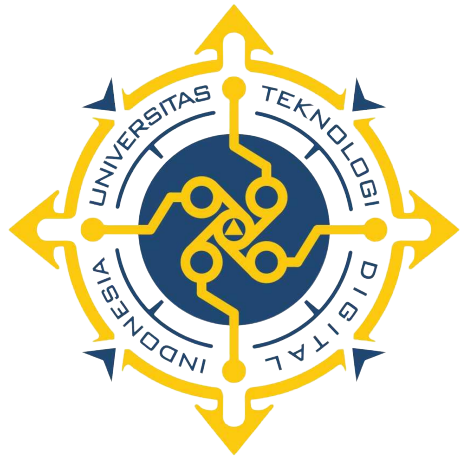
Laporan Proyek Pengembangan Aplikasi Web

Absensi Siswa



Penyusun :

1. 225410095 / Muhammad Khoirul Anam Ihsan
2. 225410089 / Arnoldus Paulus Sewa
3. 225410081 / Alfian Desna Saputra

Program Studi Informatika Klas 1

Fakultas Teknologi Informasi

Universitas Teknologi Digital Indonesia (UTDI)

2024

#### 

#### 

#### **1. Pendahuluan**

1. **Latar Belakang:**

Dengan kemajuan teknologi informasi dan komunikasi yang pesat di berbagai aspek kehidupan, dunia pendidikan turut mengadopsi inovasi ini untuk meningkatkan efisiensi pengelolaan administrasi. Salah satunya adalah melalui pengembangan aplikasi absensi berbasis web. Aplikasi ini dirancang untuk mengurangi beban pekerjaan manual, menghemat waktu, dan memastikan data absensi tersusun dengan rapi. Sistem tersebut mencakup manajemen pengguna, termasuk administrasi sekolah, guru, dan pihak terkait lainnya. Dengan fleksibilitas dan kemudahan penggunaannya, aplikasi ini diharapkan mampu menciptakan sistem absensi yang lebih efektif dan efisien.

1. **Tujuan:**

Proyek pengembangan aplikasi web absensi siswa bertujuan untuk mempermudah proses pencatatan kehadiran siswa secara otomatis, sehingga lebih cepat, akurat, dan efisien dibandingkan metode manual. Aplikasi ini dirancang untuk memungkinkan guru, wali kelas, atau pihak sekolah memantau kehadiran siswa secara real-time melalui platform berbasis web.

1. **Batasan Masalah:**

* Lingkup Pengguna
* Aplikasi hanya ditujukan untuk digunakan oleh pihak sekolah, termasuk guru, dan admin
* Fitur-Fitur yang Akan Dikembangkan
* Setiap pengguna akan memiliki akun yang dilindungi oleh sistem login dan kata sandi.
* Pihak sekolah dapat melihat laporan kehadiran siswa secara real-time, termasuk statistik kehadiran harian, mingguan, atau bulanan.
* ketika login guru hanya dapat melihat jadwal mengajar yang sudah di tetapkan oleh admin sekolah.
* guru dapat mengabsensi siswa yang hadir ataupun siswa yang tidak hadir dengan alasan tertentu
* admin dapat melakukan penjadwalan guru yang akan ditetapkan.
* admin dapat menambahkan, mengubah, dan menghapus data pelajaran.
* admin dapat menambahkan, mengubah, dan menghapus data kelas.
* admin dapat menambahkan, mengubah, dan menghapus data siswa.
* admin dapat menambahkan, mengubah, dan menghapus data pengguna.
* Fitur-Fitur yang Tidak Akan Dikembangkan
* Penilaian akademik siswa tidak menjadi bagian dari aplikasi ini. Fokus aplikasi hanya pada absensi.
* Aplikasi ini sepenuhnya berbasis web dan memerlukan koneksi internet untuk semua fungsinya.
* Teknologi biometrik seperti deteksi wajah tidak akan diterapkan dalam versi awal karena keterbatasan waktu dan sumber daya.
* Aplikasi ini tidak akan mencakup fitur pengelolaan pembayaran atau sistem sumbangan siswa.
* Orang tua siswa dapat menerima laporan kehadiran melalui notifikasi atau email (tidak memiliki akun langsung dalam aplikasi).
* Orang tua siswa dapat menerima laporan kehadiran melalui notifikasi atau email (tidak memiliki akun langsung dalam aplikasi).
* Platform Pengembangan
* Aplikasi hanya akan dikembangkan untuk platform berbasis web (browser) dan tidak mencakup pengembangan aplikasi mobile native (Android atau iOS).

#### **2. Perancangan Sistem**

1. **Rancangan Semula/Awal**
2. **Rancangan Database :**

Basis data adalah kumpulan file-file yang membentuk data yang besar dengan dikumpulkannya data kedalam *database*. Dalam aplikasi ini menggunakan *database* dengan nama *absensi* dan memerlukan tabel-tabel berikut:

**Struktur Tabel *siswa***

Tabel *siswa* berfungsi untuk menyimpan data *siswa* yang telah melakukan registrasi dan nantinya digunakan untuk proses *login* dengan memasukkan *username* dan *password*. Struktur tabel *user* dapat dilihat pada dibawah ini :

Nama tabel : siswa

Kunci primer : id\_siswa

Kunci tamu : -

Jumlah kolom : 4

| **No** | **Nama Kolom** | **Tipe** | **Keterangan** |
| --- | --- | --- | --- |
| 1 | id\_siswa | int | Auto Increment |
| 2 | nama\_siswa | varchar | Nama siswa |
| 3 | nis | varchar | Nomor induk siswa |
| 4 | kelas | varchar | Kelas siswa |

**Struktur Tabel *absensi***

Tabel *absensi* berfungsi untuk menyimpan data *absensi* yang telah melakukan registrasi dan nantinya digunakan untuk proses *login* dengan memasukkan *username* dan *password*. Struktur tabel *user* dapat dilihat pada dibawah ini :

Nama tabel : absensi

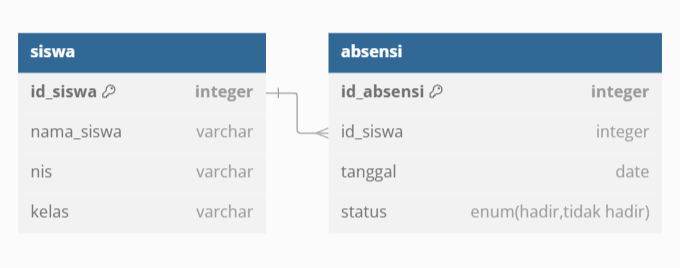
Kunci primer : id\_absensi

Kunci tamu : id\_siswa

Jumlah kolom : 4

| **No** | **Nama Kolom** | **Tipe** | **Keterangan** |
| --- | --- | --- | --- |
| 1 | id\_absensi | int | Auto Increment |
| 2 | id\_siswa | int | Foreign key referensi tabel siswa |
| 3 | tanggal | date | Tanggal untuk absensi |
| 4 | status | enum(hadir, tidak hadir) | Status kehadiran siswa |

**Relasi Tabel**

****

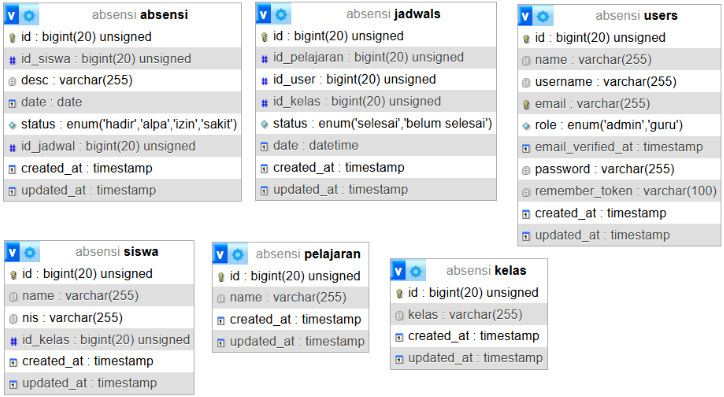
1. **Data flow diagram (DFD) level 0** :



Data Flow Diagram (DFD) Level 0 menggambarkan aliran data dalam aplikasi absensi siswa berbasis web, di mana Admin memasukkan data seperti Data Pelajaran, Data Kelas, Data Siswa, Data Pengguna, dan Data Penjadwalan Guru ke dalam sistem, serta menerima output berupa Laporan Absensi Siswa, sedangkan Guru memasukkan Data Absensi Siswa dan Data Penjadwalan Guru ke dalam sistem, yang kemudian diproses untuk menghasilkan informasi yang dibutuhkan kedua entitas tersebut.

