



Visual Workflow API Guide

Version: 2020.3.0

Copyright AppViewX, Inc.

Copyright © 2020 AppViewX, Inc. All Rights Reserved.

This document may not be copied, disclosed, transferred, or modified without the prior written consent of AppViewX, Inc. While all content is believed to be correct at the time of publication, it is provided as general-purpose information. The content is subject to change without notice and is provided “as is” and with no expressed or implied warranties whatsoever, including, but not limited to, a warranty for accuracy made by AppViewX. The software described in this document is provided under written license only, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. Unauthorized use of software or its documentation can result in civil damages and criminal prosecution.

Trademarks

The trademarks, logos, and service marks displayed in this manual are the property of AppViewX or other third parties. Users are not permitted to use these marks without the prior written consent of AppViewX or such third party which may own the mark.

External Reference Links

This product includes software developed by the CentOS Project (www.centos.org).

This product includes software developed by Red Hat, Inc. (www.redhat.com).

This product includes software developed by VMware, Inc. (www.vmware.com).

All other trademarks mentioned in this document are the property of their respective owners.

Contact Information

AppViewX, Inc.

222 Broadway, FL 19

New York, NY 10038

Email: info@appviewx.com

Web: www.appviewx.com

Contents

Preface.....	vii
Revision History.....	vii
About this Guide.....	vii
Audience.....	vii
Text Conventions.....	vii
Chapter 1. Login to Retrieve Active SessionID.....	8
Overview.....	8
Request Structure.....	8
Response Structure.....	9
Response.....	9
Status Codes.....	9
Sample Request/Response.....	10
Chapter 2. Submit Request.....	11
Overview.....	11
Before you begin.....	11
Request Structure.....	13
Task Specific Information.....	14
Service Task Payload.....	14
Header Payload.....	14
User Task Payload.....	15
Data Payload.....	16
Input Payload.....	16
Header Payload.....	16
Request Data Payload.....	16
Response Structure.....	22
Response.....	22
Status Codes.....	23

Chapter 3. Update Request.....	24
Overview.....	24
Before you begin.....	24
Request Structure.....	26
Task Specific Information.....	27
Review Task Payload.....	27
Header Payload.....	28
Data Payload.....	28
Input Payload.....	28
Grid Task Payload.....	33
Header Payload.....	34
Data Payload.....	34
Input Payload.....	34
Sample Request/Response - Grid Task.....	34
Reponse Structure.....	36
Response.....	36
Status Codes.....	36
Chapter 4. Get Current Task Detail of the Request.....	38
Overview.....	38
Before you begin.....	38
Request Structure.....	40
Payload.....	41
Response Structure.....	41
Status Codes.....	42
Sample Request/Response.....	42
Chapter 5. Get Request Summary/Log.....	46
Overview.....	46
Before you begin	46
Request Structure.....	48

Response Structure.....	49
Status Codes.....	49
Sample Request/Response.....	49
Chapter 6. Get Associated Workorder.....	52
Overview.....	52
Before you begin.....	52
Request Structure.....	54
Response Structure.....	55
Status Codes.....	55
Sample Request/Response.....	55
Chapter 7. Get Tasks which are in InProgress State.....	57
Overview.....	57
Before you begin	57
Request Structure.....	59
Payload.....	60
Response Structure.....	60
Status Codes.....	61
Sample Request/Response.....	61
Chapter 8. Upload File.....	63
Overview.....	63
Before you begin.....	63
Request Structure.....	65
Payload.....	66
Response Structure.....	66
Status Codes.....	67
Sample Request/Response.....	67
Chapter 9. Download File.....	69
Overview.....	69
Before you begin.....	69

Request Structure.....	71
Payload.....	72
Response Structure.....	72
Status Codes.....	72
Sample Request/Response.....	73
Chapter 10. Delete File.....	74
Overview.....	74
Before you begin.....	74
Request Structure.....	76
Payload.....	77
Response Structure.....	77
Status Codes.....	77
Sample Request/Response.....	77

Preface

Revision History

Revision	Description	Date
1.0	Initial release of document for Release 2020.3.0	September 2020

About this Guide

AppViewX's Visual Workflow is a hybrid cloud automation and orchestration platform that allows for simple and complex network automation within your infrastructure. Visual workflow supports a wide range of device types, vendors, actions and integrations, in order to help you manage and automate your entire network with a unified automation and orchestration platform.

Visual workflow's primary interface is through both the Web GUI interface and a HTTP Restful API. A developer would use this API for programmatic access

Audience

This guide is useful for experienced users with reasonable programming experience (Python), trying to enhance their skills with automating network tasks. You can use AppViewX pre-built workflow tasks, commands, API, SDK to automate across the L2-L7 layer, such as automate load balancers, DNS, WAF, PKI, switches, routers, and other network elements.

Text Conventions

The following text conventions are used in this document:

Convention	Description
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in the text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>codeblock</code>	Indicates commands with a paragraph, URLs, codes in examples, text that appears on the screen, or text that you enter.

Chapter 1: Login to Retrieve Active SessionID

- [Overview](#)
- [Request Structure](#)
- [Response Structure](#)
- [Response](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

This API retrieves the sessionId. This sessionId retrieved can be used for another api that ensures a valid session.

Request Structure

URL: /acctmgmt-perform-login

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	username	AppViewX login username	String	NA
Header	password	AppViewX login password	String	NA
Header	Content-Type	Specifies the nature of the data in the payload	String	The value of the param should be 'application/json'
Query	gwkey	Tenant Key. Needed only in the case of multi-tenant installations.	String	NA
Query	gwsouce	Source from which the request is triggered (Eg: web, external)	String	NA
Body	payload	Contains an empty payload	String	NA

Response Structure

200 OK returns string of type application/json with the following body params:

Name	Description	Field Type
response	Contains the response params for the search object request	response
message	Success message or failure description in case of error	String
appStatusCode	Application specific status code for the response. Will be non-null for failure response.	String
tags	More info in case of failure response	NA

Response

Name	Description	Field Type
status	Status of the request	String
appStatusCode	appStatusCode	String
statusDescription	Description of the status	String
sessionId	SessionId to be retrieved	String
availableLoginAttemptCount	Count of login attempts	String

Status Codes

HTTP Status Code	appStatusCode	Message	Possible remediation
200 OK	NA	Login successful	
400 Bad Request	NA	Username or password cannot be null or empty	Check and ensure if a non-null/non-empty value is given in the header field for username or password

HTTP Status Code	appStatusCode	Message	Possible remediation
401 Unauthorized	NA	Login failed. Invalid credentials.	Check and ensure if the username and password provided are valid and correct

Sample Request/Response

This api retrieves the sessionId. This sessionId retrieved can be used for another api that ensures a valid session.

Use Case

Submit a request for the sessionId retrieval.

Sample Request

```
{
  "payload": {
  }
}
```

Sample Response

```
{
  "response": {
    "status": "SUCCESS",
    "appStatusCode": null,
    "statusDescription": null,
    "sessionId": "dea911-f342-440b-8bce-fe7fbc204g6c",
    "availableLoginAttemptCount": "10"
  },
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}
```

Chapter 2: Submit Request

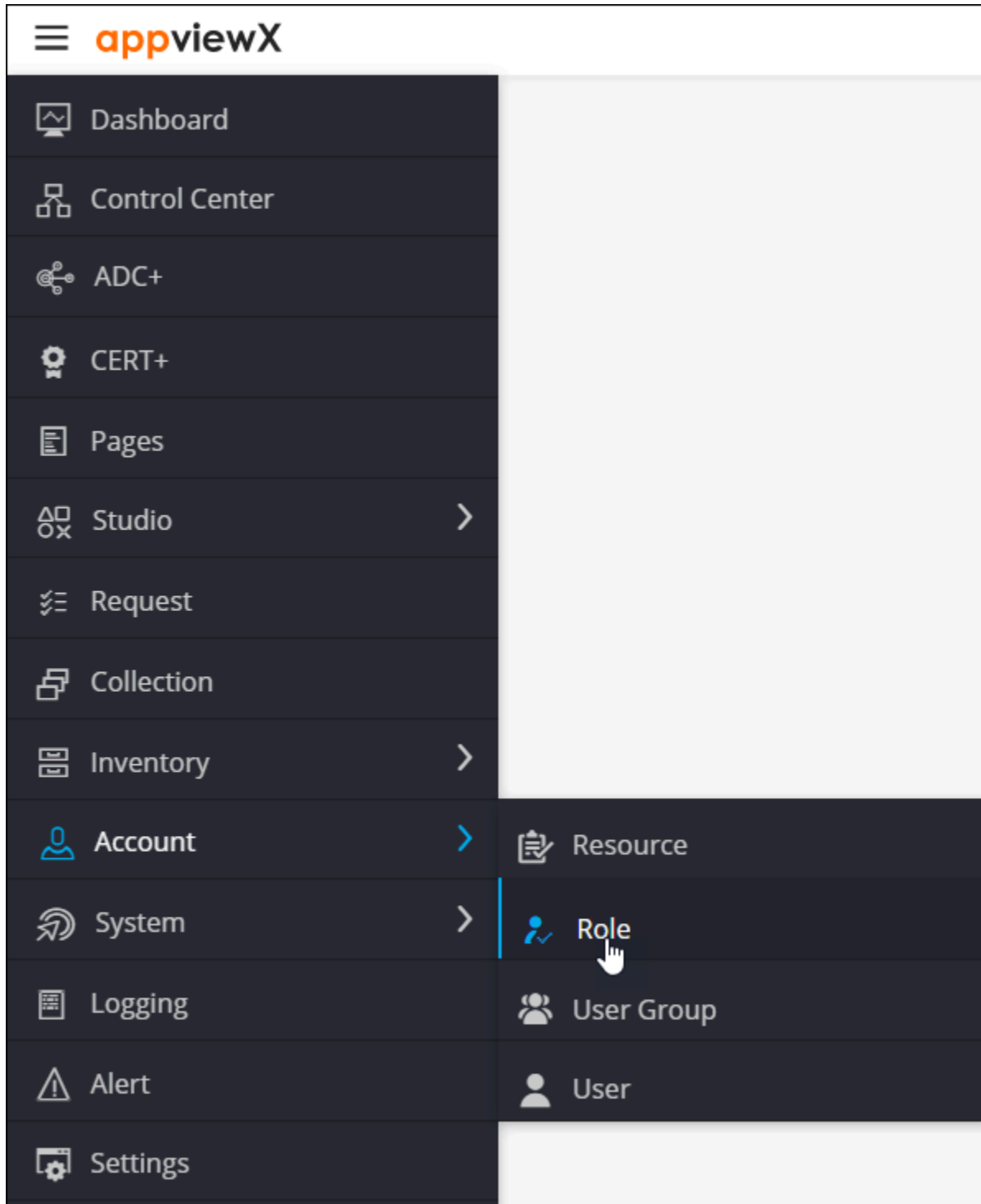
- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Task Specific Information](#)
- [Response Structure](#)
- [Response](#)
- [Status Codes](#)

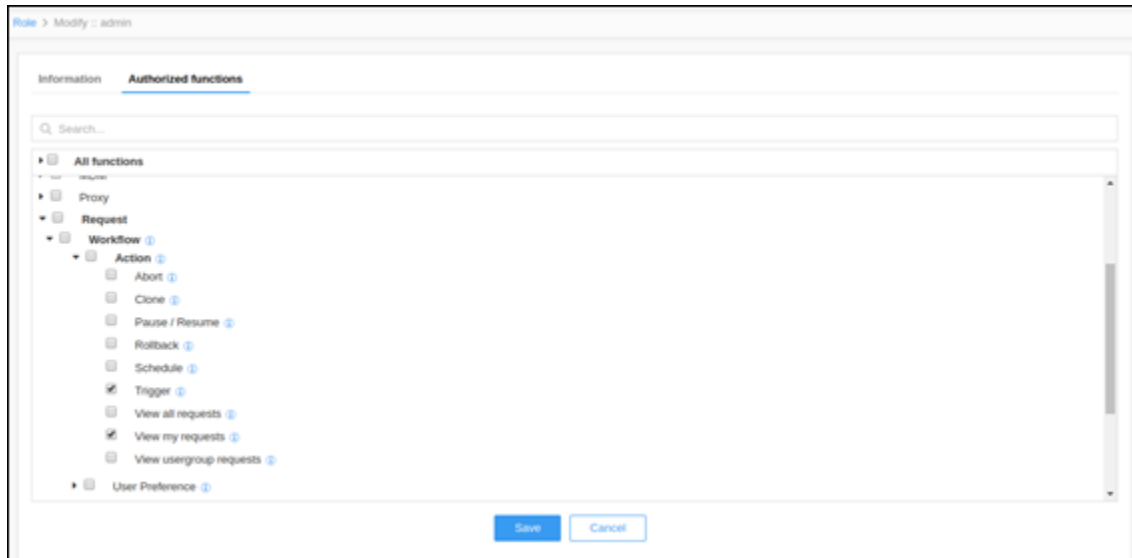
Overview

Creates a new workflow request for the given workflow name. A request id is generated and associated with each submitted workflow request which can be used later for retrieving any information about the request.

