



Skarab Group

Way Of Working

2025 - 11 - 04 v1.1

Nome
Alberto Suar
Kevin Basso
Riccardo Martinello
Alice Zago
Andrea Sgreva
Antonio Sandu
Riccardo Berengan

skarabswegroup@gmail.com

Indice

1 Definizione delle Repository	2
1.1 Dove definire una Repository	2
1.2 Quando definire una Repository	2
1.3 Gestione e Struttura delle Repository	2
2 Definizione dei Ruoli	4
2.1 Ruoli del team	4
2.2 Logica di distribuzione dei ruoli	4
3 Stesura della Documentazione	5
3.1 Obiettivi	5
3.2 Tipologie di documenti	5
3.3 Aggiornamento e revisione	5
3.4 Archiviazione	6
3.5 Strumenti	6
3.6 Tempistiche	6
3.7 Controllo di qualità	6
4 Periodicità ed Organizzazione dei Meeting	7
4.1 Tipologie di meeting	7
4.2 Frequenza	7
4.3 Ruoli e responsabilità	7
4.4 Convocazione e Agenda	7
4.5 Durata e Orario	7
4.6 Output e Follow-Up	8
4.7 Strumenti e Modalità	8
4.8 Monitoraggio e Revisione delle Regole	8

1 Definizione delle Repository

1.1 Dove definire una Repository

Ogni repository ritenuta necessaria per lo sviluppo del progetto deve essere creata all'interno dell'Organizzazione GitHub ufficiale del team, accessibile al seguente link: <https://github.com/SkarabGroup>.

All'interno dell'Organizzazione, ogni componente del team ricopre il ruolo di **Owner**, il quale garantisce pieno accesso e controllo sulle repository. Ciò consente a ciascun membro di svolgere tutte le operazioni necessarie, dalla gestione dei permessi alla creazione e modifica dei contenuti, assicurando la massima autonomia e flessibilità nello sviluppo.

1.2 Quando definire una Repository

Il principio cardine del nostro team è mantenere ordine e tracciabilità durante tutto il ciclo di vita del progetto. La creazione di una nuova repository deve essere valutata attentamente e considerata solo se ritenuta realmente necessaria dall'intero team.

Prima di procedere, è consigliato effettuare una breve analisi condivisa che risponda alle seguenti domande:

- La nuova repository migliorerà la collaborazione tra i membri del team?
- Aiuterà a isolare funzionalità complesse o a facilitare la gestione del codice?
- Eviterà duplicazioni o frammentazioni indesiderate?
- È coerente con le convenzioni e le repository già esistenti nel progetto?

Se la maggioranza del team concorda sulla sua utilità, la repository può essere creata. Qualora uno o più membri ritengano la creazione non necessaria, è opportuno discutere e documentare le motivazioni che ne evidenziano i potenziali svantaggi.

Questo approccio garantisce trasparenza, responsabilità condivisa e decisioni basate su valutazioni concrete, riducendo il rischio di repository superflue o mal organizzate.

1.3 Gestione e Struttura delle Repository

Oltre a definire quando e dove creare una repository, è fondamentale stabilire alcune linee guida per la sua gestione e organizzazione, al fine di garantire uniformità, tracciabilità e collaborazione efficace tra i membri del team.

Struttura della repository Ogni repository deve seguire convenzioni chiare per cartelle e file principali, ad esempio:

- **src/** per il codice sorgente;
- **docs/** per la documentazione;
- **tests/** per i test automatizzati.

I nomi di file e cartelle devono essere coerenti con le convenzioni stabilite dal team.

Branching model Il team adotta un modello di branching definito, come Git Flow o GitHub Flow.

- **main/master**: branch stabile e pronto per la produzione;
- **develop**: branch di integrazione delle feature;
- **feature branches**: per lo sviluppo di nuove funzionalità o bugfix.

Le pull request devono essere revisionate da almeno un altro membro prima del merge.

Commit e convenzioni I messaggi di commit devono essere chiari e descrittivi, seguendo uno standard concordato dal team. Si consiglia di effettuare commit frequenti per favorire la tracciabilità dei cambiamenti. Si consiglia inoltre di usare il comando `git commit -m "#numero_issue"` per la chiusura delle Issues su GitHub.