1. **Realisasi**

**Rancangan Database**



**Rancangan Data flow diagram (DFD) Level 1 :**

#### 

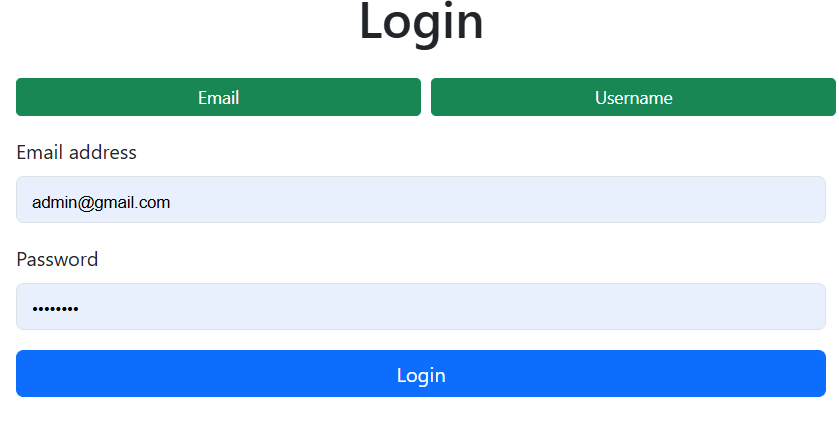
#### **3. Teknologi / Framework Laravel**

Laravel dipilih sebagai framework untuk pengembangan aplikasi web absensi siswa karena memiliki sejumlah keunggulan yang sangat mendukung kebutuhan proyek ini. Pertama, Laravel menggunakan konsep Model-View-Controller (MVC), yang memungkinkan aplikasi untuk lebih terstruktur, mudah dipahami, dan mudah dikembangkan. Dengan memisahkan logika bisnis, tampilan, dan pengelolaan data, pengembangan aplikasi absensi siswa menjadi lebih efisien dan terorganisir. Keamanan menjadi salah satu faktor utama dalam aplikasi ini, dan Laravel menyediakan fitur keamanan built-in yang kuat, seperti proteksi CSRF, hashing password, dan enkripsi data. Fitur ini memastikan bahwa data kehadiran siswa dan informasi pribadi terlindungi dengan aman dari potensi kebocoran atau penyalahgunaan.

#### **4. Implementasi**

**Implementasi *User* Admin**

**Halaman Login Admin**



public function showLoginForm(){

return view('auth.login');

}

public function login(Request $request){

$request->validate([

'username' => 'nullable',

'email' => 'nullable|email',

'password' => 'required',

]);

if ($request->username) {

$attempt = $request->only('username', 'password');

} else {

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

}

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

$request->session()->regenerate();

return redirect()->intended('/');

}

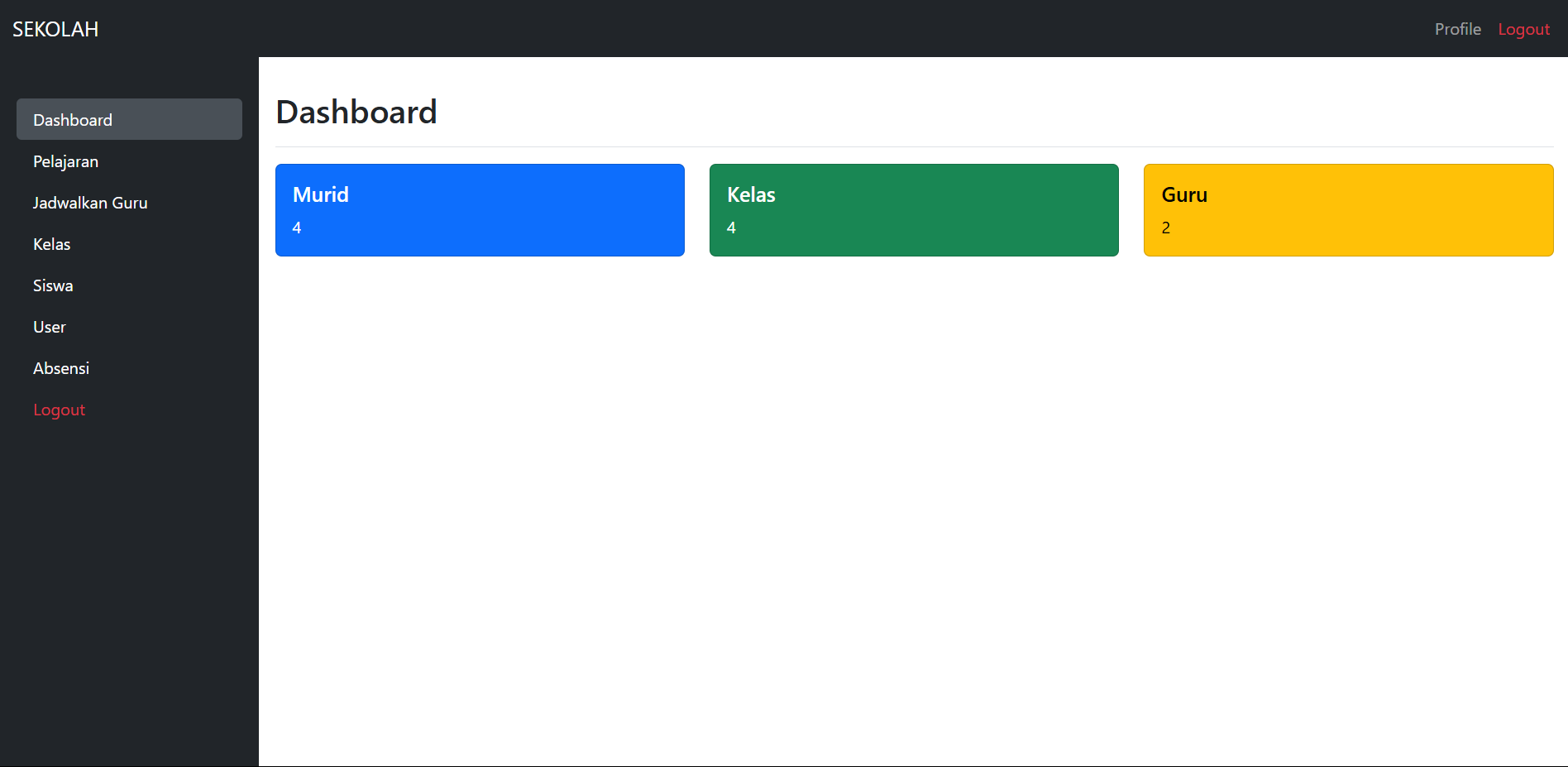
return back()->withErrors([

'email' => 'The provided credentials do not match our records.',

]);

}

**Halaman Dashboard**



public function index() {

$data['siswa'] = Siswa::count();

$data['kelas'] = Kelas::count();

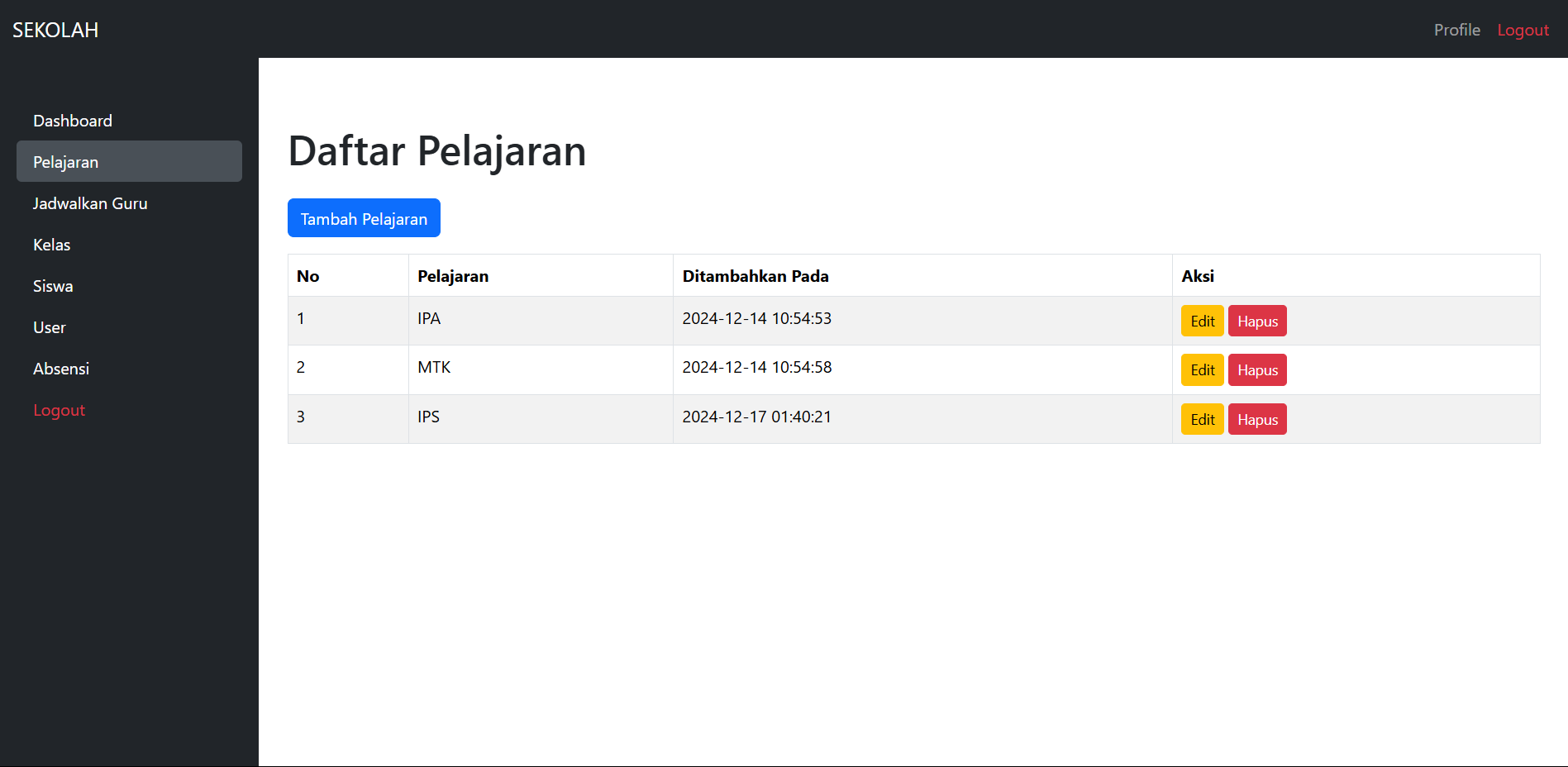
$data['guru'] = User::where('role', 'guru')->count();

