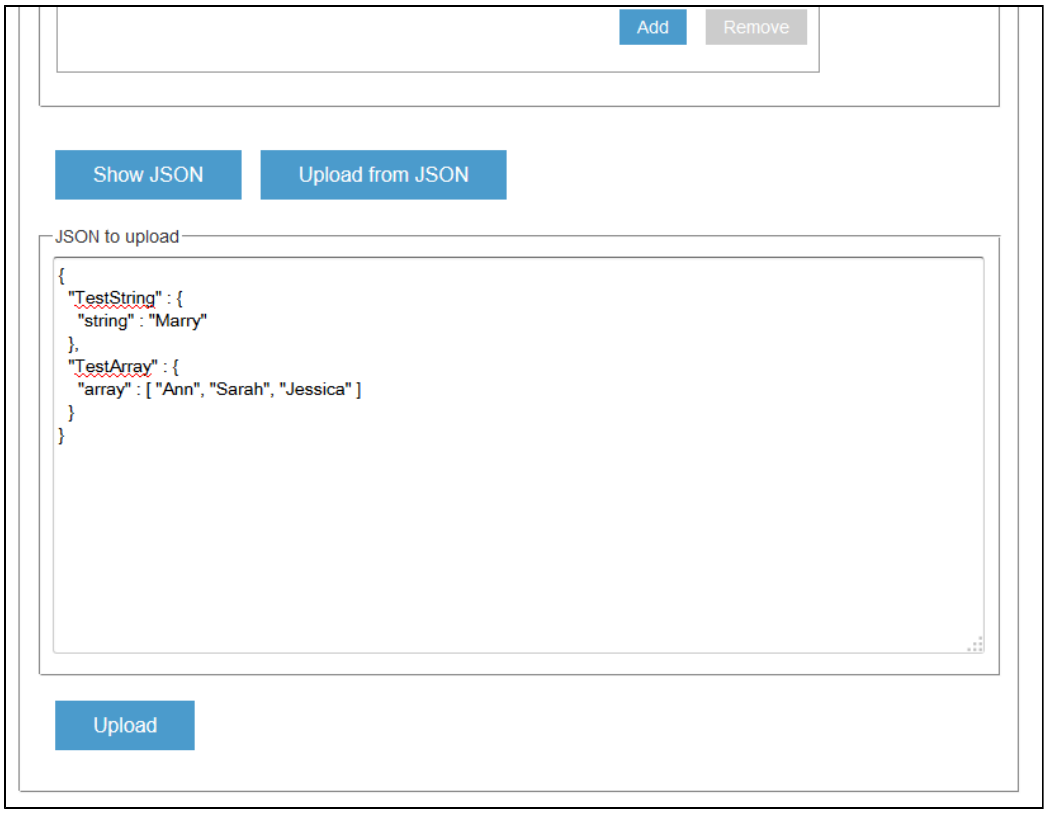
Tom

Dick

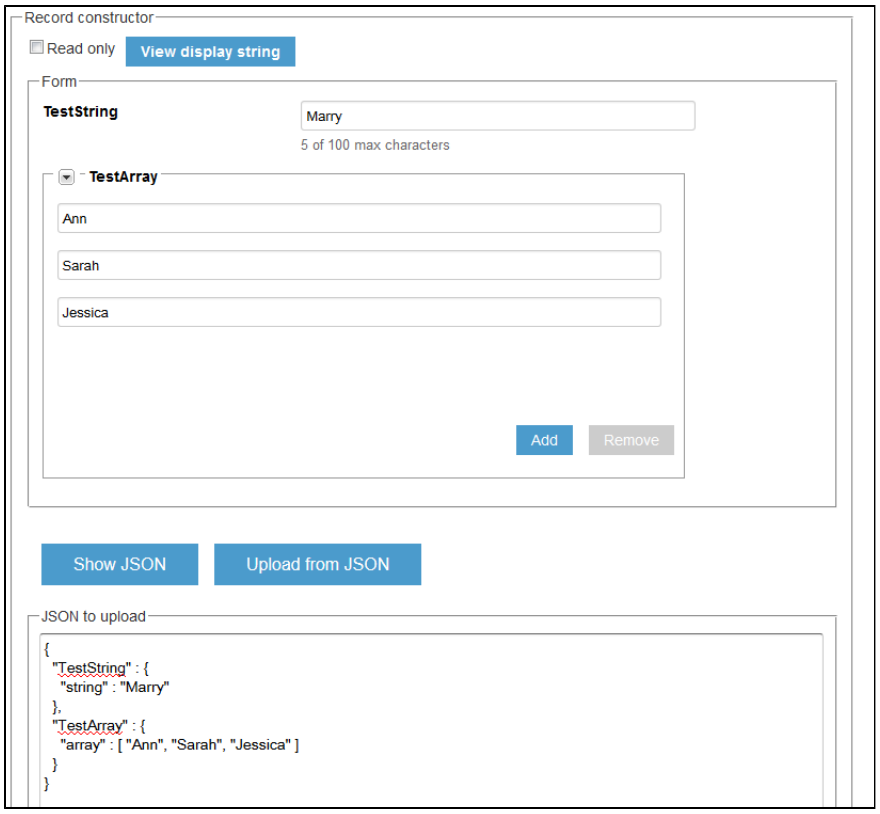
Harry

```
{
 "TestString" : {
   "string" : "John"
 },
 "TestArray" : {
   "array" : [ "Tom", "Dick", "Harry" ]
 }
}
```

To upload data from the JSON record into the record form, click **Upload from JSON**, then enter the data into the text field in the JSON format and click **Upload**.



```
{
 "TestString" : {
   "string" : "Marry"
 },
 "TestArray" : {
   "array" : [ "Ann", "Sarah", "Jessica" ]
 }
}
```



TestString: Marry

5 of 100 max characters

Ann

Sarah

Jessica

```
{
  "TestString" : {
    "string" : "Marry"
  },
  "TestArray" : {
    "array" : [ "Ann", "Sarah", "Jessica" ]
  }
}
```

Avro schema parameters

If the Avro schema is added in the JSON format rather than via GUI, the following parameters of the schema determine the layout and values displayed on the Avro UI record form.

- displayName - displays the name of the field on UI
- displayNames - displays the name of each enumeration symbol on UI (only for enumeration fields in the schema)
- displayPrompt - displays the text prompt for the field on UI
- by_default - displays the default value of the field on UI
- minRowCount - defines a minimum number of rows in a UI table (only for arrays in the schema)
- optional - defines whether the field on UI is optional or mandatory
- weight - defines a relative width of the corresponding column on UI (only for arrays in the schema)
- keyIndex - defines the order in which the fields of a record are displayed in a row (only for arrays of records in the schema; integer; if this parameter is defined selectively for specific fields of the record, the other fields of the record will not be displayed)
- overrideStrategy - defines how to merge arrays in the configuration across the endpoint groups (only for arrays in the schema; accepted values are "replace" and "append"; "replace" by default)
- fieldAccess - defines the viewing and editing restrictions for the field (accepted values are "read_only", "editable" and "hidden"; "editable" by default)
- inputType - defines whether the characters will be masked or not when the user enters the field value (accepted values are "password" and "plain"; "plain" by default)

Further reading

Use the following guides and references to make the most of Kaa.

Guide	What it is for
Design reference	Use this reference to learn about features and capabilities of Kaa.
Programming guide	Use this guide to create your own Kaa applications.
Administration UI guide	Use this guide to start working with Kaa Admin UI.