CTX-Logging-Visibility User Guide

Contents

Versions

Document Revisions

The following revisions have been made to this document

Date	Revision	Notes
15/10/2018	0.1	First Draft
24/01/2022	0.2	Second Draft

Module Versions

The following revisions have been made to this document

Date	Revision	Notes
15/10/2018	1.0	Creation of Logging-Visibility Flows and Subtasks
24/01/2022	1.1	Update of the Logging-Visibility-UI Flow an subtasks

Preface

About this document

This document is a user guide for the CTX-Logging-Visibility module.

Audience

This document is intended for those who require the use of the CTX-Logging-Visibility module for visibility of the logging done automatically by Cortex and viewing information about flow executions and interactions with external systems.

Related Material

Document
CTX-Logging-Visibility – User Guide
CTX-Logging-Visibility.studiopkg

Abbreviations used in this Document

API Application Programming Interface

OCI Oracle Call Interface

Requirements

There are no requirements for this module.

Integration

Integration with Third-Party Systems

None Required.

Integrating with Existing Infrastructure

None Required.

1 Logging-Visibility-UI

1.1 Overview

To access the logging visibility user interface, run “Logging-Visibility-UI’ from LivePortal.

When the flow is executed, the home screen will be shown (Figure 1) where the user can select which of the actions listed below, they wish to perform:

- View API Details
- View Flow History
- View Error Investigation

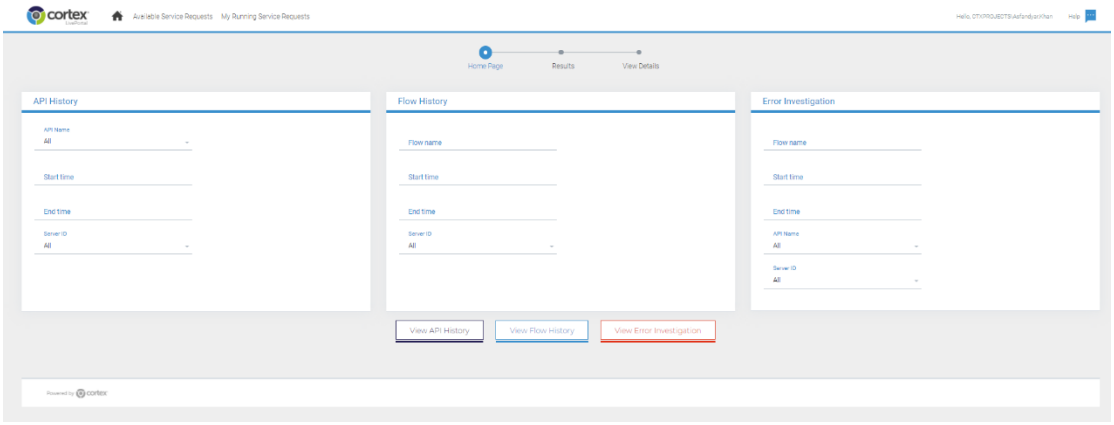

Figure 1: Logging-Visibility Home Page

The API history shows Cortex interactions with a specific (or all) API in a specified time frame.

The flow history shows executions of a flow in a specified time frame.

Error investigation allows the user to view flow executions and API interactions specified time frame when we are unsure of the flow and/or the API name.

The results can be filtered by the Server ID. Detail on specific API interactions and flow executions can be viewed in the “View API detail” and “View execution detail” pages.

1.1.1 View API history

To view interactions with an API within a time frame:

1. Select and API name from the dropdown and input the **Start time** and **End time**.
    a. You may select **All** to view all API interactions that were logged.
    b. Select server ID if required.
2. Click “View API History” navigation button.
3. View interactions with that API in the time frame.
    a. To view more detail on a specific API, select that row’s checkbox and click **View API details**
    b. From here, you may return to **Home** or **View Execution Details** for more information on that flow execution (for all APIs associated with that flow)

1.1.2 View flow history

To view executions of a flow within a time frame:

1. Input a **Flow name**, **Start time** and an **End time**.
    a. Select **Server ID** if required.
2. Click the **View Flow History** navigation button.
    a. If the flow name is not recognised, a list of possible flow names is displayed.
3. See executions of that flow within the specified time frame
    a. Select a specific execution's checkbox and click **View Execution details** to view more detail for that execution.

