Oracle ODA - Custom Component for EC

Radu Marinache (SEHUB CX Apps Tech Team - 2021)

The following documentation explains the process of installing and using this custom component which will help you integrate Oracle ODA with any instance of Engagement Cloud.

Materials

EC_GenericRESTcall-1.0.0.tgz - The Custom Component tgz archive you can use for install in any ODA skill

EXAMPLE_OF_DialogFlow.yml - An example of yml dialog flow of using the custom component

EXAMPLE_OF SKILL_radu_aut_demo(1.0).zip - An ODA skill you can import that has an exemplification of component usage

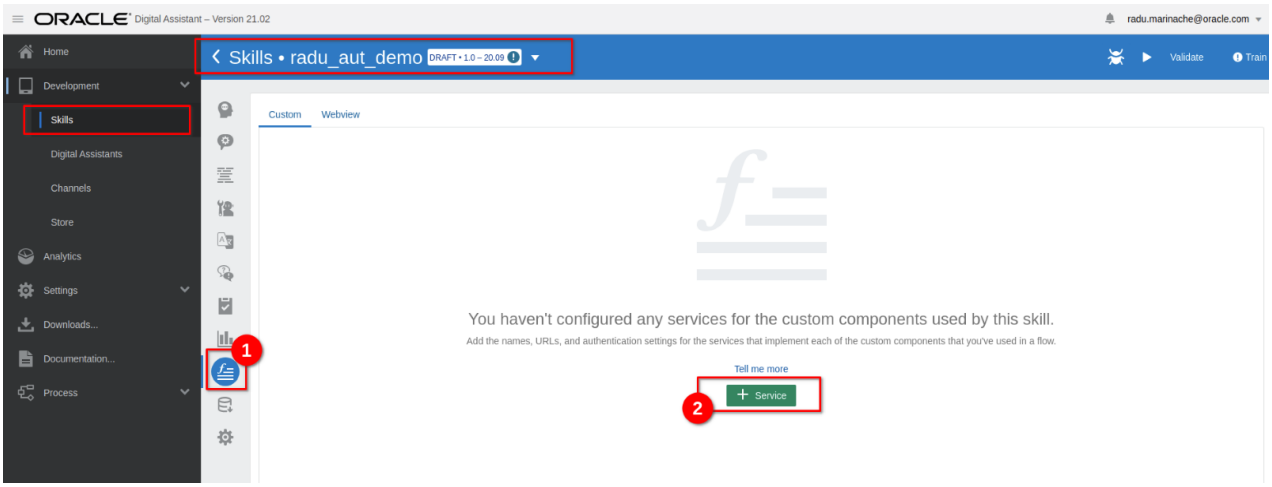
CreateRecord.mp4 - Video of using the component to create a record in EC

