



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



Configuration Management Plan

ReStart

| | |
|---------------|--------------------------------------|
| Riferimento | 2023_C08_CMP_V.1.0 |
| Versione | 1.0 |
| Data | 01/12/2023 |
| Destinatario | Ministero della Giustizia |
| Presentato da | Rebecca Di Matteo Leonardo Monaco |
| Approvato da | / |



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



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

Sommario

| | |
|---|---|
| Revision History | 4 |
| Team Members | 5 |
| 1. Introduzione | 6 |
| 1.1 Ambito..... | 6 |
| 1.2 Scopo del Documento..... | 6 |
| 1.3 Riferimenti ad altri documenti..... | 7 |
| 2. Management..... | 7 |
| 2.1 Fasi del progetto..... | 7 |
| 2.2 Organizzazione..... | 8 |
| 2.3 Ruoli e responsabilità..... | 8 |
| 3. Attività | 9 |



Revision History

| Data | Versione | Descrizione | Autori |
|------------|----------|----------------------|--------------------------------------|
| 01/12/2023 | 0.1 | Prima stesura | Rebecca Di Matteo Leonardo Monaco |
| 02/12/2023 | 0.2 | Seconda stesura | Rebecca Di Matteo Leonardo Monaco |
| 20/01/2024 | 1.0 | Revisione e Consegna | Rebecca Di Matteo Leonardo Monaco |



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

Team Members

| Ruolo | Nome e Cognome | Acronimo | Email |
|-------|--------------------|----------|--------------------------------|
| PM | Rebecca Di Matteo | RDM | r.dimatteo10@studenti.unisa.it |
| PM | Leonardo Monaco | LM | l.monaco11@studenti.unisa.it |
| TM | Gianfranco Barba | GB | g.barba14@studenti.unisa.it |
| TM | Francesco Corcione | FC | f.corcione5@studenti.unisa.it |
| TM | Giuseppe Di Palma | GDP | g.dipalma23@studenti.unisa.it |
| TM | Luigi Guida | LG | l.guida15@studenti.unisa.it |
| TM | Tullio Mansi | TM | t.mansi@studenti.unisa.it |
| TM | Matteo Panza | MP | m.panza13@studenti.unisa.it |
| TM | Serena Passiflora | SP | s.passiflora@studenti.unisa.it |



1. Introduzione

1.1 Ambito

Nell'era moderna, le aziende stanno assumendo un ruolo sempre più attivo nella costruzione di una società inclusiva e sostenibile. In risposta all'iniziativa del Ministero della Giustizia volta ad incentivare la responsabilità sociale d'impresa, l'Azienda specializzata da molti anni nel settore IT ha deciso di cogliere questa opportunità intraprendendo un ambizioso progetto: sviluppare un'applicazione innovativa dedicata alla reintegrazione degli ex detenuti nella società. L'applicazione proposta nasce dalla consapevolezza dell'importanza di fornire un sostegno significativo agli ex detenuti nel loro percorso di reintegrazione, non è solamente un prodotto tecnologico, è un impegno tangibile verso la diversità, l'inclusione e la responsabilità sociale. Attraverso questa iniziativa, l'Azienda mira a creare una piattaforma inclusiva che fornisca una serie di servizi essenziali agli ex detenuti, tra cui accesso a formazione, supporto psicologico, opportunità di lavoro e networking comunitario.

1.2 Scopo del Documento

L'obiettivo di questo documento è fornire un modello standard per identificare, controllare, mantenere e verificare le versioni di tutti i Configuration Items.

In particolare, gli scopi del CMP sono:

- mantenere l'integrità del prodotto software durante tutta la sua vita;
- supportare le attività di sviluppo e manutenzione;
- massimizzare la produttività riducendo gli errori derivanti dalla modifica degli artefatti prodotti.

In questo documento sono indicati:

- l'identificazione dei Configuration Items;
- il controllo del processo di cambiamento di un documento;
- la procedura da seguire nella richiesta di cambiamento;
- i responsabili del SCM.



