

FormService

- [Description](#)
- [Code Sample](#)
- [Fields](#)
 - [PREFIX_FOREIGN_KEY](#)
 - [PREFIX_FOREIGN_KEY_EDITABLE](#)
 - [PREVIEW_MODE](#)
- [Methods](#)
 - [createElementFromJson](#)
 - [createElementFromJson](#)
 - [executeFormActions](#)
 - [executeFormLoadBinders](#)
 - [executeFormOptionsBinders](#)
 - [executeFormStoreBinders](#)
 - [generateElementDesignerHtml](#)
 - [generateElementErrorHtml](#)
 - [generateElementJson](#)
 - [loadFormData](#)
 - [loadFormFromJson](#)
 - [previewElement](#)
 - [previewElement](#)
 - [recursiveExecuteFormStoreBinders](#)
 - [retrieveFormDataFromRequest](#)
 - [retrieveFormDataFromRequestMap](#)
 - [retrieveFormErrorHtml](#)
 - [retrieveFormHtml](#)
 - [storeElementData](#)
 - [submitForm](#)
 - [validateFormData](#)
 - [viewForm](#)
 - [viewFormFromJson](#)

Description

- `org.joget.apps.form.service.FormService`
- Under wflow-core module
- Service methods used to creates form from json and performs form feature actions

Code Sample

```
import org.joget.apps.app.dao.FormDefinitionDao;
import org.joget.apps.app.model.FormDefinition;
import org.joget.apps.app.model.AppDefinition;
import org.joget.apps.app.service.AppUtil;
import org.joget.apps.form.model.Form;
import org.joget.apps.form.service.FormService;

AppService appService = (AppService) AppUtil.getApplicationContext().getBean("appService");
FormDefinitionDao formDefinitionDao = (FormDefinitionDao) AppUtil.getApplicationContext().getBean(
("formDefinitionDao");
FormService formService = (FormService) AppUtil.getApplicationContext().getBean("formService");

String id = "crm_account";
String appId = "crm";
Long appVersion = appService.getPublishedVersion(appId);

AppDefinition appDef = appService.getAppDefinition(appId, appVersion.toString());
FormDefinition formDef = formDefinitionDao.loadById(formId, appDef);
if (formDef != null) {
    //create the form
    Form form = (Form) formService.createElementFromJson(formDef.getJson(), true);
}
```

Fields

PREFIX_FOREIGN_KEY

```
public java.lang.String PREFIX_FOREIGN_KEY = "fk_";
```

Prefix of a request parameter used to sets the value of a form field and change the field to a readonly field.

PREFIX_FOREIGN_KEY_EDITABLE

```
public java.lang.String PREFIX_FOREIGN_KEY_EDITABLE = "fke_";
```

Prefix of a request parameter used to sets the value of a form field but without change the field to readonly.

PREVIEW_MODE

```
java.lang.String PREVIEW_MODE = "_PREVIEW_MODE";
```

A key to set the FormResult as preview mode.

Methods

createElementFromJson

```
public org.joget.apps.form.model.Element createElementFromJson(java.lang.String elementJson)
```

Creates an element object from a JSON definition

createElementFromJson

```
public org.joget.apps.form.model.Element createElementFromJson(java.lang.String elementJson, boolean processHashVariable)
```

Creates an element object from a JSON definition

executeFormActions

```
public org.joget.apps.form.model.FormData executeFormActions(org.joget.apps.form.model.Form form, org.joget.apps.form.model.FormData formData)
```

Invokes actions (e.g. buttons) in the form

executeFormLoadBinders

```
public org.joget.apps.form.model.FormData executeFormLoadBinders(org.joget.apps.form.model.Element element, org.joget.apps.form.model.FormData formData)
```

Loads data for a specific row into an element by calling all load binders in the element.

