SCHOOL OF TECHNOLOGY

BACHELOR OF SCIENCE IN INFORMATION AND TECHNOLOGY

FINAL YEAR PROJECT II: BIT 04205

KCAVIBES E-COMMERCE SYSTEM IMPLEMENTATION

BY

AWANZI HASSAN MUNENE (21/04609)

PRESENTED TO: Prof. PATRICK OGAO

SUBMISSION DATE: 13/03/2024

Table of Contents

1.0 Introduction

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that will work efficiently and effectively.it involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation of change over methods. Apart from planning major tasks of preparing the implementation are education and training of users.

The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out discussion made regarding the equipment and resources and the additional equipment has to be acquired to implement the new sytem.in network backup system no additional resources are needed. Implementation is the final and the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it is found to be working according to the specification. This method also offers the greatest security since the idle system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

1.1. Documentation

The main document that will be used in the implementation process is the user manual that will guide the user who is the admin of the system on how to use it

1.2. Purpose

The purpose of this implementation strategy is to ensure that:

- **i.** The final stages of the project are managed in a satisfactory manner.
- **ii.** The planned project outcomes are achieved prior to formal project closure.
- **iii.** The success of the project outputs is assed and corrective action performed if required.
- **iv.** The utilization of the projects outputs is linked to the planned project outcomes.

1.3. Implementation strategies

They include:

1.3.1. Phased changeover

This is where only certain modules of the new system are implemented over time and the existing system is phased out as functions are successfully resumed by the new system

1.3.2. Parallel changeover

A changeover method where the current system and the new system are running concurrently comparing the results from both systems to fix any problems in the new system and review the performance.

1.3.3. Direct changeover

This is where the new system replaces the old system as a whole. The system is installed and used immediately without any gradual implementation and could be disastrous if there are errors in the system.

1.3.4. Pilot changeover

Where the new system is implemented initially only for a discrete portion of the organization.

In this implementation phase a parallel changeover will be used which allow gradual analysis and comparison of the two systems running concurrently. This will enable admin of the system to learn progressively the new system and be ready for full operations.

The existing system acts a backup to the new system until both the user and the developer are entirely satisfied with the performance and integrity of the system.

Justification for parallel changeover

i. It is the preferred form of changeover strategy because it offers security to the user in that the user will always have the confidence that if the new system does not work they can rely on the old system.

ii. It becomes easy for the user to learn the new system because they work with the system and at the same time, they are working with the existing system.

iii. It becomes easy for the new system to be accepted as the user has in mind that the old system is not being faced out immediately.

iv. It helps in comparing the performance of the new system compared to the existing system.

Advantages of the parallel changeover

In a parallel changeover the new system, runs simultaneously with the old for a given period of time .of all the techniques this trend to be the most popular mainly because it carries the lowest risk. If something goes wrong at any point, the entire system can be reverted to its original state.

Disadvantages

i. It takes longer time to realize the full benefits because the full advantages of the new system do not become available until the existing is no longer in use.

ii. It encounters a lot of repetition of work.

Implementation support

They include support software materials, equipment and facilities required for the implementation of HGHMS as well as personnel requirements and training necessary for the implementation.

1.4. System maintenance

1.4.1. Corrective maintenance

Corrective change most commonly referred to as bugs is the most typical change associated with maintenance work. Corrective changes address errors and faults in your

software that could affect various areas of your software, design, logic or code. Most commonly these changes are sprung by bug reports created by users.it is important to note that sometimes problem reports submitted by users ate actually enhancement of the system not bugs.

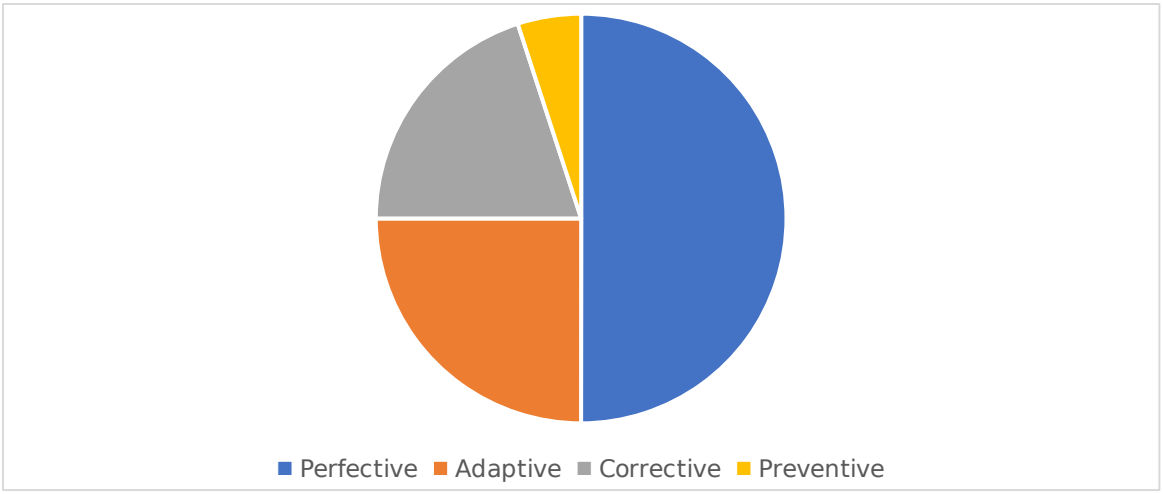
1.4.2. Adaptive maintenance

Adaptive change is triggered by changes in the environment your software lives in an adaptive change can be triggered by changes to the operating system, hardware, software dependencies and even organizational business rules and policies. These modifications to the environment can trigger changes within other parts of your software for example updating the server, and computer system can affect functionality in your software.

1.4.3. Perfective maintenance

Perfective changes refer to the evolution of requirements and features in your existing system. As your software is exposed to users, they will think of different ways to expand the system or suggest new features that they would like to see as part of the software, which in turn can become future enhancements to the system.

Perfective changes also include removing features from a system that are not effective and functional to the end goal of the system. Surprisingly 50%-55% of most maintenance work is attributed to perfective changes**.**



1.4.4. Preventive maintenance

Preventive changes refer to changes made to increase the understanding and maintainability of your software in the end. Preventive changes are focused in decreasing the deterioration of your software in the end. Restructuring, optimizing code and updating documentation are common preventive changes. Executing preventive changes reduces the amount of unpredictable effects software can have in the long term and helps it become scalable, stable, understandable and maintainable.

1.5. Recommendation

These are the improvements to be made to accommodate more features that will improve its performance. They include:

i. Having a backup to refer in case of any inconveniencies.
ii. Train more employees on how to use the system in order to get more systems admin.
iii. Buy modern computers systems which are more efficient fast and have better features
iv. Availability of (Uninterruptible Power Supply) UPS to restore power in case of power loss