

要件定義書

アプリ名

- 「日替わりおすすめMV再生アプリ」

アプリの目的

- ユーザがアプリを開いたら、その日のおすすめのMVがYoutubeから取得され表示される
- シンプルでユーザにとって「今日の未知の1本のMV」に出会える体験を提供する
- 全てのユーザが「同じ今日の1本のMV」を閲覧する

機能要件

フロントエンド(Next.js)

1. ホームページ

- 今日のおすすめMVを表示(Youtube IFrame APIを利用して動画埋め込み)
- ワンクリックで再生
- おすすめMVと同じチャンネルにある動画から、関連動画も取得して表示

2. おすすめ動画履歴ページ

- 過去の日付とそれに対応するMVの一覧を取得して表示

3. ユーザ動画視聴履歴ページ

- ユーザが視聴した動画一覧を表示する

4. お気に入りページ

- ユーザがお気に入りに保存した動画を表示

5. ユーザー認証

- ログイン/ログアウト

バックエンド(Laravel API)

1. おすすめMV候補自動収集機能

- トレンド動画収集: YouTubeトレンド (regionCode=JP, videoCategoryId=10) を1日1回取得
- 特定チャンネル収集: 「Catch Up Japan チャンネル」から取得
- 保存・更新: `videos`テーブルにUpsertで保存(重複防止)
- 定期実行: Laravel Schedulerにて毎日深夜（例: 03:00）に自動実行

2. 日替わりおすすめMV決定機能

- 自動選定: 取得済み動画から当日分をランダムに1件取得
- 選定ロジック: トレンド70%/特定チャンネル30%の割合で抽出

- 重複回避: 直近7日間に選定済み動画は除外
- 保存: `daily_recommendations` テーブルに当日分を記録

3. おすすめMV履歴API

- 指定範囲の日付に対応するMV一覧を返す

4. ユーザ視聴履歴API

- ログインユーザーのみ
- ユーザが視聴した動画一覧を返す

5. お気に入りAPI

- ログインユーザーのみ
- お気に入りに指定した動画の「保存/削除/一覧取得」を行う

6. ユーザー管理

- 認証(Laravel Sanctum(APIトークン機能))
- プロフィール

データベース設計

テーブル構成一覧

テーブル名	役割
<code>users</code>	ユーザ情報(ログイン・認証)(JP限定)
<code>videos</code>	Youtube動画キャッシュ
<code>artists</code>	アーティスト(投稿者)
<code>daily_recommendations</code>	日替わりおすすめMV履歴
<code>user_histories</code>	各ユーザーの視聴履歴
<code>userFavorites</code>	各ユーザーのお気に入り登録
<code>channels</code>	Youtubeチャンネル情報(任意・拡張用)

テーブル定義(詳細)

`users` テーブル ユーザー情報

カラム名	型	NULL	備考
<code>id</code>	BIGINT UNSIGNED	NO	主キー
<code>name</code>	VARCHAR(100)	NO, default 'ゲスト'	表示名
<code>email</code>	VARCHAR(255)	YES, UNIQUE	ログイン用メールアドレス(匿名時はNULL)
<code>password</code>	VARCHAR(255)	YES	Laravelの <code>bcrypt</code> で保存

カラム名	型	NULL	備考
country	CHAR(2)	NO, default 'JP'	国コード(日本限定)
is_admin	boolean	NO	管理者フラグ
created_at	TIMESTAMP	NO	登録日時
upadted_at	TIMESTAMP	NO	更新日時

