membuat API Register, Login, dan Logout dengan menggunakan MD5 pada Laravel dan React JS, berikut adalah langkah-langkahnya:

Bagian Laravel (Backend):

1. Pastikan Anda telah menginstal Laravel di sistem Anda. Jika belum, Anda dapat menginstalnya dengan menjalankan perintah composer global require laravel/installer.
2. Buat proyek Laravel baru dengan menjalankan perintah laravel new nama-proyek.
3. Masuk ke direktori proyek dengan menjalankan perintah cd nama-proyek.
4. Buat migrasi untuk tabel pengguna dengan menjalankan perintah php artisan make:migration create\_users\_table --create=users.
5. Buka file migrasi yang baru saja dibuat (berlokasi di direktori database/migrations) dan perbarui metode up() untuk mencakup kolom yang diperlukan seperti name, email, password, dan lainnya. Contoh:

php

public function up()

{

Schema::create('users', function (Blueprint $table) {

$table->id();

$table->string('name');

$table->string('email')->unique();

$table->string('password');

$table->timestamps();

});

}

1. Jalankan migrasi untuk membuat tabel pengguna dengan perintah php artisan migrate.
2. Buat model pengguna dengan menjalankan perintah php artisan make:model User.
3. Buka model User (berlokasi di direktori app/Models) dan tambahkan kode berikut untuk mengatur fillable dan password yang akan di-hash menggunakan MD5:

php

use Illuminate\Contracts\Auth\MustVerifyEmail;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Foundation\Auth\User as Authenticatable;

use Illuminate\Notifications\Notifiable;

use Illuminate\Support\Facades\Hash;

class User extends Authenticatable

{

use HasFactory, Notifiable;

protected $fillable = [

'name', 'email', 'password',

];

public function setPasswordAttribute($value)

{

$this->attributes['password'] = Hash::make(md5($value));

}

}

1. Buat controller API untuk penanganan registrasi, login, dan logout dengan menjalankan perintah php artisan make:controller API/AuthController.
2. Buka file controller yang baru saja dibuat (berlokasi di direktori app/Http/Controllers/API) dan tambahkan metode berikut:

php

use App\Models\User;

use Illuminate\Support\Facades\Auth;

use Illuminate\Support\Facades\Validator;

use Illuminate\Http\Request;

class AuthController extends Controller

{

public function register(Request $request)

{

$validator = Validator::make($request->all(), [

'name' => 'required',

'email' => 'required|email|unique:users',

'password' => 'required|min:6',

]);

if ($validator->fails()) {

return response()->json(['error' => $validator->errors()], 400);

}

$user = User::create($request->all());

return response()->json(['message' => 'Registration successful'], 200);

}

public function login(Request $request)

{

$credentials = $request->only('email', 'password');

if (Auth::attempt($credentials)) {

$user = Auth::user();

$token = $user->createToken('authToken')->accessToken;

return response()->json(['token' => $token], 200);

} else {

return response()->json(['error' => 'Invalid credentials'], 401);

}

}

public function logout(Request $request)

{

$request->user()->token()->revoke();

return response()->json(['message' => 'Logout successful'], 200);

}

}

1. Tambahkan rute API yang sesuai dengan file api.php (berlokasi di direktori routes/api.php):

php

use App\Http\Controllers\API\AuthController;

Route::post('/register', [AuthController::class, 'register']);

Route::post('/login', [AuthController::class, 'login']);

Route::post('/logout', [AuthController::class, 'logout'])->middleware('auth:api');

Bagian React JS (Frontend):

1. Pastikan Anda telah menginstal Node.js di sistem Anda.
2. Buat proyek React JS baru dengan menjalankan perintah npx create-react-app nama-proyek.
3. Masuk ke direktori proyek dengan menjalankan perintah cd nama-proyek.
4. Buka file src/App.js dan ubah kode di dalamnya menjadi sebagai berikut:

jsx

import React, { useState } from 'react';

function App() {

const [name, setName] = useState('');

const [email, setEmail] = useState('');

const [password, setPassword] = useState('');

const handleRegister = () => {

// Lakukan permintaan POST ke endpoint /register pada backend Laravel

fetch('http://localhost/api/register', {

method: 'POST',

headers: {

'Content-Type': 'application/json',

},

body: JSON.stringify({ name, email, password }),

})

