



Laurea Magistrale in informatica-Università di Salerno  
Corso di Gestione dei Progetti Software-Prof.ssa F. Ferrucci e Prof. F. Palomba

# Configuration Management Plan



<b>Riferimento</b>	CMP
<b>Versione</b>	3.0.0
<b>Data</b>	13/12/24
<b>Destinatario</b>	Team di sviluppo ZeroWaste Home

<b>Presentato da</b>	Raffaella Spagnuolo, Alessia Ture
----------------------	-----------------------------------

<b>Approvato da</b>	Prof.ssa Filomena Ferrucci, Prof. Fabio Palomba, Electrolux
---------------------	--



Laurea Magistrale in informatica-Università di Salerno  
Corso di Gestione dei Progetti Software-Prof.ssa F. Ferrucci e Prof. F. Palomba

## Revision History

Data	Versione	Descrizione	Autori
29/10/24	1.0.0	Prima Stesura	AT
20/11/24	2.0.0	Seconda Stesura	RS, AT
13/11/24	3.0.0	Revisione Finale	RS, AT

## Project Manager

Nome	Acronimo	Contatto
Raffaella Spagnuolo	RS	<a href="mailto:r.spagnuolo6@studenti.unisa.it">r.spagnuolo6@studenti.unisa.it</a>
Alessia Ture	AT	<a href="mailto:a.ture@studenti.unisa.it">a.ture@studenti.unisa.it</a>



## Sommario

Revision History .....	2
Project Manager .....	2
1. Introduzione .....	4
<b>1.1 Ambito.....</b>	<b>4</b>
<b>1.2 Scopo del documento.....</b>	<b>4</b>
2 Attività.....	5
<b>2.1 Configuration Identification .....</b>	<b>5</b>
<b>2.2 Configuration Item .....</b>	<b>5</b>
2.2.1 Configuration Management Database .....	6
<b>2.3 Configuration Version Release .....</b>	<b>6</b>
<b>2.4 Configuration Status Accounting .....</b>	<b>6</b>
<b>2.5 Configuration Audits .....</b>	<b>7</b>



# 1. Introduzione

---

## 1.1 Ambito

Il progetto *ZeroWaste Home* si concentra sulla gestione intelligente delle risorse alimentari domestiche per ridurre lo spreco alimentare e promuovere pratiche sostenibili. Attraverso il sistema, gli utenti possono monitorare in tempo reale il contenuto di frigoriferi e dispense, gestire le date di scadenza degli alimenti e ricevere suggerimenti per ricette basate sugli ingredienti in scadenza. Il sistema integra tecnologie avanzate come la scansione di codici a barre, dispositivi IoT per la gestione automatizzata degli alimenti e una community per la condivisione di ricette.

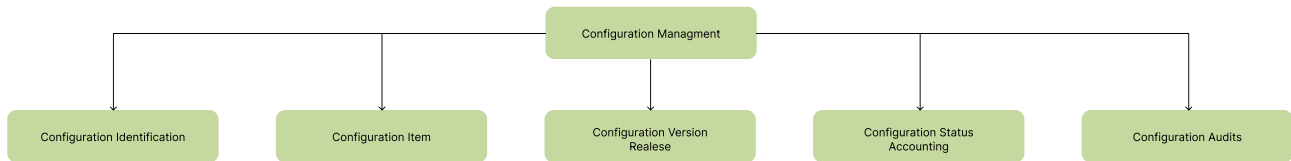
## 1.2 Scopo del documento

Il **Configuration Management Plan (CMP)** definisce gli standard per identificare, gestire e verificare i **Configuration Items (CI)** del progetto. Garantisce la coerenza e l'integrità del sistema durante l'intero ciclo di vita del progetto, minimizzando errori legati a modifiche non controllate.

Obiettivi principali:

- **Mantenere l'integrità del software** prevenendo discrepanze tra versioni e componenti.
- **Fornire un controllo** strutturato delle configurazioni per supportare sviluppo e manutenzione.
- Assicurare che ogni membro del team abbia **accesso a versioni aggiornate** degli artefatti.
- **Tracciare e gestire ogni modifica** per garantire coerenza tra codice sorgente, documentazione tecnica e altri artefatti.

