

ランキングサイト仕様書

25G1029 大橋慶眞

2025年12月27日

GitHub リポジトリ URL

https://github.com/username/webpro_06

目次

1	概要	2
1.1	開発対象システム	2
2	第1部：利用者向け仕様	2
2.1	システム概要	2
2.2	利用方法	2
2.3	利用上の注意事項	3
3	第2部：管理者向け仕様	3
3.1	システム管理	3
3.2	運用監視	4
4	第3部：開発者向け仕様	4
4.1	技術仕様	4
4.2	データ構造	5
4.3	API 仕様	6
4.4	拡張仕様	7
4.5	セキュリティ考慮事項	7

1 概要

本仕様書は、一覧表示をベースとした3つのWebアプリケーションの開発仕様を定義するものである。各アプリケーションは独立したランキング情報管理システムとして動作し、Express.jsフレームワークを使用してWebサービスとして提供される。

1.1 開発対象システム

1. FANZA 動画ランキングサイト - デジタル動画コンテンツの売れ行きランキング管理
2. DLsite ランキングサイト - 同人作品の売れ行きランキング管理
3. メロンブックスランディングサイト - 同人誌の売れ行きランキング管理

2 第1部：利用者向け仕様

2.1 システム概要

3つの独立したランキングサイトを提供し、各分野の人気コンテンツの情報を閲覧・登録・編集・削除できるWebサービスである。

2.2 利用方法

2.2.1 アクセス方法

各サイトは異なるポートで動作しており、ブラウザで以下のURLにアクセスすることで利用できる。FANZA動画ランキングは`http://localhost:8081/fanza`、DLsiteランキングは`http://localhost:8082/dlsite`、メロンブックスランキングは`http://localhost:8083/melonbooks`でアクセス可能である。

2.2.2 基本機能

■ランキング一覧表示 各サイトのトップページでランキング一覧を表形式で確認できる。表示される情報は以下の通りである：

表1 各サイトの表示項目

サイト	表示項目
FANZA	ランク、動画名、女優名
DLsite	ランク、作品名、サークル名
メロンブックス	ランク、書籍名、著者

■詳細情報表示 一覧のタイトル・作品名・書籍名をクリックすると、詳細情報ページが表示される。詳細ページでは各サイトの特性に応じた情報を確認できる。FANZAサイトではランク、動画名、ジャンル、価格、発売日、女優名、動画リンクが表示され、DLsiteサイトではランク、作品名、サークル名、ジャンル、価格、

評価、作品リンクが表示され、メロンブックスサイトではランク、書籍名、著者、サークル、ジャンル、価格、書籍リンクが表示される。

■新規データ登録 「新規登録」リンクから新しいランキングデータを追加できる。必要な情報を入力フォームに記入し、「登録」ボタンを押すことで登録できる。

■データ編集・削除 各データの詳細ページや一覧ページから「編集」リンクをクリックすることで、既存データの編集が可能である。また、「削除」ボタンから不要なデータを削除できる。削除時には確認ダイアログが表示され、誤操作を防ぐ仕組みが実装されている。

2.3 利用上の注意事項

各サイトは独立して動作するため、データは共有されないことに注意が必要である。また、登録したデータはサーバー再起動時に初期化されるため、永続的なデータ保存は行われない。

■入力値検証 以下の項目について入力値検証が実装されている：

- ランク：1 以上の整数値のみ許可
- 価格：0 以上の整数値のみ許可
- 評価 (DLsite のみ)：1 から 5 の範囲の数値のみ許可
- 必須項目：タイトル・作品名・書籍名、リンク、主要項目が空欄でないこと
- URL 形式：リンク項目は有効な URL 形式であること

不正な値を入力した場合はエラーメッセージが表示され、登録処理は実行されない。

3 第2部：管理者向け仕様

3.1 システム管理

3.1.1 サーバー起動・停止

各サイトは独立した Node.js プロセスとして動作する。

■起動方法

```
1 # npm を使用した起動（推奨）scripts
2 npm run start-fanza
3 npm run start-dlsite
4 npm run start-melonbooks
5
6 # 直接起動
7 node fanza_ranking.js
8 node dlsite_ranking.js
9 node melonbooks_ranking.js
```

