APTTUS®

Intelligent Workflow and Approvals on Salesforce Summer 2019

SOAP API Guide

07/30/2019

Table of Contents

About This Guide

With the Intelligent Workflow Approvals on Salesforce SOAP API Guide, you can find out how Application Programming Interfaces (API) for you to extend the features offered by Apttus. These extensions add more functionality to the features available through configuration on Salesforce.

Topic	Description
What's Covered	This guide is designed to provide APIs provided to work with Intelligent Workflow and Approvals objects.
Primary Audience	Admin users responsible for setting up approvals and users for Apttus Intelligent Workflow Approvals.
IT Environment	Refer to the latest Apttus Intelligent Workflow Approvals on Salesforce Release Notes for information on System Requirements and Supported Platforms.
Updates	For a comprehensive list of updates to this guide for each release, see the What's New topic.
Other Resources	• Intelligent Workflow Approvals Administrator Guide: Refer to this guide for basic admin tasks and end user experience.

This guide describes the APIs provided to work with Intelligent Workflow and Approvals objects.

Before using Intelligent Workflow Approvals, you must be familiar with the following:

- Basic Salesforce administration
- Salesforce and Apttus terms and definitions

What's New

The following table lists changes in the documentation to support each release.

Release	Topic	Description
Summer 2019	N/A	No new API added.
Spring 2019	N/A	No new API added.
Winter 2018	Adding Comments to Approval Requests	Added API info.
	Approving Approval Requests	Added API info.
	Cancelling Approvals	Added API info.
	Checking If an Approval is Required	Added API info.
	Checking if User is Authorized to Approve/Reject	Added API info.
	Creating Ad-hoc Approvals	Added API info.
	Escalating Approval Requests	Added API info.
	Previewing Approvals	Added API info.
	Reassigning Approval Requests	Added API info.
	Rejecting Approval Requests	Added API info.
	Retrieving Add Comment Page URL	Added API info.
	Retrieving Approval History	Added API info.
	Retrieving Approval Page URL	Added API info.

Release	Topic	Description
	Retrieving Approve or Reject Page URL	Added API info.
	Retrieving Reassign Page URL	Added API info.
	Submitting an Approval Request	Added API info.
	Taking Ownership of an Approval Request	Added API info.

About Apttus Intelligent Workflow Approvals

Intelligent Workflow Approvals enables you to trigger an approval request for any object and send an email notification to the concerned stakeholders.

Using Intelligent Workflow Approvals, you can set an approval request on an object header, line item, or both. Once an approval request is triggered, an approver can approve or reject a request via email or via Salesforce by logging into their org and viewing all the available requests in the My Approval or Home Page.Apttus Quote-to-Cash enhances the visibility and agility of all business processes from initiating a prospective customer, to preparing a quote to managing contracts and renewals with its intelligent workflow. By implementing with your Salesforce1 data, you can improve enterprise-wide collaboration, streamline approvals, and significantly cut down sales cycles.

Apttus provides over a dozen intelligent workflow capabilities that can help you get much more out of standard and custom Salesforce1 objects. You can gain a greater control over your sales process, discounts and exceptions so you can get higher productivity and more wins.

Key Capabilities for Intelligent Workflow Approvals

- Workflow Preview: See past and future workflow steps and know where the approval steps are going.
- What-If Scenario Planning: Change parameters in your fields to modify where approvals go.
- Out of Office Assistant: Assign backups for approvals in your absence, define the time frames and then all the approvals will go to the backup with full audit trails.
- Multiple Conditions Approvals: Define a rule with multiple conditions, from any object and drive an approval to multiple people.
- Dynamic Rules Table: Set up your own rules, conditions, and exceptions in a table and the workflow will automatically reference it without having to set up individual workflows for each rule.

SOAP API Guide Structure

Document Setup

The Apttus CPQ and CM API reference Guide is divided into two sections: *API Reference* and *Scenarios*.

API Reference	The *API reference* section details the APIs that you can use to manipulate Apttus objects through API calls and passing parameters. The API reference section also includes some code samples.
Scenarios	The *Scenarios* section details examples of the APIs you require to complete a specific task such as, adding products and constraint rules to the cart. The scenarios are classified by theme. You can refer to the generic examples of scenarios to identify the calls you can use to achieve your objective.