Before you begin

Before attempting to submit a request with this API, users must ensure the ACF permission **Request > Workflow > Action > Trigger** is available. This can be checked under **Accounts > Role > Authorized Functions**.





Request Structure

URL: /visualworkflow-submit-request

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	sessionId	SessionId received after login.	String	Required if username and password are not provided.
Header	username	AppViewX login username	String	Required if sessionId is not provided.
Header	password	AppViewX login password	String	Required if sessionId is not provided.
Header	Content-Type	Specifies the nature of the data in the payload.	String	Value of the param should be 'application/json'.
Query	gwkey	Tenant Key. Needed only in the case of multi-tenant installations.	String	NA
Query	gwsouce	Source from which the request is triggered (Eg: web, external)	String	NA

Param Type	Name	Description	Field Type	Constraints
Body	payload	Contains all the params to be sent in the request body for the post request.	Any one of the following: Service Task Payload Payload User Task Payload Payload	NA

Task Specific Information

- [Service Task Payload](#)
- [Header Payload](#)
- [User Task Payload](#)
- [Data Payload](#)
- [Input Payload](#)
- [Header Payload](#)
- [Request Data Payload](#)

Service Task Payload

Service tasks will not have any user input for submitting the request.

Name	Mandatory	Description	Field Type	Constraints
header	Yes	Contains header param details of the request	Header Payload	NA

Header Payload

Name	Mandatory	Description	Field Type	Constraints
workflowName	Yes	The Workflow for which the request is submitted	String	NA

- [Sample Request/Response](#)

Sample Request/Response

Use Case

Submit a request for the workflow servicenow test.

Sample Request

```
{
  "payload": {
    "header": {
      "workflowName": "servicenow Test"
    }
  }
}
```

Sample Response

```
{
  "response": {
    "workorderId": "0",
    "message": "Workflow Request is created with Id 401 . Request submitted to workflow engine for processing workorder.",
    "requestType": "default",
    "requestId": "401",
    "workflowVersion": "master",
    "status": "In Progress",
    "statusCode": 0
  },
  "message": null,
  "appStatusCode": null,
  "tags": null
}
```

User Task Payload

Name	Mandatory	Description	Field Type	Constraints
header	Yes	Contains header param details of the request	Header Payload	NA

Name	Mandatory	Description	Field Type	Constraints
data	Yes	Contains param for the user task	Data Payload	NA

Data Payload

Name	Mandatory	Description	Field Type	Constraints
input	Yes	The input data for the user task	Input Payload	NA
task_action	Yes	Action to be performed on the task	Integer	1 for success case and 2 for reject case

Input Payload

Name	Mandatory	Description	Field Type	Constraints
requestData	Yes	The request data for which the request is submitted	List of Request Data Payload	NA

Header Payload

Name	Mandatory	Description	Field Type	Constraints
workflowName	Yes	The Workflow for which the request is submitted	String	NA

Request Data Payload

Name	Mandatory	Description	Field Type	Constraints
scenario	Yes	The scenario name for the form	String	Na
sequenceNo	Yes	The sequence number of the scenario	Integer	NA
fieldInfo	Yes	The field info configured in the form. The value should be key value, field id and its value	Json	NA

- [Sample Request/Response - Form Task](#)
- [Sample Request/Response - Grid Task](#)
- [Sample Request/Response - Script Task](#)
- [Sample Request/Response - YAML Task](#)

Sample Request/Response - Form Task

Submit a request for the workflow with the first task as Form.

Sample Request

```
{
  "payload": {
    "header": {
      "workflowName": "Sample 2"
    },
    "data": {
      "input": {
        "requestData": [
          {
            "scenario": "scenario",
            "sequenceNo": 1,
            "fieldInfo": {
              "deviceName": "ADC"
            }
          },
          {
            "scenario": "scenario2",
            "sequenceNo": 2,
            "fieldInfo": {
              "deviceName": "A10"
            }
          }
        ]
      },
      "task_action": 1
    }
  }
}
```

```
}

```

Sample Response

```
{
  "response": {
    "workorderId": "0",
    "message": "Workflow Request is created with Id 55 . Request submitted to workflow engine for processing workorder.",
    "requestType": "default",
    "requestId": "401",
    "workflowVersion": "master",
    "status": "In Progress",
    "statusCode": 0
  },
  "message": null,
  "appStatusCode": null,
  "tags": null
}
```

Sample Request/Response - Grid Task

Submit a request for the workflow with the first task as Grid.

Sample Request

```
{
  "payload": {
    "data": {
      "input": {
        "columnLabels": {
          "IP": "IP",
          "Device Name": "Device Name"
        },
        "gridData": [
          {
            "IP": "", "Device Name": ""
          }
        ]
      },
      "globalData": { "gridData": [

```

```

{
  "IP": "192.168.15.35",
  "Device Name": "Device F5"
},
{
  "IP": "192.168.15.40",
  "Device Name": "Device AVI"
}
]),
"task_action": 1
},
"header": {
  "workflowName": "Grid"
}
}
}

```



Note: Global Variable `<%gridData%>` should be present in the “values” section in Grid Task.

Sample Response

```

{
  "response": {
    "workorderId": "0",
    "message": "Workflow Request is created with Id 24 . Request submitted to workflow engine for processing workorder.",
    "requestType": "default",
    "requestId": "401",
    "workflowVersion": "master",
    "status": "In Progress",
    "statusCode": 0
  },
  "message": null,
  "appStatusCode": null,
  "tags": null
}

```

Sample Request/Response - Script Task

Submit a request for the workflow with the first task as Script.

Sample Request

```
{
  "payload": {
    "data": {
      "globalData":{"script" : "Device 25"},
      "task_action": 1
    },
    "header": {
      "workflowName": "script"
    }
  }
}
```



Note: Values can be passed to available Global variables present inside the Script task.

Sample Response

```
{
  "response": {
    "workorderId": "0",
    "message": "Workflow Request is created with Id 26. Request submitted to workflow engine for processing workorder.",
    "requestType": "default",
    "requestId": "401",
    "workflowVersion": "master",
    "status": "In Progress",
    "statusCode": 0
  },
  "message": null,
  "appStatusCode": null,
  "tags": null
}
```

Sample Request/Response - YAML Task

Submit a request for the workflow with the first task as YAML.

Sample Request

```
{
  "payload": {
    "data": {
      "input": {
        "editorContent": "<%yaml%>",
        "type": "YAML"
      },
      "globalData": {"yaml" : "---\n\n  name: Manage BIG-IP APM policy or APM access profile imports\n  hosts: localhost\n  tasks:\n    - name: Import APM\n    profile\n      bigip_apm_policy_import:\n      name: new_apm_profile"},
      "task_action": 1
    },
    "header": {
      "workflowName": "yaml"
    }
  }
}
```



Note: Global Variable <%yaml%> should be present in the YAML task.

Sample Response

```
{
  "response": {
    "workorderId": "0",
    "message": "Workflow Request is created with Id 26 . Request submitted to workflow engine for processing workorder.",
    "requestType": "default",
    "requestId": "401",
    "workflowVersion": "master",
    "status": "In Progress",
    "statusCode": 0
  },
  "message": null,
  "appStatusCode": null,
}
```

```
"tags": null
}
```

Response Structure

200 OK, returns string of type application/json with the following body params

Name	Description	Field Type
response	Contains the response params for the search object request	response
message	Success message or failure description in case of error	String
appStatusCode	Application specific status code for the response, will be non-null for failure response	String
tags	More info in case of failure response	NA

Response

Name	Description	Field Type
requestID	Unique identifier for the request	String
workflowVersion	Version of the workflow	String
message	The message with the status and request number for the request	String
status	Status of the request	String
statusCode	Status code of the request	Integer
requestType	The type of the request	String
workorderId	The workorderid for the request If there is no workorder the value will be 0	String

Status Codes

HTTP Status Code	appStatusCode	Message	Possible remediation
200 OK	-	Success	-
400 Bad request	avx-common-028	Invalid/Incorrect payload	Check and ensure if a valid value is given in request payload field - input.
401 Unauthorized	WORKFLOW_1679	User is not authorized.	LoggedIn user is not authorized to access the workflow.
404 Not Found	engine-db-015	Workflow not found	Invalid workflow data
409 Conflict	WRKFLOW_1475	Given workflow is not in enabled state.	Given workflow is not in enabled state .
500 Internal Server Error	WORKFLOW_1617	Invalid form data. Please provide all mandatory data.	Check and ensure if a valid value is given in request payload field - input.
500 Internal Server Error		Error while processing	<ol style="list-style-type: none"> 1.Task already completed 2.Task already being processed by another user

Chapter 3: Update Request

- Overview
- Before you begin
- Request Structure
- Task Specific Information
- Grid Task Payload
- Header Payload
- Data Payload
- Input Payload
- Reponse Structure
- Response
- Status Codes

Overview

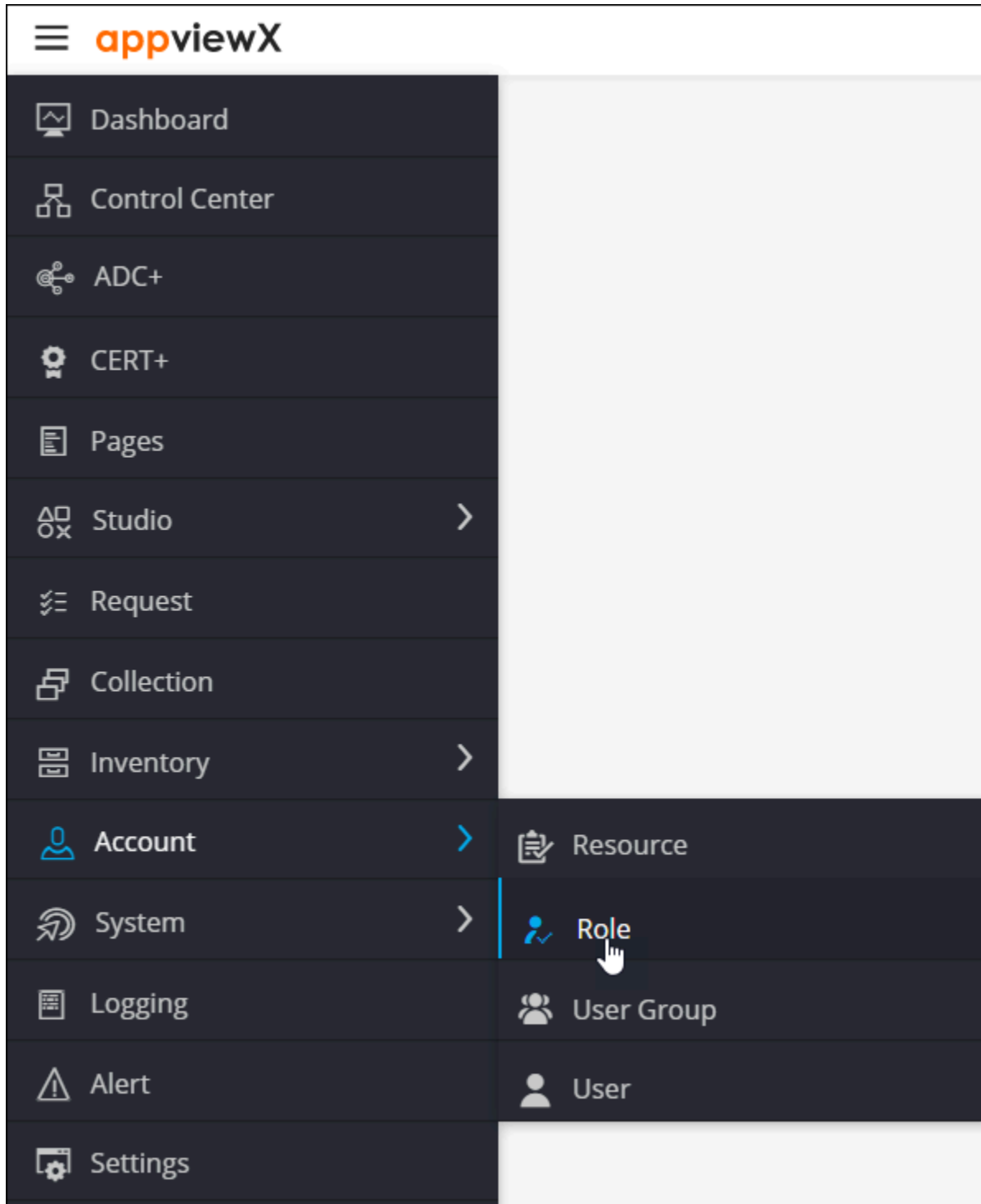
Updates an existing request waiting for a user input to progress and proceed to the next task of the workflow. This is applicable only for tasks that need manual intervention. For example, Form task.

