VBCS Lab
Building Your First Application

Introduction

In this lab we will use Quickstarts to create a list of Service Requests along with a Create page and an Edit Page.

Hands on Lab Instructions

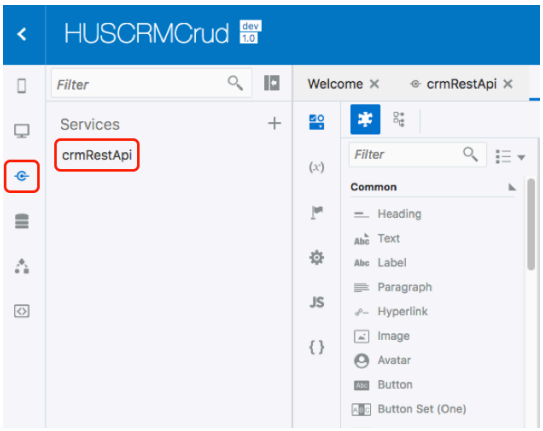
Checking Out the Project

1. Click Import > Import from Git
2. Click Add Credentials
3. Enter https://developer.em2.oraclecloud.com/developer20509-cloud01/ for the DevCS URL and your username and password. Click Finish
4. Enter the following information:
    - DevCS URL with Credentials - https://developer.em2.oraclecloud.com/developer20509-cloud01/
    - Project Selection - VBCS Training
    - Repository Selection - crmCrud.git
    - Branch Selection - basicCrud
    - Application Name - Basic Crud *yourName*
    - Application ID - accept default
5. Click Import.

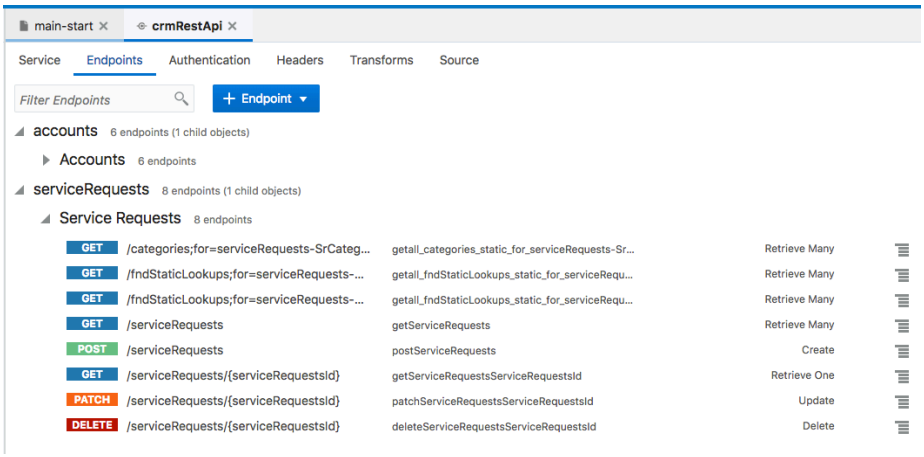
Registering REST APIs

The application already contains the /accounts endpoints from the crmRestApi REST Service. We are going to add the /serviceRequests endpoints to this connection.

1. Click the Service Connections panel and click crmRestApi to open the editor.



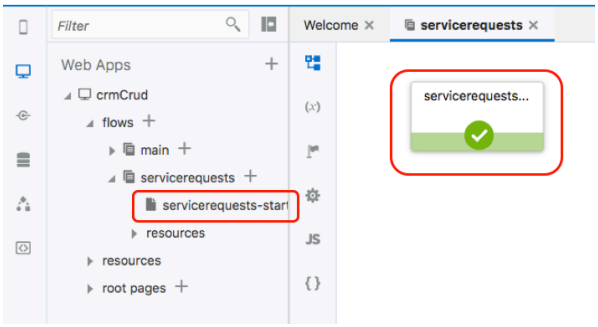
2. Click the Endpoints tab. Then click + Endpoint > From Original Service. Add all of the following:
   - GET /serviceRequests
   - POST /serviceRequests
   - GET /serviceRequests/{serviceRequestsId}
   - PATCH /serviceRequests/{serviceRequestsId}
   - DELETE /serviceRequests/{serviceRequestsId}
   - GET /categories;for=serviceRequests-SrCategoryLOV
   - GET /fndStaticLookups;for=serviceRequests-SeverityCdLookup
   - GET /fndStaticLookups;for=serviceRequests-StatusTypeLookup