$data['jadwal'] = Jadwal::where('id\_user', Auth::user()->id)->count();

return view('dashboard', $data);

}

**Halaman Daftar Pelajaran**

****

**public $title = 'Pelajaran';**

**public $view = 'admin.pelajaran.';**

**public $route = 'admin.pelajaran.';**

**public function index(){**

**$data['title'] = $this->title;**

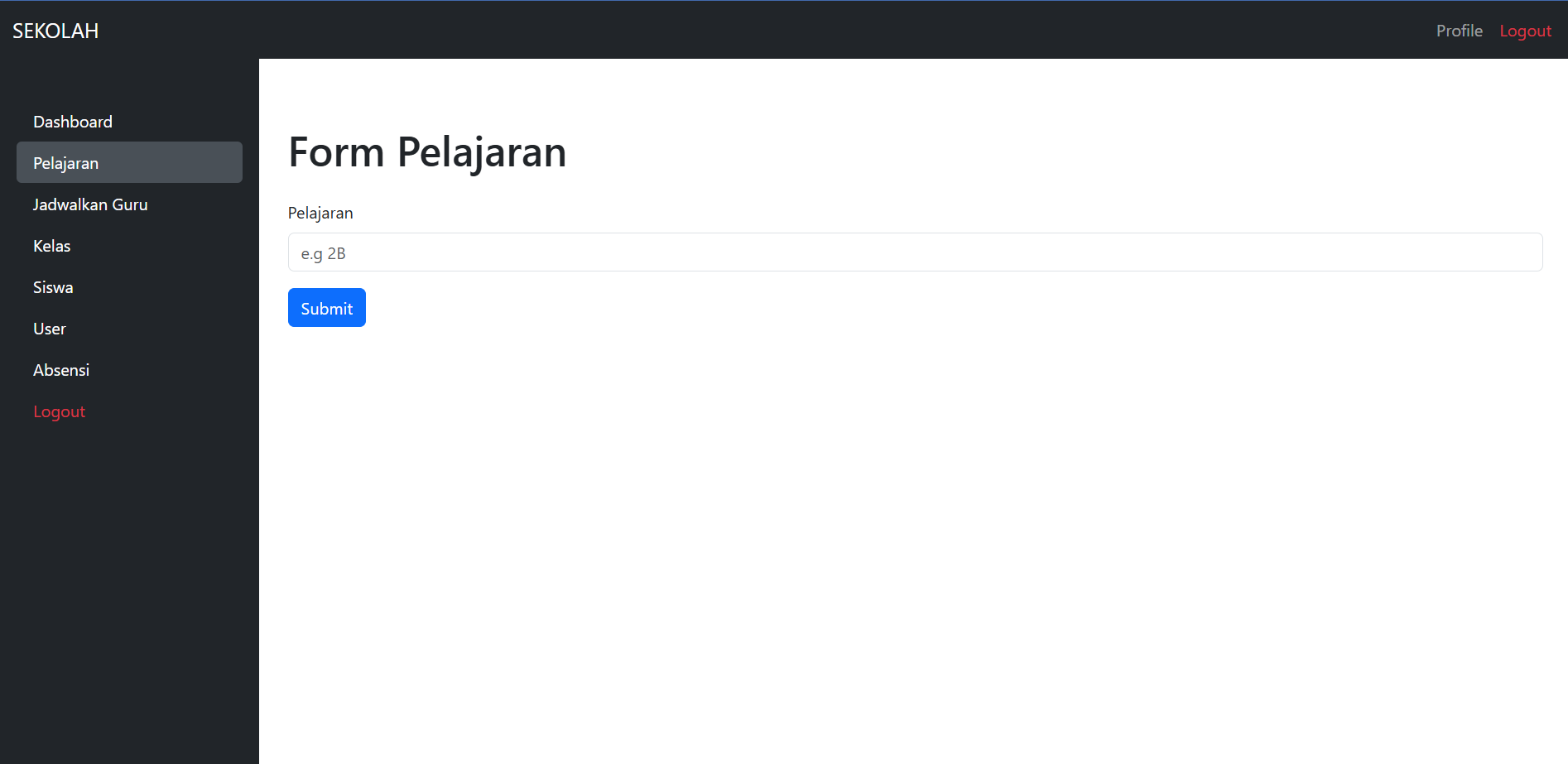
**$data['pelajaran'] = Pelajaran::get();**

