

LAPORAN UJIAN AKHIR SEMESTER
PEMROGRAMAN WEB BERBASIS FRAMEWORK
Studi Kasus Rekam Medis
4 Januari 2021



Anggota Kelompok:

Tri Wahyu Permana	(151911513004)
Moch. Rafif Meirvandy W	(151911513019)

PROGRAM STUDI D3 SISTEM INFORMASI
FAKULTAS VOKASI
UNIVERSITAS AIRLANGGA
2020

1. Latar Belakang

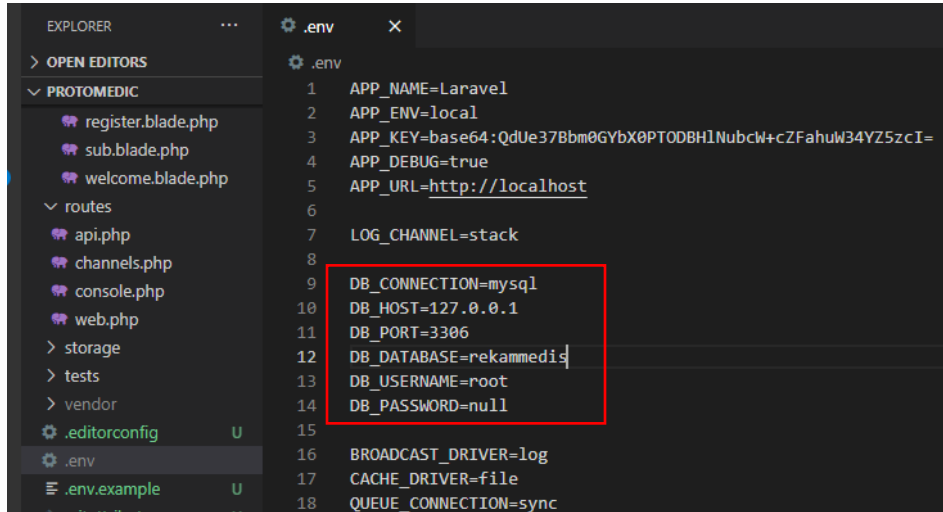
Projek yang kelompok kami buat bernama “Medical Center”. Website ini berisi segala informasi yang ada di Sistem Informasi Rekam Medis, mulai dari data pegawai seperti dokter dan petugas admin, data para pasien, obat-obatan yang disediakan hingga data rekam medis sendiri. Untuk tampilannya dibuat dengan dua role atau dua persepsi yaitu tampilan user admin yang akan mengurus segala data dalam database seperti memasukkan data baru atau merubah data, dan juga tampilan untuk pasien yang akan menampilkan segala histori rekam medis mereka.

Untuk mengakses website kami pasien harus registrasi terlebih dahulu dan setelah login mereka hanya dapat melihat histori rekam medis mereka dikarenakan segala data pasien lain bersifat rahasia dan hanya dapat diakses dengan akun admin. Untuk dapat login sebagai admin harus diregistrasikan secara khusus oleh admin lain agar tidak menimbulkan penyalahgunaan wewenang dalam mengakses data master didalamnya.

Akun admin memiliki wewenang penuh dalam website, seperti mengurus segala macam data master yaitu data dokter, pasien, obat, dan admin lain. Admin juga mengurus proses rekam medis yang nantinya histori tersebut dapat dilihat oleh masing masing pasien.

2. Pembuatan Halaman Rekam Medis

Karena pada website kita akan menggunakan database jadi sebelumnya kita harus sudah menyiapkan database tersebut dan konfigurasi ke laravel. Untuk mengkonfigurasikannya kita bisa atur pada file .env



```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:QdUe37Bm0GYbX0PT0DBH1NubcW+cZFahuW34YZ5zcI=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8
9 DB_CONNECTION=mysql
10 DB_HOST=127.0.0.1
11 DB_PORT=3306
12 DB_DATABASE=rekammedis
13 DB_USERNAME=root
14 DB_PASSWORD=null
15
16 BROADCAST_DRIVER=log
17 CACHE_DRIVER=file
18 QUEUE_CONNECTION=sync
```

