Interface Specification Technical Document

Interface Name: TSD Application Log

Generated Document Name: TSD Application Log

Market / Project:

Author:Shaik Faruk

# Document Information

### Document Location

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Name** | **Sol Man Folder Path** | **Document #** | **Solman Link** |
|  |  |  |  |
|  |  |  |  |

### Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Revision Number** | **Revision Date** | **Project** | **Summary of Changes** | **Revised By(GPID/Name)** |
| *1* | *02/07/2024* |  | *Released 1st version* | *Shaik Faruk* |

**Design Authority Approval Details**

|  |  |
| --- | --- |
| **Approval Date** | **Approved By** |
|  |  |

Table of Contents

[0. Document Information 1](#_Toc654678688)

[Document Location 2](#_Toc1909001828)

[Revision History 2](#_Toc31443981)

[1. General Information 3](#_Toc1336514945)

[2. High Level Technical Design 4](#_Toc727972226)

[2.1 High level Business Requirement 4](#_Toc952976355)

[2.2 Possible Technical approaches with brief pros & Cons 5](#_Toc702384976)

[2.3 Proposed Technical approach 5](#_Toc379192933)

[3. Technical Details / Process Flows 5](#_Toc1835066912)

[4. Technical Details 5](#_Toc665222849)

[4.1 Interface Type 6](#_Toc1068459775)

[4.2 Selection Prototype 6](#_Toc310269015)

[4.3 Selection Screen Validations 6](#_Toc219016255)

[4.4 Object/Import Type 6](#_Toc1450257453)

[4.5 Legacy System Extract / Load procedure 6](#_Toc182352675)

[4.6 Data Mapping Matrix 6](#_Toc981386573)

[4.7 Inbound / Outbound File Information 6](#_Toc1357287569)

[4.8 Custom Transaction 6](#_Toc688591736)

[5. Additional Information and attachments 6](#_Toc448926370)

[6. SAP Development Objects 7](#_Toc50227222)

[7. Unit Test Plan 7](#_Toc2138944658)

# General Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Information** | | | |
| Development ID: |  | | |
| FD Link |  | | |
| Version |  | | |
| Area: |  |  |  |
| Technical Designer: |  |  |  |
| Technical Resource: |  |  |  |
| Title: |  | | |
| Short Description: |  | | |

# High Level Technical Design

## High level Business Requirement

To show the context data along with messages while displaying application logs based on object and sub-object.

***Optional Scope***

* Generally when the application log create for application the execution of application can be tracked and can identify the error easily.
* But through slg1 transaction while displaying application log based on object and sub object the context data is not shown even context is added to the logs.
* so if context is present for logs display the context data with message while click on Application log.

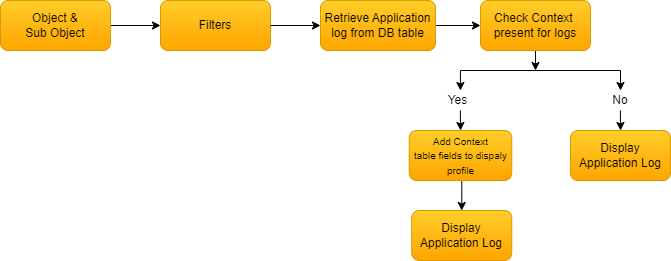
## Possible Technical approaches with brief pros & Cons

To meet the business requirements, we need to create report.

**Report**

To display context data based on the object and sub-object with filters, for that only this report is required. A selection screen is created to read the object, subject, and filter.

Based on that filter, the application log can be retrieved from the corresponding db table. Then checks if the context structure table name is present in the collected application logs. If present, the table fields will be added to the mess-fcat field in the display\_profile structure and than display the logs.



## Proposed Technical approach

Following are the objects which needs to be created.

* **Objects**

|  |  |
| --- | --- |
| *Object Type* | *Object Name* |
| Report | ZGREP\_APP\_LOG\_DISPLAY |

# Technical Details / Process Flows

Created report program – ZGREP\_APP\_LOG\_DISPLAY

* A selection screen has been created to take inputs for object and sub-object

and Date&time for filter(to get the logs Upto the mentioned Date and Time.

* Based on selection screen data the logs will be search from db table for data availability.

If the logs are found it will be loaded in the internal table.

* For display log header data a “display profile” structure is obtains from ‘BAL\_DSP\_DISPLAY\_SINGLE\_LOG\_GET’ function module ( if the no of logs equal to 1 ) or

