Suggested Management Plan for Software Maintenance

Amna Elhasi and Bilal Aljabour

Suggested Management Plan for Software Maintenance

Amna F. Elhasi

Department of Software Engineering, University of Benghazi, Benghazi, Libya, Amna.elhasi@uob.edu.ly

Bilal A. Aljabour

Department of Software Engineering, Libyan International Medical University, Benghazi, Libya, Bilal.aljabour@limu.edu.ly

Software maintenance management has become one of the most important factors that help complex software systems to perform efficiently and sustainability. Managing and controlling changes to the software can play a vital role for the success of a software product, this paper proposes an integrated plan for software maintenance management, which may help in solving most of the maintenance problems caused by poor planning.

Keywords: Software Maintenance, software management, software maintenance management plan.

1 INTRODUCTION

In software engineering, the software development life cycle is a series of steps for preparing and developing software. Software maintenance is essential at these stages. After the software is delivered, some changes need to be made to the software for some reason. These changes may be related to software bug fixes, software upgrade, etc.

In most cases, the software maintenance budget is between 50% to 80% of the entire software budget. Despite this, software maintenance is not considered a top priority when it comes to the software development phase. According to results of a past investigation revealed that about $30 billion was spent worldwide on software maintenance[1].

Software maintenance is divided into four categories which are adaptive maintenance, perfective maintenance, corrective maintenance, and preventive maintenance, however, 75% of the maintenance efforts are used in adaptive and perfective maintenance[2][1].

Due to the ever increasing importance of maintenance in organizations, maintenance management has been extensively studied. However, it is found that there is a gap between the proposed maintenance strategies and those adopted in the industries. Muchiri et al. Found a gap in the evaluation of procedures and practices in manufacturing industries although there is much literature focusing on maintenance and improvement modeling [3].

2 BACKGROUND

A. Software Maintenance

according to IEEE, software maintenance is defined 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 and the set of activities that takes place such that software installed for operational use continues to perform as intended and fulfill its intended role in system operation".

Software maintenance includes improvements, aid to users, and related activities, It covers more than a half-life time of software development life cycle for a software product and the cost of maintaining software is more than that of its development[3][4].

also, most of the problems related to maintenance have its origin in planning and development phase of software lifecycle. Indiscipline and lack of control in software development activities leads to maintenance problems in the future[4].

In [4] they defined most demanding maintenance problems as:

a) Outdated Legacy Systems

The problem of maintaining outdated legacy systems or components is there. Thus, maintaining systems that were developed with older technologies with the newer technologies seems to be difficult.

b) Lack of Adequate Documentation

The absence of adequate documentation also becomes one of the factors that cause problems while maintaining software. A lack of documentation makes it more difficult for the maintainers to understand the system and there is a probability that the changes made to the system are left undocumented.

c) Notification About Changes

Managing and controlling changes to the software can play a vital role for the success of a software product. Whenever there is any of the modification in software, the changes should be notified by manager to different stakeholders of the components. All the stakeholders should be notified that the component has been changed. It should also be notified that what changes are made, the type of changes made, nature of changes made, and the size of changes made. It should keep all the information related to software such as log of change, log of reuse, and log of maintenance issues.

B. **Software Maintenance Management**

Maintenance management has become one of the most important factors that help complex buildings to perform effectively. As a result, software lifespan has dramatically increased especially when using recent technology.

To maintain, repair or even replace a component in a building it should be done through a process and decisions taken by maintenance management team, taking in consideration that this decision may affect a software function, which affect the indirect cost of a project through its operational phase [5].

Tamer et al. Divide the maintenance management process into two parts:

- Maintenance strategy.

- Implementation of Strategy [5].

To define a maintenance strategy, a proper definition of the maintenance objectives as an input needs to be addressed. The definition can be obtained directly from the business plan. This initial part of the maintenance management process shows the success of maintenance in an organization and determines the effectiveness of the subsequent implementation of the maintenance plans, schedules, controls, and improvements.