Karena menggunakan eloquent buat model rekam_medis untuk handle tabel rekam_medis pada database. Model dibuat pada folder app/



```
1 <?php
2
3 namespace App;
4
5 use Illuminate\Database\Eloquent\Model;
6
7 class rekam_medis extends Model
8 {
9     protected $table = "rekam_medis";
10     protected $primaryKey = "id_rekam_medis";
11     protected $fillable = ['id_rekam_medis', 'id_pasien', 'id_petugas', 'id_dokter', 'tgl_periksa', 'diagnosa', 'keterangan'];
12
13     public function pasien(){
14         return $this->belongsTo(pasien::class, 'id_pasien');
15     }
16     public function petugas_admin(){
17         return $this->belongsTo(petugas_admin::class, 'id_petugas');
18     }
19     public function dokter(){
20         return $this->belongsTo(dokter::class, 'id_dokter');
21     }
22
23     public function resep_obat(){
24         return $this->hasMany(rekam_medis::class, 'id_rekam_medis');
25     }
26 }
```

Selanjutnya buat route yang nantinya mengarahkan ke index yang menampilkan isi dari tabel rekam medis pada database.

```
Route::get('/rekam_medis', 'Rekam_MedisController@indexRekam_Medis');
```

Pada route tersebut akan memerintahkan method `indexRekam_Medis` pada controller `Rekam_MedisController` pada saat route `/rekam_medis` diakses.

Buat controller untuk memanggil `indexRekam_Medis` pada `Rekam_MedisController`.

```
class Rekam_MedisController extends Controller
{
    public function indexRekam_Medis(Request $request)
    {
        if($request->has('cari')){
            $pasien = pasien::where('nama_pasien','LIKE','%'.$request->cari.'%');
            $petugas_admin = petugas_admin::where('nama_petugas','LIKE','%'.$request->cari.'%');
            $dokter = dokter::where('nama_petugas','LIKE','%'.$request->cari.'%');
            $rekam_medis = rekam_medis::where('id_rekam_medis','LIKE','%'.$request->cari.'%')
            ->orWhere('tgl_periksa','LIKE','%'.$request->cari.'%')
            ->orWhere('diagnosa','LIKE','%'.$request->cari.'%')
            ->orWhere('biaya_jasa','LIKE','%'.$request->cari.'%')
            ->get();
        }else{
            $pasien = pasien::all();
            $petugas_admin = petugas_admin::all();
            $dokter = dokter::all();
            $rekam_medis = rekam_medis::all();
        }
        // mengirim data ke view rekam medis
        return view('index/indexRekam_Medis',compact('rekam_medis'),compact('pasien'),compact('petugas_admin'),compact('dokter'));
    }
}
```

Karena kita menggunakan model jangan lupa untuk menambahkan model yang akan digunakan nantinya. Pada `rekam_medis` membutuhkan model `pasien`, `dokter`, `petugas`, dan `rekam_medis`.

```
p > Http > Controllers > Rekam_MedisController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\DB;
7  use App\rekam_medis;
8  use App\pasien;
9  use App\petugas_admin;
10 use App\dokter;
```

Dan buat viewnya untuk menampilkan data `IndexRekam_Medis`. view berada pada folder `app/view/`

```

<table id="bootstrap-data-table" class="table table-striped table-bordered">
<thead>
<tr>
<th>Id Rekam Medis</th>
<th>Nama Pasien</th>
<th>Nama Petugas</th>
<th>Nama Dokter</th>
<th>Tgl Periksa</th>
<th>Diagnosa</th>
<th>Keterangan</th>
<th>Biaya Jasa</th>
<th>Ops</th>
</tr>
</thead>
<tbody>
@foreach($rekam_medis as $rm)
<tr>
<td>{{ $rm->id_rekam_medis }}</td>
<td>{{ $rm->pasien->nama_pasien }}</td>
<td>{{ $rm->petugas_admin->nama_petugas }}</td>
<td>{{ $rm->dokter->nama_dokter }}</td>
<td>{{ $rm->tgl_periksa }}</td>
<td>{{ $rm->diagnosa }}</td>
<td>{{ $rm->keterangan }}</td>
<td>{{ $rm->biaya_jasa }}</td>
<td>
<a href="/rekam_medis/edit/{{ $rm->id_rekam_medis }}" class="btn btn-primary btn-sm">Edit</a>
</td>
</tr>
@endforeach
</tbody>
</table>

```

Coba akses php artisan serve pada terminal dan buka localhost:8000/rekam_medis

Id Rekam Medis	Nama Pasien	Nama Petugas	Nama Dokter	Tgl Periksa	Diagnosa	Keterangan	Biaya Jasa
1	Abel	Andy	dr. Oki	2021-01-01	Flu		25000
2	Abel	Wahyu	dr. Ade	2021-01-01	Batuk	Terserah	20000
3	Mochammad	Rafif	dr. Ipul	2021-01-01	Diabetes	Terserah	100000
4	Anto	Rafif	dr. Ade	2021-01-01	Asam urat		120000
5	Abel	Wahyu	dr. Oki	2021-01-01	Flu	Terserah	20000

