

## PHP と Laravel を用いた効率的な図書管理システムの開発

Development of an efficient library management system using PHP and Laravel

20216109 高橋 祐貴 【関澤研究室】

### 1 はじめに

現代の図書管理システムはデジタル化されており、効率性と正確性が求められる状況にある。しかし、高信頼性システム研究室では、現在も書籍の貸出・返却管理を紙媒体で行っており、この運用にはいくつかの課題が存在する。具体的には、手作業による管理は手間がかかるだけでなく、人為的なミスリスクも高い。また、在庫の状況を即時に把握することが困難であり、管理効率の低下につながっている。

本研究では、書籍の貸出・返却および在庫管理を PHP とデータベースを用いて web 開発し、図書管理を行う。また、研究室サーバーでの実装を試みる。

### 2 環境

本研究は、開発および本番環境では Ubuntu を使用した。Ubuntu は、Linux ベースのオープンソースのオペレーティングシステムであり、サーバー環境として広く利用されているため、開発環境と本番環境を統一することで、移行時のトラブルを軽減できる。従来の紙管理を Web 化する目的から、信頼性やセキュリティの高い環境が必要だったため、Ubuntu を採用した。

フロントエンドには HTML と CSS を使用し、ユーザーが直感的にわかりやすいデザインを実現した。

バックエンドには PHP と Laravel を採用し、効率的な開発と保守性を確保した。データベースは MariaDB を使用し、安定したデータ管理を実現する。

#### 2.1 PHP Laravel

PHP (Hypertext Preprocessor) は、Web アプリケーション開発に広く用いられるオープンソースのサーバーサイドスクリプト言語である。主に Web サーバー上で動作し、動的な Web ページやアプリケーションの構築に使用される。Laravel は PHP で構築された Web アプリケーションフレームワークである。特に、ルーティング、データベース操作、認証機能など、Web アプリケーション開発を効率化するための機能が豊富に提供されており、扱いやすいフレームワークである。<sup>[1]</sup>

#### 2.2 MVC モデル

Laravel は MVC (Model-View-Controller) モデルを採用しており、本研究で開発した図書管理システムも MVC モデルに基づいて構築している。図 1 に MVC モデルを示す。MVC モデルは、システムの構成要素を「モデル」「ビュー」「コントローラー」の 3 つのコンポーネントに分け、それぞれが特定の役割を担うことでソフトウェアの構造を論理的かつ効率的に管理する手法である。

**モデル**はデータの処理全般を担当し、データベースとの連携によりデータの保存や取得を行う。処理結果はコントローラーへ送られ、後の操作に利用される。

**ビュー**はユーザーインターフェースの構築を担当し、HTML や CSS を用いて ユーザーにデータを視覚的に表示する役割を持つ。コントローラーから送られたデータをもとに画面を生成し、ユーザーが操作できる環境を提供

する。

**コントローラー**はシステム全体の動作を制御し、ユーザーからのリクエストを受け取る役割を担う。リクエスト内容に基づき、必要なデータをモデルから取得し、そのデータをビューに渡して表示を行う。コントローラーはモデルとビューを繋ぐ中間的な役割を果たす。<sup>[2]</sup>

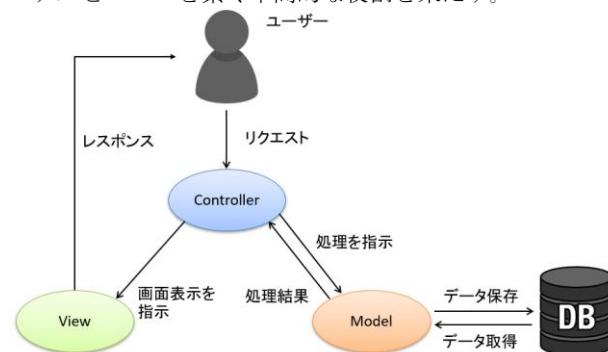


図 1 MVC モデル

### 3 設計

図 2 は本システムの機能階層図である。本システムは、一般ユーザー機能と管理者機能に分けて設計されている。

一般ユーザー機能では、図書の貸出・返却および検索が可能である。これにより、ユーザーが目的の本を簡単に見つけ、必要に応じて利用できるようにすることを目指した。貸出や返却の際には、バーコードを活用して操作を簡略化し、操作ミスを防ぐ設計としている。

