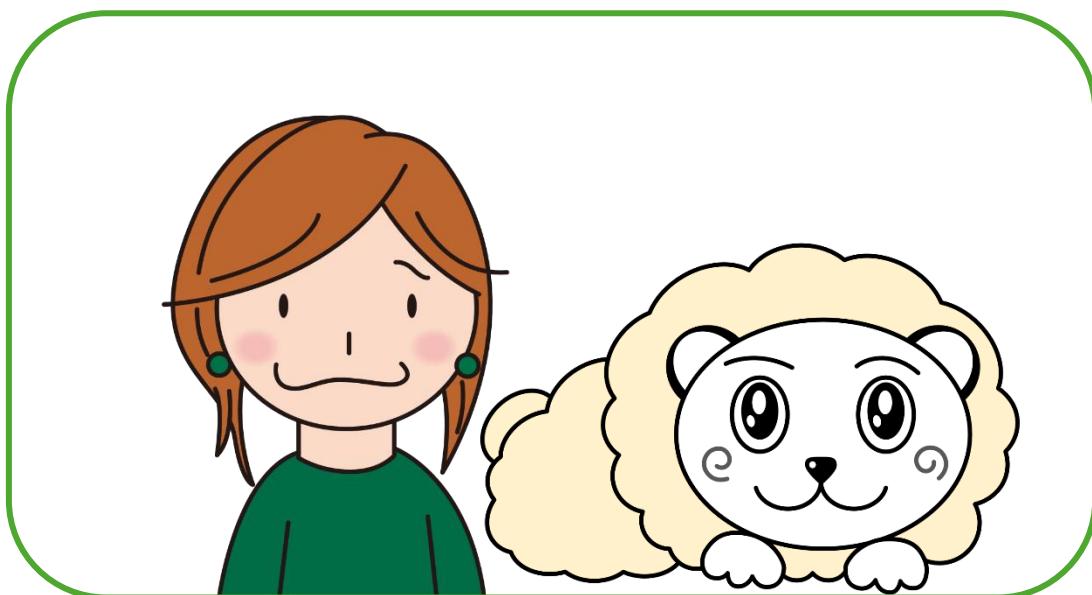


# 書籍【Laravel の教科書】

## Laravel11 対応サポートガイド



Version 1.1

作成日：2024年4月10日

## 内容

はじめに .....	3
タイムゾーンと言語設定の登録場所の変更.....	5
Laravel のディレクトリ（フォルダ）構造の変更.....	7
<b>Laravel11 の app ディレクトリ .....</b>	<b>8</b>
<b>Laravel11 の bootstrap ディレクトリ .....</b>	<b>9</b>
Middleware（ミドルウェア）はデフォルトで非表示に.....	10
Middleware（ミドルウェア）の登録場所の変更 .....	11
Gate（ゲート）の登録場所の変更 .....	13
ホームの登録場所の変更.....	15
その他の変更事項.....	16



# はじめに

書籍では Laravel10 の使い方をご紹介していますが、Laravel は 2024 年 3 月 12 日にバージョンアップし Laravel11 がリリースされました。

基本的な使い方に変わりはありませんが、**Laravel11 ではファイル数が大幅に減り、各種登録場所が変わっています。**本ファイルは、Laravel11 を使って書籍にあるコードを実行した際に、変更する必要がある部分について解説します。

～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～



「Laravel11 では、ファイル数がかなり減ったんだね。

なんで？」



「ファイル数を減らすことで、分かりやすい構造にしたかったみたいだよ。」



「なるほど。それは良いことだね。ただ、一気にファイルが減ると、色々影響がありそう。」



「うん。正直あるよ。ここから先、Laravel11 になったことで、どんな変更があるか解説していくね。」

Laravel11 の変更点について詳しく知りたい場合は、下記ブログ記事も参考にしてください。

<https://biz.addisteria.com/laravel11/>

## タイムゾーンと言語設定の登録場所の変更

- 該当する章：2-8 設定とメッセージの日本語化
- 該当ページ：p 96～97

Laravel10までは、タイムゾーンや言語の設定は、config/app.php ファイルに行っています。  
した。Laravel11では、.env ファイルにて、タイムゾーンや言語の設定を行います。