■停止方法 各プロセスで Ctrl+C を押下してサーバーを停止する。

3.1.2 ポート管理

各サイトは以下のポートで動作する：

表2 ポート割り当て

サイト	ポート番号
FANZA ランキング	8081
DLSite ランキング	8082
メロンブックスランキング	8083

3.1.3 データ管理

現在の実装では、データはメモリ内の配列として管理されている。サーバー再起動時にはサンプルデータに戻る。

■初期データ 各サイトには5件のサンプルデータが設定されている。

■データバックアップ 本格運用時には、データベースへの移行を推奨する。

3.2 運用監視

3.2.1 ログ監視

各サーバーの起動時にコンソールに起動メッセージが表示される。エラー発生時はコンソールログを確認する必要がある。

3.2.2 アクセス監視

Express.js の標準ログ機能により、アクセス状況を監視できる。

4 第3部：開発者向け仕様

4.1 技術仕様

4.1.1 開発環境

本システムの開発には Node.js (推奨バージョンは 14.x 以上)、Express.js 4.x、および EJS テンプレートエンジンが必要である。

■環境セットアップ 依存関係のインストールには以下のコマンドを実行する：

```
1 npm install
```

■package.json プロジェクトの設定と依存関係を定義するファイル。以下の内容が含まれる：

- Express.js 4.x - Web アプリケーションフレームワーク
- EJS 3.x - サーバーサイドテンプレートエンジン

- 起動スクリプト - 各サイトの個別起動コマンド定義

■ package-lock.json 依存関係の正確なバージョンを記録し、異なる環境での再現性を保証するファイル。
npm install の実行時に自動生成される。

4.1.2 ディレクトリ構成

```

1 kadai/
2   fanza_ranking.js      # サイトメインFANZA
3   dlsite_ranking.js     # サイトメインDLsite
4   melonbooks_ranking.js # メロンブックスサイトメイン
5   package.json          # 依存関係定義
6   package-lock.json     # 依存関係バージョン固定
7   views/                # テンプレートファイル
8     fanza_list.ejs |   |
9     fanza_detail.ejs |  |
10    fanza_new.ejs |   |
11    fanza_edit.ejs |  |
12    dlsite_list.ejs |   |
13    dlsite_detail.ejs |  |
14    dlsite_new.ejs |   |
15    dlsite_edit.ejs |  |
16    melonbooks_list.ejs |   |
17    melonbooks_detail.ejs |  |
18    melonbooks_new.ejs |  |
19    melonbooks_edit.ejs |
20  public/               # 静的ファイル（未使用）

```

4.2 データ構造

4.2.1 FANZA 動画ランキングデータ

```

1 {
2   id: Number,           // 一意識別子（現在の最大ID + で自動生成）1
3   rank: Number,         // ランキング順位
4   title: String,        // 動画タイトル
5   genre: String,        // ジャンル
6   price: Number,        // 価格（円、以上）0
7   releaseDate: String, // 発売日（YYYY-MM-）DD
8   actress: String,      // 女優名
9   link: String,         // 動画リンク（）URL
10 }

```

4.2.2 DLsite ランキングデータ

```

1 {
2   id: Number,           // 一意識別子（現在の最大ID + で自動生成）1

```

```

3   rank: Number,           // ランキング順位
4   productName: String,    // 作品名
5   genre: String,          // ジャンル
6   price: Number,          // 価格 (円、以上) 0
7   rating: Number,         // 評価 (の範囲) 1-5
8   circle: String,         // サークル名
9   link: String            // 作品リンク () URL
10 }

```

4.2.3 メロンブックスランキングデータ

```

1 {
2   id: Number,             // 一意識別子 (現在の最大ID + で自動生成) 1
3   rank: Number,           // ランキング順位
4   bookName: String,       // 書籍名
5   author: String,         // 著者
6   circle: String,         // サークル
7   genre: String,          // ジャンル
8   price: Number,          // 価格 (円、以上) 0
9   link: String            // 書籍リンク () URL
10 }

```

4.3 API 仕様

4.3.1 共通エンドポイント

各サイトで統一された URL 構造を使用する:{site} は FANZA の場合は「fanza」, DLsite の場合は「dlsite」, メロンブックスの場合は「melonbooks」となる。

表 3 API エンドポイント

メソッド	URL	機能
GET	/site	一覧表示
GET	/site/create	新規登録フォーム表示
POST	/site/create	新規データ登録処理
GET	/site/:id	詳細情報表示
GET	/site/edit/:id	編集フォーム表示
POST	/site/edit/:id	データ編集処理
POST	/site/delete/:id	データ削除処理

4.3.2 レスポンス形式

■HTML レスポンス 全てのエンドポイントは HTML ページを返す。EJS テンプレートエンジンを使用してサーバーサイドレンダリングを行う。

■リダイレクト POST 処理後は一覧ページ (/) にリダイレクトする。

4.4 拡張仕様

4.4.1 データベース連携

将来的な拡張として、データの永続化のためにデータベース連携を想定している。候補としては MongoDB (NoSQL データベース)、MySQL (リレーションナルデータベース)、SQLite (軽量データベース) がある。

4.4.2 追加機能

現在の基本機能（一覧表示、詳細表示、新規登録、編集、削除）に加えて、将来的には検索・フィルタ機能、ページネーション、ユーザー認証などの機能を追加することを想定している。

4.5 セキュリティ考慮事項

本システムのセキュリティ対策として、入力値検証の実装、SQL インジェクション対策、XSS 攻撃対策、CSRF 攻撃対策などを実装する必要がある。