User Requirements Document (URD)

Introduction

This document specifies the content and scope of the User Requirements Document (URD). The purpose of the document is twofold:

■	demonstrates to interested parties that the project is well controlled and is destined for financial and technical success
■	enables effective project control, successful implementation against well defined parameters and user requirements, to provide top level documentation of the system, facilitate future enhancement and maintenance

The URD is the highest level expression of the required functionality of the system. It forms the basis against which success of the system is measured, and against which compliance can be objectively tested. All lower level documentation must contain a compliance matrix, so that design and testing documents can be shown by reference to URD statements to satisfy the needs and objectives of the system, in terms of function, schedule, budget, quality, resources, testability, extensibility and maintainability. This need to be able to trace compliance of requirements implies that requirements carry a unique numeric reference. Requirements must also be given priorities, and it is recommended that they be labelled 'mandatory', 'preferred' or 'optional'.

The URD should express the user's view of what is required, rather than his view of the implementation process ('what', not 'how'). For example, a URD might specify a storage requirement for archiving of a conference; this should be specified in terms of maximum duration of archived conference, rather that the actual computer memory/disk requirement, which might be the implementation consequence of meeting the user's need.

Other documents which may be derived from a URD, according to project size and complexity, include:

■	Project Plan Document (PPD), a set of guidelines, rules, procedures and philosophies for the project management of the entire project lifecycle
■	**System Response Document (SRD)**, a response to the URD, describing how the URD requirements will be met in high-level computer terms
■	Architectural Design Document (ADD), specifying the overall system design, identifying data structures and functional subsystems
■	Detailed Design Documents (DDD), a suite of documents, specifying the detailed design of subsystems and individual programs, often down to symbolic representation of the coding using a chosen method
■	System Test Plan (STP), specifying testing philosophy and mechanisms
■	Individual test scripts and results
■	System Users' Manual (SUM), a reference document for users
■	Quality Control Plan (QCP), describing how product quality will be assured

The URD shall contain the following sections as a minimum.

Specification

1	**Assumptions and Prerequisites**
	Describes assumptions, and prerequisite conditions, for the user's requirements to be met. Assumptions are intended to limit the design of the system, so that no attempt is made to 'over-design' and cater for situations, which are extremely unlikely to, or cannot, occur. Any over-design of a system has very significant impacts on cost, schedule, quality and maintainability, since it implies increased, unnecessary complexity. Prerequisites are mandatory conditions, which must exist for the project to be viable.
2	**Functional Description and Performance Requirements**
	Describes the individual functions, which the system must support, as experienced by the end user. The functions must be detailed, discrete and separate, and as non-interdependent as possible. System performance requirements (e.g. throughput, response times) are specified here.
3	**Resources and Schedule Requirements**
	Specifies the maximum resources that the development and operation of the system may consume, in terms of those resources under the user's control. These might include, for example, maximum staffing levels. Schedule requirements are also included here, as are budgetary and other constraints.
4	**Data Requirements**
	Specifies the requirements on system data handling and storage. Such details as the type of data to be input, processed and output are included, as are volumes of such data, sources of it, and any ancillary data to be generated and/or stored, such as historical transaction to allow replay of scenarios or regression to previous states. Volumes of data must be specified in user requirement terms, rather than computer terms (e.g. number of days of storage or number of transaction, rather than in Megabytes).
5	**Interface Requirements**
	This section specifies interfaces to other systems. At URD level, this means the interfaces that the overall system must handle, including user terminals and electronic interfaces to other computer systems (e.g. protocols, e-mails, etc.).
6	**Testing Requirements**
	This section specifies the requirements placed on system testing, including information on traceability of tests back to requirements. Also included are requirements on testing under stress conditions (maximum loading, or even loading exceeding maximum, as defined in previous sections of this document) and regression testing. Which means re-testing to ensure that no unexpected effects have been introduced as a result of a modification or enhancement, both in the modified modules and in the rest of the system (since no system is absolutely modular, and some cross-correlation always exists between components)