executeFormOptionsBinders

```
public org.joget.apps.form.model.FormData executeFormOptionsBinders(org.joget.apps.form.model.Element element, org.joget.apps.form.model.FormData formData)
```

Preloads data for an element, e.g. field options, etc. by calling all option binders in the element.

executeFormStoreBinders

```
public org.joget.apps.form.model.FormData executeFormStoreBinders(org.joget.apps.form.model.Form form, org.joget.apps.form.model.FormData formData)
```

Executes store binders for a form

generateElementDesignerHtml

```
public java.lang.String generateElementDesignerHtml(org.joget.apps.form.model.Element element, org.joget.apps.form.model.FormData formData, boolean includeMetaData)
```

Generates HTML for the form element to be used in the Form Builder

generateElementErrorHtml

```
public java.lang.String generateElementErrorHtml(org.joget.apps.form.model.Element element, org.joget.apps.form.model.FormData formData)
```

Generates error HTML for the form element

generateElementJson

```
public java.lang.String generateElementHtml(org.joget.apps.form.model.Element element, org.joget.apps.form.model.FormData formData)
```

Generates HTML for the form element

loadFormData

```
public org.joget.apps.form.model.Form loadFormData(org.joget.apps.form.model.Form form, org.joget.apps.form.model.FormData formData)
```

Main method to load a form with data loaded.

loadFormFromJson

```
public org.joget.apps.form.model.Form loadFormFromJson(java.lang.String formJson, org.joget.apps.form.model.FormData formData)
```

Load a form from its JSON definition, with data loaded.

previewElement

```
public java.lang.String previewElement(java.lang.String json)
```

Use case to generate HTML from a JSON element definition.

previewElement

```
public java.lang.String previewElement(java.lang.String json, boolean includeMetaData)
```

Use case to generate HTML from a JSON element definition.

recursiveExecuteFormStoreBinders

```
public org.joget.apps.form.model.FormData recursiveExecuteFormStoreBinders(org.joget.apps.form.model.Form form, org.joget.apps.form.model.Element element, org.joget.apps.form.model.FormData formData)
```

Recursively executes all the store binders in a form

retrieveFormDataFromRequest

```
public org.joget.apps.form.model.FormData retrieveFormDataFromRequest(org.joget.apps.form.model.FormData formData, javax.servlet.http.HttpServletRequest request)
```

Retrieves form data submitted via a HTTP servlet request

retrieveFormDataFromRequestMap

```
public org.joget.apps.form.model.FormData retrieveFormDataFromRequestMap(org.joget.apps.form.model.FormData formData, java.util.Map requestMap)
```

Retrieves form data submitted via a HTTP servlet request parameters map

retrieveFormErrorHtml

```
public java.lang.String retrieveFormErrorHtml(org.joget.apps.form.model.Form form, org.joget.apps.form.model.FormData formData)
```

Used to retrieves the form HTML when there is errors in form

retrieveFormHtml

Used to retrieves the Form HTML

storeElementData

```
public org.joget.apps.form.model.FormData storeElementData(org.joget.apps.form.model.Form form, org.joget.apps.form.model.Element element, org.joget.apps.form.model.FormData formData)
```

Store the data of a form field element

submitForm

```
public org.joget.apps.form.model.FormData submitForm(org.joget.apps.form.model.Form form, org.joget.apps.form.model.FormData formData, boolean ignoreValidation)
```

Process form submission

validateFormData

```
public org.joget.apps.form.model.FormData validateFormData(org.joget.apps.form.model.Form form, org.joget.apps.form.model.FormData formData)
```

Validates form data submitted for a specific form

viewForm

```
public java.lang.String viewForm(org.joget.apps.form.model.Form form, java.lang.String primaryKeyValue)
```

Use case to load and view a form, with data loaded

viewFormFromJson

```
public java.lang.String viewFormFromJson(java.lang.String formJson, java.lang.String primaryKeyValue)
```

Use case to view a form from its JSON definition, with data loaded