

DOCUMENTO INTERNO



CARBON7TEAM

carbon7team@gmail.com

11 Gennaio 2022

Organizzazione github: [Carbon7team](#)

Norme Di Progetto

v0.2.6

Redattori

Matteo Noro
Andrea Polato

Revisori

Filippo Brugnolaro
Adnan Latif Gazi

Sommario

Documento Interno relativo alle Norme di Progetto_G
del Carbon7team

Storico modifiche al documento

Legenda:

- +: Prima redazione di contenuto
- #: Estensione di contenuto
- [n]: Sezione n del documento

Versione	Operazione	Autore	Data
0.2.6	# Processi Primari [2.2]	Andrea Polato	11/01/2022
0.2.5	# Processi Primari [2.2] + Sistema di Qualità [5]	Andrea Polato	06/01/2022
0.2.4	# Processi di Supporto [3.1] + Introduzione [1]	Matteo Noro	22/12/2021
0.2.3	#Processi Primari [2.1],[2.2] # Processi di Supporto [3.2]	Matteo Noro	13/12/2021
0.2.2	+ Processi Primari [2] # Processi Organizzativi [4.3]	Matteo Noro	29/11/2021
0.2.1	# Processi di Supporto [3.1] # Processi Organizzativi [4.2]	Matteo Noro	26/11/2021
0.2.0	Revisione Documento	Adnan Latif Gazi	25/11/2021
0.1.2	# Processi di Supporto [3.1]	Andrea Polato	24/11/2021
0.1.1	# Processi Organizzativi [4.1],[4.2] # Processi di Supporto [3.2]	Andrea Polato	23/11/2021
0.1.0	Revisione Documento	Filippo Brugnolaro	22/11/2021
0.0.4	# Processi di Supporto [3.1]	Matteo Noro	21/11/2021

Versione	Operazione	Autore	Data
0.0.3	+ Processi Organizzativi [4]	Matteo Noro	21/11/2021
0.0.2	+ Processi di Supporto [3]	Matteo Noro	20/11/2021
0.0.1	Generazione Documento	Matteo Noro	17/11/2021

Indice

1	Introduzione	5
1.1	Scopo del documento	5
1.2	Glossario	5
1.3	Riferimenti	5
1.3.1	Riferimenti normativi	5
1.3.2	Riferimenti informativi	5
2	Processi Primari	7
2.1	Piano di progetto _G	7
2.2	Piano di qualifica _G	7
2.2.1	Descrizione delle metriche _G	8
2.3	Analisi dei Requisiti _G	8
2.3.1	Scopo	8
2.3.2	Casi d'uso _G	9
2.3.3	Requisiti	9
3	Processi di Supporto	10
3.1	Documenti	10
3.1.1	Versionamento _G	10
3.1.2	Classificazione dei documenti	11
3.1.3	Gestione dei documenti nelle Repository _G	12
3.1.4	Struttura generica cartelle	12
3.1.5	Struttura generica documento	13
3.1.6	Verbalì	13
3.1.7	Glossario	14
3.2	Norme Tipografiche	14
3.2.1	Convenzioni per la denominazione e la siglatura	14
3.2.2	Formattazione del testo	14
3.2.3	Elementi testuali	15
3.2.4	Tabelle	15
4	Processi Organizzativi	15
4.1	Comunicazione interna ed esterna	15
4.1.1	Strumenti di comunicazione interna	15
4.1.2	Strumenti di comunicazione esterna	15
4.2	Organizzazione del lavoro	16
4.2.1	Incontri	16
4.3	Ruoli di Progetto	16
4.3.1	Assegnazione Ruoli	18
5	Sistema di Qualità	19
5.1	Obiettivi	19
5.2	Descrizione	19
5.3	Spiegazione metriche _G - Qualità di processo	19
5.4	Spiegazione metriche _G - Qualità di prodotto	20

1 Introduzione

1.1 Scopo del documento

Lo scopo di tale documento gestionale Interno è quello di definire procedure, strumenti e metriche di qualità con la funzione di normare il Team.

Tale documento si pone quindi come un "manuale di comportamento", da seguire in modo preciso e rigoroso.

La sua redazione avviene in maniera *Just-In-Time*, per assicurare che il prossimo futuro sia normato. Bisogna quindi ricercare un miglioramento continuo al fine di garantire la qualità desiderata.

Ogni norma sarà descritta e inserita nel documento previa approvazione dei membri del Team, i quali sono tenuti a restare aggiornati con le future modifiche che saranno apportate al documento, in modo da svolgere i compiti loro assegnati in modo coerente come previsto.

