



JISC Project Quality Plan Template

This document defines the quality expectations the project must achieve and how they will be met.

1. Quality Expectations

The JISC programme manager completes this section defining the standards and level of quality expected to be achieved by the project.

The project will deliver the eLearning Tool(s) as specified in their proposal and refined in the JISC project plan document in line with following standards/guidelines:

- JISC (draft) Open Source Policy May 2004
- JISC (draft) Software Quality Assurance August 2004
- JISC Project Management Guidelines December 2003
- Release versions of development and final code are to placed with <http://sourceforge.net/>
- CETIS project page be maintained to communicate development progress and mapping of software to the ELF (eLearning Framework). <http://www.cetis.ac.uk/>
- Software should meet the high level functional specification as specified in the project plan.
- Software should be robust, maintainable and extendable (see JISC (draft) Software Quality Assurance August 2004).

Tolerances

- Cost – project must be completed within agreed grant.
- Time – project must be completed by 31st March 2005.
- Scope – given the short time scale of the project the scope of the deliverable (i.e. eLearning Tool(s)) may be narrowed to ensure completion on time and to budget. Any changes to scope must be agreed with the programme manager and documented via the change control procedure.
- Quality – project must adhere to the standards as defined for open standards, open source and software quality

2. Acceptance Criteria

For each of the main deliverables of the project criteria for its acceptance / completion are defined.

Successful completion of an external evaluation of the projects software outputs and development process.

3. Quality Responsibilities

List of who is responsible for monitoring and ensuring quality for different aspects of the project?

All software will be reviewed and quality controlled within MyKnowledgeMap and hence Rob Arntsen will ultimately be responsible for Quality control of software.

All reports will be reviewed and quality controlled by a small team at MMU including Mark Stubbs, Robin Johnson and an independent member of the MMU Academic staff.

4. Standards and Technologies

Referenced list of standards and technologies to be used by this project.

UML 1.4, WSDL 1.0, HTML 4.01, XML 1.0, XSLT 1.0, Java 1.4.2, IMS 1.0, IMS Content Packaging (CP v1.3.1), IMS Question & Test Interoperability (QTI v2), IMS Simple Sequencing (SS v1.0) and IMS Learner Information Packaging (LIP v1.0). Also, the latest HR-XML definition. Reload v1.3. and Uportal 2.4, Relax NG 1.0. Javadoc 1.4.2 or Doxygen v1.3.9

5. Quality Control and Audit Processes

Description of the process to be used to control project quality and enable auditing.

The project will use many of the DSDM project management concepts to plan, monitor and control the project, e.g. timeboxing and re-planning after each timebox, prioritised requirements, use of a “functional verification iteration” before the “final build iteration”, etc. Timeboxes will be 2 weeks long. Each timebox will be reviewed and used as a checkpoint. After review the next timebox will be planned for. The listed outputs, review comments and next timebox plan will be submitted to the project manager fortnightly. All interim outputs produced will be posted to SourceForge at the end of the timebox. See attachment for further detail

The development team have limited experience with Java and hence don't have a set of coding standards that can be referred to. We expect to adapt the Sun coding standards - <http://java.sun.com/docs/codeconv/html/CodeConvTOC.doc.html>

XML validation will be achieved using Relax NG and a Relax NG based validation tool, e.g. jing.

Compliance with IMS standards will be achieved by reference to CETIS and projects with prior experience – at this point in time it is hard to specify how we will deal with this aspect of the project as we have very limited experience with IMS standards.

The project will make use of some concepts from Prince 2 in the management of the project, e.g. Issue and risk logs, tolerances.

The project will maintain a project website where all project management documents will be posted. Copies of all documents will be kept in a project file at MMU

6. Change Control and Configuration Management Processes

Description of the process to be used to manage change and configuration management.

The change management process will be controlled by the work package leaders, project manager and the project Board. Small changes will be controlled by work package leaders. Any changes that are deemed by the work package leader to be “significant” will be escalated to the project board for consideration. Similarly if the Project Manager identifies “significant” slippage then he may also escalate it to the board. This does not quantify the slippage in the way that a Prince 2 Tolerances might but does provide a formal mechanism based on the same sort of procedures. Informally, the project has agreed that a “significant” slippage is one that stops a work package from achieving its overall objectives. A “non-significant” slippage may modify/reduce the scope of a deliverable but the deliverable should still be able to fulfil its role in the context of the rest of the project, e.g. if investigations of the re-usability of IMS XML standards took longer than expected and so did not include all the listed specifications but still delivered recommendations about most of them, the reduced scope would still support the rest of the project and it would be classed as a “non-significant” slippage.

Configuration management will be handled at two levels. First, all developed software and associated documentation will be lodged with SourceForge and stored on MMU servers. Second, All reports, software, documentation, project management documents and paperwork used to monitor and manage the project will be logged in a spreadsheet and stored in the project file at MMU.

7. Quality Tools

List any tools to be used to help ensure quality.

MyKnowledgeMap uses a set of internal checklists for all of its projects – examples attached to project plan. A coding standard for Java will be developed as experience with Java is gained.