

# Creating Custom VDAB Forms

---

VDAB supports the definition of input forms that create custom data events that can be used initiate processing for appropriate flows.

<b>Before This</b>	You should have reviewed the VDAB introductory tutorial and documentation to understand how to create basic flows.
--------------------	--

## Related Documentation

The following document and tutorials either are a) available or b) being developed to further support this subject. Those available are highlighted in green while those under development are currently grey.

Related Guides	Details

Related Tutorial	Details

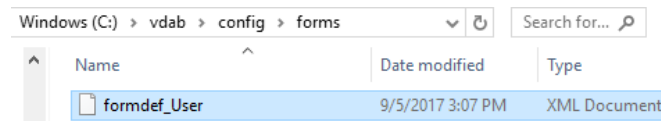
## Create a custom form file

A form file is an xml file with a specific structure that describes the form. This file should have the extension xml and can be used by VDAB once it is placed in the config/forms directory.

1. Using sample forms as a guide, create a form definition file that defines the form name, form group and form attributes.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<forms>
<form name="createUser" >
<attr name="Description">Add a role for a specified user</attr>
<attr name="FormGroup">MyForms</attr>
<attrdef name="Username">
<attr name="Category">Required</attr>
<attr name="MaxChars" type="Integer">24</attr>
<attr name="Desc">The username of the person responsible fo
</attrdef>
<attrdef name="Password">
<attr name="HashOutput" type="Boolean">>true</attr>
<attr name="Category">Required</attr>
<attr name="Desc">The password for the person associated wi
<attr name="SpecialType">Password</attr>
</attrdef>
<attrdef name="EMail">
<attr name="Category">Required</attr>
<attr name="Desc">A valid Email is required.</attr>
</attrdef>
</form>
</forms>
```

2. Copy the form definition xml file into the VDAB directory. *./config/forms*



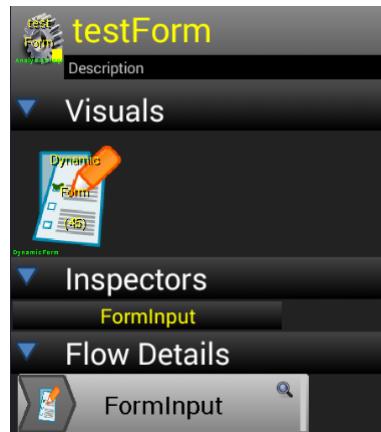
3. Restart VDAB

## Using the custom form in a flow

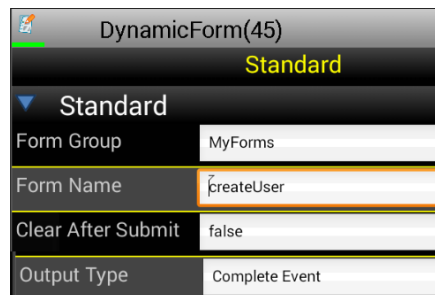
The custom form you have created can now be used by creating a flow with a form input. The recently created defined form can be selected and after the node is selected the form edited.

### Create a flow for alert handling

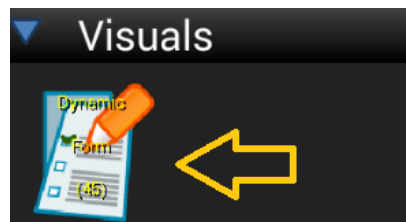
1. Create a flow and add an *FormInput* node to the flow. Add an inspector on the flow.



2. Select the edit option on the *FormInput* node and select the Form Group and Form Name defined in the form definition file



3. Start the flow and click on the form icon to enter data in the form. After finishing entry, note that the date event created by the form can be seen in the inspector.



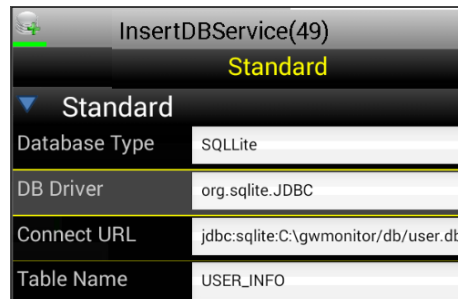
### Sending the Form data to a database.

A simple but useful example of how form data can be used involves writing the form data directly to a database using the *InsertDBService*.

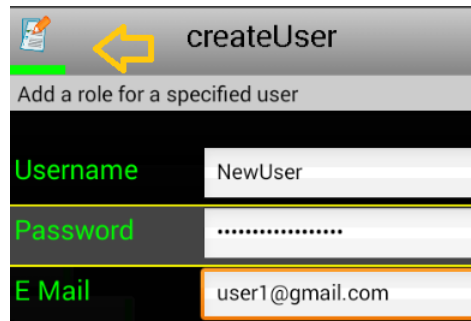
1. Click on the *FormInput* node select the Add Node option. Add an *InsertDBService* node to the *FormInput* node.



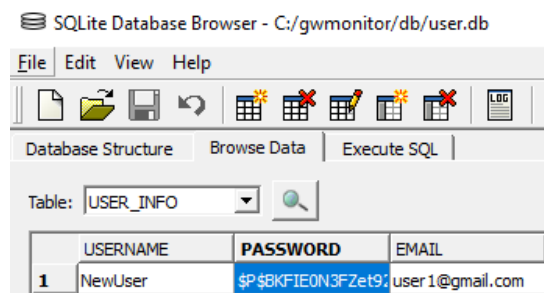
2. Click on the *InsertDBService* node and select the option to Edit the node. Define the DB information including the database type, connect URL, user and password. Set a table name.



3. Restart the flow and click on the form Icon. Enter data into the form and click in the upper left to save the changes.





4. The data will be written to the database table defined in the InsertDBService



## Appendix: Documentation for Nodes

This same information is available for all nodes by going to your local vdab web server at [http://localhost/vdab/docs\\_node](http://localhost/vdab/docs_node).

 <b>FormInput</b> <span style="float: right;">Beta</span>	
FormInput supports the input of data defined by a form found in the formdef.xml file. Check the VDAB documentation for information on how to create forms.	
FormGroup	The form group for the form to be selected
FormName	The form to be input, select the form group first
ClearAfterSubmit	If set, clears all form attributes after each from entry
OutputNullFields	If set, includes null fields that are output from the form
ViewGroups	The view groups that this Form will be associated with.
OutputType	The type of output that will published by this node. Supported types include Complete Events,Triggers and a Boolean value
Comments	Additional comments about this node, flow or container.

 <b>InsertDBService</b> <span style="float: right;">Beta</span>	
Inserts a single row in a table based on the structure of the event and the table specified.	
DatabaseType	The database type that will be used by this node. Support types include H2, SQLite, Oracle, MySql and SQLServer
DBDriver	The JAVA database driver that will be used to access the database
ConnectURL	The Connection URL appropriate for that database type
User	The username or user id associated with the access required for this node.
Password	The password associated with the user specified on this node.
ReconnectAfter	If set, forces the DB connection to be reestablished after the given number of seconds.
TableName	The tablename for events. If the incoming data complex, the tablename is taken from the data and this value is ignored.
OutputType	The type of output that will published by this node. Supported types include Complete Events,Triggers and a Boolean value
Comments	Additional comments about this node, flow or container.