

## Track Finder – Kurz-Dokumentation (Bullet Points)

### Dark Theme UI for Track Finder – Figma Make

#### 1) Projektübersicht

- Web-App zur Musik-Discovery: Tracks finden, Vibes (Energy, Valence, BPM) verstehen, ähnliche Songs erkunden.
- Klare, moderne UI; responsiv für Mobile und Desktop.
- Zielgruppe: Musikfans, DJs, Neugierige mit Fokus auf Stimmung/Energie.

#### 2) Ziele

- Intuitive Suche mit schneller Rückmeldung.
- Visuelle Darstellung musikalischer Eigenschaften.
- Empfehlungslogik für ähnliche Titel.
- Erweiterbare Architektur und konsistentes Design-System.

#### 3) Technologie (High-Level)

- Frontend-Framework + TypeScript, Vite als Build-Tool.
- Styling per Utility-CSS & Design-Tokens; Komponentenbibliothek für UI-Bausteine.
- Icons/Animationen/Toasts für moderne UX.
- Linting/Formatting für Code-Qualität (intern).

#### 4) Projektstruktur (vereinfacht)

- Komponenten (Suche, TrackItem, VibeBars, Review/Album-Cards, Modals, Design-Specs).
- Globale Styles/Design-Tokens.
- Einstiegs- und Routing-Komponenten.

#### 5) Kern-Features

- **Suche:** Eingabe + Enter; Ergebnisliste für Tracks/Alben; schnelle Interaktion.
- **Vibe-Analyse:** Balken/Indikatoren für Energy, Valence, BPM; sanfte Animationen.
- **Ähnliche Titel:** Liste basierend auf Vibe-Ähnlichkeit.
- **Reviews & Alben:** Tab-Wechsel; Karten/Grids; Album-Detail mit Tracklist.
- **Dynamischer Hintergrund:** wechselt sanft abhängig von Energy/Valence.

#### 6) Wichtige UI-Bausteine

- **Suchleiste:** klare Interaktion; Enter-Trigger; optional Mikrofon-Icon (Erkennung später).
- **TrackItem:** Cover, Titel, Artist, Aktionen (z. B. „Ähnlich“).

- **VibeBar:** farbcodierte Fortschrittsbalken (0–100 %).
- **AlbumCard & Detailmodal:** Grid-Ansicht und Detail-Overlay.
- **Design-Specs:** lebende Dokumentation von Farben, Schrift, Komponenten.

## 7) Design-System

- **Farben:** hell/freundlich mit deutlichem Akzent; hohe Lesbarkeit (AA-Kontrast).
- **Typografie:** System-Fonts; klare Hierarchie (h1–p).
- **Spacing/Radius/Schatten:** konsistente Skalen; subtile Elevation.
- **Motion:** kurze Transitions, sanfte Gradients; zurückhaltende Animationen.

## 8) User-Flows

- **Discovery:** Suche → Ergebnisse → Vibe → Ähnliche.
- **Reviews & Alben:** Tabs → Karten/Grids → Detail-Modal.
- **Specs:** Überblick über Design-Tokens und UI-Muster.

## 9) Interaktions-Details

- Enter löst Suche aus; Buttons haben klare Zustände.
- Smooth-Scroll zu Vibe-Sektion nach „Ähnlich“.
- Toast-Meldungen für Erfolg/Info/Fehler.
- Fokus-Ringe und klare States für Tastatur-Nutzung.

## 10) Responsive & A11y

- Mobile-First; adaptive Grids (2–4 Spalten).
- Mindest-Touch-Flächen; semantische Elemente; Alt-Texte.
- Sichtbare Focus-Outlines; ausreichende Kontraste.
- Optionale Rücksicht auf „reduce motion“.

## 11) Installation & Nutzung (kurz)

- Node + Paketmanager installieren.
- Abhängigkeiten installieren; Dev-Server starten.
- Build/Preview für Deployment verfügbar.
- (API-Keys erst bei echter Integration nötig.)

## 12) Roadmap / Erweiterungen

- Echte Musik-API (Suche, Previews, Auth/Playlists).
- Mehr Audio-Features (Danceability, Acousticness, ...).
- Personalisierung (Favoriten, History, gespeicherte Suchen).

- Social-Funktionen (Teilen, kollaborative Playlists).
- Qualität: Tests (Unit/E2E), Error-Monitoring, Analytics.
- Performance: Lazy-Loading, Virtual Scrolling, Offline.

### 13) Fazit

- Klarer Fokus auf **schnelle Discovery** und **visuelle Vibes**.
- **Skalierbares Design-System, modulare Architektur**.
- Bereit für die schrittweise Anbindung echter Musik-Daten und Personalisierung.