Sekarang tambah button create atau tambah data pada indexRekam_Medis

```

<a href="/rekam_medis/tambah" button type="button" class="btn btn-success btn-sm">
<i class="fa fa-plus-square"></i>&nbsp; Tambah Data Rekam Medis</button></a>

```

Button tersebut akan diarahkan ke route /rekam_medis/tambah

```

Route::get('/rekam_medis/tambah', 'Rekam_MedisController@tambahRekam_Medis');

```

Buat juga controller untuk mengarahkan route ke view tambahRekam_Medis

```

public function tambahRekam_Medis()
{
    // memanggil view tambah
    $pasien = DB::table('pasien')->get();
    $petugas_admin = DB::table('petugas_admin')->get();
    $dokter = DB::table('dokter')->get();
    return view('tambah/tambahRekam_Medis', ['pasien' => $pasien, 'petugas_admin' => $petugas_admin, 'dokter' => $dokter]);
}

```

Dan buat form view tambahRekam_Medis nya

```

39 <br>
40 <a href="/rekam_medis"><button type="button" class="btn btn-danger btn-sm"><i class="fa fa-arrow-left"></i>&nbsp;Kembali</button></a>
41 <br>
42 </div>
43 <div class="card-body">
44 <form action="/rekam_medis/store" method="post">
45 {{ csrf_field() }}
46 <input type="hidden" name="id_rekam_medis"> <br/>
47
48 <div class="form-group">
49 <label>Nama Pasien</label>
50 <select name="id_pasien" required="required" class="form-control" placeholder=". . .">
51 <option selected></option>
52 @foreach ($pasien as $p)<option value="{{ $p -> id_pasien }}">{{ $p -> id_pasien }} - {{ $p -> nama_pasien}}</option>@endforeach</select></div>
53 </div>
54
55 @foreach ($petugas_admin as $pa)
56 @if (Auth::user()->id == $pa -> id)
57 <input type="hidden" name="id_petugas" value="{{ $pa -> id_petugas }}">
58 @endif
59 @endforeach
60
61 <div class="form-group">
62 <label>Nama Dokter</label>
63 <select name="id_dokter" required="required" class="form-control" placeholder=". . .">
64 <option selected></option>
65 @foreach ($dokter as $d)<option value="{{ $d -> id_dokter }}">{{ $d -> id_dokter }} - {{ $d -> nama_dokter}}</option>@endforeach</select></div>
66 </div>
67
68 <div class="form-group">
69 <label>Tgl Periksa</label>
70 <input type="date" name="tgl_periksa" required="required" class="form-control" placeholder=". . .">
71 </div>

```

Perhatikan bagian bold form yang telah diisi akan diarahkan ke route /rekam_medis/store.

Dan jangan lupa beri button submit pada akhir form

```

<div class="form-group">
<label>Diagnosa</label>
<input type="text" name="diagnosa" class="form-control" placeholder=". . .">
</div>

<div class="form-group">
<label>Keterangan</label>
<input type="text" name="keterangan" class="form-control" placeholder=". . .">
</div>

<div class="form-group">
<label>Biaya Jasa</label>
<input type="number" name="biaya_jasa" required="required" class="form-control" placeholder=". . .">
</div>
<br>

<div class="form-group">
<input type="submit" class="btn btn-success" value="Simpan Data">
</div>
</form>

```

Selanjutnya buat lagi route rekam_medis/store

```

Route::post('/rekam_medis/store', 'Rekam_MedisController@storeRekam_Medis');

```

Route tersebut menuju metode storeRekam_Medis pada controller Rekam_MedisController.

```
// method untuk insert data ke table
public function storeRekam_Medis(Request $request)
{
    // insert data ke table
    DB::table('rekam_medis')->insert([
        'id_rekam_medis' => $request->id_rekam_medis,
        'id_pasien' => $request->id_pasien,
        'id_petugas' => $request->id_petugas,
        'id_dokter' => $request->id_dokter,
        'tgl_periksa' => $request->tgl_periksa,
        'diagnosa' => $request->diagnosa,
        'keterangan' => $request->keterangan,
        'biaya_jasa' => $request->biaya_jasa
    ]);
    // alihkan halaman ke halaman
    return redirect('/rekam_medis');
}
```

Metode store digunakan untuk mencapture inputan dari view form tambah tadi dan dimasukan ke database tabel rekam_medis.

Tambah Data Rekam Medis

Kembali

Nama Pasien

Nama Dokter

Tgl Periksa

hh/bb/tttt

Diagnosa

...

Keterangan

...