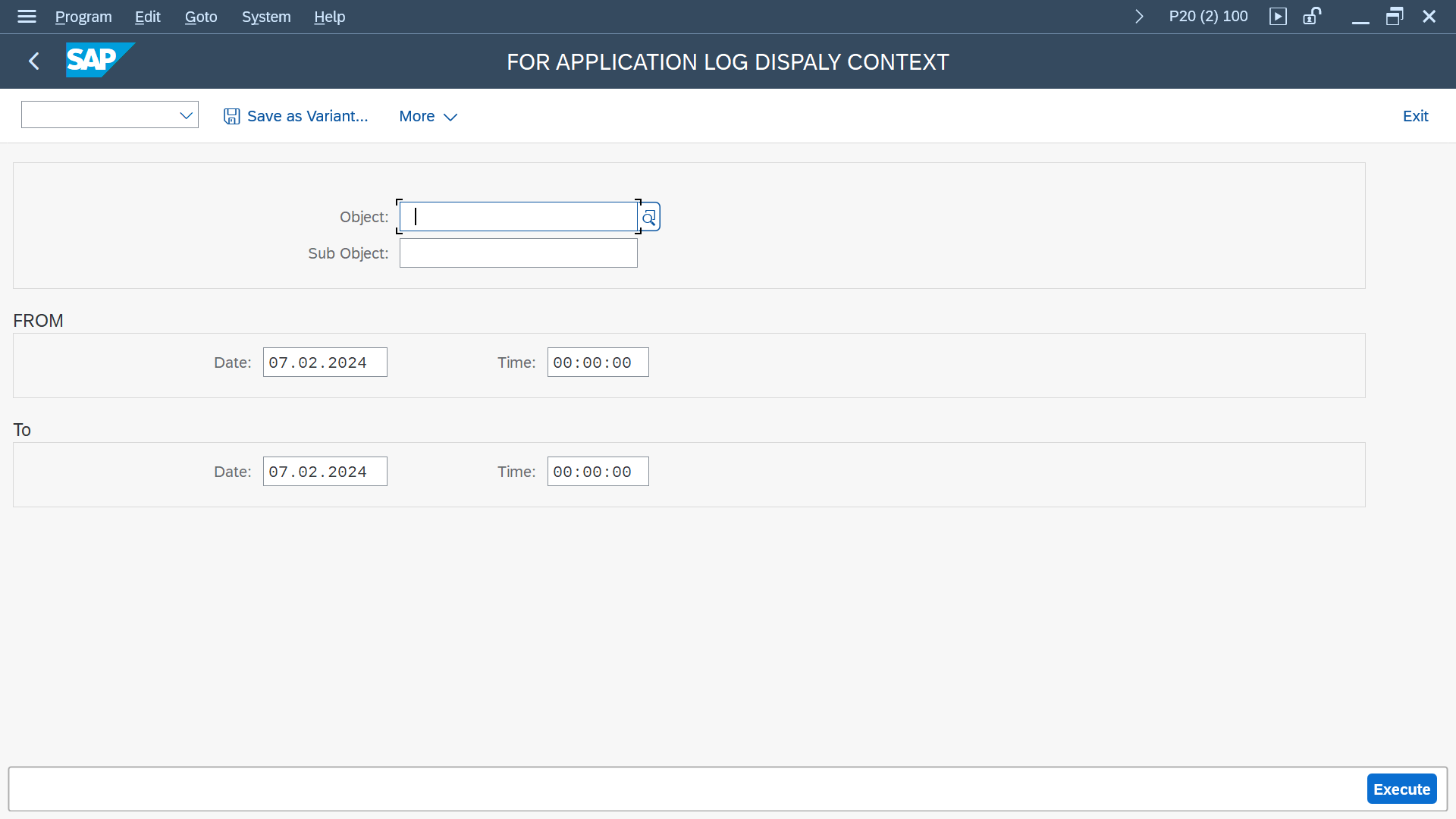
‘BAL\_DSP\_DISPLAY\_STANDARD\_GET’ function module (if no of logs > 1 ).

* Then check will apply on the log internal table that tabname field is not initial.

