Configuration Management Plan

DOCUMENT CONFIGURATION PLAN CODE: D-GDC002 **VERSION:** 1.0 **PAGE:** 1-10

Configuration Plan

Project: Classroom Reservation System
Product: Information System for Classroom Reservation Management (SIGRA)
Version: 1.0.1

Revision History

VERSION	DATE	AUTHOR	DESCRIPTION
1.0.1	18/03/2025	Maximiliano Arroyo, Enrique Rodríguez, Miguel González	Document creation

1. Introduction

This document details the Configuration Management Plan for the Information System for Classroom Reservation Management (SIGRA). Its objective is to establish and maintain the integrity of software products throughout their lifecycle. Additionally, it seeks to ensure that any changes made to the system are managed in an orderly and controlled manner, minimizing risks and ensuring software quality.

1.1 Objective and Scope

The objective of Software Configuration Management is to ensure that changes in software elements are properly controlled and managed. The goal is to maintain the integrity of the classroom reservation system, allowing users to manage reservations efficiently and securely. This plan applies to all phases of software development, including design, implementation, testing, and deployment.

1.2 Terminology

- **SCM (Software Configuration Management):** Software Configuration Management.
- **SCR (System/Software Change Request):** Request for Change in the System/Software.
- **CCB (Configuration Control Board):** Configuration Control Committee.
- **CI (Configuration Item):** Element under Configuration Management.
- **SCA (Software Change Authorization):** Authorization for Software Change.
- **Baseline:** Set of components with a specific version that ensures system stability.
- **Code Repository:** Location where all versions of the source code are stored.

2. System Configuration Management

2.1 Computing Environment and Tools

This system will be developed using modern technologies to ensure its scalability, security, and performance. Tools will be used to efficiently manage code and changes made.

TASKS	PRODUCTS	PARTICIPANTS
Establishment of the Management Plan	Configuration document	Development team
Implementation of version control	Repository in GitHub/GitLab	Developers
Test automation	Testing frameworks	QA Engineers
Incident management	Tracking platform (JIRA, Trello)	Project managers
CI/CD Implementation	Continuous integration pipelines	DevOps

2.2 Organization and Responsibilities

The development team will be structured as follows:

NAME	ROLE	EMAIL
Maximiliano Arroyo	Project Manager	max.arroyo@example.com
Enrique Rodríguez	Documentation Manager	enrique.rod@example.com
Miguel González	Lead Developer	miguel.gonz@example.com

Each team member is responsible for ensuring the correct implementation of configuration management in their respective areas.

3. SCM Activities

3.1 Configuration Identification

Configuration items include:

- Source code.
- Technical and user documentation.
- Requirement specifications.
- Data and process models.
- Database scripts.
- Server and deployment configurations.

3.2 Change and Configuration Control

3.2.1 Change Processing and Approval

Each change request must be evaluated and approved by the Configuration Control Board. The following steps are established to manage changes:

1. Receipt of the change request.

2. Evaluation of impact and feasibility.
3. Approval or rejection of the request.
4. Implementation of the change by the development team.
5. Testing and validation of the change.
6. Documentation and version update.

3.2.2 Members and Procedures

RESPONSIBLE	ACTIVITY
SCMR	Identify CIs and define naming standards.
Developers	Implement and validate approved changes.
QA Engineers	Perform validation tests before release.
Infrastructure Administrators	Deploy changes in test and production environments.

3.3 Configuration Audits and Reviews

Periodic audits will be conducted to validate the status of configuration items and ensure their alignment with established requirements. These audits will verify:

- Proper identification and control of configuration items.
- Proper documentation of changes.
- Version consistency and storage in the repository.

3.4 Interface Control

Interface control will ensure that changes to configuration items do not affect system interoperability. Compatibility tests will be established before each version release.

4. Schedule

Deliverables will be defined in the process model schedule, with periodic reviews and audits. Biweekly meetings are planned to review configuration status.

5. Resources and Training

RESOURCES	DESCRIPTION
Control Tools	GitHub, GitLab, JIRA
Infrastructure	Cloud servers
Security	HTTPS, two-factor authentication

RESOURCES	DESCRIPTION
Training	Version control and change management training

6. Control Points

Configuration control processes will ensure secure access and version synchronization in the system repository. Additionally, control points will be defined to validate software integrity at each release.

7. Maintenance of the SCM Plan

The plan will be reviewed in each phase of the project to adapt to changing development needs. Any changes to this plan will be communicated to the team through meetings and updated documentation.