Questo definisce il *Way Of Working_G* del Team.

1.2 Glossario

Per assicurare la massima trasparenza e fruibilità del documento, il *Carbon7team* ha deciso di stilare il *Glossario*. Qui verranno inseriti tutti i termini ambigui o relativi all'attività del progetto, che il gruppo individua come degni di nota. I termini qui presenti saranno identificati attraverso una 'G' a pedice.

1.3 Riferimenti

1.3.1 Riferimenti normativi

- **Documenti Carbon7team**

- Verbale_interno_carbon7team_13NOV21
- Verbale_interno_carbon7team_24NOV21
- Verbale_interno_carbon7team_17DIC21

1.3.2 Riferimenti informativi

- **Materiale didattico del corso di Ingegneria del Software**

- Gestione di progetto
- Regolamento del Progetto Didattico
- Analisi dei Requisiti_G
- Diagrammi Use Case_G
- Libro di testo: Iam Sommerville, Software Engineering
Part 1: Introduction to Software Engineering; Chapter 4: Requirements

- Documentazione esterna

—

2 Processi Primari

2.1 Piano di progetto_G

Deve essere redatto un piano di progetto_G da seguire e produrre *Just in Time*_G durante lo svolgimento del progetto stesso.

Tale documento deve contenere:

- **Analisi dei Rischi:** dove vengono analizzati i rischi che potrebbero presentarsi nel corso del progetto e vengono preventivate delle contromisure per mitigare gli stessi. Tali rischi vengono inoltre codificati con un identificativo univoco e un livello di pericolo in una scala da 1 a 5;
- **Pianificazione di Progetto:** dove si descrivono nel dettaglio i periodi in cui è stato preventivamente suddiviso il lavoro. La descrizione avverrà dunque per fasi di sviluppo;
- **Preventivo e consuntivo:** dove viene fornita, in due sezioni separate, una stima del lavoro necessario per ciascuna fase, ottenendo prima un preventivo per il costo totale e fornendo poi un consuntivo di periodo relativo all'andamento della fase secondo quanto preventivato.

Descrizione dei rischi Il codice identificativo dei rischi è descritto come segue:

R[Tipologia][NumeroBase]

dove:

- **Tipologia:** che descrive se un rischio è di tipo:
 - Tecnologico [T];
 - Organizzativo [O];
- **NumeroBase:** numero progressivo che identifica il rischio in base alla tipologia.

2.2 Piano di qualifica_G

Il documento ha lo scopo di illustrare in che modo si intende perseguire la qualità all'interno del progetto. Il documento deve essere composto da quattro parti fondamentali:

- Qualità di Processo;
- Qualità di Prodotto;
- Suite di Test_G;
- Resoconto e Valutazione della Verifica.

Le due sezioni di Qualità di Processo e Qualità di Prodotto devono a loro volta essere suddivise in più processi, per quanto riguarda la prima sezione, e in più aspetti, per la seconda. Ognuna di queste divisioni è a sua volta così caratterizzata:

- Nome del processo/aspetto;
- Descrizione;
- Elenco delle metriche_G utili alla misurazione;
- Tabella con valori obiettivo.

La sezione di *Resoconto e Valutazione della Verifica* deve essere composta prevalentemente da grafici che dimostrino immediatamente e chiaramente i valori ottenuti.

2.2.1 Descrizione delle metriche_G

Le metriche_G sono descritte come segue:

$$M[\text{Tipologia}][\text{NumeroBase}]$$

dove:

- **Tipologia**: descrive se la metrica_G si riferisce a un:
 - Processo [PS];
 - Aspetto del prodotto [PT];
- **NumeroBase**: numero progressivo che identifica la metrica_G in base alla tipologia.

Inoltre, i valori obiettivo delle metriche_G devono essere rappresentati in forma tabellare come segue:

Codice Metrica	Valore Accettabile	Valore Ottimale
----------------	--------------------	-----------------

2.3 Analisi dei Requisiti_G

2.3.1 Scopo

Il processo di analisi dei requisiti_G è quello di individuare le necessità del proponente_G, convertendole in requisiti. Tali requisiti possono essere impliciti o espliciti. Il documento di analisi dei requisiti_G è redatto dagli analisti, i quali devono:

- definire lo scopo del prodotto che si andrà a realizzare;
- definire le funzionalità e i requisiti concordati con il proponente_G;
- fornire ai progettisti una base solida da dove partire per applicare le tecnologie scelte;
- fornire ai verificatori i riferimenti per il processo di verifica;
- fornire una stima oraria del lavoro per definire una stima dei costi.