.then(response => response.json())

.then(data => {

console.log(data);

// Tambahkan logika atau tindakan lain yang diinginkan setelah registrasi berhasil

})

.catch(error => {

console.error('Error:', error);

// Tambahkan logika atau tindakan lain yang diinginkan dalam kasus kesalahan

});

};

const handleLogin = () => {

// Lakukan permintaan POST ke endpoint /login pada backend Laravel

fetch('http://localhost/api/login', {

method: 'POST',

headers: {

'Content-Type': 'application/json',

},

body: JSON.stringify({ email, password }),

})

.then(response => response.json())

.then(data => {

console.log(data);

// Tambahkan logika atau tindakan lain yang diinginkan setelah login berhasil

})

.catch(error => {

console.error('Error:', error);

// Tambahkan logika atau tindakan lain yang diinginkan dalam kasus kesalahan

});

};

const handleLogout = () => {

// Lakukan permintaan POST ke endpoint /logout pada backend Laravel

fetch('http://localhost/api/logout', {

method: 'POST',

headers: {

'Content-Type': 'application/json',

'Authorization': `Bearer ${token}`,

},

})

.then(response => response.json())

.then(data => {

console.log(data);

// Tambahkan logika atau tindakan lain yang diinginkan setelah logout berhasil

})

.catch(error => {

console.error('Error:', error);

// Tambahkan logika atau tindakan lain yang diinginkan dalam kasus kesalahan

});

};

return (

<div>

<h1>Registration</h1>

<input

type="text"

placeholder="Name"

value={name}

onChange={e => setName(e.target.value)}

/>

<input

type="email"

placeholder="Email"

value={email}

onChange={e => setEmail(e.target.value)}

/>

<input

type="password"

placeholder="Password"

value={password}

onChange={e => setPassword(e.target.value)}

/>

<button onClick={handleRegister}>Register</button>

<h1>Login</h1>

<input

type="email"

placeholder="Email"

value={email}

onChange={e => setEmail(e.target.value)}

/>

<input

type="password"

placeholder="Password"

value={password}

onChange={e => setPassword(e.target.value)}

/>

<button onClick={handleLogin}>Login</button>

<h1>Logout</h1>

<button onClick={handleLogout}>Logout</button>

</div>

);

}

export default App;

Pastikan Anda mengganti URL endpoint API (misalnya http://localhost/api/register) dengan URL yang sesuai untuk backend Laravel Anda.

1. Jalankan aplikasi React JS dengan perintah npm start.

Sekarang Anda telah membuat API Register, Login, dan Logout dengan MD5 pada Laravel dan React JS. Pastikan untuk menyesuaikan kode dengan kebutuhan Anda dan melakukan penyesuaian keamanan tambahan seperti penggunaan HTTPS, token refresh, dan lainnya sesuai kebutuhan Anda.

Coba:

//php artisan make:model User -m

//table create user

public function up()

{

    Schema::create('users', function (Blueprint table) {

        table->id();

        table->string('name');

        table->string('email')->unique();

        table->string('password');

        table->string('token')->nullable();

        table->timestamps();

    });

// User.php

<?php

namespace App\Models;

use Illuminate\Contracts\Auth\MustVerifyEmail;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Foundation\Auth\User as Authenticatable;

use Illuminate\Notifications\Notifiable;

use Illuminate\Support\Facades\Hash;

use Illuminate\Support\Str;

class User extends Authenticatable

{

    use HasFactory, Notifiable;

    protected fillable = [

        'name',

        'email',

        'password',

    ];

    protected hidden = [

        'password',

        'remember\_token',

    ];

    public function setPasswordAttribute(value)