管理者機能では、書籍とユーザーの登録、書籍の削除を行うことができる。これにより、システム管理者は図書のデータを効率的に管理することができる。

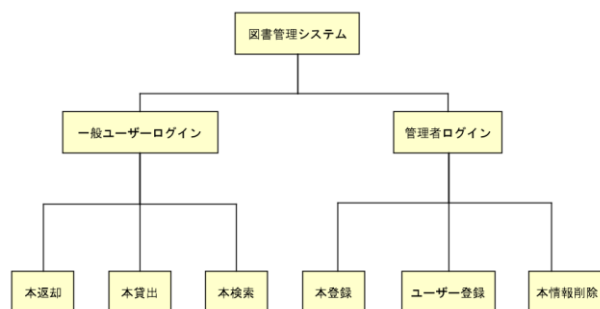


図 2 機能階層図

### 4 手法

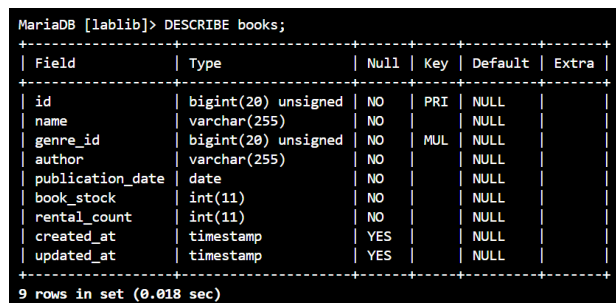
本研究システムでは、まずユーザーインターフェースの利便性を考慮して、画面のデザインテンプレートを作成し、その後 HTML と CSS を用いて画面設計を行った。次に、システムに必要なデータを格納するためのテーブルをマイグレーションで作成する。その後、PHP の Laravel フレームワークを用いてコントローラーを実装することで、画面とデータベースを連携させ、図書管理機能を実現する。

#### 4.1 データベースのテーブル作成

Laravel のマイグレーション機能を使用することで、SQL コマンドを直接記述することなく、データベースの設計や構築を効率的に行うことが可能である。この機能では、PHP 内でマイグレーションファイルを作成し、その中にテーブル構造を定義することにより、データベースの作成や更新を実現する。具体的には、php artisan make:migration create\_books\_table というコマンドを実行することで、マイグレーションファイルを生成することができる。

生成されたマイグレーションファイルには、テーブルのカラムやその属性を定義するためのコードを記述する。その後、PHP artisan migrate コマンドを入力することでデータベースにテーブルが作成される。マイグレーション機能を用いて図3の books テーブルを作成した。図3のテーブルには、題名(id)、著者名(author)、ジャンル(genre\_id)、貸出状況 (rental\_count) などのフィールドを定義しており、書籍登録や本検索で用いられる情報である。

マイグレーション機能の活用は、データベースの設計や更新を効率化するだけでなく、コードベースでのバージョン管理が可能となる。<sup>[3]</sup>



Field	Type	Null	Key	Default	Extra
id	bigint(20) unsigned	NO	PRI	NONE	
name	varchar(255)	NO		NONE	
genre_id	bigint(20) unsigned	NO	MUL	NONE	
author	varchar(255)	NO		NONE	
publication_date	date	NO		NONE	
book_stock	int(11)	NO		NONE	
rental_count	int(11)	NO		NONE	
created_at	timestamp	YES		NONE	
updated_at	timestamp	YES		NONE	

9 rows in set (0.018 sec)