Le informazioni sopra vengono tratte dallo studio del capitolato d'appalto, da incontri interni al Team e con il proponente_G.

Il documento di Analisi dei Requisiti_G dovrà essere quindi sviluppato secondo tali sezioni:

- Scopo del Documento;
- Scopo prodotto;
- Attori;
- Casi d'uso_G;
- Requisiti:
 - Requisiti funzionali;
 - Requisiti di qualità;
 - Requisiti di vincolo;
 - Requisiti prestazionali;

2.3.2 Casi d'uso_G

Descrizione dei casi d'uso_G I casi d'uso_G dovranno essere descritti secondo i seguenti punti:

- Immagine * ;
- Attore primario;
- Attore secondario * ;
- Precondizione;
- Postcondizione;
- Scenario principale;
- Inclusioni * ;
- Estensioni * .

Nota: * indica un punto che non è sempre richiesto e pertanto non presente per ogni caso d'uso_G.

Classificazione dei casi d'uso_G Ogni caso d'uso_G è identificato obbligatoriamente da un codice identificativo univoco.

Il codice identificativo è descritto come segue:

UC[NumeroBase].[NumeroSottoCaso]

dove:

- **NumeroBase**: numero che identifica il caso d'uso_G generico;
- **NumeroSottoCaso**: numero progressivo che identifica i sotto casi a partire dal caso base. Può a sua volta diramarsi in ulteriori sottocasi.

2.3.3 Requisiti

Descrizione dei requisiti Ogni requisito dovrà avere un codice identificativo univoco, corredato da una breve descrizione. Ogni requisito dovrà inoltre palesare i casi d'uso_G di riferimento.

I requisiti dovranno essere descritti in modo tabellare, nella seguente forma:

ID Requisito | Descrizione | Fonti

Pertanto ogni requisito sarà corredato di tali informazioni aggiuntive:

- **Descrizione**: descrizione breve e concisa del requisito;
- **Fonte**: specifica la fonte da cui deriva il requisito:
 - capitolato;
 - verbale interno;
 - verbale esterno;
 - caso d'uso_G.

Classificazione dei requisiti Ogni requisito è identificato obbligatoriamente da un codice identificativo univoco.

Il codice identificativo è descritto come segue:

R[Importanza][Tipologia][Codice]

- **Importanza** : indica l'importanza di tale requisito attraverso i seguenti valori:
 - 1: requisito obbligatorio;
 - 2: requisito desiderabile;
 - 3: requisito opzionale.
- **Tipologia**
 - V: requisito di vincolo;
 - F: requisito funzionale;
 - P: requisito prestazionale;
 - Q: requisito di qualità.
- **Codice**: identificatore univoco in forma gerarchica: Requisito Base - Sotto Requisito.

[NumeroReqBase][NumeroSottoReq]

dove:

- **NumeroReqBase**: numero che identifica il requisito base
- **NumeroSottoReq**: numero progressivo che identifica i sotto requisiti a partire dal caso base. Può a sua volta diramarsi in ulteriori sotto requisiti.

3 Processi di Supporto

3.1 Documenti

In questo capitolo verranno illustrate le regole e gli standard_G che normano la documentazione. Tali regole permetteranno di mantenere uno stile grafico e organizzativo coerente.

3.1.1 Versionamento_G

Ogni documento è una successione di date fasi di vita, quali:

1. **Pianificazione**

Si identificano le necessità del documento, per delinearne il contenuto fondamentale;

2. **Redazione**

Viene costruito lo scheletro del documento e messo per esteso da un redattore_G/autore. La prima stesura cerca di coprire tutte le necessità richieste, fornendo un contenuto quanto più completo;

3. **Revisione**

Il documento previa redazione, viene revisionato sul lato grammaticale e contenutistico. In caso di errori, questi verranno corretti in una nuova versione del documento;

4. Approvazione

Il documento, valutato dal Responsabile, soddisfa quanto richiesto nella fase corrente: assunto come revisionato, esso è corretto in forma e contenuto ed è pronto per esser valutato dal committente_G.