1.3 Riferimenti ad altri documenti

Di seguito inseriamo un riferimento ad altri documenti di management:

- [Risk Management](#)

Di seguito inseriamo anche collegamenti ai due CMDB:

- [GitHub](#)
- [OneDrive \(solo account UNISA\)](#)

2. Management

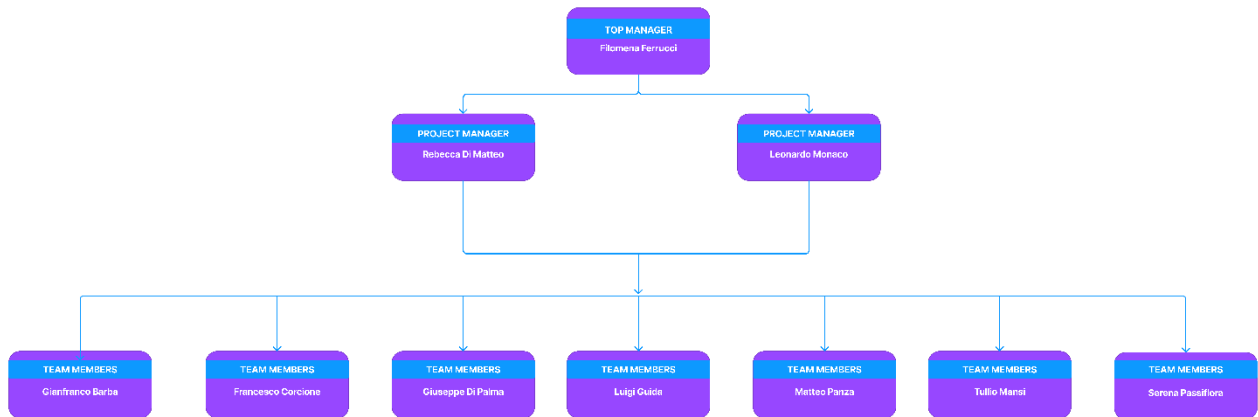
2.1 Fasi del progetto

Sono state identificate le seguenti principali fasi del progetto:

1. Avvio del progetto
2. Requirements Elicitation
3. Requirements Analysis
4. System Design
5. System Test Plan and Specification Design
6. Object Design
7. Source Code Implementation
8. Testing
9. Rilascio (creazione manuali d'installazione e d'uso)

2.2 Organizzazione

Di seguito è riportato un diagramma rappresentante l'organizzazione del progetto.



2.3 Ruoli e responsabilità

Di seguito verranno definiti i ruoli e le relative responsabilità nella gestione delle configurazioni.

La figura del **Top Manager** si occuperà dell'accettazione dei deliverable.

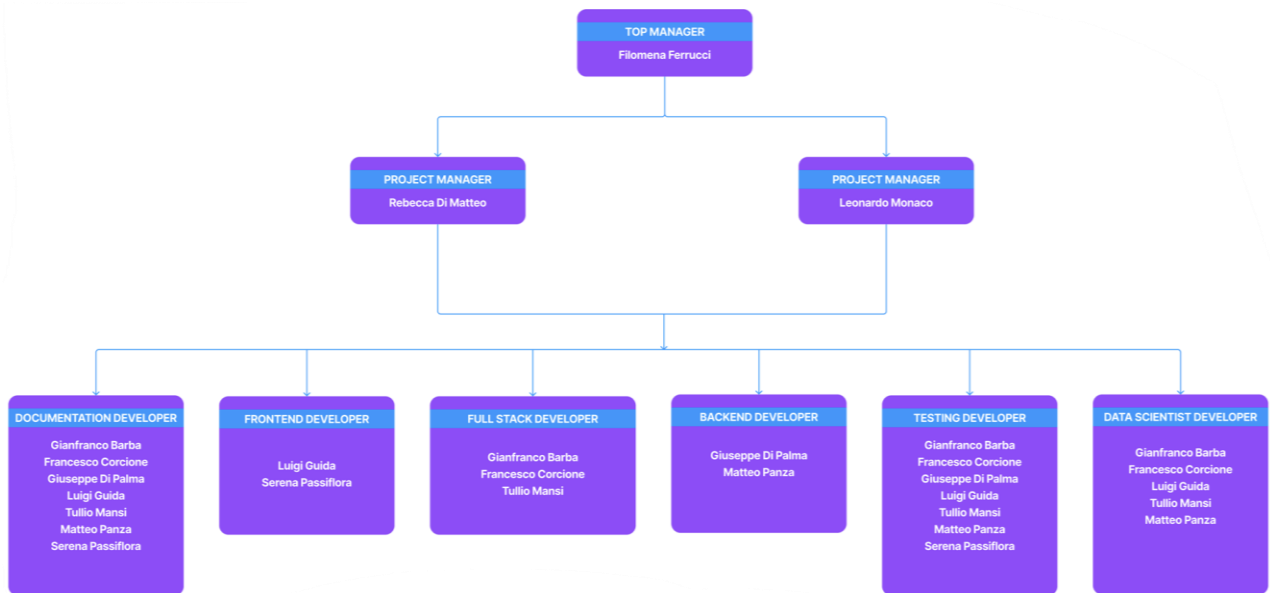
I **Project Managers** sono responsabili di tutta la gestione degli aspetti manageriali. In particolare, si occuperanno di:

- identificare i Configuration Items;
- accettare o rifiutare Change Request;
- assegnare delle responsabilità circa eventuali Change Requests.