Installation.mp4 - Video of installing the component into a skill

RetriveData.mp4 - Video showing how to use the component to get data from EC

INSTALLATION

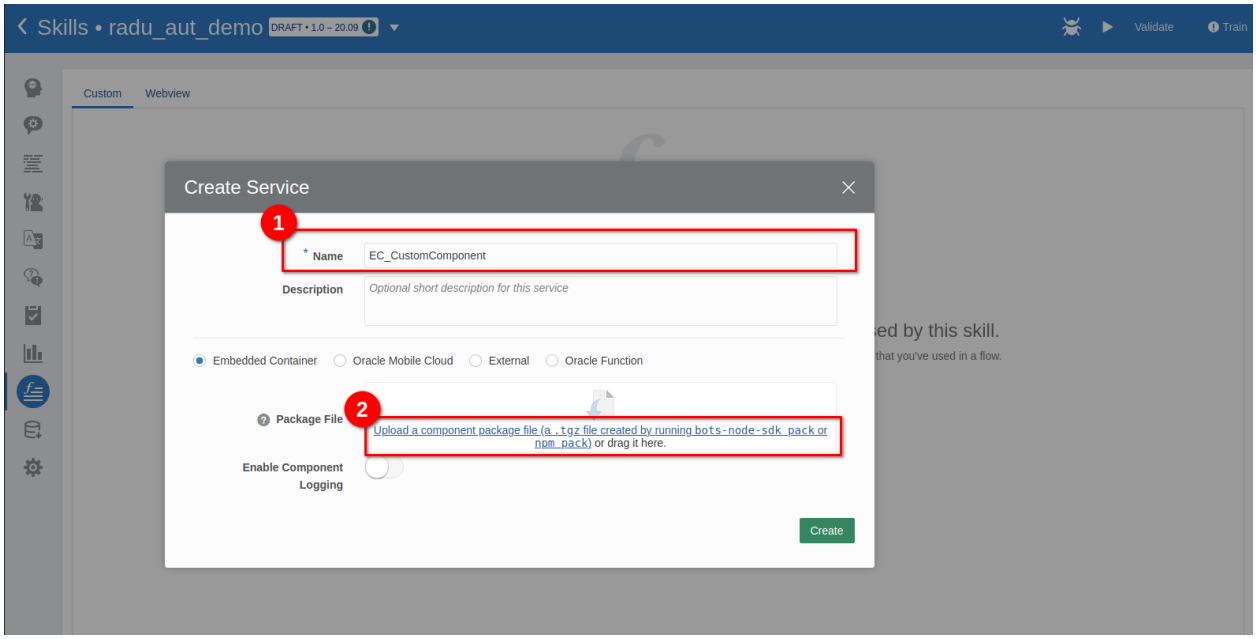
Navigate to your skill and go to Components tab and click Service button.



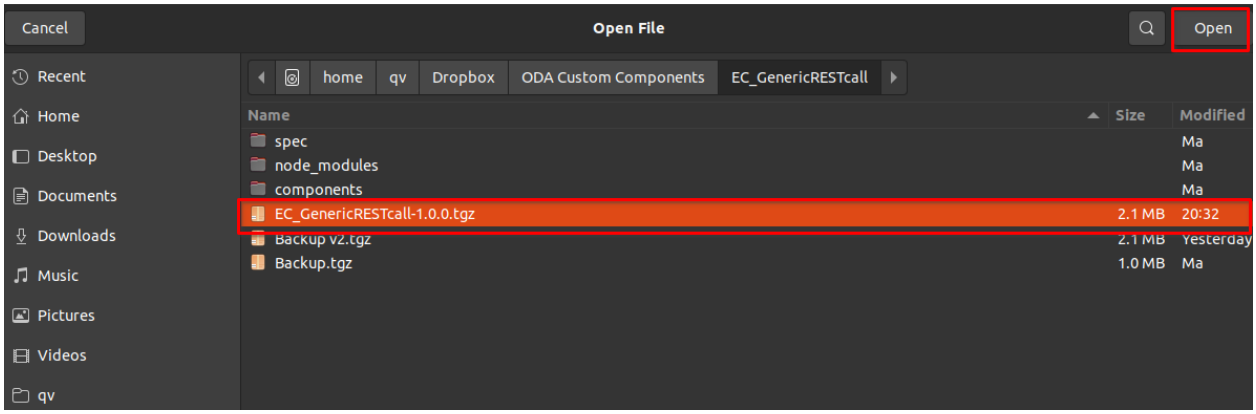
Provide a Name to the service (It can be any name you want) .

Make sure the Embedded Container is selected.

