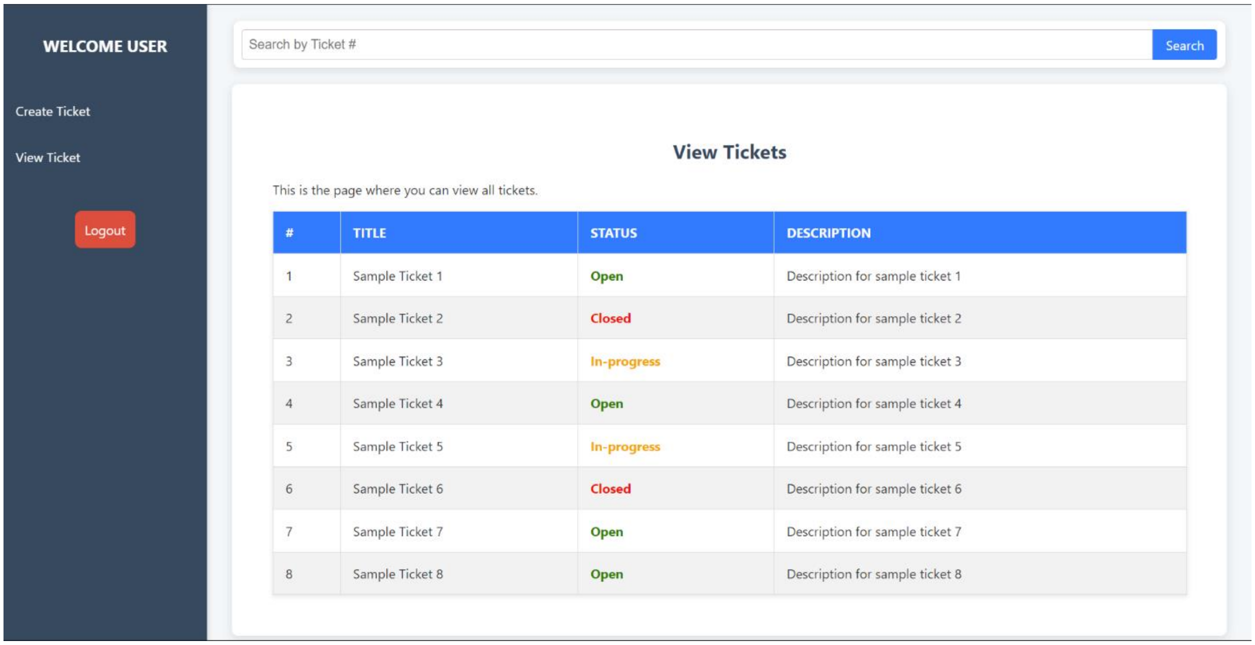
Documentations:

1. **Ticket View that loads the contents using the controller**



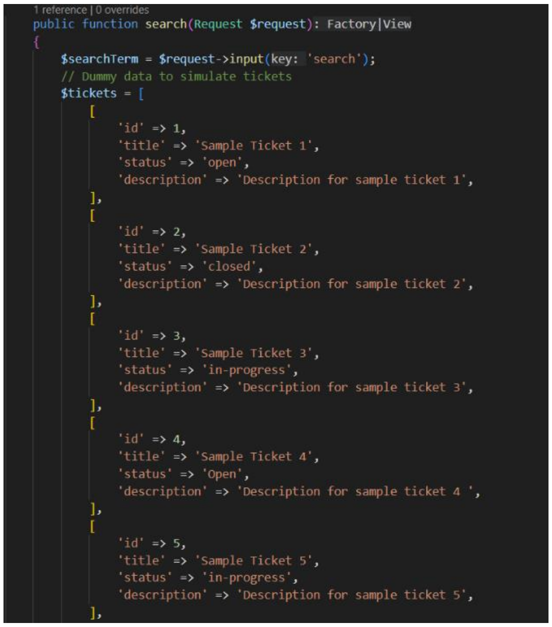
View Tickets

This is the page where you can view all tickets.

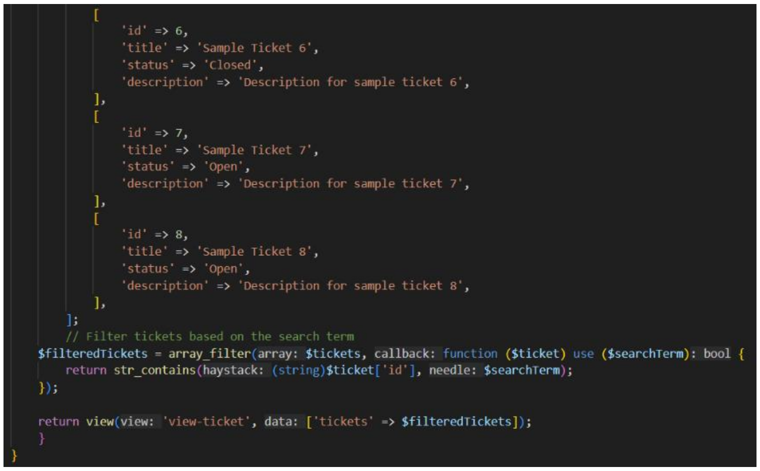
#	TITLE	STATUS	DESCRIPTION
1	Sample Ticket 1	Open	Description for sample ticket 1
2	Sample Ticket 2	Closed	Description for sample ticket 2
3	Sample Ticket 3	In-progress	Description for sample ticket 3
4	Sample Ticket 4	Open	Description for sample ticket 4
5	Sample Ticket 5	In-progress	Description for sample ticket 5
6	Sample Ticket 6	Closed	Description for sample ticket 6
7	Sample Ticket 7	Open	Description for sample ticket 7
8	Sample Ticket 8	Open	Description for sample ticket 8

