



---

# Norme di Progetto

v2.0.0

---

## Registro delle Modifiche

Data	Versione	Descrizione	Redattore	Verificatore
2026/04/19	2.0.0	Revisione per PB		Sandu Antonio
2026/03/30	1.1.0	Rielaborazione documento con modifiche proposte dal docente Vardanega	Suar Alberto	Sgreva Andrea
2026/03/03	1.0.0	Revisione per RTB		Suar Alberto
2026/02/28	0.14.0	Rifinitura e fix minori di tutto il documento	Sgreva Andrea	Suar Alberto
2026/02/27	0.13.1	Rifinitura della sezione Processo di Sviluppo e sezione Processi Organizzativi	Sgreva Andrea	Suar Alberto
2026/02/27	0.13.0	Rifinitura della sezione Piano di Qualifica	Berengan Riccardo	Sgreva Andrea
2026/02/23	0.12.0	Rifinitura della sezione Analisi dei Requisiti per la RTB	Suar Alberto	Basso Kevin
2026/02/21	0.11.0	Rielaborazione Processi di Supporto e Organizzativi	Zago Alice	Suar Alberto
2026/02/20	0.10.1	Fix sezione AdR in processo di fornitura	Sgreva Andrea	Suar Alberto
2026/02/14	0.10.0	Riorganizzazione sezione Processo di Sviluppo	Zago Alice	Suar Alberto
2026/02/13	0.9.0	Ampliamento sezioni Verifica e Validazione. Criteri di Verifica	Zago Alice	Suar Alberto

Data	Versione	Descrizione	Redattore	Verificatore
		Processi di Supporto e Sviluppo		
2026/02/11	0.8.1	Ultimo fix/ miglioramento sezione AdR	Sgreva Andrea	Suar Alberto
2026/02/11	0.8.0	Sistemazione sezione AdR con nuove sezioni e miglioramenti	Sgreva Andrea	Basso Kevin
2026/02/07	0.7.0	Rielaborazione sezioni relative al Piano di Qualifica	Zago Alice	Basso Kevin
2026/02/04	0.6.0	Rielaborazione delle Norme di Progetto relative al Piano di Progetto	Suar Alberto	Basso Kevin
2026/02/03	0.5.0	Riscrittura delle sezioni relative all'AdR	Sgreva Andrea	Suar Alberto
2026/02/03	0.4.0	Integrazioni a PdQ e Metriche	Zago Alice	Suar Alberto
2026/01/15	0.3.0	Aggiunta prima versione sezione documenti	Basso Kevin	Suar Alberto
2026/01/09	0.2.1	Correzioni minori e miglioramenti alla documentazione	Basso Kevin	Martinello Riccardo
2026/01/02	0.2.0	Arricchimento con standard industriali, norme di codifica dettagliate e integrazioni progetto- specifiche	Martinello Riccardo	Suar Alberto, Basso Kevin

---

<b>Data</b>	<b>Versione</b>	<b>Descrizione</b>	<b>Redattore</b>	<b>Verificatore</b>
2025/12/28	0.1.0	Rilascio iniziale con norme, processi e strumenti aggiornati	Martinello Riccardo	Suar Alberto, Basso Kevin
2025/12/20	0.0.1	Prima bozza iniziale	Martinello Riccardo	Suar Alberto, Basso Kevin

# Indice

<b>1 Introduzione</b>	<b>8</b>
1.1 Contesto del Progetto	8
1.2 Finalità del Documento	8
1.3 Glossario	8
1.4 Riferimenti	8
1.4.1 Riferimenti Normativi	8
1.4.2 Riferimenti Informativi	9
<b>2 Processi Primari</b>	<b>10</b>
2.1 Processo di Fornitura	11
2.1.1 Descrizione del Processo	11
2.1.2 Attività del Processo	11
2.1.2.1 Attività FOR-A01 – Pianificazione della Fornitura	11
2.1.2.1.1 Procedura PR-FOR-01: Avvio dello Sprint	11
2.1.2.1.2 Procedura PR-FOR-02: Gestione dei Rischi	11
2.1.2.2 Attività FOR-A02 – Monitoraggio e Controllo dell'Avanzamento	12
2.1.2.2.1 Procedura PR-FOR-03: Rendicontazione dello Sprint	12
2.1.2.2.2 Procedura PR-FOR-04: Gestione delle Comunicazioni Esterne	13
2.1.2.3 Attività FOR-A03 – Produzione della Documentazione	13
2.1.2.3.1 Procedura PR-FOR-05: Redazione degli Artefatti	13
2.1.2.3.2 Procedura PR-FOR-06: Gestione Terminologica (Glossario)	14
2.1.2.4 Attività FOR-A04 – Ingegneria dei Requisiti	15
2.1.2.4.1 Procedura PR-FOR-07: Modellazione dei Casi d'Uso	15
2.1.2.4.2 Procedura PR-FOR-08: Classificazione e Formalizzazione dei Requisiti	15
2.1.2.5 Attività FOR-A05 – Pianificazione Economica e Allocazione delle Risorse	15
2.1.2.5.1 Procedura PR-FOR-09: Definizione del Preventivo	15
2.1.2.6 Attività FOR-A06 – Consegna Formale degli Artefatti	16
2.1.2.6.1 Procedura PR-FOR-10: Checklist di Consegna	16
2.2 Processo di Sviluppo	17
2.2.1 Attività del Processo	17
2.2.1.1 Attività SVIL-A01 – Definizione del Ciclo di Vita	17
2.2.1.1.1 Procedura PR-SVIL-01: Adozione del Modello Incrementale	17
2.2.1.2 Attività SVIL-A02 – Analisi e Progettazione	17
2.2.1.2.1 Procedura PR-SVIL-02: Progettazione Architetture	17
2.2.1.2.2 Procedura PR-SVIL-03: Progettazione di Dettaglio	18
2.2.1.3 Attività SVIL-A03 – Implementazione	18
2.2.1.3.1 Procedura PR-SVIL-04: Codifica	18
2.2.1.4 Attività SVIL-A04 – Integrazione	19
2.2.1.4.1 Procedura PR-SVIL-05: Integrazione e Gestione della Configurazione	19
2.2.1.5 Attività SVIL-A05 – Verifica e Validazione	19
2.2.1.5.1 Procedura PR-SVIL-06: Testing a Livelli	19
2.2.1.5.2 Procedura PR-SVIL-07: Supporto all'Accettazione	20
<b>3 Processi di Supporto</b>	<b>21</b>
3.1 Processo di Documentazione	22
3.1.1 Attività SUPP-A01 – Gestione del Ciclo di Vita Documentale	22
3.1.1.1 Procedura PR-SUPP-01: Redazione e Approvazione	22

