

Spotify-like Web App – Full Architecture & Feature List

1. Tech Stack Overview

Frontend:

- Next.js (React + TypeScript) for SSR/SSG and SPA behavior.
- Tailwind CSS with a component library (e.g. Headless UI/Radix UI style).
- React Query (TanStack Query) for server state.
- Zustand/Recoil/Jotai for global player state.
- HLS.js + HTML5 Audio / Web Audio API for playback & visualizations.
- OAuth2 / OIDC-based auth integration with backend.
- PWA features for installable app and offline caching of some metadata.

Backend:

- Node.js + TypeScript.
- NestJS for modular architecture, DI, microservices support.
- REST for external APIs, gRPC or message bus (Kafka/NATS/RabbitMQ) for internal communication.
- API Gateway (Nginx/Envoy/Kong) for routing, rate-limiting, auth at edge.

Databases & Storage:

- PostgreSQL for relational data (users, playlists, subscriptions, etc.).
- Redis for caching and sessions.
- Elasticsearch/OpenSearch for full-text search & autocomplete.
- ClickHouse or similar data warehouse for analytics events.
- S3-compatible object storage for audio and images.
- CDN (CloudFront/Fastly/Cloudflare) for asset and audio delivery.

Infrastructure & DevOps:

- Docker for containerization.
- Kubernetes (GKE/EKS/AKS) for orchestration.
- Vault / Secrets Manager for secrets.
- CI/CD via GitHub Actions / GitLab CI.
- Observability via Prometheus + Grafana, central logging, and tracing with OpenTelemetry.

2. Microservices (High-Level)

1) API Gateway / Backend for Frontend (BFF)

- Single HTTP endpoint for clients.
- Handles auth, rate-limiting, schema validation.
- Aggregates data from underlying microservices.

2) Auth & Identity Service

- User registration/login, password reset, social logins.
- JWT/opaque token issuance and refresh tokens.
- Device sessions management.
- Core tables: users, credentials, sessions, devices.

3) User Profile Service

- Manages display name, avatar, profile settings.
- User preferences (language, explicit content filter, theme).
- Core tables: user_profile, user_settings.

4) Catalog Service (Music Metadata)

- Stores metadata for tracks, albums, artists, genres, moods.
- Manages relationships: album-track, artist-track, etc.
- Core tables: artists, albums, tracks, genres, moods, track_audio_files.

5) Search & Discovery Index Service

- Indexes tracks, albums, artists, playlists, podcasts.
- Autocomplete suggestions.
- Powered by Elasticsearch/OpenSearch.

6) Library & Playlist Service

- Implements “Your Library” (liked songs, saved albums, followed artists).
- Manages playlists (create, edit, delete, reorder, collaborative, privacy).
- Core tables: playlists, playlist_tracks, user_liked_tracks, user_saved_albums, user_followed_artists, user_followed_playlists.

7) Playback & Session Service

- Stores playback state (current track, position, shuffle, repeat, queue).
- Manages playback sessions and active devices.
- Provides WebSocket channel for real-time player updates.
- Core tables: devices, playback_sessions, playback_queue.

8) Streaming / Media Delivery Service

- Generates secure signed URLs to audio HLS manifests/segments.
- Enforces subscription tier (bitrate, ads vs ad-free).
- Logs stream-start events for analytics.

9) Recommendation & Personalization Service

- Personalized homepage and playlists (Discover Weekly-like, Daily Mix, Radios).
- Computes user and item embeddings using listening history and audio features.
- Serves recommendations by seeds (artists, tracks, genres).

10) Social & Activity Service

- Following system between users.
- Friend activity feed (“X is listening to...”).
- Collaborative playlist permissions and activity.
- Link shortening and share tracking.

11) Subscription & Billing Service

- Manages subscription plans (Free, Premium, Family, Student).
- Integrates with payment gateways (Stripe/Razorpay/etc.).
- Handles trials, renewals, invoicing, and cancellations.
- Core tables: plans, subscriptions, invoices, payment_methods.

12) Ads Service (Free Tier)

- Manages ad campaigns, creatives, and targeting rules.
- Decides which audio ad to serve.
- Tracks impressions, clicks, frequency caps.
- Core tables: ad_campaigns, ad_creatives, ad_impressions, ad_events.

13) Content Ingestion & Creator Portal Service

- Artist/label portal: upload tracks, edit metadata, manage catalogs.
- Podcast ingestion and hosting (episodes, RSS import).
- Core tables: uploads, rights, creator_profiles, podcasts, episodes.

14) Media Processor Service

- Background worker for transcoding uploads (FFmpeg).
- Creates multiple bitrates and HLS manifests.
- Generates waveforms, loudness normalization, and audio analysis.
- Writes processed files to object storage and notifies Catalog/Streaming services.

15) Analytics & Events Service

- Central ingestion point for all user and system events.
- Stores events in ClickHouse/warehouse.

- Provides aggregated analytics for internal dashboards and creator analytics.
- Feeds Recommendation Service with listening data.

16) Notification Service

- Sends emails, push notifications, and in-app notifications.
- Supports notification templates and user preference management.
- Digest emails for new releases, playlist updates, etc.

17) Admin & Backoffice Service

- Admin panel for staff: user search, content management, moderation.
- Feature flags and configuration (A/B testing).
- System metrics and health views.