Tambahkan juga tombol edit di view index

```
<td>
|   <a href="/rekam_medis/edit/{{ $rm->id_rekam_medis }}" class="btn btn-primary btn-sm">Edit</a>
|
|   </td>
```

Sama seperti sebelumnya akan diarahkan ke route baru

```
Route::get('/rekam_medis/edit/{id_rekam_medis}','Rekam_MedisController@editRekam_Medis');
```

Buat juga metode untuk menampilkan view form edit pada controller

```
public function editRekam_Medis($id_rekam_medis)
{
    $rekam_medis = DB::table('rekam_medis')->where('id_rekam_medis',$id_rekam_medis)->get();
    $pasien = DB::table('pasien')->get();
    $petugas_admin = DB::table('petugas_admin')->get();
    $dokter = DB::table('dokter')->get();
    // passing data rekam_medis yang didapat ke view edit.blade.php
    return view('edit/editRekam_Medis',['rekam_medis' => $rekam_medis,'pasien' => $pasien,'petugas_admin' => $petugas_admin,'dokter' => $dokter]);
}
```

Dan isi viewnya seperti ini

```
@foreach($rekam_medis as $rm)
<form action="/rekam_medis/update" method="post">
{{ csrf_field() }}
<input type="hidden" name="id_rekam_medis" value="{{ $rm->id_rekam_medis }}"> <br/>

<div class="form-group">
<label>Nama Pasien</label>
<select name="id_pasien" required="required" class="form-control" placeholder=". . .">
@foreach ($pasien as $p)
    @if ($rm->id_pasien == $p->id_pasien)<option value="{{ $p -> id_pasien }}" selected>{{ $p -> nama_pasien }}</option>
    @else <option value="{{ $p -> id_pasien }}">{{ $p -> nama_pasien }}</option>
    @endif
@endforeach
</select></div>

@foreach ($petugas_admin as $pa)
@if (Auth::user()->id == $pa -> id)
<input type="hidden" name="id_petugas" value="{{ $pa -> id_petugas }}">
@endif
@endforeach

<div class="form-group">
<label>Nama Dokter</label>
<select name="id_dokter" required="required" class="form-control" placeholder=". . .">
@foreach ($dokter as $d)
    @if ($rm->id_dokter == $d->id_dokter)<option value="{{ $d -> id_dokter }}" selected>{{ $d -> nama_dokter }}</option>
    @else <option value="{{ $d -> id_dokter }}">{{ $d -> nama_dokter }}</option>
    @endif
@endforeach
</select></div>

<input type="submit" value="Update">
</form>
```

View edit mirip dengan view tambah tetapi view ini juga menampilkan data lama yang nantinya akan diubah.

Bagian bold menunjukan kalo submit dari form tersebut dicapture atau ditujukan ke route /rekam_medis/update

```
Route::post('/rekam_medis/update','Rekam_MedisController@updateRekam_Medis');
```

Buat metode updateRekam_medis pada controller untuk menangkap perubahan dan disimpan di database.


```

public function updateRekam_Medis(Request $request)
{
    // update data
    DB::table('rekam_medis')->where('id_rekam_medis',$request->id_rekam_medis)->update([
        //'id_rekam_medis' => $request->id_rekam_medis,
        'id_pasien' => $request->id_pasien,
        'id_petugas' => $request->id_petugas,
        'id_dokter' => $request->id_dokter,
        'tgl_periksa' => $request->tgl_periksa,
        'diagnosa' => $request->diagnosa,
        'keterangan' => $request->keterangan,
        'biaya_jasa' => $request->biaya_jasa
    ]);
    return redirect('/rekam_medis');
}

```

View untuk form edit seperti berikut.

Nama Pasien

Abel

Nama Dokter

dr. Oki

Tgl Periksa

01/01/2021

Diagnosa

Flu

Keterangan

...

Biaya Jasa

25000

Simpan Data

Powered by 000webhost

3. Integrasi Halaman CRUD Ke Menu Sesuai Aktor

Karena terdapat dua aktor buat 2 tampilan home dengan menu yang berbeda.

