



*HangOut Application*

**Software Configuration Management Plan**

Version 1.0

Prepared by Mac & Cheese

Jethro Phuah An Ping

Lam Zhi Fah

Alicia Chua Jieying

Ernest Ang Cheng Han

Ong Jing Heng Shaun

Ong Sim Hao

Shannon Tan Xinyi

13 October 2021

**VERSION HISTORY**

| **Version #** | **Implemented By** | **Revision Date** | **Approved By** | **Approval Date** | **Reason** |
| --- | --- | --- | --- | --- | --- |
| *1.0* | *Shaun Ong*  *Ong Sim Hao*  *Lam Zhi Fah* | *13/10/2021* | *Jethro Phuah An Ping* | *15/10/2021* | *Initial Software Configuration Management Plan* |

**TABLE OF CONTENTS**

1. [**Identification**](#_heading=h.gry8e1xa3mpb) **4**
   1. [Document Overview](#_heading=h.puccmoc9epz7) 4
   2. [Abbreviations and Glossary](#_heading=h.6198opzbtftq) 4
      1. [Abbreviations](#_heading=h.ts6ky0cwzeyn) 4
      2. [Glossary](#_heading=h.bo502ropsv0o) 4
   3. [References](#_heading=h.d7qh297fi8i8) 5
      1. [Project References](#_heading=h.t9da0wnlwrol) 5
      2. [Standard and Regulatory References](#_heading=h.mwi9ym7z1nj1) 5
   4. [Conventions](#_heading=h.7g396wl5l0ur) 5
2. [**Organization**](#_heading=h.h684lqiofbdq) **7**
   1. [Activities and Responsibilities](#_heading=h.hxxcj6en7c6d) 7
      1. [Decisions Process and Accountabilities](#_heading=h.hpi4ytnmdbs1) 8
3. [**Configuration identification**](#_heading=h.1hac8tmkeuc) **10**
   1. [Identification rules](#_heading=h.tjrn3g9rirwv) 10
      1. [Identification rules of configuration items](#_heading=h.2mp1gp79xhj2) 10
         1. [Identification of a configuration item](#_heading=h.5n8qlug50f7z) 10
         2. [Version number of a configuration item](#_heading=h.2gukdgtwsx1c) 10
      2. [Identification rules of SOUPs](#_heading=h.ktqux29q3blw) 11
         1. [Identification of a SOUP](#_heading=h.akrjy5ioafys) 11
         2. [Version number of a SOUP](#_heading=h.xlmayanxw83w) 11
      3. [Identification rules of documents](#_heading=h.ay99t0mosg2v) 11
         1. [Description of documents identifiers](#_heading=h.o293vivvdsed) 11
         2. [Definition and evolution of the revision index](#_heading=h.er2avmvl3gfl) 12
      4. [Identification rules of a media](#_heading=h.e5p1m6lcd8f) 12
         1. [Internal Identification](#_heading=h.vadxx7fngv64) 12
   2. [Reference configuration identification](#_heading=h.38frjuj1gikv) 12
   3. [Configuration Baseline Management](#_heading=h.uqav3vez3sck) 12
4. [**Configuration Control**](#_heading=h.mt4xe1ugfwie) **14**
   1. [Change Management](#_heading=h.irchj757sa49) 14
   2. [Interface Management](#_heading=h.w9mvdpmwur9h) 15
   3. [Evolutions control of SOUP items](#_heading=h.csach5aefqq) 15
5. [**Configuration support activities**](#_heading=h.yud78xtjg8s6) **16**
   1. [Configuration Status Accounting](#_heading=h.yhwdld2khvph) 16
      1. [Evolutions Traceability](#_heading=h.u2hb8e8fteue) 16
      2. [Setting up Configuration Status](#_heading=h.3lq7a9pvxl82) 16
      3. [Configuration Status Diffusion](#_heading=h.mrhkgu7taw68) 16
      4. [Configuration Status Records Storage](#_heading=h.43toiny1ep72) 16
   2. [Configuration Audits](#_heading=h.e8x1hmfo4c3a) 17
   3. [Reviews](#_heading=h.tidic4vj70nd) 17
   4. [Configuration Management Plan Maintenance](#_heading=h.9qjvfaccdvs3) 19

# Identification

This document amplifies the “Configuration management” of the Project Management Plan.