---

3.1.2	Attività SUPP-A02 – Controllo di Versione e Configurazione .....	22
3.1.2.1	Procedura PR-SUPP-02: Gestione Archivi e Versionamento .....	22
3.2	Processo di Qualifica .....	24
3.2.1	Attività SUPP-A03 – Verifica degli Artefatti .....	24
3.2.1.1	Procedura PR-SUPP-03: Esecuzione delle Verifiche .....	24
3.2.2	Attività SUPP-A04 – Validazione del Sistema .....	24
3.2.2.1	Procedura PR-SUPP-04: Validazione e Accettazione .....	24
<b>4</b>	<b>Processi Organizzativi .....</b>	<b>25</b>
4.1	Processo di Gestione dei Processi .....	25
4.1.1	Attività ORG-A01 – Pianificazione e Assegnazione .....	25
4.1.1.1	Procedura PR-ORG-01: Pianificazione Sprint e Gestione Task .....	25
4.1.2	Attività ORG-A02 – Coordinamento Informativo .....	25
4.1.2.1	Procedura PR-ORG-02: Esecuzione delle Comunicazioni .....	25
4.2	Processo di Infrastruttura .....	26
4.2.1	Attività ORG-A03 – Gestione Ambienti e Strumenti .....	26
4.2.1.1	Procedura PR-ORG-03: Configurazione e Manutenzione Repository .....	26
4.3	Processo di Miglioramento .....	26
4.3.1	Attività ORG-A04 – Monitoraggio e Retrospettiva .....	26
4.3.1.1	Procedura PR-ORG-04: Esecuzione Retrospettiva di Sprint .....	26
4.4	Processo di Formazione .....	26
4.4.1	Attività ORG-A05 – Sviluppo Competenze .....	26
4.4.1.1	Procedura PR-ORG-05: Pianificazione Formativa .....	26

## Indice tabelle

<b>Table 1</b>	<b>Matrice di Rischio .....</b>	<b>11</b>
<b>Table 2</b>	<b>Artefatti documentali del processo di fornitura .....</b>	<b>14</b>
<b>Table 3</b>	<b>Standard di nomenclatura tecnica .....</b>	<b>18</b>
<b>Table 4</b>	<b>Livelli di test e relative metodologie .....</b>	<b>20</b>
<b>Table 5</b>	<b>Strumenti di redazione documentale .....</b>	<b>22</b>

# 1 Introduzione

## 1.1 Contesto del Progetto

Il presente documento descrive le Norme di Progetto <sup>G</sup> applicate al progetto Code Guardian <sup>G</sup>, proposto dall'azienda Var Group <sup>G</sup> e realizzato dal team di sviluppo Skarab Group <sup>G</sup> nell'ambito del corso di Ingegneria del Software <sup>G</sup> presso l'Università degli Studi di Padova.

Il progetto ha come obiettivo lo sviluppo di una piattaforma software ad agenti <sup>G</sup> finalizzata all'audit <sup>G</sup> e alla remediation <sup>G</sup> automatizzata delle vulnerabilità presenti nei repository <sup>G</sup> di codice sorgente. Il sistema è progettato in conformità ai requisiti e ai vincoli definiti nel Capitolato C2.

La piattaforma supporta attività di analisi statica <sup>G</sup> del codice sorgente, di individuazione delle principali criticità di sicurezza e di analisi di file informativi come README e OpenAPI, fornendo suggerimenti di correzione attraverso meccanismi automatizzati basati su modelli di linguaggio di grandi dimensioni (LLM <sup>G</sup>).

## 1.2 Finalità del Documento

Le Norme di Progetto stabiliscono il quadro di riferimento metodologico, organizzativo e tecnologico adottato per lo sviluppo del sistema software <sup>G</sup>.

Il presente documento costituisce lo standard normativo interno del team; esso è vincolante per l'intero gruppo di lavoro e persegue i seguenti obiettivi:

- definire un insieme di regole comuni e convenzioni condivise al fine di garantire uniformità e coerenza nello sviluppo del prodotto software;
- individuare e descrivere gli strumenti, le tecnologie e gli ambienti di lavoro adottati;
- formalizzare i processi operativi, organizzativi e di controllo applicati durante l'intero ciclo di vita <sup>G</sup> del progetto;
- assicurare l'allineamento delle attività progettuali agli standard di riferimento e alle buone pratiche dell'ingegneria del software.

Il rispetto delle norme definite nel presente documento è obbligatorio per tutti i membri del gruppo di lavoro e costituisce un requisito fondamentale per garantire qualità, tracciabilità e controllo dell'intero ciclo di sviluppo.

## 1.3 Glossario

Al fine di prevenire ambiguità interpretative, è stato redatto un glossario che definisce in modo univoco la terminologia tecnica, gli acronimi e i concetti di dominio utilizzati all'interno della documentazione.

Nel testo, **ogni termine evidenziato tramite una G come apice**, rimanda alla voce corrispondente del Glossario pubblicato sul sito ufficiale del gruppo, consentendo al lettore di accedere direttamente alla definizione associata.

La versione più recente del Glossario è disponibile al seguente link: [Link al Glossario \(v2.0.0\)](#).

## 1.4 Riferimenti

### 1.4.1 Riferimenti Normativi

I seguenti documenti hanno valore vincolante per la redazione delle Norme di Progetto e per la definizione delle regole da rispettare durante il ciclo di vita del prodotto software:

- **Regolamento del Progetto Didattico:** Regole e vincoli del corso di Ingegneria del Software. <https://www.math.unipd.it/~tullio/IS-1/2025/Dispense/PD1.pdf> (ultimo accesso: 19/04/2026)

- **Capitolato C2:** Piattaforma ad agenti per l'audit e la remediation dei repository software.  
<https://www.math.unipd.it/~tullio/IS-1/2025/Progetto/C2.pdf>  
(ultimo accesso: 19/04/2026)

#### 1.4.2 Riferimenti Informativi

- **Dispense del corso di Ingegneria del Software relative ai Processi di Ciclo di Vita del Software**  
<https://www.math.unipd.it/~tullio/IS-1/2025/Dispense/T02.pdf>  
(ultimo accesso: 19/04/2026)
- **Dispense del corso di Ingegneria del Software relative alla Gestione di Progetto**  
<https://www.math.unipd.it/~tullio/IS-1/2025/Dispense/T04.pdf>  
(ultimo accesso: 19/04/2026)
- **IEEE 829 – Standard for Software and System Test Documentation:** definizione della struttura dei piani di test, casi di test, procedure e report.  
<https://ieeexplore.ieee.org/document/4578383>  
(ultimo accesso: 19/04/2026)
- **IEEE 830 – Standard for Software Requirements Specifications:** guida alla redazione e classificazione dei requisiti funzionali <sup>G</sup> e non funzionali.  
<https://ieeexplore.ieee.org/document/720574>  
(ultimo accesso: 19/04/2026)
- **IEEE 1016 – Recommended Practice for Software Design Descriptions:** linee guida per la descrizione dell'architettura software e dei principali design pattern.  
<https://ieeexplore.ieee.org/document/741934>  
(ultimo accesso: 19/04/2026)
- **ISO/IEC/IEEE 12207 – Software Life Cycle Processes:** framework internazionale per la gestione dei processi di sviluppo, manutenzione e dismissione del software.  
<https://ieeexplore.ieee.org/document/8100771>  
(ultimo accesso: 19/04/2026)