Effectiveness shows how well a department or function meets its goals or company needs. This would be able to minimize the maintenance indirect costs, those costs associated with production losses, and ultimately, with customer dissatisfaction. Effectiveness concentrates on the correctness of the process and whether the process produces the required result or not [5].

Figure [1] shows maintenance management model that suggested by Tamer et al., they finalized maintenance management model as consisting of eight sequential management-building elements, first three building elements discuss maintenance effectiveness while fourth and fifth elements ensure maintenance efficiency. Elements 6 and 7 are devoted to maintain asset life cycle cost assessment. Finally, element eight ensures continuous maintenance management improvement.
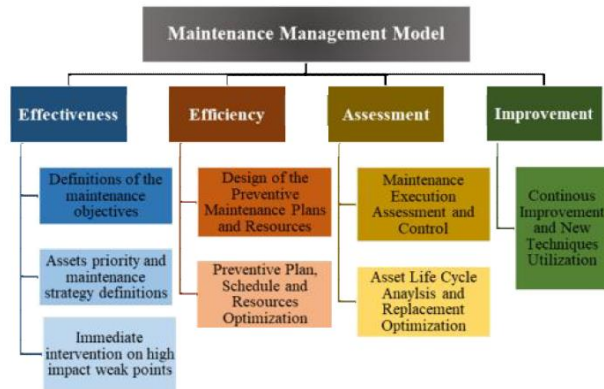
Figure 1: Maintenance management model [5].

Software maintenance is a healthy, appropriately recommended activity that meets users' needs and expectations in a dynamic environment and in [6] they identified some measures and activities that helps to keep the system young, which have been classified into measures and activities that must be taken to achieve these measures:

a. **Software performance**
   • By monitoring the memory usage.

   • Improving the quality aspects of the software product.

b. **Software failure**
   • Correct the fault and error accordingly and systematically.

   • Improve the change request and process.

c. **Business Demand-Changes in business process**
   • Services to ensure the demands in business processes are maintained.

d. **Environment change**
   • Easy maintenance for environment change.

e. **Technology change**
   • Improve the software for meeting new hardware or software demand and compatibility.

f. **Expertise**
   • Train staff for software quality practices and implementation.

   • Focus on training for staff in software maintenance and related

3   PROBLEM STATEMENT

The software maintenance is not only a matter of editing source code to fix bugs or to add new features, if the planning for maintenance is neglected the software being in a risk that may negatively affects software quality, cause system failure or at least delays the recovery after failure. In a survey published in 2020, by Saeed et al., [1] they summarized software maintenance problems into 26 problems, the most important of which are:

1). Quality of documentation. 2). Changes in software and hardware 3). Constant demand for additions and improvements 4). Maintaining programmer's skills. 5). Availability of maintenance programmers. 6). Challenge programmer time pressure 7). Insufficient user attention. 8). System failure. 9). Motivated by programmer maintenance. 10). Forecasting maintenance requirements for programs. 11). Management support. 12). Capacity system design details. 13). Budget pressures.

4   RESEARCH METHODOLOGY

In this paper, it was based on data collection and study of the problem, according to the following steps:

1.   Identify and study the types of software maintenance.

2.   Identify and study the impact of software maintenance management.

3.   Provide a plan to help software mangers and users by improving the process of software maintenance management.

4.   Iidentify tools and techniques that can be used in the maintenance stage.

5   PROPOSED MAINTENANCE PLAN

The planning and scheduling of software maintenance is at the heart of asset management and maintenance. By plan for change requests management and maintenance management, businesses can optimize the efficiency of their software assets, particularly the assets that they rely heavily on to generate revenue.

The maintenance management plan is a plan developed from the beginning of the project management plan to be included in the project charter and project contract. The proposed maintenance management plan depends on four basic stages, as shown in Figure (2). The following is the explanation of proposed plan stages:
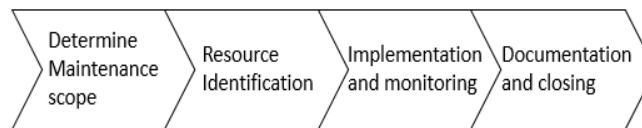

Figure 2: Proposed Software Maintenance Management Process

5.1 Determine Maintenance scope

Maintenance scope is the part of project planning that involves determining and documenting the following:

- List of specific maintenance goals.
- Tasks description.
- Deliverables.
- Costs.
- Deadlines.

