

## 1. Routes (web.php)

```
> Users > User > Desktop > webdev proj > Controller_WebDV > routes > web.php
1  <?php
2
3  use App\Http\Controllers\AuthController;
4  use App\Http\Controllers\TicketController;
5  use Illuminate\Support\Facades\Route;
6
7  // Make the login page the landing page
8  Route::get('/', function () {
9      return redirect()->route('login');
10 });
11
12 // Route to show the create ticket form - restricted to users with 'user' role
13 Route::get('/create-ticket', [TicketController::class, 'create'])
14     ->name('create-ticket')
15     ->middleware('role:user');
16
17 // Route to handle form submission (store the ticket) - restricted to users with 'user' role
18 Route::post('/view-ticet', [TicketController::class, 'store'])
19     ->name('view-ticket')
20     ->middleware('role:user');
21
22 // Login routes
23 Route::get('/login', [AuthController::class, 'showLoginForm'])->name('login');
24
25 // Separate POST routes for user and admin logins
26 Route::post('/login', [AuthController::class, 'loginUser'])->name('login.submit');
27 Route::post('/admin/login', [AuthController::class, 'loginAdmin'])->name('admin.login');
28
29 // Logout route
30 Route::post('/logout', [AuthController::class, 'logout'])->name('logout');
31
32 Route::get('/view-ticket', [TicketController::class, 'view'])
33     ->name('view-ticket')
34     ->middleware('role:admin,user');
35
36 Route::post('/store-ticket', [TicketController::class, 'store'])->name('store-ticket');
37 Route::get('/search', [TicketController::class, 'search'])->name('search-ticket');
```

## 2. TicketController.php

```
{
    // Other methods...

    // Method for admins to view tickets

    public function create()
    {
        return view('create-ticket'); // Make sure this view exists
    }

    public function view()
    {
        // Dummy data to simulate tickets
        $tickets = [
            [
                'id' => 1,
                'title' => 'Sample Ticket 1',
                'status' => 'open',
                'description' => 'Description for sample ticket 1',
            ],
            [
                'id' => 2,
                'title' => 'Sample Ticket 2',
                'status' => 'closed',
                'description' => 'Description for sample ticket 2',
            ],
            [
                'id' => 3,
                'title' => 'Sample Ticket 3',
                'status' => 'in-progress',
                'description' => 'Description for sample ticket 3',
            ],
            [
                'id' => 4,
                'title' => 'Sample Ticket 3',
                'status' => 'Open',
                'description' => 'Description for sample ticket 4 ',
            ],
            [
                'id' => 5,
                'title' => 'Sample Ticket 3',
                'status' => 'in-progress',
                'description' => 'Description for sample ticket 5',
            ],
            [
                'id' => 6,
                'title' => 'Sample Ticket 3',
                'status' => 'Closed',
                'description' => 'Description for sample ticket 6',
            ],
            [
                'id' => 7,
                'title' => 'Sample Ticket 3',
                'status' => 'Open',
                'description' => 'Description for sample ticket 7',
            ],
            [
                'id' => 8,
                'title' => 'Sample Ticket 3',
                'status' => 'Open',
                'description' => 'Description for sample ticket 8',
            ],
        ];
    }
}
```