## 2 Processi Primari

In conformità allo standard ISO/IEC/IEEE 12207, il ciclo di vita del sistema software è descritto attraverso un insieme strutturato di processi. Ogni processo si articola in **attività operative**, ciascuna delle quali è realizzata attraverso **procedure attuative** con obiettivi e prodotti definiti, supportate da **strumenti** specifici. Nel contesto del presente progetto sono rilevanti i processi di **Fornitura e Sviluppo**.

## 2.1 Processo di Fornitura

### 2.1.1 Descrizione del Processo

Il Processo di Fornitura comprende le attività svolte dal gruppo Skarab Group in qualità di fornitore nei confronti del proponente Var Group. Esso governa la pianificazione, il coordinamento, la produzione e la consegna degli artefatti di progetto in conformità ai requisiti del Capitolato C2.

### 2.1.2 Attività del Processo

#### 2.1.2.1 Attività FOR-A01 – Pianificazione della Fornitura

Consiste nella definizione della struttura di lavoro, nell'assegnazione delle responsabilità operative e nella stima dell'impegno per ogni Sprint <sup>G</sup>.

##### 2.1.2.1.1 Procedura PR-FOR-01: Avvio dello Sprint

**Obiettivo:** Garantire la saturazione pianificata delle risorse all'inizio di ogni iterazione.

**Passi operativi:**

1. Il Responsabile <sup>G</sup> definisce le attività dello Sprint e le scompone in task <sup>G</sup> atomiche;
2. Ogni task viene stimata in ore, assegnata a un membro del team e registrata su Jira <sup>G</sup>
3. Le dipendenze tra task vengono esplicitate tramite relazioni "blocks/blocked-by" su Jira;
4. Il Responsabile verifica che la distribuzione del carico rispetti il preventivo approvato.

**Prodotti:** Sprint backlog <sup>G</sup> su Jira con task atomiche assegnate e stimate.

**Strumenti:** Jira.

**Criteri di conformità:**

- Tutte le task sono atomiche: eseguibili da un singolo membro, senza dipendenze bloccanti non tracciate;
- Ogni task ha un responsabile identificabile e una stima oraria;
- Il totale stimato è coerente con il preventivo dello Sprint.

##### 2.1.2.1.2 Procedura PR-FOR-02: Gestione dei Rischi

**Obiettivo:** Identificare, valutare e monitorare i fattori che possono compromettere gli obiettivi di progetto.

**Metodologia:** Il team adotta un approccio **semi-quantitativo**. Ogni rischio <sup>G</sup> è valutato tramite:

- **Probabilità (P):** scala discreta 1-3;
- **Impatto (I):** scala discreta 1-3;
- **Livello di Rischio (R):**  $R = P \times I$ .

Probabilità	Impatto		
	Basso (1)	Medio (2)	Alto (3)
Bassa (1)	1 (Basso)	2 (Basso)	3 (Basso)
Media (2)	2 (Basso)	4 (Medio)	6 (Alto)
Alta (3)	3 (Basso)	6 (Alto)	9 (Critico)

Table 1: Matrice di Rischio

**Categorie di rischio:**

- **RT** – Rischi Tecnologici (LLM, agenti, integrazioni);
- **RI** – Rischi Interpersonali (disponibilità del team, comunicazione);
- **RCO** – Rischi di Costo e Organizzativi (stime, schedulazione, vincoli esterni).

**Passi operativi:**

1. Ogni membro segnala l'insorgere di una problematica al Responsabile **entro il termine della giornata lavorativa**;
2. L'evento viene tracciato immediatamente nel sistema di issue tracking con categoria, P, I e R;
3. Il Responsabile attiva la strategia di risposta in base al livello R:
  - **R 1–3 (Basso)**: monitoraggio periodico;
  - **R 4 (Medio)**: definizione obbligatoria di una strategia di mitigazione;
  - **R 6–9 (Alto/Critico)**: attivazione di azioni preventive e piano di contingenza;
4. Al termine dello Sprint il Responsabile analizza nel consuntivo l'efficacia delle contromisure adottate.

**Prodotti:** Registro dei rischi aggiornato su Jira; sezione “Analisi dei Rischi” del Piano di Progetto <sup>G</sup>.

**Strumenti:** Jira.

**Criteri di conformità:**

- Ogni rischio è classificato (RT/RI/RCO) con valori P, I, R espliciti;
- Per ogni  $R \geq 4$  è definita una strategia di mitigazione o contingenza;
- Il registro è aggiornato a ogni Sprint.

**2.1.2.2 Attività FOR-A02 – Monitoraggio e Controllo dell'Avanzamento**

Consiste nella verifica sistematica dell'andamento del progetto rispetto alla pianificazione tramite iterazioni temporali (Sprint).

**2.1.2.2.1 Procedura PR-FOR-03: Rendicontazione dello Sprint**

**Obiettivo:** Misurare lo scostamento tra pianificato ed effettivo e produrre evidenza per le azioni correttive.

**Passi operativi:**

1. Ogni membro aggiorna quotidianamente su Jira lo stato delle proprie task, le ore effettive e l'avanzamento, **entro il termine della giornata lavorativa**;
2. Le attività non pianificate (bugfix, criticità emergenti) sono tracciate creando nuove task su Jira; l'omissione di tali ore è **vietata**;
3. Al termine dello Sprint il Responsabile confronta il Consuntivo <sup>G</sup> (dati Jira) con il Preventivo e ne analizza le cause di scostamento;
4. Il Responsabile valida le azioni correttive per lo Sprint successivo.

**Prodotti:** Consuntivo di Sprint (sezione del Piano di Progetto) con analisi delle deviazioni e azioni correttive.

**Strumenti:** Jira.

**Criteri di conformità:**

- Preventivo e Consuntivo di Sprint sono entrambi presenti e confrontati esplicitamente;
- Tutte le ore lavorate, incluse quelle extra-piano, sono tracciate;
- Le azioni correttive sono validate dal Responsabile e tracciate su Jira.