API Standards and Development Platforms

Apttus APIs are based on Salesforce APIs and use the same standards and platforms.

Standards

Name	Reference
Simple Object Access Protocol (SOAP) 1.1	http://www.w3.org/TR/2000/NOTE-SOAP-20000508
Web Service Description Language (WDSL) 1.1	http://www.w3.org/TR/2001/NOTE-wsdl-20010315
WS-I Basic Profile 1.1	http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html

Development Platforms

Apttus SOAP API works with standard SOAP development environments. For a list of compatible development platforms, see Salesforce Developer Force API details.

> **Note**
> A later version of this content will contain sample code.

Field Types

Apttus APIs use a subset of the supported data and field types on Salesforce.

The following table lists the APIs that Apttus provides. For a comprehensive list of all field types supported by Salesforce, see Salesforce Data Types.

Type	Description
Boolean	The **Boolean** field has a `true` (or 1) or `false` (or 0) value.
Data object	The **Data Object** field is an ID type and is represented by `CPQ.nnDO` in this document.

Type	Description
Date	The **Date** field contains date values only and do not contain relevant time values. Time in a date field is always set to midnight in the UTC time zone. If you want a timestamp you must use a **dateTime** field.
Decimal	The **Decimal** field provides an exact numeric value and you can arbitrarily size the precision and scale of the value.
ID	The **ID** field is an alphanumeric field that acts like the primary key for a specific record associated with an object. The ID value includes a three-character code that identifies which object the record is associated with. The ID for a specific record does not change. For some objects, this field may also be a`reference`type value, which contains the ID value for a related record. They are identified by field names ending in 'Id', such as`priceListId`. The ID field acts like foreign keys and their values can be changed using an`update()`call.
Integer	The **Integer** field contains whole numbers only. There are no digits after the decimal.
List	The **List** field includes a fixed set of values from which you must select a single value. Picklists are available as drop-down lists. If a picklist is unrestricted, the API does not limit entries to only currently active values.
String	The **String** field contains text and may have differing length restrictions based on the data you store in the specific field. For instance,`City`may be limited to 50 characters, while`AddressLine1`is limited to 255 characters.

IWA Web Service

Apttus IWA APIs are based on Salesforce APIs and use the same standards and platforms.

You can invoke Apttus IWA APIs belonging to the ApprovalsWebService Class from the following command:

```
Apttus_Approval.ApprovalWebService.<Name of the Function>
where the name of the function is API Name and it parameters.
```

Approvals Web Serice class contains the following APIs:

Adding Comments to Approval Requests

This API enables you to add comments to approval requests. This API accepts the ID of the approval request and comments as input parameters.

addCommentsToRequest

Request Parameters			
Name	**Type**	**Required?**	**Description**
requestId	ID	Yes	ID of the approval request.
comments	String	Yes	Comments to add to the approval request.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the comments are added to the approval request.

Approving Approval Requests

This API approves the approval request. This API accepts the ID of the approval request and comments as input parameters.

approveRequest

Request Parameters			
Name	**Type**	**Required?**	**Description**
requestId	ID	Yes	ID of the approval request.
comments	String	Yes	Comments to add to the approval request.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the approval request is approved successfully.

Cancelling Approvals

This API cancels the entire approval process associated with the context object. Currently, assigned approvers will be notified via email. This API accepts the object type and the object ID of the approval request as input parameters.

cancelApprovals

Request Parameters			
Name	**Type**	**Required?**	**Description**
sObjectId	ID	Yes	ID of the approval context object.
sObjectType	String	Yes	Type of the approval context object.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the approval is cancelled successfully.

Checking If an Approval is Required

This API determines if given context object, in its current state, requires approval. This API accepts ID of the context object and type of the context object as request parameters.

isApprovalRequired

Request Parameters			
Name	**Type**	**Required?**	**Description**
sObjectId	ID	Yes	ID of the business context object.
sObjectType	String	Yes	Type of the context object.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if an approval is required.

checkIfApprovalRequired

This API determines if given list of context objects, in its current state, requires approval. It returns list of true/false values corresponding to the order in which ids were input.

Request Parameters			
Name	**Type**	**Required?**	**Description**
sObjectIds	List<ID>	Yes	List of IDs of business context objects.