View untuk aktor admin akan memiliki menu pengelola seluruh data master dan data transaksi.

```
<div>
  <a class="navbar-brand" href="">MEDICAL CENTER</a>
</div>
<div class="text-muted">Rafif - Wahyu</div>
<br/>
<br/>
</div>

<div id="main-menu" class="main-menu collapse navbar-collapse">
  <ul class="nav navbar-nav">
    <li>
      <a href="/"> <i class="menu-icon fa fa-home"></i>Home </a>
    </li>
    <li>
      <a href="/dokter"> <i class="menu-icon fa fa-stethoscope"></i>Dokter </a>
    </li>
    <li>
      <a href="/obat"> <i class="menu-icon fa fa-medkit"></i>Obat </a>
    </li>
    <li>
      <a href="/pasien"> <i class="menu-icon fa fa-wheelchair"></i>Pasien </a>
    </li>
    <li>
      <a href="/petugas_admin"> <i class="menu-icon fa fa-user-md"></i>Petugas </a>
    </li>
    <li>
      <a href="/rekam_medis"> <i class="menu-icon fa fa-hospital-o"></i>Rekam Medis </a>
    </li>
    <li>
      <a href="/resep_obat"> <i class="menu-icon fa fa-flask"></i>Resep Obat </a>
    </li>
  </ul>
</div>
```

Masing masing menu akan diarahkan ke route setiap halaman yang dituju seperti halaman pengelolaan rekam medis yang telah dibuat sebelumnya.

Untuk view aktor pasien hanya berisi halaman histori rekam medisnya saja.

```

<div>
  <a class="navbar-brand" href="">MEDICAL CENTER</a>
</div>
<div class="text-muted">Rafif - Wahyu</div>
<br/>
<br/>
</div>

<div id="main-menu" class="main-menu collapse navbar-collapse">
  <ul class="nav navbar-nav">
    <li>
      <a href="/homep"> <i class="menu-icon fa fa-home"></i>Home </a>
    </li>
    <li>
      <a href="/rekam_medis/histori"> <i class="menu-icon fa fa-square"></i>Histori Rekam Medis</a>
    </li>
  </ul>
</div><!-- /.navbar-collapse -->

```

Diarahkan ke route untuk halaman histori rekam medis pasien tersebut. Tampilan jika kedua menu berbeda tadi dijalankan akan seperti ini.

Tampilan CRUD dan menu aktor admin.

- Home
- Dokter
- Obat
- Pasien
- Petugas
- Rekam Medis
- Resep Obat

Data Rekam Medis

Tambah Data Rekam Medis

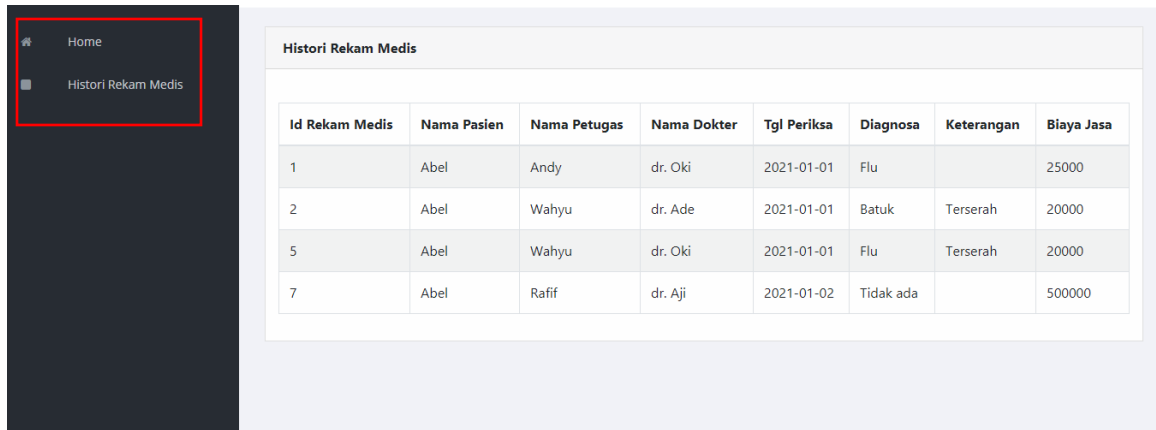
Lihat per Pasien

Search ...

Id Rekam Medis	Nama Pasien	Nama Petugas	Nama Dokter	Tgl Periksa	Diagnosa	Keterangan	Biaya Jasa	Opsi
1	Abel	Andy	dr. Oki	2021-01-01	Flu		25000	Edit
2	Abel	Wahyu	dr. Ade	2021-01-01	Batuk	Terserah	20000	Edit
3	Mochammad	Rafif	dr. Ipul	2021-01-01	Diabetes	Terserah	100000	Edit
4	Anto	Rafif	dr. Ade	2021-01-01	Asam urat		120000	Edit
5	Abel	Wahyu	dr. Oki	2021-01-01	Flu	Terserah	20000	Edit

Terdapat menu pengelolaan data master dan transaksi, juga terdapat menu CRUD data.

Sedangkan tampilan menu aktor pasien.



