**LEARNING JOURNAL**

**Student Name:** Habeeb Dashti

**Course:** Software Project Management

**Journal URL:** <https://github.com/habeebdashti/SOEN6841journal>

**Week 4:** 04/02/2024 – 10/02/2024

**Date:** 10/02/2024

**Key Concepts Learned:**

The summary of key concepts learned in week 4 sessions are:

* **Configuration Management System**

It is a set of tools, processes, and policies used to systematically manage changes to an organization's IT infrastructure, software applications, hardware devices, and documentation. It ensures that all elements within a system or environment are consistently configured, documented, and tracked throughout their lifecycle. A CMS typically includes features such as version control, change management, configuration tracking, and auditing capabilities to maintain integrity, stability, and reliability in complex IT environments. The primary goal of a configuration management system is to enable efficient management, control, and coordination of configuration items to support the organization's objectives and minimize the risk of errors or inconsistencies caused by changes.

* **Purpose of Configuration Management System**

The purpose of CMS is to ensure and uphold the integrity of work products through four key functions:

* **Identification -** Configuration identification by defining baseline components.
* **Change Control -** Management of configuration by providing a mechanism i.e., documentation, organizational body, procedures, etc., for preparing, evaluating, approving or disapproving all the changes throughout the lifecycle.
* **Accounting Status -** Tracking configuration status by providing a mechanism for maintaining a record of the evolution of a system at any time and reporting the traceability of all changes to the baseline throughout the lifecycle.
* **Auditing -** Conducting configuration audits by providing mechanism for determining the degree to which the current state of the system mirrors the system pictured in baseline and requirements documentation, establishing baseline and ensuring SCM process and procedures are performed.
* **Smoke Testing**

Smoke testing’s function is testing a method in a functionality. It is a general way to judge and test that method instead of using regression testing, stress testing, etc. It helps development teams identify critical issues early in the development process, allowing them to address them promptly and ensure the stability and quality of the software before proceeding with more extensive testing or deployment.

* **Strategies to deploy CMS**

Deploying a configuration management system (CMS) successfully for a project requires careful planning, communication, and implementation. By following the below mentioned strategies, organizations can deploy a configuration management system successfully for their projects, leading to improved collaboration, efficiency, and quality throughout the software development lifecycle.

**Strategies: -**

* Define clear objectives
* Select right CMS tool
* Develop a deployment plan
* Involve Stakeholders
* Provide training and support
* Start with baby steps and iterate
* Customize and configure
* Implement best practices
* Monitor and Evaluate
* Improve with each iteration

**Reflections on Case Study/course work:**

From the case study 5.5.1 provided, the following reflections can be drawn: -

1. **Global Team Collaboration**
2. **Role-Based Access Control**
3. **Local Testing for Build Verification**
4. **Efficient Configuration Management System**
5. **Automated Testing for Build Stability**
6. **Timely Issue Escalation**

The case study underscores the importance of effective configuration management, automated testing, collaboration, and issue resolution mechanisms in supporting successful software development projects, particularly in distributed development environments.

**Collaborative Learning:**

Due to Delivery #1 of Project for Group #25, most of my focus and time has been devoted to make our work successful. Thus, we the collaborative learning still withstands in the project sector currently. We got our documentation draft prepared; the format was edited by necessity and hence, several points to be included and excluded were discussed among the teammates. Me and one of my teammates have successfully done the market analysis part by finding several similar systems and filtering out fitting one. The execution of the selected similar systems was tested and checked on to find problems and then we discovered solutions to the existing problems to which we can make our project presentable in the market by offering what was missing.

**Further Research/Readings:**

My teammate and I spoke to several students at Concordia University to know more about why they would hesitate to use the existing similar projects. This kind of extrovert interactive research session gave us a lot more of insights that we could consider for our project. One of the examples that really felt crucial to us was many people pointed out that existing systems do not have instantaneous responsive feature which would allow any one at any time from any end of the earth to visit the system and get instant response for their services due unavailability of agents which is a drawback as they would have to wait for response each time.

**Adjustments to Goals:**

My adjustment to the goals is adding up baby steps one by one according to the plan or ideas and reactionary adaptations according to the situations as we move further. For example, this would include researching very well about the established organizations in the market and booking an interview with them to understand why their user base and why they have so many features missing in their project that would have enabled them to become great.