**SOFTWARE CONFIGURATION MANAGEMENT**

**PROJECT**

FINAL REVIEW

TOPIC: **Grocery Shopping (G-Market)**

**DONE BY:**

**Pooja V H (13MSE0080)**



Faculty

Prof. Jagadeesh G

Table of Contents

* Introduction
* SCM Management
* SCM Activities
* **SCM standards**
* **Function point analysis**
* **SCM Tool**
* **Tool Introduction**
* **Features**
* **Screenshots**
* **Conclusion**

**Introduction**

The purpose of the SCM plan is to describe how configuration management will be conducted throughout the project. Online grocery shopping Software Configuration Management Plan (SCMP) describes processes for documenting the software configuration management requirement approach for the online Shopping system. Configuration items (CI) are made by user based on the existing criteria CI is not mandatory in this domain but having version management and altering a specific baseline version of the same CI itself. This Software Configuration Management (SCM) Plan specifically addresses configuration management for software. The purpose of Software Configuration Management (SCM), in general, is to establish and maintain the integrity of work products using:

* + - * Configuration Identification
* Configuration Control
* Configuration Status Accounting
* Configuration Audit

**SCM Management:**

It describes the allocation of responsibilities authorities of SCM activities to the particular organization and individuals within the project structure.

USERS:

The registered users can login to the account by using username and password. They can buy any products which are cheaper in rate and good quality. They can also replace the products which are not satisfied for them. They can pay the money by online payment also.

ADMIN:

They can add new products to the portal and also can remove the older ones. They can contact the customers for the verification and confirmation of products.

**Risks Involved:**

* Product Size Risks

- Estimated size of product in number of programs, files, transactions.

- Size of database created or used by the product.

- Number of projected changes to the requirements for the product. Before delivery and after delivery.

* Business Risks - Amount and quality of product documentation that must be produced and delivered to the customer. - Delivery deadlines. - Costs associated with late delivery and defective product.
* Client Risks - When working for the first time. - Will the customer agree to spend time in formal requirements gathering meetings to identify project scope. - Communicating with developers. - Understanding of the SDLC process.
* Resource Risks - Number of Resources and skill sets. - Commitment for entire duration.
* Technology Risks - Customer demanding new technology - Requirements put excessive performance constraints on the product.

**SCM Activities:**

The following sections describe the components of the Software Configuration Management organization which are necessary to support the management of NEA. Software Configuration Management interfaces directly with systems development, testing, change management, QA and release management to incorporate new and updated product deliverables. The SCM control should be passed from the project or supplier to the service provider at the scheduled time and with accurate configuration records.

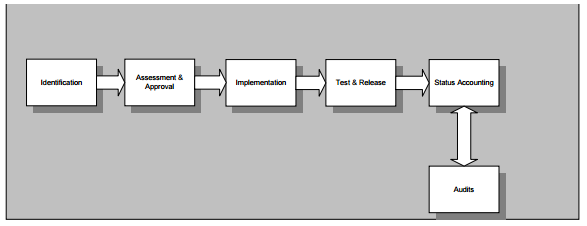


Figure 1 - SCM Flow Chart Process

1. **Configuration Identification:**

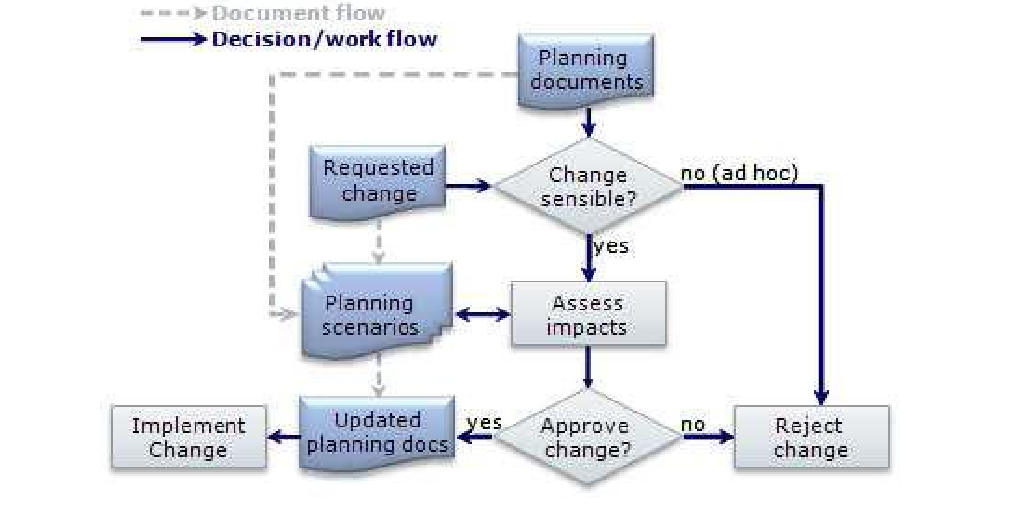
The purpose of Configuration Identification is to define the functional and physical characteristics of a CI in sufficient detail so that it may be developed, tested, evaluated, produced, competitively procured, accepted, operated, maintained, and supported. Configuration identification is established by baselines plus approved changes.