3. Data Storage Layout

PostgreSQL:

- Users, profiles, subscriptions, playlists, follows.
- Metadata for tracks, albums, artists (or partially split with a separate catalog DB if needed).

Redis:

- Authentication/session cache.
- Playback state cache (optionally).
- Rate limits and hot metadata caches (trending playlists, top charts).

Elasticsearch/OpenSearch:

- Full-text index of tracks, artists, albums, playlists, and podcasts.
- Autocomplete index and suggestion index.

ClickHouse / Data Warehouse:

- Playback events (play, pause, skip, complete).
- Search events and clicks.
- Aggregations: daily active users, time listened, streams per track, etc.

Object Storage (S3-like) + CDN:

- HLS segments and manifests per track and bitrate.
- Cover images, avatars, thumbnails.
- Podcast episodes.

Message Queue (Kafka/NATS/RabbitMQ):

- track_uploaded, track_encoded, playback_event, subscription_updated, notification events.

4. Frontend Structure (Next.js)

Key pages/routes:

- Public:
 - / → Landing/marketing page.
 - /login → Auth.
 - /signup → Auth.
 - /artist/:id → Public artist view.
 - /album/:id → Public album view or preview.
- Authenticated app:
 - /home → Personalized home feed.
 - /search → Full search UI.
 - /collection/tracks → Liked songs.
 - /collection/playlists → User playlists.
 - /collection/albums → Saved albums.
 - /collection/artists → Followed artists.

- /playlist/:id → Playlist details & play.
- /queue → Queue detail (could also be a modal).
- /settings/account → Account info.
- /settings/subscription → Plan & billing.
- /settings/privacy → Content & privacy preferences.

- Creator portal:
 - /creator/dashboard
 - /creator/uploads
 - /creator/analytics

- Admin:
 - /admin → Admin dashboard.
 - /admin/users
 - /admin/content
 - /admin/flags

Global UI components:

- Persistent bottom player bar.
- Left navigation (Home, Search, Your Library).
- Top bar with search, profile dropdown.
- Queue / Now Playing side panel.

5. Feature Checklist (Spotify-like)

A. User & Auth

- Email/password auth & social login.
- Email verification & password reset.
- Profile editing (name, avatar).
- Preferences: explicit content, language, theme.
- Device management and sign-out from all devices.

B. Playback & Player

- Play, pause, next, previous, seek.
- Shuffle & repeat (track, playlist/album).
- Volume & mute.
- Playback queue with reorder & removal.
- Cross-device control and sync.
- Optional crossfade/gapless playback.
- Auto-play recommendations when playlist ends.

C. Library & Playlists

- Liked songs collection.
- Create/edit/delete playlists.
- Public/private/unlisted playlists.
- Collaborative playlists.
- Save albums to library.
- Follow artists and playlists.
- Sorting and filtering in library.

D. Catalog & Browsing

- Detailed artist pages (top songs, albums, related artists).
- Album pages (track list, total duration).
- Browse by genres & moods.
- New releases and charts.
- Podcasts & episodes (optional if you want full parity).

E. Search

- Unified search (tracks, artists, albums, playlists, podcasts).

- Autocomplete suggestions.
- Search history and clear history.
- Filter by content type.

F. Recommendations & Discovery

- Personalized home feed.
- "Because you listened to X" rows.
- Song Radio and Artist Radio.
- Auto-generated Daily Mix-style playlists.
- Weekly discovery playlist (like Discover Weekly).

G. Social

- Follow users (friends).
- Friend activity feed.
- Collaborative playlists with multiple editors.
- Share links for tracks, albums, artists, and playlists.
- Optional embeddable mini-player for shared content.

H. Subscription / Monetization

- Free tier (ad-supported with lower bitrate).
- Premium tier (ad-free, high bitrate, offline for native apps).
- Payment integration (Stripe/Razorpay/etc.).
- Plan change flows, cancellations, proration behavior.
- Billing history and invoices.

I. Ads (Free Tier)

- Audio ad slots between tracks.
- Targeting rules (location, device, time).
- Frequency capping and rotation.
- Campaign analytics (impressions, clicks, conversions).

J. Creator / Label Portal

- Artist/label onboarding and verification.
- Track and album upload flows.
- Metadata management (ISRC, lyrics, credits).
- Bulk uploads for labels.
- Analytics dashboard (streams, listeners, countries, playlists).

K. Admin / Moderation

- Search / inspect users and content.
- Ban/suspend accounts.
- Takedown and restore content.
- Abuse reports queue and resolution tools.
- Feature flags and A/B tests configuration.
- System status and error monitoring dashboards.

L. Analytics & Reporting

- Event tracking for plays, skips, completes, likes, searches, shares.
- DAU/MAU, retention cohorts, time spent listening.
- Track- and playlist-level performance analytics.
- Creator- and label-facing analytics.
- Internal dashboards for product and ops teams.

M. Security & Compliance

- HTTPS everywhere, secure cookies, HSTS.
- Input validation and rate limiting at the gateway.
- RBAC for admin and internal tools.
- GDPR-style data export and account deletion.
- Audit logging for sensitive actions.

This document summarizes the architecture, core services, data layout, frontend structure, and feature checklist required to build a large-scale Spotify-like web application capable of serving 100,000+ daily users.