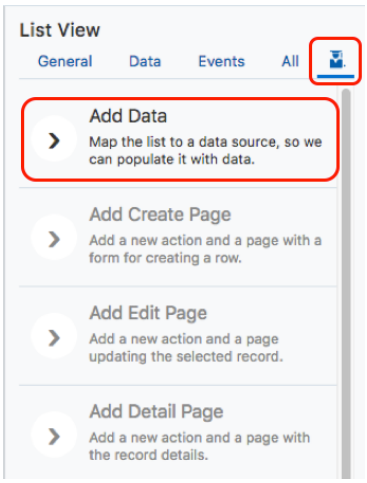
Adding the List

1. Create a new flow called servicerequests

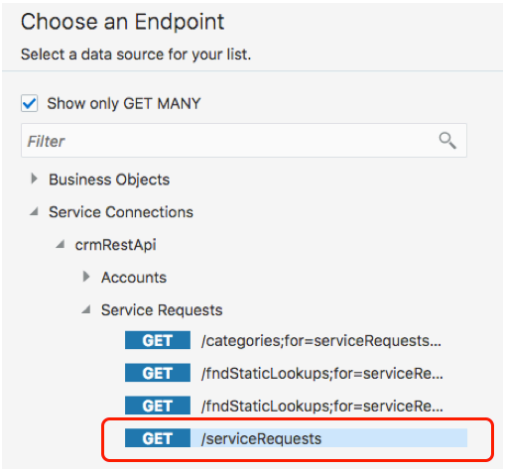
2. Open servicerequests-start in the Page Designer. You can either click the servicerequests-start tile in the Application Flow diagram of the Flow Editor, or navigate to it in the Web Applications pane.



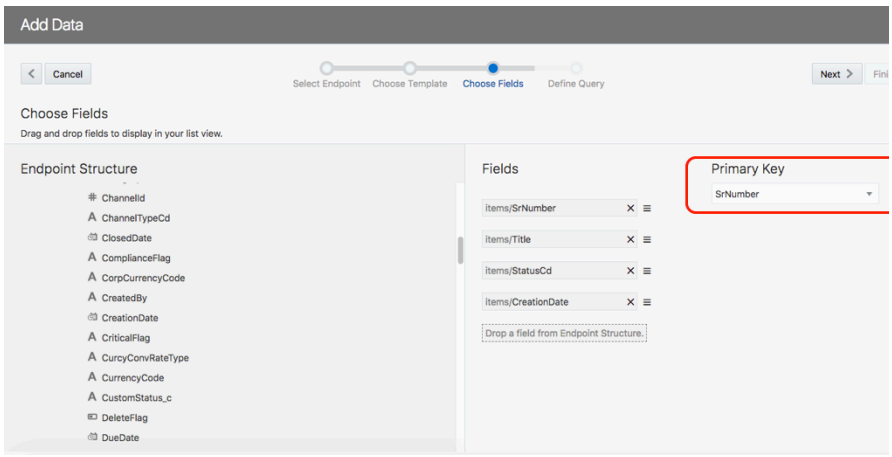
3. Refresh your browser (known bug workaround).
4. Drop a List View onto the page.
5. Select the Live View in the Page Designer. (If you have just dropped it in, it will be selected already.) Then open the Property Inspector and select the Quickstarts page.



6. Select Service Connections > crmRestApi > Service Requests > GET /serviceRequests. Click Next.



7. Choose the Custom list row template, click Next.
8. Select these fields in the left panel of the wizard:
   - SrNumber
   - Title
   - StatusCd
   - CreationDate
9. Set the Item ID Attribute to SrNumber. This is important as the QuickStart incorrectly guesses the ID Attribute and this will cause your app to fail later.