- 将来的にGoogle OAuthログインも追加可能

videos テーブル Youtube動画キャッシュ

カラム名	型	NULL	備考
id	BIGINT UNSIGNED	NO	主キー
youtube_id	VARCHAR(20)	NO, UNIQUIE	Youtube動画ID
artist_id	BIGINT UNSIGNED	NO	アーティスト(投稿者) artists.id 外部キー
title	VARCHAR(255)	NO	動画タイトル
description	TEXT	NO	説明文
channel_id	VARCHAR(50)	YES	チャンネルID(外部キーにしても良い)
channel_title	VARCHAR(255)	NO	チャンネル名
thumbnail_url	VARCHAR(255)	NO	サムネイルURL
published_at	DATETIME	YES	Youtube上の公開日時
view_count	BIGINT	YES	視聴回数(Youtube Data APIから取得)
like_count	BIGINT	YES	高評価数(Youtube Data APIから取得)
source_type	ENUM('trend', 'channel', 'mixed')	NO	どのソースから取得したか(default: 'youtube')
created_at	TIMESTAMP	NO	登録日時
upadted_at	TIMESTAMP	NO	更新日時(サムネイル、タイトル、説明文、視聴回数、高評価数などの更新)

- Youtube Data APIから取得した動画をキャッシュして再利用

[artists](#) テーブル アーティスト(投稿者) (将来的に「アーティスト別MV一覧を表示したい」など拡張したい時のために、[videos](#) テーブルから分離(正規化))

カラム名	型	NULL	備考

カラム名	型	NULL	備考
id	BIGINT UNSIGNED	NO	主キー
name	VARCHAR(255)	NO	アーティスト名(投稿者名)
channel_id	VARCHAR(50)	YES	チャンネルID
channel_title	VARCHAR(255)	NO	チャンネル名
thumbnail_url	VARCHAR(255)	YES	チャンネル画像(任意)
created_at	TIMESTAMP	NO	登録日時
updated_at	TIMESTAMP	NO	更新日時(最新情報(チャンネル名、アイコン、登録者数など)を更新)

daily_recommendations テーブル 日替わりおすすめ履歴

カラム名	型	NULL	備考
id	BIGINT UNSIGNED	NO	主キー
video_id	BIGINT UNSIGNED	NO, Foreign Key→videos.id	おすすめ動画
recommended_date	DATE	NO, UNIQUE	日替わりおすすめ日
created_at	TIMESTAMP	NO	登録日時

- 1日1本を保証するため recommended_date に UNIQUE 制約を付与

user_histories テーブル 個人視聴履歴

カラム名	型	NULL	備考
id	BIGINT UNSIGNED	NO	主キー
user_id	BIGINT UNSIGNED	NO, Foreign Key→users.id	視聴したユーザー
video_id	BIGINT UNSIGNED	NO, Foreign Key→videos.id	視聴した動画
viewed_at	DATETIME	NO	視聴日時
watched_seconds	INT	YES	再生秒数(トラッキング用)
created_at	TIMESTAMP	NO	登録日時

- 同じ動画を何度も見た場合は、複数行登録され、再生傾向を分析できる

userFavorites テーブル お気に入り登録

カラム名	型	NULL	備考
id	BIGINT UNSIGNED	NO	主キー
user_id	BIGINT UNSIGNED	NO, Foreign Key→videos.id	登録したユーザー
video_id	BIGINT UNSIGNED	NO, Foreign Key→videos.id	お気に入り動画
created_at	TIMESTAMP	NO	登録日時

- 同一ユーザーが同じ動画を複数回登録できないように、**UNIQUE KEY (user_id, video_id)**(複合 UNIQUE制約)

channels テーブル チャンネル情報(任意)

カラム名	型	NULL	備考
id	BIGINT UNSIGNED	NO	主キー
channel_id	VARCHAR(50)	NO, UNIQUE	チャンネルID
channel_title	VARCHAR(255)	NO	チャンネル名
subscriber_count	BIGINT	YES	登録者数(Youtube Data APIから取得)
category	VARCHAR(100)	YES	カテゴリ(J-POPなど)
created_at	TIMESTAMP	NO	登録日時

- 将来的に「特定チャンネルから定期的に取得」機能を強化可能

API設計(エンドポイント仕様)