#### 2.1.2.2.2 Procedura PR-FOR-04: Gestione delle Comunicazioni Esterne

**Obiettivo:** Garantire professionalità e tracciabilità nelle comunicazioni con il proponente Var Group.

**Passi operativi:**

1. Le comunicazioni formali avvengono **esclusivamente** tramite email istituzionale o incontri verbalizzati;
2. Il Responsabile redige un verbale esterno **entro 48 ore lavorative** da ogni incontro con il proponente;
3. Il verbale viene inviato al proponente con identificatore univoco (es. VE\_2026-03-02);
4. Il silenzio-assenso del proponente dopo 24 ore dalla pubblicazione costituisce approvazione formale.

**Prodotti:** Verbale esterno in formato PDF, archiviato nel repository.

**Strumenti:** Email istituzionale; Typst <sup>G</sup> GitHub <sup>G</sup>.

**Criteri di conformità:**

- Il verbale è redatto entro le 48 ore;
- Include: data, partecipanti, ordine del giorno, decisioni, azioni con assegnatario;
- È archiviato nel repository ufficiale in formato PDF.

#### 2.1.2.3 Attività FOR-A03 – Produzione della Documentazione

Consiste nella redazione, verifica <sup>G</sup> e rilascio di tutti gli artefatti documentali previsti dal processo di fornitura.

##### 2.1.2.3.1 Procedura PR-FOR-05: Redazione degli Artefatti

**Obiettivo:** Garantire la conformità stilistica, la tracciabilità delle modifiche e la qualità dei documenti prodotti.

**Passi operativi:**

1. Il redattore crea o aggiorna il documento in Typst, nel rispetto delle convenzioni stabilite nelle presenti NdP;
2. Il documento viene versionato secondo lo schema x.y.z e associato a una task su Jira;
3. L'integrazione in `main` e in `develop` richiede una Pull Request approvata da un verificatore <sup>G</sup> **diverso** dal redattore;
4. Il verificatore controlla la conformità ai criteri definiti nel PdQ <sup>G</sup> e, se conforme, porta il documento allo stato **Approved**.

**Prodotti:** Artefatto documentale nello stato **Approved**, versione PDF, storico delle modifiche su GitHub.

**Strumenti:** Typst; GitHub (Pull Request, branch protection).

**Documenti prodotti da questa attività:**

Documento	Redattori	Destinatari	Uso
Lettera di Presentazione	Responsabile di Progetto	Proponente, Cardin Riccardo, Vardanega Tullio, Skarab Group	Esterno
AdR – Analisi dei Requisiti	Analisti	Proponente, Cardin Riccardo, Vardanega Tullio, Skarab Group	Esterno
Glossario	Tutti i membri	Stakeholder interni ed esterni	Esterno
NdP – Norme di Progetto	Responsabili di processo	Skarab Group, Cardin Riccardo, Vardanega Tullio	Interno
PdP – Piano di Progetto	Project Manager	Proponente, Cardin Riccardo, Vardanega Tullio, Skarab Group	Esterno
PdQ – Piano di Qualifica	Responsabili Qualità	Skarab Group, Cardin Riccardo, Vardanega Tullio, Proponente	Esterno
Verbali delle Riunioni	Responsabile	Proponente (se esterni), Cardin Riccardo, Vardanega Tullio, Skarab Group	Interno / Esterno
ST - Specifica Tecnica	Progettisti	Proponente, Cardin Riccardo, Vardanega Tullio, Skarab Group	Esterno
MU - Manuale Utente	Amministratori	Proponente, Cardin Riccardo, Vardanega Tullio, Skarab Group	Esterno

Table 2: Artefatti documentali del processo di fornitura

**Criteri di conformità:**

- Il documento è presente tra gli artefatti previsti e rispetta la struttura minima obbligatoria;
- Adotta le convenzioni tipografiche delle NdP;
- È nello stato **Approved** al momento della consegna.

**2.1.2.3.2 Procedura PR-FOR-06: Gestione Terminologica (Glossario)**

**Obiettivo:** Eliminare le ambiguità semantiche e garantire uniformità comunicativa nell'intera documentazione.

**Passi operativi:**

1. Il redattore aggiunge la voce al Glossario in ordine alfabetico, con definizione sintetica, non tautologica e contestualizzata;
2. In ogni documento, la prima occorrenza di un termine del Glossario viene marcata con l'apice <sup>G</sup> e linkata alla versione web ufficiale;
3. L'uso di sinonimi duplicati senza rimandi è **vietato**;
4. Il Glossario viene pubblicato sia in formato PDF statico sia come risorsa ipertestuale.

**Prodotti:** Glossario aggiornato (PDF + web); documentazione con marcatura <sup>G</sup> coerente.

**Strumenti:** Typst; sito web del progetto (anchor link).

**Criteri di conformità:**

- Ogni termine marcato <sup>G</sup> nella documentazione ha una voce corrispondente nel Glossario;
- Le voci sono in rigoroso ordine alfabetico;
- Ogni anchor link punta a una sezione esistente della pagina web;
- Le definizioni non sono tautologiche.

#### 2.1.2.4 Attività FOR-A04 – Ingegneria dei Requisiti

Consiste nell'elicitazione, analisi, formalizzazione e verifica dei requisiti del sistema a partire dal Capitolato e dai feedback del proponente.

**Base di comparazione:** IEEE 830-1998; ISO/IEC/IEEE 29148:2018; UML 2.5.

##### 2.1.2.4.1 Procedura PR-FOR-07: Modellazione dei Casi d'Uso

**Obiettivo:** Descrivere in modo non ambiguo i comportamenti attesi del sistema dal punto di vista degli attori.

**Passi operativi:**

1. Gli Analisti <sup>G</sup> identificano gli attori del sistema e li censiscono;
2. Per ogni scenario viene prodotta una descrizione testuale con nomenclatura UC $x.y.z$ , dove  $x$  è l'identificatore principale,  $y$  il numero di inclusioni e  $z$  il numero di estensioni;
3. Ogni caso d'uso <sup>G</sup> include obbligatoriamente: Attori, Pre/Post-condizioni, Scenario Principale, Trigger e, se presenti, Inclusioni, Estensioni o Generalizzazioni;
4. La descrizione testuale viene corredata da un diagramma UML 2.5 con corrispondenza 1:1.

**Prodotti:** Sezione "Casi d'Uso" dell'Analisi dei Requisiti; diagrammi UML <sup>G</sup>.

**Strumenti:** StarUML; Draw.io <sup>G</sup> Typst.

**Criteri di conformità:**

- Ogni identificatore UC $x.y.z$  è univoco;
- Ogni attore citato è stato precedentemente censito;
- Esiste corrispondenza 1:1 tra descrizione testuale e diagramma UML;
- La descrizione è limitata al "cosa fa il sistema", non al "come lo fa".