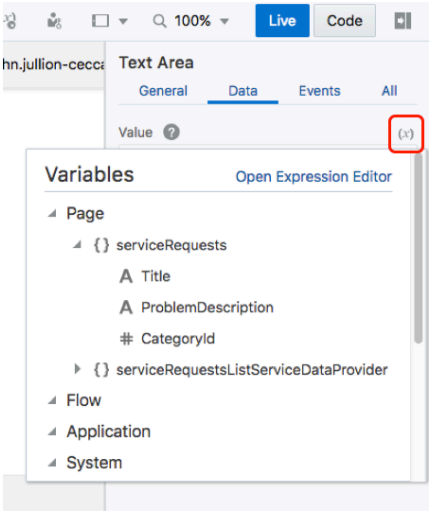


10. Next and Finish

Adding the Create Page

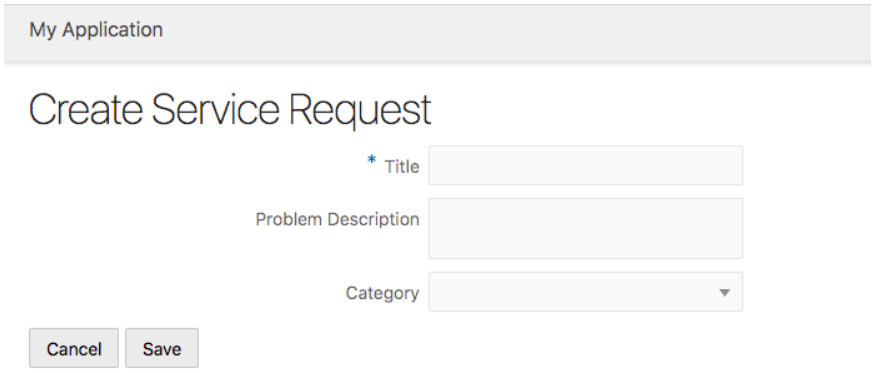
Now we will generate the Create page. Then we will fix up the UI generated by the Quickstart. Specifically, we are going to delete the Problem Description simple Input Text Field and replace it with an Input Text Area. You may be asking, why do I add a Problem Description field if I'm just going to delete it again? Why not generate the form without it and then add the field? The reason is that we want the Quickstart to generate the Type with the ProblemDescription field so it's fetched in the REST call. We could edit the Type to add ProblemDescription after it's been generated, but this is simpler.

1. Select the List View in the Page Designer.
2. In the QuickStart tab of the Property Inspector, click Add Create Page.
3. In the Select Endpoint page, leave the default POST /serviceRequests selected and click Next.
4. Add these fields:
   - Title
   - Problem Description
   - CategoryId
5. Finish and open Create Page
6. Change display name for CategoryId to just Category
7. Delete the ProblemDescription field. Add an Input Text Area field.
   Note: Be careful when selecting that you have selected the individual component and not the enclosing Form Layout. The Property Inspector tells you which component you have selected. If you accidentally delete something, use the Undo button to get it back.
8. In the General tab of the Property Inspector, set the Text Area's display label to Problem Description.
9. In the Data tab, set the value to {{ $page.variables.serviceRequests.ProblemDescription }}. You can manually type in the value or use the Expression Picker to fill it in automatically.



Comment [JC1]: Say how to use variables picker

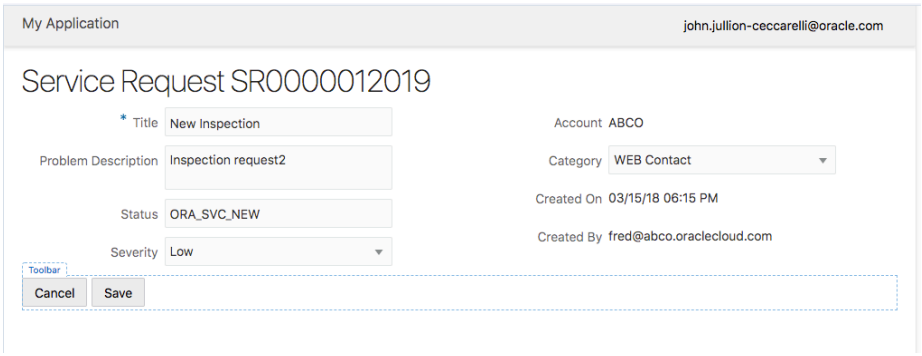
10. Your finished page should look like this:



Adding the Edit Page

Like with the Create Page, we are going to edit this page after it's been created. Note we can use variables and expressions anywhere, including in the <h1> title of the page.

1. Go back to servicerequests-start tab in the editor.
2. Select the list then open the QuickStarts tab of the Property Inspector. Click Add Edit Page
3. Accept the default GET and PATCH /servicerequests endpoints
4. Add these fields:
   - SrNumber
   - Title
   - ProblemDescription
   - CategoryId
   - StatusCd
   - SeverityCd
   - AccountPartyName
   - CreatedBy
   - CreationDate
5. Finish and open Edit Page
6. Delete the SrNumber field.
7. Click the Heading and change the text to {{ "Service Request " + $page.variables.serviceRequestsRecord.SrNumber }}
8. Replace the Problem Description Input Text Field with a Text Area as described above.
9. Fix the display names of the other fields.
10. Click any blank space in the form to select the Form Layout component. In the Property Inspector, set Max Columns to 2


My Application
john.jullion-ceccarelli@oracle.com
Service Request SR0000012019
* Title
New Inspection
Problem Description
Inspection request2
Status
ORA_SVC_NEW
Severity
Low
Account
ABCO
Category
WEB Contact
Created On
03/15/18 06:15 PM
Created By
fred@abco.oraclecloud.com
Toolbar
Cancel
Save