Response Parameter		
Name	**Type**	**Description**
result	List<Boolean>	Returns a list of boolean indicators. The indicator is true if an approval is required.

checkIfApprovalRequired2

This API determines if a context header object (For example: Agreement) or context child object (For example: AgreementLineItems), in its current state, requires approval. It returns a single consolidated list with updated approval status (a01U0000017FRf9IAG__Approval Required) against the corresponding Id.

> headerIdStatus & childIdStatus are constructed as follows:
> Example: (a01U0000017FRf9IAG__None) - ID and Approval_Status__c value of the object, separated by '__' (double-underscore)

Request Parameters

Name	Type	Required?	Description
headerIdStatus	String	Yes	Header ID and approval status.
childIdStatusList	List<String>	Yes	List of approval status ID of child objects.
modifiedChildObjectIds	Lsit<ID>	Yes	List of IDs of modified child objects.

Response Parameter

Name	Type	Description
result	List<String>	Returns a list of the header ID and child object id approval statuses separated by '__'.

Checking if User is Authorised to Approve/Reject

This API checks if the context user is authorized to approve or reject an approval request. This API accepts the request ID of the approval as an input parameter.

canApproveRejectRequest

Request Parameters

Name	Type	Required?	Description
requestId	ID	Yes	ID of the approval request.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the user is authorized to approve or reject a request.

Creating Ad-hoc Approvals

This API creates an ad-hoc approval step. This API accepts Context Request ID and creates an approval step for it.

createAdhocApproval

Request Parameters			
Name	**Type**	**Required?**	**Description**
contextRequestId	ID	Yes	ID of the context request in relation to which you want to create an approval step.
stepLable	String	Yes	Text label of the step.
assigneeType	String	Yes	Type of the assignee. Assignee can be: • User • Role • Queue • Related User
assigneeID	ID	Yes	ID of the assignee.
submissionComment	String	Yes	Comments for approval submission.
relativeLocationToContext	String	Yes	Relative Location can be: • Before • After • In-parallel
dependsOnContextRequestIdList	List<ID>	Yes	List of request identifiers that the ad-hoc approval depends on.
dependsOnMeContextRequestIdList	List<ID>	Yes	List of request identifiers that depend on this ad-hoc approval.
sendEmail	Boolean	No	Set it as true to send an email request. The default value is true.
notifyOnly	Boolean	No	The default value is false

Response Parameter		
Name	**Type**	**Description**
result	Boolean	The API returns a true value if the ad-hoc approval is created successfully

Enabling auto re-approvals on Proposal and Proposal Line Items

This API takes several parameters and updates the reapproval data in the process instance so that proposal line items can be automatically reapproved when resubmitted after finalizing new items in the cart. Note that this API is available in the base Approvals package and can be used with the optional unmanaged custom package "Apttus Proposal Auto-Reapprovals" available on request with Apttus Release Management. This package contains the complete sample code that can used along with the API in the base package to make auto-reapprovals work with Proposals and Proposal Line Items.

updateApprovalData

Request Parameters			
Name	**Type**	**Required?**	**Description**
instanceId	ID	Yes	The process instance id that is being currently executed in the system.
sObjectType	String	Yes	ThesObjecttypethat is used to identify the object type.
contextObjId	ID	Yes	The ID of the context object.
oldContextObjIds	List<ID>oldContextObjIds	Yes	The list of old approval data context object ids that are to be replaced with the new ones.
newContextObjIds	List<ID> newContextObjIds	Yes	The list of new approval data context object id store place with.

Response Parameter			
Name	**Type**	**Required**	**Description**
Ok	Boolean	Yes	Defines if the operation was successful. Returns *true* if yes, otherwise *false*.

Code Sample

The sample below enables you to execute auto re-approvals over Proposal Line Items by considering the input parameters, such as process instance, sObject type, context object, and lists of old and new context object ids. These IDs represent the Proposal Line Item IDs before and after the cart is finalized. You can use this API in a scenario where a Quote/Proposal is approved but the Approvals Manager decides to change the discount from 10 to 15% on a Proposal Line Item, you can resubmit the approval request for this for re-approval. The system honors the auto-reapproval criteria, using which a request once approved by certain assignees are auto-approved.