## Document Overview

This document details the software configuration management plan of Software Project Hangout (Mobile Application). The purpose of the document is to establish and maintain the integrity of the Software Configuration using the following:

1. Configuration Identification
2. Configuration Control
3. Configuration Status Accounting
4. Configuration Audit

## Abbreviations and Glossary

### Abbreviations

| **Short Form** | **Long Form** |
| --- | --- |
| **CI** | Configuration Item |
| **CM** | Configuration Management |
| **CMWG** | Configuration Management Working Group |
| **ITIL** | Information Technology Infrastructure Library |
| **SCM** | Software Configuration Manager |
| **TS** | Technical Specification |
| **QA** | Quality Assurance |

### Glossary

| **Name** | **Definition** |
| --- | --- |
| **Configuration Item** | The ITIL 4 defines a configuration item as any component that has to be managed to effectively deliver an IT service |
| **Configuration Management** | Configuration Management ensures that the physical and functional arrangement of configuration items meet requirements throughout the life cycle of the software project |
| **Configuration Management Working Group** | The configuration management working group’s main purpose is to ensure that configuration management is properly managed |
| **Information Technology Infrastructure Library** | The ITIL describes the best practices for the delivery of IT services |

## References

### Project References

| **#** | **Document Identifier** | **Document Title** |
| --- | --- | --- |
| 1 | ID 1 | NASA Configuration Management Plan (CMP) Template |
| 2 | ID 2 | State of Michigan Software Configuration Management Plan |
| 3 | ID 3 | Configuration Management Overview and Phase 1 Implementation Guidance |

### **Standard and** R**egulatory References**

| **#** | **Document Identifier** | **Document Title** |
| --- | --- | --- |
| 1 | ID 1 | Standard IEC 62304 for Software Development |

## Conventions

**Typographical Conventions:**

| **Convention** | **Meaning** |
| --- | --- |
| *Italicised* | Italics are used to represent documents, product names, program names, or book titles |
| **Bolded** | Names of fields, buttons, (GUI) elements, text fields, flags, login masks, shortcuts, entities and windows appear in boldface type |

**Naming Conventions:**

| **Type of Document** | **Naming Convention** |
| --- | --- |
| **Configuration Item** | <Configuration item name>\_Major.Minor |
| **SOUP Configuration Item** | <Manufacturer name>\_<Software name>\_X.Y.Z |
| **Document Identification** | <document type>\_<document number>\_<revision index> |

# 

# **Organization**

The software configuration is managed by members of the project, each with their varying roles.   
  
Responsibilities are shared between:

1. **The Software Configuration Manager (SCM)**

The software configuration manager is in charge of reviewing and constantly updating existing configuration management plans. They work with the team to develop and maintain configuration management tools.

1. **The Auditor**

The auditor is responsible for ensuring that all information in Configuration Management is accurate and that all CIs are properly identified and recorded in Configuration Management.

1. **The Project Manager**

The project manager plays an integral role in planning, directing and organising modular parts of the software project while ensuring that these parts are completed on time, in budget and relevant to the scope of the project.

1. **The Technical Manager**

Technical managers are generally responsible for making technical decisions for technical issues that arise in a software project after considering all aspects of the software and hardware.

1. **The Quality Manager**

The quality manager plays an integral role in ensuring that the software product meets the end users’ expectations. This is achieved by working closely with the QA team across various levels of testing, while maintaining close communication with the other teams to ensure that the project stays on scope and in line with the demands of the end users.