    {

        this->attributes['password'] = Hash::make(value);

    }

    public function generateToken()

    {

        this->token = Str::random(60);

        this->save();

        return this->token;

    }

}

//php artisan make:controller AuthController

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Auth;

class AuthController extends Controller

{

    public function register(Request request)

    {

        request->validate([

            'name' => 'required',

            'email' => 'required|email|unique:users,email',

            'password' => 'required|min:6',

        ]);

        user = User::create([

            'name' => request->name,

            'email' => request->email,

            'password' => request->password,

        ]);

        token = user->generateToken();

        return response()->json([

            'message' => 'User registered successfully',

            'token' => token,

        ]);

    }

    public function login(Request request)

    {

        request->validate([

            'email' => 'required|email',

            'password' => 'required',

        ]);

        if (Auth::attempt(['email' => request->email, 'password' => request->password])) {

            user = Auth::user();

            token = user->generateToken();

            return response()->json([

                'message' => 'User logged in successfully',

                'token' => token,

            ]);

        }

        return response()->json(['message' => 'Invalid credentials'], 401);

    }

    public function logout(Request request)

    {

        user = request->user();

        user->token = null;

        user->save();

        return response()->json(['message' => 'User logged out successfully']);

    }

}

//routes/api.php

use App\Http\Controllers\AuthController;

Route::post('register', [AuthController::class, 'register']);

Route::post('login', [AuthController::class, 'login'])->name('login');

Route::post('logout', [AuthController::class, 'logout'])->middleware('auth:api');

//REACT-------

//npm install axios react-router-dom

//Login.js

import React, { useState } from 'react';

import axios from 'axios';

const Login = () => {

  const [email, setEmail] = useState('');

  const [password, setPassword] = useState('');

  const handleSubmit = async (e) => {

    e.preventDefault();

    try {

      const response = await axios.post('http://localhost:8000/api/login', {

        email,

        password,

      });

      console.log(response.data); // Menampilkan token ke console

    } catch (error) {

      console.error(error);

    }

  };

  return (

    <form onSubmit={handleSubmit}>

      <input

        type="email"

        placeholder="Email"

        value={email}

        onChange={(e) => setEmail(e.target.value)}

      />

      <input

        type="password"

        placeholder="Password"

        value={password}

        onChange={(e) => setPassword(e.target.value)}

      />

      <button type="submit">Login</button>

    </form>

  );

};

export default Login;

//Register.js

import React, { useState } from 'react';

import axios from 'axios';

const Register = () => {

  const [name, setName] = useState('');

  const [email, setEmail] = useState('');

  const [password, setPassword] = useState('');

  const handleSubmit = async (e) => {

    e.preventDefault();

    try {

      const response = await axios.post('http://localhost:8000/api/register', {

        name,

        email,

        password,

      });

      console.log(response.data); // Menampilkan token ke console

    } catch (error) {

      console.error(error);

    }

  };

  return (

    <form onSubmit={handleSubmit}>

      <input

        type="text"

        placeholder="Name"

        value={name}

        onChange={(e) => setName(e.target.value)}

      />

      <input

        type="email"

        placeholder="Email"

        value={email}

        onChange={(e) => setEmail(e.target.value)}

      />

      <input

        type="password"

        placeholder="Password"

        value={password}

        onChange={(e) => setPassword(e.target.value)}

      />

      <button type="submit">Register</button>

    </form>

  );

};

export default Register;

//App.js

import React from 'react';

import { BrowserRouter as Router, Route, Link } from 'react-router-dom';

import Login from './components/Login';

import Register from './components/Register';

const App = () => {

  return (

    <Router>

      <div>

        <nav>

          <ul>

            <li>

              <Link to="/login">Login</Link>

            </li>

            <li>

              <Link to="/register">Register</Link>

            </li>

          </ul>

        </nav>

        <Route path="/login" component={Login} />

        <Route path="/register" component={Register} />

      </div>

    </Router>

  );

};

export default App;

}