**REALIZZAZIONE DI UN PROGETTO INFORMATICO**

# 

# **RUOLI E COMPITI**

Per il progetto **Spotifinto** sono necessari diversi ruoli per svolgere compiti differenti.

| **Nome** | **Ruolo** | **Compiti principali** |
| --- | --- | --- |
| Pietro Salvatore Colosimo | Project manager | Coordinamento generale, gestione tempi e risorse, comunicazione con stakeholders |
| Bruno Mars | DevOps | Monitoraggio ambienti di produzione, sviluppo infrastruttura e gestione dei backup. |
| Donald Trump | Backend developer /  Database administator | Realizzazione API, gestione database, logica di riproduzione musicale |
| Giorgia Meloni | Frontend developer | UI/UX Web e Mobile, integrazione API |
| Vladimir Putin | Designer | Progettazione interfaccia utente, branding, mockup |
| Volodymyr Zelens'kyj | Tester | Verifica funzionalità, test di usabilità, bug tracking |
| Elon Musk | Responsabile marketing | Pianifica e attua strategie promozionali, cura la presenza online e comunica il lancio |

# 

# **MODALITÀ DI COMUNICAZIONE E COLLABORAZIONE**

* **Strumenti usati:** Microsoft Teams per la comunicazione, Trello per il task management, GitHub per la gestione versionata del codice.
* **Meeting settimanali:** ogni lunedì alle 17:00.
* **Revisioni intermedie:** demo ogni 2 settimane con presentazione dello stato di avanzamento.

# 

# **OBIETTIVO DEL PROGETTO**

**Descrizione del problema nel mercato:** molti utenti cercano una piattaforma musicale semplice, intuitiva e con un buon equilibrio tra contenuti gratuiti e premium. Le alternative esistenti possono risultare complesse o costose.

**Funzioni offerte dalla piattaforma Spotifinto:**

* Riproduzione musicale in streaming.
* Playlist personalizzate e suggerimenti basati sugli ascolti.
* Versione gratuita con pubblicità, versione premium senza pubblicità e con audio ad alta qualità.
* App mobile e Web compatibili.
* Supporto per podcast.

**Vantaggi:**

* Accesso rapido alla musica.
* Scoperta di nuovi artisti.
* Esperienza fluida e personalizzata.
* Modello freemium con possibilità di upgrade.

# **COSTI E RISORSE**

* **Hardware**: €3.000 (server, dispositivi di test).
* **Software**: €1.000 (licenze, strumenti di sviluppo).
* **Licenze musicali e diritti d’autore**: €20.000.
* **Personale**:

| **Ruolo** | **Seniority** | **Costo orario (€)** | **Ore stimate** | **Totale (€)** |
| --- | --- | --- | --- | --- |
| Project Manager | Senior | 50 | 120 | 6.000 |
| Backend Developer/DBA | Mid-Level | 35 | 200 | 7.000 |
| Frontend Developer | Mid-Level | 35 | 200 | 7.000 |
| Designer UI/UX | Junior | 25 | 100 | 2.500 |
| QA Tester | Junior | 25 | 100 | 2.500 |
| DevOps / Infra Engineer | Mid-Level | 40 | 100 | 4.000 |
| **Totale personale** |  |  |  | **29.000** |

**Budget disponibile**: €60.000

**Possibili finanziamenti**: Crowdfunding, sponsor musicali, startup incubator.

# **RISCHI E OPPORTUNITÀ**

**Rischi principali:**

* Concorrenza elevata (Spotify, Apple Music).
* Costi per i diritti musicali.
* Problemi tecnici per la complessità della piattaforma.

**Strategie di mitigazione:**

* Offrire una funzione innovativa esclusiva.
* Test accurati prima del rilascio.
* Partnership con etichette indipendenti.

**Opportunità:**

* Ampliare il servizio a podcast.
* Collaborazioni con artisti emergenti.
* Espansione internazionale.

# **PIANIFICAZIONE DEL PROGETTO**

**Previsioni per le durate delle diverse fasi del progetto**

| **Fase** | **Durata** | **Attività** |
| --- | --- | --- |
| 1. Analisi | 1 settimana | Studio di mercato, raccolta requisiti |
| 2. Progettazione | 2 settimane | Mockup, architettura software |
| 3. Sviluppo | 6 settimane | Backend + Frontend, interfacce mobile e Web |
| 4. Testing | 2 settimane | QA, bug fixing, test di carico |
| 5. Rilascio | 1 settimana | Deploy versione beta, feedback utenti |

**Diagramma di Gantt**



**Calendario**

* **Inizio:** 1 gennaio 2025
* **Fine:** 31 Marzo 2025

# **DOCUMENTAZIONE**

**Stack tecnologico**

* **Frontend:** Laravel Blade
* **Backend:** Laravel
* **Database:** MySQL (con Eloquent ORM)
* **Repository:** GitHub
* **Marketing:** Google Ads
* **Analisi del traffico:** Google Analytics

**Specifiche dell'infrastruttura hardware**

* **Sistema operativo:** Proxmox (Linux)
* **Gestione dei servizi:** Container Docker
* **DNS:** Cloudflare
* **Web server:** NGINX
* **CPU:** AMD Ryzen Threadripper 7995x
* **RAM:** 128GB RAM DDR5
* **Storage:** 10TB SSD

**Entità principali del Database**

* **User** – l’utente registrato
* **Artist** – chi pubblica musica
* **Track** – il singolo brano
* **Album** – collezione di tracce
* **Playlist** – raccolta personalizzata di tracce
* **Playback** – cronologia di ascolto