Configuration items need to change in online shopping

* Providing high security
* Low cost and high quality products
* Products ratings using user comments

1. **Change control**

Initially the configuration items can be identified based on the faults and failure result. In this project we are identifying configuration item based on Products quality problem, high cost problem, security problem. In our project identified CI are Low Budget products and products ratings. Product ratings are more effective in comparing the good products among them and it will suggest the good products to the customers.



CHANGE REQUEST FORM

• Project name • Project sponsor • Request number • Date • Originator • Change requested by • Description of requested changes • Reason for change • Areas of impact of proposed change (scope/ cost/ schedule/ risk) • Disposition (approved/ disproved/ deferred) • Priority ( emergency/ urgent/ low) • Funding scheme ( mgmt reserve, budget reserve, others) • Sign off approvals ( project manager, project customer, project sponsor, other, dates )

**SCM Standard:**

* MIL\_STD-2549 CM data interface
* This interface prescribes the data element, the data element definition, and the data element relationships that define the conceptual schema for CM data.
* Responsible for procuring, recording , maintaining and disseminating CM information.
* Good interface are always making the customers attractive and trustworthy among the products.

1. **Configuration Status Accounting:**

This section should:

* define how configuration item status information is to be collected, stored, processed and reported;
* identify the periodic reports to be provided about the status of the configured items, and their distribution;
* state what dynamic inquiry capabilities, if any, are to be provided;
* describe how to implement any special status accounting requirements specified by users.

It is details of recording the status of the configuration items and reporting them to the people who need to know about them. This information requirements of the project is Customer reviews and the product details would be saved by server. The admin frequently monitoring all the processes which is placed by Rating system using clustering techniques.

1. **AUDIT AND REVIEW:**

**SCM SCHEDULE:**

This section describes the sequence of the SCM activities, their interdependencies and relationship of the project life cycle, and the project milestones. In this project the phases of life cycle are requirement analysis, design, implementation and testing. Graphical representation using pert or Gantt chart help enhance the useful of this project description.

**SCM resource:**

In this project we are using the GIT to check the project implementation and version control. The accuracy of predicting the ratings of the products using the Comments is very high.

**SCM PLAN MAINTENANCE:**

The project plan maintenance is done by the team members to avoid the correction .The plan is scheduled by the analysis of the maintenance .The plan should be monitored and synchronized with the activities of the project.

**FUNCTION POINT:**

It measures the size in terms of the amount of functionality in a system. Function point is computed by calculating an unadjusted function point (UFC) and Technical complexity factor (TCF).

Unadjustment Function Point:

-- External Input-2

-- External Output-0

-- External Inquiry-2

UFP= 2\*4+0+2\*4=16

Value Adjustment Factor:

Data Communications- 4

Distributed Functions -2

Performance-4

Heavily Used Configuration- 3

Transaction Rate- 4

Online Data Entry-3

End User Efficiency-4

Online Update- 4

Complex Processing-3

Reusability- 2

Installation Ease- 1

Operational Ease -1

Multiple Sites- 1

Facilitate Change –3

VAF = 0.65 + 0.1 \* ∑ = 0.65 + 0.01 \* 39 = 1.04.

**Total Fp= 16 \* 1.04 = 16.64**

**SCM TOOL:**

We are choosing an open source SCM tool called GIT.

GIT is mainly user for version Control.

**Version Control:**

* Version control (or revision control, or source control) is all about managing multiple versions of documents, programs, web sites, etc.
  + Almost all “real” projects use some kind of version control
  + Essential for team projects, but also very useful for individual projects
* Some well-known version control systems are CVS, Subversion, Mercurial, and Git
  + CVS and Subversion use a “central” repository; users “check out” files, work on them, and “check them in”
  + Mercurial and Git treat all repositories as equal
* Distributed systems like Mercurial and Git are newer and are gradually replacing centralized systems like CVS and Subversion

**GIT Advantages**:

* Resilience
  + No one repository has more data than any other
* Speed
  + Very fast operations compared to other VCS (I’m looking at you CVS and Subversion)
* Space
  + Compression can be done across repository not just per file
  + Minimizes local size as well as push/pull data transfers
* Simplicity
  + Object model is very simple
* Large userbase with robust tools

Drawbacks:

Definite learning curve, especially for those used to centralized systems

* + Can sometimes seem overwhelming to learn
    - Conceptual difference
    - Huge amount of commends