See the complete code included in the package to understand how this works, especially CustomProposalLineItemSupport.cls. Note that the code in this package is based on a trigger on the Proposal Line Item object which saves the old ids of the Proposal Line Items before they are deleted as well as the new ids of the Proposal Line Items that get inserted after the cart is finalized. You can extend this concept for other custom child objects to the proposal object by creating a trigger on them and using the same pattern in the sample code to save ids before and after new items are created.

```
/**
*Apttus Approvals Management
*CustomProposalLineItemTrigger
*@2017 Apttus Inc. All rights reserved.
*/


trigger CustomProposalLineItemTrigger on Apttus_Proposal__Proposal_Line_Item__c (before
delete, after insert)
{
    if (Trigger.isBefore && Trigger.isDelete)
        {
       // save map of old line items to attributes before they are deleted
                CustomProposalLineItemSupport.doBeforeDelete(Trigger.old,
Trigger.oldMap);
        }
        if (Trigger.isAfter && Trigger.isInsert)
        {
        // update reapproval data with new line items after they are inserted
                CustomProposalLineItemSupport.doAfterInsert(Trigger.new,
Trigger.newMap);
    }
}
/**
*Apttus Approvals Management
*CustomProposalLineItemSupport
* @2017 Apttus Inc. All rights reserved.
*/

public with sharing class CustomProposalLineItemSupport extends CustomApprovalsConstants
{
        // line item types
    private static final String LINETYPE_PRODUCT = 'Product/Service';
    private static final String LINETYPE_OPTION = 'Option';

    // proposal
        private static final String PROPOSAL_SOBJECT_TYPE =
'Apttus_Proposal__Proposal__c';
    private static ID quoteId = null;

    // associated process instance
        private static Apttus_Approval__ApprovalProcessInstance__c quoteProcessInstance
= null;

    // map of old line item ids to attribute key
    private static Map<ID,String> oldLineId2KeyMap = new Map<ID,String>();

    // map of new line item attribute key to id
    private static Map<String,ID> newLineKey2IdMap = new Map<String,ID>();
```

```
    // map of old line item ids to new line item ids
    private static Map<ID,ID> lineOldId2NewIdMap = new Map<ID,ID>();

    /**
     * Process old ProposalLineItems before they are deleted when a cart is finalized
     *        @param oldLineItems - a list of the old versions of the sObject records
     *  @param oldLineItemsMap - a map of IDs to the old versions of the sObject records
     */

    public static void doBeforeDelete( List<Apttus_Proposal__Proposal_Line_Item__c>
oldLineItems, Map<ID, Apttus_Proposal__Proposal_Line_Item__c> oldLineItemsMap)
{
        // iterate over line items about to deleted
        for (Integer i=0; i<oldLineItems.size(); i++)
        {
                Apttus_Proposal__Proposal_Line_Item__c oldLineItem = oldLineItems[i];
        // get quote from line item
        quoteId = oldLineItem.Apttus_Proposal__Proposal__c;
    // get line item attributes
                String lineNumber =
String.valueOf(oldLineItem.Apttus_QPConfig__PrimaryLineNumber__c);
                String lineType = oldLineItem.Apttus_QPConfig__LineType__c;
                String productId = null;

                if (lineType == LINETYPE_PRODUCT)
                {
                        productId = oldLineItem.Apttus_Proposal__Product__c;
                }
                else if (lineType == LINETYPE_OPTION)
                {
                        productId = oldLineItem.Apttus_QPConfig__OptionId__c;

        //productId = oldLineItem.Apttus_QPConfig__ProductOptionId__c;
                }

                String chargeType = oldLineItem.Apttus_QPConfig__ChargeType__c;
                // create attribute key
                        String oldLineKey = lineNumber + ':' + productId + ':' +
chargeType;
                // save in map
                        oldLineId2KeyMap.put(oldLineItem.Id, oldLineKey);
    }
}
/**
* Process new ProposalLineItems after they are inserted when a cart is finalized
* @param newLineItems - a list of the new versions of the sObject records
* @param newLineItemsMap - a map of IDs to the new versions of the sObject records
*/
```

```
        public static void doAfterInsert(List<Apttus_Proposal__Proposal_Line_Item__c>
newLineItems, Map<ID, Apttus_Proposal__Proposal_Line_Item__c> newLineItemsMap)
        {
                // iterate over line items about to deleted
        for (Integer i=0; i<newLineItems.size(); i++)
                {
                        Apttus_Proposal__Proposal_Line_Item__c newLineItem =
newLineItems[i];

                // get quote from line item
                        quoteId = newLineItem.Apttus_Proposal__Proposal__c;

                // get line item attributes
                        String lineNumber =
String.valueOf(newLineItem.Apttus_QPConfig__PrimaryLineNumber__c);
                        String lineType = newLineItem.Apttus_QPConfig__LineType__c;
                        String productId = null;

                        if (lineType == LINETYPE_PRODUCT)
                        {