**$data['route'] = route($this->route.'create');**

**return view($this->view.'index', $data);**

**}**

**Halaman Tambah Daftar Pelajaran**

****

**public function create(){**

**$data['title'] = $this->title;**

**$data['route'] = route($this->route.'store');**

**return view($this->view.'.form', $data);**

**}**

**public function store(Request $request){**

**$data['title'] = $this->title;**

**$validate = $request->validate([**

**'name' => 'required',**

**]);**

**Pelajaran::create($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Edit Daftar Pelajaran**

**public function edit(string $id){**

**$data['title'] = $this->title;**

**$data['model'] = Pelajaran::find($id);**

**$data['route'] = route($this->route.'update', $id);**

**return view($this->view.'.form', $data);**

**}**

**public function update(Request $request, string $id){**

**$data['title'] = $this->title;**

**$pelajaran = Pelajaran::find($id);**

**$validate = $request->validate([**

**'name' => 'required',**

**]);**

**$pelajaran->update($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Hapus Daftar Pelajaran**

**public function destroy(string $id){**

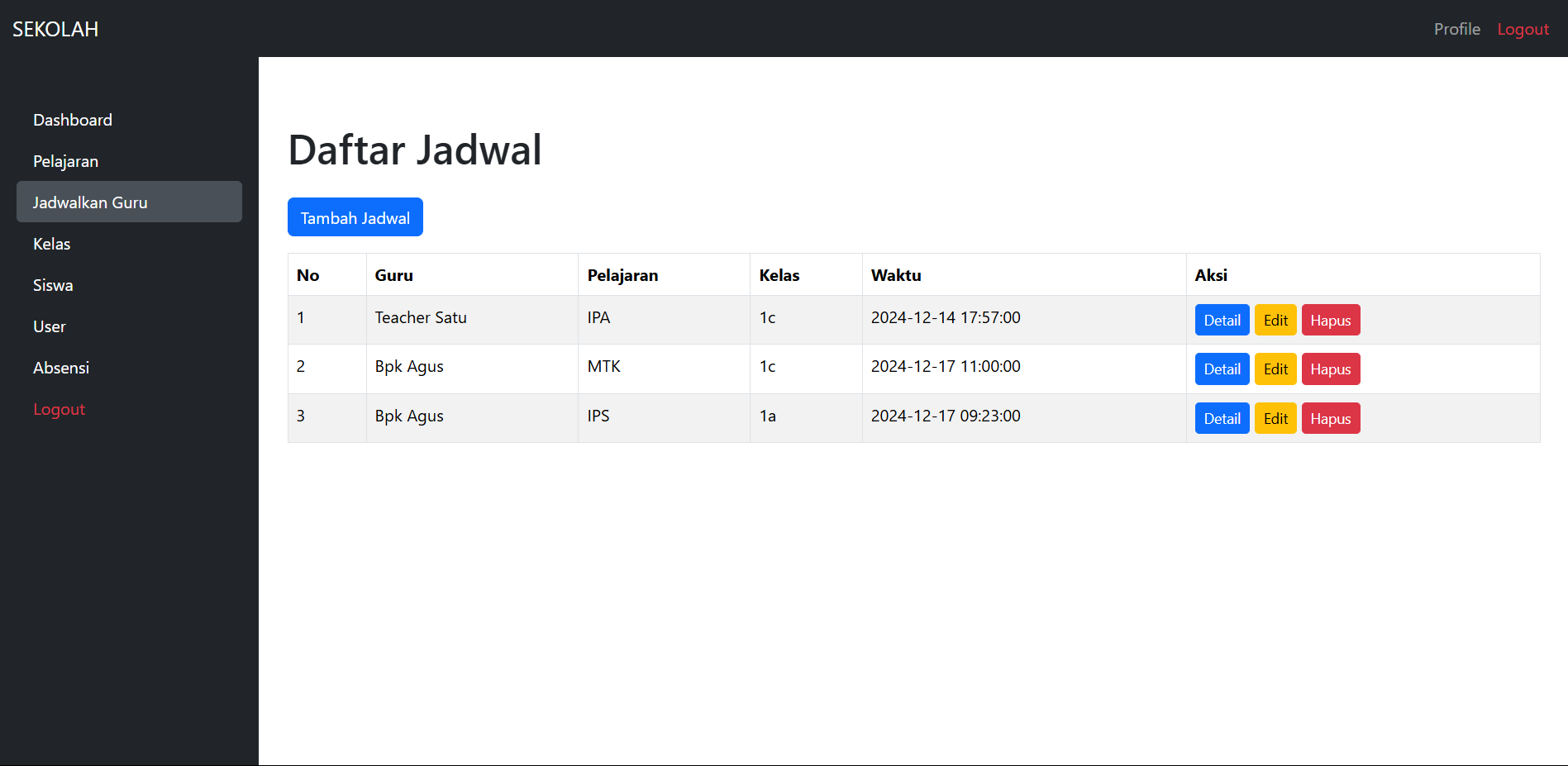
**$pelajaran = Pelajaran::find($id);**

**$pelajaran->delete();**

**return redirect()->back();**

**}**

**Halaman Jadwal Guru**

****

**public $title = 'Jadwal';**

**public $view = 'admin.jadwal.';**

**public $route = 'admin.jadwal.';**

**public function index(){**

**$data['title'] = $this->title;**

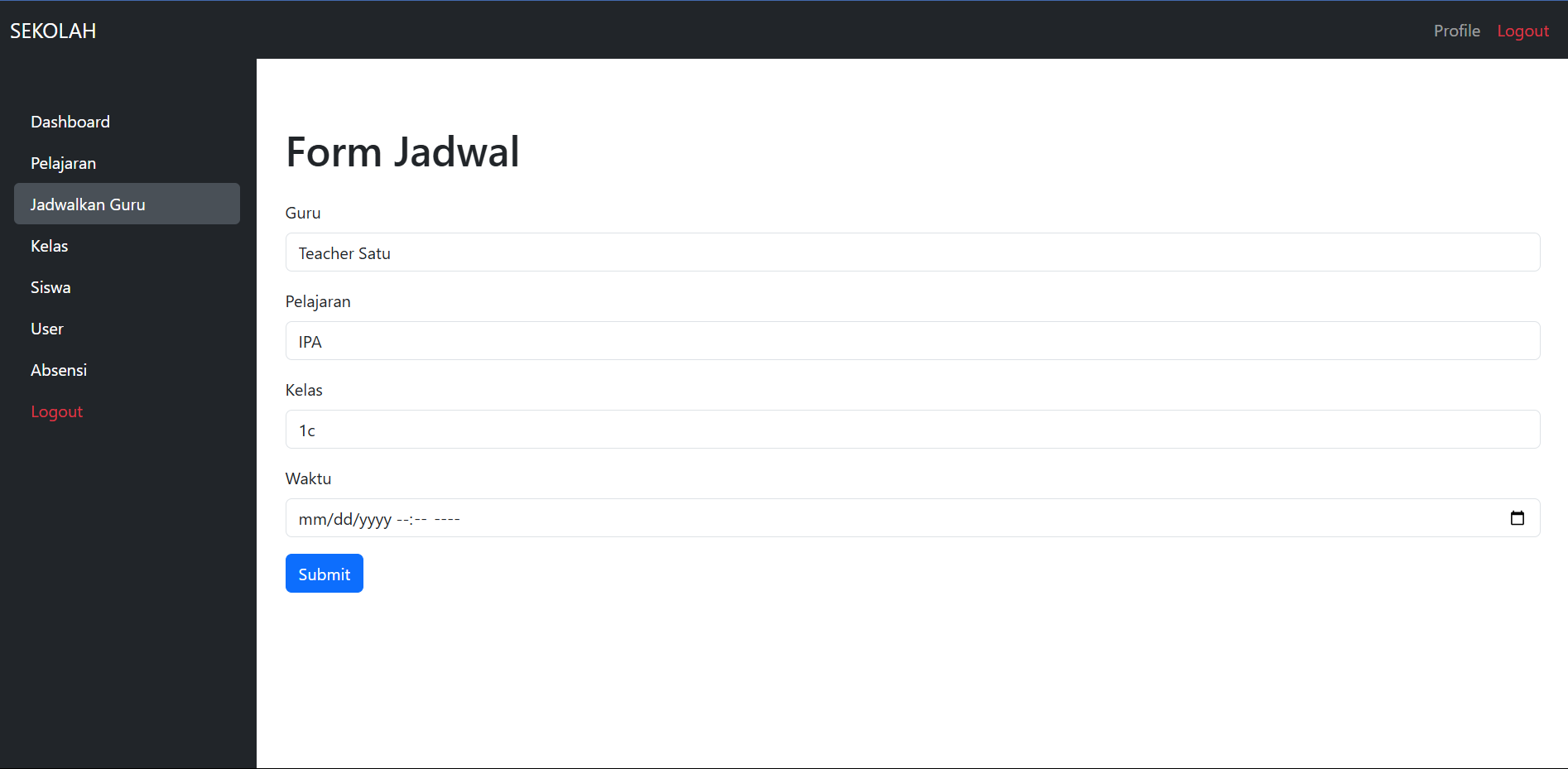
