

Software Configuration Management Plan of Friendstagram	
Version: 1.0	Page 1 / 10

Group: Inspiration

TABLE OF CONTENTS

Identification	2
Document overview	2
Abbreviations and Glossary	2
Abbreviations	2
Glossary	2
References	2
Project References	2
Organization	3
Activities and responsibilities	3
Decisions process and responsibilities	3
Configuration identification	4
Identification rules	4
Identification rules of configuration items	4
Identification rules of SOUPs	5
Identification rules of documents	5
Reference configuration identification	5
Configuration Baseline Management	6
Configuration control	6
Change Management	6
Evolutions control of SOUP items	7
Configuration Support Activities	7
Configuration Status Accounting	7
Evolutions traceability	7
Setting up Configuration status	8
Configuration status diffusion	8
Configuration status records storage	8
Configuration audits	8
Reviews	9
Configuration management plan maintenance	9

Software Configuration Management Plan of Friendstagram	
Version: 1.0	Page 2 / 10

1. Identification

a. Document overview

This document contains the Software Configuration Management plan of Friendstagram.

b. Abbreviations and Glossary

i. Abbreviations

Abbreviation	Phrase
SOUP	Software Of Unknown Provenance
D01	Document 1
SCM	Software Configuration Manager
PM	Project Manager
QM	Quality Manager
FBL	Functional Baseline
ABL	Allocated Baseline
PBL	Product Baseline
PR	Product Resolution
MC	Multiple Configuration
VDD	Version Description Document

ii. Glossary

Word	Meaning
Configuration Item	A fundamental structural unit of a configuration management system.

Software Configuration Management Plan of Friendstagram		
Version: 1.0		Page 3 / 10

c. ***References***

i. **Project References**

#	Document Identifier	Document Title
01	D01	D01_Project_Proposal_V1.0.pdf
02	D02	D02_Use_Case_Model_V2.0.pdf
03	D03	D03_SystemRequirementSpecification_V1.0.pdf
04	D04	D04_QualityPlan_V1.0.pdf
05	D05	D05_ProjectPlan_V1.0.pdf
06	D06	D06_RiskManagementPlan_V1.0.pdf
07	D07	D07_DesignReportOnSoftwareMaintainability_V1.0.pdf
08	D08	D08_ConfigurationManagementPlan_V1.0.pdf
09	D09	D09_ChangeManagementPlan_V1.0.pdf
10	D10	D10_ReleasePlan_V1.0.pdf
11	D11	D11_TestPlan_V1.0.pdf
12	D12	D12_TestCasesAndRequirementsTestCoverageReport_V1.0.pdf

Software Configuration Management Plan of Friendstagram

Version: 1.0

Page 4 / 10

2. Organization

The software configuration is managed by members of the project, with specific tools. Responsibilities are shared between

- The Software Configuration Manager (SCM),
- The Project Manager,
- The Technical Manager.

a. Activities and responsibilities

Specific tools will be used by personnel of the Friendstagram project team to manage software configuration. There are 3 responsibility roles, namely: Software Configuration Manager (SCM), Project Manager (PM) & Quality Manager (QM).

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
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

i. Decisions process and responsibilities

Software Configuration Management Plan of Friendstagram	
Version: 1.0	Page 5 / 10

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	Project Manager
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

3. Configuration identification

a. Identification rules

i. Identification rules of configuration items

1. Identification of a configuration item

This naming convention format will be used for configuration items:

Format: <CI_ID>_<NAME>_<VERSION>. <FILETYPE>

Example: CI001_Friendstagram_V1.0.apk

2. Version number of a configuration item

Before delivering any configuration items, version numbers must be attributed. These version numbers must be incremented before a new delivery if changes were made to the project or its respective document files. The defining rules for version numbering are as follow:

Incrementation Criteria	
0.X	X is incremented* only when a modification or bug fix is performed on products or documents which are not ready and not delivered yet.
X.0	X is incremented* only when an upgrade is performed on products or documents which are already ready and delivered.

*Every incrementation will be documented

ii. Identification rules of SOUPs

1. Identification of a SOUP

This naming convention format will be used for SOUP, where manufacturer name is represented as <MANUFACTURER>:

Software Configuration Management Plan of Friendstagram

Version: 1.0

Page 6 / 10

Format: <NAME>_<VERSION>_<MANUFACTURER>.<FILETYPE>
Example: React_v1.01_Facebook.js

2. Version number of a SOUP

This naming convention format will be used for SOUP, where version number is represented as <VERSION>:

Format: <NAME>_<VERSION>_<MANUFACTURER>.<FILETYPE>
Example: React_v1.01_Facebook.js

iii. Identification rules of documents

1. Description of documents identifiers

The identification of documents is described below:

XXX_<document type>_<document number>_<revision index>

where:

"document type" is:

- word for Word documents,
- excel for Excel documents

"document number" is a incremental number, with a separate list for each document type,
"revision index" designates the approved iteration of the document. The revision index is V1 for the first iteration, V2 for the second and so on.

2. 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:

Incrementation Criteria	
0.X	X is incremented* only when a modification or bug fix is performed on products or documents which are not ready and not delivered yet.
X.0	X is incremented* only when an upgrade is performed on products or documents which are already ready and delivered

*Every incrementation will be documented

b. Reference configuration identification

Each reference configuration is defined by:

- An identifier,
- Its content listed in the corresponding Version Delivery Description document,
- 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.