Ogni documento, esclusi i verbali, è soggetto a versionamento_G, rappresentato nella forma $x.y.z$. Il versionamento_G è definito come segue:

- Numero z
Tale numero verrà aggiornato ogni qual volta verrà aggiunto, o eventualmente rimosso, contenuto al documento;
- Numero y
Tale numero verrà aggiornato ogni qual volta verrà effettuata una revisione del documento;
- Numero x
Tale numero verrà aggiornato ogni qual volta verrà effettuata un'approvazione del documento: in questo modo il documento sarà pronto per essere oggetto di valutazione.

La versione di ogni documento partirà, pertanto, dalla 0.0.1.

Dopo la revisione z torna a 0 e dopo l'approvazione y e z tornano a 0.

3.1.2 Classificazione dei documenti

I documenti prodotti dal team devono essere classificati come segue:

- **Formali:** Documenti che richiedono la verifica di un revisore e l'approvazione da parte del responsabile di progetto;
- **Informali:** Documenti che permettono ai membri del gruppo di condividere e tener nota delle informazioni circa le decisioni prese.

I documenti formali e informali sono a loro volta suddivisi in:

- **Interni**
- **Esterni**

Documenti formali: I documenti formali interni sono tutti quelli che interessano i membri del Team e che li aiutano nelle scelte e nella programmazione della redazione dei documenti successivi. Tali documenti saranno:

- **Norme di Progetto_G:** si tratta del seguente documento, esso contiene tutte le norme che i membri del team devono seguire.
- **Glossario dei termini:** contiene un elenco di tutti i termini che ricorrono nei documenti e che necessitano di una definizione esplicita. Nel glossario sono inseriti inoltre termini di carattere generale riguardanti il progetto, che potrebbero ricorrere sia nei documenti che nel parlato.

Quelli esterni invece sono di interesse per committente_G e proponente_G, infatti vengono consegnati loro nell'ultima versione, quindi previa approvazione. Tali documenti saranno:

- **Piano di progetto_G**: pianificazione delle attività che il team dovrà svolgere, indicando come avverrà l'utilizzo delle risorse;
- **Piano di qualifica_G**: contiene tutti gli standard_G e le metriche_G da utilizzare per la valutazione della qualità;
- **Analisi dei Requisiti_G**: vengono descritti tutti i requisiti che il prodotto finale dovrà soddisfare.
- **Candidatura***: contiene la proposta di candidatura al capitolato scelto dal Team, con motivazioni e impegni;

Nota : * indica che il documento rappresenta un'eccezione in quanto non soggetto a versionamento_G

Documenti informali: I documenti che appartengono a questa tipologia sono i **verbali**. Essi non sono soggetti a versionamento_G in quanto non vengono modificati nel tempo.

I verbali **interni** sono di interesse per i componenti del team e aiutano a ricapitolare le decisioni prese durante i meeting interni.

Quelli **esterni** invece riguardano i meeting con committente_G e proponente_G.

3.1.3 Gestione dei documenti nelle Repository_G

L'organizzazione GitHub_G avrà a disposizione due repository_G distinte per la gestione dei documenti: una pubblica, accessibile da utenti esterni, e una privata accessibile solo dal Team.

Di seguito vengono riportati nomi e contenuti delle repo_G sopracitate:

- **Docs/...** Per documentazione pubblica, solo file **.pdf** e complementari;
- **Docs-Private/...** Per documentazione completa di tutti i file sorgenti L^AT_EX_G, accessibile solo dal Team.

Ogni documento stilato dovrà essere presente nella repository_G **Docs-Private**.

Al contrario, soltanto il singolo PDF aggiornato all'ultima versione dovrà essere presente in **Docs**, sempre rispettando le regole di caricamento e nomenclatura che verranno riportate di seguito.

I redattori, al fine di mantenere la documentazione più aggiornata possibile sulla Repository_G, devono aver cura di caricare ogni nuova stesura, con annessa versione.

I revisori possono correggere gli errori minori o grammaticali direttamente nel file L^AT_EX_G, mentre gli errori più consistenti vengono segnalati nel file L^AT_EX_G sotto forma di commento e vengono corretti dal redattore_G a seguito di discussione con il team.

3.1.4 Struttura generica cartelle

La cartella contenente i file dei singoli documenti dovrà essere strutturata come segue:

- Nome cartella: <nomeDocumento>;
- Contenuto cartella:
 - file **main.tex**;
 - file **main.pdf** (rinominabile nella repository_G pubblica);
 - cartella **res**, cui contenuto è composto da:
 - * cartella con il logo del Team ed eventuali immagini;

- * cartella con varie ed eventuali *
- * copertina nella forma: `copertina_<nomeDocumento>.tex`;
- * storico delle versioni nella forma: `storicoModifiche_<nomeDocumento>.tex`;
- * file di contenuto nella forma: `<sezione/contenuto>_<nomeDocumento>.tex`,

Dove la parte `<sezione/contenuto>` sta a indicare, appunto, a quale sezione o contenuto del documento fa riferimento.

