

# 動画CM分析アプリ - 残課題一覧

作成日: 2026-02-08 / プロジェクト: movie-analysis-d05fb

今回の対応で完了した項目	
項目	カテゴリ
APIキーのAES-GCM暗号化（DB保存時の保護）	セキュリティ
ストレージパス・トラバーサル脆弱性の修正	セキュリティ
ログインブルートフォース防御	セキュリティ
Firebase Hostingセキュリティヘッダー追加	セキュリティ
エラーメッセージからのAPIキー漏洩防止	セキュリティ
ファイルタイプ/入力値バリデーション強化	セキュリティ
CSVエクスポートの型不一致修正 / 重複コード統一	バグ修正
transcriptionsテーブルのedited/edited_atカラム追加	スキーマ
ストレージ使用量表示コンポーネント	機能追加
書き起こしセグメントのインライン編集UI	機能追加
未使用コード/アセットの削除（VideoTag型、app-guide.html）	整理
.gitignore強化	セキュリティ

1. アーキテクチャ改善

課題	優先度	詳細
Supabase Auth の導入	高	<p>現在はクライアントサイドのみのパスワードゲート。Supabase Authに移行することで:</p> <ul style="list-style-type: none"><li>RLSポリシーを <code>auth.uid()</code> ベースに厳格化できる</li><li>DB直接アクセス (Supabase anon key悪用) を防止</li><li>ユーザー別の操作ログが取れる</li></ul> <p>影響範囲: <code>PasswordGate.tsx</code>, <code>supabase.ts</code>, <code>supabase-schema.sql</code> (RLSポリシー全書き換え)</p>
RLSポリシーの厳格化	高	<p>全テーブルが <code>USING (true)</code> で全操作許可。Auth導入後に制限:</p> <ul style="list-style-type: none"><li><code>videos</code> : 認証ユーザーのみ READ/WRITE</li><li><code>settings</code> : 認証ユーザーのみ READ、管理者のみ WRITE</li><li><code>storage.objects</code> : 同様に制限</li></ul> <p>前提: <i>Supabase Auth</i>導入が必要</p>
バンドルサイズ最適化	低	<p>現在 959KB (gzip 274KB)。Viteの chunk size警告が出ている。</p> <ul style="list-style-type: none"><li><code>React.lazy()</code> + <code>dynamic import</code>でページ単位のコード分割</li><li><code>recharts</code> を軽量チャートライブラリに置き換え検討</li><li><code>vite.config.ts</code> で <code>manualChunks</code> 設定</li></ul>

2. 機能改善

課題	優先度	詳細
----	-----	----

コンバージョンCSVエクスポートボタン	中	<code>exportConversionCSV()</code> 関数は <code>utils/csv.ts</code> に実装済みだが、 <code>VideoDetailPage</code> または <code>AnalysisPage</code> にエクスポートボタンが未設置。  対象ファイル: <code>VideoDetailPage.tsx</code> または <code>AnalysisPage.tsx</code>
動画タグ機能	低	動画にタグを付けてフィルタリング・分類を強化する機能。 <ul style="list-style-type: none"><li>DBテーブル <code>video_tags</code> の作成</li><li>タグ付け/削除のAPI実装</li><li>動画一覧でのタグフィルタUI</li></ul>
動画サムネイル生成	低	<code>getVideoThumbnailUrl()</code> は静的 SVG プレースホルダーを返している。動画の最初のフレームからサムネイルを生成する機能に置き換えることで一覧性が向上。 <ul style="list-style-type: none"><li>Canvas API で動画の特定フレームをキャプチャ</li><li>生成したサムネイルを Storage に保存</li></ul>

### 3. インフラ・運用

課題	優先度	詳細
Firebase Hosting へのデプロイ	高	今回の全変更を本番に反映するには: <pre>firebase deploy --only hosting</pre> セキュリティヘッダーの適用にもデプロイが必要

Supabase スキーマ変更の適用	高	<p><code>transcriptions</code> テーブルに <code>edited</code> / <code>edited_at</code> カラムを追加する ALTER文の実行が必要:</p> <pre>ALTER TABLE transcriptions   ADD COLUMN IF NOT EXISTS edited BOOLEAN NOT NULL DEFAULT false; ALTER TABLE transcriptions   ADD COLUMN IF NOT EXISTS edited_at TIMESTAMPTZ;</pre> <p><i>Supabase Dashboard &gt; SQL Editor</i> で実行</p>
backend/ デレクトリの整理	低	<p>Supabase移行前のFastAPI + Whisperバックエンド。現在は使用されていない。</p> <ul style="list-style-type: none"><li>アーカイブまたは別リポジトリに移動</li><li>必要であれば <code>backend-legacy/</code> にリネーム</li></ul>

4. 既知の制約事項

- クライアントサイド認証の限界 - パスワードゲートはUI保護のみ。Supabase anon keyはJSバンドルに含まれるため、直接APIアクセスは防げない（Supabase Auth導入で解消）
- API暗号化キーのバンドル同梱 - AES暗号化キーは `VITE_APP_PASSWORD_HASH` から導出されるため、バンドル解析で復元可能。Supabase Auth + サーバーサイド暗号化で根本解決
- ファイルサイズ50MB制限 - Supabase Storage free tierの制約。クライアントサイドのみでの検証（バイパス可能）
- ストレージ容量1GB制限 - Supabase Storage free tierの制約。Pro planで250GB