

**Software Engineering**

**Quality Plan**

# Contents

1.	Purpose .....	1
2.	Quality Plan Management .....	1
3.	Organisation and Responsibilities .....	1
4.	Life Cycle Management .....	3
5.	Documentation .....	4
6.	Standards, Practices and Codes of Practice .....	6
7.	Audits, Reviews, Inspections and Walkthroughs .....	6
8.	Software Configuration Management .....	7
9.	Problem Reporting and Corrective Action .....	8
10.	Code Control .....	8
11.	Media Control .....	9
12.	Supplier Control .....	9
13.	Quality Records .....	9
14.	Test Management .....	9
15.	Maintenance Management .....	9
16.	Bibliography .....	9

## 1. Purpose

The purpose of this Quality Plan is to set down the procedures and activities that are to be carried out during the development of the Lunar Soil Sampler Arm software project to ensure that the software is developed to the required standards for the customer.

The purpose of the project is to produce software that will control the Lunar Soil Sampler Arm of the Lunar Module as described in the Requirements Specification.

The software that is to be developed can be divided into the following processes:

- i) The arm selects the soil container.
- ii) The arm places the soil container in the sample collection digger.
- iii) The arm remove soil container from digger on completion of sample.
- iv) The soil sample container is placed in secure docking mount.

## 2. Quality Plan Management

The Project Manager is responsible for the preparation, implementation and maintenance of this Quality Plan.

This Quality Plan will be subject to an initial review by the Quality Assurance Manager and then monthly reviews by the Project Manager to assure continued relevance and completeness of each of the activities / procedures described herein. The quality plan is subject to change control and any changes must be approved by the Quality Assurance Manager.

This plan is produced in parallel with the Project Plan which contains the schedule of the production of each of the deliverables and budgetary information. It must be read in conjunction with the User Statement of Requirements and Requirements Specification where the relevant Quality Characteristics are defined.

## 3. Organisation and Responsibilities

The developer's project team will consist of 4 members of staff:

- Project Manager who will work 50% on this project.
- A Consultant who will work 50% on this project.
- A Senior Software Engineer who will be full time.
- Two Junior Software Engineers, full time.
- A Quality Engineer seconded from the Quality Assurance department who will work 30%

on the project.

In addition to the above, assistance will be required from senior software engineers outside this project participate in the peer review of the system design.

All staff are responsible for the quality of their own work.

The individual responsibilities are as follows:

### **Project Manager**

The Project Manager is responsible for:

- The production and maintenance of this quality plan.
- Creation and monitoring of the project plan.
- The acquisition of resource necessary to carry out the project plan.
- The organisation of the final acceptance testing of the system prior to handover to the customer.
- Administrative supervision of all the software engineering staff on the project.

### **Consultant**

The Consultant is responsible for:

- The accurate documentation of the customer requirements in the form of a Requirements Specification.
- On the successful completion of this document, retaining an involvement in the project to provide a technical interface between the customer and the development team on the interpretation of the Requirements.
- The production of the User Manual.

### **Senior Software Engineer**

The Senior Software Engineer is responsible for:

- The production of the Acceptance test plan.
- The High Level Design.
- The Integration Plan.
- The Integration of the system and system testing to ensure that the system is prepared for the acceptance tests.
- The technical supervision of the Junior Software Engineers through their allocated tasks.

### **Junior Software Engineers**

The Junior Software Engineers are responsible for:

- The detailed design.
- Module test specification.
- Coding and module testing as directed by the Senior Software Engineer and the Project Manager.

### **The Quality Engineer**

The Quality Engineer is responsible for:

- The overall quality fo the system and for ensuring that the members of the project team take their quality responsibilities seriously.
- The Configuration Management of the System.

## **4. Life Cycle Management**

The project will adopt the ‘V’ life cycle model as recommended in the STARTS Guide and the STARTS Purchasers’ Handbook.

The items delivered to the customer will be:

- The Requirements Specification.
- The User Manual.
- A copy of the Acceptance Test specification and their data in a form suitable for regression testing.
- A copy of the files containing the source code for the system and the design diagrams form which the source code was derived.
- The object code of the software.