**$data['jadwal'] = Jadwal::get();**

**$data['route'] = route($this->route.'create');**

**return view($this->view.'index', $data);**

**}**

**Halaman Tambah Jadwal Guru**

****

**public function create(){**

**$data['title'] = $this->title;**

**$data['route'] = route($this->route.'store');**

**$data['pelajaran'] = Pelajaran::get();**

**$data['guru'] = User::where('role', 'guru')->get();**

**$data['kelas'] = Kelas::get();**

**return view($this->view.'.form', $data);**

**}**

**public function store(Request $request){**

**$data['title'] = $this->title;**

**$validate = $request->validate([**

**'id\_pelajaran' => 'required',**

**'id\_user' => 'required',**

**'id\_kelas' => 'required',**

**'date' => 'required',]);**

**Jadwal::create($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Edit Jadwal Guru**

**public function edit(string $id){**

**$data['title'] = $this->title;**

**$data['model'] = Jadwal::find($id);**

**$data['pelajaran'] = Pelajaran::get();**

**$data['guru'] = User::where('role', 'guru')->get();**

**$data['route'] = route($this->route.'update', $id);**

**$data['kelas'] = Kelas::get();**

**return view($this->view.'.form', $data);**

**}**

**public function update(Request $request, string $id){**

**$data['title'] = $this->title;**

**$jadwal = Jadwal::find($id);**

**$validate = $request->validate([**

**'id\_pelajaran' => 'required',**

**'id\_kelas' => 'required',**

**'date' => 'required',**

**]);**

**$jadwal->update($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Hapus Jadwal Guru**

**public function destroy(string $id){**

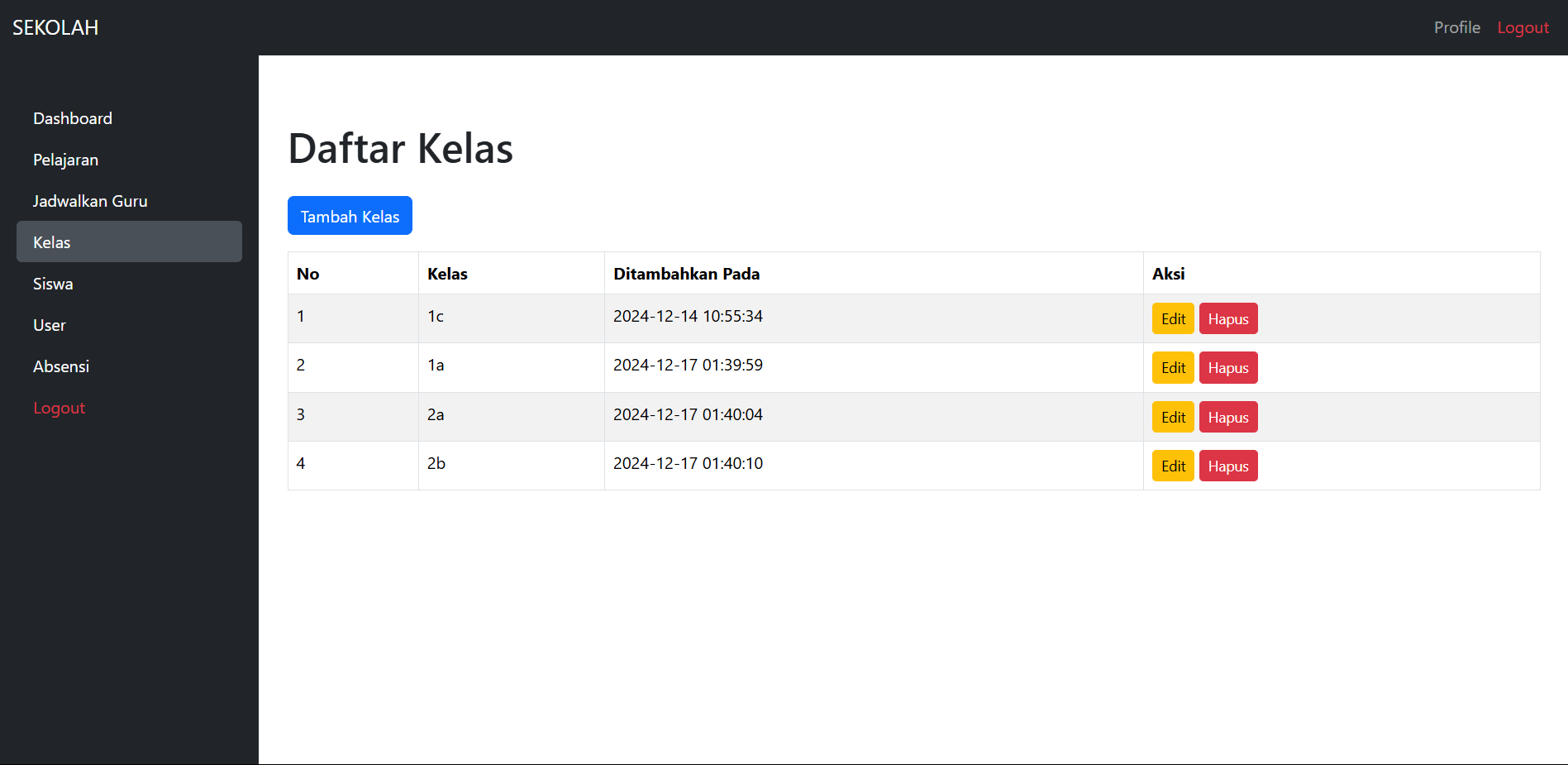
**$jadwal = Jadwal::find($id);**

**$jadwal->delete();**

**return redirect()->back();**

**}**

**Halaman Kelas**

****

**public $title = 'Kelas';**

**public $view = 'admin.kelas.';**

**public $route = 'admin.kelas.';**

**public function index(){**

**$data['title'] = $this->title;**

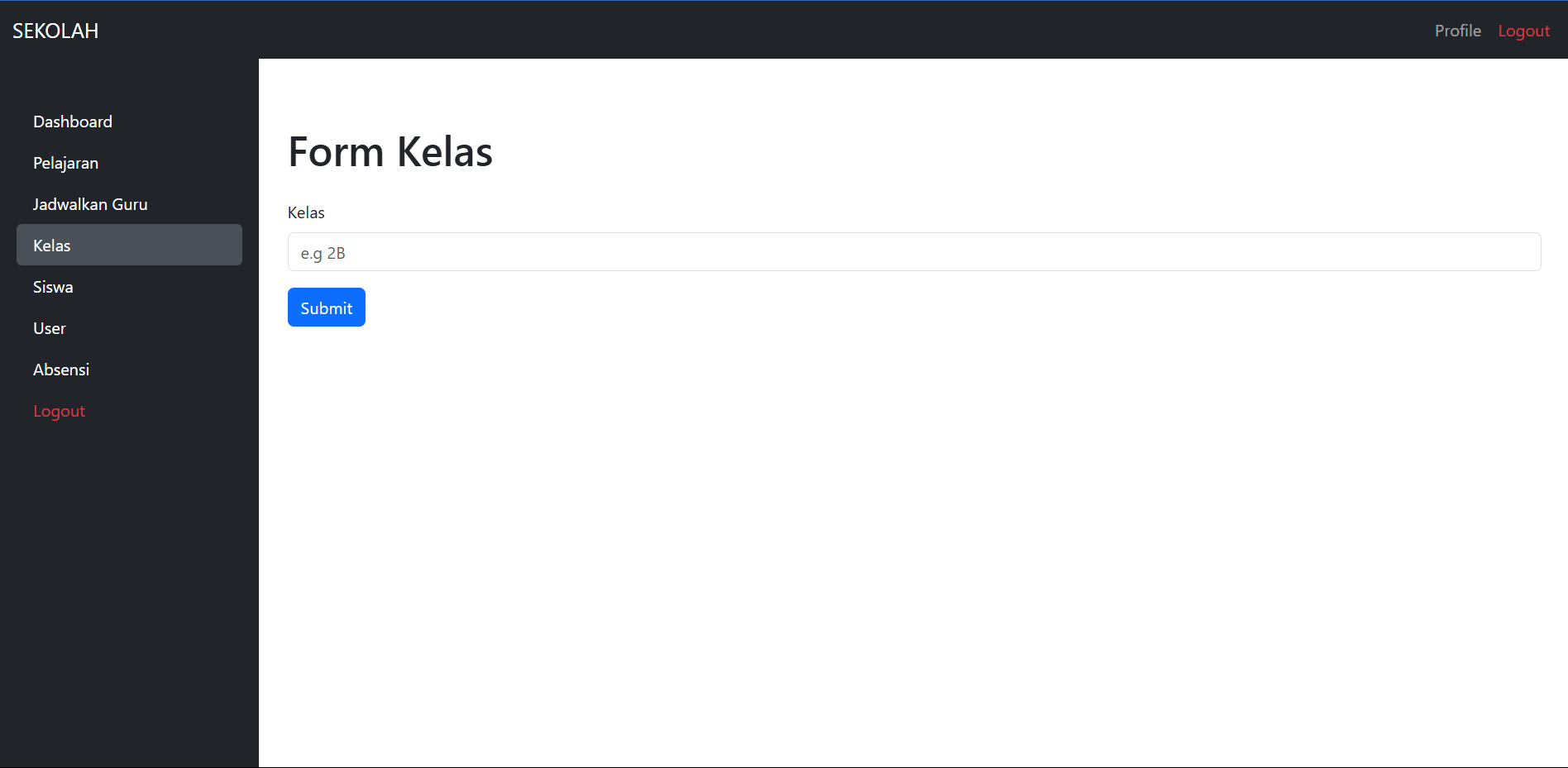