## Activities and Responsibilities

| **Activities when setting up the project** | **Person responsible** |
| --- | --- |
| Identify the configuration items | SCM |
| Install the bug repository tool and set up the database | SCM |
| Install the software configuration repository tool and set up the database | SCM |
| Manage and structure the reference space | SCM |
| Define the configuration processes | SCM |

| **Activities during the project lifecycle** | **Person responsible** |
| --- | --- |
| Export components for modification, test or delivery | SCM |
| Set under control validated components | SCM |
| Create version, write version delivery document | SCM |
| Approve reference configurations | Project manager |
| Verify version to be delivered and authorise deliveries | Project manager |
| Backup spaces | SCM |
| Do configuration audits | Quality Manager |
| Inspect configuration records | Quality Manager |
| Archive reference version | SCM |

| **Management activities** | **Person responsible** |
| --- | --- |
| Ensure team follows the SCM process | SCM |
| Approve or reject change requests | SCM |
| Manage versions and archives | SCM |
| Manage configuration records | SCM |
| Produce reports and statistics | SCM |
| Manage reference space and its access control list | SCM |
| Manage spaces backup and archive media | SCM |
| Manage quality reports | Quality Manager |
| Maintaining configuration of code | Technical Manager |
| Check changes and resolve conflicts | Technical Manager |
| Ensure that the product is developed within a certain time frame | Project Manager |
| Monitors the progress of development and recognises issues in the SCM process | Project Manager |
| Generate reports about the status of the software system | Project Manager |
| Make sure that processes and policies are followed for creating, changing and testing. | Project Manager |

### Decisions Process and Accountabilities

Responsibilities during reviews, audits and approvals are listed below:

At the end of an activity of the project:

| **Activities** | **Person Responsible** |
| --- | --- |
| Do a configuration freeze | SCM |
| Present a configuration state of the components impacted by the activity | SCM |
| Present a documentation state of the components impacted by the activity | SCM |

During a configuration management process audit:

| **Activities** | **Person Responsible** |
| --- | --- |
| Do the configuration management process audit | Auditor |
| Ensure consistency and completeness of release | Auditor |
| Present the records of the configuration management process | SCM |
| Present the quality records of the configuration management process | Quality Manager |
| Present the records of the documentation management process | SCM |

# 

# Configuration identification

## **Identification rules**

### Identification rules of configuration items

#### Identification of a configuration item

For our software project, configuration items to be identified are as follows:

1. Hardware
2. Software
3. Tangible Assets
4. Documentation
5. Equipment

Each configuration item has the following naming convention:

*<Configuration item name>\_Major.Minor*

where:

"Major.Minor" is the version of the configuration item.

#### Version number of a configuration item

The version number is an integral identifier to the delivery of any configuration item. If the product or its documentation was modified, this number shall be incremented.

The Modification rules for the version number are as follows:

Given a version number Major.Minor, increment the:

1. Major version when major changes are made
2. Minor version when minor changes are made

### Identification rules of SOUPs

**SOUP** refers to software of unknown provenance. SOUPs are identified by:

1. The name of the manufacturer
2. The name of the software or the library
3. The version of the library or software

For our project, we will be adopting our own identification instead of SOUP to maintain consistency throughout our project.

#### Identification of a SOUP

Each configuration item has the following naming conventions:

*<Manufacturer name>\_<Software name>\_X.Y.Z* where: "X.Y.Z" is the version of the SOUP.

#### Version number of a SOUP

The definition rules for version number are as follows:

Given a version number X.Y.Z, increment the:

1. X version when incompatible API changes are made
2. Y version when functionality in a backwards compatible manner is added
3. Z version when backwards compatible bug fixes are made

### Identification rules of documents

#### Description of documents identifiers

The identification of documents have the following naming conventions:

*<document type>\_<document number>\_<revision index>*

1. "Document type" is:
   1. Foo for FOO documents
   2. BAR for bar documents
2. “Document number" is a incremental number, with a separate list for each document type
3. "Revision index" designates the approved iteration of the document. The revision index is V1.1 for the first iteration, V1.2 for the second and so on

#### Definition and evolution of the revision index

The attribution of a revision index is a prerequisite to any delivery of a document or file. This index shall be incremented before the diffusion of a modified document.

The definition rules of a revision index are the following:

Given a revision index Major.Minor increment the:

1. Major version when major changes are made
2. Minor version when minor changes are made

### Identification rules of a media

A media is for example, a video tape or a DVD.

#### Internal Identification

The identification of a media is described below:

*<configuration item identification >\_<media>\_<volume>*

where:

"media" is the media number,

"volume" is an incremental number to distinguish the media if the delivery contains more than one media.