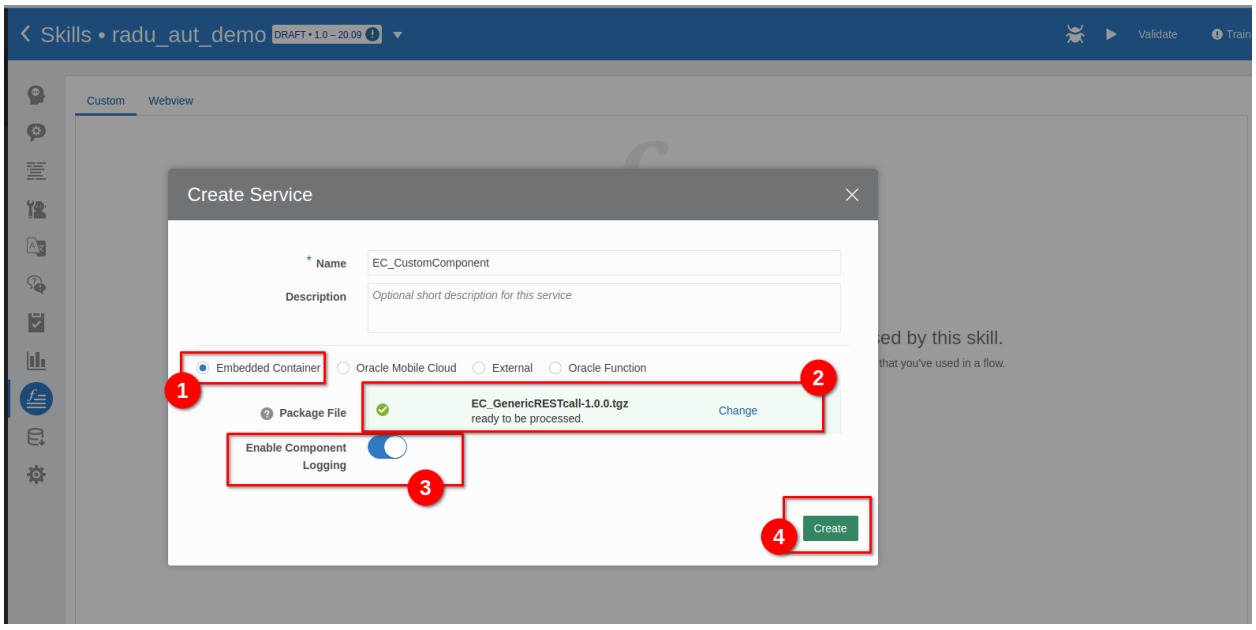
And click upload a component package.



