Part 3 Client Change Requests

ScottishGlen

Introduction

As a software engineer, changing specifications is an inevitable event that you must prepare for during design and implementation. After further internal discussions at ScottishGlen changes to the specifications are requested. The changes are described below.

Client Change Requests

The client has identified that it will also be required to track software being used on the company network. Once again it must be possible to add, view, edit, or delete this asset data.

For now, it would be sufficient to store data about the operating system name, version, and manufacturer. The client would like your software to get this data automatically from the system on which it is running. It should also be possible to link hardware assets with software assets, allowing the client to determine where software is installed.

For both hardware and software assets, it would be useful if errors in data could be prevented, or identified and corrected, at the point of entry. Because of security concerns access to the system must be controlled and any user must be authenticated with a username and password. Pre-configured credentials may be used but must be stored securely. Reasonable precautions should be implemented to ensure the system functions reliably and securely.

The client envisions that asset data will assist in identifying vulnerable assets, which will allow the IT department to act. Because it is difficult for the client to track when assets become vulnerable to new exploits, the system must provide a feature to search for vulnerabilities to assets in an online database. The security consultant has advised that the NIST National Vulnerability Database (NVD)[1] would be suitable for this purpose. The system should enable checking an asset against the NVD and providing relevant information if a serious issue is found. This information can then be forwarded to the IT department for further investigation.

[1] https://nvd.nist.gov

Instructions

The client would like to see the final system by **Week 13** of the Term. You must submit the following:

1. Documentation which includes:
    a. Updated requirements specification for the final system, classified as functional and non-functional requirements.
    b. UML diagrams to illustrate your system design, consisting of:
        i. Updated Class diagram of your system.
        ii. Sequence diagram illustrating an interaction between objects in your system.
        iii. (optional) Any other UML diagrams that you think adds to the clarity of the final system.
    c. One-page discussion of how you identified and applied relevant design principles and patterns.
    d. Up to three-page discussion of how you implemented reliability, secure coding practices, and testing using a suitable methodology.
    e. One-page critical evaluation of your system and development process, suggesting improvements as appropriate.
2. Git repository containing the source code of the system and showing your commit history.
3. Screen recording with voice explanation of how your final system works. The recording should demonstrate the features from the change requests above. Briefly discuss your development tools and any code construction concepts that you used. The recording should be no more than 10-minutes.