Ogni Team Members riveste il ruolo di **Developer**, che è responsabile di:

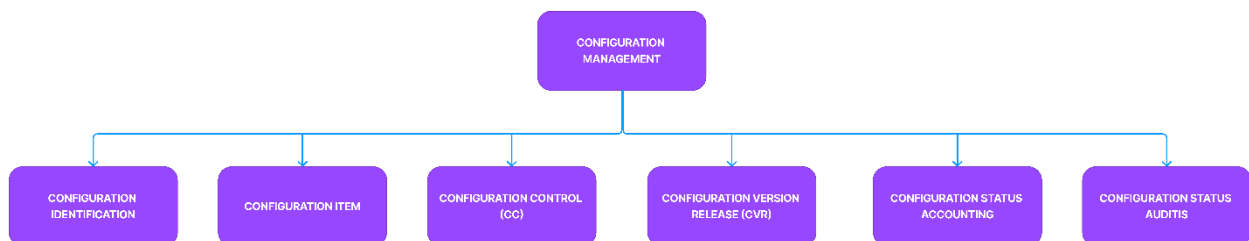
- compilare i documenti;
- implementare le change request per gli artefatti prodotti.

Inoltre, per ogni task assegnato, viene evidenziato un Developer che assume il ruolo di **Accurability**, ciò significa che prima di sottomettere il task in revisione ai Project Managers, viene effettuata una prima revisione.



3. Attività

Il Configuration Management consta delle seguenti attività:





3.1 Configuration Identification

Il Configuration Management si applica sia alla documentazione che ai sorgenti del progetto.

Una volta identificati i Configuration Item, i Project Managers, seguendo le direttive del Top Manager, assegneranno un nome all'item e lo inseriranno nel rispettivo CMDB in uno stato iniziale. Subito dopo, l'item sarà assegnato al team di sviluppo che dovrà lavorarci e tenere traccia delle modifiche attraverso un log con identificativo incrementale (Revision History), spesso situato nell'artefatto stesso.

3.2 Configuration Item

I Configuration Item rappresentano tutti i tipi di oggetti che sono coinvolti nel Configuration Control.

Gli elementi che entreranno a fare parte dei Configuration Item sono elencati di seguito:

- Documenti per la gestione e l'esecuzione del progetto.
- Documenti riguardanti lo sviluppo del sistema.
- Documenti di carattere tecnico del sistema.
- Applicativo software con relativa documentazione.
- Altri documenti a discrezione dei Project Managers.

Ogni Configuration Item è caratterizzato da:

- Un identificativo univoco solitamente della forma:
“2023_C08_[AcronimoDocumento]_V.[x.y]”
- Un numero di versione, della forma “x.y” dove x è usato per indicare la consegna o il raggiungimento di una milestone, mentre y per indicare modifiche di minore entità.

Al momento della sua individuazione e inserimento nel CMDB, all'item viene assegnata la versione 0.1 indicante lo stato iniziale. Successive modifiche vanno ad incrementare solamente la y della versione.



Alla prima release del documento o al raggiungimento di una milestones ad esso dedicata, la versione diventa 1.0. La x e la y verranno nuovamente incrementate solo in caso di nuove consegne, nuove milestones, o modifiche particolarmente importanti.

3.2.1 Configuration Management Database

Nell'ambito del progetto ReStart verranno utilizzati due diversi CMDB:

- **One Drive:** per la gestione dei documenti riguardanti il design dell'applicativo (RAD, SDD, ODD, Documenti di Testing, Manuali, ecc.).
- **GitHub:** per la gestione del codice sorgente del sistema.

Entrambi i sistemi offrono nativamente funzioni per il versioning degli artefatti. In particolare, per quanto concerne il CMDB One Drive, ogni documento ha un riferimento alla versione nel nome e, all'interno di esso, una tabella di Revision History, la quale consente di associare ad ogni versione le relative modifiche ed i rispettivi partecipanti. Invece, per quanto concerne la gestione del codice sorgente, ci si affiderà alle funzionalità offerte dal CMDB GitHub. Inoltre, per garantire una migliore comprensione delle modifiche al codice sorgente, si favoriscono commit di piccola taglia (poche modifiche) e si utilizzerà, per la stesura del messaggio di commit, il seguente standard ideato da Google <https://www.conventionalcommits.org/en/v1.0.0/>.

