

Step 2 – Definizione e consolidamento del Game Design Document (GDD)

1. Attività svolte

- **Analisi della documentazione interna:** ho consultato i file `docs/DesignDoc-Overview.md`, `docs/tri-sorgente/overview.md`, `docs/Canvas/feature-updates.md`, `docs/11-REGOLE_D20_TV.md`, `docs/20-SPECIE_E_PARTI.md` e `docs/22-FORME_BASE_16.md` per estrarre i principi di design e le meccaniche di gioco.
- **Visione e statement:** il gioco prevede campagne co-op televisive/app in cui i giocatori guidano forme bio-meccaniche della resistenza attraverso missioni procedurali; ogni scelta deve comunicare sia l'impatto immediato (risk/cohesion) sia le conseguenze a lungo termine ¹.
- **Pilastri:** cooperazione situazionale, mutazione significativa, telemetria visibile e narrazione reattiva ².
- **Loop di gioco:** briefing → setup squadra → incursione → eventi dinamici → debriefing → fase Nido ³.
- **Progressione:** livello squadra (slot PI e missioni), prestigio forma, reputazione fazioni e StressWave ⁴.
- **Sistema TV/d20 & companion:** utilizzo di un d20 centrale, clock per eventi a tempo e app companion per drafting e telemetria ⁵ ⁶.
- **Jobs e tratti:** famiglie di job (vanguard, skirmisher, warden, ecc.) con bias nei pacchetti PI, tratti suddivisi in tier T1/T2/T3, prestigio/mutazioni e validatori automatici ⁷.
- **Meccanica Tri-Sorgente:** sistema di scelta di carte che unisce contesto (roll), personalità (MBTI/Enneagramma) e azioni recenti; punteggio delle carte basato su base, componenti di contesto, personalità e azioni, bonus sinergia e penalità per duplicati/esclusioni ⁸.
- **Workflow tratti e quality gates:** processo di aggiunta/aggiornamento dei tratti, sincronizzazione con il glossario, validazione naming, generazione baseline e report di copertura; quality gates prevedono audit tratti-ambienti, matrice specie, gate di bilanciamento e analytics canary ⁹.
- **Aggregazione dei requisiti di design:** ho organizzato le informazioni raccolte per costruire un indice del GDD e identificare dove integrare i diversi documenti e script presenti nel repo (dataset YAML, script di validazione, telemetria, pipeline Tri-Sorgente).

2. Outline proposto del GDD

Il Game Design Document dovrà fungere da riferimento unico per tutto il team, integrando concetti di game design, dati, tool e processi di QA già sviluppati nel repository. La struttura proposta è la seguente:

1. **Visione e obiettivi**
2. Visione generale, statement e pubblico target ¹⁰.
3. Esperienza di gioco (sessioni ~90 minuti, gruppo 3-4 giocatori, onboarding <10 min).
4. **Pilastri di design**
5. Cooperazione situazionale, mutazione significativa, telemetria visibile, narrazione reattiva ².

6. Implicazioni di ciascun pilastro su meccaniche, UI/HUD e flusso narrativo.

7. Loop di gioco

8. Descrizione delle fasi: Briefing, Setup Squad, Incursione (TV/d20), Eventi dinamici, Debriefing e Fase Nido ³.

9. Riferimenti alle interfacce (Mission Control, HUD, companion) e alle tabelle YAML utilizzate per generare missioni e telemetria.

10. Sistemi di progressione e metriche

11. Livello squadra, prestigio Forma, reputazione fazioni, StressWave e loro impatto su spawn, ricompense e difficoltà ⁴.

12. Formula e componenti dell'indice StressWave e integrazione con il monitoraggio telemetrico.

13. Sistema TV/d20 & Companion

14. Adattamento della risoluzione a d20, bande di successo/fallimento, clock eventi e dashboard condivisa ⁵ ⁶.

15. Funzionalità della companion app: drafting di Forme/Job, macro azioni, chat tattica e upload telemetria ⁵.

16. Jobs, tratti e mutazioni

17. Descrizione delle famiglie di job, dei pacchetti PI e dei bias di roll in `data/packs.yaml` ¹¹.

18. Spiegazione dei tratti T1/T2/T3, della progressione delle Forme e dell'uso della telemetria MBTI/Ennea per il seed temperamentale ¹².

19. Regole per specie e parti (slot, budget, sinergie, ibridi, counter) ¹³.

20. Tri-Sorgente (Roll + Personalità + Azioni)

21. Scopo di fornire tre scelte curate più opzione Skip; pipeline di selezione tabella → tiro → fusione pool → scoring → sampling e regole anti-power-creep ⁸.

22. Formula di punteggio e motivazioni di design (agency senza overload, varianza controllata, coerenza build, prevenzione power-creep).

23. Workflow tratti e quality gates

24. Procedura per aggiungere nuovi tratti, sincronizzare glossari, integrare nei biomi e rigenerare baseline ¹⁴.

25. Quality gates: audit tratti-ambienti, matrice specie, gate di bilanciamento e canary analytics ¹⁵.

26. Checklist da integrare nel ciclo Secure SDLC: risk assessment, threat modeling e validazioni automatiche.

27. Stato attuale & roadmap

28. Vertical slice VC in test con tre missioni giocabili; priorità su onboarding, bilanciamento PI/EMA e contenuti Nido ¹⁶ .
29. Collegamento con roadmap operativa e milestone attive (es. Smart HUD & SquadSync, export telemetria, release RC).

3. Prossimi passi per il team

- **Raccolta e consolidamento:** centralizzare tutte le sezioni elencate in un unico documento Markdown (docs/GDD.md), includendo riferimenti ai file YAML, script di validazione e documenti di supporto.
- **Completa la sezione Tri-Sorgente:** integrare la formula e la pipeline nel GDD, definendo parametri (w_roll , w_pers , w_actions , w_syn , w_dup , w_excl) e linee guida di tuning.
- **Verifica dati e coerenza:** verificare che i dataset (data/packs.yaml , data/core/biomes.yaml , data/core/telemetry.yaml , ecc.) siano allineati alle descrizioni del GDD; usare gli script di validazione (traits_validator.py , report_trait_coverage.py) per assicurarsi che tratti e specie rispettino le regole.
- **Aggiornamento continuo:** stabilire un processo per mantenere il GDD aggiornato con i cambiamenti introdotti nelle pull request, sfruttando il workflow di daily PR summary e i log playtest.
- **Condivisione e formazione:** rendere il GDD accessibile al team (es. tramite README e Canvas) e organizzare sessioni di presentazione per garantire che tutti comprendano la visione e le meccaniche.

Il documento GDD consolidato fungerà da guida unica per lo sviluppo, l'analisi e il testing, assicurando che tutte le parti interessate (designers, sviluppatori, QA, data scientists) seguano un linguaggio comune e mantengano allineati software, dataset e pipeline di automazione.

1 2 3 4 5 7 9 10 11 12 14 15 16 DesignDoc-Overview.md

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

6 11-REGOLE_D20_TV.md

https://github.com/MasterDD-L34D/Game/blob/HEAD/docs/11-REGOLE_D20_TV.md

8 overview.md

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

13 20-SPECIE_E_PARTI.md

https://github.com/MasterDD-L34D/Game/blob/HEAD/docs/20-SPECIE_E_PARTI.md