

Step 7 – Aggiornare la documentazione e formare il team

Analisi della documentazione esistente

Il repository Evo-Tactics contiene numerosi documenti organizzati in varie directory (`docs/`, `docs/checklist/`, `docs/process/`, `docs/presentations/`), oltre a appendici e FAQ. Alcuni punti salienti:

- Le sessioni di onboarding e di training sono già formalizzate. Ad esempio, il documento di onboarding del 18 novembre 2025 indica obiettivi, agenda e materiali da preparare (slide, demo, accessi) e prevede un aggiornamento della documentazione e delle FAQ ¹. Include anche una checklist pre-sessione e azioni post-sessione come raccogliere feedback e aggiornare i documenti ².
- Il registro delle registrazioni di onboarding elenca le sessioni registrate con focus, relatori e link alle registrazioni ³.
- Le FAQ di Support/QA vengono aggiornate dopo ogni retro settimanale. Forniscono risposte rapide e indicano i file da aggiornare (es. token rotation, versioning CLI) ⁴. La sezione "Post-onboarding" stabilisce procedure per verificare l'esecuzione dei workflow e indica come usare i report generati ⁵.
- Checklist e processi (in `docs/checklist/` e `docs/process/`) descrivono procedure operative step-by-step, come la gestione telemetria, la pipeline di handoff web, l'audit dei tratti e la procedura di incidente reporting. Molte checklist sono collegate al tracker (Step 6) tramite caselle di spunta.

Raccomandazioni per l'aggiornamento della documentazione

1. **Centralizzare e normalizzare i documenti**
2. **Indice e naming coerenti:** mantenere aggiornato `docs/00-INDEX.md` tramite il tracker automatico. Aggiungere ogni nuovo documento (GDD, registri di sicurezza, piani sprint) al file `config/tracker_registry.yaml` in modo che compaia nell'indice.
3. **Standardizzare titoli, metadata e marcatori:** definire un template per i documenti (titolo, data, revisione, owner, scopo) e applicarlo a tutte le nuove pagine.
4. **Aggiornare i contenuti con le nuove best practice**
5. Integrare nel Game Design Document le decisioni e i modelli formulati negli step precedenti (loop di gioco, Tri-Sorgente, progressione, sicurezza).
6. Aggiornare le checklist operative per includere i nuovi flussi di CI/CD, i controlli di sicurezza e le procedure di risk assessment introdotte nello Step 5.
7. Documentare la struttura del backlog (Step 3) e la metodologia di sprint/monitoraggio (Step 6) in un nuovo file `docs/process/agile_methodology.md`.
8. **Aggiornare le FAQ e il changelog**

9. Dopo ogni retro settimanale, incorporare le domande emerse nella sezione FAQ, indicando la data e l'owner. Il documento FAQ già specifica che nuove domande vengono raccolte dopo ogni retro ⁴.
10. Aggiornare il changelog (`docs/ideas/changelog.md` o equivalente) e la roadmap con le modifiche principali (nuove feature, correzioni, refactor). Includere un record delle revisioni della pipeline CI/CD e dei controlli di sicurezza.
- 11. Preparare risorse di formazione**
12. **Materiali di onboarding:** consolidare slide, script di demo e registrazioni in una cartella `docs/presentations/assets`. Il documento di onboarding del 18 novembre suggerisce di preparare un deck di 15 slide, un demo script e accessi temporanei ⁶. Creare un template replicabile per le future sessioni.
13. **Guide pratiche:** redigere tutorial sintetici (es. "Come eseguire uno smoke test con CLI", "Come aggiornare il dataset", "Come gestire un incidente di sicurezza") e collegarli nelle FAQ e nei processi.
14. **Registrazioni e link:** mantenere aggiornato il file `docs/presentations/2025-11-onboarding-recordings.md` con le registrazioni e note. È indicato che questa tabella deve essere aggiornata dopo ogni sessione ³.
- 15. Formazione continua del team**
16. Pianificare sessioni ricorrenti (mensili o trimestrali) per presentare le novità del progetto, le revisioni delle pipeline e i risultati di security audit.
17. Organizzare workshop pratici basati su "role play": ad esempio, simulare il lancio della pipeline `daily-pr-summary` o la rotazione dei token di supporto.
18. Definire un programma di onboarding per i nuovi membri che li guidi attraverso il setup dell'ambiente, l'utilizzo dei tool e la lettura dei documenti chiave.
- 19. Gestione collaborativa e feedback**
20. Utilizzare GitHub Discussions o un canale chat dedicato per raccogliere feedback sui documenti e consentire commenti/annotazioni.
21. Applicare la regola "documentation as code": le modifiche alla documentazione seguono lo stesso workflow delle PR (review, approvazione, merge) per garantire qualità e tracciabilità.
22. Alla fine di ogni sessione di formazione o retrospettiva, assegnare l'aggiornamento della documentazione a un owner specifico e verificare che l'azione sia completata entro la successiva riunione.

Prossimi passi

- Creare un template standard per i documenti e predisporre un file `CONTRIBUTING-docs.md` che descriva come creare, formattare e revisionare la documentazione.
- Aggiungere al tracker (tramite `tracker_registry.yaml`) i nuovi file di processo e formazione.
- Schedulare la prossima sessione di onboarding e definire gli argomenti da trattare in base alle novità introdotte negli step precedenti.

Conclusioni

Mantenere la documentazione aggiornata e strutturata è essenziale per la scalabilità di Evo-Tactics. L'integrazione di training formale (onboarding e sessioni di aggiornamento), l'adozione di un template unico e la gestione collaborativa dei documenti renderanno più efficace la condivisione della conoscenza e aiuteranno il team a rimanere allineato su processi, strumenti e obiettivi. L'aggiornamento continuo di FAQ, checklist e guide permetterà inoltre di rispondere prontamente alle domande emergenti durante lo sviluppo ¹.

¹ ² ⁶ 2025-11-18-onboarding.md

<https://github.com/MasterDD-L34D/Game/blob/HEAD/docs/presentations/2025-11-18-onboarding.md>

³ 2025-11-onboarding-recordings.md

<https://github.com/MasterDD-L34D/Game/blob/HEAD/docs/presentations/2025-11-onboarding-recordings.md>

⁴ ⁵ faq.md

<https://github.com/MasterDD-L34D/Game/blob/HEAD/docs/faq.md>