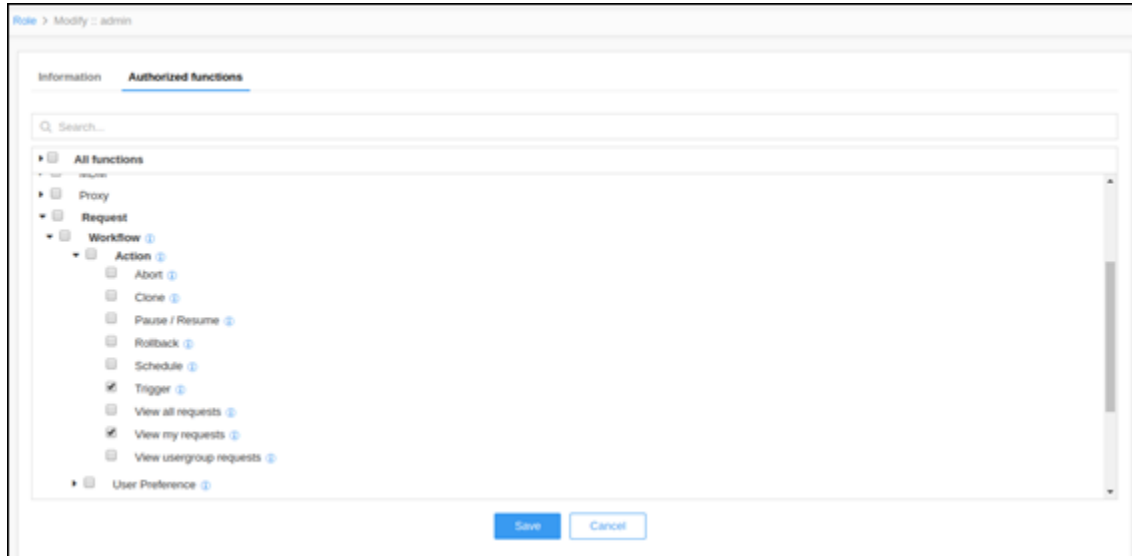
Before you begin

Before attempting to submit a request with this API, you must ensure that the following ACF permission is available:

Request > Workflow > Action > Trigger

This can be checked under **Accounts > Role > Authorized Functions**.





Request Structure

URL: /visualworkflow-update-request

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	sessionId	SessionId received after login.	String	Required if username and password are not provided.
Header	username	AppViewX login username	String	Required if sessionId is not provided.
Header	password	AppViewX login password	String	Required if sessionId is not provided.
Header	Content-Type	Specifies the nature of the data in the payload.	String	Value of the param should be 'application/json'
Query	gwkey	Tenant Key. Needed only in case of multi-tenant installations.	String	

Param Type	Name	Description	Field Type	Constraints
Query	gwsource	Source from which the request is triggered (Eg: web, external)	String	
Body	payload	Contains all the params to be sent in the request body for the post request.	Any one of the following: Review Task Payload User Task Payload	

Task Specific Information

- [Review Task Payload](#)
- [Header Payload](#)
- [Data Payload](#)
- [Input Payload](#)

Review Task Payload

Service tasks will not have any user input for submitting the request.

Name	Mandatory	Description	Field Type	Constraints
header	Yes	Contains header param details of the request.	Header Payload	NA
data	Yes	Contains param for the user task.	Data Payload	NA

Header Payload

Name	Mandatory	Description	Field Type	Constraints
request_id	Yes	The request id to update the request details.	String	NA
workorder_id	Yes	The workorder id to update the request details.	String	If there is no workorder in the workflow then the default value will be zero(0)
task_id	Yes	The task id which will be updated.	String	NA

Data Payload

Name	Mandatory	Description	Field Type	Constraints
input	Yes	The input data for the user task.	Input Payload	NA
task_action	Yes	Action to be performed on the task.	Integer	1 for success case and 2 for reject case

Input Payload

Name	Mandatory	Description	Field Type	Constraints
config	Yes	The prevalidation, implementation, post validation and rollback configuration for the devices.	Object	

- [Sample Request/Response - Review Task](#)
- [Sample Request/Response - Review Task within a Subflow](#)

Sample Request/Response - Review Task

Update a request with a review palette.

Sample Request

```
{
  "payload": {
    "header": {
      "request_id": "936",
      "workorder_id": "0",
      "task_id": "reviewComponent_1"
    },
    "data": {
      "input": { "config": {
        "implement": {
          "deviceList": [
            {
              "deviceName": "192.168.41.101",
              "commands": [
                {
                  "command": "<push></push>",
                  "deviceName": "",
                  "uniqueNumber": 2,
                  "sequenceNumber": 0,
                  "properties": {
                    "sleep": 0
                  },
                  "type": "push"
                }
              ]
            }
          ]
        }
      }
    },
    "prevalidation": {
      "deviceList": [
        {
          "deviceName": "192.168.41.101",
          "commands": [
            {
              "command": "<push></push>",
              "deviceName": "",
```

```
"uniqueNumber": 2,
"sequenceNumber": 0,
"properties": {
  "sleep": 0
},
"type": "push"
}
]
}
]
},
"postvalidation": {
  "deviceList": [
    {
      "deviceName": "192.168.41.101",
      "commands": [
        {
          "command": "<push></push>",
          "deviceName": "",
          "uniqueNumber": 2,
          "sequenceNumber": 0,
          "properties": {
            "sleep": 0
          },
          "type": "push"
        }
      ]
    }
  ]
}
}
},
"task_action":1
}
}
}
```

Sample Response

```
{
  "response": {
    "message": "Request submitted to workflow engine for processing workflow request 936",
    "requestId": "936"
  },
  "message": null,
  "appStatusCode": null,
  "tags": null
}
```

Sample Request/Response - Review Task within a Subflow

Update a request with a review palette, where the review task is within a subflow.

Sample Request

```
{
  "payload": {
    "header": {
      "request_id": "936",
      "workorder_id": "0",
      "task_id": "reviewComponent_1#subflowid"
    },
    "data": {
      "input": { "config": {
        "implement": {
          "deviceList": [
            {
              "deviceName": "192.168.41.101",
              "commands": [
                {
                  "command": "<push></push>",
                  "deviceName": "",
                  "uniqueNumber": 2,
                  "sequenceNumber": 0,
                  "properties": {
                    "sleep": 0
                  }
                }
              ]
            }
          ]
        }
      }
    }
  }
}
```

```
    },
    "type": "push"
  }
]
}
]
},
"prevalidation": {
  "deviceList": [
    {
      "deviceName": "192.168.41.101",
      "commands": [
        {
          "command": "<push></push>",
          "deviceName": "",
          "uniqueNumber": 2,
          "sequenceNumber": 0,
          "properties": {
            "sleep": 0
          },
          "type": "push"
        }
      ]
    }
  ]
},
"postvalidation": {
  "deviceList": [
    {
      "deviceName": "192.168.41.101",
      "commands": [
        {
          "command": "<push></push>",
          "deviceName": "",
          "uniqueNumber": 2,
          "sequenceNumber": 0,
          "properties": {
```

```

    "sleep": 0
  },
  "type": "push"
}
]
}
]
}
}
},
"task_action":1
}
}
}

```

Sample Response

```

{
  "response": {
    "message": "Request submitted to workflow engine for processing workflow request 936",
    "requestId": "936"
  },
  "message": null,
  "appStatusCode": null,
  "tags": null
}

```

Grid Task Payload

Service tasks will not have any user input for submitting the request.

Name	Mandatory	Description	Field Type	Constraints
header	Yes	Contains header param details of the request	Header Payload	
data	Yes	Contains param for the user task	Data Payload	

Header Payload

Name	Mandatory	Description	Field Type	Constraints
request_id	Yes	The request id to update the request details.	String	
workorder_id	Yes	The workorder id to update the request details.	String	If there is no workorder in the workflow then the default value will be zero(0)
task_id	Yes	The task id which will be updated.	String	NA

Data Payload

Name	Mandatory	Description	Field Type	Constraints
input	Yes	The input data for the user task	Input Payload	
task_action	Yes	Action to be performed on the task	Integer	1 for success case and 2 for reject case

Input Payload

Name	Mandatory	Description	Field Type	Constraints
gridData	Yes	The grid config to be updated to the task	Object	NA

- [Sample Request/Response - Grid Task](#)

Sample Request/Response - Grid Task

Update a request with a Grid task.

Sample Request

```
{
  "payload": {
```

```
"header": {
  "request_id": "937",
  "workorder_id": "0",
  "task_id": "gridComponent_1"
},
"data": {
  "input": {
    "gridData": [
      {
        "name": "Appviewx",
        "designation": "Network"
      },
      {
        "name": "ADC",
        "designation": "Device"
      }
    ]
  },
  "task_action": 1
}
```

Sample Response

```
{
  "response": {
    "message": "Request submitted to workflow engine for processing workflow request 936",
    "requestId": "936"
  },
  "message": null,
  "appStatusCode": null,
  "tags": null
}
```

Response Structure

Name	Description	Field Type
response	Contains the response params for the search object request	response
message	Success message or failure description in case of error	String
appStatusCode	Application specific status code for the response. Will be non-null for failure response	String
tags	More info in case of failure response	NA

Response

Name	Description	Field Type
requestId	Unique identifier for the request	String
message	The message with the status and request number for the request	String

Status Codes

HTTP Status Code	appStatusCode	Message	Possible remediation
200 OK	-	Success	-
400 Bad request	avx-common-028	Invalid/Incorrect payload	Check and ensure if a valid value is given in request payload field - input.
404 Not Found	engine-db-011	Request not found	Invalid workflow data
500 Internal Server Error	WORKFLOW_1056	Workflow request already closed.	Ensure workflow is not closed.
500 Internal Server Error		Internal Server Error	<ul style="list-style-type: none"> Task already completed Task already processing by another user Workflow Request is in paused state Workflow Request is in aborted state

HTTP Status Code	appStatusCode	Message	Possible remediation
			<ul style="list-style-type: none">• Workflow request already closed• Given task is not the right task to be executed with Id

Chapter 4: Get Current Task Detail of the Request

- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Payload](#)
- [Response Structure](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