## 2 Attività



### 2.1 Configuration Identification

Nel progetto *ZeroWaste Home*, il **Configuration Management** si applica sia alla documentazione che ai sorgenti del sistema. Una volta identificati i CI, i Project Managers assegneranno un nome univoco a ciascun item e lo inseriranno nello stato iniziale all'interno del **Configuration Management Database** (CMDB).

Successivamente, il Configuration Item sarà assegnato al team di sviluppo, che sarà responsabile delle attività operative sull'item, tenendo traccia di tutte le modifiche tramite un registro delle revisioni (**Revision History**). Ogni modifica dovrà essere documentata nel CMDB per assicurare che il processo di configurazione sia trasparente e tracciabile, garantendo una gestione efficace del ciclo di vita dei componenti.

### 2.2 Configuration Item

I **CI** rappresentano tutti gli elementi soggetti al controllo di configurazione. Nel progetto *ZeroWaste Home*, i principali CI includono:

- **Documentazione di progetto:** requisiti, piani di progetto, diagrammi UML e documentazione di test.
- **Codice sorgente:** gestito tramite GitHub, con sistema di versioning e pull request per tracciare e verificare ogni modifica.
- **Documentazione tecnica:** risorse necessarie per l'utilizzo e la manutenzione del sistema.

Ogni CI è identificato da un formato standard, C10\_[Nome\_Item]\_V[x.y], che ne facilita la gestione durante tutte le fasi del progetto.



### 2.2.1 Configuration Management Database

Il **CMDB** è il punto centrale per tracciare e gestire i CI. Garantisce che ogni elemento sia monitorato e aggiornato in tutte le fasi del progetto.

Strumenti utilizzati:

- **OneDrive**: Per la gestione e il versioning della documentazione, compresi requisiti e piani di test.
- **GitHub**: Per la gestione del codice sorgente e della documentazione tecnica. Ogni modifica al codice utilizza un sistema di branch con **pull request**, che consente una revisione collaborativa delle modifiche e garantisce l'approvazione prima dell'integrazione nel branch principale.

Questa combinazione di strumenti consente una gestione integrata e centralizzata di documentazione e codice, assicurando trasparenza e coerenza.

### 2.3 Configuration Version Release

Le nuove versioni dei CI vengono rilasciate in specifiche condizioni, tra cui:

- risoluzione di bug o problemi emersi durante il testing,
- modifiche o aggiornamenti minori ai documenti,
- completamento di documenti o artefatti con modifiche significative.

Ogni versione viene registrata nel CMDB e resa disponibile al team per garantire la tracciabilità e la sincronizzazione tra i componenti.

### 2.4 Configuration Status Accounting

Il **Configuration Status Accounting** è il processo di monitoraggio dello stato dei CI. Durante il progetto, i CI sono archiviati su **OneDrive** (documenti) e **GitHub** (codice). Il CMDB viene aggiornato a ogni milestone per riflettere lo stato corrente e assicurare che le versioni siano accessibili e coerenti con le modifiche approvate.

Alla consegna del sistema, tutta la documentazione sarà consolidata su GitHub per facilitarne la consultazione.



## 2.5 Configuration Audits

Gli **audit** verificano che i CI rispettino le specifiche e che le modifiche siano state tracciate correttamente.

Questi vengono eseguiti a ogni milestone o prima di una release significativa.

Prima degli audit, ai Team Members viene fornita una checklist per il controllo preliminare di documenti e artefatti, permettendo di identificare e risolvere errori o incongruenze.

Durante l'audit, i Project Managers verificano:

- numerazione corretta delle versioni,
- coerenza tra CI collegati
- qualità delle modifiche apportate,
- presenza e completezza di tutti i CI previsti,
- corretta organizzazione e aggiornamento del CMDB.

Questo processo assicura coerenza, qualità e tracciabilità lungo tutto il ciclo di vita del progetto.