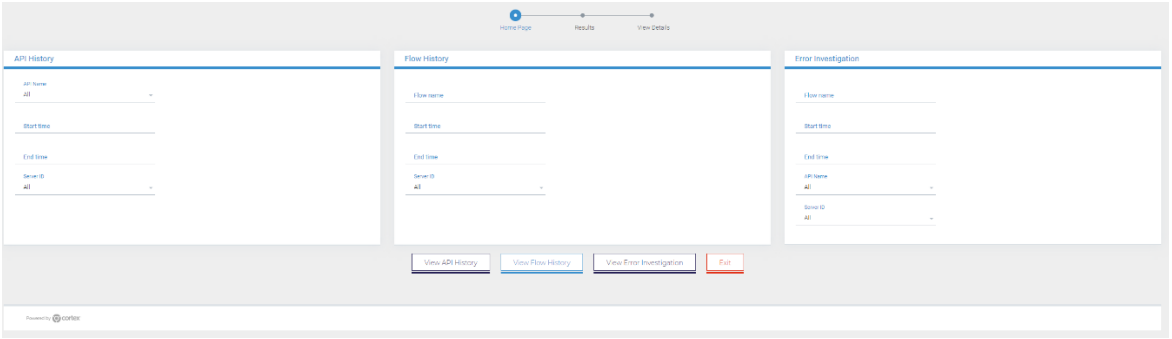

Figure 2: Flow History page showing a list of the specified flow within a selected timeframe

1.1.3 Error investigation

The error investigation function allows you to view both the Flow and API History, and is used for when either the Flow name, API name or both are unknown.

On the home screen under the Error Investigation panel:

    a. Input a **Flow name** if known.
    b. Input a **Start time** and **End time**.
    c. Select **API name** if known.
    d. Select **Server ID** if required.

2. Click **View Error Investigation** navigation button.
    a. If the flow name is not recognised, a list of possible flow names is displayed.
3. The screen will show executions and/or API interactions within the specified time frame. You may now either:
    a. Select a specific flow execution under the **Flow History** panel and click **View Flow Execution** to view more detail on that execution.
    b. Select a specific API interaction and click **View API History** to view more detail on that interaction.

More detail on specific flow executions and API interactions is provided in the "View API detail" and "View Execution detail" pages.

1.1.4 View Flow Execution detail

To view details relating to a specific execution of a flow:

    a. Select a single API interaction or flow execution on the pages **API History**, **Flow History** or **API detail** by ticking its respective checkbox
        I. On the **Error Investigation** page, the results can show both the Flow and API History. In this case, please select a single flow execution underneath the **Flow History** tab alone, as shown in Figure 3.

b. Click **View Execution Details**

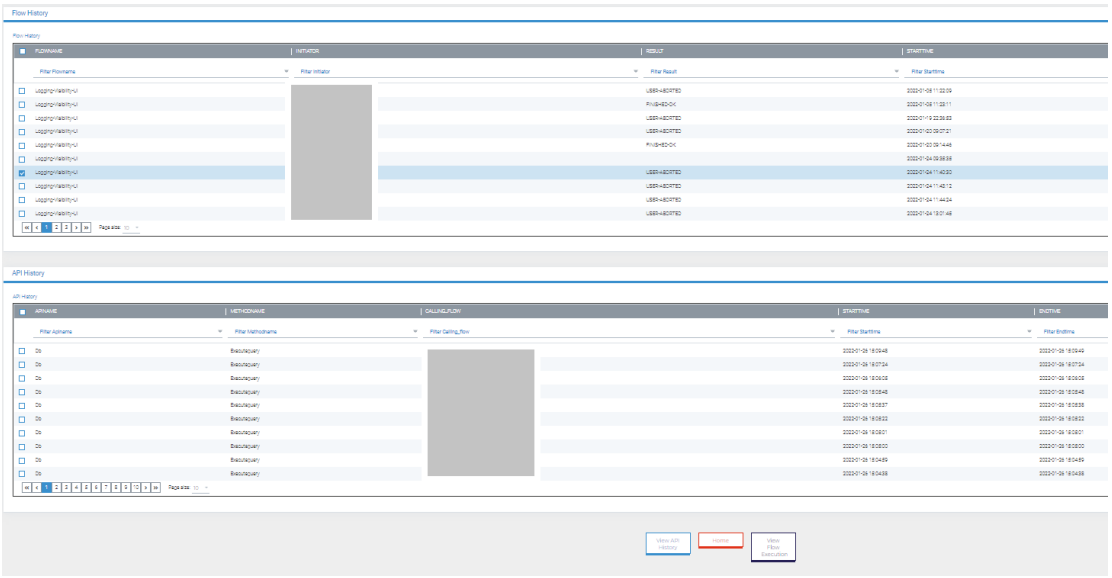

Figure 3: Error Investigation page showing both Flow and API History

To view details relating to a specific API interaction:

c. Select a specific API interaction on the pages **API History**, **Flow History** or **Flow Execution detail**.

d. On the **Error Investigation** page, the results can show both the Flow and API History. In this case, please select a single flow execution underneath the **API History** tab alone

e. Click **View API details**.

1.1.5 Invalid search handling

If no log entries are found the user is prompted to change the search parameters.

If many log entries are returned, the top 100 are displayed and the user is informed.

If the user-specified time frame is invalid, the flow will default to a valid alternative:

- If no time frame has been entered, defaults to **Start time** 3 days before now, and **End time** to the current system time.
- If no **End time** has been entered, defaults to a day after start time.
- If no **Start time** has been entered, defaults to a day before start time.
- If **Start time** is after **End time**, they are switched.

1.2 Inputs

None.