Note:

- * indica che la cartella non è sempre presente in quanto non sempre strettamente necessaria.
- Si tratta di una prassi fondamentale **non** rinominare `main.pdf` all'interno della repository_G privata, al fine di evitare la duplicazione futile del file in fase di compilazione.

3.1.5 Struttura generica documento

In prima pagina si trova sempre **la copertina**, che deve avere i seguenti elementi:

1. Tipologia documento e sua destinazione: verbale/documento, interno/esterno;
2. Logo del Carbon7team;
3. Info di contatto del Team;
4. Data ultimo aggiornamento;
5. Organizzazione github_G del Team;
6. Titolo del documento;
7. Versione attuale (se soggetto a versionamento_G);
8. Elenco degli autori e revisori;
9. Sommario.

Unica eccezione alla regola è il documento di Candidatura, dove a seguito del titolo del documento, non vi saranno né la versione attuale, in quanto non soggetta a versionamento_G, né l'Elenco degli autori e revisori. Vi saranno invece i componenti del gruppo e il committente_G.

In seconda pagina vi sarà l'indice dei contenuti e, se il documento è soggetto a versionamento_G, nella pagina seguente deve essere presente uno **Storico delle modifiche al documento** (scrittura, revisione, accettazione) strutturata come segue:
autore | operazione svolta | versione | data.

3.1.6 Verballi

I verballi, sia in forma interna che esterna, non sono soggetti al versionamento_G, in quanto non vengono modificati nel tempo. Non presentano quindi la versione attuale e i revisori. Non sarà nemmeno presente l'indice, per via delle dimensioni e della struttura del documento.

Quest'ultimi dovranno comunque rispettare sempre le regole strutturali e di nomenclatura già citate, con la convenzione che il sommario consiste nella lista dei membri presenti e assenti al meeting d'interesse.

Ogni verbale presenta il suo **contenuto** nella forma:

1. Ordine del giorno;
2. Contenuto, con struttura questioni - conclusioni;
3. Considerazioni finali.

3.1.7 Glossario

Il glossario dei termini, data la natura stessa del documento, avrà una struttura uguale alla documentazione prima descritta per quanto riguarda copertina e storico modifiche, ma differirà nel formato del contenuto.

Tale documento per la definizione di un termine ha all'interno un comando apposito invocabile con `\entry{<Termine>}{<Ricorrenza nei documenti>}{<Significato>}`.

Per semplicità, i documenti riportati nelle loro ricorrenza andranno trascritti tramite apposita sigla.

3.2 Norme Tipografiche

Le norme tipografiche hanno il fine di normare la stesura tipografica dei documenti, al fine di mantenere consistenza nella scrittura e coesione fra i documenti.

3.2.1 Convenzioni per la denominazione e la siglatura

I vari documenti dovranno essere nominati e citati sempre con la prima lettera in maiuscolo, eccezion fatta per le preposizioni.

Tale convenzione dovrà essere rispecchiata anche nella siglatura di questi, i quali sono:

- Analisi dei Requisiti_G: AdR_G;
- Piano di progetto_G: PdP_G;
- Piano di qualifica_G: PdQ_G;
- Norme di Progetto_G: NdP_G.

Tale convenzione di siglatura dovrà essere seguita anche in tutti quei prodotti che la richiedono (per esempio PoC_G per il Proof of Concept_G).

3.2.2 Formattazione del testo

Il testo dovrà essere formattato in modo differente in base a quello che vuole rappresentare.

Dove il testo non formattato è privo di particolare significato, le varie formattazioni per il testo dovranno essere:

- **Bold**: per tutti i titoli, sottotitoli, sezioni e varie sottosezioni. Viene utilizzato anche per tutte quelle componenti testuali che richiedono maggiore enfasi;
- **Monospace**: per i blocchi di codice e per tutte quelle parti di testo che vogliono rimandare a cartelle, nomi di file, estensioni e affini. Le parti di codice variabile dovranno essere inserite fra parentesi angolari, quali: < >;
- *Italic*: per nomi propri, tecnologie, citazioni e terminologie tecniche.