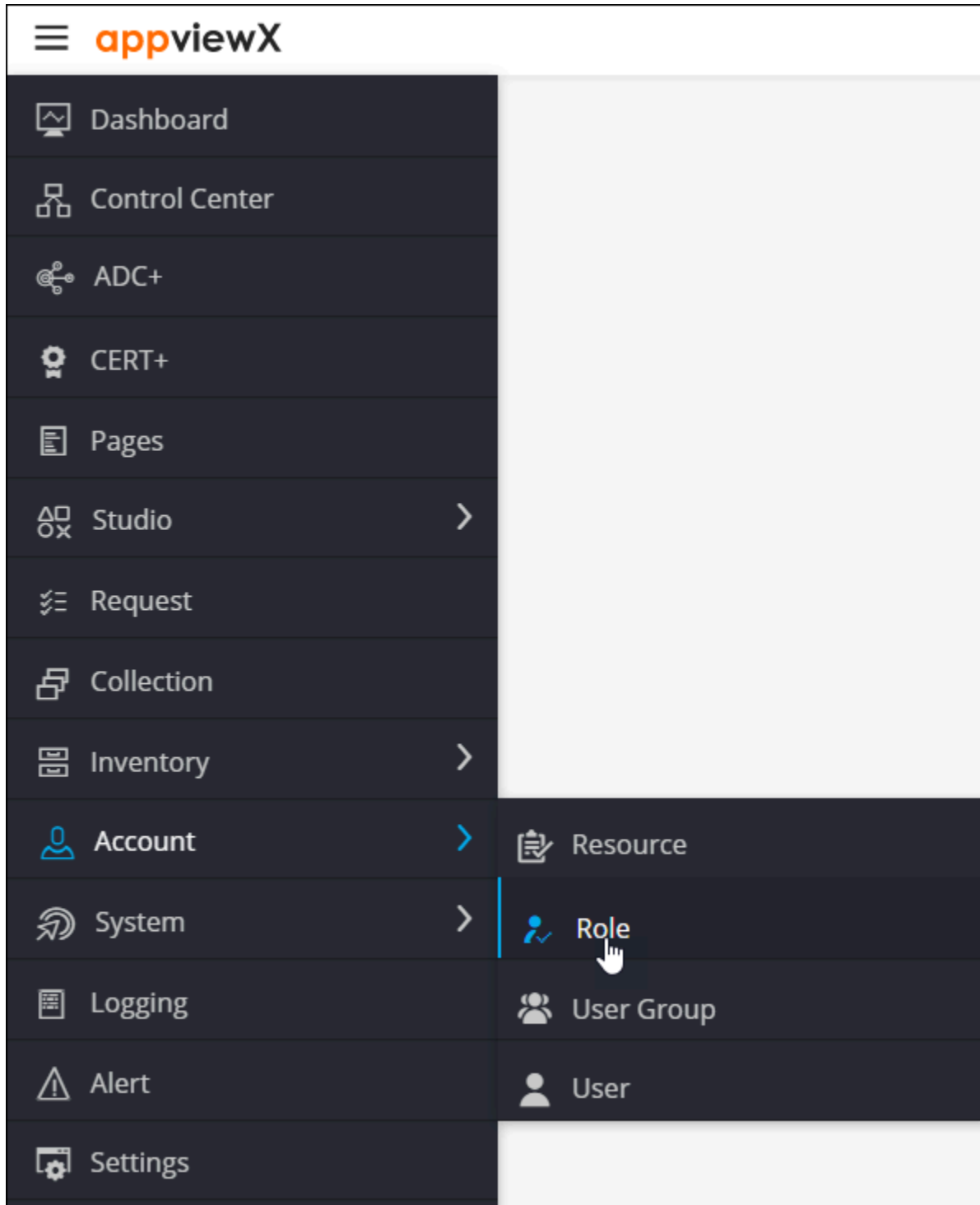
To get the details of the current active task of a specific workflow request. An action can be performed only on the current active task of a workflow.

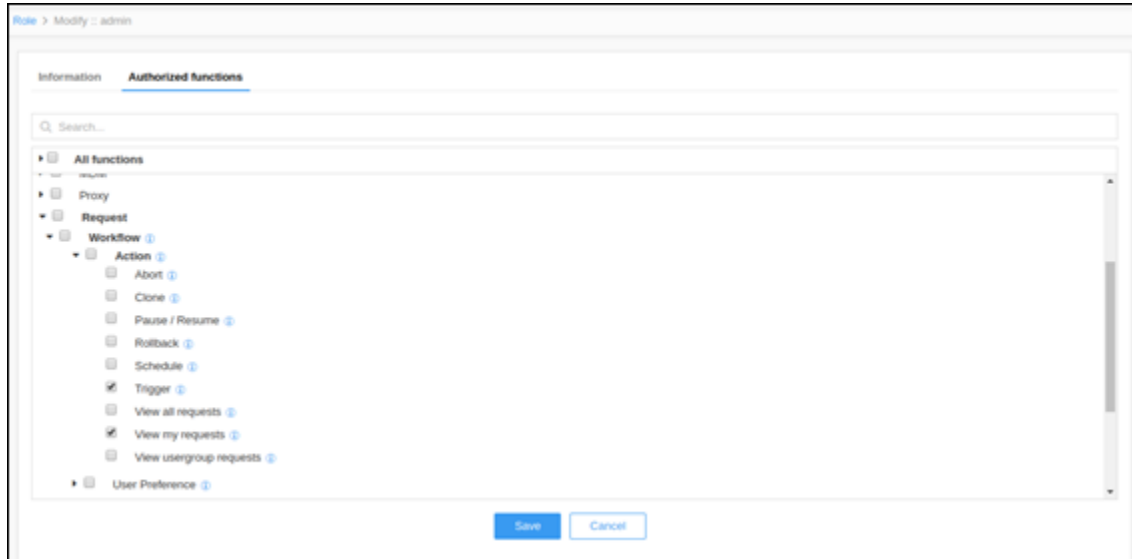
Before you begin

Before attempting to submit a request with this API, ensure that any one of the following ACF permissions are available:

- **Request > Workflow > Action > View All Request**
- **Request > Workflow > Action > View My Request**
- **Request > Workflow > Action > View Role Request**

This can be checked under **Accounts > Role > Authorized Functions**.





Request Structure

URL: /visualworkflow-get-request-current-task-details

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	sessionId	Session Id received after login	String	Required if username and password are not provided.
Header	username	AppViewX login username	String	Required if sessionId is not provided.
Header	password	AppViewX login password	String	Required if sessionId is not provided.
Header	Content-Type	Specifies the nature of the data in the payload.	String	Value of the param should be 'application/json'
Query	gwkey	Tenant Key. Needed only in case of multi tenant installations.	String	
Query	gwsouce	Source from which the request is triggered (Eg: web, external)	String	

Param Type	Name	Description	Field Type	Constraints
Body	payload	Contains all the params to be sent in the request body for the post request.	Any one of the following: Payload User Task Payload	

Payload

Name	Mandatory	Description	Field Type	Constraints
request_id	Yes	The request id to update the request details	String	NA
workorder_id	Yes	The workorder id to update the request details.	String	If there is no workorder in the workflow then the default value will be zero(0)
task_id	Yes	The task id which will be updated.	String	NA

Response Structure

200 OK returns string of type application/json with the following body params:

Name	Description	Field Type
response	Contains the response object along with the hierarchy of the object mapped in field children	Key value pair
message	Success message or failure description in case of error	String
appStatusCode	Application specific status code for the response Will be non-null for failure response	String

Name	Description	Field Type
tags	More info in case of failure response	NA

Status Codes

HTTP Status Code	appStatusCode	Message	Possible remediation
200 OK	-	Success	-
400 Bad request	avx-common-028	Invalid/ Incorrect payload	Check and ensure if a valid value is given in request payload field - input.
404 Not Found	engine-db-011	Request not found	Check and ensure if a valid requestId/workorderId is given.
500 Internal Server Error	engine-db-022	Unable to fetch task status	Check and ensure if a valid task id is given.

Sample Request/Response

Use Case

To get the current task details for a request.

Sample Request

```
{
  "payload": {
    "requestId": "409",
    "workorderId": "1",
    "taskId": "reviewComponent_1"
  }
}
```

Sample Response

```
{
  "response": {
    "jobId": null,
  }
}
```

```

"requestStatus": 0,
"createdTime": 0,
"request_id": "409",
"workorder_id": "1",
"workorder_status": "Completed",
"workflowName": "Sample",
"data": {
  "comments": "Hi Tester of RC",
  "state": 1,
  "taskStatus": "Success",
  "task_id": "reviewComponent_1",
  "task_name": "Review Component",
  "input": {
  },
  "output": {
    "comments": "Hi Tester of RC",
    "configLabel": {
      "Prevalidation": "Pre Validation",
      "implementation": "Implementation",
      "PostValidation": "Post Validation"
    },
    "enablePrevalidation": true,
    "fetchOnLoad": true,
    "proceedProcess": "Submit",
    "ticketDetails": {
      "ticketNumber": "",
      "startTime": null,
      "endTime": null
    },
    "configureChangeManagement": true,
    "isAuthorized": true,
    "vendor": "CHANGE",
    "breakdownProcess": "Cancel",
    "readWrite": true,
    "prevalidationValue": "Check",
    "config": {

```

```
"Prevalidation": {
  "deviceList": [
    {
      "deviceName": "f5",
      "commands": [
        "tmsh",
        "list gtm pool pool41_161_demo"
      ]
    }
  ],
  "implementation": {
    "deviceList": [
      {
        "deviceName": "f5",
        "commands": [
          "tmsh",
          "create gtm pool pool41_161_demo"
        ]
      }
    ]
  },
  "PostValidation": {
    "deviceList": [
      {
        "deviceName": "f5",
        "commands": [
          "tmsh",
          "show sys version"
        ]
      }
    ]
  },
  "logs": [
    "Request doesn't have valid ticket details."
  ],
}
```

```
"implementationType": "Manual"  
  
},  
  
"is_picked": false,  
  
"roles": null,  
  
"tags": null,  
  
"label": null,  
  
"component_type": "review",  
  
"task_process_state": 1,  
  
"is_completed": true,  
  
"task_category": "review"  
  
}  
  
},  
  
"message": null,  
  
"appStatusCode": null,  
  
"tags": null  
  
}
```

Chapter 5: Get Request Summary/Log

- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Response Structure](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

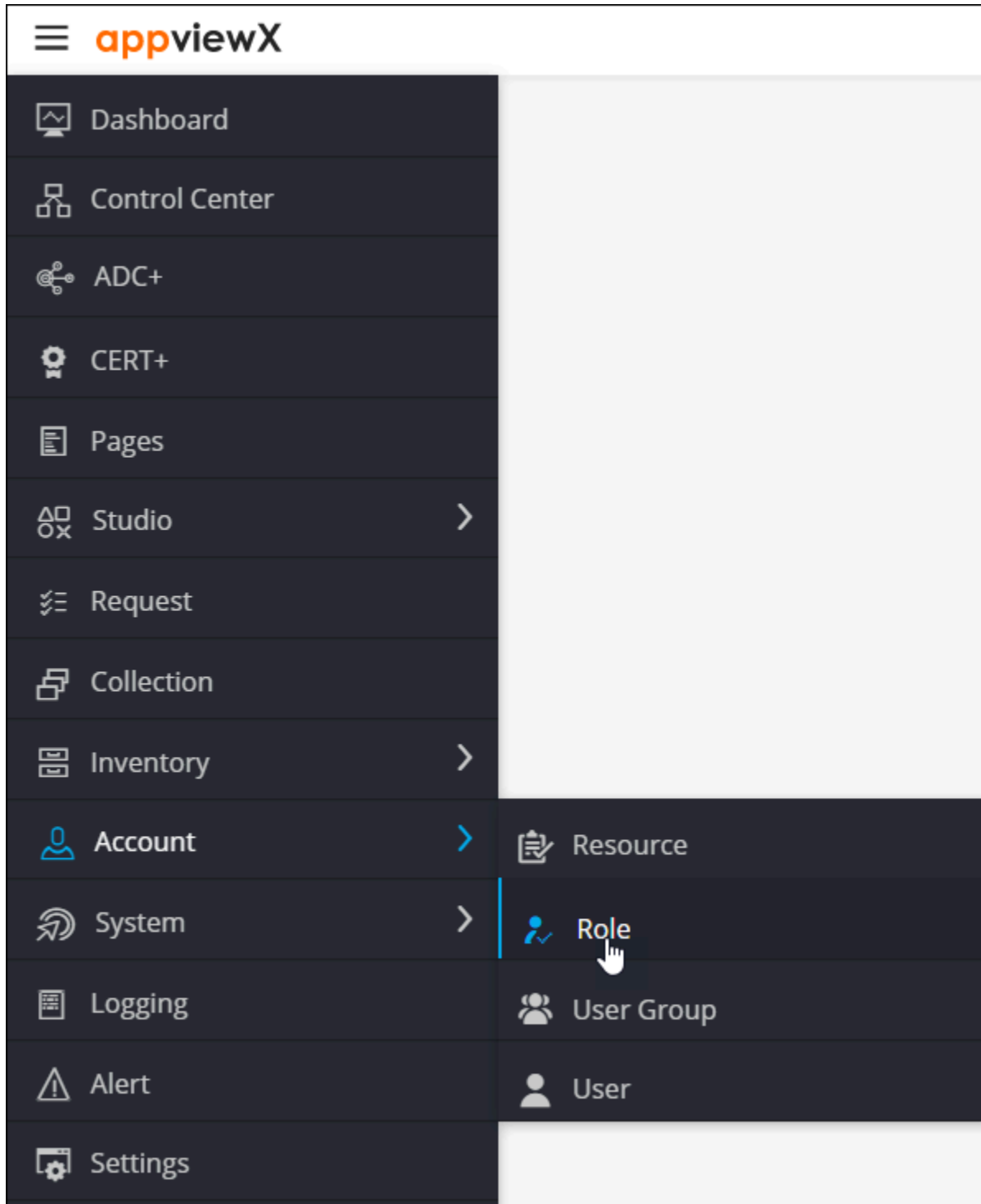
Get the request summary/logs details of a particular workflow request. The execution details of the workflow can be viewed from these logs.

