**1. Migrasi Database**

**Buat Migrasi untuk Tabel histori\_akun**

|  |
| --- |
| php artisan make:migration create\_histori\_akun\_table --create=histori\_akun |

Buka file migrasi di database/migrations dan tambahkan kolom yang diperlukan:

|  |
| --- |
| public function up()  {  Schema::create('histori\_akun', function (Blueprint $table) {  $table->id();  $table->string('nama\_akun');  $table->string('role');  $table->timestamp('waktu\_login');  $table->timestamps();  });  } |

Jalankan migrasi:

|  |
| --- |
| php artisan migrate |

### 2. Model

#### Buat Model HistoriAkun

|  |
| --- |
| php artisan make:model HistoriAkun |

Update app/Models/HistoriAkun.php:

|  |
| --- |
| namespace App\Models;  use Illuminate\Database\Eloquent\Factories\HasFactory;  use Illuminate\Database\Eloquent\Model;  class HistoriAkun extends Model  {  use HasFactory;  protected $fillable = ['nama\_akun', 'role', 'waktu\_login'];  } |

### 3. Controller

#### Buat Controller HistoriAkunController

|  |
| --- |
| php artisan make:controller HistoriAkunController |

Update app/Http/Controllers/HistoriAkunController.php:

|  |
| --- |
| namespace App\Http\Controllers;  use Illuminate\Http\Request;  use App\Models\HistoriAkun;  class HistoriAkunController extends Controller  {  public function destroyAll()  {  HistoriAkun::truncate();  return redirect()->back()->with('status', 'Histori akun berhasil dihapus.');  }  public function index()  {  $historiAkuns = HistoriAkun::all();  return view('historiakun.index', compact('historiAkuns'));  }  } |

### 4. Middleware

#### Buat Middleware HistoriLogin

|  |
| --- |
| php artisan make:middleware HistoriLogin |

Update app/Http/Middleware/HistoriLogin.php:

|  |
| --- |
| namespace App\Http\Middleware;  use Closure;  use Illuminate\Http\Request;  use Illuminate\Support\Facades\Auth;  use App\Models\HistoriAkun;  class HistoriLogin  {  public function handle(Request $request, Closure $next)  {  if (Auth::check()) {  $user = Auth::user();  if (in\_array($user->role, ['admin', 'super\_admin'])) {  HistoriAkun::create([  'nama\_akun' => $user->name,  'role' => $user->role,  'waktu\_login' => now(),  ]);  }  }  return $next($request);  }  } |

Tambahkan middleware ke kernel di app/Http/Kernel.php:

|  |
| --- |
| protected $middlewareGroups = [  'web' => [  // ...  \App\Http\Middleware\HistoriLogin::class,  ],  // ...  ]; |

### 5. Routes

Tambahkan route di routes/web.php:

|  |
| --- |
| use App\Http\Controllers\HistoriAkunController;  Route::delete('/historiakun/destroyAll', [HistoriAkunController::class, 'destroyAll'])->name('historiakun.destroyAll');  Route::get('/historiakun', [HistoriAkunController::class, 'index'])->name('historiakun.index'); |

### 6. View

Update view untuk menampilkan data log akun:

|  |
| --- |
| {{-- @extends('layouts.app')  @section('content') --}}  <x-layout>  <x-slot name="header">  <h2 class="font-semibold text-xl text-gray-800 leading-tight">  {{ \_\_('Histori Akun Admin') }}  </h2>  </x-slot>  {{-- Display all admin accounts with edit and delete buttons --}}  <div class="py-12">  <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">  <div class="bg-white overflow-hidden shadow-xl sm:rounded-lg">  <div class="p-6 sm:px-20 bg-white border-b border-gray-200">  <div class="mt-8 text-2xl">  Welcome to Histori Akun Admin  </div>  <div class="mt-6 text-gray-500">  <p>Here you can see all admin accounts and add a new admin account by clicking the button below</p>  </div>    </div>  <div class="p-6 sm:px-20 bg-white border-b border-gray-200">  <div class="mt-8 text-2xl">  Histori Akun  </div>  <div class="mt-6 text-gray-500">  <p>Here you can see all Histori accounts</p>  </div>  <div class="mt-6">  <form action="{{ route('historiakun.destroyAll') }}" method="POST" onsubmit="return confirm('Apakah Anda yakin ingin menghapus semua histori akun?');">  @csrf  @method('DELETE')  <x-danger-button type="submit">Hapus Histori</x-danger-button>  </form>  </div>  <div class="mt-6 rounded-md border-solid border-2 border-stone-950">  <table class="table-auto w-full ">  <thead>  <tr>  <th scope="col" class="border-solid border-2 border-stone-950 py-3">Nama Akun</th>  <th scope="col" class="border-solid border-2 border-stone-950 py-3">Role</th>  <th scope="col" class="border-solid border-2 border-stone-950 py-3">Waktu Login</th>  </tr>  </thead>  <tbody>  @foreach($historiAkuns as $historiAkun)  <tr>  <td class="border-solid border-2 border-stone-950 py-3">{{ $historiAkun->nama\_akun }}</td>  <td class="border-solid border-2 border-stone-950 py-3">{{ $historiAkun->role }}</td>  <td class="border-solid border-2 border-stone-950 py-3">{{ $historiAkun->waktu\_login }}</td>  </tr>  @endforeach  </tbody>  </table>  </div>  </div>  </div>  </div>  </div>  </x-layout>  {{-- @endsection --}} |

### 7. Controller Method to Pass Data to View

Update your HistoriAkunController to pass the log data to the view:

|  |
| --- |
| use App\Models\HistoriAkun;  public function index()  {  $historiAkuns = HistoriAkun::all();  return view('historiakun.index', compact('historiAkuns'));  } |

### jadikan fungsi program mendata waktu login hanya ketika autentifikasi akun berhasil login. Selain itu jangan di data

### 1. Buat Event Listener

#### Generate Event Listener

|  |
| --- |
| php artisan make:listener LogSuccessfulLogin |

#### Update Event Listener

Update app/Listeners/LogSuccessfulLogin.php:

|  |
| --- |
| namespace App\Listeners;  use Illuminate\Auth\Events\Login;  use Illuminate\Contracts\Queue\ShouldQueue;  use Illuminate\Queue\InteractsWithQueue;  use App\Models\HistoriAkun;  class LogSuccessfulLogin  {  /\*\*  \* Create the event listener.  \*  \* @return void  \*/  public function \_\_construct()  {  //  }  /\*\*  \* Handle the event.  \*  \* @param \Illuminate\Auth\Events\Login $event  \* @return void  \*/  public function handle(Login $event)  {  $user = $event->user;  if (in\_array($user->role, ['admin', 'super\_admin'])) {  HistoriAkun::create([  'nama\_akun' => $user->name,  'role' => $user->role,  'waktu\_login' => now(),  ]);  }  }  } |

### 2. Register Event Listener

Register event listener di EventServiceProvider. Update app/Providers/EventServiceProvider.php:

|  |
| --- |
| namespace App\Providers;  use Illuminate\Auth\Events\Login;  use App\Listeners\LogSuccessfulLogin;  use Illuminate\Foundation\Support\Providers\EventServiceProvider as ServiceProvider;  use Illuminate\Support\Facades\Event;  class EventServiceProvider extends ServiceProvider  {  /\*\*  \* The event listener mappings for the application.  \*  \* @var array  \*/  protected $listen = [  Login::class => [  LogSuccessfulLogin::class,  ],  ];  /\*\*  \* Register any events for your application.  \*  \* @return void  \*/  public function boot()  {  parent::boot();  //  }  } |

### 3. Migrasi Database

Jika belum dilakukan, buat migrasi untuk tabel histori\_akun.

|  |
| --- |
| php artisan make:migration create\_histori\_akun\_table --create=histori\_akun |

Buka file migrasi di database/migrations dan tambahkan kolom yang diperlukan:

|  |
| --- |
| public function up()  {  Schema::create('histori\_akun', function (Blueprint $table) {  $table->id();  $table->string('nama\_akun');  $table->string('role');  $table->timestamp('waktu\_login');  $table->timestamps();  });  } |