##### 2.1.2.4.2 Procedura PR-FOR-08: Classificazione e Formalizzazione dei Requisiti

**Obiettivo:** Produrre specifiche atomiche, non ambigue, verificabili e tracciate verso la loro fonte.

**Passi operativi:**

1. Gli Analisti derivano i requisiti dai casi d'uso e dal Capitolato;
2. Ogni requisito viene catalogato con la sintassi **TipologiaPrioritàNumero**:
  - **Tipologia:** F (Funzionale), Q (Qualità), V (Vincolo);
  - **Priorità:** Ob (Obbligatorio), De (Desiderabile), Op (Opzionale);
3. Per ogni requisito viene indicata la fonte (UC, sezione del Capitolato o decisione interna);
4. I requisiti vengono verificati rispetto ai criteri di atomicità, non ambiguità e testabilità.

**Prodotti:** Tabella dei requisiti nell'Analisi dei Requisiti.

**Strumenti:** Typst; Jira (tracciabilità requisiti-task).

**Criteri di conformità:**

- Ogni requisito esprime una singola necessità non ulteriormente scomponibile;
- Il linguaggio è tecnico, oggettivo e privo di termini soggettivi;
- Ogni requisito permette la definizione di un test di accettazione binario;
- I requisiti di qualità sono espressi tramite metriche quantitative o soglie numeriche.

#### 2.1.2.5 Attività FOR-A05 – Pianificazione Economica e Allocazione delle Risorse

Consiste nella stima dell'impegno, nella distribuzione dei ruoli e nella definizione del preventivo di progetto.

##### 2.1.2.5.1 Procedura PR-FOR-09: Definizione del Preventivo

**Obiettivo:** Produrre una baseline economica verificabile e aggiornabile a ogni Sprint.

**Passi operativi:**

1. Il Responsabile definisce il preventivo iniziale rispettando i seguenti vincoli:
  - Le ore di auto-apprendimento precedenti al primo Sprint **non sono rendicontabili**;
  - La distribuzione dell'impegno orario pro-capite è esplicitata e motivata;
2. Per la milestone **RTB**<sup>G</sup> l'allocazione prevede un impiego prevalente delle figure di Amministratore<sup>G</sup> e Analista, in ragione dell'alto carico documentale;
3. Per la milestone **PB**<sup>G</sup> il preventivo è gestito con metodologia **rolling-wave**: il Responsabile lo riesamina alla chiusura di ogni Sprint e ogni revisione deve essere approvata;
4. Ogni scostamento rispetto alle stime è documentato e motivato nei verbali o su Jira.

**Prodotti:** Sezione "Preventivo e Allocazione delle Risorse" del Piano di Progetto.

**Strumenti:** Jira; Typst.

**Criteri di conformità:**

- Il preventivo iniziale è coerente con le baseline individuate;
- La ripartizione dei ruoli per fase (RTB/PB) è coerente con le attività pianificate;
- Gli scostamenti dal preventivo sono motivati e tracciati.

**2.1.2.6 Attività FOR-A06 – Consegna Formale degli Artefatti**

Consiste nel consolidamento, nel tagging e nella notifica ufficiale della consegna al proponente in occasione delle scadenze di milestone.

**2.1.2.6.1 Procedura PR-FOR-10: Checklist di Consegna**

**Obiettivo:** Assicurare che ogni consegna sia completa, verificata e formalmente notificata.

**Passi operativi:**

1. Il Responsabile accerta che tutti i documenti siano nello stato **Approved** e soddisfino i criteri del PdQ;
2. Vengono generati i PDF definitivi dai sorgenti Typst congelati;
3. Viene creata una Release sul repository GitHub con tag semantico (es. v1.0.0-RTB);
4. Il Responsabile invia comunicazione formale di avvenuta consegna al proponente indicando il link alla release.

**Prodotti:** Release GitHub con tag semantico; email di notifica al proponente.

**Strumenti:** Typst; GitHub (Releases); email istituzionale.

**Criteri di conformità:**

- Tutti i documenti sono in stato **Approved**;
- Il tag segue la convenzione semantica (vX.Y.Z-MILESTONE);
- La notifica formale è inviata entro la scadenza contrattuale.

## 2.2 Processo di Sviluppo

Il Processo di Sviluppo adottato da Skarab Group definisce le attività necessarie alla realizzazione tecnica del prodotto *Code Guardian*, in conformità allo standard ISO/IEC/IEEE 12207.

Lo stack tecnologico adottato è:

- **TypeScript**<sup>G</sup>: Frontend<sup>G</sup> (React<sup>G</sup>) e Backend<sup>G</sup> (NestJS<sup>G</sup>);
- **Python**<sup>G</sup>: Agenti AI, Orchestratore e script di analisi dati;
- **MongoDB**<sup>G</sup> e **Node.js**<sup>G</sup>: Database<sup>G</sup> con Mongoose e ambiente di esecuzione server-side.

Le attività del processo sono distribuite tra le due baseline<sup>G</sup> principali:

- **RTB (Requirements and Technology Baseline)**: Analisi dei Requisiti, Proof of Concept<sup>G</sup>, definizione degli standard normativi;
- **PB (Product Baseline)**: Progettazione Architeturale, Codifica, consolidamento del prodotto.

### 2.2.1 Attività del Processo

#### 2.2.1.1 Attività SVIL-A01 – Definizione del Ciclo di Vita

Consiste nella selezione e nella formalizzazione del modello di ciclo di vita adottato per lo sviluppo di *Code Guardian*.

##### 2.2.1.1.1 Procedura PR-SVIL-01: Adozione del Modello Incrementale

**Obiettivo:** Garantire un approccio di sviluppo graduato, verificabile e allineato alle milestone RTB e PB.

**Passi operativi:**

1. Il team adotta un modello iterativo-incrementale<sup>G</sup> articolato in Sprint di durata fissa;
2. Ogni Sprint produce un incremento verificabile del prodotto o della documentazione tecnica;
3. Al termine di ogni Sprint viene svolta una retrospettiva per identificare azioni di miglioramento.

**Prodotti:** Definizione del ciclo di vita nel Piano di Progetto; retrospettive documentate.

**Strumenti:** Jira; Typst.

**Criteri di conformità:**

- Ogni Sprint è avviato con task stimate su Jira e concluso con una retrospettiva documentata;
- Le milestone RTB e PB sono allineate con gli incrementi previsti.

#### 2.2.1.2 Attività SVIL-A02 – Analisi e Progettazione

Consiste nella definizione logica e strutturale del sistema, dall'analisi dei requisiti<sup>G</sup> alla specifica architeturale e di dettaglio.