                                productId = newLineItem.Apttus_Proposal__Product__c;
                        }
                        else if(lineType == LINETYPE_OPTION)
                        {
                                productId = newLineItem.Apttus_QPConfig__OptionId__c;
                                //productId =
newLineItem.Apttus_QPConfig__ProductOptionId__c;
                        }
                        String chargeType = newLineItem.Apttus_QPConfig__ChargeType__c;

                        // create attribute key
                                String newLineKey = lineNumber + ':' + productId + ':' +
chargeType;
                        // save in map
                                newLineKey2IdMap.put(newLineKey, newLineItem.Id);

        }
                // create map of old line item ids to new line item ids
                        Set<ID> oldLineItemIds = oldLineId2KeyMap.keySet();
                        List<ID> newLineItemIds = new List<ID>();

                        for (String oldLineItemId : oldLineItemIds)
                        {
                                // get attribute key
                                String attrKey = oldLineId2KeyMap.get(oldLineItemId);
                                // lookup key in new line items map
                                String newLineItemId = null;
                                if(newLineKey2IdMap.containsKey(attrKey))
                                {
                                        newLineItemId = newLineKey2IdMap.get(attrKey);
                                        newLineItemIds.add(newLineItemId);
```

```
                                        // associate old key with new one
                                        lineOldId2NewIdMap.put(oldLineItemId,
newLineItemId);
                                }

                        }

      // update reapprovals data by calling API in approvals package

                List<ID> oldContextObjIds = new List<ID>(oldLineItemIds);
                List<ID> newContextObjIds = newLineItemIds;

        system.debug(LoggingLevel.INFO,'sObjectType='+PROPOSAL_SOBJECT_TYPE);

        system.debug(LoggingLevel.INFO,'contextObjId='+quoteId);

        system.debug(LoggingLevel.INFO,'oldContextObjIds='+oldContextObjIds);

        system.debug(LoggingLevel.INFO,'newContextObjIds='+newContextObjIds);

        if (!oldContextObjIds.isEmpty() && !newContextObjIds.isEmpty() &&
oldContextObjIds.size() == newContextObjIds.size())
                {
            // get process instance associated with the old quote
                        Apttus_Approval__ApprovalProcessInstance__c instanceSO =
getProcessInstance(quoteId);
                        system.debug(LoggingLevel.INFO,'instanceSO='+instanceSO);

            // call API to update reapproval data
            Boolean ok =
Apttus_Approval.ApprovalsWebService.updateApprovalData(instanceSO.Id,
PROPOSAL_SOBJECT_TYPE, quoteId, oldContextObjIds, newContextObjIds);


system.debug(LoggingLevel.INFO,'Apttus_Approval.ApprovalsWebService.updateApprovalData='+
ok);

        }
    }

/**
* Get process instance for the given proposal id
* @param proposalId
* @return process instance object
*/

        private static Apttus_Approval__ApprovalProcessInstance__c getProcessInstance(ID
proposalId)
        {
```

```
                List<Apttus_Approval__ApprovalProcessInstance__c> instanceList = [select
Id, Name, LastModifiedDate, LastModifiedById, LastActivityDate, CreatedDate,
CreatedById, Apttus_Approval__Status__c, Apttus_Approval__StartTime__c,
Apttus_Approval__ReassignmentEmailTemplate__c,
Apttus_Approval__PrevProcessInstanceId__c, Apttus_Approval__NotifyOnlyEmailTemplate__c,
Apttus_Approval__InstanceNumber__c, Apttus_Approval__EscalationEmailTemplate__c,
Apttus_Approval__EndTime__c, Apttus_Approval__Data__c,
Apttus_Approval__ConsolidationVersionNumber__c,
Apttus_Approval__CancellationEmailTemplate__c, Apttus_Approval__BusinessObjectType__c,
Apttus_Approval__BusinessObjectLink__c, Apttus_Approval__BusinessObjectId__c,
Apttus_Approval__AssignmentEmailTemplate__c, Apttus_Approval__ApprovalProcessId__c From
Apttus_Approval__ApprovalProcessInstance__c where Apttus_Approval__BusinessObjectId__c
= :proposalId order by CreatedDate DESC limit 1];
                if ( ! nullOrEmpty(instanceList))
                {
                        return instanceList[0];
        }

                return null;
    }
/**
* Checks if the given string value is null or empty.
* @param strValue the string to check
* @return <code>true</code> if the string value is null or empty, <code>false</code>
otherwise
*/
        public static Boolean nullOrEmpty(String strValue)
        {
                // check if null or zero length string
                return (strValue == null || strValue.trim().length() == 0);
    }

/**
* Checks if the given list of objects is null or empty.
* @param objList the list of objects to check
* @return <code>true</code> if the list is null or empty, <code>false</code> otherwise
*/
        public static Boolean nullOrEmpty(List<Object> objList)
        {
                // check if null or empty
                return (objList == null || objList.isEmpty());
        }
}
```