- This is a view-ticket view that reads the id, title, status, and description inside the controller array so that it can load the information into the table.

2. **Controller code that loads information using an array**



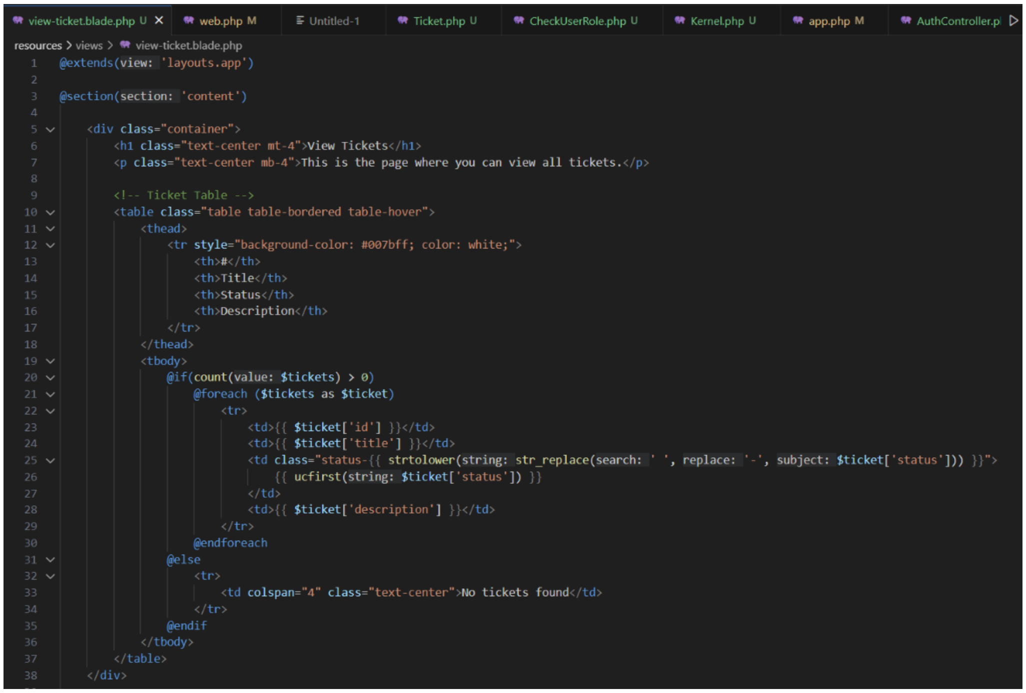
```
    1 reference | 0 overrides
    public function search(Request $request): Factory|View
    {
        $searchTerm = $request->input(key: 'search');
        // Dummy data to simulate tickets
        $tickets = [
            [
                'id' => 1,
                'title' => 'Sample Ticket 1',
                '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 4',
                'status' => 'Open',
                'description' => 'Description for sample ticket 4 ',
            ],
            [
                'id' => 5,
                'title' => 'Sample Ticket 5',
                'status' => 'in-progress',
                'description' => 'Description for sample ticket 5',
            ],
```



```
            [
                'id' => 6,
                'title' => 'Sample Ticket 6',
                'status' => 'Closed',
                'description' => 'Description for sample ticket 6',
            ],
            [
                'id' => 7,
                'title' => 'Sample Ticket 7',
                'status' => 'Open',
                'description' => 'Description for sample ticket 7',
            ],
            [
                'id' => 8,
                'title' => 'Sample Ticket 8',
                'status' => 'Open',
                'description' => 'Description for sample ticket 8',
            ],
        ];
        // Filter tickets based on the search term
    $filteredTickets = array_filter(array: $tickets, callback: function ($ticket) use ($searchTerm): bool {
        return str_contains(haystack: (string)$ticket['id'], needle: $searchTerm);
    });

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

- This code in TicketController has an array of information from id 1 to 8 with its own title, description and status. At the bottom part it has a code that filters it based on the ID searched in the search bar of the view-ticket view.

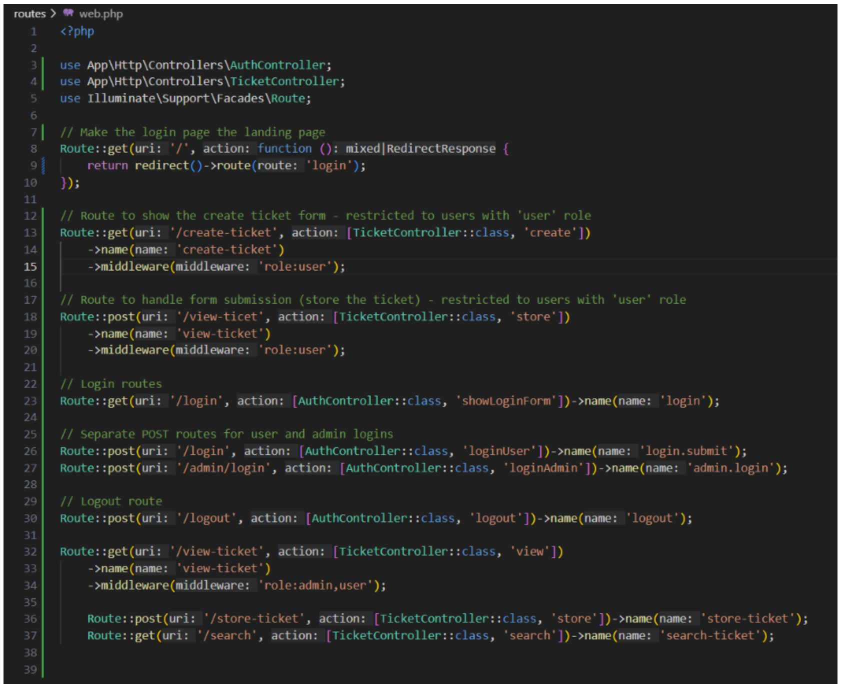
3. Code that reads and loads the information from the TicketController