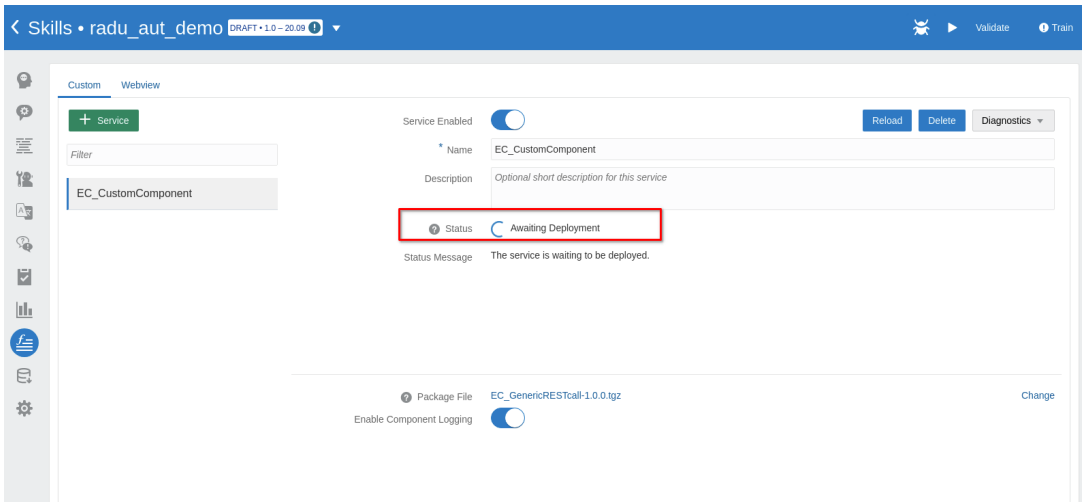
Select the tgz archive

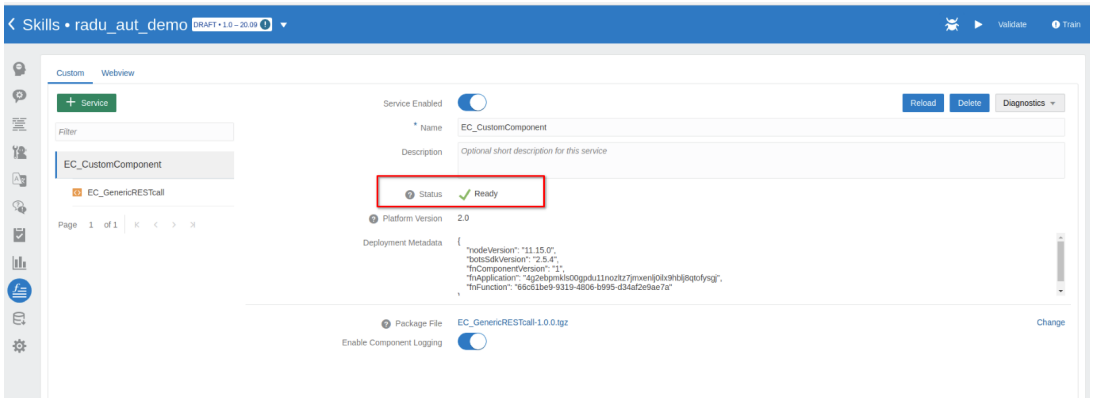


Click Create



Wait for the status to Change to Ready





If the status will change to ready the installation is finished and you can use the component in your conversation flow.

HOW TO USE

Properties (Custom Component Variables)

Variable Name	Type	Required	Value	Description
var_username	text	yes	User for calling the REST	It is used to store the user's name of EC used to do the REST call
var_password	text	yes	Weekly password for EC	It is used to store the weekly password of EC
var_ecdomain	text	yes	Endpoints for EC REST api	used to provide the endpoint for EC REST call
var_operation	text	yes	GET/POST/PATCH/KA	used to specify the REST operation (GET/POST/PATCH/KA - Knowledge article search)
var_contentType	text	yes	application/vnd.oracle.adf.resourceitem+json	Specify the Content Type of the REST call
var_payload	text	optional	E.g:	Payload for POST or

			`{"Title":"${srTitle}"}'`	PATCH
var_FieldsToExtract	text	optional	API_NAME-DIPLAY_LABLE ,API_NAME-DIPLAY_LABLE	a comma separated list of the fields you want to retrieved from the REST call response
var_KAcontentEndpoint	text	optional	ENDPOINT FOR EC KA QUESTION	Used to provide the endpoint for Knowledge article search
var_ODAchannel	text	optional	web	Specify where the ODA will be used e.g Web, Facebook so on . Only web is supported at this point to have html rendering

The response of the REST calls will be stored in a user variable - > **user.payload**

So, this user variable can be used in a next state to display the results of the component execution.

Actions

This component does not have a specific action list supported; it will simply transition to the next immediate state after it runs.

Retrieve data from Engagement Cloud (GET)

Add in your skill conversation yaml the following

```
GET:
  component: "EC_GenericRESTcall"
  properties:
    var_password: "<EC WEEKLY PASSWORD>"
    var_username: "<EC USERNAME>"
    var_ecdomain: "<REST ENDPOINT>"
    var_operation: "GET"
    var_ODAchannel: "web"
    var_contentType: "application/vnd.oracle.adf.resourceitem+json"
    var_FieldsToExtract: "<LIST_OF_FIELDS TO RETURN FROM REST RESPONSE>"
```

var_FieldsToExtract

- This is the comma separated list of fields that will be returned from the response payload; however this list of fields will be actually a list of pairs where you specify the API name and the Display Name you want to see in the chat dialog box.

e.g : Let's presume we want to get service request information and we want to diplay to the chatbot user SR Number and Status of the service request.

In this case we can specify into this variable the field lis as below:

```
var_FieldsToExtract: "SrNumber-SR Number,StatusTypeCdMeaning-Status"
```

So, notice the format is: **API_NAME-DISPLAY NAME, API_NAME-DISPLAY NAME**

The API_NAME is the exact field name you find in the Application Composer or in the REST Payload

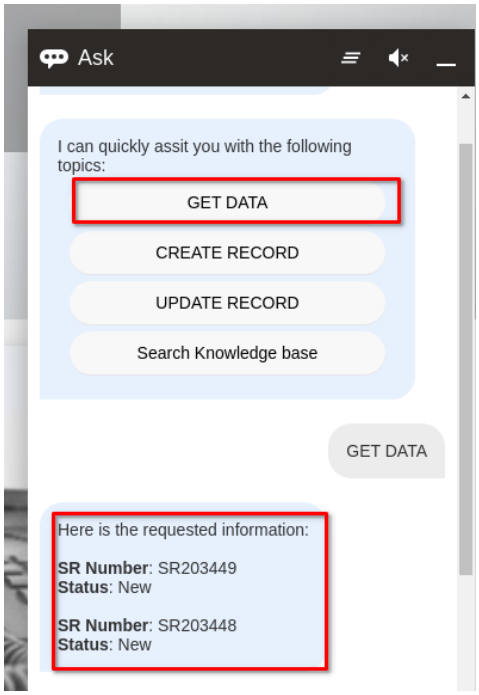
The DISPLAY NAME can be whatever label you want, it is the value that will be shown after the component executes

