

**Aprimo DAM Collections Salesforce Lightning Component Guide**

**Last Updated: March 19, 2021**

Contents

[1. Introduction 3](#_Toc66977122)

[2. Installation Instruction 3](#_Toc66977123)

[Prerequisites 3](#_Toc66977124)

[3. Configuration Options 5](#_Toc66977125)

[Search DAM Paging 5](#_Toc66977126)

[Salesforce Object and Fields 5](#_Toc66977127)

[Aprimo Collection IDs 7](#_Toc66977128)

[Authorization 7](#_Toc66977129)

[Selection Login 8](#_Toc66977130)

# Introduction

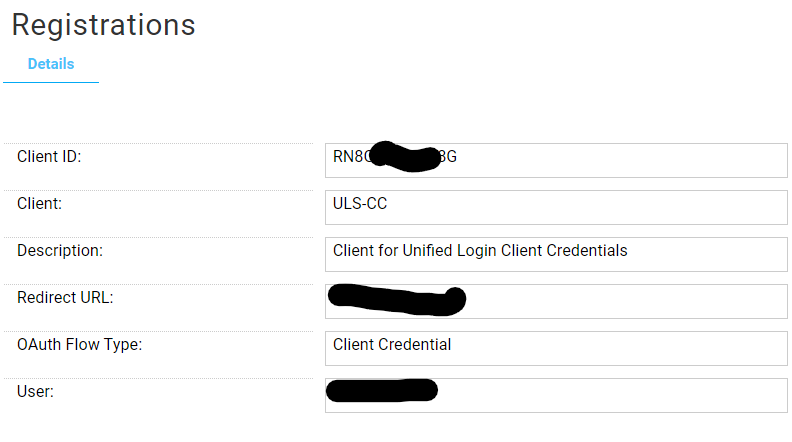
The Aprimo DAM Collections Salesforce Lightning Component can be placed on Salesforce record pages and will surface the contents of an Aprimo DAM Collection in Salesforce. The collection chosen to be surfaced is configurable and can be set up based on an opportunity’s stage, an account’s industry, a campaign’s type, or other options.

# Installation Instruction

## Prerequisites

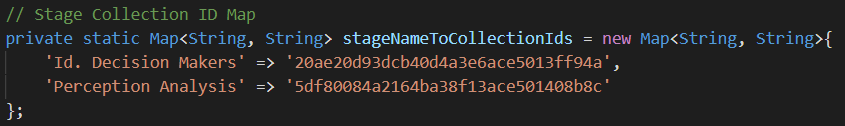
* Pull the code from the Aprimo Connect GitHub
* Permissions to create new Aprimo client registrations
* Permissions to deploy code to the Salesforce instance
* Permissions to access the ‘Edit Page’ functionality on Salesforce record pages
* Permission to access the Remote Site Settings in Salesforce Setup

1. Download the LWC and Classes folders from the github repo
2. In the “aprimocollections.js” file replace the value of the “tenant” variable with your specific Aprimo environment name
3. In the “AprimoCollectionsConnector.cls” file replace the “aprimoTenant” variable with your specific Aprimo environment name
4. Go to your Aprimo administration settings and configure a new Aprimo registration
   1. The secret is non-recoverable. So do not forget it. Make sure the authorization flow is set to Client Credentials

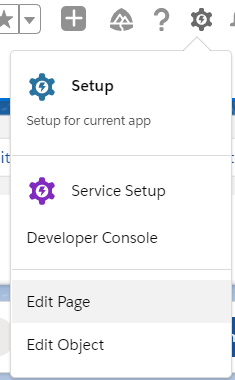


* 1. In the “AprimoCollectionsConnector.cls” file replace the “aprimoClientId” and “aprimoClientSecret” with the appropriate value from your new Aprimo client registration

1. In the “AprimoCollectionsConnector.cls” file alter the collections ids in the stageNameToCollectionIds to reflect collections that exist in your Aprimo environment
   1. If this configuration is not what you are looking for, this document will detail where and how to change configuration later on



1. Deploy the following files:
   1. aprimocollections.html
   2. aprimocollections.js
   3. aprimocollections.js-meta.xml
   4. AprimoCollectionsConnector.cls
   5. AprimoCollectionsConnector.cls-meta.xml
2. Navigate to an Opportunity object in Salesforce
3. Enter the ‘Edit Page’ functionality