##### 2.2.1.2.1 Procedura PR-SVIL-02: Progettazione Architeturale

**Obiettivo:** Definire la struttura di alto livello del sistema garantendo scalabilità e separazione delle responsabilità.

**Passi operativi:**

1. Gli Analisti e i Progettisti identificano le componenti hardware e software del sistema;
2. Viene prodotto un diagramma architeturale UML 2.5 delle componenti principali;
3. Per ogni componente sono specificate le responsabilità, le interfacce esposte e le dipendenze;
4. La progettazione garantisce la separazione tra logica di business e interfaccia utente.

**Prodotti:** Documento di Specifica Tecnica<sup>G</sup> (diagrammi architeturali, descrizione delle componenti).

**Strumenti:** StarUML; Draw.io; Typst.

**Criteri di conformità:**

- Ogni requisito dell'AdR è tracciato verso almeno una componente progettuale;
- La strategia di gestione delle eccezioni è specificata;
- I diagrammi UML sono aggiornati e coerenti con la documentazione tecnica.

**2.2.1.2.2 Procedura PR-SVIL-03: Progettazione di Dettaglio**

**Obiettivo:** Specificare classi, interfacce, strutture dati e algoritmi prima della codifica.

**Passi operativi:**

1. I Progettisti definiscono le classi e le interfacce di ogni componente secondo gli standard di nomenclatura (vedi PR-SVIL-05);
2. Per ogni interfaccia sono specificati i metodi esposti, i parametri e i valori di ritorno;
3. La specifica viene prodotta in Typst prima dell'avvio della codifica del componente.

**Prodotti:** Specifica di dettaglio (diagrammi di classe UML, descrizione delle interfacce).

**Strumenti:** StarUML; Draw.io; Typst.

**Criteri di conformità:**

- Interfacce, classi e algoritmi sono specificati prima della fase di codifica;
- I diagrammi di classe sono coerenti con il codice prodotto.

**2.2.1.3 Attività SVIL-A03 — Implementazione**

Consiste nella trasformazione delle specifiche progettuali in codice sorgente eseguibile, nel rispetto degli standard tecnici adottati.

**2.2.1.3.1 Procedura PR-SVIL-04: Codifica**

**Obiettivo:** Produrre codice corretto, leggibile, modulare e conforme agli standard tecnici del progetto.

**Passi operativi:**

1. Il Programmatore implementa le specifiche rispettando gli standard di nomenclatura:

Tipo	Standard Obbligatorio
camelCase	Variabili e funzioni
PascalCase	Classi, componenti React, interfacce (prefisso I)
kebab-case	Nomi dei file di logica e Collezioni MongoDB (al plurale)
SCREAMING_SNAKE_CASE	Costanti e variabili d'ambiente (.env)

Table 3: Standard di nomenclatura tecnica

1. Ogni file deve **obbligatoriamente** essere nominativo della classe che descrive, indicando come estensione il tipo di componente che ci si aspetta all'interno del file (es: analysisIdt.vo.ts per indicare che è un Value Object). Per nomi come Value Object, Use Case, Domain Service è sufficiente l'acronimo;
2. La logica di business è separata dall'interfaccia utente; ogni funzionalità è isolata in componenti atomici (**Single Responsibility Principle**);
3. Ogni chiamata asincrona è protetta da blocchi try-catch con logging specifico;
4. Prettier (2 spazi, apici singoli) ed ESLint <sup>G</sup> sono eseguiti obbligatoriamente in pipeline CI.
5. Prima dell'avvio di ogni sessione di codifica, è **obbligatorio** leggere e rispettare le direttive di lavoro indicate in CONTRIBUTING.md.

**Prodotti:** Codice sorgente versionato su GitHub; report di analisi statica.

**Strumenti:** VS Code <sup>G</sup> ESLint; Prettier; Docker <sup>G</sup>.

**Criteri di conformità:**

- Il codice supera i controlli di linting (ESLint/Prettier) senza errori;
- La densità di commenti rispetta la metrica MPD04 del PdQ;
- La complessità ciclomatica <sup>G</sup> (MPD05) non supera la soglia definita nel PdQ.

#### 2.2.1.4 Attività SVIL-A04 – Integrazione

Consiste nella combinazione progressiva delle unità software e nella verifica del loro funzionamento congiunto.

##### 2.2.1.4.1 Procedura PR-SVIL-05: Integrazione e Gestione della Configurazione

**Obiettivo:** Garantire la coerenza e la stabilità del codebase <sup>G</sup> a ogni integrazione di nuove unità.

**Passi operativi:**

1. Il team adotta la strategia di branching: main (stabile), develop (integrazione), feature/ (task atomiche);
2. I commit seguono lo standard **Conventional Commits** (es. feat(parser): add python support) e devono essere redatti **tassativamente** in lingua inglese;
3. L'integrazione in develop e main è **vietata** senza Pull Request approvata da un componente terzo (Peer Review);
4. Il Verificatore esegue l'analisi statica prima di approvare la PR;
5. Le unità integrate vengono sottoposte ai test di integrazione prima del merge in develop.

**Prodotti:** Codebase aggiornato su GitHub; report della CI.

**Strumenti:** GitHub (Pull Request, branch protection rules); pipeline CI.

**Criteri di conformità:**

- Non sono presenti commit diretti sui rami protetti (develop, main);
- Ogni PR supera i controlli automatizzati di CI;
- Non sono presenti regressioni a seguito dell'integrazione di nuove unità.

#### 2.2.1.5 Attività SVIL-A05 – Verifica e Validazione

Consiste nell'esecuzione sistematica dei test per accertare la correttezza, la robustezza e la conformità del sistema rispetto ai requisiti.

##### 2.2.1.5.1 Procedura PR-SVIL-06: Testing a Livelli

**Obiettivo:** Verificare il comportamento del sistema a ogni livello di granularità, dalla singola unità all'intero sistema.

**Livelli di test adottati:**

Livello	Metodologia Obbligatoria
Unit Test	Verifica delle singole unità tramite Mocking. Nomenclatura: UT-XXX.
Integration Test	Verifica interazione tra moduli (es. Python e NestJS). Nomenclatura: IT-XXX.
System Test	Simulazione di scenari reali per verificare interazione e robustezza. Nomenclatura: ST-XXX.
Analisi Statica	Controllo automatico della qualità prima di ogni merge.

Table 4: Livelli di test e relative metodologie

**Passi operativi:**

1. Il Verificatore implementa i test di unità contestualmente alla codifica di ogni componente;
2. I test di integrazione vengono eseguiti a ogni merge in develop;
3. I test di sistema vengono eseguiti su scenari reali prima di ogni rilascio;
4. I risultati vengono tracciati nel Piano di Qualifica <sup>G</sup> (Cruscotto di Valutazione);
5. Il superamento delle soglie di non conformità comporta l'apertura immediata di una segnalazione su Jira.