In addition to the above, the customer will have right of access for audit purposes tot he following project outputs:

- The Integration Plan.
- The module and system test specifications.

The notations and methods used for the production of each of the deliverable will be as follows:

### **The Requirements Specification**

The Requirements Specification will be formatted to comply with the structure defined in

Edition 2 of the STARTS Purchasers Handbook, written in English supplemented with diagrams, photographs and dataflow diagrams showing the operational activities of the Lunar Soil Sampler Arm.

### **The User Manual**

The User Manual will be written in English supplemented with diagrams and photographs where necessary.

### **The Design and Coding**

The design will be undertaken using the Yourdon Structured Method and the modules will be developed and coded in a C environment.

### **Integration and Testing**

The system will be tested and integrated incrementally based around the four major parts of the system.

## **5. Documentation**

All documents are prepared, issued and maintained using the Version control procedures listed in section 8 below.

All documents are subject to review as outlined in section 7 below.

After each document is issued, a copy is held in a fire-proof safe on the developer's premises.

The documentation that is to be prepared during the project is as follows:

### **The User Statement of Requirements**

This is prepared by the customer and is provided as a starting document for the discussion between the Customer and the Consultant.

### **The Requirements Specification**

This is prepared as a result of a full investigation and the discussions held between the Customer and the Consultant, and together with the User Statement of Requirements forms a complete specification of the system to be developed.

### **The Quality Plan (this document)**

This Quality Plan defines the procedures that are to be used during the development of the

system. It can only be amended by agreement with the QA Manager.

## **The Project Plan**

The Project Plan defines:

- The available resources.
- The timescales and scheduling of resources.
- Budgetary information outlining the costs of the resources (both personnel and equipment).

It can only be amended by agreement with the QA Manager and, where User deliverables are affected, the users must be notified and the approval obtained.

## **Acceptance Test Plan**

The Acceptance Test Plan outlines:

- The various test that will need to be carried out.
- How they are to be carried out, to demonstrate the system to the customer prior to handover.
- Defines the authorities and responsibilities of the development and customer staff at this point of the project.

## **The High Level Design Documentation**

The High Level Design Documentation defines the system in overview terms and describes the major functions and data flows about the system. It also include a data dictionary of all data and control interfaces, which are expanded in the detailed design documentation. In addition it contains:

- Data flow diagrams expanding on the information supplied above.
- Process specifications for all the processes given in the data flow diagrams.
- The most detailed DFD incorporating control flow.

## **The Detailed Design Documentation**

The Detailed Design Documentation for each part of the system consists of the following:

- A state transition diagram showing the time ordering of the processes.
- A state transition table, expanding on the information supplied in the diagram.
- Program and module structure charts.
- Logical Data Structure diagrams and accompanying Entity Life Histories.
- Textural description of the processes.

## **Module Test Specification**

A Module Test Specification is to be produced for every module of the system (where a module is the smallest identifiable unit of work that exists). Each module test specification will cover the following testing techniques:

- White box testing
- Black box testing

The Test Effectiveness Ratios (TER) must at least equal the following values:

- TER 1 (percentage of statements exercised) - 100%
- TER 2 (percentage of branches exercised in a harness environment) - 100%
- TER 3 (percentage of paths exercised) - 85% of the theoretical maximum.

## **6. Standards, Practices and Codes of Practice**

The company developing this system has a certificate of conformance to ISO 9001 2000, therefore in addition to those procedures outlined in section 7, the company may at any time be visited by the Standards Authorities who may wish to see evidence that any or all of the defined procedures have been carried out.

## **7. Audits, Reviews, Inspections and Walkthroughs**

The Requirements Specification and the User Manual will be subject to a Fagan Inspection.

The High Level Design Documentation, the acceptance test plan, and the integration plan will be subject to a peer group review which will require assistance from development personnel outside this project. The detailed design and module test specification will be subjected to a review by at least one more senior person than the producer of the document.