Jalankan migrasi:

|  |
| --- |
| php artisan migrate |

### 4. Model

Jika belum dibuat, buat model HistoriAkun.

|  |
| --- |
| php artisan make:model HistoriAkun |

Update app/Models/HistoriAkun.php:

|  |
| --- |
| namespace App\Models;  use Illuminate\Database\Eloquent\Factories\HasFactory;  use Illuminate\Database\Eloquent\Model;  class HistoriAkun extends Model  {  use HasFactory;  protected $fillable = ['nama\_akun', 'role', 'waktu\_login'];  } |

### 5. Controller

Update HistoriAkunController jika belum dilakukan:

|  |
| --- |
| namespace App\Http\Controllers;  use Illuminate\Http\Request;  use App\Models\HistoriAkun;  class HistoriAkunController extends Controller  {  public function destroyAll()  {  HistoriAkun::truncate();  return redirect()->back()->with('status', 'Histori akun berhasil dihapus.');  }  public function index()  {  $historiAkuns = HistoriAkun::all();  return view('historiakun.index', compact('historiAkuns'));  }  } |

### 6. Routes

Tambahkan route di routes/web.php:

|  |
| --- |
| use App\Http\Controllers\HistoriAkunController;  Route::delete('/historiakun/destroyAll', [HistoriAkunController::class, 'destroyAll'])->name('historiakun.destroyAll');  Route::get('/historiakun', [HistoriAkunController::class, 'index'])->name('historiakun.index'); |

### 7. View

Update view untuk menampilkan data log akun:

|  |
| --- |
| {{-- @extends('layouts.app')  @section('content') --}}  <x-layout>  <x-slot name="header">  <h2 class="font-semibold text-xl text-gray-800 leading-tight">  {{ \_\_('Histori Akun Admin') }}  </h2>  </x-slot>  {{-- Display all admin accounts with edit and delete buttons --}}  <div class="py-12">  <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">  <div class="bg-white overflow-hidden shadow-xl sm:rounded-lg">  <div class="p-6 sm:px-20 bg-white border-b border-gray-200">  <div class="mt-8 text-2xl">  Welcome to Histori Akun Admin  </div>  <div class="mt-6 text-gray-500">  <p>Here you can see all admin accounts and add a new admin account by clicking the button below</p>  </div>    </div>  <div class="p-6 sm:px-20 bg-white border-b border-gray-200">  <div class="mt-8 text-2xl">  Histori Akun  </div>  <div class="mt-6 text-gray-500">  <p>Here you can see all Histori accounts</p>  </div>  <div class="mt-6">  <form action="{{ route('historiakun.destroyAll') }}" method="POST" onsubmit="return confirm('Apakah Anda yakin ingin menghapus semua histori akun?');">  @csrf  @method('DELETE')  <x-danger-button type="submit">Hapus Histori</x-danger-button>  </form>  </div>  <div class="mt-6 rounded-md border-solid border-2 border-stone-950">  <table class="table-auto w-full ">  <thead>  <tr>  <th scope="col" class="border-solid border-2 border-stone-950 py-3">Nama Akun</th>  <th scope="col" class="border-solid border-2 border-stone-950 py-3">Role</th>  <th scope="col" class="border-solid border-2 border-stone-950 py-3">Waktu Login</th>  </tr>  </thead>  <tbody>  @foreach($historiAkuns as $historiAkun)  <tr>  <td class="border-solid border-2 border-stone-950 py-3">{{ $historiAkun->nama\_akun }}</td>  <td class="border-solid border-2 border-stone-950 py-3">{{ $historiAkun->role }}</td>  <td class="border-solid border-2 border-stone-950 py-3">{{ $historiAkun->waktu\_login }}</td>  </tr>  @endforeach  </tbody>  </table>  </div>  </div>  </div>  </div>  </div>  </x-layout>  {{-- @endsection --}} |

### 8. Register Middleware

Pastikan middleware HistoriLogin tidak terdaftar di kernel, karena kita sudah menggunakan event listener untuk mencatat login.