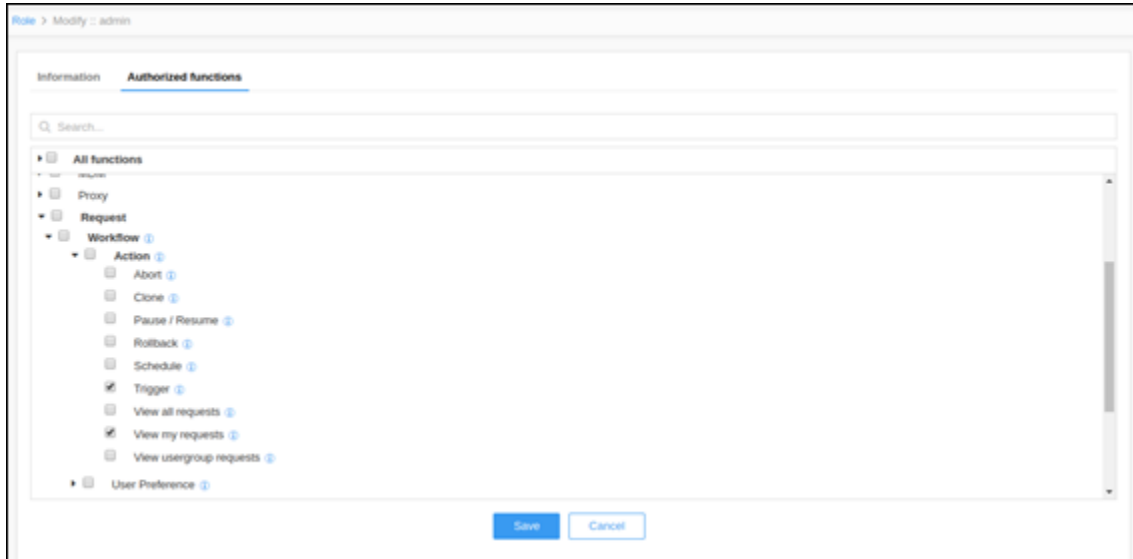
Before you begin

Before attempting to submit a request with this API, ensure that any one of the following ACF permissions are available:

- **Request > Workflow > Action > View All Request**
- **Request > Workflow > Action > View My Request**
- **Request > Workflow > Action > View Role Request**

This can be checked under **Accounts > Role > Authorized Functions**.





Request Structure

URL: /visualworkflow-request-logs

Type: GET

Param Type	Name	Description	Field Type	Constraints
Header	sessionId	Session Id received after login	String	Required if username and password are not provided
Header	username	AppViewX login username	String	Required if sessionId is not provided
Header	password	AppViewX login password	String	Required if sessionId is not provided
Header	Content-Type	Specifies the nature of the data in the payload	String	Value of the param should be 'application/json'
Query	gwkey	Tenant Key, needed only in case of multi-tenant installations.	String	NA
Query	gwsouce	Source from which the request is triggered (Eg: web, external)	String	NA

Param Type	Name	Description	Field Type	Constraints
Query	ids	Request ids	String	NA

Response Structure

200 OK returns string of type application/json with the following body params:

Name	Description	Field Type
response	Contains the response object along with the hierarchy of the object mapped in field children	Key value pair
message	Success message or failure description in case of error	String
appStatusCode	Application specific status code for the response Will be non-null for failure response	String
tags	More info in case of failure response	NA

Status Codes

HTTP Status Code	appStatusCode	Message	Possible remediation
200 OK	-	Success	-
400 Bad request	WORKFLOW_1300	Workflow Request Id is Mandatory.	Check and ensure if a valid request id is given.
404 Not Found	engine-db-011	Request not found	Invalid workflow data

Sample Request/Response

Use Case

To get the request summary/logs for a request.

Sample Request

NA

Sample Response

```

{
  "response": {
    "workOrderList": [
      {
        "statusCode": 1,
        "requestId": "410",
        "created_by": "admin",
        "status": "Completed",
        "workflowName": "Copy of reviewComponent Test",
        "toolTip": null,
        "workorder_id": null,
        "description": null,
        "ticket_details": null,
        "tasks": [
          {
            "state": 1,
            "comments": null,
            "task_name": "WorkOrder",
            "task_id": "startWorkOrder_1",
            "user": "admin",
            "task_status": "Success",
            "task_start_time": 1502696313667,
            "task_end_time": 1502696729781,

            "logs": [
              {
                "time": 1502696313815,
                "message": "Child WorkOrder ID: 1, has been
created for the Child workflow: WorkOrder"
              },
              {
                "time": 1502696729781,
                "message": "WorkOrder Completed"
              }
            ]
          }
        ]
      }
    ]
  }
}

```

```
    ]
  }
],
"created_time": 1502696305239,
"rollback_reference_id": "",
"clone_reference_ids": [],
"last_updated": 1502696305239,
"running_time": 0
}
]
},
"message": null,
"appStatusCode": null,
"tags": null
}
```

Chapter 6: Get Associated Workorder

- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Response Structure](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

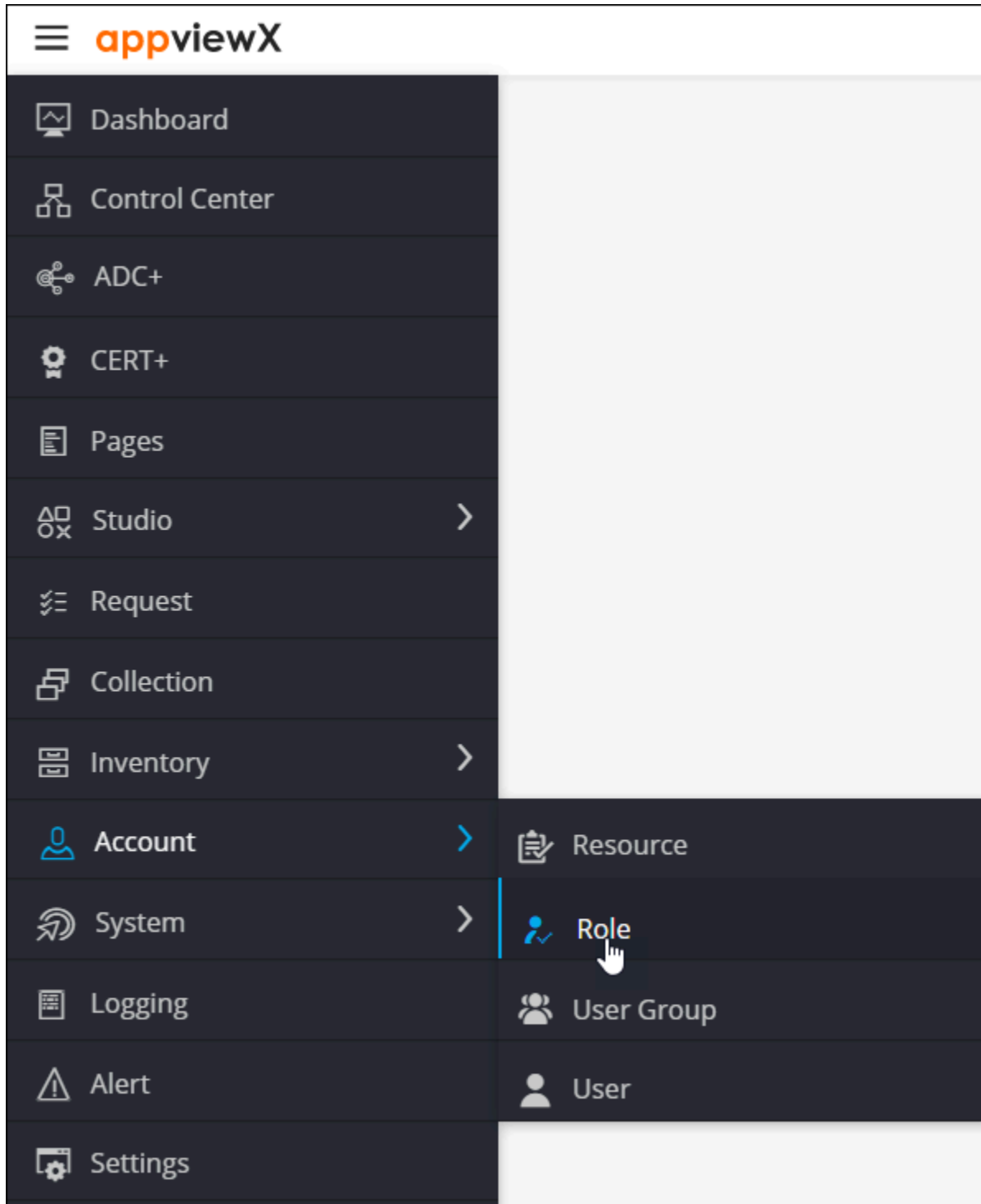
Retrieve the workorders associated with a specific workflow request. Workorders are used when there are multiple sub workflows involved with a workflow. Each workorder takes a separate parallel execution path.

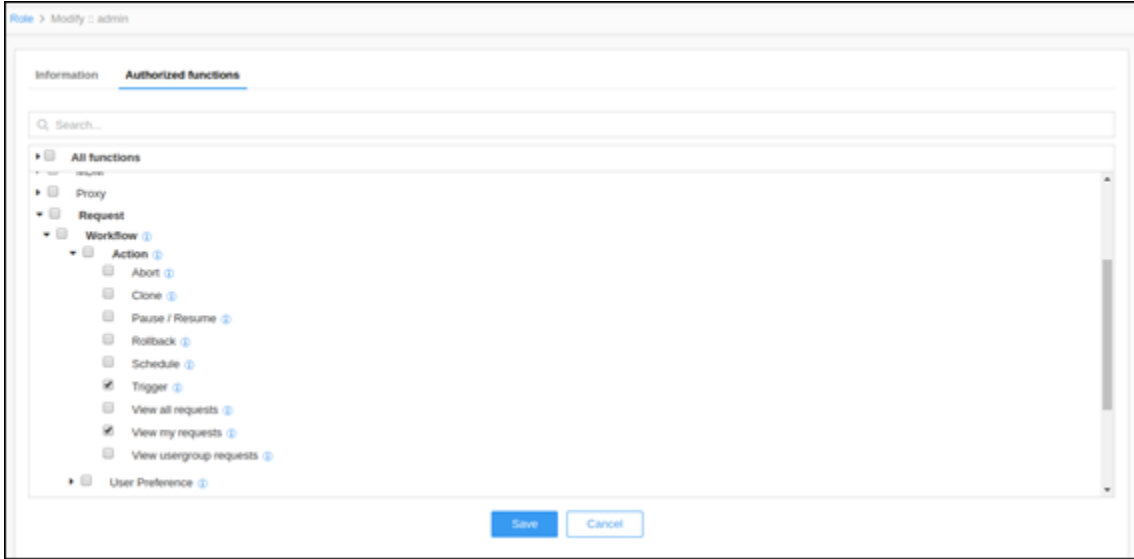
Before you begin

Before attempting to submit a request with this API, ensure that any one of the following ACF permissions are available:

- **Request > Workflow > Action > View All Request**
- **Request > Workflow > Action > View My Request**
- **Request > Workflow > Action > View Role Request**

This can be checked under **Accounts > Role > Authorized Functions**.