Next.js(フロントエンド) x Laravel API(バックエンド)構成に対応したRESTful API設計(エンドポイント仕様)を以下にまとめます。

全体設計方針

項目	内容
API仕様	RESTful API(JSONベース)
認証	Laravel Sanctum(APIトークン認証)
返却方式	JSON(ステータス+データ)
バージョン管理	/api/v1/... 形式で提供
対象	日本限定一般公開サービス

エンドポイント一覧(カテゴリ別)

認証・ユーザー関連

メソッド	パス	概要	認証
POST	/api/v1/register	ユーザー登録	不要
POST	/api/v1/login	ログイン	不要
POST	/api/v1/logout	ログアウト	必要
GET	/api/v1/user	ログイン中ユーザー情報	必要
POST	/api/v1/user	ユーザー情報変更(ゲスト以外)	必要

動画 (Videos)

メソッド	パス	概要	認証
GET	/api/v1/videos/{id}	特定動画の詳細取得	不要
GET	/api/v1/videos/trending	トレンド動画一覧(関連動画)(Youtube Data APIキー必須)	不要
GET	/api/v1/search?keyword=xxx	キーワード検索	不要
POST	/api/v1/videos	新規登録(Youtube Data APIから取得)	管理者のみ

アーティスト(投稿者) (Artisits)

- 今後の拡張用。「アーティスト別MV動画一覧」を取得したい時など。当面はartists.nameでアーティスト名(投稿者名)が取得されれば十分なので、このエンドポイントは使わない。

メソッド	パス	概要	認証
GET	/api/v1/artists	フロント用。登録済みアーティスト一覧を取得	不要
GET	/api/v1/artists/{id}	指定チャンネルの情報(アーティスト詳細)を取得	不要
GET	/api/v1/artists/{id}/videos	指定アーティストの動画一覧を取得	不要
POST	/api/v1/artists/sync	Youtube Data APIから取得・更新(バッチ用)。アーティスト情報自動同期	管理者/CRONのみ

- POST /api/v1/artists/syncについて
 - CRONジョブやArtisanコマンドなどから呼び出す(定期実行)
 - Youtube Data APIのchannels.listを使用して最新情報を取得
 - 既存のchannel_idがある場合は更新
 - 新しいチャンネルが検出されたらINSERT
 - バッチ運用イメージ
 - 毎日午前3時などにphp artisan schedule:runで起動
 - php artisan sync:artistsコマンド内で/api/v1/artists/syncを実行
 - 更新日時(updated_at)を見て必要なチャンネルのみ再取得
 - ログを残してレート制限を回避

日替わりおすすめ (Daily Recommendations)

メソッド	パス	概要	認証
GET	/api/v1/recommendations/today	今日のおすすめMVを取得	不要
GET	/api/v1/recommendations	日替わり履歴一覧(例: 最近7日分)	不要
GET	/api/v1/recommendations/{date}	指定日のおすすめ取得	不要

視聴履歴 (User Histories)

メソッド	パス	概要	認証
GET	/api/v1/histories	自分の視聴履歴一覧(最新20件など)	必要
POST	/api/v1/histories	視聴履歴を登録(動画を再生したとき)	必要
DELETE	/api/v1/histories/{id}	特定の履歴を削除	必要
DELETE	/api/v1/histories	全履歴を削除	必要

お気に入り (User Favorites)

メソッド	パス	概要	認証
GET	/api/v1/favorites	自分のお気に入り一覧	必要
POST	/api/v1/histories	お気に入り登録(video_id指定)	必要
DELETE	/api/v1/favorites/{video_id}	お気に入り解除	必要
GET	/api/v1/favorites/check/{video_id}	特定動画をお気に入り済みか確認	必要

チャンネル (Channels)

- 今後の拡張用

メソッド	パス	概要	認証
GET	/api/v1/channels	登録チャンネル一覧	不要
GET	/api/v1/channels/{id}	特定チャンネル情報	不要
GET	/api/v1/channels/{id}/videos	チャンネルの動画一覧	不要