1. Search “Aprimo” and drag the new component onto the page
2. Navigate to the Salesforce Setup > Security > Remote Site Settings
3. Whitelist your Aprimo order download url
   1. EX: https://[your environment].dam.aprimo.com
4. Whitelist your Aprimo access token url
   1. EX: https://[your environment].aprimo.com

# Configuration Options

## How to Use Authorization

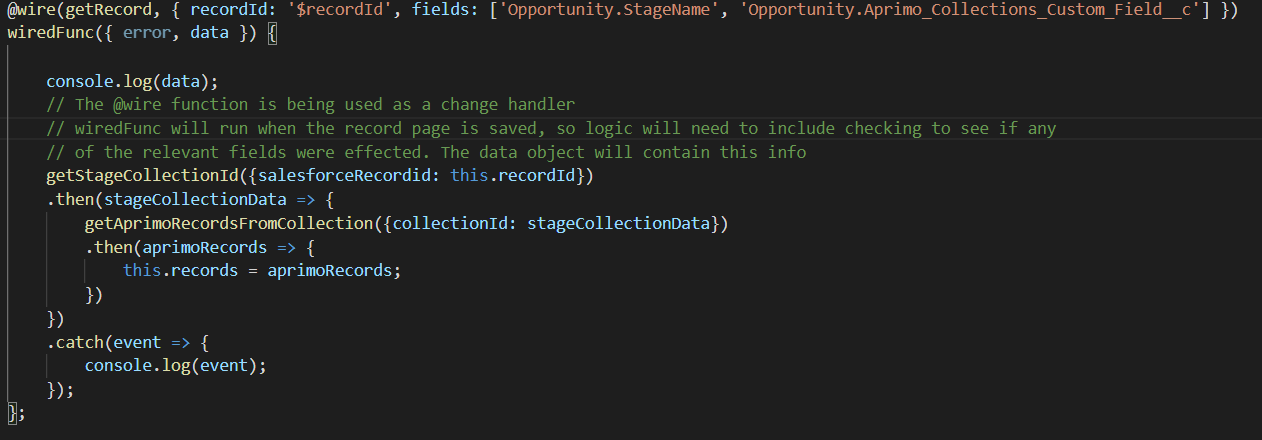
Navigate to the AprimoCollectionsConnector.cls file. The following variables are the authorization relevant variables:

* aprimoTenant
* aprimoClientId
* aprimoClientSecret

If the Aprimo environment needs to change, change the aprimoTenant variable. If a new client registration is created, then the aprimoClientId and aprimoClientSecret need to change.

## How to Use Different Salesforce Objects and Fields

Navigate to the aprimocollections.js file. You define which Salesforce object to monitor in the @wire decorated function.



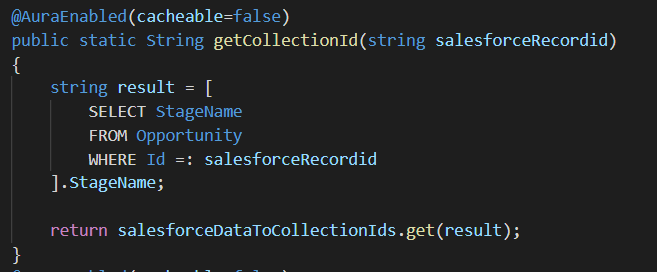
When a record page is changed in Salesforce the “wiredFunc” will receive an object containing data on the fields defined in the fields attribute. In the above screenshot that is the Opportunity’s StageName field and an Opportunity custom field called Aprimo\_Collections\_Custom\_Field\_\_c. This fields array can contain either a single or multiple fields, including custom fields. If the connector needs to be configured with the Account object’s Industry field, you would set the fields array to [‘Account.Industry’] and the same thing for other Salesforce objects.

EX:

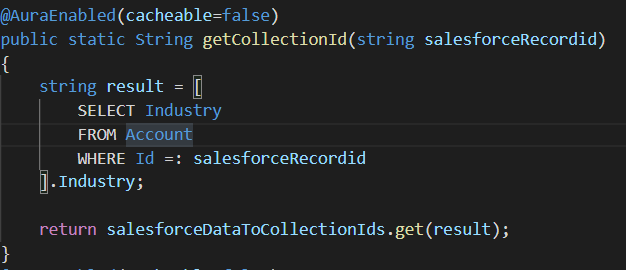




The getCollectionId() function in AprimoCollectionsConnector.cls is also relevant in this configuration.