**Prodotti:** Suite di test versionata; report di esecuzione; aggiornamento del Cruscotto di Valutazione nel PdQ.

**Strumenti:** Jest; Pytest; Postman.

**Criteri di conformità:**

- La Code Coverage <sup>G</sup> (MPC13) non scende sotto la soglia definita nel PdQ;
- Il Test Success Rate (MPC14) è al 100% prima di ogni rilascio;
- Ogni unità è corredata dai relativi Unit Test.

**2.2.1.5.2 Procedura PR-SVIL-07: Supporto all'Accettazione**

**Obiettivo:** Fornire al proponente evidenza del soddisfacimento dei requisiti obbligatori e risolvere eventuali non conformità.

**Passi operativi:**

1. Il team prepara la documentazione di evidenza per i requisiti obbligatori (MPD01 = 100%);
2. Vengono eseguiti i test di accettazione in presenza del proponente o tramite evidenza documentata;
3. Le non conformità rilevate dal proponente vengono tracciate su Jira e risolte prima della consegna finale.

**Prodotti:** Report di accettazione; verbale della sessione di collaudo (se presente).

**Strumenti:** Jira; Postman.

**Criteri di conformità:**

- Tutti i requisiti obbligatori sono coperti da evidenza di test;
- Le non conformità sono tracciate e risolte prima della consegna.

### **3 Processi di Supporto**

Basandosi sullo standard ISO/IEC/IEEE 12207, i Processi di Supporto comprendono le attività trasversali che accompagnano l'intero ciclo di vita del progetto, garantendo la qualità, la tracciabilità e la verificabilità degli artefatti prodotti.

### 3.1 Processo di Documentazione

#### 3.1.1 Attività SUPP-A01 – Gestione del Ciclo di Vita Documentale

##### 3.1.1.1 Procedura PR-SUPP-01: Redazione e Approvazione

**Obiettivo:** Disciplinare le fasi di scrittura, controllo e rilascio ufficiale di ogni documento di progetto.

**Passi operativi:**

1. **Creare** o **modificare** il contenuto in formato sorgente (Typst);
2. **Tracciare** obbligatoriamente l'attività di redazione tramite una task assegnata su Jira;
3. **Sottoporre** il documento a verifica di conformità (ortografia, stile, contenuti, rispetto delle NdP) assegnando la task a un membro del team diverso dal redattore;
4. **Approvare** e validare il documento (da parte del Responsabile di Progetto) per procedere al rilascio ufficiale.

**Prodotti:** Documentazione ufficiale di progetto (copertina, registro modifiche, indice, header/footer con numerazione).

**Strumenti:**

Strumento	Descrizione e Utilizzo Obbligatorio
VS Code	Editor principale per la scrittura dei documenti.
Typst	Linguaggio di markup per la formattazione. I template comuni sono centralizzati nella cartella <code>lib/</code> .
Tinymist <sup>G</sup>	Estensione per VS Code per l'anteprima in tempo reale.

Table 5: Strumenti di redazione documentale

**Criteri di conformità:**

- Il documento possiede la struttura minima obbligatoria;
- Il Registro delle modifiche riporta i campi compilati e il corretto incremento di versione;
- La task di verifica su Jira risulta completata da un membro indipendente.

#### 3.1.2 Attività SUPP-A02 – Controllo di Versione e Configurazione

##### 3.1.2.1 Procedura PR-SUPP-02: Gestione Archivi e Versionamento

**Obiettivo:** Standardizzare le nomenclature, tracciare le modifiche in modo univoco e mantenere integra la repository.

**Passi operativi:**

1. **Applicare** il versionamento semantico (x.y.z) a ogni iterazione documentale (x per approvazione formale, y per modifiche sostanziali, z per modifiche minori);
2. **Aggiornare** il changelog all'interno del documento in concomitanza con ogni incremento di versione;
3. **Denominare** i file sorgente seguendo lo schema SIGLA.typ (es. NdP.typ) e **utilizzare** il formato data AAAA/MM/GG;
4. **Integrare** i file completati e verificati nel branch `develop` esclusivamente tramite Pull Request al termine di ogni Sprint.

**Prodotti:** Repository DocumentazioneProgetto strutturata e mantenuta; storico dei commit.

**Strumenti:** Git, GitHub.

**Criteri di conformità:**

- Versionamento semantico applicato senza salti incongruenti;
- Repository aderente alla struttura gerarchica definita (es. lib/, RTB/, verba/i/).

## 3.2 Processo di Qualifica

### 3.2.1 Attività SUPP-A03 – Verifica degli Artefatti

#### 3.2.1.1 Procedura PR-SUPP-03: Esecuzione delle Verifiche

**Obiettivo:** Intercettare e risolvere i difetti nel software e nella documentazione prima dell'integrazione nel branch principale.

**Passi operativi:**

1. **Aprire** una Pull Request su GitHub per ogni nuovo incremento di codice o aggiornamento documentale;
2. **Assegnare** la revisione obbligatoria della Pull Request a un Verificatore diverso dall'autore;
3. **Calcolare** l'indice di Gulpease <sup>G</sup> per i documenti utilizzando il tool preposto e **confrontarlo** con le metriche del PdQ;
4. **Eseguire** l'analisi statica sul codice applicando review manuali (Ispezione/Walkthrough) e strumenti di linting automatico;
5. **Eseguire** le suite di analisi dinamica (Test di Unità, Integrazione, Sistema e Regression);
6. **Effettuare** il merge nel branch main solo a fronte del superamento di tutti i controlli qualitativi.

**Prodotti:** Codice sorgente e documenti verificati; Esiti di verifica documentati nel Piano di Qualifica.

**Strumenti:** [farfalla-project.org](https://farfalla-project.org) Linter <sup>G</sup> (ESLint, Pylint/Black); Framework di testing.

**Criteri di conformità:**

- Test Success Rate pari al 100% prima del merge in main;
- Pull Request approvata esplicitamente da un verificatore indipendente su GitHub;
- Indici di leggibilità conformi alle soglie del PdQ.

### 3.2.2 Attività SUPP-A04 – Validazione del Sistema

#### 3.2.2.1 Procedura PR-SUPP-04: Validazione e Accettazione

**Obiettivo:** Accertare la conformità del prodotto finale ai requisiti concordati con il proponente.

**Passi operativi:**

1. **Tracciare** ogni requisito derivato dall'**Analisi dei Requisiti** fino alla sua implementazione nel sistema;
2. **Sviluppare** i Test di Accettazione (TA) basandoli su scenari d'uso reali;
3. **Eseguire** i Test di Accettazione e **verificare** il soddisfacimento dei criteri di successo oggettivi per ciascun requisito obbligatorio.