## Reference configuration identification

Each reference configuration is defined by:

1. An identifier
2. Its content listed in the corresponding Version Delivery Description document
3. The acceptance or validation reviews associated with the building of the reference configuration

A reference configuration is established for each design review and each test review of the project.

## Configuration Baseline Management

These are the 3 baselines that will be established:

1. Functional baseline (FBL), is the approved configuration documentation that describes the system functional, interoperability and interface characteristics. It also highlights the verification required to determine if those specified characteristics are achieved successfully
2. Allocated baseline (ABL), which describes the design of the functional, performance and interface characteristics that are allocated from a higher level requirements document or CI. It also highlights the verification required to determine if those specified characteristics are achieved successfully
3. Product baseline (PBL), which is the approved technical documentation that describes the configuration of a CI during its production, fielding as well as operation support phases. It also consists of completed and accepted system components and documentation that identifies these products

# 

# Configuration Control

The process for managing configuration changes and variances in configurations are as follows:

1. The software configuration manager reviews the modification of an existing CI.
2. Verify that all information to update the CI is available and correct. The task should contain a description of the attributes that are to be modified
3. If not all the information required is complete and correct, reject the modification. A valid reason and additional improvements should be suggested
4. The quality manager is in charge of assessing the reason for rejection
5. Document all the details relating to the rejected modification task
6. If modification is successful, the software configuration manager will then modify the configuration details in the configuration management database. Configuration modifications to be documented include:
   1. Status
   2. Location
   3. Transfer of Ownership
7. After completion of the modification task to CI, the task can now be closed

## Change Management

A formal change control process should be used to ensure that changes to the project baselines are properly considered as unmanaged changes can easily snowball out of control and send a project into a disarray.

The process for controlling changes to the baselines and for tracking the implementation of those changes are as follows:

1. All change requests must follow a formal format. The change request must include a description of the change, the reason for the change and lastly, the impact on project schedule and cost baselines
2. Change requests are emitted from by the project manager according to the problem resolution process and are to be reviewed by the software configuration manager and quality manager
3. When a change request is approved by the project manager/product manager, a branch is created in the database and a new project baseline is established
4. The branch identification is <baseline name>\_VN where the version number (VN) has to be incremented each time a new modification is made to that particular baseline
5. Branch content is the format in the change request

## Interface Management

Interfaces we will be using include 3rd party APIs which is an interface for our software to control and communicate with other software.

The procedures for identification of interface requirements are as follows:

1. Prepare or update interface management procedures
2. Conduct interface management during product integration activities
3. Conduct interface control if there are any interface changes

## Evolutions control of SOUP items

Our group will freeze SOUPS at the beginning of the project. Although this method of evolution control might not always be feasible, it is easy to implement and not costly at all.

# 

# Configuration support activities

## Configuration Status Accounting

Configuration Status Accounting (CSA) is the process to record, store, maintain and report the status of configuration items during the software lifecycle. All software and related documentation should be tracked throughout the software life. CSA provides our group with the ability to identify the configuration of CIs at a given point in time. This is useful for processes such as problem management, audits and compliance.

### Evolutions Traceability

The traceability of modifications of items given their types:

1. Document: The modification sheet number identifies the origin of the modification
2. The modified paragraphs in the document are identified, if possible, by revision marks
3. Source file: The software configuration management tool records, for each source file or group of source files, a comment which describes the modification.
4. Configuration item: The Version Delivery Description of the article identifies the modification sheet included in the current version

The modification sheet describes the modifications done to the components with enough precision to identify the modified parts.

### Setting up Configuration Status

The SCM is in charge of writing the VDD, setting up the state of all versions and of each configuration article with:

1. The label
2. The version number
3. The creation date of the VDD

### Configuration Status Diffusion

The SCM and the quality manager write the VDD.

### Configuration Status Records Storage

The records are stored in a configuration folder, which contains:

1. The requests sorted by record number
2. The software documents
3. The VDDs
4. The configuration states are sorted chronologically

## Configuration Audits

Our team will be adopting the following two types of configuration audits:

1. **Functional Configuration Audit**

The functional configuration audit provides an independent evaluation of a software product to verify that the CI actual functionality and performance is consistent with the relevant requirement specification. The functional configuration audit is conducted prior to or in conjunction with the physical configuration audit.

