

PRECIOUS GWYNTH A. MALLO	WEBDEV
BSIT 3C	LAB 5

Part 1: Create and Register Controllers (Homepage Controller & Dashboard Controller)

The Homepage and Dashboard Controllers manage the homepage and dashboard logic of our app or website. You need to register these controllers in the web.php or route file so they can be used in routes. For instance, the HomeController's index method takes users to the "homepage" view, while the DashboardController directs users to the "dashboard" for site navigation and specific functions.

```
<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\HomeController;
use App\Http\Controllers\DashboardController;
use App\Http\Controllers\LoginController;

Route::get(uri: '/', action: [HomeController::class, 'index'])->name(name: 'home');

Route::middleware(middleware: ['auth'])->group(callback: function (): void {
    Route::get(uri: '/dashboard/{userId}', action: [DashboardController::class, 'index'])->name(name: 'dashboard');
});
```

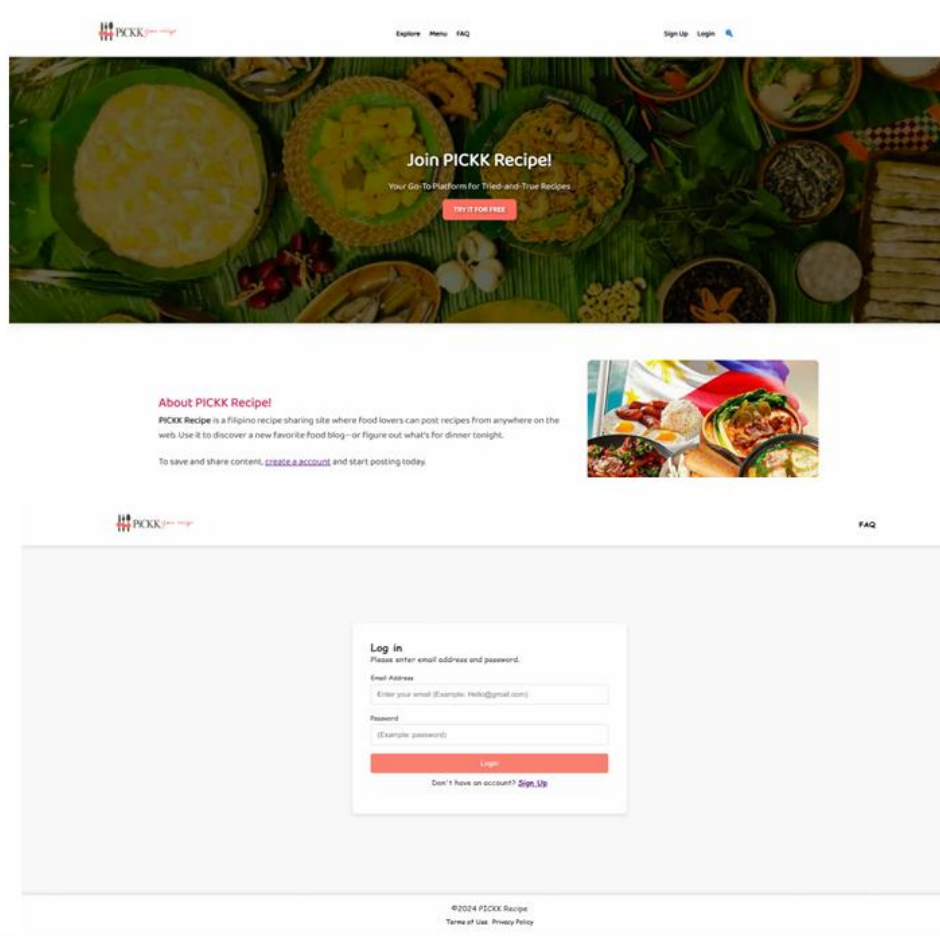
```
app > Http > Controllers > DashboardController.php > ...
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class DashboardController extends Controller
8  {
9      public function index($userId): Factory|View
10     {
11         //Fetch user-specific data based on the user ID
12         $user = User::find($userId);
13
14         return view(view: 'dashboard', data: ['user' => $user]);
15     }
16 }
```

Part 2: Assign Controllers to Routes

Use "auth" middleware to make sure only logged-in users can access certain routes. This means users must be authenticated to view specific pages like "/", "/dashboard". Test these routes to ensure they load properly.

```
routes > web.php
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\HomeController;
5  use App\Http\Controllers\DashboardController;
6  use App\Http\Controllers\LoginController;
7
8
9  Route::get(uri: '/', action: [HomeController::class, 'index'])->name(name: 'home');
10
11 Route::middleware(middleware: ['auth'])->group(callback: function (): void {
12     Route::get(uri: '/dashboard/{userId}', action: [DashboardController::class, 'index'])->name(name: 'dashboard');
```

Test routes (e.g. /, /dashboard) to ensure proper page loading.



Part 3: Controllers with Parameters

The DashboardController uses a dynamic parameter, \$userId, to get user data and show it on the "Dashboard" view, displaying information specific to each user.

```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  2 references | 0 implementations
8  class DashboardController extends Controller
9  {
10     1 reference | 0 overrides
11     public function index($userId): Factory|View
12     {
13         // Example: Fetch user-specific data based on the user ID
14         $user = User::find($userId); // Make sure to import the User model
15
16         return view('dashboard', data: ['user' => $user]);
17     }
18 }
```