.env ファイルを開き、下記のように変更してください。

### **[.env] Laravel11**

※赤文字部分が変更箇所です

APP\_TIMEZONE=Asia/Tokyo

APP\_URL=http://localhost

APP\_LOCALE=ja

APP\_FALLBACK\_LOCALE=en

APP\_FAKELOCALE=ja\_JP

Laravel11では、config/app.php ファイルにおいて、次のようにタイムゾーンが設定されています。

### **[config/app.php] Laravel11**

```
'timezone' => env('APP_TIMEZONE', 'UTC'),
```

これは、「.env の'APP\_TIMEZONE'に設定があれば、そちらを採用する、なければ、'en'とする」といった意味です。この設定により、.env ファイルにある設定が優先されます。言語設定も同じ方法で設定されています。

なお書籍でご紹介したとおり、**config/app.php** ファイルに下記のようにコードをいれて**も動作します**。この場合は.env ファイルのタイムゾーン設定は使用されず、config/app.php のタイムゾーン設定が有効となります。

### **[config/app.php] Laravel10**

```
'timezone' => 'Asia/Tokyo',
```

～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～★～☆彌～



「つまり以前の方法でも動作するんだね。」



「うん。 Laravel は柔軟にできているんだ。」



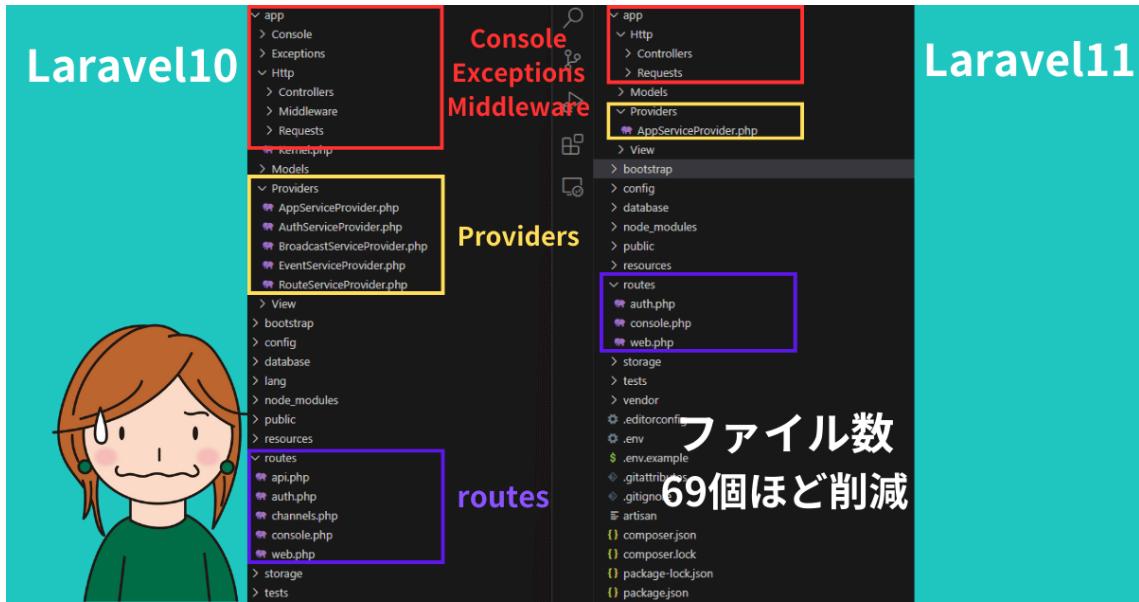
「よかった。でも、以前の方法では無理なことも、たくさんあるんでしょ？」



「いや、そんなに多くないから安心して！ 分かりやすく説明するために、次に構造の変更を見ていくね。」

## Laravel のディレクトリ（フォルダ）構造の変更

- 該当する章：3-2 Laravelのディレクトリ（フォルダ）構造
- 該当ページ：p 123～125



Laravel11 では、Laravel10 と比べて大幅にファイル数が減りました。減ったファイル数

は、およそ 69 個ほど。そのため、書籍の 123 ページからご説明しているディレクトリ構  
造にも変更点があります。大きく変わったのは、**app ディレクトリと bootstrap ディレク  
トリ**です。書籍の 124 ページから 125 ページに説明があります。この部分について、変更  
点をお伝えします。