During the functional configuration audit, the following issues will be addressed:

1. Whether the risk management plans are kept up to date
2. Whether critical project achievements, success criterion and success metrics are established
3. Whether there are any issues with deployment, operations, support or production

The responsibility for the conducting of the functional configuration audit lies mainly on the project manager and the auditor.

1. **Physical Configuration Audit**

The physical configuration audit provides an independent evaluation of a software product’s CI to confirm that all the components have the specifications required.

Types of Physical Configuration Audits:

* Software Physical Configuration Audit

The software physical configuration audit measures the system software against its design or deliverable documentation.

* Hardware Physical Configuration Audit

The hardware physical configuration audit measures the physical hardware against its design or deliverable documentation.

## Reviews

Across the lifetime of the software project, the following technical reviews shall be carried out in compliance with the IEEE Standard 1028:

1. **Software Peer Reviews**

The process of software peer reviews entails the evaluation of the technical code and quality of the software project. The primary purpose of the software peer review is to identify and resolve defects in the software. It is carried out between members of the software project team.  
  
Software peer reviews can be further broken down into different types:

1. Code Reviews:
   1. Code reviews shall be conducted between peers to correct errors and redundancies in the project’s source code. Code reviews are conducted with the purpose of improving the quality and security of the final software product
   2. In code reviews, the project manager shall play an integral role in managing time and ensuring that the code reviews are conducted religiously across all members
2. Pair Programming:
   1. Pair programming is a type of code review in which two programmers work together to develop a modular piece of code together
   2. In pair programming, the project manager shall play an integral role in splitting the team into pairs based on their individual expertises and ensuring that they work together on each modular piece of code as a pair
3. Technical Reviews:
   1. Technical reviews entail the involvement of qualified personnel to run through the software to identify discrepancies between the product and its intended functionalities
   2. In technical reviews, the quality manager and technical manager shall play integral roles in ensuring that the reviews are carried out religiously, and in taking notes on discrepancies between the product and its intended functionalities
4. **Software Management Reviews**

Software management reviews are conducted by management representatives to ensure that the work completed by the team is what is required from the employer.

In software management reviews, the project manager shall work hand in hand with management representatives to ensure that the work completed by each team member is properly disseminated to the employers.

1. **Software Audit Reviews**

Software audit reviews are external reviews conducted by auditors who are not part of the project development team. The purpose of software audit reviews is to assess if a software product complies with its intended functionalities, standards required, and other important criterions.

For software audit reviews, the auditor shall play an integral role in working with outside auditors to ensure that the software project complies with its intended functionalities.

## Configuration Management Plan Maintenance

To ensure the proper maintenance of the configuration management plan, the responsibilities of each of the 5 managers are as follows:

1. **Software Configuration Manager**

The overall maintenance of the configuration management plan will be overseen by the Software Configuration Manager. The configuration management plan is to be updated on a bi-weekly basis. The updating of the configuration management plan must be conducted with the assistance of the other 4 managers to ensure the accuracy of the updates. Therefore, the responsibility of the SCM extends to the orchestration of biweekly meetings to ensure that each component of the configuration plan is updated accurately.

1. **Quality Manager**

The quality manager is in charge of conducting configuration audits, inspecting configuration records, and presenting the quality records of the configuration management process. The quality manager shall update the sections of the configuration plan that fall under his jurisdiction on a bi-weekly basis.

1. **Technical Manager**

The technical manager is in charge of making decisions for the technical components of the software project, after thorough analysis of the software and hardware. He shall therefore update the sections of the configuration plan that fall under his jurisdiction on abi-weekly basis.

1. **Auditor**

The auditor is responsible for ensuring that all information in Configuration Management is accurate and that all CIs are properly identified and recorded in Configuration Management. Therefore, he shall play an integral role in inspecting the updated configuration plan on a bi-weekly basis to ensure that there are no errors or conflicts in the document.

1. **Project Manager**

The project manager plays an integral role in planning, directing and organising modular parts of the software project while ensuring that these parts are completed on time, in budget and relevant to the scope of the project. Therefore, he will play a pivotal role to ensure that the bi-weekly schedule for the updating of the configuration plan is adhered to.