**Relazioni tra le entità**

* **User crea Playlist**: 1:N (un utente può avere più playlist)
* **Playlist contiene Track**: N:N (una playlist ha più tracce, una traccia può stare in più playlist)
* **Artist pubblica Album**: 1:N (un artista può pubblicare più album)
* **Album contiene Track**: 1:N (un album contiene più tracce)
* **Track è di Artist**: N:1 (ogni traccia ha un artista)
* **User ascolta Track**: N:N (riproduzioni) con timestamp

**Organizzazione del codice**

Il progetto possiede più rami di sviluppo del codice, ognuno con obiettivi diversi:

* **ramo di sviluppo:** è quello più soggetto a modifiche frequenti del codice, spesso sommerso di bug, viene testato sul momento il codice scritto e verificato con dati fittizi
* **ramo di testing:** è dedicato all’analisi di bug nel codice quando le nuove feature sono state ormai scritte, i test vengono fatti con dati reali presi da una copia del Database principale
* **ramo di produzione:** è il ramo ufficialmente accessibile al pubblico, contiene il codice ormai confermato essere a prova di bug e offre un’esperienza utente di qualità.

**Gestione backup**

**Frequenza:** giornaliera per DB, settimanale per il file system, continua per il codice.

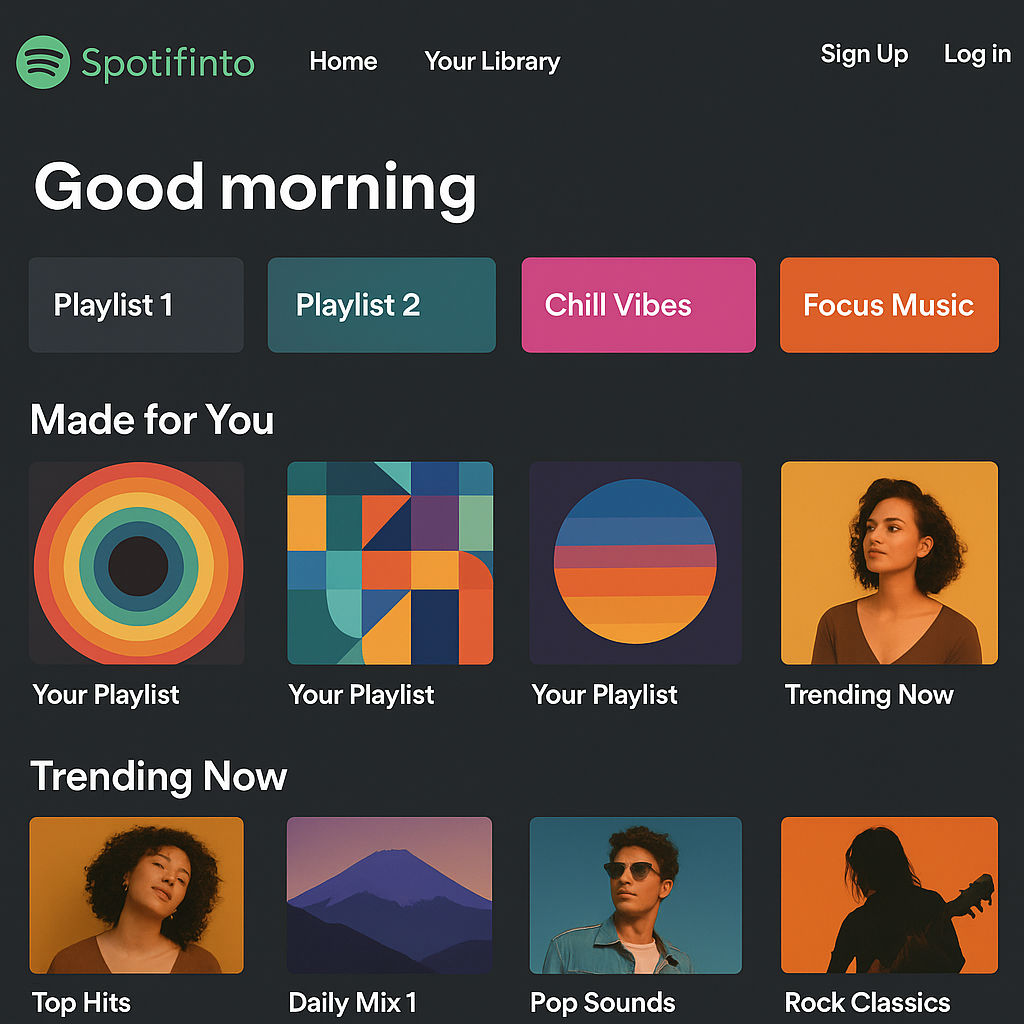
**Retention:** DB per 30 giorni, file system per 14 giorni.

**Marketing e Analisi del traffico**

Il progetto ormai accessibile al pubblico deve poter massimizzare il traffico di utenti, per farlo è stato scelto Google Ads come strumento per il marketing, collaborando con gli artisti per promuoverli ed attirare futuri utenti, per ottimizzare i costi vengono eseguiti ripetuti “AB test” per identificare i più performanti.

Con Google Analytics siamo capaci di tenere traccia di tutto il traffico degli utenti sulla nostra piattaforma e la loro provenienza.

**Interfaccia**

****

# **CONCLUSIONI**

# Il progetto Spotifinto è un esempio di applicativo cui sono stati analizzati tutti gli aspetti, a partire dall’hardware che ospiterà il nostro programma, al suo sviluppo, e fino alla sua sponsorizzazione.

Per la creazione di progetti simili è fondamentale pianificare e organizzare i vari punti del progetto per separare gli incarichi ai diversi membri del team, sia per minimizzare gli imprevisti, ridurre i costi e evitare dilatamenti nelle attese, ma anche per avere un prodotto qualitativamente all’altezza delle esigenze del mercato attuale.