Hanya ada menu untuk menampilkan histori rekam medis tanpa menu CRUD data.

4. Halaman Login dan Integrasi Menu sesuai Aktor

Membuat route register, login, dan logout

```
Route::get('/', 'AuthController@showFormLogin')->name('login');
Route::get('login', 'AuthController@showFormLogin')->name('login');
Route::post('login', 'AuthController@login');
Route::get('register', 'AuthController@showFormRegister')->name('register');
Route::post('register', 'AuthController@register');
Route::get('logout', 'AuthController@logout')->name('logout');
```

Untuk register admin berada dalam middleware admin yang artinya admin baru hanya bisa diregisterkan oleh admin yang sudah memiliki akun.

```
Route::group(['middleware' => 'admin'], function () {
    Route::get('/petugas_admin/regis', 'Petugas_AdminController@aksesAdmin');
```

Membuat controller AuthController.php

php artisan make:controller AuthController

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\Auth;
7 use Validator;
8 use Hash;
9 use Session;
10 use App\User;
11
12
13 class AuthController extends Controller
14 {
15     public function showFormLogin()
16     {
17         if (Auth::check()) { // true sekalian session file
18             //Login Success
19             return redirect()->route('home');
20         }
21         return view('login');
22     }
23 }
```

Login function, jika role user login tersebut = 1 yaitu admin maka akan diarahkan ke dashboard admin, jika tidak maka akan diarahkan ke dashboard pasien.

```
public function login(Request $request)
{
    $rules = [
        'email'          => 'required|email',
        'password'        => 'required|string'
    ];
    $messages = [
        'email.required'  => 'Email wajib diisi',
        'email.email'     => 'Email tidak valid',
        'password.required' => 'Password wajib diisi',
        'password.string' => 'Password harus berupa string'
    ];
    $validator = Validator::make($request->all(), $rules, $messages);
    if($validator->fails()){
        return redirect()->back()->withErrors($validator)->withInput($request->all());
    }
    $data = [
        'email' => $request->input('email'),
        'password' => $request->input('password'),
    ];
    Auth::attempt($data);
    if (Auth::check()) { // true sekalian session field di users nanti bisa dipanggil via Auth
        //Login Success
        $role = Auth::user()->role;
        if($role == "1"){
            return redirect()->route('home');
        }
        else{
            return redirect()->route('homep');
        }
    } else { // false
        //Login Fail
        Session::flash('error', 'Email atau password salah!');
    }
}
```

Register function untuk user pasien

```
public function showFormRegister()
{
    return view('register');
}

public function register(Request $request)
{
    $rules = [
        'name'          => 'required|min:3|max:35',
        'email'          => 'required|email|unique:users,email',
        'password'        => 'required|confirmed'
    ];
    $messages = [
        'name.required'  => 'Nama Lengkap wajib diisi',
        'name.min'       => 'Nama lengkap minimal 3 karakter',
        'name.max'       => 'Nama lengkap maksimal 35 karakter',
        'email.required' => 'Email wajib diisi',
        'email.email'     => 'Email tidak valid',
        'email.unique'    => 'Email sudah terdaftar',
        'password.required' => 'Password wajib diisi',
        'password.confirmed' => 'Password tidak sama dengan konfirmasi password'
    ];
    $validator = Validator::make($request->all(), $rules, $messages);

    if($validator->fails()){
        return redirect()->back()->withErrors($validator)->withInput($request->all());
    }

    $user = new User;
    $user->name = ucwords(strtolower($request->name));
    $user->email = strtolower($request->email);
    $user->password = Hash::make($request->password);
}
```

Register untuk user admin berada pada controller petugas_admin karena hanya admin yang telah login yang bisa mendaftarkan user admin baru.

```
pemesanan.php AuthController.php web.php Petugas_AdminController.php
119 //UNTUK MENJADIKAN PETUGAS SEBAGAI ADMIN
120 public function aksesAdmin()
121 {
122     return view('tambah/tambahAdmin');
123 }
124 public function storeAdmin(Request $request)
125 {
126     $rules = [
127         'name' => 'required|min:3|max:35',
128         'email' => 'required|email|unique:users,email',
129         'password' => 'required|confirmed',
130         'role' => 'required'
131     ];
132
133     $messages = [
134         'name.required' => 'Nama Lengkap wajib diisi',
135         'name.min' => 'Nama lengkap minimal 3 karakter',
136         'name.max' => 'Nama lengkap maksimal 35 karakter',
137         'email.required' => 'Email wajib diisi',
138         'email.email' => 'Email tidak valid',
139         'email.unique' => 'Email sudah terdaftar',
140         'password.required' => 'Password wajib diisi',
141         'password.confirmed' => 'Password tidak sama dengan konfirmasi password'
142     ];
143
144     $validator = Validator::make($request->all(), $rules, $messages);
145
146     if($validator->fails()){
147         return redirect()->back()->withErrors($validator)->withInput($request->all());
148     }
149
150     $user = new User;
151     $user->name = ucwords(strtolower($request->name));
152     $user->email = strtolower($request->email);
```

View login