## Laravel11 の app ディレクトリ

Laravel11では、appには、次のディレクトリが入っています。

**Http**

**Models**

**Providers**

**View**

なおLaravel10以前では、MiddlewareもHttpの直下に入っていましたが、Laravel11では、デフォルトではMiddlewareは表示されなくなりました。

appディレクトリ内のディレクトリをひとつずつ説明します。

### ▶ **Http**

Httpは非常によく使う部分です。デフォルトでは、次のディレクトリが入っています。

**Controllers**

**Requests**

Controllersには、コントローラファイルが入ります。

### ▶ **Models**

Modelsには、モデルファイルが入ります。

## ▶ Providers

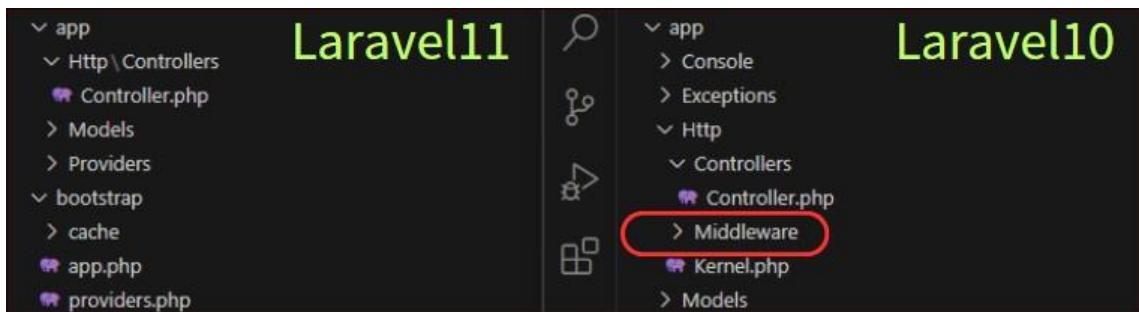
Providersには、 Laravel起動時の処理を設定します。最初から使用する必要はありませんが、 Webアプリを開発する中で、 編集する機会は出てくるでしょう。なお Laravel10では Providersの中にはデフォルトで5個のファイルが入っていましたが、 Laravel11では、 AppServiceProvider.phpファイルしか入っていません。

## Laravel11 の bootstrap ディレクトリ

bootstrapには、 フレームワークの起動時の処理を行うapp.phpファイルが入っています。以前はこのファイルを操作する機会は、 ほとんどありませんでした。 ですが Laravel11からは、 このapp.phpファイルにミドルウェアなどを登録します。

## Middleware（ミドルウェア）はデフォルトで非表示に

- 該当する章：8-1 ミドルウェアって何？
- 該当ページ：p 239～240



これまでapp/Http/Middlewareには、デフォルトで9個のミドルウェアファイルがあります。Laravel11では、フォルダごと、ごっそりMiddleware(ミドルウェア) が消えています。さらに、Middlewareを登録するためのapp/Http/Kernel.phpも消えていました。

ただミドルウェアファイルは、単に表示されていないだけで vendor ディレクトリの中に入っています。そのため、これまでどおり使えます。vendor ディレクトリについては、書籍の 259-260 ページを参照してください。