3.3 Configuration Control (CC)

Una volta che un Configuration Item rientra nella baseline, tutte le modifiche ad esso fatte devono seguire un preciso protocollo. Chiunque facente parte del team può sottomettere una Change Request. I Project Managers valuteranno se approvare o rifiutare la Change Request sulla base di diversi fattori quali: l'impatto dei cambiamenti sul progetto, se la modifica rientra o meno nell'ambito delle versioni correnti, priorità, risorse, livello di sforzo, rischio e qualsiasi altro criterio ritenuto rilevante dai Project Managers. Se approvata, l'implementazione della modifica verrà assegnata ad uno o più Developer, eventualmente lo stesso che l'ha proposta. A termine dell'implementazione il Developer si assicurerà



della sua corretta propagazione e consistenza. Per quanto riguarda gli accessi, l'intero team di sviluppo avrà privilegi completi sulla cartella One Drive, intesi come scrittura e lettura degli artefatti. Nel caso della repository GitHub invece, i Developers dovranno lavorare usando il sistema di branch offerto dalla piattaforma secondo le seguenti modalità: tutte le modifiche verranno apportate al contenuto del branch "dev", al termine di uno Sprint allora si provvederà ad inviare una pull request per effettuare il merge sul branch main. I Project Managers, una volta sicuri della qualità delle modifiche implementate, accetteranno la richiesta e integreranno i commit di modifica nel progetto effettivo.

3.3.1 Presentazione Change Request

La proposta di una Change Request può essere sottomessa ai project manager da ogni stakeholder rispettando la struttura dell'esempio fornito di seguito:

| Change Request Form | |
|---|----------------------------------|
| Project: ReStart | Number: 1 |
| Change Requester: Rebecca Di Matteo, Leonardo Monaco | Date: 02/12/2023 |
| Requested Change: Il Collaboratore Aziendale può contattare gli utenti che si candidano ad un annuncio di lavoro | |
| Request Type: Aggiunta di un nuovo requisito funzionale | |
| Change Analyzer: RDM, LM | Analysis date: 03/12/2023 |
| Oggetto: Aggiungere la possibilità al collaboratore aziendale di contattare all'interno dell'app gli utenti che si candidano ad un annuncio di lavoro. | |
| Descrizione: Il collaboratore aziendale dopo aver visualizzato il profilo di un utente che si è candidato ad un annuncio di lavoro deve poterlo contattare all'interno dell'app. | |
| Commenti: | Data di cambiamento: |



3.3.2 Valutazione Change Request

I Project Managers analizzeranno l'impatto del cambiamento in termini di modifiche da apportare, rischi associati e corrispettivo valore di business. A seconda della natura della CR, i Project Managers potrebbero accettare il cambiamento, rifiutarlo o richiedere un consulto col Top Management ed il Cliente.

3.3.3 Approvazione o rifiuto Change Request

La CR verrà accettata o rifiutata dopo aver effettuato delle analisi circa la stima per la realizzazione (in combinazione con lo stato attuale del progetto) e dall'eventuale accettazione da parte del Cliente. Non è stato previsto un tempo di decorrenza tra la sottomissione e l'accettazione/rifiuto.

3.4 Configuration Version Release (CVR)

L'attività di management delle release è svolta nel momento in cui si verificano condizioni che determinano il rilascio di una nuova release. Esse sono:

- Modifica o aggiunta di piccole parti di un documento, che porta all'incremento del valore .y nella versione di quest'ultimo;
- Raggiungimento di una milestone del progetto, che porta all'aumento del valore .x nella versione di quest'ultimo.



3.5 Configuration Status Accounting

Durante il tempo che intercorre tra due consegne o milestones, i CI sono conservati nella cartella One Drive o su GitHub, nelle quali avvengono le principali attività di modifica. L'accounting verrà effettuato ad ogni milestone importante del progetto.

3.6 Configuration Audits

Al raggiungimento di una milestone, o poco prima di una consegna, saranno effettuati dei lavori di revisione da parte di tutto il team di sviluppo su ogni Configuration Items facente parte della baseline. Dopo tali lavori, i Project Managers visioneranno personalmente gli items assicurando:

- la corretta numerazione delle versioni;
- la consistenza delle modifiche tra items collegati;
- la qualità delle descrizioni delle modifiche implementate;
- la presenza di tutti gli items;
- la corretta organizzazione dei CMDB.