



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



Configuration Management Plan Progetto TalkAid

Riferimento	
Versione	1.0
Data	20/01/2024
Destinatario	Prof. Ferrucci Filomena
Presentato da	C. Pastore, N. Laurino
Approvato da	



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

Revision History

Data	Versione	Descrizione	Autori
18/12/2023	0.1	Prima stesura	C. Pastore N. Laurino
20/01/2024	1.0	Revisione pre-consegna	C. Pastore N. Laurino



Sommario

Revision History	2
1. Introduzione.....	4
1.1 Ambito	4
2. Management.....	4
2.1 Fasi del progetto.....	4
2.2 Organizzazione (OBS).....	5
2.3 Ruoli e responsabilità	5
3. Attività	6
3.1 Configuration Identification	6
3.2 Configuration Item	6
3.3 Configuration Control.....	7
3.4 Configuration Version Release (CVR)	8
3.5 Configuration Status Accounting.....	8
3.6 Configuration Audit	8

1. Introduzione

1.1 Ambito

Il progetto TalkAId rappresenta un'iniziativa innovativa volta a migliorare la vita delle persone affette da disturbi del linguaggio, della voce e della pronuncia attraverso l'impiego di tecnologie avanzate di riconoscimento vocale e di terapia logopedica. TalkAId offrirà una soluzione altamente personalizzata che consentirà agli utenti di migliorare le proprie abilità di comunicazione in modo efficiente permettendo di ridurre in modo considerevole i tempi di cura.

Le tecnologie di riconoscimento vocale avanzate offrono un'enorme opportunità per migliorare la terapia logopedica. TalkAId sfrutta questo potenziale per offrire terapie altamente personalizzate ed efficaci.

TalkAId non sostituisce i logopedisti, ma li supporta nell'offrire terapie più adeguate. Questo rappresenta un valore aggiunto sia per i professionisti della salute che per i pazienti.

1.2 Scopo del documento

Per gestire al meglio il progresso del nostro progetto è necessario produrre il Configuration Management Plan, andremo a trattare il configuration management, configuration item e le modifiche nel corso dello sviluppo del progetto.

2. Management

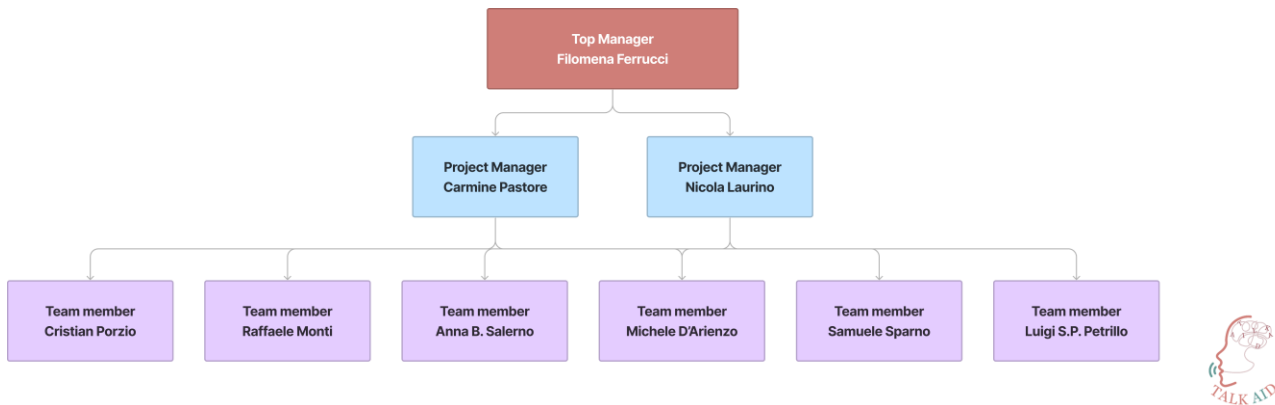
2.1 Fasi del progetto

Sono state definite le fasi del progetto così come segue:

1. Operazioni preliminari
2. Requirements Elicitation and Analysis
3. System Design
4. System Testing Design
5. Object Design
6. Implementazione
7. Testing
8. Release

Durante queste fasi, i membri del team saranno costantemente supportati dai PM.

2.2 Organizzazione (OBS)



2.3 Ruoli e responsabilità

Sono stati definiti i seguenti ruoli:

Project Manager

È la figura che si occupa di gestire tutti gli aspetti manageriale del progetto.

Il Project Manger è responsabile per:

- Identificazione dei Configuration Items
- Accettazione di change request
- Assegnazione delle responsabilità delle change request

Revision Leader

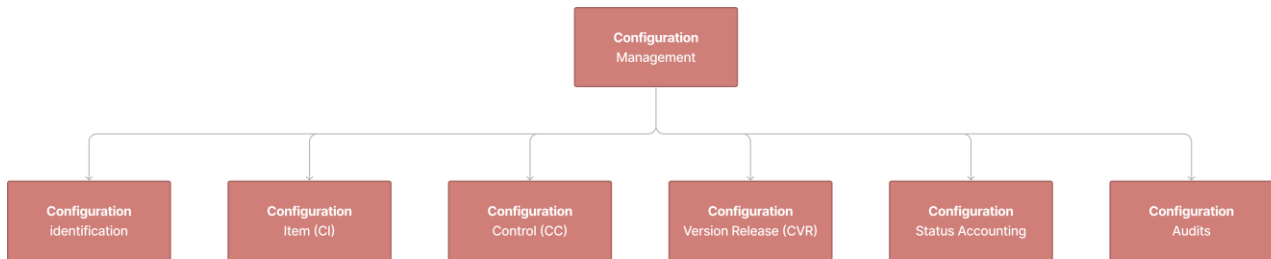
Questa figura è un membro del team che ha maggiori responsabilità rispetto ai suoi colleghi e si occupa di eseguire revisioni periodiche degli artefatti prodotti e comunica ai PM eventuali modifiche da apportare agli artefatti.