3.2.3 Elementi testuali

I documenti redatti per loro stessa forma sintattica presentano molti elenchi testuali che necessitano di essere normati per mantenere la coerenza tipografica citata nello scopo della sottosezione.

Gli elenchi, che essi siano numerati o meno, dovranno presentare un ";" alla fine di ogni item, a meno che non introduca un sotto-elenco con ":". Solo nell'ultimo item dovrà vi dovrà essere un "." o una "," in base al contesto.

Il titolo di ogni item, se presente, dovrà essere in **bold**. Per eventuali sotto-elenchi, le modalità rimangono uguali. Un esempio è il seguente:

- **Item 1** Contenuto 1;
 - **Item 1.1** Contenuto 1.1;
- **Item 2** Contenuto 2;
- **Item 3** Contenuto 3.

3.2.4 Tabelle

Le tabelle possono variare nello stile in base alla loro locazione. È fondamentale che queste siano fruibili in termini di colore, formattazione, grafica e dimensioni nelle colonne.

Ogni redattore_G dovrà aver particolare cura nel scegliere la quantità di contenuto da inserire ad ogni record, prediligendo uno stile breve e conciso.

4 Processi Organizzativi

In questa sezione verranno illustrate le attività e i compiti generici che vengono impiegati per gestire i vari processi. Il Responsabile di progetto è responsabile della gestione del prodotto, del progetto, e delle attività o processi ad esso applicabili.

4.1 Comunicazione interna ed esterna

Vengono illustrati di seguito le modalità di comunicazione interne ed esterne al gruppo.

4.1.1 Strumenti di comunicazione interna

Si fa obbligo di utilizzare i seguenti strumenti di comunicazione in base alle necessità riportate:

- Comunicazione generica, discussioni: **Telegram_G**, **Discord_G**
- Annunci importanti: **WhatsApp**
Struttura annuncio: #<argomento>, a capo, <testo annuncio>.
L'importanza dell'argomento è quella di poter recuperare eventuali commenti relativi ad esso in altri canali di comunicazione.

4.1.2 Strumenti di comunicazione esterna

La comunicazione con l'azienda Socomec avverrà tramite **GMail** e **Microsoft Teams**

4.2 Organizzazione del lavoro

Si utilizza il metodo SCRUM_G per organizzare i periodi lavorativi e impostare gli obiettivi.

- Sprint di 1/2 settimane (corrispondenti al cambio ruoli)
- Piccoli obiettivi semplici da raggiungere
- Daily SCRUM_G della durata di 15 minuti circa.
- Il **Responsabile** assume il ruolo di **SCRUM Master_G**, il quale esprime le proprie opinioni in merito all'argomento trattato solo per ultimo, al fine di garantire il rispetto dei tempi stabiliti e concludere la seduta.
- Il piano di lavoro è configurato in modo da operare dal lunedì al venerdì, è **FONDAMENTALE** ritagliarsi il tempo per rispettare quanto pattuito. Solo in caso di necessità reali è possibile assentarsi dal daily SCRUM_G.
- Come strumento di supporto si utilizza la kanban_G di GitHub_G, adibita a SCRUM_G Board.

4.2.1 Incontri

Come specificato sopra, il Team si impegna ad incontrarsi giornalmente per un **Daily SCRUM_G** della durata di 15 minuti, per dare un breve resoconto dei compiti assegnati al Responsabile.

Inoltre, viene concordato **un giorno a settimana** per poter sviluppare un incontro più approfondito e duraturo. Tale incontro servirà per definire gli obiettivi settimanali ed eventualmente per aggiornare i ruoli dei membri del Team.

4.3 Ruoli di Progetto

I ruoli saranno assegnati in modo tale che ogni componente del gruppo ruoti attorno a tali figure professionali. Nel fare ciò bisognerà evitare conflitti di interesse, cioè un componente non potrà verificare un prodotto da lui stesso realizzato. I ruoli che saranno ricoperti nel corso del progetto sono:

- Responsabile
- Amministratore
- Analista
- Progettista
- Programmatore
- Verificatore

Di seguito le funzioni che ogni ruolo ha:

Responsabile Il Responsabile di progetto è un ruolo fondamentale che deve essere ricoperto per l'intera durata del progetto. Il suo compito principale è coordinare il team e rappresentarlo verso l'esterno (Livello Customer).

In particolare questo ruolo comporta:

- responsabilità di scelte e approvazioni;

- responsabilità sulla pianificazione delle attività rispettando le scadenze;
- coordinare i membri del gruppo e i compiti da svolgere;
- controllare, coordinare e relazionarsi verso soggetti esterni;
- avere conoscenze e capacità per saper valutare rischi, scelte, alternative.

