Sample pages of the

TEMPLATE FOR A SOFTWARE MAINTENANCE PLAN

Introduction

Background

The hype surrounding the Year 2000 (Y2K) software crisis identified the need for solid software maintenance policies and practices. Many organizations were forced to deal with significant changes to their software inventory and expended considerable funds accomplishing the needed tasks. Software systems are developed and evolve; they are not static. The last phase of the software engineering lifecycle, operation and maintenance, often takes the majority of life cycle funds. It is therefore prudent to possess software maintenance plans and procedures to contain life cycle costs, and to operate an efficient organization.

Software Engineering Process Technology (SEPT) in conjunction with the noted Software Maintenance expert Thomas Pigoski has developed this template for a Software Maintenance Plan to aid the software engineer in implementing software maintenance requirements. This template is easy to use, self-explanatory, and does not require expensive training or extensive experience.

About Software Maintenance

Software maintenance is the totality of activities required to provide cost-effective support to a software system. Activities are performed during the pre-delivery stage as well as the post-delivery stage. Pre-delivery activities include planning for post-delivery operations, supportability, and logistics determination. Post-delivery activities include software modification, training, and operating a help desk.

How to use this Document

This document is designed to aid a person with limited knowledge of software maintenance requirements and methods to plan for software maintenance of a project or system. This template may be applied to manual or automated (computer processes) methods and can be easily implemented by one or more persons. It is applicable to small, highly critical 10-line software programs and to programs over 1 million lines of code.

Organizational Requirements

Maintenance is performed by the developer, a separate maintainer, or by a third-party organization. It is important that the organization responsible for maintenance be identified in writing with full responsibilities. The Maintenance Plan accomplishes this. The maintainer should develop the Maintenance Plan as well as the supporting procedures. Since software maintenance activities invoke the use of organizational resources, it is recommended that the highest level of management in the organization approves of this undertaking and approves the final version of the plan and the procedures. Other functions that should also review and approve this plan include Software Quality Assurance, Software Engineering, Software Testing, Project Management (when applicable), the organization’s Software Configuration Management Function (when applicable), and the customer (when applicable).

This template's illustrative text is designed for use as is, by stripping the tutorial notations underlined in the text. The text can also be modified for requirements and guidance to meet organizational needs, and unique environments. It is not a requirement to use the paragraph numbers contained in this document and additional comments are encouraged whenever appropriate to fully comply with an organizations' specific requirements. This template is applicable to all types of software from information technology, commercial, scientific, and other non-business applications (such as creating a complex web site). The user of this template should spell out all of the issues that are prevailing regarding the need for software maintenance prior to tailoring the template to ascertain that all such organizational issues are addressed.

Reference Software Configuration Management Standards

International Standards

- ISO/IEC 12207 – 1995. Software Engineering - Software Life Cycle Processes.
- ISO/IEC 14764 – 1999. Software Engineering -Software Maintenance

USA Standards – (International application)

- IEEE/EIA 12207.0 – 1996, 12207.1, 12207.2,
- IEEE 1219 - 1998. Software Maintenance

Defense Standards (USA)

- J-STD-16-1998 (30 Sept 95), Standard for Information Technology, Software Life Cycle Processes.

Standards and Specifications may be procured through SEPT at www.12207.com.

Reference Books

Practical Software Maintenance: Best Practices for Managing Your Software Investment, Thomas M. Pigoski, John Wiley and Sons, New York, 1997. To order this book **click** on to www.12207.com.

Guide to Software Engineering Standards and Specifications, S. Magee and L Tripp. Artech House, 1997. To order this book **click** on to www.12207.com.

Warranties and Liability

SEPT makes no warranties, implied or stated with respect to this template and it is provided on an "*as is basis*". SEPT will have no liability for any indirect, incidental, special or consequential damages or any loss of revenue or profits arising under, or with respect to the use of this document.

5.0 General Requirements

This section should describe the policies and responsibilities of the program/project team as it plans for software maintenance. Policies and responsibilities are usually spelled out in an organization policy/directives manual and may be referenced here. It is highly recommended, however, that such doctrine be summarized in the plan as illustrated below.

5.1 Introduction

This section describes the system to be supported and identifies the initial status of the system. Complete identification of the system should include formal and common names, nomenclature, identification number and system abbreviations. Subsystems and interfacing systems should be identified.

This plan describes the processes and procedures necessary to provide software maintenance for the XYZ system. System XYZ, Version 1.0 is being developed by the Software Development of Company ABC and the Software Maintenance Department of Company ABC will perform all software maintenance functions. NOTE: If the developer will perform maintenance, the following would be used. The Software Development Department of Company ABC is developing SYSTEM XYZ Version 1.0. Software maintenance will also be performed by the Software Development Department. NOTE: If maintenance were outsourced to another company, the following would be used. System XYZ, Version 1.0 is being developed by Company ABC and will transition to Company DEF for software maintenance support. System XYZ contains subsystems COLLECTION, PROCESSING, and REPORTING. System XYZ interfaces with systems QRS and STU. This plan details the activities required and specifies the various responsibilities in order to provide software maintenance for System XYZ.

5.1.1. System. Describe the mission of the system to include mission need and employment. Identify interoperability requirements. Describe system functions. Describe the system to include descriptions of: system architecture, components and interfaces; hardware and software. Use separate subparagraphs to describe each subsystem and major hardware/software component.