図3 books テーブル

#### 4.2 PHP でのコントローラー作成

コントローラーは、バックエンドでの処理を担う主要な役割を持つ。本システムでは、book manage Controller に処理を書き込み貸出・返却を行っている。具体的には、貸出処理ではユーザー情報を認証し、書籍の在庫状況を確認した上で、貸出可能な場合に貸出情報と書籍情報を更新する。一方、返却処理では、該当する貸出情報を検索し、返却処理を行った後に書籍の貸出回数を減少させる仕組みである。これらの機能をコントローラーで統合的に管理することで、モデルやビューとの連携が容易となり、システム全体の拡張性や保守性が向上している。

### 5 開発・実装の進捗

本研究は、ローカル環境で開発を行い、十分な動作 確認を経た後、研究室サーバーに移行して本番環境として実装を目指す。開発した図書管理システムは、一般ユーザー向け機能および管理者機能がローカル環境で実装され、主要な機能が完成した。これらの機能は、ユーザーインターフェースの使いやすさを考慮し設計され、直感的な操作が可能であることが確認できた。本章は一般ユーザーが使う機能を紹介する。

#### 5.1 貸出・返却

本の貸出は、本のバーコードと貸出日を入力することで行う事ができる。入力されたバーコードに対応する本がデータベース上に存在し、かつ貸出可能な在庫が残っ

ている場合にのみ貸出できる。貸出が正常に完了した際には、本の貸出完了画面が表示される。一方で、入力されたバーコードに対応する本がデータベース上に存在しない場合や貸出可能な在庫が無い場合は貸出処理が実行されず、貸出失敗画面が表示される。

本の返却は、本のバーコードを入力することで行う事ができる。入力したバーコードの本をログインしているユーザーが貸出している場合に返却できる。返却が正常に完了した際には、本の返却完了画面が表示される。入力されたバーコードの本をユーザーが貸出していない場合は返却処理が実行されず、返却失敗画面が表示される。

#### 5.2 本検索

本の検索は、図3に示す books テーブルを基に、書籍情報の検索機能を提供している。本検索機能では、データベースから題名(id)、著者名(author)、ジャンル(genre\_id)、貸出状況 (rental\_count) の情報を取得し、これらを検索結果として一覧表示する。

検索機能には、以下の2つの主要な操作が可能である。

1つ目は、題名や著者名をキーワードとして入力し、該当する書籍を絞り込むことである。この機能により、利用者は必要な書籍を迅速かつ効率的に検索することが可能となる。

2つ目は、ジャンルの選択である。ジャンルは「書籍」と「卒業論文」の2種類に分類されており、プルダウンメニューを使用して選択できる仕様となっている。これにより、ジャンルに応じた絞り込みが簡易に行える。以上の機能により、利用者は目的の書籍情報を効率よく検索できるよう設計されており、利便性の高い機能になっている。

### 6 評価

開発した図書管理システムの操作性を検証するため、研究室のメンバーにシステムを実際に利用してもらい、正常に動作するかを確認したい。また、操作性や改善点のアンケートを取り、改善していくのが将来の課題である。

### 7 まとめ

現時点では、ローカル環境において基本的な機能の実装と動作確認を完了しているが、研究室サーバーへの導入と本格的な運用開始には至っていない。今後は、サーバー環境への移行を進めるとともに、研究室のニーズに応じた機能拡張や運用テストを実施することで、実用的なシステムへと仕上げる予定である。

#### 参考文献

- [1] 発注ラウンジ編集部. PHP とは？プログラミング初心者向けに PHP の基礎知識を解説. 発注ラウンジ. <https://hnavi.co.jp/knowledge/blog/php/>, (参照 2024-12-08)
- [2] メディアファイブ株式会社. 【解説】MVC モデルとは？メリット・デメリット. <https://system-kaihatu.com/archives/3204>, (参照 2024-12-05)
- [3] MiyakeNatsuho. Laravel のマイグレーションを理解しよう. Qiita. <https://qiita.com/MiyakeNatsuho/items/32a70adfbba646200a3>, (参照 2024-12-05)