Request Structure

URL: /visualworkflow-get-associated-workorder

Type: GET

Param Type	Name	Description	Field Type	Constraints
Header	sessionId	SessionId received after login.	String	Required if username and password are not provided.
Header	username	AppViewX login username	String	Required if sessionId is not provided.
Header	password	AppViewX login password	String	Required if sessionId is not provided.
Header	Content-Type	Specifies the nature of the data in the payload.	String	Value of the param should be 'application/json'
Query	gwkey	Tenant Key, needed only in case of multi-tenant installations.	String	NA
Query	gwsouce	Source from which the request is triggered (Eg: web, external)	String	NA

Param Type	Name	Description	Field Type	Constraints
Query	requestId	Request id	String	NA

Response Structure

200 OK returns string of type application/json with the following body params:

Name	Description	Field Type
response	Contains the response object along with the hierarchy of the object mapped in field children	Key value pair
message	Success message or failure description in case of error	String
appStatusCode	Application specific status code for the response Will be non-null for failure response	String
tags	More info in case of failure response	NA

Status Codes

HTTP Status Code	appStatusCode	Message	Possible remediation
200 OK	-	Success	-
400 Bad request		Workflow Request Id is Mandatory	Check and ensure if a valid request id is given
404 Not Found		Request not found	Check and ensure if a valid request id is given

Sample Request/Response

Sample Request

NA

Sample Request

```
{
  "response": {
    "workorderList": [
      {
        "workorderId": "1",
        "createdTime": 1528957336385,
        "lastUpdatedTime": 1528957336385,
        "description": "startWorkOrder_1",
        "toolTip": {

        },
        "log": [
          {
            "time": 1528957341359,
            "message": "Initiating Script"
          }
        ]
      }
    ],
    "iTotalDisplayRecords": 1
  }
}
```

Chapter 7: Get Tasks which are in InProgress State

- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Payload](#)
- [Response Structure](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

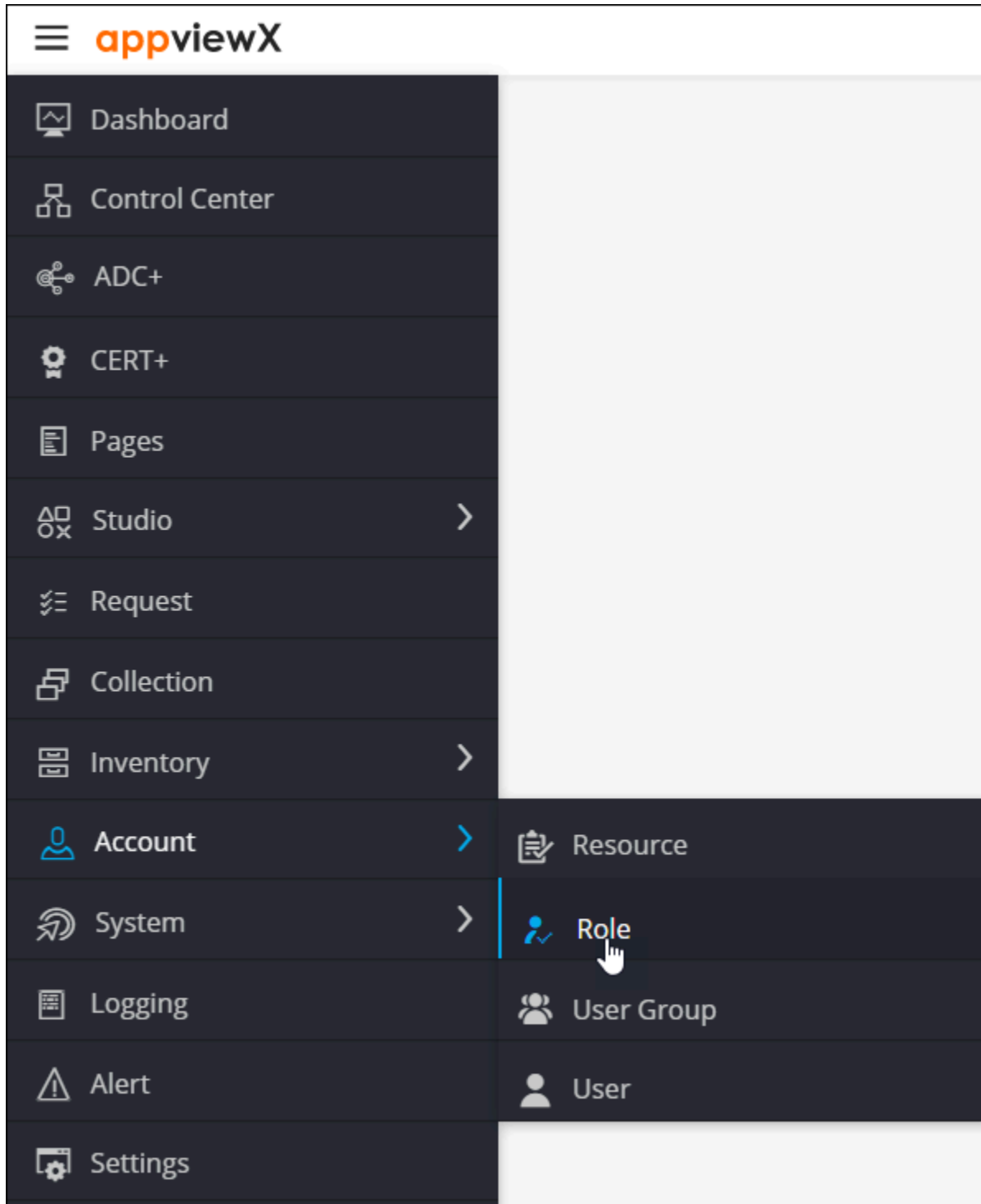
Retrieve all the tasks that are in progress for a specific request. When there are sub workflows involved there could be more than one active in-progress tasks.

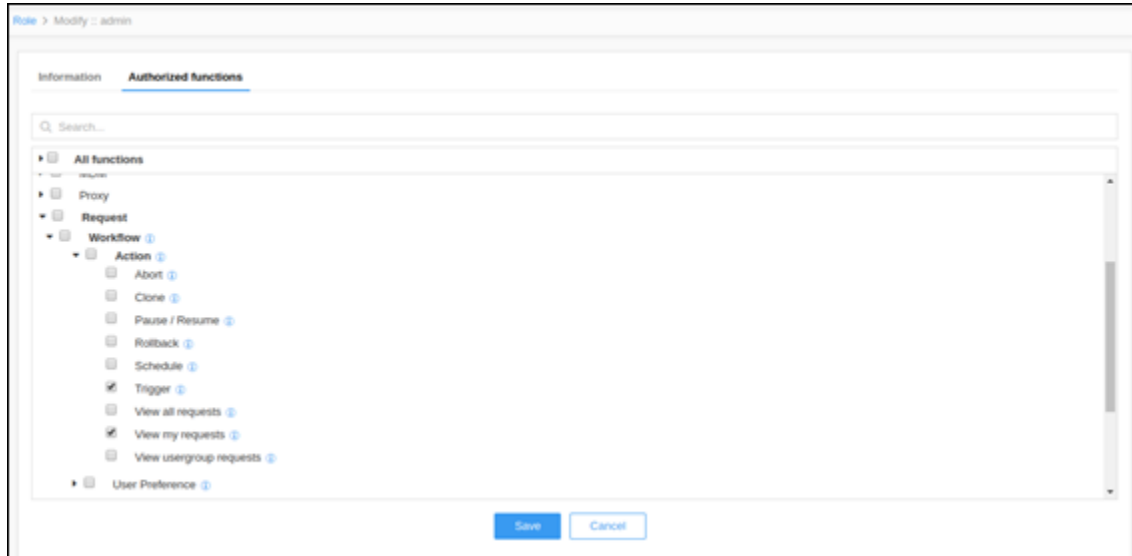
Before you begin

Before attempting to submit a request with this API, ensure that any one of the following ACF permissions are available:

- **Request > Workflow > Action > View All Request**
- **Request > Workflow > Action > View My Request**
- **Request > Workflow > Action > View Role Request**

This can be checked under **Accounts > Role > Authorized Functions**.





Request Structure

URL: /visualworkflow-get-tasks-inprogress

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	sessionId	Session Id received after login	String	Required if username and password are not provided
Header	username	AppViewX login username	String	Required if sessionId is not provided
Header	password	AppViewX login password	String	Required if sessionId is not provided
Header	Content-Type	Specifies the nature of the data in the payload.	String	Value of the param should be 'application/json'
Query	gwkey	Tenant Key. Needed only in case of multi tenant installations	String	NA
Query	gwsouce	Source from which the request is triggered (Eg: web, external)	String	NA

Param Type	Name	Description	Field Type	Constraints
Body	payload	Contains all the params to be sent in the request body for the post request	Any one of the following: Payload	NA

Payload

Name	Mandatory	Description	Field Type	Constraints
request_id	Yes	The request id to update the request details.	String	NA
workorder_id	Yes	The workorder id to update the request details.	String	If there is no workorder in the workflow then the default value will be zero(0)

Response Structure

200 OK returns string of type application/json with the following body paras:

Name	Description	Field Type
response	Contains the response object along with the hierarchy of the object mapped in field children.	Key value pair
message	Success message or failure description in case of error.	String
appStatusCode	Application specific status code for the response. Will be non-null for failure response.	String
tags	More info in case of failure response.	NA

Status Codes

HTTP Status Code	appStatusCode	Message	Possible remediation
200 OK	-	Success	-
400 Bad request	avx-common-028	Invalid/Incorrect payload	Check and ensure if a valid value is given in request payload field - input.
400 Bad request	common-service-007	Input data is not valid	Check and ensure if a valid value is given in request payload field - input requestId/workorderId
404 Not Found	engine-db-011	Request not found	Check and ensure if a valid value is given in request payload field - input requestId/workorderId

Sample Request/Response

To retrieve all the tasks of a request, which are in In progress state.