**$data['kelas'] = Kelas::get();**

**$data['route'] = route($this->route.'create');**

**return view($this->view.'index', $data);**

**}**

**Halaman Tambah Kelas**

****

**public function create(){**

**$data['title'] = $this->title;**

**$data['route'] = route($this->route.'store');**

**return view($this->view.'.form', $data);**

**}**

**public function store(Request $request){**

**$data['title'] = $this->title;**

**$validate = $request->validate([**

**'kelas' => 'required',**

**]);**

**Kelas::create($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Edit Kelas**

**public function edit(string $id){**

**$data['title'] = $this->title;**

**$data['model'] = Kelas::find($id);**

**$data['route'] = route($this->route.'update', $id);**

**return view($this->view.'.form', $data);**

**}**

**public function update(Request $request, string $id){**

**$data['title'] = $this->title;**

**$kelas = Kelas::find($id);**

**$validate = $request->validate([**

**'kelas' => 'required',**

**]);**

**$kelas->update($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Hapus Kelas**

**public function destroy(string $id){**

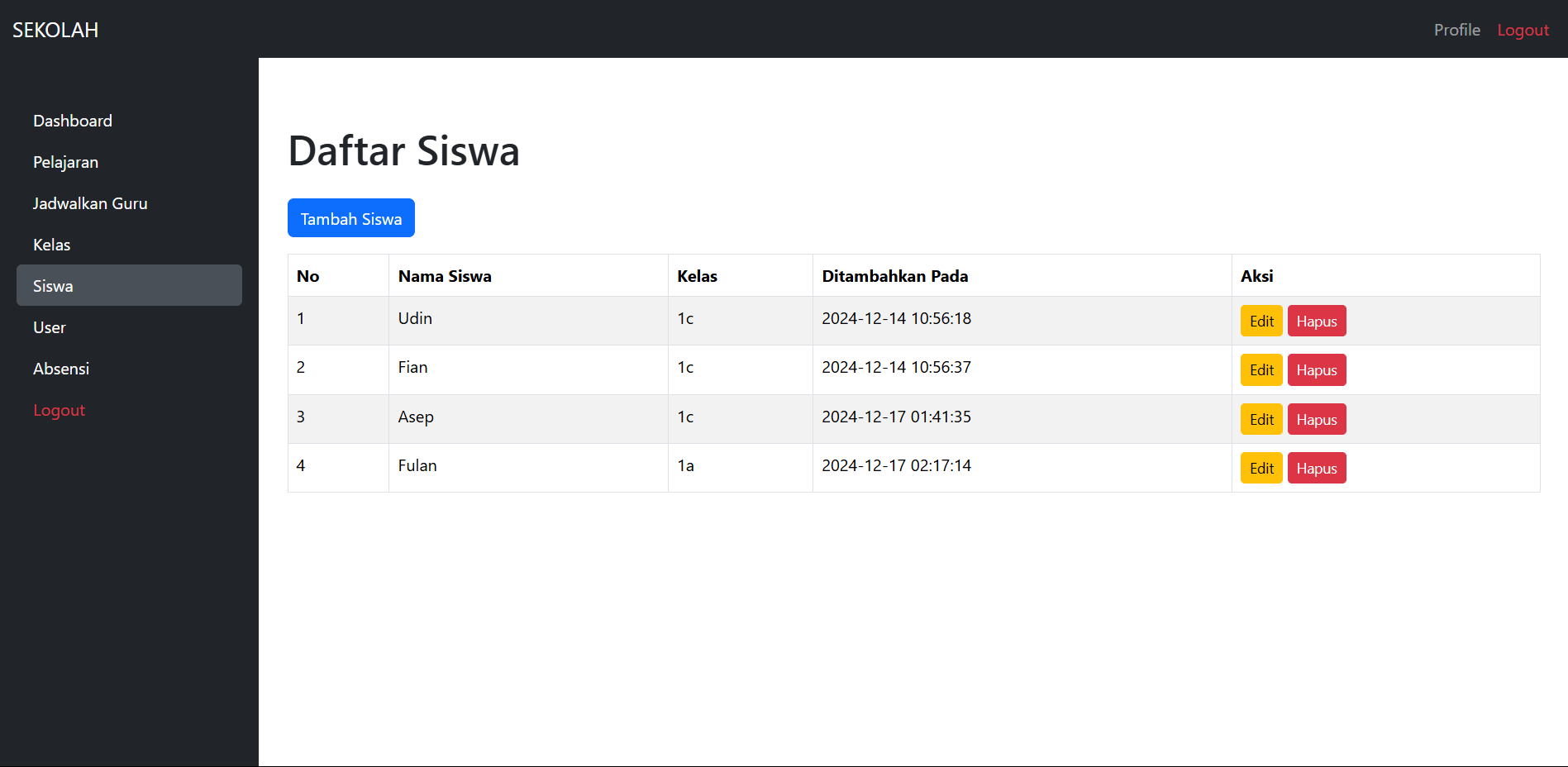
**$kelas = Kelas::find($id);**

**$kelas->delete();**

**return redirect()->back();**

**}**

**Halaman Siswa**

****

**public $title = 'Siswa';**

**public $view = 'admin.siswa.';**

**public $route = 'admin.siswa.';**

**public function index(){**

**$data['title'] = $this->title;**

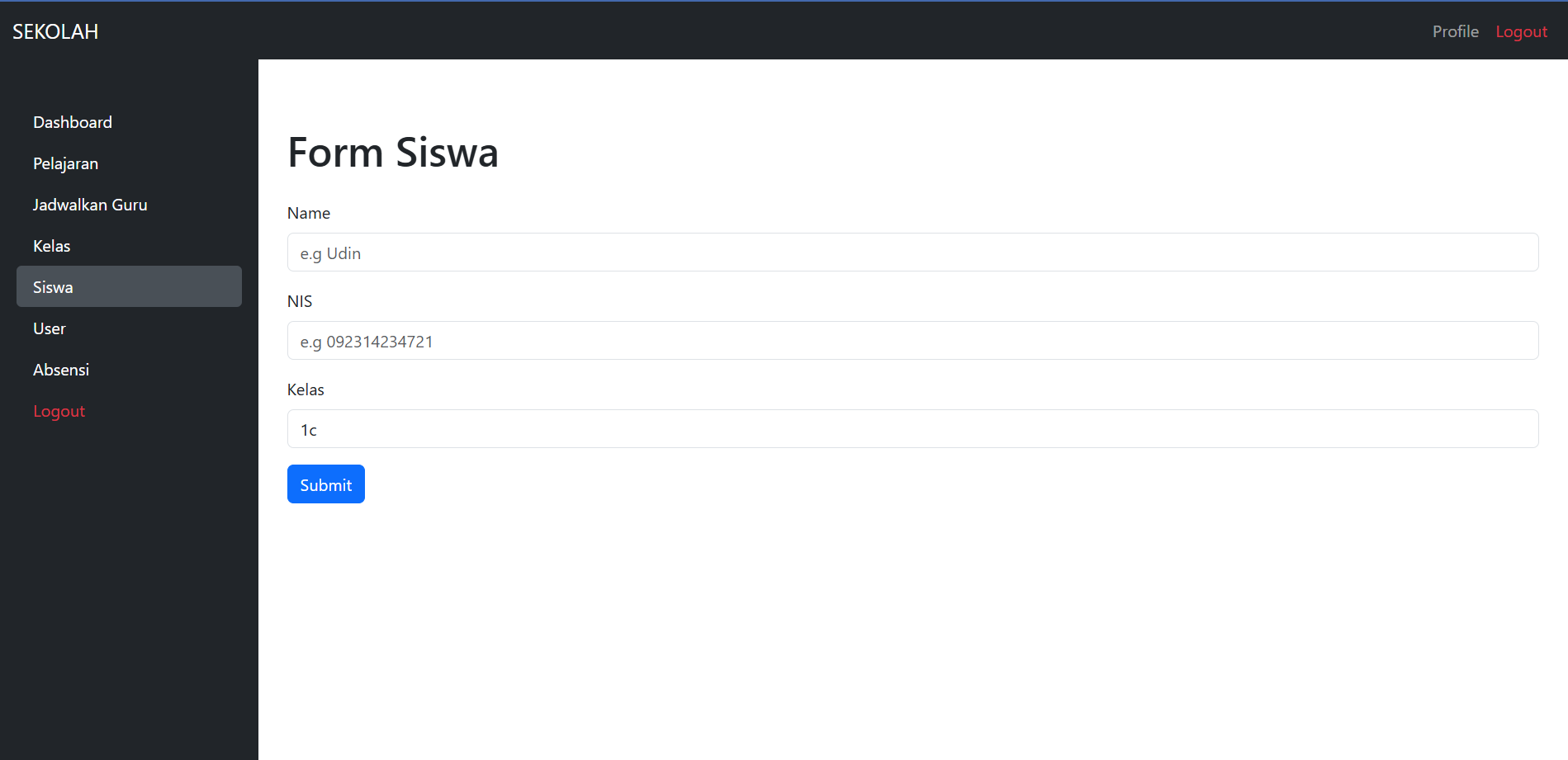
**$data['siswa'] = Siswa::get();**

**$data['route'] = route($this->route.'create');**

**return view($this->view.'index', $data);**

**}**

**Halaman Tambah Siswa**

****

**public function create(){**

**$data['title'] = $this->title;**

**$data['kelas'] = Kelas::get();**

**$data['route'] = route($this->route.'store');**

**return view($this->view.'