Amministratore L'Amministratore di progetto è la figura che ha il controllo sull'ambiente di lavoro, pertanto il suo ruolo comporta:

- amministrare le infrastrutture e servizi di supporto;
- risolvere problemi legati alla gestione dei processi;
- salvaguardare la documentazione di progetto, controllando che venga verificata e corretta;
- controllare il versionamento_G e le configurazioni dei prodotti;
- individuare strumenti che portino ad una maggiore automazione dei processi.

Analista L'Analista si occupa di studiare a pieno il problema comprendendone tutte le caratteristiche. Conosce il dominio del problema e ha esperienza professionale. Questo ruolo è fondamentale nella fase iniziale, in particolare nella stesura dell'Analisi dei Requisiti_G. Tale ruolo ha molta influenza sul successo del progetto

Si deve occupare di:

- studiare il dominio del problema;
- analizzare e definire le richieste, quindi i requisiti del proponente_G (anche ciò che non è stato descritto esplicitamente);
- analizzare dove verrà applicato il prodotto finito, quindi i relativi casi d'uso_G;
- redigere l'Analisi dei Requisiti_G.

Progettista Il Progettista ha competenze tecniche e tecnologiche aggiornate. Deve sviluppare una soluzione al problema precedentemente analizzato dagli analisti, soddisfacendone i requisiti individuati.

I suoi compiti sono:

- creare un'architettura che sia coerente e consistente nelle sue parti;
- scegliere una soluzione che sia realizzabile nei costi stabiliti;
- cercare di limitare le dipendenze tra le varie componenti;
- far sì che il prodotto possa essere nelle sue parti riusabile.

Programmatore Il Programmatore partecipa alla realizzazione e alla manutenzione del progetto. Ha competenze tecniche ma autonomia e responsabilità circoscritte. È colui che si occupa della codifica, cioè deve implementare l'architettura che gli viene data dal Progettista.

I suoi compiti sono:

- codificare ciò che viene passato dal Progettista, documentando e versionando il tutto per agevolare la manutenzione;
- scrivere il Manuale utente del codice del prodotto.

Verificatore Il Verificatore è un ruolo presente per l'intera durata del progetto. Ha competenze tecniche, esperienza professionale, conoscenza del way of working_G e si occupa, quindi, di controllare che le attività vengano svolte nel rispetto delle norme e della qualità aspettata.

I suoi compiti sono:

- controllare che le attività si siano concluse senza la presenza di errori;
- incaricare chi di dovere di correggere eventuali problemi riscontrati durante lo svolgimento di un'attività;
- redigere la parte di retrospettiva del piano di qualifica_G.

4.3.1 Assegnazione Ruoli

I ruoli vengono assegnati ai singoli membri del team ruotando. Tali devono essere assegnati per garantire che non vi sia alcuna sovrapposizione in termini di richiesta di risorse.

I ruoli verranno assegnati cercando di attribuire lo stesso quantitativo di ore ad ogni membro. Al momento di ogni nuova assegnazione si devono considerare le abilità nel ricoprire quel ruolo per ogni singolo componente, in modo da aumentare la produttività.

La parola finale sulle assegnazioni andrà sempre al Responsabile.

5 Sistema di Qualità

Nelle seguenti sezioni verranno trattati gli elementi che costituiscono il sistema garante della qualità del prodotto e dei processi che lo compongono.

5.1 Obiettivi

Gli obiettivi del sistema di qualità istituito dal gruppo sono:

- Garantire una documentazione chiara, senza ambiguità e di livello;
- Affrontare ogni processo nella maniera più efficiente ed efficace possibile;
- Realizzare un prodotto conforme alle aspettative e piacevole da usare.

5.2 Descrizione

La descrizione di ogni processo e ogni aspetto del prodotto è consultabile tramite il documento di *Piano di qualifica_G*.