1.3 Outputs

None.

2 LVGLD-Logging-Visibility-Get-Log-Data

2.1 Overview

Depending on its inputs, this subtask will return the following:

- A flow's history, i.e., its name, which user started it, what times it was run between and its result.
- An API's history, i.e., its name, the method by which it was called, the flow that called it, when it was called and its result.
- An API's Details.
- An Execution's details, including the history of the APIs it calls
- A list of all executions from the past month and a list of all available APIs, to be displayed on the home page.

2.2 Inputs

Input Variables	Type	Description
LVGLD_i_Data-required	Text	The data to search the database for. • Allowed Values: "HOME-PAGE", "FLOW-HISTORY", "API-HISTORY", "EXECUTION-DETAIL", "API-DETAIL", "ERROR-DETAIL"
LVGLD_i_DB-connection-string	Text	Connection string for use in connection to the required database. Example: "Server=V-CTXAPPDB33; Database=Reactor; Trusted_Connection=True;"
LVGLD_i_Server-ID	Text	ID of the server about which to find data. Example: "1"
LVGLD_i_Flow	Structure	Contains 3 elements describing the properties of a flow. Example: { "FLOWNAME": "LOGGING-VISIBILITY-UI", "STARTTIME": "2018/09/24 00:00:00", "ENDTIME": "2018/09/28 11:01:19" }

LVGLD_i_Execution	Structure	Contains 6 elements describing properties of one execution of a flow. Example: { "FLOWNAME": "Logging-Visibility-Ui", "INITIATOR": "SmithJ", "RESULT": "FINISHED-OK", "STARTTIME": "2018-09-26 15:58:52", "ENDTIME": "2018-09-26 16:10:20", "EXECUTIONID": "ae60d39fc19c11e8966a005056910a64" }
LVGLD_i_API	Structure	Contains 3 elements describing the properties of an API. Example: { "APINAME": "PowerShell", "STARTTIME": "2018/09/24 00:00:00", "ENDTIME": "2018/09/28 11:01:19" }

LVGLD_i_API-details	Structure	Contains 9 elements describing properties of one call of an API.
		
		Example:
		{
		"APINAME": "Powershell",
		"FLOWNAME": "<name of flow in which this API was called>",
		"RESULT": "Success",
		"METHODNAME":
		"Executepowershellscript",
		"STARTTIME": "2018-09-27 17:06:33",
		"ENDTIME": "2018-09-27 17:06:33",
		"ARGUMENTS": <PowerShell script>
		"RETURNVALUE": “Structure with 6 values: …”
		"EXECUTIONID":
		"4a1de5d5c26f11e8966a005056910a64"
		}

2.3 Outputs

Output Variables	Type	Description
LVGLD_o_Flow-names	List	List of all flows executed in the last month.
LVGLD_o_API-names	List	List of available APIs.
LVGLD_o_Server-IDs	List	List of all server Ids.
LVGLD_o_Flow-history	Table	Table of Flows that fit the search requirements. Fields are “Flowname”, “Initator”, “Result”, “Starttime”, “EndTime”, “ExecutionID”.
LVGLD_o_Execution-detail	Table	Table of properties of one particular execution of a flow. Fields are “Apiname”, “Methodname”, “Result”, “Starttime”, “Endtime”, “Executionid”.
LVGLD_o_Execution-variables	Table	Table of global variables used in an execution and their final values.
LVGLD_o_API-History	Table	Table of all calls of an API within a user specified time period. Fields are “Apiname”, “Methodname”, “ExecutionID”, “Callingflow”, “Starttime”, “EndTime”, “Result”.
LVGLD_o_API_Detail	Table	Table of properties of one call of an api. Fields are “Apiname”, “Flowname”, “Result”, “Methodname”, “Starttime”, “EndTime”, “Arguments”, “Returnvalue”, “ExecutionID”.
LVGLD_o_Result	List	The Result of an API call.

3 LVQD-Logging-Visibility-Query-Database

3.1 Overview

This subtask will perform a query on a SQL database. It is a generic subtask which is utilised by the CTX-Logging-Visibility module, invoked by the LVGLD-Logging-visibility-Get-Log-Data subtask.

Exceptions will be raised if:

- The specified database cannot be found.
- The SQL query is invalid.
- The database connection cannot be closed.

3.2 Inputs

Input Variables	Type	Description
LVQDB_i_env	Structure	Environmental Structure Variable
LVQDB_i_DBConnectionString	Text	Connection string for use in connection to the required database. Example: "Server=V-CTXAPPDB33; Database=Reactor; Trusted_Connection=True;"
LVQDB_i_DBQuery	Text	SQL Query to be executed

3.3 Outputs

Output Variables	Type	Description
LVQDB_o_QueryResults	Table	Result of the query i_qdb-DBQuery
LVQDB_o_OutputStructure	Structure	If the number of rows returned by the query is zero, then this empty structure will be returned.
LVQDB_o_RowCount	Integer	Number of rows in the table returned by the query