```
AuthController.php web.php Petugas_AdminController.php login.blade.php
<div class="container">
<div class="col-md-4 offset-md-4 mt-5">
<div class="card">
<div class="card-header">
<h3 class="text-center">Login</h3>
</div>
<form action="{{ route('login') }}" method="post">
@csrf
<div class="card-body">
@if(session('errors'))
<div class="alert alert-danger alert-dismissible fade show" role="alert">
Something it's wrong:
<button type="button" class="close" data-dismiss="alert" aria-label="Close">
<span aria-hidden="true"></span>
</button>
<ul>
@foreach ($errors->all() as $error)
<li>{{ $error }}</li>
@endforeach
</ul>
</div>
@endif
@if (Session::has('success'))
<div class="alert alert-success">
{{ Session::get('success') }}
</div>
@endif
@if (Session::has('error'))
<div class="alert alert-danger">
{{ Session::get('error') }}
</div>
@endif
<div class="form-group">
<label for="username">Email</label>
```

Aktor pasien dan admin login melalui view yang sama

rafifwahyu.000webhostapp.com/login

Login

Email

Password

Log In

Belum punya akun? [Register sekarang!](#)

View register user pasien

```
AuthController.php | web.php | Petugas_AdminController.php | login.blade.php | register.blade.php
<div class="col-md-4 offset-md-4 mt-5">
  <div class="card">
    <div class="card-header">
      <h3 class="text-center">Register</h3>
    </div>
    <form action="{{ route('register') }}" method="post">
      @csrf
      <div class="card-body">
        @if(session('errors'))
          <div class="alert alert-danger alert-dismissible fade show" role="alert">
            Something it's wrong:
            <button type="button" class="close" data-dismiss="alert" aria-label="Close">
              <span aria-hidden="true">x</span>
            </button>
            <ul>
              @foreach ($errors->all() as $error)
                <li>{{ $error }}</li>
              @endforeach
            </ul>
          </div>
        @endif
        <div class="form-group">
          <label for=""><strong>Nama Lengkap</strong></label>
          <input type="text" name="name" class="form-control" placeholder="Nama Lengkap">
        </div>
        <div class="form-group">
          <label for=""><strong>Email</strong></label>
          <input type="text" name="email" class="form-control" placeholder="Email">
        </div>
        <div class="form-group">
          <label for=""><strong>Password</strong></label>
          <input type="password" name="password" class="form-control" placeholder="Password">
        </div>
        <div class="form-group">
```


Tampilan register user pasien

rafiwahyu.000webhostapp.com/register

Register

Nama Lengkap

Email

Password

Konfirmasi Password

[Register](#)

[Sudah punya akun? Login sekarang!](#)

View register user admin

```
AuthController.php web.php Petugas_AdminController.php login.blade.php register.blade.php tambahAdmin.blade.php
<div class="container">
  <div class="col-md-4 offset-md-4 mt-5">
    <div class="card">
      <div class="card-header">
        <h3 class="text-center">Register Untuk Mengakses Admin</h3>
      </div>
      <form action="/petugas_admin/storeAdmin" method="post">
        @csrf
        <div class="card-body">
          @if(session('errors'))
            <div class="alert alert-danger alert-dismissible fade show" role="alert">
              Something it's wrong:
              <button type="button" class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">x</span>
              </button>
              <ul>
                @foreach ($errors->all() as $error)
                  <li>{{ $error }}</li>
                @endforeach
              </ul>
            </div>
          @endif
          <div class="form-group">
            <label for=""><strong>Username</strong></label>
            <input type="text" name="name" class="form-control" placeholder="Nama Lengkap">
          </div>
          <div class="form-group">
            <label for=""><strong>Email</strong></label>
            <input type="text" name="email" class="form-control" placeholder="Email">
          </div>
          <div class="form-group">
            <label for=""><strong>Password</strong></label>
            <input type="password" name="password" class="form-control" placeholder="Password">
          </div>
        </div>
      </form>
    </div>
  </div>
</div>
```


Register user admin terletak pada menu tabel petugas, untuk mengaksesnya diperlukan login melalui admin yang sudah terdaftar.

u.000webhostapp.com/petugas_admin

Petugas

Data Petugas

Tambah Data Petugas

Search ...

Regis Akun Admin

Id	Nama	User	Alamat	Kota	NoTelp	Opsi
3	Rafif	rafif@gmail.com	Bratang	Surabaya	089089089089	Edit
6	Andy	andy2@gmail.com	Mulyorejo	Sidoarjo	081081081081	Edit

rafifwahyu.000webhostapp.com/petugas_admin/regis

Register Untuk Mengakses Admin

Username

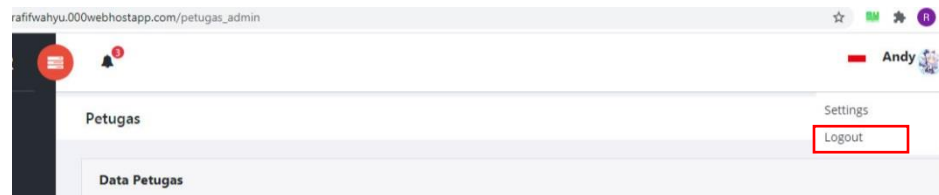
Email

Password

Konfirmasi Password

Register

Logout berada di kanan atas



Membuat batasan/middleware tiap aktor agar hanya bisa mengakses sesuai menu aktornya saja.

Middleware admin, role = 1 sebagai user admin.