Escalating Approval Requests

System follows the escalation scheme specified in the current approval process. Approval Request is reassigned accordingly. This API accepts the ID of the approval request as an input parameter.

escalateRequest

Request Parameters			
Name	**Type**	**Required?**	**Description**
requestId	ID	Yes	ID of the approval request.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the approval request is escalated.

Previewing Approvals

This API generates required approval requests for the given context object. You must query the approval request table to fetch them. This API accepts the object type and the object ID of the approval request as input parameters.

previewApprovals

Request Parameters			
Name	**Type**	**Required?**	**Description**
sObjectId	ID	Yes	ID of the approval context object.
sObjectType	String	Yes	Type of the approval context object.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the API is executed successfully.

Reassigning Approval Requests

This API reassigns the approval request. This API accepts the ID of the approval request, Id of the assignee and comments as input parameters.

reassignRequest

Request Parameters			
Name	**Type**	**Required?**	**Description**
requestId	ID	Yes	ID of the approval request.
toAssigneeId	ID	Yes	ID of the new assignee.
comments	String	Yes	Comments

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the approval request is reassigned.

Rejecting Approval Requests

This API rejects the approval request. This API accepts the ID of the approval request and comments as input parameters.

rejectRequest

Request Parameters			
Name	**Type**	**Required?**	**Description**
requestId	ID	Yes	ID of the approval request.
comments	String	Yes	Comments to add to the approval request.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the approval request is rejected successfully.

Retrieving Add Comment Page URL

This API gets the URL of the add comments page for the given approval request. This API accepts the approval request object as input parameter.

getAddCommentPageUrl

Request Parameters			
Name	**Type**	**Required?**	**Description**
Approval_Request__c requestSO	Object	Yes	The approval request object.

Response Parameter		
Name	**Type**	**Description**
addCommentPageURL	String	The URL of the add comment page.

Retrieving Approval History

This API gets the approval history associated with the approval request. This API accepts the approval request object as an input parameter and returns a list of the approval history objects.

getApprovalHistory

Request Parameters			
Name	**Type**	**Required?**	**Description**
Approval_Request__c requestSO	Object	Yes	The approval request object.

Response Parameter		
Name	**Type**	**Description**
approvalHistorySO	List<Approval_Request__c>	List of approval history objects.

Retrieving Approval Page URL

This API gets the approvals page URL for the given approval request. This API accepts the approval request object as input parameter.

getMyApprovalsPageUrl

Request Parameters			
Name	**Type**	**Required?**	**Description**
Approval_Request__c requestSO	Object	Yes	The approval request object.

Response Parameter		
Name	**Type**	**Description**
myApprovalPageURL	String	The URL of the approval page.

Retrieving Approve or Reject Page URL

This API gets the approve/reject page URL for the given approval request. This API accepts the approval request object as input parameter.

getApproveRejectPageUrl

Request Parameters			
Name	**Type**	**Required?**	**Description**
Approval_Request__c requestSO	Object	Yes	The approval request object.

Response Parameter		
Name	**Type**	**Description**
approveRejectPageURL	String	The URL of the approve/reject page.