After the component runs it will return the response in a user variable called user.payload and it will transition immediately to the next state, so you could use immediately after the component state a System.Output component to show the results.

```
# GET REST CALL

  GET:
    component: "EC_GenericRESTcall"
    properties:
      var_password: "kzJ77956"
      var_username: "cole.mitchell_zvqv"
      var_ecdomain: "https://adc2-zvqy-fa-ext.oracledemos.com:443/crmRestApi/resources/11.13.18.05/serviceRequests?q=ReportedByPartyId=100000000409225&limit=2"
      var_operation: "GET"
      var_ODAchannel: "web"
      var_contentType: "application/vnd.oracle.adf.resourceitem+json"
      var_FieldsToExtract: "SrNumber-SR Number,StatusTypeCdMeaning-Status"

  viewGETTresults:
    component: "System.Output"
    properties:
      text: "Here is the requested information:\n ${user.payload}"
      keepTurn: true
    transitions:
      next: "helpAgain"
```



Create a record in Engagement Cloud (POST)

To create records or better say do a POST call the yaml is similar to get to which you need to a the payload variable and has the general form as below:

```
POST:
  component: "EC_GenericRESTcall"
  properties:
    var_password: "<EC WEEKLY PASSWORD>"
    var_username: "<EC USERNAME>"
    var_ecdomain: "<REST ENDPOINT>"
    var_operation: "POST"
    var_ODAchannel: "web"
    var_contentType: "application/vnd.oracle.adf.resourceitem+json"
    var_FieldsToExtract: "<LIST OF FIELDS TO RETURN FROM REST RESPONSE>"
    var_payload: '<POST PAYLOAD>'
```

var_FieldsToExtract

This is similar format as for GET operations and in the POST context will retrive the fields that will be found in the POST response payload

Var_payload

Payload string must be in a JSON format just like you would do it in POSTMAN, but it has to be arranged in 1 line of text(You can use https://lingojam.com/TexttoOneLine to convert the JSON payload to 1 line text)

Example

Lets presume a Service Request has to be created, the minimum fields that need to be sent in the POST payload is title , so the payload should look as bellow:

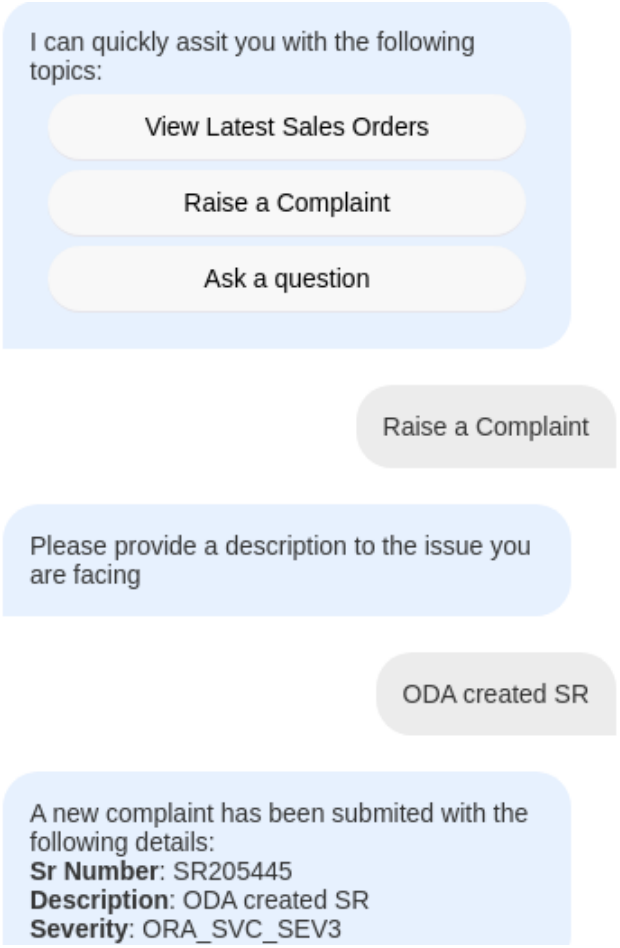
```
var_payload: '{"Title":"${srTitle}"}'
```

