Week 5: Interactive Assignment, Part V

Software Evolution and Maintenance

Eric Vara

The University of Arizona Global Campus

CST 301: Software Technology & Design

Professor John Howerton

February 02, 2023

**Software Evolution and Maintenance**

**Diagram

Description automatically generated**

**Four Clusters of N Systems**

The four clusters of N systems based on "quality" versus "business value" coordinates can be described as follows:

High Quality, High Business Value: These systems have both high quality and high business value, meaning they deliver high value to the business and are of high quality.

Low Quality, High Business Value: These systems have high business value but low quality, meaning they deliver high value to the business but have poor technical characteristics.

High Quality, Low Business Value: These systems have high quality but low business value, meaning they have strong technical characteristics but do not deliver significant value to the business.

Low Quality, Low Business Value: These systems have low quality and low business value, meaning they have poor technical characteristics and deliver little value to the business.

**Types of Software Maintenance**

Software maintenance refers to the process of modifying and updating a software system after it has been delivered to the customer. There are three main types of software maintenance:

Corrective Maintenance: This type of maintenance is performed to correct faults or bugs in a software system. It is the process of fixing errors and problems that have been detected in the software.

Adaptive Maintenance: This type of maintenance is performed to adapt the software system to changes in the operating environment. It is the process of modifying the software to work with new hardware or software systems.

Perfective Maintenance: This type of maintenance is performed to improve the functionality and performance of a software system. It is the process of adding new features, improving the user interface, and enhancing the performance of the software.

Each type of maintenance is performed for different reasons and has a different focus. The types of maintenance can overlap and may be performed in combination to ensure the continued operation and effectiveness of the software system.

**Reengineering of the Software Process**

A general model for the reengineering of software process involves several phases that are used to take a legacy program as input and produce an improved version of the same program as an output. Here are the Phases:

Assessment: This phase involves analyzing the existing software system to determine its strengths and weaknesses, as well as to identify any areas that need to be improved.

Reverse Engineering: This phase involves decompiling the existing software to understand its architecture and design, as well as to identify any issues that need to be addressed.

Requirements Analysis: This phase involves identifying the requirements for the improved software system, including any new features, changes to existing features, and performance improvements.

Design: This phase involves creating a detailed design for the improved software system, including its architecture, data structures, and algorithms.

Implementation: This phase involves actually implementing the design, using a programming language, to produce the improved software system.

Testing: This phase involves testing the improved software system to ensure that it meets the requirements and works as intended.

Deployment: This phase involves deploying the improved software system to the production environment, making it available to end users.

The reengineering of software process is a complex and time-consuming process, but it can lead to significant improvements in the quality, reliability, and performance of the software system.

**Work Cited**

Gomez, J., &amp; 20, U. J. (2022, June 20). The four types of software maintenance. Koombea. Retrieved February 2, 2023, from <https://www.koombea.com/blog/types-of-software-maintenance/>

Full Scale. (2022, June 24). What is software reengineering. Full Scale. Retrieved February 2, 2023, from <https://fullscale.io/blog/software-reengineering/>