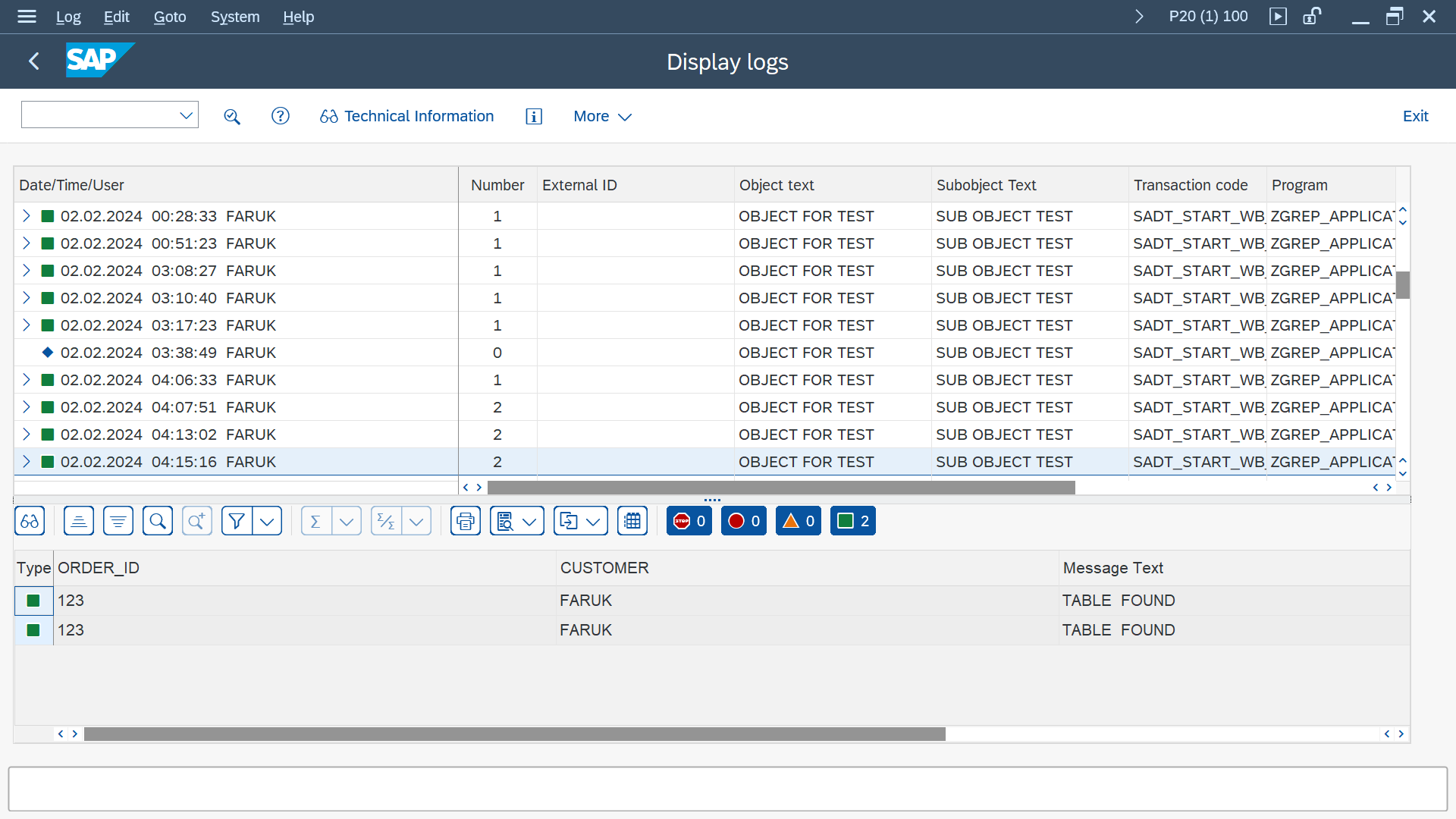
If tabname is not empty add the field of tabname(Global Structure) to the mess-fcat field in the display profile structure.

* Based on the Application log table and display profile structure the data will be displayed through ‘BAL\_DSP\_LOG\_DISPLAY’ function module ( the log header data will be display in ALV tree and its replaced messages with context will be displayed in grid using docking container ).

Selection screen for object , sub-object and date&time



Display Application log header data with messages including context.



# Technical Details

## Interface Type

## Selection Prototype

## Selection Screen Validations

## Object/Import Type

**Standard Batch/Direct Input**

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Name** | **Method Name** | **Program Name** | **Type** |
|  |  |  |  |

**Batch Input Recording**

|  |  |
| --- | --- |
| **Transaction Code** | **Recording Name** |
|  |  |

**Business Object Method (BAPI)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Business Object** | **Method** | **Message Type** | **Basic Type** |
|  |  |  |  |

**Intermediate Doc (IDoc)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Message Type** | **Basic Type** | **Extension** | **Settings Information** |
|  |  |  |  |

## Legacy System Extract / Load procedure

## Data Mapping Matrix

## Inbound / Outbound File Information

## Custom Transaction

# Additional Information and attachments

* 1. **Reconciliation Reporting**

If a specific report or additional information is required is required for reconciliation

* 1. **Attachments**

# SAP Development Objects

|  |  |
| --- | --- |
| SAP Development Objects Development Class |  |
| Message Class |  |
| Package |  |
| Transport Request Number |  |

|  |  |
| --- | --- |
| Object type | Object Name |
| Table |  |
| Data element |  |
| Domain |  |
| Program Name | ZGREP\_APP\_LOG\_DISPLAY |
| Include | ZGINC\_APP\_LOG\_DISPLAY\_SEL  ZGINC\_APP\_LOG\_DISPLAY\_CLS  ZGINC\_APP\_LOG\_DISPLAY\_EVT |
| Structure |  |
| Transaction | ZG\_APLOG |
| Single Message |  |
| Text Module |  |
| Class/Interface |  |

# Unit Test Plan

Normal Functionality - test cases that ensure the workflow functions as it should. (e.g. updates fields correctly, processes all records)

| **Test Condition** | **Step** | **Step Description** | **Test Data** | **Expected Result** | **Actual Result** | **Executed By/Date** | **Remarks** |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

Exception - special logic or exceptions (e.g. do not process Government Markets customers, only process pre-packs)

| **Test Condition** | **Step** | **Step Description** | **Test Data** | **Expected Result** | **Actual Result** | **Executed By/Date** | **Remarks** |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

Error Handling - functionality in case of errors (e.g. Customer not found, Record already exists)

| **Test Condition** | **Step** | **Step Description** | **Test Data** | **Expected Result** | **Actual Result** | **Executed By/Date** | **Remarks** |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |