LARGE SYNOPTIC SURVEY TELESCOPE

PROD Software Test Specification

prepared by: theAuthor
approved by: theApprover
reference: LDM-xxxx
issue: 0D
revision: 1
date: 2017-01-15
status: draft

Abstract

This is a template for the Software Test Specification document for PROD. It encompasses the test design, test case and test procedure specification.

LARGE SYNOPTIC SURVEY TELESCOPE

Document History

Issue	Revision	Date	Author	Comment
D	1	yyyy-mm-dd	WOM	First draft

Contents

1 Introduction

This section should provide an overview of the entire document and a description of the scope of testing.

1.1 Objectives

Test specification document helps in refining the test approach that was planned for executing the test plan. It identifies the test cases, procedures and the pass/fail criteria for the assignment. It outlines the actual values required as input parameters in the testing process and the expected outputs of the testing results. It also identifies the various constraints related to the test case.
It is important to note that test cases can be reusable components and one test case can be used in various test designs.
The test procedure outlines all the processes that are required to test the system and implement the test cases.

1.2 Scope

Define the scope (refining the detailed in the Software Test Plan RG-004) of the testing whose specifications are described in this document.

1.3 Applicable Documents

RG-004	DPAC Software Test Plan
RD-010	Gaia DPAC Project Development Plan
TL-001	DPAC Product Assurance Plan
WOM-011	DPAC Software Engineering Guidelines
TLO-001	ECSS Tailoring

1.4 References

[1] **[RD-010]**, Drimmel, R., Els, S., O'Mullane, W., et al., 2014, *DPAC Project Development Plan*,
GAIA-CD-PL-INAF-RD-010,
URL http://www.rssd.esa.int/cs/livelink/open/2786669

[2] **[RG-004]**, Guerra, R., leaders CU leaders, D., 2013, *DPAC System Validation and Test Plan*,

GAIA-C1-SP-ESAC-RG-004,
URL http://www.rssd.esa.int/cs/livelink/open/2898933

[3] **[TL-001]**, Levoir, T., Damery, J., Hoar, J., et al., 2012, *DPAC Product Assurance Plan*,
GAIA-C1-PL-CNES-TL-001,
URL http://www.rssd.esa.int/cs/livelink/open/2439085

[4] **[TLO-001]**, Lock, T., 2007, *Software Engineering Standards (ECSS-E-40B) - Tailored for Gaia Science Ground Segment*,
GAIA-C1-TN-ESAC-TLO-001,
URL http://www.rssd.esa.int/cs/livelink/open/2786522

[5] **[WOM-011]**, O'Mullane, W., Hoar, J., Levoir, T., et al., 2011, *Software Engineering Guidelines for DPAC*,
GAIA-C1-UG-ESAC-WOM-011,
URL http://www.rssd.esa.int/cs/livelink/open/2760364

1.5   Definitions, acronyms, and abbreviations

The following table has been generated from the on-line Gaia acronym list:

Acronym	Description
AGIS	Astrometric Global Iterative Solution
AO	Announcement of Opportunity
CCB	Configuration Control Board
CM	Configuration Management
CU	Coordination Unit (in DPAC)
DOC	Department of Commerce (USA)
DPAC	Data Processing and Analysis Consortium
DPC	Data Processing Centre
DU	Detection Unit
ECSS	European Cooperation for Space Standardisation
ESA	European Space Agency
ESAC	European Space Astronomy Centre (VilSpa)
GWP	Gaia Work Package
PA	Product Assurance
QA	Quality Assurance
SCMP	Software Configuration Management Plan
SRS	Software Requirements Specification
SW	Software (also denoted S/W)

TOC	Table of Contents
WBS	Work Breakdown Structure
WP	Work Package

1.6   Document Overview

This section shall describe the contents of the document and explains how the rest of the report
is organized.

2   Approach

Describe the approach to be utilized for the software testing specification. It should identify the major activities, methods and tools that are to be used to test the designated group of features.

2.1   Tasks and criteria

Describe which are the items under tests, as well as criteria to be utilized. Activities should be described in sufficient detail to allow identification of the major testing tasks and estimation of the resources and time needed for the tests.

2.2   Features to be tested

Describe the GENERAL features to be tested.

2.3   Features not to be tested

Describe all the features and significant combinations not to be tested and explain why. If it is not possible to test some features at their most appropriate level of testing but which will be tested at a later level, this information should be included here

2.4   Pass - Fail criteria

GENERAL criteria to be used to determine whether or not tests are passed.

2.5   Suspension criteria and resumption requirements

Describe the criteria used to suspend all, or a part of, the testing activities on the test items associated with the plan, as well as the activities to be repeated when testing is resumed.

2.6   Testing Procedure Compliance

This section must specify whether the system adheres to the testing procedure defined in RG-004 or not.
In case it adheres to it it can state something like: ”The System testing is in compliance with the Testing Procedure defined for DPAC, consisting on a Test Readiness Review and a Test Review Board (details in RG-004). The Test Review Board members are ...”

3   Specification Design Overview

Specify refinements of the test approach described in RG-004 if any. If not suppress this section
.

4   DM-PROD-SUBSYST-VER-XX

4.1   Objective

This test design includes all the verification methods different to the dynamic testing, needed
for the demonstration of the fulfillment of specific requirements of the SRS.
These are the requirements that do not need the execution of the software systems, but a careful
document or code inspection, review or analysis.

4.2   Features to be tested

Identify the test items and describe the features and combinations of features that are the object
of this design specification. E.g.:

- Demonstrate that the CU has produced an specific document (e.g. the Software
  Development Plan)

- Demonstrate the behaviour or value of an specific parameter (e.g. "As a minimum
  the PPN parameter shall be included in the global model"

- Static analysis using tools such as Findbugs or checkstyle, etc.

- etc

The features to be tested depends on the content of the Software Requirements Document.
Therefore the above list is just an example.

4.3   Approach refinements

Specify the verification methods to be used to verify the features describe in the section above.
These methods are described in the DPAC SVTP [RG-004]. E.g.:

- Code inspection

- Document inspection

- Review

- Static analysis

- etc

4.4 Test case identification

List the identifier and a brief description of each test case associated with this design.

Test Case	Description
DM-PROD-SUBSYST-VER-XX-05	SDP inspection
DM-PROD-SUBSYST-VER-XX-10	SDD inspection
DM-PROD-SUBSYST-VER-XX-15	Code inspection
DM-PROD-SUBSYST-VER-XX-20	Testing review
etc	

The number and scope of the test cases depend on the requirements to be verified. The list provide above is only an example.

4.5 Feature pass/fail criteria

Specify the specific criteria for this design to be used to determine whether the feature or feature combination has passed or failed.

4.6 Test Case DM-PROD-SUBSYST-VER-XX-YY

4.6.1 Requirements

Specify the requirements that fulfill the test case, comma-separated and ending with '.'
e.g. CU3-IDT-XM-FUN-30,CU3-IDT-ASD-FUN-20.
Note that the format is essential for the script that creates the traceability matrices works fine.

4.6.2 Test items

Identify and briefly describe the items and features to be exercised by this test case (e.g.:

- demonstrate that an specific document is written or that it contains the information

required

- or...

- demonstrate that a parameter is set to a specific value

- or...

- the code is written in java

4.6.3 Intercase dependencies

List the identifiers of the test cases that must be executed prior to this test case. Summarize the nature of the dependencies.

4.6.4 Procedure

If the procedure is shared by various test cases it would be recommended to separate the procedure description from the test case definition. Thus, only a reference to the procedure identification shall be provided. The complete description of the test procedure shall be given otherwise.

Describe any special constraints on the test procedures that execute this test case. These constraints may involve special set up, operator intervention, output determination procedures, and special wrap up.

5 DM-PROD-SUBSYST-SCOPE-XX

5.1 Objective

Specify the objective of this test design.

5.2 Features to be tested

Identify the test items and describe the features and combinations of features that are the object of this design specification. Other features may be exercised, but need not be identified.
For each feature or feature combination, a reference to its associated requirements should be included.

5.3 Approach refinements

Specify refinements to the approach described in the test plan. Include specific test techniques to be used. The method on analysing test results should be identified.
Specify the results of any analysis that provides a rationale for test case selection.
Summarize the common attributes of any test cases. This may include input constraints that must be true for every input in the set of associated test cases, any shared environmental needs, any shared special procedural requirements and any shared case dependencies.

5.4 Test case identification

List the identifier and a brief description of each test case associated with this design.

Test Case	Description
DM-PROD-SUBSYST-SCOPE-XX-YY	Description of the test case

5.5 Feature pass/fail criteria

Specify the specific criteria for this design to be used to determine whether the feature or feature combination has passed or failed.

5.6 Test Case DM-PROD-SUBSYST-SCOPE-XX-YY

5.6.1 Requirements

Specify the requirements that fulfill the test cases, comma-separated and ending with '.'
e.g. CU3-IDT-XM-FUN-30,CU3-IDT-ASD-FUN-20.
Note that the format is essential for the script that creates the traceability matrices works fine.

5.6.2 Test items

Identify and briefly describe the items and features to be exercised by this test case.
For each item, consider supplying references to the following test item documentation: requirements specification, design specification, user guide, operations guide, installation guide, etc.

5.6.3 Input specification

Specify each input required to execute the test case. Some of the inputs will be specified by value (with tolerances where appropriate), while others will be specified by name.

Identify all appropriate databases, files, terminal messages, memory resident areas, and values passed by the operating system.
Specify all required relationships between inputs (e.g. timing).

5.6.4 Output specification

Specify all the outputs and features (e.g. response time) required of the test items.

5.6.5 Environmental needs

5.6.5.1 Hardware   Specify the characteristics and configurations of the hardware required to execute this test case.

5.6.5.2 Software   Specify the system and application software required to execute this test case. This may include system software such as operating systems, compilers, simulators, and test tools.

5.6.5.3 Other   Any other special requirements such as unique facility needs or specially trained personnel.

5.6.6 Intercase dependencies

List the identifiers of the test cases that must be executed prior to this test case. Summarize the nature of the dependencies.

5.6.7 Procedure

If the procedure is shared by various test cases it would be recommended to separate the procedure description from the test case definition. Thus, only a reference to the procedure identification shall be provided. The complete description of the test procedure shall be given otherwise.

Describe any special constraints on the test procedures that execute this test case. These constraints may involve special set up, operator intervention, output determination procedures, and special wrap up.

6 Test Procedure Specification

This section could be removed if the test cases specification describe their specific procedure, i.e. the procedure is included in the test case definition.

6.1 Introduction

The purpose is to specify the steps for executing a set of test cases or, more generally, the steps used to analyze a software in order to evaluate a set of features.
For every test procedure:

6.2 [PROCEDURE IDENTIFIER]

6.2.1 Purpose

Describe the purpose of this procedure. Provide the reference of the test cases that are executed by the procedure.

6.2.2 Special requirements

Identify any special requirements that are necessary for the execution of this procedure. These may include prerequisite procedures, special skills requirements and special environmental requirements.

6.2.3 Procedure steps

Describe every step of each procedure execution. Include the following steps as applicable:

6.2.3.1 Log Describe any special methods or format for logging the results of test execution, the incidents observed, and any other events pertinent to the test.

6.2.3.2 Set up Describe the sequence of actions necessary to set up the procedure execution.

6.2.3.3 Start Describe the actions necessary to begin the procedure execution.

6.2.3.4 Proceed Describe the actions necessary during the procedure execution.

6.2.3.5 Measure   Describe how the test measurements is made.

6.2.3.6 Shut down   Describe the action necessary to suspend testing when interruption is forced by unscheduled events.

6.2.3.7 Restart   Identify any procedural restart points and describe the actions necessary to restart the procedure at each of these points.

6.2.3.8 Wrap up   Describe the actions necessary to terminate testing.

6.2.3.9 Contingencies   Describe the actions necessary to deal with anomalous events that may occur during execution.

A TRACEABILITY

The backward and forward traceability (i.e. requirements trace to a test and a test to a requirement, respectively) shall be provided in order to ascertain the correlations between the test cases and the requirements they fulfill.

They can be created using the ant target: *ststrace*

ant -Dsts="abs path for STS tex" ststrace

It is necessary that DOCCOMMON/scripts directory is in the PATH. The target must be run under the directory where the SRS is located.

e.g.

ant -Dsts="../STS/GAIA-C3-SP-ESAC-RG-012.tex" ststrace

This target calls:

MakeReqList.rb to create the Requirements.csv and,

system_tests_traceability.rb -d Requirements.csv -s (abs path for STS tex)

A.1 Forward Traceability Matrix

Call the *SystemTestReqTable.tex* created

A.2 Backward Traceability Matrix

Call the *SystemTestReqTableInvers.tex* created

A.3 Other verification methods

According the SVTP [RG-004], the software verification and validation in DPAC employs reviews, inspections, analysis and testing techniques to determine whether the software system comply with requirements.

This section provides the traceability for those requirements that are not completely verified by testing. The information provided is:

- SRS Requirement

- Verification method (following the VV Strategy of RG-004): Inspection (I), Review of the design (R), Analysis (A)

- The requirement is completely verified with the stated method, or only partially (need of testing)

SRS Requirement	Verification Method	Partially/Completely Verified