All module and integration testing will be witnessed and signed off by a more senior person than the author.

At any time during the development of the system, an audit may be carried out (either by the company's own Quality Assurance department or the customers Quality Assurance representative) to show adherence to required standards and procedures laid down in this Quality Plan. No notice or prior warning is necessary to perform an audit. All internal audit reports will be sent to the Deputy Managing Director and copied to the Project Manager.

## 8. Software Configuration Management

### Software Configuration Management Responsibilities

The Quality Engineer is responsible for ensuring that the system is permanently kept under a rigorous configuration control system.

### Configuration Identification

Each subsystem will be given an official name at design time and these names will be kept in the configuration management database. The format of the name will be:

<project name>.<document name> for documents

<project name>.<subsystem name>.<module name>

### Configuration Item Status

A module or subsystem can exist in one of three states; under development, registered or bonded. Registered means that a module has passed its module tests or a subsystem its subsystem tests. Bonded means that the module or subsystem is part of a successful integration.

### Change Control

A Log Book is kept detailing all the changes that occur on registered or bonded items. This Log Book contains the following information:

- Number of the changes as they occur.
- Source of change request.
- Reason for change request.
- Module status (registered or bonded - see authorisation requirement below).
- Date change requested.
- Any other parts of the system affected.
- Date that change is implemented.

The authority for any changes depends on the state of the deliverable under development as follows:

- Under Development, freely changeable by the author.
- Registered, Quality Engineer must authorise changes.
- Bonded, Project Manager must authorise changes.

## **Version Control**

Each module subsystem and system is given an issue number and version number, e.g. 2.3 is issue 2 version 3.

The version number is used by the developer / owner of the module or document and increased every time a modification is made. The initial issue number is zero.

When a module or document is “registered” the issue number is incremented and the version number reset to zero.

When a system is released to the customer, a release number is assigned and each released system consists of modules of various issue number s(note - all the version numbers should be zero on a released system).

The configuration management database contains a record of the names and issue / version numbers of all the components of all attempted integrations, including failed ones.

## **9. Problem Reporting and Corrective Action**

A log of all post installation faults will be kept and one software engineer will be assigned to the project to take responsibility for the incorporation of all corrections to problems as and when they occur.

The configuration, change and version control procedures are as defined in section 8.

## **10. Code Control**

Whilst the code modules are under development, they are resident only on the individuals network file area in a directory specific to the project. Once the module is registered, it is transferred to a temporary directory on a shared drive on the network. When transferred, the Quality Engineer transfers the code to a read-only master directory and deletes the code from the temporary directory.

Changes in the code will also be reflected in the ‘Source Safe’ software for automatic change tracking whilst on individual PCs.

All project documentation and master directories are backed up on to CD-ROM on Mondays, Wednesdays and Fridays. These CD-ROMs are kept in a fire-proof safe on the developers premises. Following the warranty period, it is the customers responsibility to ensure the security of the source code via an escrow or equivalent facility.

## **11. Media Control**

As outlined in sections 5 and 10, all issued documents and back up CD-ROMs are stored in a fire-proof safe on the developers premises.

## **12. Supplier Control**

This is not relevant to this project, as no software is being purchased.

## **13. Quality Records**

The hard-copy Quality Records for this project will be stored in an allocated filing cabinet, electronic Quality Records will be stored in a shared network directory. Hard-copy records are to be stored in chronological order. On completion of the project warranty period all the quality records will be held in an archive (this will include electronic copies transferred to CD-ROM) for a period of 5 years.

## **14. Test Management**

The test plan, test strategy, test specification, test procedures, and test records are to be defined during the production of the Requirements Specification.

## **15. Maintenance Management**

No specific activities are currently scheduled for the maintenance tasks of the system.

## **16. Bibliography**

The STARTS Guide, 2<sup>nd</sup> Edition, NCC 1987.

The STARTS Purchasers Handbook, 2<sup>nd</sup> Edition, "Procuring Software Based Systems", NCC 1989.