.form', $data);**

**}**

**public function store(Request $request){**

**$data['title'] = $this->title;**

**$validate = $request->validate([**

**'name' => 'required',**

**'nis' => 'required|numeric',**

**'id\_kelas' => 'required|exists:kelas,id',**

**]);**

**Siswa::create($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Edit Siswa**

**public function edit(string $id){**

**$data['title'] = $this->title;**

**$data['kelas'] = Kelas::get();**

**$data['model'] = Siswa::find($id);**

**$data['route'] = route($this->route.'update', $id);**

**return view($this->view.'.form', $data);**

**}**

**public function update(Request $request, string $id){**

**$data['title'] = $this->title;**

**$siswa = Siswa::find($id);**

**$validate = $request->validate([**

**'name' => 'required',**

**'nis' => 'required|numeric',**

**'id\_kelas' => 'required|exists:kelas,id',**

**]);**

**$siswa->update($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Hapus Siswa**

**public function destroy(string $id) {**

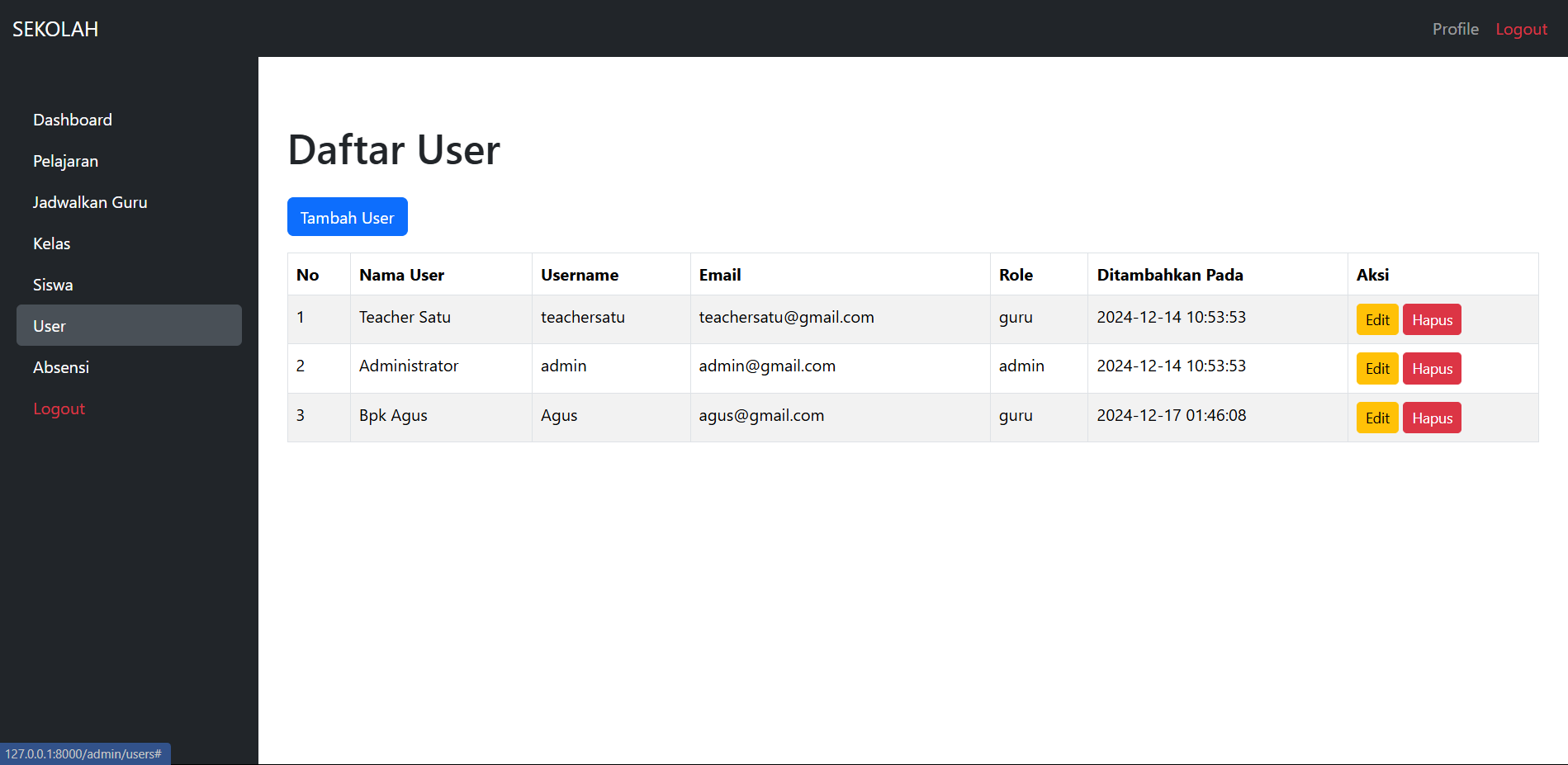
**$siswa = Siswa::find($id);**

**$siswa->delete();**

**return redirect()->back();**

**}**

**Halaman User**

****

**public $title = 'User';**

**public $view = 'admin.user.';**

**public $route = 'admin.users.';**

**public function index(){**

**$data['title'] = $this->title;**

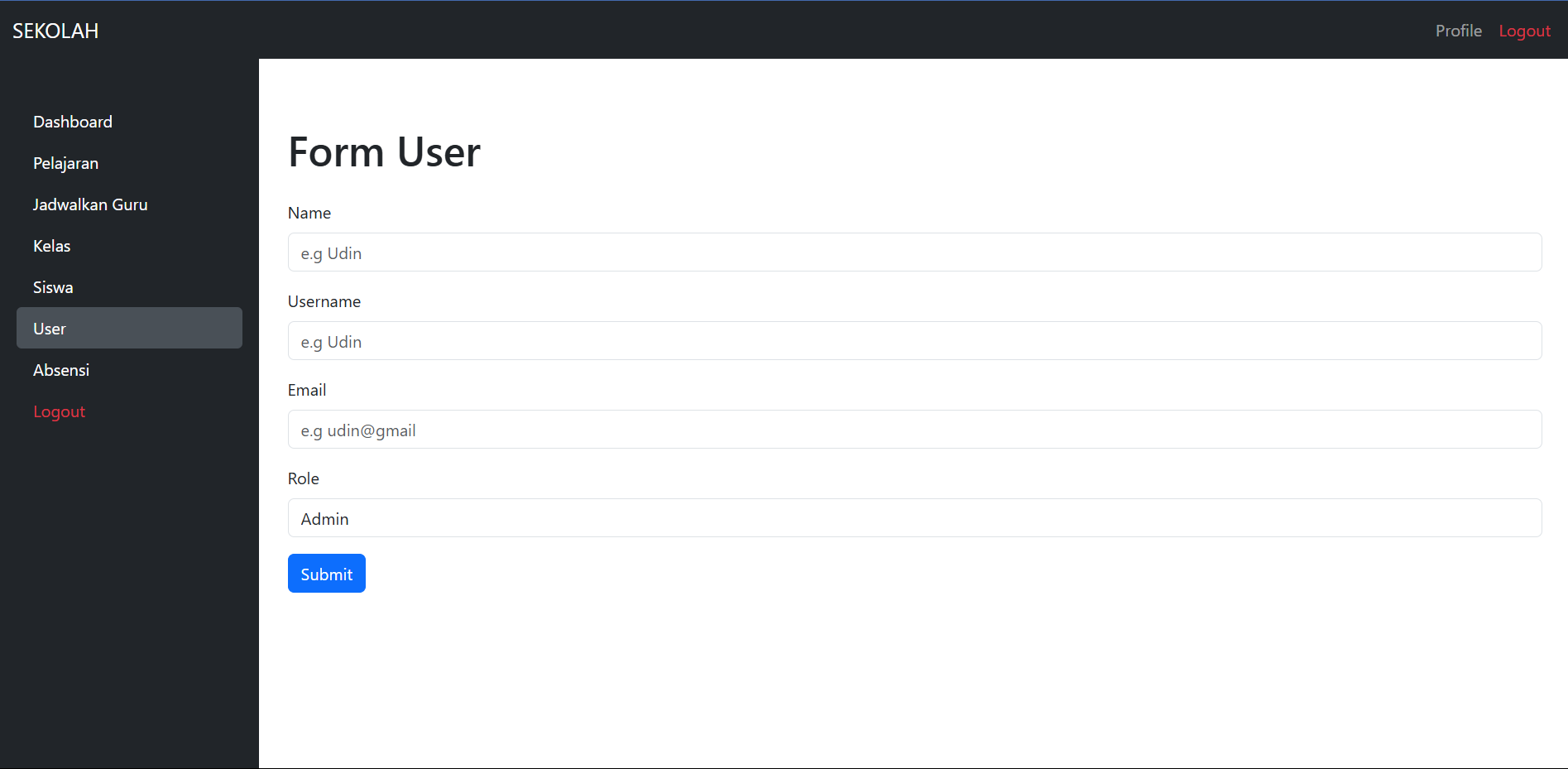
**$data['user'] = User::get();**

**$data['route'] = route($this->route.'create');**

**return view($this->view.'index', $data);**

**}**

**Halaman Tambah User**

****

**public function create(){**

**$data['title'] = $this->title;**

**$data['guru'] = User::where('role', 'guru')->get();**

**$data['route'] = route($this->route.'store');**

**return view($this->view.'