```
@extends(view: 'layouts.app')

@section(section: 'content')

    <div class="container">
        <h1 class="text-center mt-4">View Tickets</h1>
        <p class="text-center mb-4">This is the page where you can view all tickets.</p>

        <!-- Ticket Table -->
        <table class="table table-bordered table-hover">
            <thead>
                <tr style="background-color: #007bff; color: white;">
                    <th>#</th>
                    <th>Title</th>
                    <th>Status</th>
                    <th>Description</th>
                </tr>
            </thead>
            <tbody>
                @if(count(value: $tickets) > 0)
                    @foreach ($tickets as $ticket)
                        <tr>
                            <td>{{ $ticket['id'] }}</td>
                            <td>{{ $ticket['title'] }}</td>
                            <td class="status-{{ strtolower(string: str_replace(search: ' ', replace: '-', subject: $ticket['status'])) }}">
                                {{ ucfirst(string: $ticket['status']) }}
                            </td>
                            <td>{{ $ticket['description'] }}</td>
                        </tr>
                    @endforeach
                @else
                    <tr>
                        <td colspan="4" class="text-center">No tickets found</td>
                    </tr>
                @endif
            </tbody>
        </table>
    </div>
```

- This code reads the id, title, status, and description from the TicketController using the $ticket and loads it to the table so we can see it on the website.

4. Updates on routes for the TicketController



```
<?php

use App\Http\Controllers\AuthController;
use App\Http\Controllers\TicketController;
use Illuminate\Support\Facades\Route;

// Make the login page the landing page
Route::get(uri: '/', action: function (): mixed|RedirectResponse {
    return redirect()->route(route: 'login');
});

// Route to show the create ticket form - restricted to users with 'user' role
Route::get(uri: '/create-ticket', action: [TicketController::class, 'create'])
    ->name(name: 'create-ticket')
    ->middleware(middleware: 'role:user');

// Route to handle form submission (store the ticket) - restricted to users with 'user' role
Route::post(uri: '/view-ticet', action: [TicketController::class, 'store'])
    ->name(name: 'view-ticket')
    ->middleware(middleware: 'role:user');

// Login routes
Route::get(uri: '/login', action: [AuthController::class, 'showLoginForm'])->name(name: 'login');

// Separate POST routes for user and admin logins
Route::post(uri: '/login', action: [AuthController::class, 'loginUser'])->name(name: 'login.submit');
Route::post(uri: '/admin/login', action: [AuthController::class, 'loginAdmin'])->name(name: 'admin.login');

// Logout route
Route::post(uri: '/logout', action: [AuthController::class, 'logout'])->name(name: 'logout');

Route::get(uri: '/view-ticket', action: [TicketController::class, 'view'])
    ->name(name: 'view-ticket')
    ->middleware(middleware: 'role:admin,user');

    Route::post(uri: '/store-ticket', action: [TicketController::class, 'store'])->name(name: 'store-ticket');
    Route::get(uri: '/search', action: [TicketController::class, 'search'])->name(name: 'search-ticket');


```

- Updated the web.php routes for the changes on TicketController so that it is implemented on the view-ticket view and is reflected properly on the desired pages.