Documentazione Ogni repository deve contenere un file `README.md` che includa:

- Scopo della repository;
- Istruzioni per l'installazione e l'utilizzo;

CI/CD e test automatizzati Se applicabile, ogni repository deve integrare pipeline di Continuous Integration per automatizzare build e test. Le pull request devono superare tutti i test prima del merge ed essere validate dal componente che in quel momento starà ricoprendo il ruolo di **Verificatore**.

2 Definizione dei Ruoli

Il team è composto da 7 membri, ciascuno con una disponibilità dalle 10 alle 20 ore settimanali dedicate al progetto. Per garantire una distribuzione equa delle responsabilità e un'esperienza completa per tutti i membri, i ruoli fondamentali vengono ruotati periodicamente.

2.1 Ruoli del team

I ruoli previsti all'interno del progetto sono i seguenti:

- **Responsabile:** coordina l'elaborazione di piani e di scadenze, approvando il rilascio di prodotti parziali o finali e coordinando le attività di gruppo. Questo ruolo deve **SEMPRE** essere ricoperto da al più un componente del gruppo;
- **Amministratore:** assicura l'efficienza di procedure, strumenti e tecnologie a supporto del WoW;
- **Analista:** svolge le attività di Analisi dei Requisiti;
- **Progettista:** svolge le attività di progettazione;
- **Sviluppatore:** svolge le attività di codifica.
- **Verificatore:** svolge le attività di verifica.

2.2 Logica di distribuzione dei ruoli

La distribuzione dei ruoli segue questi principi:

1. **Rotazione obbligatoria:** ogni membro del gruppo deve ricoprire almeno una volta ciascun ruolo durante il ciclo di vita del progetto. Si consiglia, idealmente, di effettuare la rotazione dei ruoli con cadenza bisettimanale e in base alla componente del progetto che si decide di progettare e sviluppare.
2. **Equilibrio settimanale:** le attività vengono assegnate in modo da distribuire uniformemente il carico settimanale di 10-20 ore tra i membri del team.
3. **Assegnazione basata sulle competenze:** i membri possono iniziare in ruoli in cui hanno maggiore esperienza, ma devono successivamente ricoprire anche ruoli meno familiari per favorire crescita e versatilità.
4. **Supervisione e verifica:** il Responsabile aggiorna la pianificazione dei ruoli, monitorando il rispetto della rotazione e riequilibrando eventuali carichi non distribuiti equamente.

3 Stesura della Documentazione

3.1 Obiettivi

La documentazione ha lo scopo di garantire la tracciabilità delle decisioni, la coerenza tra le diverse fasi del progetto e l'allineamento costante tra i membri del gruppo e gli stakeholders esterni. Ogni documento deve essere redatto in modo chiaro, completo e conforme agli standard stabiliti.

3.2 Tipologie di documenti

La documentazione prodotta durante il ciclo di vita del progetto si suddivide nelle seguenti categorie:

- **Verbali interni:** redatti dopo ogni meeting interno tra i membri del gruppo. Ogni verbale deve contenere:

- data della riunione e partecipanti;
- argomenti trattati;
- decisioni prese;
- attività assegnate e relative scadenze.

Il segretario di riunione, scelto a rotazione tra i membri, è responsabile della stesura del verbale, che deve essere revisionato dal Responsabile di Progetto entro 24 ore.

- **Verbali esterni:** redatti dopo ogni incontro con il docente o con il referente aziendale. Ogni verbale deve riportare:

- data, orario e partecipanti (inclusi i referenti esterni);
- argomenti discussi, i quali possono richiedere la definizione di documenti specializzati (come nel caso dell'Analisi dei Requisiti);
- osservazioni o richieste del referente esterno;
- decisioni e azioni da intraprendere.

Il segretario di riunione redige il documento sotto la supervisione del Responsabile di Progetto. Se previsto, il verbale deve essere approvato dal referente esterno.

- **Documentazione di Specifica:** la redazione di questi documenti è altamente variabile a seconda della tipologia di documento. Per tal motivo, verrà definita specificamente una volta comprese le possibili tipologie.

3.3 Aggiornamento e revisione

Ogni documento deve riportare:

- numero di versione (es. v1.1);
- data di ultima modifica;
- autore e revisore;

- changelog sintetico.

La revisione deve essere effettuata da un membro diverso da chi ha redatto il documento, secondo il principio della doppia verifica.