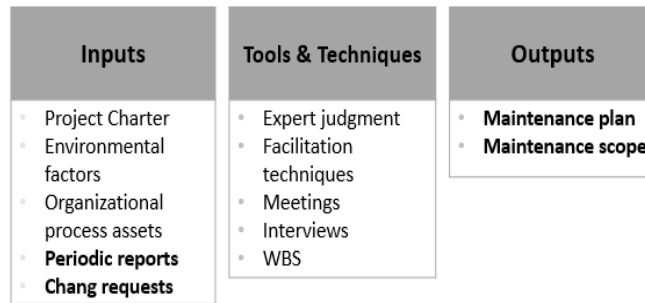Figure (3) shows the process of determining the scope of maintenance:

Figure 3: Determine Maintenance Scope Process

- **Periodic reports:** are the results of periodically software operation, the main objective of this document is to monitor the performance of the software. Periodic reports must be collected and archived in a systematic way, they will include important information, in the four types of maintenance.

- **Change requests:** written by users to request some change in the software system, includes description of defects and/or description of the required change.

Outputs:

- **Maintenance plan:** A document describes the operations to be taken during the maintenance process including cost, tasks and deadlines.
- **Maintenance Scope**: It is a document describes the requirements that must be achieved in this maintenance period and the components that must be maintained.

5.2 Resource Identification

Identification of resources is the process of identifying and estimating the material and human resources needed for this maintenance task. Figure (4) shows the process of identifying maintenance resources, and the tools and procedures used for the process:
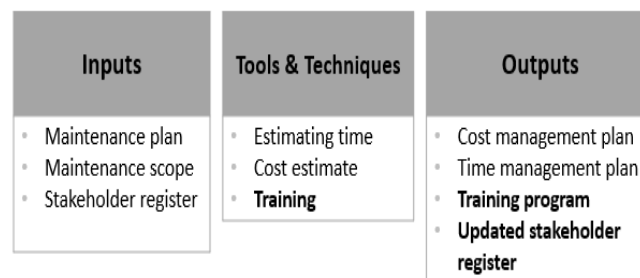

Figure 4: Resource Identification Process

- **Training**: It is the process of train new programmers to maintain the relevant software. The main objective of using this technique is to avoid a shortage of maintenance programmers in the future. Also, it increases the programmers' readiness and decrease the required maintenance time as well.

Outputs:

- **Updated Training program:** Every time members are trained for maintaining a software, the training program for this software is updated. For example if a new version of the used web framework has been launched, the programmers should updated their skill and be familiar with additions and deprecations between the old and the new versions.
- **Updated stakeholder register**: When training new members for software maintenance, the trainees and the training details are added to the stakeholder register.

5.3 Implementation and monitoring

Implementation and monitoring phase is usually the longest phase in the project management life cycle. It begins after the approval of maintenance plans and the allocation of the resources necessary for executing the task. Figure (5) shows the process of determining the scope of maintenance, and the tools used in the process:
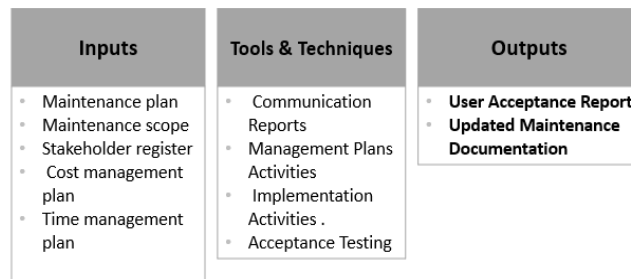
Inputs	Tools & Techniques	Outputs
Maintenance plan	Communication Reports	**User Acceptance Report**
Maintenance scope	Management Plans Activities	**Updated Maintenance Documentation**
Stakeholder register	Implementation Activities .	
Cost management plan	Acceptance Testing	
Time management plan		

Figure 5: Implementation and monitoring Process

- **Acceptante testing:** Performing acceptance testing with user and to obtain a formal acceptance sign-off, signifying that the maintenance has met the objectives and requirements. It will be more effective if the acceptance test has been designed when determining maintenance scope.