5.3 Spiegazione metriche_G - Qualità di processo

Codice	Nome	Descrizione	Formula
MPS1	Schedule Variance [SchV]	Mostra di quanto ci si è discostati dalla previsione delle tempistiche	$(\frac{h_i}{h_p} - 1) \cdot 100$
MPS2	Budget Variance [ResV]	Mostra di quanto ci si è discostati dalla previsione dei costi	$(\frac{bud_i}{bud_p} - 1) \cdot 100$
MPS3	Requirements Variance [ReqV]	Mostra quanto siano evoluti i requisiti dall'inizio del progetto	$\frac{req_m}{req_t} \cdot 100$
MPS4	Testing Success Rate [TSR]	Mostra la percentuale di test _G eseguiti con successo in rapporto al totale di questi	$\frac{t_p}{t_t} \cdot 100$
MPS5	Quality Level Obtained [QLO]	Mostra il livello di qualità complessivo ottenuto dai test _G precedenti	$\frac{SchV + ResV + ReqV + TSR}{4}$

Continua nella pagina successiva...

Legenda:

- $h_{i/p}$: numero di ore impegnate/previste;

- $bud_{i/p}$: numero di risorse impegnate/previste;
- $req_{m/t}$: numero di requisiti mutati/totali;
- t_p/t : numero di test_G positivi/totali.

5.4 Spiegazione metriche_G - Qualità di prodotto

Codice	Nome	Descrizione	Formula
MPT1	Orthographic Correctness [OC]	Mostra la percentuale di parole grammaticamente corrette in rapporto al totale	$\frac{pr_c}{pr_t} \cdot 100$
MPT2	Gulpease Index [GuI]	Indica la difficoltà di lettura del testo (più alto = più leggibile)	$89 + \frac{300 \cdot nr_f - 10 \cdot nr_l}{nr_p}$
MPT3	Mandatory Requirements Coverage [MRC]	Mostra la percentuale di requisiti obbligatori soddisfatti	$\frac{mReq_s}{mReq_t} \cdot 100$
MPT4	Optional Requirements Coverage [ORC]	Mostra la percentuale di requisiti opzionali soddisfatti	$\frac{oReq_s}{oReq_t} \cdot 100$
MPT5	Code Failure Rate [CFR]	Indica la percentuale di esecuzioni che terminano prematuramente	$\frac{ex_f}{ex_a} \cdot 100$
MPT6	Newcomer Max Learning Time [NMLT]	Indica la quantità massima di tempo necessaria a un nuovo utente per imparare ad usare l'app	$n^o \text{ minuti}$
MPT7	Functionality Max Depth [FMD]	Indica la quantità massima di click necessaria a raggiungere una funzione	$n^o \text{ click}$

Continua nella pagina successiva...

Codice	Nome	Descrizione	Formula
MPT8	Average Cyclomatic Complexity [ACC]	Indica in numero di cammini indipendenti presenti nel programma.	$v(G) = e - n + p$ <i>N.B.: dipende dal grafo di controllo di flusso</i>
MPT9	Application Response Time [ART]	Tempo massimo necessario al caricamento di una qualsiasi schermata dell'app	<i>n° secondi dal click al completo caricamento</i>
MPT10	Server Response Time [SRT];	Tempo massimo necessario al server per rispondere ad una richiesta	<i>n° secondi dall'invio della richiesta alla ricezione della risposta</i>
MPT11	UPS Refresh Time [UpsRT]	Tempo massimo necessario all'UPS per inviare i dati aggiornati all'app	<i>n° secondi dall'invio della richiesta alla ricezione della risposta</i>
MPT12	Web Application Refresh Time [WebRT]	Indica il tempo di aggiornamento dei dati dell'UPS visibili dalla web app	<i>UpsRT + n° secondi dall'invio dei dati dall'app alla ricezione nella web app</i>
MPT13	Automated Test Coverage [ATC]	Indica quanto codice è coperto dai test _G automatici	$\frac{cod_{test}}{cod_{tot}} \cdot 100$
MPT14	Application Installability [AppI]	Indica la percentuale di dispositivi su cui è installabile l'app	<i>valore ricavato dalle statistiche online</i>
MPT15	Server Installability [SrvI]	Indica la percentuale di dispositivi su cui è installabile il nodo server	<i>valore ricavato dalle statistiche online</i>
MPT16	WebApp Installability [WebI]	Indica la percentuale di browser _G compatibili con l'applicazione web	<i>valore ricavato dalle statistiche online</i>

Legenda:

- $pr_{c/t}$: numero di parole corrette/totali;
- $nr_{f/l/p/c}$: numero di frasi/lettere/parole/parole complesse;
- $mReq_{s/t}$: numero di requisiti obbligatori soddisfatti/totali;
- $oReq_{s/t}$: numero di requisiti opzionali soddisfatti/totali;
- $ex_{f/a}$: numero di esecuzioni fallimentari/avviate;
- $cod_{test/tot}$: quantità di codice coperto da $test_G$ /totale.