Developer

Questo ruolo è ricoperto da tutti i membri del team, ed è responsabile di eseguire e implementare le richieste dei PM.

3. Attività

Il Configuration Management è formato dalle seguenti attività:



3.1 Configuration Identification

I PM avranno il compito di identificare i Configuration Item, seguendo le direttive del Top Manager, assegneranno un nome all'item e lo inseriranno nel rispettivo CMDB in uno stato iniziale. Successivamente l'item sarà assegnato al team che sarà tenuto a lavorarci e tenere traccia di ogni modifica attraverso un log, con identificativo incrementale, presente nell'artefatto stesso.

3.2 Configuration Item

Appartengono a questa categoria:

- Documenti per la gestione ed esecuzione del progetto
- Documenti riguardo lo sviluppo del sistema
- Documenti tecnici del sistema
- Applicativo finale con relativa documentazione
- Altri documenti a discrezione del PM

Ogni Configuration Item è caratterizzato da:

- Identificativo univoco, espresso come segue “2023_C16_ [Acronimo del documento]_Ver[x.y]”
- Un numero di versione, della forma x.y.

Al momento della sua identificazione al Configuration Item viene assegnata la versione 0.1, successivamente viene incrementata solo la y, fino alla prima release del documento dove avremo la versione 1.0, e successivamente inizieremo di nuovo ad incrementare la y fino alla release successiva.

Configuration Management Database (CMDB)

Sono stati individuati due diversi CMDB:

- Google Drive: gestione dei documenti riguardanti il design dell'applicativo
- GitHub: gestione del codice sorgente dell'applicativo

Per il versioning, Github offre funzioni native per permettere ciò, mentre per quanto riguarda il versioning dei documenti è prevista all'interno di ogni artefatto una tabella "Revision History", la quale va compilata ogni volta che si mette mano al documento.

3.3 Configuration Control

Una volta che un CI rientra nella baseline, tutte le modifiche apportate ad esso devono seguire un preciso protocollo. Tutti i membri del team possono presentare una Change Request, al Revision Leader, che ha il compito di effettuare una prima valutazione. Quest'ultimo ha il compito di passare la Change request ai PM, che la valuteranno e decideranno se approvarla, in base all'impatto sul progetto, all'importanza della richiesta e ai rischi collegati alla richiesta avanzata. Se approvata viene assegnata a uno o più Developers per l'implementazione, ed una volta implementata sarà compito del developer assicurarsi la corretta propagazione e consistenza. Nel caso di modifiche al codice, il developer dovrà assicurare il corretto build e pass dei test dopo aver implementato la change request. Per quanto riguarda la repository GitHub, i developers dovranno utilizzare il sistema di branch offerto dalla piattaforma nel seguente modo:

Viene creata una nuova branch per ogni funzionalità sviluppata, una volta che la funzionalità è stata sviluppata del tutto allora si potrà effettuare il merge nel branch "*development*", e una volta testato tutto sarà possibile sottomettere la pull request per effettuare il merge sul branch *main*.

3.4 Configuration Version Release (CVR)

Il management delle release viene svolto seguendo le linee guida sottostanti:

- Effettuiamo un incremento della y in caso di risoluzione di bug all'interno del codice sorgente
- Effettuiamo un incremento della y in caso di modifica di documenti a causa di inconsistenza o inserimento paragrafi
- Effettuiamo un incremento della X in caso di aggiunta di funzionalità significative o risoluzione bug notevoli all'interno del codice sorgente
- Effettuiamo un incremento della X in caso di release o completamento con aggiunta di numerose sezioni.

3.5 Configuration Status Accounting

Prima della consegna i CI, sono conservato su Google Drive o su GitHub, dove è possibile eseguire operazioni di modifica.

L'accounting dello stato della configurazione avverrà ogni milestones importante o release degli artefatti prodotti. Il PM dovrà creare una cartella contenente tutte le versioni più recenti, degli artefatti richiesti e se necessario aggiungere le versioni precedenti per motivi didattici. All'interno della cartella dovrà essere presente un file "readme" per informare chi visiona gli artefatti.

3.6 Configuration Audit

Al raggiungimento di una milestone, o poco prima di una release, saranno effettuati dei lavori di revisione da parte del team su ogni CI facente parte della baseline. Successivamente sarà compito dei PM revisionare gli item assicurandosi che:

- Corretta numerazione delle versioni
- Consistenza delle modifiche tra items collegati
- Qualità delle descrizioni delle modifiche implementate
- Presenza di tutti gli items
- Corretta organizzazione dei CMDB

In caso di necessità i PM potranno richiedere il rollback di alcune modifiche.