Retrieving Reassign Page URL

This API gets the reassign page URL for the given approval request. This API accepts the approval request object as input parameter.

getReassignPageUrl

Request Parameters			
Name	**Type**	**Required?**	**Description**
Approval_Request__c requestSO	Object	Yes	The approval request object.

Response Parameter		
Name	**Type**	**Description**
reassignPageURL	String	The URL of the reassign page.

Submitting an Approval Request

With this API, you can submit an approval request. This API accepts the object type and the object ID of the approval request as input parameters.

submitForApprovals

Request Parameters			
Name	**Type**	**Required?**	**Description**
sObjectId	ID	Yes	ID of the approval context object.
sObjectType	String	Yes	Type of the approval context object.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the API is executed successfully.

submitForApprovalsWithComments

With this API, you can submit an approval context with comments.

Request Parameters			
Name	**Type**	**Required?**	**Description**
sObjectId	ID	Yes	ID of the approval context object.
sObjectType	String	Yes	Type of the approval context object.
comments	Object	Yes	The SubmissionComments object.

SubmissionComments Class

```
global with sharing class SubmissionComments {
 public static final String PROCESS_LEVEL_COMMENTS = 'Process';
 public static final String STEP_LEVEL_COMMENTS = 'Step';
 // comments level
 public String commentsLevel = null;
 // comments count
 public Integer CommentsCount = 1;
 // process name
 public String processName = null;
 // process comments label
 public String processCommentLabel = null;
 // process comments
 public String processComment = null;
 // step comments list
 public List<StepComment> stepCommentList = new
 List<StepComment>();
 /**
 * Public constructor
 */
 public SubmissionComments() {
}
/**
 * Gets process name
 */
global String getProcessName() {
 return processName;
}
/**
 * Gets process comment label
*/
global String getProcessCommentLabel() {
 return processCommentLabel;
}
/**
 * Gets process comment
 */
global String getProcessComment() {
 return processComment;
}
/**
 * Sets process comment
 * @param comment
 */
global void setProcessComment(String comment) {
 this.processComment = comment;
}
```

```
/**
 * Is comment at process level?
 */
global Boolean isProcessLevelComment() {
 return (PROCESS_LEVEL_COMMENTS == commentsLevel);
}
/**
* Is comment at step level?
*/
global Boolean isStepLevelComment() {
 return (STEP_LEVEL_COMMENTS == commentsLevel);
}
/**
* Get comments count - max of 3 at step-level
and 1 at process-level
*/
global Integer getCommentsCount() {
 return commentsCount;
}
/**
 * Gets step comment list
 */
global List<StepComment> getStepCommentList() {
 return stepCommentList;
}
/**
 * Add stepComment to list
 * @param comment
 */
global void addStepComment(StepComment comment) {
 stepCommentList.add(comment);
}
/**
 * Inner class to hold step level comments
*/
global class StepComment {
 // step name
 public String stepName = null;
 // step comment label
 public String stepCommentLabel = null;
 // step comments
 public String stepComment = null;
 /**
  * Public constructor
  */
 public StepComment() {
 }
 /**
  * Gets step name
  */
 global String getStepName() {
```

```
 return stepName;
 }
/**
 * Gets step comment label
 */
 global String getStepCommentLabel() {
  return stepCommentLabel;
 }
 /**
 * Gets step comment
  */
 global String getStepComment() {
 return stepComment;
 }
/**
  * Set step comment
  * @param comment
  */
 global void setStepComment(String comment) {
  this.stepComment = comment;
 }
}
}
```

Response Parameter		
Name	**Type**	**Description**
result	Boolean	Returns true if the API is executed successfully.

Taking Ownership of an Approval Request

This API takes ownership of the given approval request. This API accepts process definition ID, list of request IDs, user ID and ID of the context object as request parameters.

takeOwnership

Request Parameters			
Name	**Type**	**Required?**	**Description**
processDefnId	ID	Yes	ID of the process definition.
requestIdList	Set<ID>	Yes	Set of request identifiers.
userId	ID	Yes	ID of the requesting owner.

Request Parameters			
Name	**Type**	**Required?**	**Description**
ctxObjectId	ID	Yes	ID of the business context object.

Response Parameter		
Name	**Type**	**Description**
result	Boolean	The API returns a true value upon successful execution of the API.

Apttus Copyright Disclaimer

For additional resources and support, please visit https://community.apttus.com.

DOC ID: IWASFSUM19API20190801