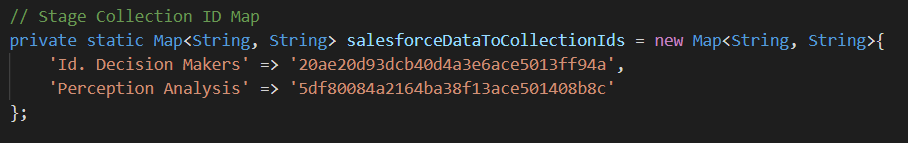


The wired function will call this function to query Salesforce to get the data of the relevant metadata fields. If this need to be changed to support the Account object it may look like the screenshot below.

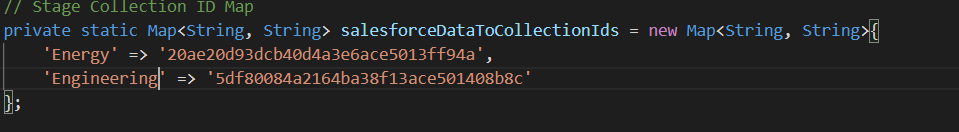


## How to Change Aprimo Collection IDs

Navigate to the AprimoCollectionsConnector.cls file. This class contains the Map object that associates Salesforce data to Aprimo collection IDs.

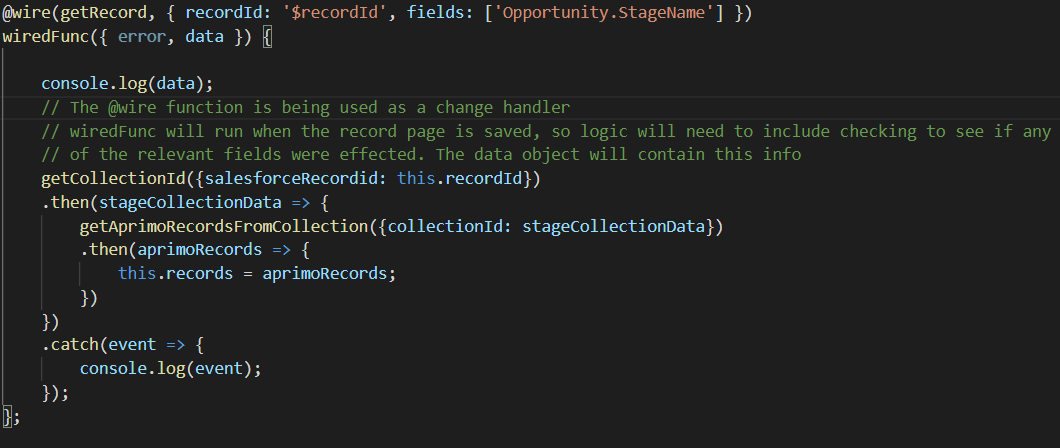


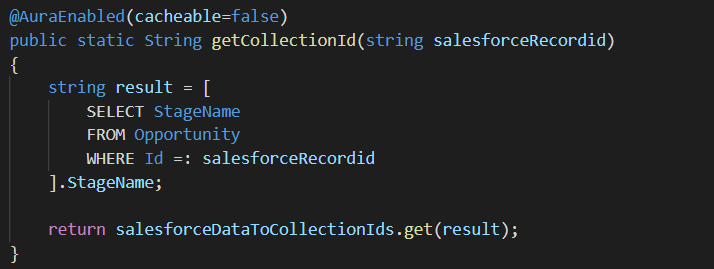
Set the values appropriate to your configuration. If the connector is going to base the collection off of an Account’s Industry field, then the Map object would look something more like this:



## How to Change Selection Logic

The login for determining which collection will be selected for surfacing is decided in the @wire decorated function in aprimocollections.js as well as the getCollectionId() function in AprimoCollectionsConnector.cls

**@wire**

**getCollectionId()**

If your configuration required more advanced selection logic that login will be placed in these 2 functions.