Sample Request

```
{
  "payload": {
    "requestId": "2",
    "workorderId": 0
  }
}
```

Sample Response

```
{
  "response": {
    "tasksInProgress": [
      {
        "status": "In Progress",
        "taskName": "Implementation"
      }
    ]
  }
}
```

```
}
```

Chapter 8: Upload File

- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Payload](#)
- [Response Structure](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

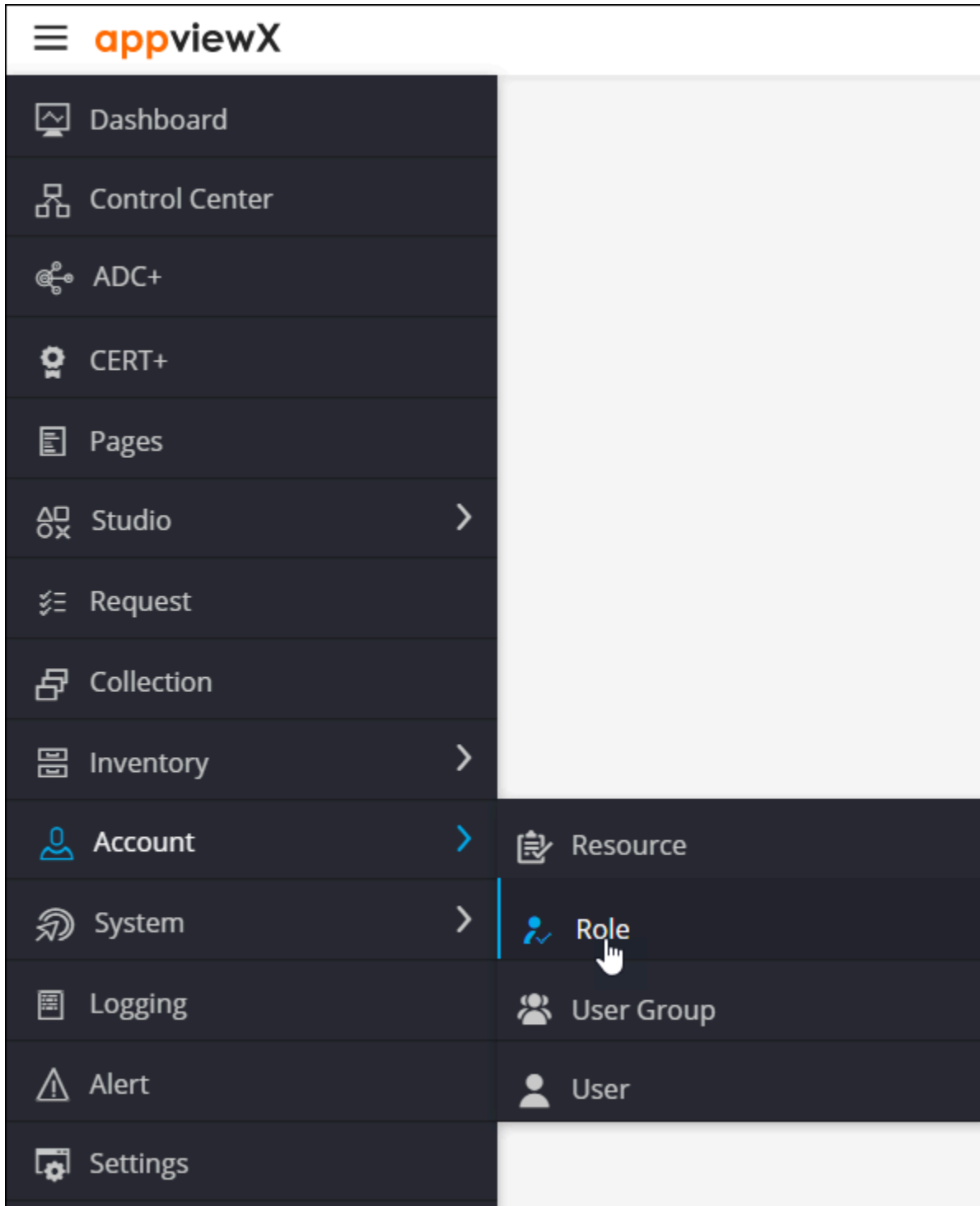
Upload file supports uploading a file and overwriting an uploaded file.

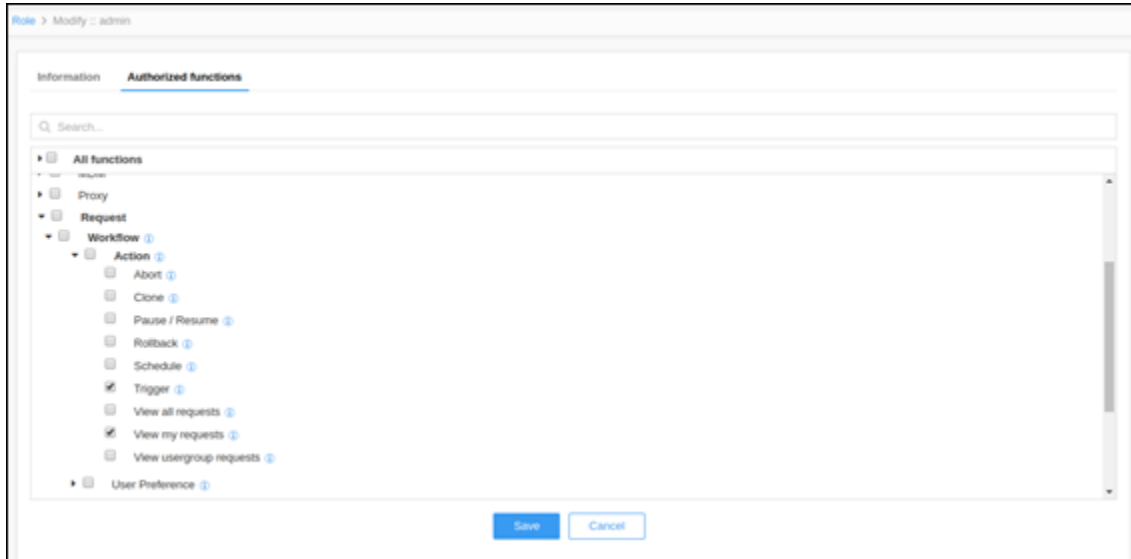
Before you begin

Before attempting to submit a request with this API, users must ensure that the following ACF permission is available:

Studio > Workflow > Import.

This can be checked under **Accounts > Role > Authorized functions.**





Request Structure

URL: /visualworkflow-file-operation-upload

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	sessionId*	Session Id received after login	String	Required if username and password are not provided
Header	username	AppViewX login username	String	Required if sessionId is not provided
Header	password	AppViewX login password	String	Required if sessionId is not provided
Header	Content-Type	Specifies the nature of the data in the payload.	String	Value of the param should be 'multipart/form-data'
Query	gwkey	Tenant Key. Needed only in case of multi tenant installations.	String	NA
Query	gwsouce**	Source from which the request is triggered (Eg: web, external)	String	NA

** - Mandatory value

Payload

Name	Mandatory	Description	Field Type	Constraints
fileName	Yes	File name, allowed file formats : img, jpg, jpeg, png, doc, pdf, txt, gif, zip, ppt, pptx, xls, xlsx, csv, p7b, p7c, pfx, p12, der, pem, csr, crt, cer, key	String	NA
uniqueId	Yes	Unique Id for the uploading file	String	NA
fileOverWrite	Yes	Determines whether to overwrite the uploading file if a file with the same uniqueId exists.	Boolean	False (upload file only if a file with same uniqueId not present) True (upload file and overwrite if file with same uniqueId present, if file not present then upload the given file)
file	Yes	File to be uploaded	Binary	NA

Response Structure

200 OK, returns string of type application/json with the following body params

Name	Description	Field Type
response	Contains the response object along with the hierarchy of the object mapped in field children.	Key value pair
message	Success message or failure description in case of error.	String
appStatusCode	Application specific status code for the response. Will be non-null for failure response.	String
tags	More info in case of failure response.	NA

Status Codes

HTTP code	appStatus code	Message	Possible Remediation
200 OK		Success	
400 Bad Request	StudioService_078	uniqueId field not found	Check and ensure if valid uniqueId is given.
400 Bad Request	StudioService_084	Data cannot be null or empty	Check and ensure if valid data is provided.
400 Bad Request	StudioService_077	File content not found	Check and ensure for valid file.
400 Bad Request	StudioService_071	Invalid file name	Check and ensure if valid fileName is given.
400 Bad Request	StudioService_076	File Name not found	Check and ensure if valid fileName is given
400 Bad Request	StudioService_040	Invalid field format	Check and ensure if valid file and fileName is given.
400 Bad Request	StudioService_075	File with uniqueId already exists	File with uniqueId already exists.

Sample Request/Response

Use Case

Upload file to collection

Request URL

<http://appviewxapi.com/avxapi/visualworkflow-file-operation-upload?gwkey=f000ca01&gwsorce=external>

Sample Request

```
{
  "payload": {
    "fileName": "sample.zip",
    "uniqueId": "uniqueId",
    "fileOverWrite": false,
  }
}
```

```

"file":<%File to be uploaded%>
}
}

```

Sample Response

```

{
  "response": "File Uploaded successfully",
  "message": null,
  "appStatusCode": null,
  "tags": null,
  "headers": null
}

```

Use Case - Overwrite file

Upload a file and overwrite if a file with the same uniqueID is present in the database.

Request URL

<http://appviewxapi.com/avxapi/visualworkflow-file-operation-upload?gwkey=f000ca01&gwsorce=external>

Sample Request

```

{
  "payload": {
    "fileName":"sample.zip",
    "uniqueId":"uniqueId",
    "fileOverWrite":true,
    "file":<%File to be uploaded%>
  }
}

```

Sample Response

```

{
  "response": "File Uploaded successfully",
  "message": "File Uploaded successfully",
  "appStatusCode": null,
  "tags": null,
  "headers": null
}

```

Chapter 9: Download File

- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Payload](#)
- [Response Structure](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

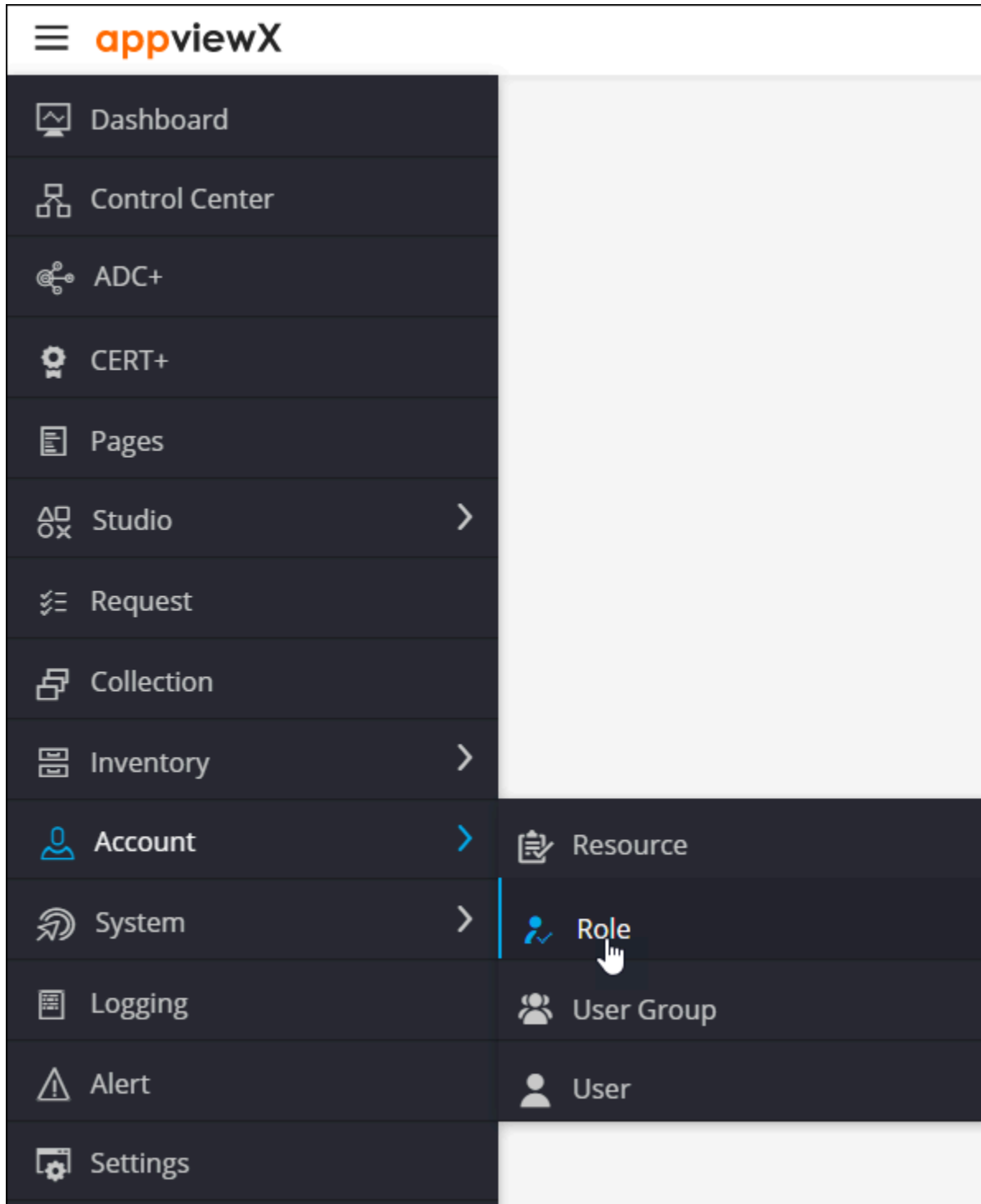
API to download the uploaded file from the database.