Software Configuration Management Plan of Friendstagram	
Version: 1.0	Page 7 / 10

c. Configuration Baseline Management

A baseline is an agreed description of the attributes of a product, at a point in time, which serves as a basis for defining change. This is usually agreed upon through a review, walk-through or inspection process. The goal is to describe the baselines to be established and explain when and how they will be defined and controlled. Examples are as follow:

Baseline	Description	Person in-charge
Functional (FBL)	It describes the functional characteristics of the Friendstagram web application. It consists of a system's top level configuration item's performance requirements. It also includes requirements needed to show attainment of specified characteristics.	Project Manager
Allocated (ABL)	It describes the design of the functional and interface characteristics. It is also the approved performance-oriented configuration documentation for a configuration item to be developed. It also extends the top-level performance requirements of the functional baseline to sufficient detail for defining the functional and performance characteristics and for initiating detailed design for a configuration item.	Quality Analysis Manager
Product (PBL)	This describes the completed and accepted system components, as well as documentation that identifies these products. It also shows the configuration information of the item during the different phases of its life cycle, such as production, deployment and operational support.	Software Configuration Manager

4. Configuration control

Before any modification, the project manager must approve them. After approval has been granted, the version numbers of the modified items will increase according to the type of changes made. Afterwards, the items can be uploaded to the working folder and backup folder in the SVN, Github and Wiki. Having a backup folder helps to serve as a prior reversion checkpoint for files whenever there are bugs, mistakes or errors that cannot be fixed.

Software Configuration Management Plan of Friendstagram

Version: 1.0

Page 8 / 10

a. Change Management

For both problem resolution and multiple configuration, they will have the same process for controlling changes to the baselines and tracking the implementation of those changes. But, both processes will be separated through different folder structures and names.

Examples for problem resolution & multiple configuration:

- Project Manager have to approve the document changes
- Software Configuration Manager must create a branch after change request has been approved by project manager / product manager
- The branch identification will be done using version number
- Branch content will includes source code and documentations
- Branch information will be documented through commit messages
- Abbreviations such as 'PR' and 'MC' will be used to label each item that has been changed to avoid misunderstandings

b. Evolutions control of SOUP items

As the project evolves, SOUP items may evolve as well and change to newer versions. However, this may cause compatibility issues with other current SOUP items that are being used in this project. Therefore, by default, we will not update SOUP items at all from the beginning of the project. In the event where updating SOUP items is critical to the project, we will check to ensure that all other SOUP items will still be compatible with it before updating it to a newer version. If it is incompatible, we will not upgrade the SOUP item, but instead, find alternatives. Alternatives could mean finding other SOUP items that provide the same functionality as the one in the new version of the SOUP item which we wanted to upgrade.

5. Configuration Support Activities

a. Configuration Status Accounting

Configuration Status Accounting (CSA) is the process to record, store, maintain and report the status of configuration items during the software development life cycle of the Friendstagram project. All software and related documentation must be tracked throughout the software life of the Friendstagram project.

i. Evolutions traceability

The traceability of modifications of items given their types:

Type	Description
Document	Modification sheet number identifies modification origin. Revision marks are used to identify the modified paragraphs in the document.
Source File	A comment which describes the modification recorded for each source file, or a group of source files, by the software configuration management tool.

Software Configuration Management Plan of Friendstagram	
Version: 1.0	Page 9 / 10

Configuration Item	The modification sheet included in current versions is identified by the Version Delivery Description. It consists of the modifications done to the components with enough precision to identify the modified parts.
Current baselines	Baselines of the current configuration items should be made available to all technical developers at all times
CM reports	The periodic reports on the status of the configuration management items should be made available to all stakeholders on an agreed-to frequency and at key life cycle reviews

ii. Setting up Configuration status

The SCM sets up the state of all versions of each configuration article with:

- The label,
- The version number,
- The creation date of the Version Description Document,

The SCM writes the Version Description Document to make sure all subsequent versions of software are tracked properly.

iii. Configuration status diffusion

The SCM and the quality manager write the VDD.

iv. Configuration status records storage

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

- The requests sorted by record number,
- The software documents,
- The VDD's,
- The configuration states are sorted chronologically.

b. Configuration audits

This section describes peer review audits and formal audits that are used to assess the Friendstagram project's compliance with the Configuration Management Plan. Examples of audits are listed as follow:

Audit Type	Description
Baseline	Baseline audit will be done among developers and stakeholders to find potential improvements. There will also be a formal audit that checks whether processes are followed.
Functional Configuration	The Software Configuration Management Planning (SCMP) process planning should begin at the beginning of the project. This will allow it to follow the

Software Configuration Management Plan of Friendstagram	
Version: 1.0	Page 10 / 10

	functional requirements stated previously in the System Requirements Specification (SRS) document.
Software Configuration	Software Configuration audits verify that all the software product satisfies the baseline needs. It ensures that what is built is what is delivered. The configuration manager will conduct an audit to ensure that the configuration items are in place before conducting a formal audit to make sure that configuration items are working well.

c. *Reviews*

After the User Acceptable Testing, developers will conduct technical reviews which will be collated by the configuration manager. The configuration manager will then check to make sure that the reviews are done correctly according to the latest baseline.

The developers should continue to modify the configuration until the established baseline standards are met. Afterwards, the configuration manager has to communicate with the project manager to follow up on any outdated baselines. Overall, the approval of the project must come from the project manager.

If there are configurations that have not been tested thoroughly and undergo audits, the users might be the one who finds out the errors or missing features. This can potentially lead to early termination of contract as users can get unhappy especially if the missing feature is extremely important. Hence, there should be early review to avoid such issues.

d. *Configuration management plan maintenance*

The project manager will be responsible for taking the initiative and being in charge of updating the configuration management plan. All the changes have to be agreed by the project manager. The changes must then be made known to every member by the Project Manager on a weekly basis to ensure that the configuration management plan is always in its latest version. Also, the Lead Developer will assist the Project Manager in filling up the technical aspect of the configuration management plan.