また、Laravel11でのMiddlewareの登録場所は次ページでご紹介します。

## Middleware（ミドルウェア）の登録場所の変更

- 該当する章：8-2 ミドルウェアで管理者のみがアクセス可能にする
- 該当ページ：p 246

Laravel10 では作成した Middleware は app/Http/Kernel.php に登録しますが、

Laravel11 では bootstrap/app.php に登録します。 Laravel11 ご利用の際は、書籍内で

作成する RoleMiddleware は、下記のように bootstrap/app.php 内に登録してください

い。

### **[bootstrap/app.php]**

```
<?php

use Illuminate\Foundation\Application;
use Illuminate\Foundation\Configuration\Exceptions;
use Illuminate\Foundation\Configuration\Middleware;
// 追加
use App\Http\Middleware\RoleMiddleware;

return Application::configure(basePath: dirname(__DIR__))
    ->withRouting(
        web: __DIR__.'/../routes/web.php',
        commands: __DIR__.'/../routes/console.php',
        health: '/up',
    )
    ->withMiddleware(function (Middleware $middleware) {
        // 追加
        $middleware->alias([
            'admin' => RoleMiddleware::class
        ])
    })

```

```
]);
}

->withExceptions(function (Exceptions $exceptions) {
    //
})->create();inatemyServiceProvider;
```

## Gate (ゲート) の登録場所の変更

- 該当する章 : 8-3 Gate (ゲート) を使った動作や表示の制限
- 該当ページ : p 251~252

Laravel10では、Gate (ゲート) は、app/Providers/AuthServiceProvider.phpに登録しました。ですがLaravel11では、AuthServiceProvider.php はなくなっています。

**Laravel11では、Gateはapp/Providers/AppServiceProvider.phpに登録します。**

書籍ではAuthServiceProvider.phpにGateを登録するよう記述していますが、 Laravel11ご利用の際は、下記のように、AppServiceProvider.phpにGateを記述してください。なお記述するコード自体に変更はありません。

### 【AppServiceProvider.php】

```
<?php

namespace App\Providers;

use Illuminate\Support\ServiceProvider;
// 追加
use Illuminate\Support\Facades\Gate;
use App\Models\User;

class AppServiceProvider extends ServiceProvider
{
    /**
     * Register any application services.
     */
    public function register(): void
```

```
{  
    //  
}  
  
/**  
 * Bootstrap any application services.  
 */  
public function boot(): void  
{  
    // 追加  
    Gate::define('test', function (User $user) {  
        if($user->id === 1) {  
            return true;  
        }  
        return false;  
    });  
}  
}
```

## ホームの登録場所の変更

- 該当する章：10-4 メニューとロゴをカスタマイズ
- 該当ページ：p 313

デフォルトでは、ログイン直後はダッシュボード（/dashboard）が表示されます。書籍では、この部分を変更し、投稿の一覧ページ（/post）が表示されるようにしました。

ただ、この部分でも変更が必要です。 Laravel10 では、ログイン直後に表示されるページを変更するために、app/Providers/RouteServiceProvider.php のコードを変更しました。ですが、Laravel11 では、このファイルがなくなっています。そのため、同じ動作をするには下記のように、routes/web.php ファイル内のダッシュボードのルート設定を変える必要があります。

### 【web.php】

```
// 無効にする
// Route::get('/dashboard', function () {
//     return view('dashboard');
// })->middleware(['auth', 'verified'])->name('dashboard');

// 追加
Route::get('/dashboard', [PostController::class, 'index'])->middleware(['auth'])
    ->name('dashboard');
```

※この変更により、URL は/dashboard のまま、画面には投稿一覧が表示されるようになります。

## その他の変更事項

ほか、重要度は低いものの、書籍に記した説明と、Laravel11版との相違を下記の表にしました。

ページ	該当箇所	修正前	修正後
92	マイグレートでデータベースにテーブルを作成する		現在は Breeze インストール後にマイグレートが実行済みなので、本操作は不要です。ただ、実施しても特に問題はありません。
166	Laravel と連携できるデータベース	MariaDB10.3 以上（バージョンポリシー） MySQL5.7 以上（バージョンポリシー） PostgreSQL10.0 以上（バージョンポリシー） SQLite3.8.8 以上 SQL Server2017 以上（バージョンポリシー）	MariaDB10.3 以上（バージョンポリシー） MySQL5.7 以上（バージョンポリシー） PostgreSQL10.0 以上（バージョンポリシー） <b>SQLite3.35.0 以上</b> SQL Server2017 以上（バージョンポリシー）