3.4 Archiviazione

Tutta la documentazione deve essere salvata in una delle seguenti repository:

- la **repository specifica** del componente o modulo del progetto a cui il documento fa riferimento;
- la **repository centrale** dell'organizzazione SkaraBGroup/DocumentazioneProgetto, dedicata alla raccolta ufficiale di tutta la documentazione approvata.

3.5 Strumenti

Per la redazione e gestione della documentazione si utilizzano:

- **Git** per il versionamento dei documenti;
- **Typst** o **L^AT_EX** per la stesura collaborativa;
- **Issue tracker** (GitHub o GitLab) per associare modifiche documentali a task specifici;
- **Template condivisi** per garantire uniformità formale e stilistica.

3.6 Tempistiche

- Verbali interni: redazione entro 24 ore dal meeting;
- Verbali esterni: redazione e revisione entro 48 ore;
- Documentazione tecnica: aggiornamento entro 2 giorni lavorativi dalla definizione o modifica dell'elemento interessato.

3.7 Controllo di qualità

Tutti i documenti devono essere approvati dal Responsabile di Progetto prima del rilascio o della consegna. Ogni documento deve rispettare i seguenti criteri:

- correttezza formale e linguistica;
- chiarezza e completezza dei contenuti;
- coerenza con il resto della documentazione del progetto.

4 Periodicità ed Organizzazione dei Meeting

4.1 Tipologie di meeting

- **Meeting interni:** riunioni tra i membri del gruppo di progetto.
- **Meeting esterni:** incontri con il docente o con il referente aziendale.
- **Meeting straordinari:** convocati in caso di urgenze, problemi critici o decisioni fuori ciclo.

4.2 Frequenza

- Meeting interni: ogni settimana / ogni due settimane (da definire in fase iniziale).
- Meeting esterni: almeno uno per milestone significativa (es. inizio progetto, avanzamento significativo, revisione con docente/azienda).
- Meeting straordinari quando necessario, su richiesta del Responsabile di Progetto o del docente / azienda.

4.3 Ruoli e responsabilità

- **Organizzatore / Convocatore:** Responsabile di Progetto.
- **Moderatore / Facilitatore:** a rotazione tra i membri o designato per ciascun meeting.
- **Segretario di riunione:** persona incaricata di redigere l'agenda e il verbale (a rotazione).
- **Timekeeper (opzionale):** monitora il rispetto dei tempi previsti.

4.4 Convocazione e Agenda

- L'invito al meeting deve essere inviato almeno 24 ore (o almeno 48 ore) prima dell'orario stabilito.
- L'agenda deve essere preparata e distribuita prima del meeting (es. almeno 24 ore prima), con i punti all'ordine del giorno.
- Prima del meeting ogni membro può proporre punti da aggiungere all'agenda entro una scadenza prefissata (es. 12 ore prima).

4.5 Durata e Orario

- Durata tipica: 30/60 minuti per meeting interno; meeting esterni o di revisione potrebbero durare fino a 90 minuti.
- Orario fisso preferibile **Mattinata del Sabato**, per facilitare la pianificazione del gruppo.
- Inizio puntuale e fine entro l'orario stabilito.

4.6 Output e Follow-Up

- Al termine del meeting viene redatto un verbale (interno o esterno) secondo le regole definite nella sezione “Stesura della Documentazione”.
- Il verbale deve essere distribuito ai partecipanti entro 24/48 ore dal meeting.
- Le azioni / task emerse nel meeting devono essere assegnate con responsabile e scadenza.
- I punti aperti vengono verificati nei meeting successivi.

4.7 Strumenti e Modalità

- Convocazioni via calendario condiviso / piattaforma scelta dal gruppo (Discord per gli Interni, Microsoft Teams / Zoom per quelli Esterni).
- Se il meeting è online, fornire il link all'accesso; se in presenza, specificare luogo (fisico).
- Utilizzo di un template per l'agenda (file condiviso nella documentazione del progetto).

4.8 Monitoraggio e Revisione delle Regole

- Dopo ogni milestone importante o al termine di ciascun periodo definito, il gruppo valuta l'efficacia della periodicità dei meeting (frequenza, durata, partecipanti).
- Se necessario, propone modifiche alla periodicità / modalità e aggiorna il documento.