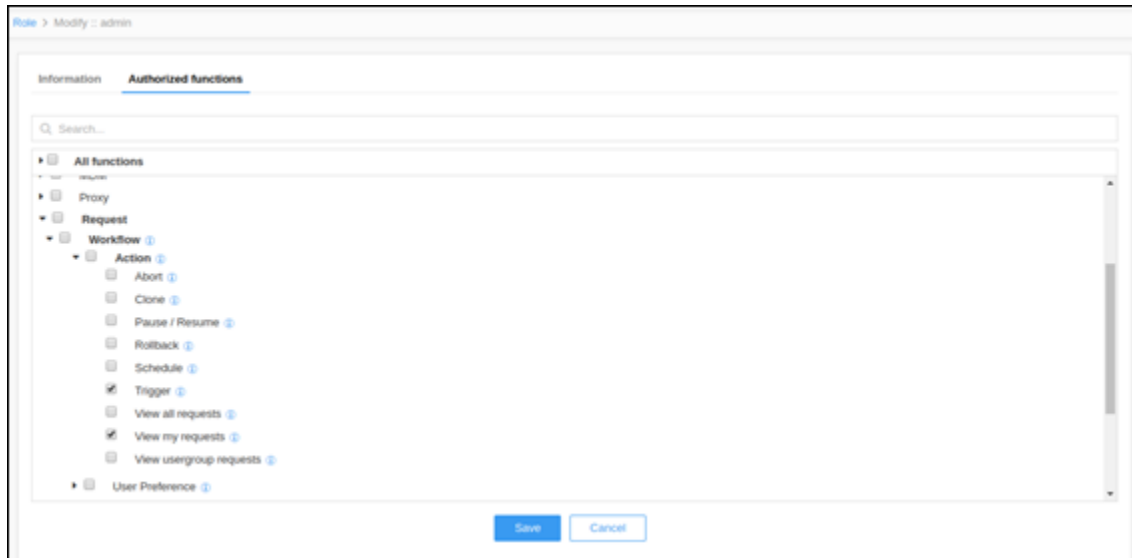
Before you begin

Before attempting to submit a request with this API, users must ensure that the following ACF permission is available:

Studio > Workflow > Export

This can be checked under **Accounts > Role > Authorized functions**.





Request Structure

URL: /visualworkflow-file-operation-download

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	sessionId*	Session Id received after login	String	Required if username and password are not provided
Header	username	AppViewX login username	String	Required if sessionId is not provided
Header	password	AppViewX login password	String	Required if sessionId is not provided
Header	Content-Type	Specifies the nature of the data in the payload	String	Value of the param should be 'application/json'
Query	gwkey	Tenant Key. Needed only in case of multi tenant installations	String	NA
Query	gwsouce**	Source from which the request is triggered (Eg: web, external)	String	NA

Payload

Name	Mandatory	Description	Field Type	Constraints
uniqueId	Yes	Uniqueld of the file to be downloaded	String	NA
decodeFile	No	Decide whether the file is to be decoded while downloading.	Boolean	NA

Response Structure

200 OK, returns string of type application/json with the following body params

Name	Description	Field Type
response	Contains the response object along with the hierarchy of the object mapped in field children.	Key value pair
message	Success message or failure description in case of error.	String
appStatusCode	Application specific status code for the response. Will be non-null for failure response.	String
tags	More info in case of failure response.	NA

Status Codes

HTTP Code		Description	Response message
200 OK		Success	
400 Bad Request	StudioService_078	uniqueID field not found	Check and ensure if valid uniqueId is given.
400 Bad Request	avx-common-028	Invalid/Incorrect payload	Check and ensure if a valid value is given in request payload field - input
404 Not Found	studio-db-043	File not found	Check and ensure if file with a uniqueId is not present.

Sample Request/Response

Request URL

<http://appviewxapi.com/avxapi/?gwkey=f000ca01&gwsouce=external>

Sample Request

```
{  
  "payload": {  
    "uniqueId": "uniqueId" }  
}
```

Sample Response

```
File to be downloaded
```

Chapter 10: Delete File

- [Overview](#)
- [Before you begin](#)
- [Request Structure](#)
- [Payload](#)
- [Response Structure](#)
- [Status Codes](#)
- [Sample Request/Response](#)

Overview

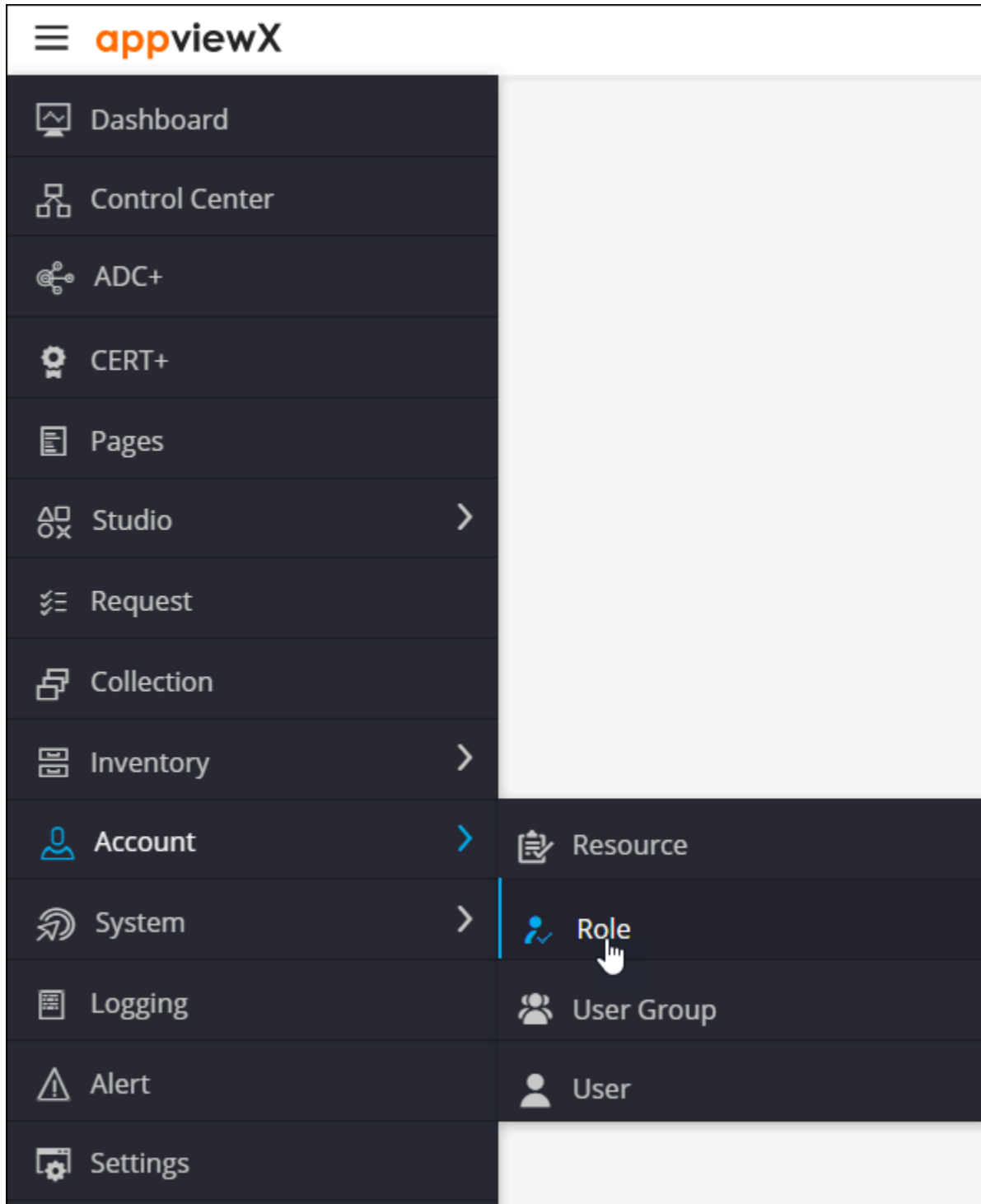
API to delete an uploaded file from the database.

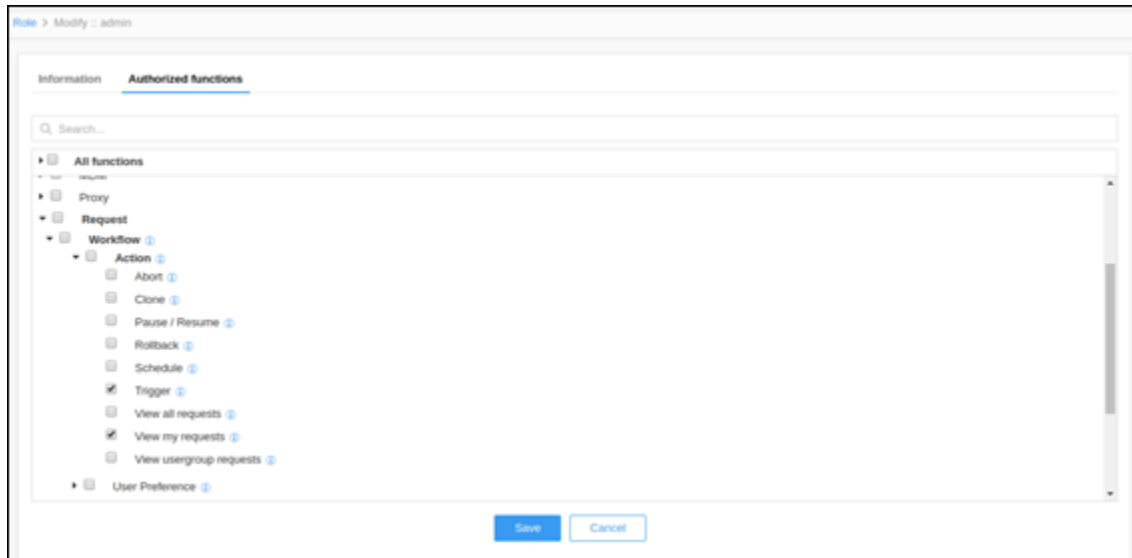
Before you begin

Before attempting to submit a request with this API, users must ensure that any one of the ACF permission is available:

Studio > Workflow > Delete

This can be checked under **Accounts > Role > Authorized functions**.





Request Structure

URL: /visualworkflow-file-operation-delete

Type: POST

Param Type	Name	Description	Field Type	Constraints
Header	sessionID**	Session Id received after login	String	NA
Header	Content-Type	Specifies the nature of the data in the payload	String	Value of the param should be 'application/json'
Query	gwkey	Tenant Key. Needed only in case of multi tenant installations	String	NA
Query	gwsouce**	Source from which the request is triggered (Eg: web, external)	String	NA

** - Mandatory value

Payload

Name	Mandatory	Description	Field Type	Constraints
uniqueId	Yes	Uniqueld of the file to be deleted	String	NA

Response Structure

200 OK returns string of type application/json with the following body paras:

Name	Description	Field Type
response	Contains the response object along with the hierarchy of the object mapped in field children.	Key value pair
message	Success message or failure description in case of error.	String
appStatusCode	Application specific status code for the response. Will be non-null for failure response.	String
tags	More info in case of failure response.	NA

Status Codes

HTTP code	appStatus Code	Message	Possible Remediation
200 OK		Success	
400 Bad Request	StudioService_078	uniqueId field not found	Check and ensure if valid uniqueId is given
404 Not Found	studio-db-043	File not found	Check if file with uniqueid not present
400 Bad Request	avx-common-028	Invalid/Incorrect payload	Check and ensure if a valid value is given in request payload field - input

Sample Request/Response

Request URL

<http://appviewxapi.com/avxapi/visualworkflow-file-operation-delete?gwkey=f000ca01&gwsource=external>

Sample Request

```
{  
  "payload": {  
    "uniqueId": "uniqueId" }  
}
```

Sample Response

```
{  
  "response": "File Deleted successfully",  
  "message": null,  
  "appStatusCode": null,  
  "tags": null,  
  "headers": null  
}
```