Application software maintenance standard

The purpose of this standard is to establish a common definition of business application system maintenance, the different types of system maintenance and how they are handled at a high-level.

This standard applies to:

a) In-house developed and managed applications (custom software) including solutions developed on *PaaS* platforms
b) Any other business application where some aspect of the technology maintenance is the responsibility of UNSW

Version	Approved by	Approval date	Effective date	Next full review
1.0	Christine Burns	11/12/24	01/01/25	01/02/27
Accountabilities				
Responsible Officer		Sally Anderson, Director, StARS IT		
Contact Officer		Sally Anderson, Director, StARS IT		
Supporting Information				
Relevant University-wide Policy		n/a		
Supporting Documents		Service Management Processes		
Superseded Documents		StARS Software Maintenance definition, January 2023		

1. Overview

Software maintenance is defined by the IEEE as:

"Modification of a software product after delivery to correct faults, to improve performance or other attributes, or to adapt the product to a modified environment."

There are four types of software maintenance required to ensure that business systems are appropriately managed throughout their lifecycle:

2. Corrective maintenance

Corrective software maintenance addresses the errors and faults within software applications that could impact various parts of your software, including the design, logic, and code. The role of corrective maintenance is to perform changes to the system to align it with its original requirements and specifications.

Corrective maintenance includes:

- Resolution of Incident tickets (CASD) in line with Service Levels
- Identifying and resolving errors/bugs
- Root cause analysis

Incidents and Problems will be managed in the ITSM tool (CASD at time of writing) using standard ITSM/SMO processes.

Errors or bugs identified outside of incidents will be logged and managed via JIRA Software.

3. Adaptive maintenance

Adaptive software maintenance becomes important when the environment of your software changes. This is necessary when an application needs to run on a new platform or operating system, or it needs to interface with new software. Adaptive maintenance also responds to organizational policies or compliance changes.

Adaptive maintenance includes:

- Redesign and development of small portions of an existing software application in response to compliance or policy changes

- Redesign and development of an existing software application in response to changes in the technical environment

Adaptive maintenance may be generated from a range of source including business requests or IT projects and will be managed via JIRA Software.

4. Preventative maintenance

Maintenance performed to prevent future problems with the software. The focus of the type of maintenance is to prevent the deterioration of your software as it continues to adapt and change. These services can include optimizing code and updating documentation as needed. As a result, software will be more stable, understandable, and maintainable.

Preventative maintenance includes:

- Ensuring platforms are kept patched in line with UNSW IT approved policy including acting on urgent patch releases from vendors
- Ensuring required certificates are kept up to date
- Periodic review of application code and environments
- Identification of repeating faults
- Risk assessment and reporting on applications and platforms

Preventative maintenance will be generated through both pro-active practices and external sources and will be managed via JIRA backlog.

5. Perfective maintenance

Maintenance to improve performance or maintainability. The goal is to ensure that the system remains current and that users are satisfied with the experience because of the added value perfective maintenance contributes. Adding features that can enhance user experience and removing features that are not effective and functional. Fixing issuing pertaining to the performance of your system is a frequent manifestation of perfective maintenance.

Perfective maintenance includes:

- Development of small portions of an existing software application in response to customer requests for minor enhancements
- Work can typically be easily carried out with low effort by the DevOps team with only standard requests made of other IT teams

Perfective maintenance will mostly be generated by the business and managed via JIRA backlog.

All changes to systems will adhere to UNSW IT procedures including but not limited to:

- Management through the UNSW IT Change Advisory Board (CAB)
- Testing and release management
- Updating system and technical documentation

Appendix 1: Types of work examples

	IT generated	Customer generated
Operational maintenance (BAU Funded)	• Code bug or error identified by team (corrective) • Security fix identified through code scan (preventative) • Deployment of a patch (preventative) • Code change to make better use of new technology (adaptive)	• Incident logged in CASD (corrective) • Change to system required as result of business compliance or policy change (adaptive) • Minor enhancement request such as: addition or removal of fields, small feature request (perfective)
Strategic **(Project-funded or chargeback to customer)**	• Major upgrade to system required which is complex and requires extensive resources and time	• Addition of entirely new business capability or significant functionality • Major system changes or merger of multiple systems

Revision History				
Version	Approved by	Approval date	Effective date	Modifications
1.0	Christine Burns	11/12/24	01/01/25	Version 1 created from superseded document