**Prodotti:** Matrice di tracciamento requisiti aggiornata; verbali di validazione <sup>G</sup> e accettazione.

**Strumenti:** Sistema di issue tracking (Jira); Framework di automazione test.

**Criteri di conformità:**

- Copertura completa (100%) dei requisiti obbligatori implementati e tracciati;
- Superamento con successo di tutti i Test di Accettazione definiti.

## 4 Processi Organizzativi

Prendendo spunto dallo standard ISO/IEC/IEEE 12207, i Processi Organizzativi costituiscono l'insieme delle attività trasversali che garantiscono il corretto andamento del progetto dal punto di vista gestionale, infrastrutturale e della crescita professionale del team.

### 4.1 Processo di Gestione dei Processi

#### 4.1.1 Attività ORG-A01 – Pianificazione e Assegnazione

##### 4.1.1.1 Procedura PR-ORG-01: Pianificazione Sprint e Gestione Task

**Obiettivo:** Allocare le risorse in modo efficiente ed equo, strutturando il lavoro in unità misurabili.

**Passi operativi:**

1. **Assegnare** i ruoli di progetto ai membri del team all'inizio di ogni Sprint, rispettando la rotazione ciclica e la matrice di responsabilità;
2. **Scomporre** le attività di progetto in task atomiche, prive di dipendenze bloccanti non esplicitate;
3. **Registrare** ogni task su Jira definendo obbligatoriamente: descrizione univoca, assegnatario, stima in ore, scadenza e Sprint di riferimento;
4. **Rendicontare** quotidianamente l'avanzamento su Jira aggiornando lo stato delle task e le ore effettivamente consumate.

**Prodotti:** Backlog di Sprint; Board Jira aggiornata.

**Strumenti:** Jira.

**Criteri di conformità:**

- Assenza di task prive di stima oraria o assegnatario;
- Ore consuntivate aggiornate quotidianamente a sistema.

#### 4.1.2 Attività ORG-A02 – Coordinamento Informativo

##### 4.1.2.1 Procedura PR-ORG-02: Esecuzione delle Comunicazioni

**Obiettivo:** Garantire il corretto allineamento informativo tra i membri del team e verso gli stakeholder <sup>G</sup> esterni.

**Passi operativi:**

1. **Condurre** riunioni periodiche sincrone su Discord <sup>G</sup> per condividere l'avanzamento e intercettare eventuali criticità;
2. **Redigere** e archiviare un verbale interno a seguito di ogni riunione sincrona del team;
3. **Utilizzare** sistemi di messaggistica istantanea (WhatsApp <sup>G</sup>/Telegram <sup>G</sup>) esclusivamente per coordinamenti operativi rapidi e asincroni;
4. **Comunicare** con il proponente (Var Group) o i docenti unicamente tramite Email istituzionale;
5. **Redigere** un verbale esterno ufficiale a seguito di ogni incontro con gli stakeholder.

**Prodotti:** Verbali interni ed esterni archiviati nella repository.

**Strumenti:** Discord, WhatsApp/Telegram, Email.

**Criteri di conformità:**

- Presenza a repository di un verbale per ogni SAL <sup>G</sup> (Stato Avanzamento Lavori) o riunione esterna effettuata.

## 4.2 Processo di Infrastruttura

### 4.2.1 Attività ORG-A03 – Gestione Ambienti e Strumenti

#### 4.2.1.1 Procedura PR-ORG-03: Configurazione e Manutenzione Repository

**Obiettivo:** Assicurare la continuità operativa fornendo ambienti di lavoro protetti e automatizzati.

**Passi operativi:**

1. **Configurare** e mantenere il toolkit tecnologico (Git <sup>G</sup>, GitHub, GitHub Actions);
2. **Applicare** le regole di protezione dei branch su GitHub, impedendo l'esecuzione di commit diretti sui branch main e develop;
3. **Applicare** la nomenclatura standardizzata per la creazione di nuovi branch (feature/nome, fix/nome, [ACRONIMO-DOC]);
4. **Mantenere** funzionanti gli script di automazione CI/CD per la generazione documentale e l'aggiornamento del sito web.

**Prodotti:** Infrastruttura IT di progetto operativa; repository protetta.

**Strumenti:** Git, GitHub, GitHub Actions.

**Criteri di conformità:**

- Nessun commit effettuato direttamente nei branch protetti.
- Pipeline di automazione passanti (green) sui branch principali.

## 4.3 Processo di Miglioramento

### 4.3.1 Attività ORG-A04 – Monitoraggio e Retrospezione

#### 4.3.1.1 Procedura PR-ORG-04: Esecuzione Retrospezione di Sprint

**Obiettivo:** Applicare il ciclo PDCA <sup>G</sup> (Plan-Do-Check-Act) per individuare le inefficienze metodologiche e incrementare la qualità del lavoro.

**Passi operativi (Da eseguire al termine di ogni Sprint):**

1. **Raccogliere** i valori delle metriche misurate e **confrontarli** con le soglie accettabili definite nel PdQ;
2. **Analizzare** collettivamente le inefficienze operative riscontrate durante lo Sprint appena concluso;
3. **Definire** specifiche azioni correttive e **registrarle** su Jira con assegnatario e scadenza precise;
4. **Aggiornare** la documentazione di governo (NdP o PdQ) nel caso in cui le azioni correttive richiedano una modifica permanente dei processi.

**Prodotti:** Issue di azioni correttive su Jira; Documenti di governo aggiornati.

**Strumenti:** Jira, cruscotto delle metriche di progetto.

**Criteri di conformità:**

- Presenza su Jira delle issue relative alle azioni di miglioramento concordate;
- Metric report elaborato per lo Sprint in esame.

## 4.4 Processo di Formazione

### 4.4.1 Attività ORG-A05 – Sviluppo Competenze

#### 4.4.1.1 Procedura PR-ORG-05: Pianificazione Formativa

**Obiettivo:** Garantire l'acquisizione delle competenze tecnologiche (TypeScript, React, NestJS, AI, MongoDB, AWS) senza generare costi per il committente.

---

**Passi operativi:**

1. **Pianificare** sessioni di studio e approfondimento tecnologico nelle fasi di progetto a minor carico operativo;
2. **Escludere** tassativamente le ore dedicate all'auto-formazione individuale dal rendiconto delle ore produttive di progetto;
3. **Partecipare** alle sessioni formative e di supporto fornite dall'azienda proponente (**Var Group**).

**Prodotti:** Dimostrazioni pratiche (PoC) funzionanti; team operativo sul tech-stack.

**Strumenti:** Documentazione ufficiale dei framework, materiale Var Group.

**Criteri di conformità:**

- Nessuna ora rendicontata a sistema risulta imputabile a studio autonomo o formazione.