



JISC Project Quality Plan Template

1. Quality Expectations

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

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

3. Quality Responsibilities

Overall Project Responsibility lies with Paul Drummond (Deputy Director, Faculty of Medical Sciences Computing).

Responsibility for adherence to relevant specifications and standards lies with Paul Drummond and Tony McDonald (Assistant Director, Faculty of Medical Sciences Computing)..

Responsibility for hardware, operating systems, development servers, and network performance and security issues lies with John Snowdon (Systems Administrator, Faculty of Medical Sciences Computing)

4. Standards and Technologies

REST (REpresentational State Transfer): is not a standard, nor is it a technology as such, but refers to an architecture style of networked systems. It provides an alternative to the use of SOAP messaging in the provision of Web Services and is currently being widely debated as the architecture of the next generation of Web Services [Ref: http://www.prescod.net/rest/rest_vs_soap_overview/]

XML (Extensible Markup Language, Version 1.0): a subset of SGML that is designed to allow generic SGML to be served, received, and processed over the Web in the same way as HTML. XML is designed for ease of implementation and for interoperability with both SGML and HTML. [Ref: <http://www.w3.org/TR/2004/REC-xml-20040204/>]

IMS LIP (Learner Information Packaging, Version 1.0): is a specification designed to address the interoperability of internet-based Learner Information systems with other systems that support the Internet learning environment. The intent of the specification is to define a set of packages that can be used to import data into and extract data from an IMS compliant Learner Information server. [Ref: <http://www.imsglobal.org/profiles/index.cfm>]

WSDL (Web Services Description Language, Version 2.0): an XML language for describing Web services. WSDL is used to describe Web services based on an abstract model of what the service offers. [Ref: <http://www.w3.org/TR/2004/WD-wsdl20-20040803/>]

CSS2 (Cascading Style Sheets, Version 2.0): a language that allows web authors and users to attach style (e.g., fonts, spacing, and aural cues) to structured documents (e.g., HTML documents and XML applications). [Ref: <http://www.w3.org/TR/1998/REC-CSS2-19980512/>]

MySQL (Version 3.23.58): an open source SQL compliant database [Ref: <http://www.mysql.com/>]

ZOPE (Version 2.6): an object-oriented, open source software development platform [Ref: <http://www.zope.com/>]

BlackBoard (Version 6.1): a staging server which provides part of the Institution's implementation of the BlackBoard VLE will be used to test integration with this environment. [Ref: <http://www.blackboard.com/>]

5. Quality Control and Audit Processes

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

Project Management: The project will follow a rapid prototyping approach to the development of the RESTian ePortfolio. The team will meet regularly (weekly) and a project log and risk log will be maintained detailing requirements, issues and progress.

Software User Testing: The ePortfolio is currently being evaluated by user groups as part of three other projects (evaluation groups are drawn from Undergraduate Medicine, Undergraduate Dentistry, Postgraduate Dentistry, and Contract Research Staff). The generic ePortfolio will be kept up-to-date with other versions of the ePortfolio to reflect the results of this user testing and evaluation and incorporate their feedback.

Data Transfer Testing: IMS LIP conformant outputs from the software will be evaluated against the schemas published on the IMS website. However this will only evaluate technical compliance to the specification and not the ability to transfer data between systems which may embrace differing workflow management processes. To overcome this, small data transfer trials will be organised with other projects, and early adopters of the ePET using IMS LIP conformant I/O processes.

Budget: The project budget will be managed via the University of Newcastle finance system (SAP) reports can be generated automatically showing spending against agreed budget heads in the JISC project template.

6. Change Control and Configuration Management Processes

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

Software Development and Release: The project makes use of staging servers on which all software development takes place. Software is not migrated to the distribution server until it has been tested both in terms of desired functionality and conformance with the standards and technologies described in Section 4 of this document.

Development Environment: The software is developed in ZOPE (Z Object Publishing Environment) which is a multi-author development system with built in version management, multi-author change tracking and rollback functionality.

Backup and Recovery: All databases, ePortfolio scripts, and static HTML content are backed up onsite at 12:00 and 18:00 and offsite to tape once nightly. Onsite backups are maintained until overwritten by the subsequent backup. Offsite backups to tape are kept for five business days. Database recovery from onsite backup can be completed in 1hr. ePortfolio website recovery from onsite backup can be completed in 3hrs.

Onsite Archival of Software: All databases, ePortfolio scripts, and static HTML content will be downloaded once per month, providing an incremental development archive.

Offsite Archival of Software: In addition to the Backup and recovery strategies described above, the software source code will be archived at all appropriate project milestones (see accompanying Workpackages document) using the JISC archival service.

7. Quality Tools

The tools listed below will be employed to help the software development team construct a robust high quality software product however, the tools described are, in the main, automated parsing or checking devices and used on their own will not achieve the desired level of quality. It is the overarching attitude of the development team which will be instrumental in achieving a product which conforms to the expectations laid out in sections 1 & 2 of this document. Documentation of conformance will be posted on the project website as it is acquired to enable the programme's software evaluation team to monitor conformance progress.

IMS LIP Schemas: IMS LIP conformant outputs from the software will be evaluated against the schemas published on the IMS website. In addition small data transfer trials will be organised with other projects using IMS LIP conformant I/O processes. [Ref: <http://www.imsglobal.org/profiles/index.cfm>]

WS-I testing tools: will make use of toolkits designed to help developers determine whether their Web Services are conformant with Profile Guidelines. [Ref: <http://www.ws-i.org/implementation.aspx>]

W3C CSS Validation Service: will be used to check Cascading Style Sheets (CSS) in (X)HTML documents or as standalone files for conformance to W3C recommendations. [Ref: <http://jigsaw.w3.org/css-validator/>]

Bobby: will be used to test web pages to identify accessibility issues using the guidelines established by the World Wide Web Consortium's (W3C) Web Access Initiative (WAI), as well as Section 508 guidelines from the Architectural and Transportation Barriers Compliance Board (Access Board) of the U.S. Federal Government. [Ref: <http://bobby.watchfire.com/bobby/html/en/index.jsp>]