Note: you can add as many fields as you need in the var_payload variable, basically create your payload in POSTMAN and just take the working payload and make it 1 line text and you are ready to use it.

```
getDescription:
  component: "System.Text"
  properties:
    prompt: "Please provide a description to the issue you are facing"
    variable: "srTitle"
    keepTurn: false
  transitions:
    actions:
      next: "createComplaint"


createComplaint:
  component: "EC_GenericRESTcall"
  properties:
    var_password: "ZGE88389"
    var_username: "cole.mitchell_zvqy"
    var_ecdomain: "https://adc2-zvqy-fa-ext.oracledemos.com:443/crmRestApi/resources/latest/serviceRequests"
    var_operation: "POST"
    var_ODAchannel: "web"
    var_contentType: "application/vnd.oracle.adf.resourceitem+json"
    var_payload: '{"Title":"${srTitle}","ChannelTypeCd":"ODA"}'
    var_FieldsToExtract: "SrNumber-Sr Number,Title-Description,SeverityCd-Severity"

viewSrDetails:
  component: "System.Output"
  properties:
    text: "A new complaint has been submited with the following details:\n${user.payload}"
    keepTurn: true
  transitions:
    next: "helpAgain"
```



Update a record in Engagement Cloud (PATCH)

The usage for PATCH operation is virtually the same as for POST, the only difference is the EC endpoint used, which needs to specify the record ID it needs to be updated.

```
PATCH:
  component: "EC_GenericRESTcall"
  properties:
    var_password: "<EC WEEKLY PASSWORD>"
    var_username: "<EC USERNAME>"
    var_ecdomain: "<EC ENDPOINT FOR PATCH>"
    var_operation: "PATCH"
    var_ODAchannel: "web"
    var_contentType: "application/vnd.oracle.adf.resourceitem+json"
    var_payload: '<PATCH PAYLOAD>
    var_FieldsToExtract: "<LIST OF FIELDS TO RETURN FROM REST RESPONSE>"
```

Example

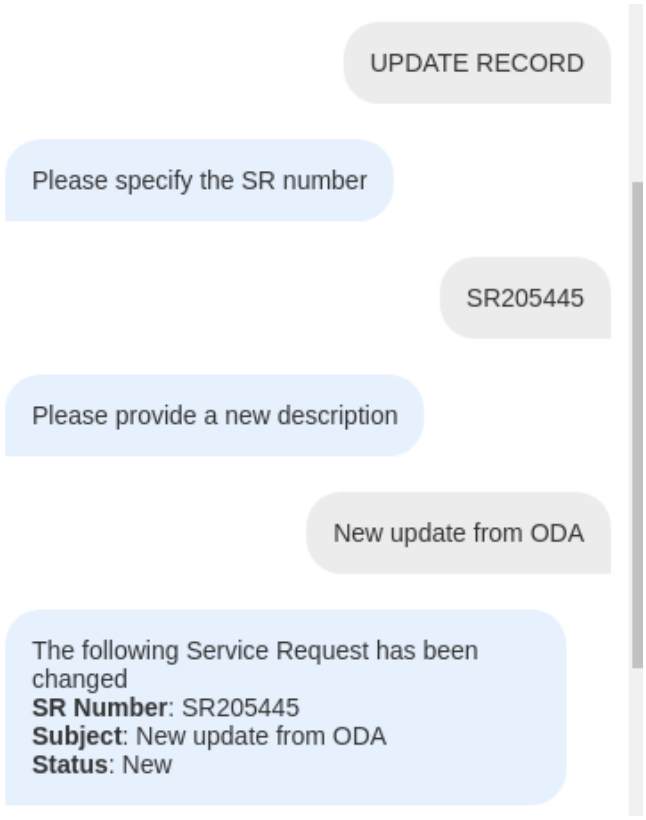
For example, to update a service request, the SR number has to be pass to the EC endpoint and in the payload the field or fields to be updated will be provided.

```
getSrnumber:
  component: "System.Text"
  properties:
    prompt: "Please specify the SR number"
    variable: "SrNumber"
    keepTurn: false
  transitions:
    actions:
      next: "getNewDescription"

getNewDescription:
  component: "System.Text"
  properties:
    prompt: "Please provide a new description"
    variable: "srTitle"
    keepTurn: false
  transitions:
    actions:
      next: "PATCH"

PATCH:
  component: "EC_GenericRESTcall"
  properties:
    var_password: "ZGE88389"
    var_username: "cole.mitchell_zvqy"
    var_ecdomain: "https://adc2-zvqy-fa-ext.oracledemos.com:443/crmRestApi/resources/11.13.18.05/serviceRequests/${SrNumber}"
    var_operation: "PATCH"
    var_ODAchannel: "web"
    var_contentType: "application/vnd.oracle.adf.resourceitem+json"
    var_payload: '{"Title":"${srTitle}"}'
    var_FieldsToExtract: "SrNumber-SR Number,StatusTypeCdMeaning-Status"

viewPATCHresults:
  component: "System.Output"
  properties:
    text: "The following Service Request has been changed\n${user.payload}"
    keepTurn: true
  transitions:
    next: "helpAgain"
```



