## **1) Análisis del Caso: NewSound**

### **1.1 Contexto**

• Tendencia: consumo de música vía streaming; proliferación de artistas independientes; baja visibilidad local.  
 • Problemas actuales  
 – Artistas: dificultad para publicar, segmentar audiencias y monetizar; poca analítica accesible.  
 – Oyentes: sobreoferta; difícil descubrir talentos emergentes; poca conexión social real.

• Entorno: académico (DUOC UC), semestre acotado; recursos de software libre; trabajo mixto remoto/presencial.

### **1.2 Problema y oportunidad**

• Problema: falta un espacio web **simple, local y social** que permita a artistas **subir música** y a oyentes **descubrir y reseñar** fácilmente.  
 • Oportunidad: combinar **streaming + comunidad + tienda básica** en una SPA con back-end robusto.

### **1.3 Objetivo del producto**

Construir un **MVP funcional** de plataforma de streaming social para descubrir música emergente, habilitando **upload**, **reproducción**, **reseñas** y **merchandising** básico.

### **1.4 Hipótesis y supuestos**

• Si facilitamos el **upload** y damos **visibilidad** (home + trending + reseñas), los artistas suben ≥2 tracks/mes.  
 • Si proveemos **filtros sencillos** y un **player estable**, los oyentes escuchan ≥60 min/semana.  
 • **Actores/roles**: **Usuario (artista/fan/crítico)**, **Administrador**, **Editorial** y **Soporte**. Para el **MVP**, *Usuario* y *Administrador* cubren el 90–95% de los casos; **Editorial** y **Soporte** pueden mapearse como sub-roles del administrador o incorporarse en fases posteriores.

### **1.5 KPIs de éxito (académicos/funcionales)**

• **North Star Metric (NSM)**: minutos de escucha semanales por usuario activo.  
 • ≥ 30 tracks subidos (piloto); **p50 home < 2.5 s**; ≥ 20 usuarios de prueba con ≥ 60 min/semana; ≥ 10 reseñas publicadas.

### **1.6 Alcance**

• **Incluye (MVP)**: auth/roles, perfil artista, upload, player, reseñas, búsqueda simple, home/trending, admin mínimo, merch básico, **integración con Cloudinary (media/CDN) y Supabase (DB)**.  
 • **Excluye ahora**: motor de recomendación por ML, pagos reales, apps móviles nativas, panel editorial avanzado.

### **1.7 Riesgos y mitigación**

• **Tiempo limitado** → priorizar Must (MoSCoW); rebanadas verticales por sprint.  
 • **Rendimiento streaming** → **Cloudinary (CDN) con HTTP 206/Range y caching**; Nginx como reverse proxy para la **API Django**; optimizar bitrates e imágenes.  
 • **Seguridad** → sesiones HttpOnly + CSRF; RBAC; validación y sanitización de inputs; límites de subida.