Ajax Searching

Ajax Multiple Searching Using Model in LARAVEL

MIGRATION Page –

Command :

php artisan make:migration create_searches_table

php artisan migrate

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('searches', function (Blueprint $table) {
            $table->id();
            $table->string('name');
            $table->string('email');
            $table->string('phone');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('searches');
    }
};
```

Command:

php artisan make:controller SearchController

```php
<?php

namespace App\Http\Controllers;

use App\Models\Search;
use Illuminate\Http\Request;

class SearchController extends Controller
{
    public function search(Request $request)
    {
        $query = Search::query();
        if ($request->ajax()) {
            $users = $query->where('name','LIKE','%'.$request->search.'%')
            ->orWhere('email','LIKE','%'.$request->search.'%')
            ->orWhere('phone','LIKE','%'.$request->search.'%')
            ->get();
            return response()->json(['users' => $users]);
        }
        else
        {
            $users = $query->get();
            return view('search', compact('users'));
        }
    }
}
```

MODEL Page –

Make model file –

php artisan make:model Search

ROUTE Page –

```php
<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\SearchController;

/*
|----------------------------------------------------------------------
| Web Routes
|----------------------------------------------------------------------
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider and all of them will
| be assigned to the "web" middleware group. Make something great!
|
*/

// Route::get('/', function () {
//     return view('welcome');
// });

Route::get('user-search', [SearchController::class, 'search'])->name('userSearch');
```

View page –

search.blade.php

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Search</title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
</head>
<body align="center" >
    <h1>Searching</h1>
    <input  type="search" name="search" id="search" placeholder="Search Here......" ><br><br><br>

    <table border="1" align="center" width="50%">
```

```html
        <tr>
            <th>S.No</th>
            <th>Name</th>
            <th>Email</th>
            <th>Phone</th>
        </tr>
        <tbody id="tbody">
            @if (count($users) > 0)
                @foreach ($users as $user)
                    <tr>
                        <td>{{ $user->id }}</td>
                        <td>{{ $user->name }}</td>
                        <td>{{ $user->email }}</td>
                        <td>{{ $user->phone }}</td>
                    </tr>
                @endforeach
            @else
                <tr>
                    <td>No user found.</td>
                </tr>
            @endif
        </tbody>
</table>

<script>
    $(document).ready(function() {
        $("#search").on('keyup', function(){
            var value = $(this).val();
            $.ajax({
                url:"{{ route('userSearch') }}",
                type:"GET",
                data:{'search':value},

                success: function(data) {
                    var users = data.users;
                    var html = '';
                    if(users.length > 0)
                    {
                        for(let i = 0; i < users.length; i++)
                        {
                            html +='<tr>\
                                <td>'+users[i]['id']+'</td>\
                                <td>'+users[i]['name']+'</td>\
                                <td>'+users[i]['email']+'</td>\
                                <td>'+users[i]['phone']+'</td>\
                            </tr>';
                        }
                    }
                    else
                    {
                        html +='<tr>\
```
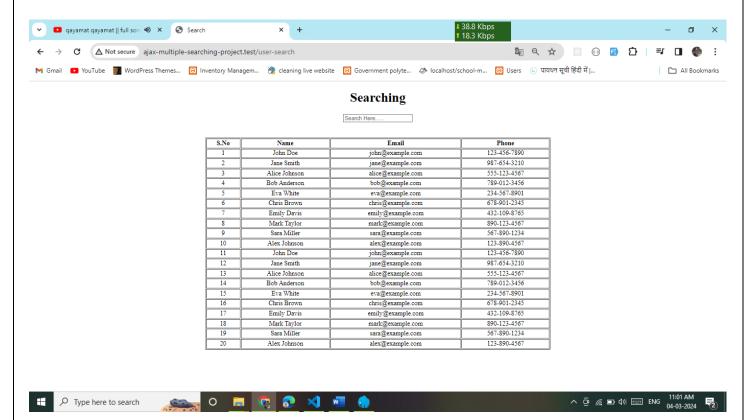
```
                                    <td>No users found</td>\
                                </tr>';
                        }


                        $("#tbody").html(html);
                    }
                });
            });
        });
    </script>
</body>
</html>
```

Frontend

Searching

S.No	Name	Email	Phone
1	John Doe	john@example.com	123-456-7890
2	Jane Smith	jane@example.com	987-654-3210
3	Alice Johnson	alice@example.com	555-123-4567
4	Bob Anderson	bob@example.com	789-012-3456
5	Eva White	eva@example.com	234-567-8901
6	Chris Brown	chris@example.com	678-901-2345
7	Emily Davis	emily@example.com	432-109-8765
8	Mark Taylor	mark@example.com	890-123-4567
9	Sara Miller	sara@example.com	567-890-1234
10	Alex Johnson	alex@example.com	123-890-4567
11	John Doe	john@example.com	123-456-7890
12	Jane Smith	jane@example.com	987-654-3210
13	Alice Johnson	alice@example.com	555-123-4567
14	Bob Anderson	bob@example.com	789-012-3456
15	Eva White	eva@example.com	234-567-8901
16	Chris Brown	chris@example.com	678-901-2345
17	Emily Davis	emily@example.com	432-109-8765
18	Mark Taylor	mark@example.com	890-123-4567
19	Sara Miller	sara@example.com	567-890-1234
20	Alex Johnson	alex@example.com	123-890-4567

Created by – Vishal Kumar Maurya Laravel developer