```
1 <?php
2
3 namespace App\Http\Middleware;
4
5 use Closure;
6
7 class Admin
8 {
9     /**
10      * Handle an incoming request.
11      *
12      * @param \Illuminate\Http\Request $request
13      * @param \Closure $next
14      * @return mixed
15      */
16     public function handle($request, Closure $next)
17     {
18         if ($request->user()->role == 1){
19             return $next($request);
20         }
21         abort(403, "Cannot access to restricted page");
22     }
23 }
24
```

Middleware pasien, role = 2 sebagai user pasien.

```
pemesanan.php AuthController.php tambahAdmin.blade.php Admin.php Pasien.php
1 <?php
2
3 namespace App\Http\Middleware;
4
5 use Closure;
6
7 class Pasien
8 {
9     /**
10      * Handle an incoming request.
11      *
12      * @param \Illuminate\Http\Request $request
13      * @param \Closure $next
14      * @return mixed
15      */
16     public function handle($request, Closure $next)
17     {
18         if ($request->user()->role == 2) {
19             return $next($request);
20         }
21         abort(403, "Cannot access to restricted page");
22     }
23 }
24
```

Menambah middleware admin dan pasien pada variabel \$routeMiddleware di kernel.php

```
pemesanan.php AuthController.php tambahAdmin.blade.php Admin.php Pasien.php Kernel.php
53 protected $routeMiddleware = [
54     'auth' => \App\Http\Middleware\Authenticate::class,
55     'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
56     'bindings' => \Illuminate\Routing\Middleware\SubstituteBindings::class,
57     'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,
58     'can' => \Illuminate\Auth\Middleware\Authorize::class,
59     'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
60     'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,
61     'signed' => \Illuminate\Routing\Middleware\ValidateSignature::class,
62     'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
63     'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
64     'admin' => \App\Http\Middleware\Admin::class,
65     'pasien' => \App\Http\Middleware\Pasien::class,
66 ];
67
```

Membuat middleware group function di web.php

```
pemesanan.php AuthController.php tambahAdmin.blade.php Admin.php Pasien.php Kernel.php web.php
23 Route::group(['middleware' => 'pasien'], function () {
24     Route::get('home', 'HomeController@index')->name('home');
25     Route::get('/rekam_medis/histori', 'Rekam_MedisController@indexRekam_MedisHistori');
26
27 });
28
29 Route::group(['middleware' => 'admin'], function () {
30     Route::get('home', 'HomeController@index')->name('home');
31
32     Route::get('/dokter', 'DokterController@indexDokter');
33     Route::get('/dokter/tambah', 'DokterController@tambahDokter');
34     Route::post('/dokter/store', 'DokterController@storeDokter');
35     Route::get('/dokter/edit/{id_dokter}', 'DokterController@editDokter');
36     Route::post('/dokter/update', 'DokterController@updateDokter');
37     Route::get('/dokter/hapus/{id_dokter}', 'DokterController@hapus');
38
39     Route::get('/obat', 'ObatController@indexObat');
40     Route::get('/obat/tambah', 'ObatController@tambahObat');
```

Pengujian user pasien mencoba mengakses menu yang hanya bisa diakses oleh admin.

| rafifwahyu.000webhostapp.com/petugas_admin



403 | Cannot access to restricted page