System XYZ provides payroll processing for a small business. It provides input to a third party payroll company called ADP Inc. The data provided must be in a format compatible with ADP Inc. requirements. System XYZ takes input from timecards, aggregates them, and formats the data to be forwarded to ADP Inc.

5.1.2. Status. Identify the initial status of the system. System XYZ is under development and replaces System XY. System XY is a semi-automated system that is not integrated into corporate operations. System XYZ will provide that integration and additional functionality.

5.1.3. Support. Describe why support is needed. System XYZ has a projected life of 3 years. During that period, corrections and enhancements will be required. Corrective maintenance will accommodate latent defects as reported by users. Enhancements, or improvements will be submitted in order to improve performance and provide additional functionality for the users. As a result, maintenance support is required.

5.1.4. Maintainer. Identify the maintainer. The maintainer for System XYZ is the Software Maintenance Department of the ABC Company. (NOTE: or the Software Maintenance Department of the DEF company if maintenance is provided by an outside source. Or the Software Development Department if there is no transition to a separate maintenance organization.)

5.1.5. Contracts. Describe any contractual protocols between customer and supplier. The Software Development Department of Company ABC has a signed Memorandum of Agreement with the Software Maintenance Department of Company ABC to provide software maintenance for system XYZ. Through a Configuration Control Board (CCB), agreement will be reached on what corrections and enhancements will be provided in the next release. Emergency support is on an hourly basis. NOTE: For outside support use the following: Company ABC has a signed Memorandum of Agreement with Company DEF to provide software maintenance for system XYZ. Through a Configuration Control Board, agreement will be reached on which corrections and enhancements will be provided in the next release. Emergency support is on an hourly basis.

5.2 Maintenance Concept

Describe the concept. The maintenance concept addresses: The scope of software maintenance; the tailoring of the post-delivery process; the designation of who will provide maintenance; and an estimate of life-cycle costs. The customer should develop it early in the development effort with help from the maintainer. Defining the scope of maintenance helps the customer determine exactly how much support the maintainer will give to the customer. Scope relates to how responsive the maintainer will be to the users.

The maintenance concept also addresses the activities of post-delivery software maintenance. Different organizations often perform different activities in the post-delivery process. An early attempt should be made to identify these organizations and to document them in the maintenance concept. In many cases, a separate maintenance organization performs the maintenance functions.

Figuring out who or which maintenance organization should perform maintenance for a particular system involves many factors. If the system only has a lifespan of two years, perhaps the developer should maintain it.

5.2.1. Concept. Responsiveness to the user community is the primary consideration in determining the scope of software maintenance. The scope of software maintenance should be tailored to satisfy operational response requirements. Scope relates to how responsive the Maintainer will be to proposed changes. For example, a full scope software maintenance concept suggests that the Maintainer will provide full support for the entire deployment phase. This includes responding to all approved software change categories (i.e., corrections and enhancements) within a reasonable period. Software maintenance concepts that limit the scope of software maintenance are referred to as "limited scope concepts." Limited scope concepts limit the support period, the support level, or both.

The Software Maintenance objective for System XYZ is to release two operational versions during each year. The Configuration Control Board will determine the target dates and the content of each release. Support for Version 1.0 will be limited to priority one (cannot operate the system) corrective actions. All other problems will be saved and included in the next release. All enhancements will be held until a scheduled release.

5.2.2. Level of support. Describe the level of support for the system. Support will be provided for 3 years and will include support for two major releases each year. All corrective and enhancements approved by the Configuration Control Board will be included in releases. Tracking of all change requests is required. A Help Desk will be maintained and technical support will be provided as needed.

5.2.3 Support period. Describe the support period from pre-delivery to post-delivery support. The maintainer will provide support during the development phase. This support will be on an on-call basis to review requirements, plans, etc. The post-delivery support period will be 3 years.

5.2.4 Tailoring the maintenance process. Tailor the maintenance process by referring to the maintainer's software maintenance process manual. Any activities and/or task in the process manual that will not be performed for System XYZ must be specifically tailored out, i.e., deleted. In this section, identify the specific sections of the process manual that will be deleted. Software Maintenance for System XYZ will be performed in accordance with Company ABC's "Organization Software Maintenance Process Manual." Or, for outside support, in accordance with Company DEF's "Software Maintenance Process Manual." As an example, System XYZ may not require sections 3.3, 3.4 and 3.5 of the Process Manual. Thus, the following might be used. Specific activities tailored out for System XYZ are items: 3.3, 3.4, and 3.5 of the "Organization Software Maintenance Process Manual."

5.3 Organization and Maintenance Activities

Once the maintenance organization, i.e., the maintainer, is identified, the specific maintenance activities are specified. General software engineering activities are performed during pre-delivery and post-delivery. The role of the user is defined as well as any interfaces with other organizations.

Modification Request

Help Desk Number	Date Received	Product	Originator

Short Name of Problem	User Priority 1 □ 2 □ 3 □ 4 □ 5 □
Description:	

MR Number	Engineer Assigned

SW Version	Test Level	Test ID	Problem Category	Priority

CSUs Affected	Documents Affected
Work Around:	

Analysis Results:		□ Corrective □ Enhancement □ Re-creatable □ Design Change □ Document Change
Software Engineer:	Date	
Software Engineer Comments:		Resource Estimate

Software Manager:	Date	□ Proceed □ Hold □ Reject
Software Manager Comments:		
Quality Assurance:	Date	□ Concur □ Hold □ Reject
Quality Assurance Comments:		

Configuration Control Board: FORM XXXX Modification Request pg 1	Date	□ Approved □ Hold □ Disapproved