Outputs:

User Acceptance Report: A document that proves the user's assessment and satisfaction whether the deliverables meet the requirements or not. Both functional and quality attributes are considered here.

5.4 Documentation and closing

The last stage, once the defined processes are completed within all of the process groups to close maintenance process, as appropriate. In this is the stage where all deliverables are finalized and formally transferred, and all documentation is signed off, approved, and archived. The new version of the systems should be versioned, and a copy of the old version should be preserved. Figure (6) shows the process of documentation and closing.
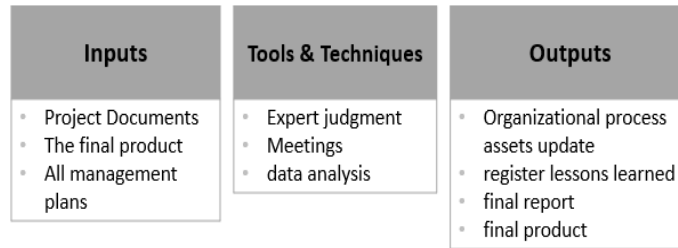
Figure 6: Documentation and Closing Process

6 DISCUSSION

The plan covered most of the problems in the maintenance process, and therefore it may help organizations to reduce effort, cost and risk by early planning for maintenance. Training programs were also included in the maintenance plan in order to reduce the unavailability of programmers, and the plan focused on documenting all maintenance operations. The plan has not been practically implemented to reach actual output to ensure the effectiveness of the plan's work.

The proposed maintenance management plan has the flexibility to help organizations and project managers to manage the maintenance in normal circumstances and when technologies and/or team members changed.

7 CONCLUSION

The proposed plan was developed according to well-known concepts and techniques in the field of software project management, and according to the guide of the project management body of knowledge PMBOK. The start of the plan depends on a change request or noticing a danger in the periodic reports, thus helping corrective and preventive maintenance to help prevent system failure. It also relied on continuous training of the team to support the improvement and the availability of programmers. The plan includes continuous reports that help inform stakeholders of changes

8 FUTURE WORK

   a) Study the effect of maintenance planning lack for a set of in-production software systems.
   b) A case study of the implementation of the plan on an in-production software system to study the applicability of the plan and the degree of its compatibility with the work environment.

REFERENCES

[1]   Effort Estimation Problems in Software Maintenance – A
       Survey " Syed Sarmad Ali ,Muhammad Shoaib Zafar Muhammad, Tallal Saeed, 2020 IEEE.

[2]   The Emergence of Agile Maintenance" Ku Saimah Ku Ibrahim, Jamaiah Yahaya, Zulkefli Mansor, Aziz Deraman, 2019 .

[3]   A mixed methods research toward a framework of a maintenance management model: A survey in Moroccan industries, Amal Naji, Mohamed EL Oumami, Otmane Bouksour and Zitouni Beidouri, 2019.

[4]   Concerns in Maintaining Reusable Software Components and the Possible Solutions ,Indian Journal of Science and Technology ,June 2017 ,Anshul Kalia1 and Sumesh Sood.

[5]   IDENTIFYING THE IMPACT OF INTEGRATING BUILDING INFORMATION MODELING WITH MAINTENANCE MANAGEMENT A LITERATURE REVIEW Tamer Ahmed Y. Mohamed Hossam, Ali Fathi Eid and Laila Khodeir April, 2019.

[6]   Software Ageing Prevention from Software Maintenance Perspective – A Review, Zuriani Hayati Abdullah, Jamaiah H. Yahaya, Zulkefli Mansor1 and Aziz Deraman, 2018.

[7]   A Guide to the Project Management Body of Knowledge (PMBOK® Guide).