Sample data being populated at the dashboard

```

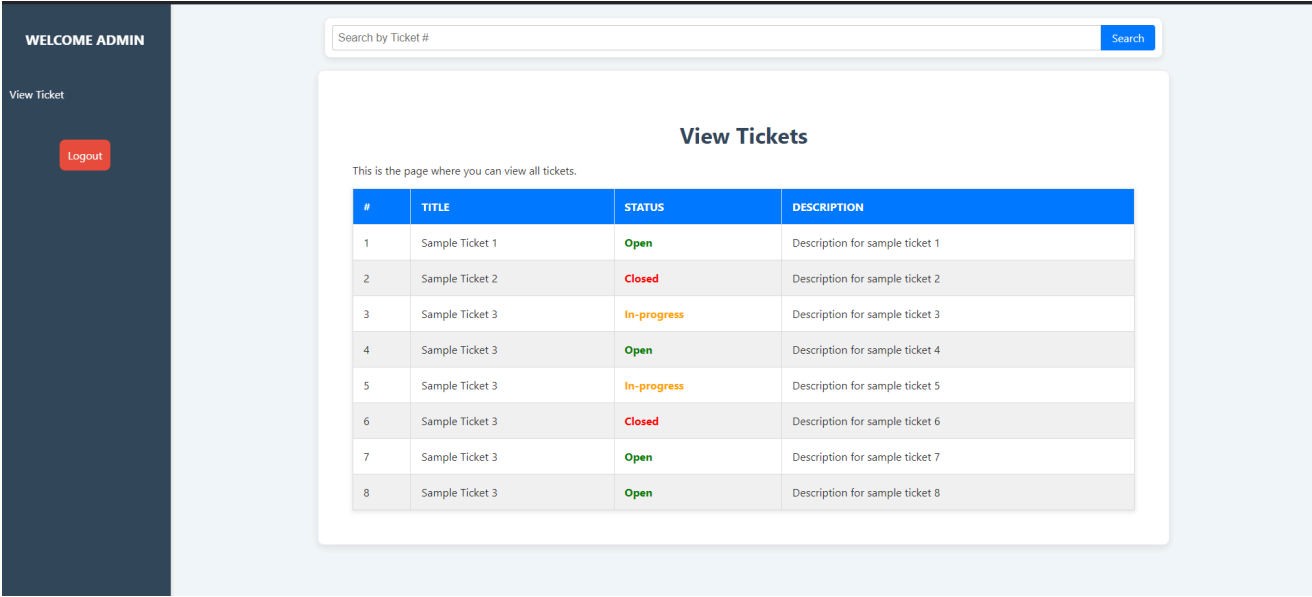
public function search(Request $request)
{
    $searchTerm = $request->input('search');
    // Dummy data to simulate tickets
    $tickets = [
        [
            'id' => 1,
            'title' => 'Sample Ticket 1',
            'status' => 'open',
            'description' => 'Description for sample ticket 1',
        ],
        [
            'id' => 2,
            'title' => 'Sample Ticket 2',
            'status' => 'closed',
            'description' => 'Description for sample ticket 2',
        ],
        [
            'id' => 3,
            'title' => 'Sample Ticket 3',
            'status' => 'in-progress',
            'description' => 'Description for sample ticket 3',
        ],
        [
            'id' => 4,
            'title' => 'Sample Ticket 3',
            'status' => 'Open',
            'description' => 'Description for sample ticket 4 ',
        ],
        [
            'id' => 5,
            'title' => 'Sample Ticket 3',
            'status' => 'in-progress',
            'description' => 'Description for sample ticket 5',
        ],
        [
            'id' => 6,
            'title' => 'Sample Ticket 3',
            'status' => 'Closed',
            'description' => 'Description for sample ticket 6',
        ],
        [
            'id' => 7,
            'title' => 'Sample Ticket 3',
            'status' => 'Open',
            'description' => 'Description for sample ticket 7',
        ],
        [
            'id' => 8,
            'title' => 'Sample Ticket 3',
            'status' => 'Open',
            'description' => 'Description for sample ticket 8',
        ],
    ];
    // Filter tickets based on the search term
    $filteredTickets = array_filter($tickets, function ($ticket) use ($searchTerm) {
        return str_contains((string)$ticket['id'], $searchTerm);
    });

    return view('view-ticket', ['tickets' => $filteredTickets]);
}
}

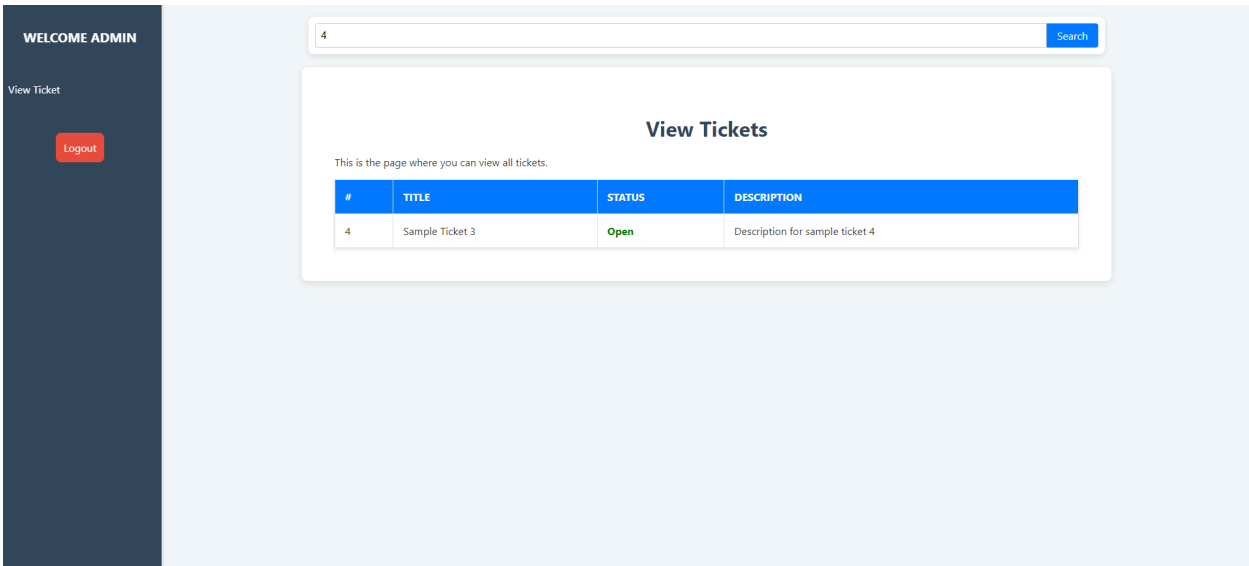
```

A search function to check the status of the tickets

### 3. Demo of the website



Dashboard with the sample tickets



Ticket search via number (#)