Search the Knowledge Base in Engagement Cloud

Searching the Knowledge Base the state in the conversation flow has to have the following format.

```
KA:
  component: "EC_GenericRESTcall"
  properties:
    var_password: "<EC WEEKLY PASSWORD>"
    var_username: "<EC USERNAME>"
    var_ecdomain: "<EC ENDPOINT FOR QUESTION ENDPOINT>"
    var_KAcontentEndpoint: "<EC DOMAIN>/km/api/latest/content/answers/"
    var_operation: "KA"
    var_contentType: "application/json"
    var_payload: '{"transactionId":0}'
```

var_ecdomain

Endpoint for question Knowledge search has the following format:

<EC_DOMAIN>/srt/api/latest/search/question?startOverFlag=true&question=**${question}**&expand=true

${question} - this will be a conversation variable that will hold the word or phrase to be search din the Knowledge Base.

var_KAcontentEndpoint

Endpoint for answer rest api of knowledge ec api, has the following format

<EC DOMAIN>/km/api/latest/content/answers/

The component uses the question REST api to search the Knowledge article containing the search term and answer api to retrieve the content of the article.

Usage Example:

```
#Knowledge base search

  askQuestion:
    component: "System.Text"
    properties:
      prompt: "What would you like to ask?"
      variable: "question"
      keepTurn: false
    transitions:
      actions:
        next: "KA"

  KA:
    component: "EC_GenericRESTcall"
    properties:
      var_password: "ZGE88389"
      var_username: "john.dunbar"
      var_ecdomain: "https://adc2-zvqy-fa-ext.oracledemos.com/srt/api/latest/search/question?startOverFlag=true&question=${question}&expand=true"
      var_KAcontentEndpoint: "https://adc2-zvqy-fa-ext.oracledemos.com/km/api/latest/content/answers/"
      var_operation: "KA"
      var_contentType: "application/json"
      var_payload: '{"transactionId":0}'

  viewAnswer:
    component: "System.CommonResponse"
    properties:
      keepTurn: true
      maxPrompts:
      autoNumberPostbackActions:
      metadata:
        responseItems:
        - type: "text"
          text: '${user.payload}'
    transitions:
      actions:
        next: "helpAgain"
```

What would you like to ask?

EPS usage

APPLICATIONS : Used in the ceilings and walls of buildings to isolate the heat. - Thermal insulation for refrigeration cabinets . - Used in sound insulation . - Pipe Insulation
KEY FEATURES : Reduces the consumption of electric power - Reduce the loads imposed adjacent and underlying soils and structures. - lowers the cost of construction because it is easily to handle without need of special equipment - Keeps building appropriate temperatures for a long time - Prevents the intensification of moisture in the walls
INSTALLATION : Use mortar of cement, sand, water and latex or using bituminous protective coating or non solvent adhesive substance