.form', $data);**

**}**

**public function store(Request $request){**

**$data['title'] = $this->title;**

**$validate = $request->validate([**

**'name' => 'required',**

**'username' => 'required',**

**'role' => 'required|in:admin,guru',**

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

**]);**

**$validate['password'] = Hash::make('12345678');**

**User::create($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Edit User**

**public function edit(string $id){**

**$data['title'] = $this->title;**

**$data['guru'] = User::where('role', 'guru')->get();**

**$data['model'] = User::find($id);**

**$data['route'] = route($this->route.'update', $id);**

**return view($this->view.'.form', $data);**

**}**

**public function update(Request $request, string $id){**

**$data['title'] = $this->title;**

**$user = User::find($id);**

**$validate = $request->validate([**

**'name' => 'required',**

**'username' => 'required',**

**'role' => 'required|in:admin,guru',**

**'email' => 'required',**

**]);**

**$validate['password'] = Hash::make('12345678');**

**$user->update($validate);**

**return redirect()->route($this->route.'index');**

**}**

**Code Hapus User**

**public function destroy(string $id){**

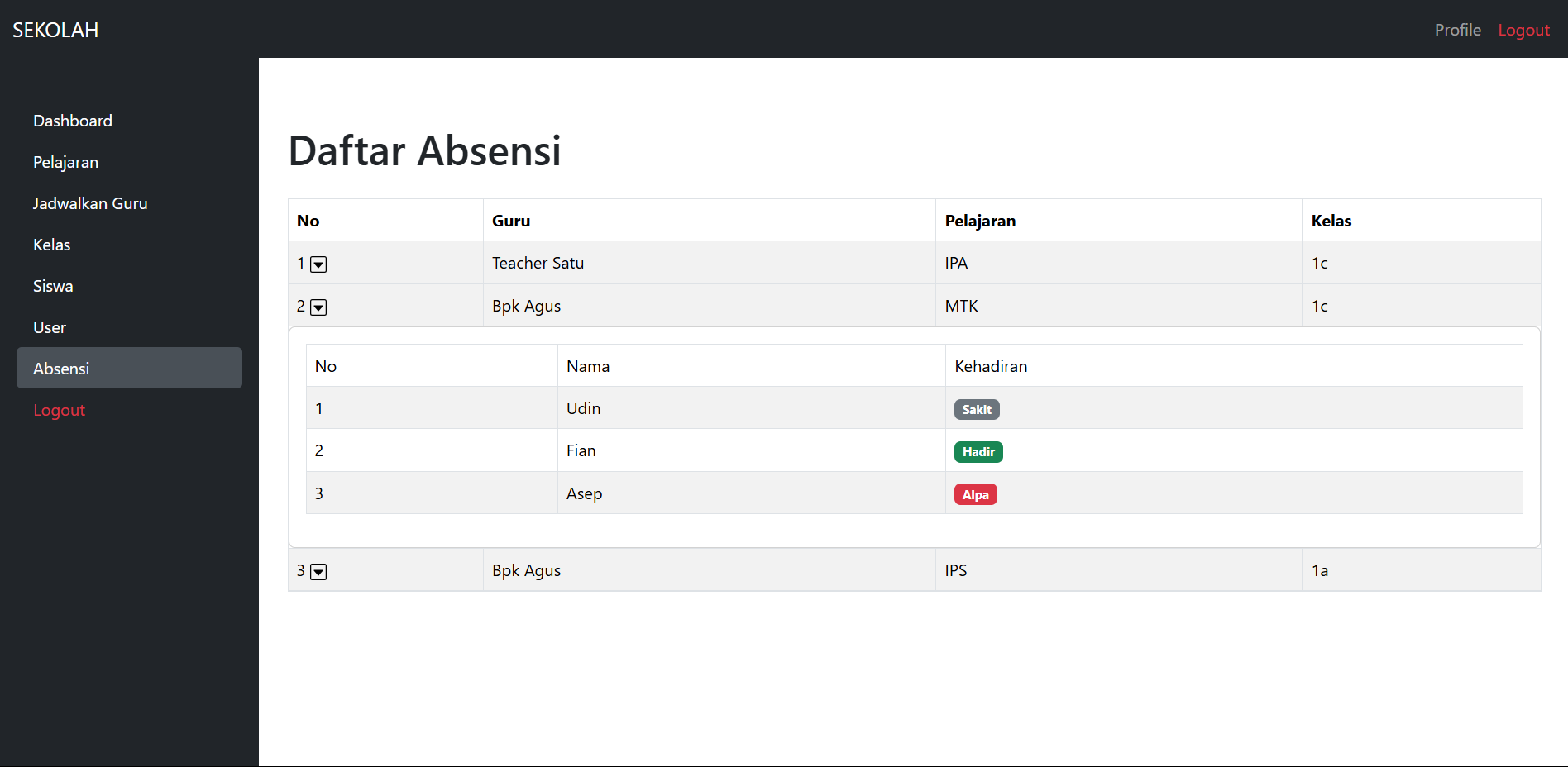
**$user = User::find($id);**

**$user->delete();**

**return redirect()->back();**

**}**

**Halaman Absensi**

****

**public $title = 'Absensi';**

**public $view = 'admin.absensi.';**

**public $route = 'admin.absensi.';**

**public function index(){**

**$data['title'] = $this->title;**

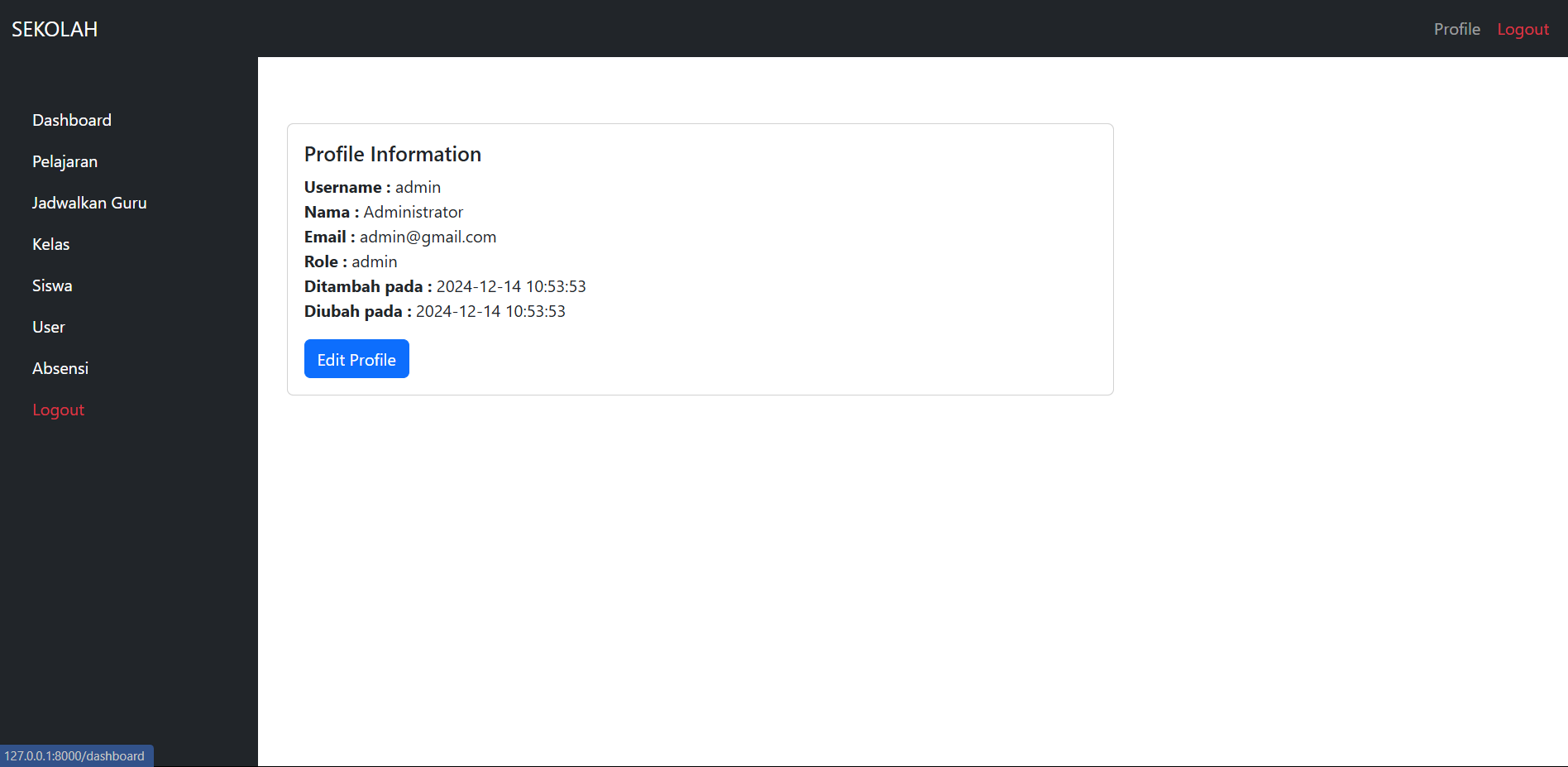
**$data['jadwal'] = Jadwal::get();**

**$data['route'] = route($this->route.'create');**

**return view($this->view.'index', $data);**

**}**

**Halaman Profil**

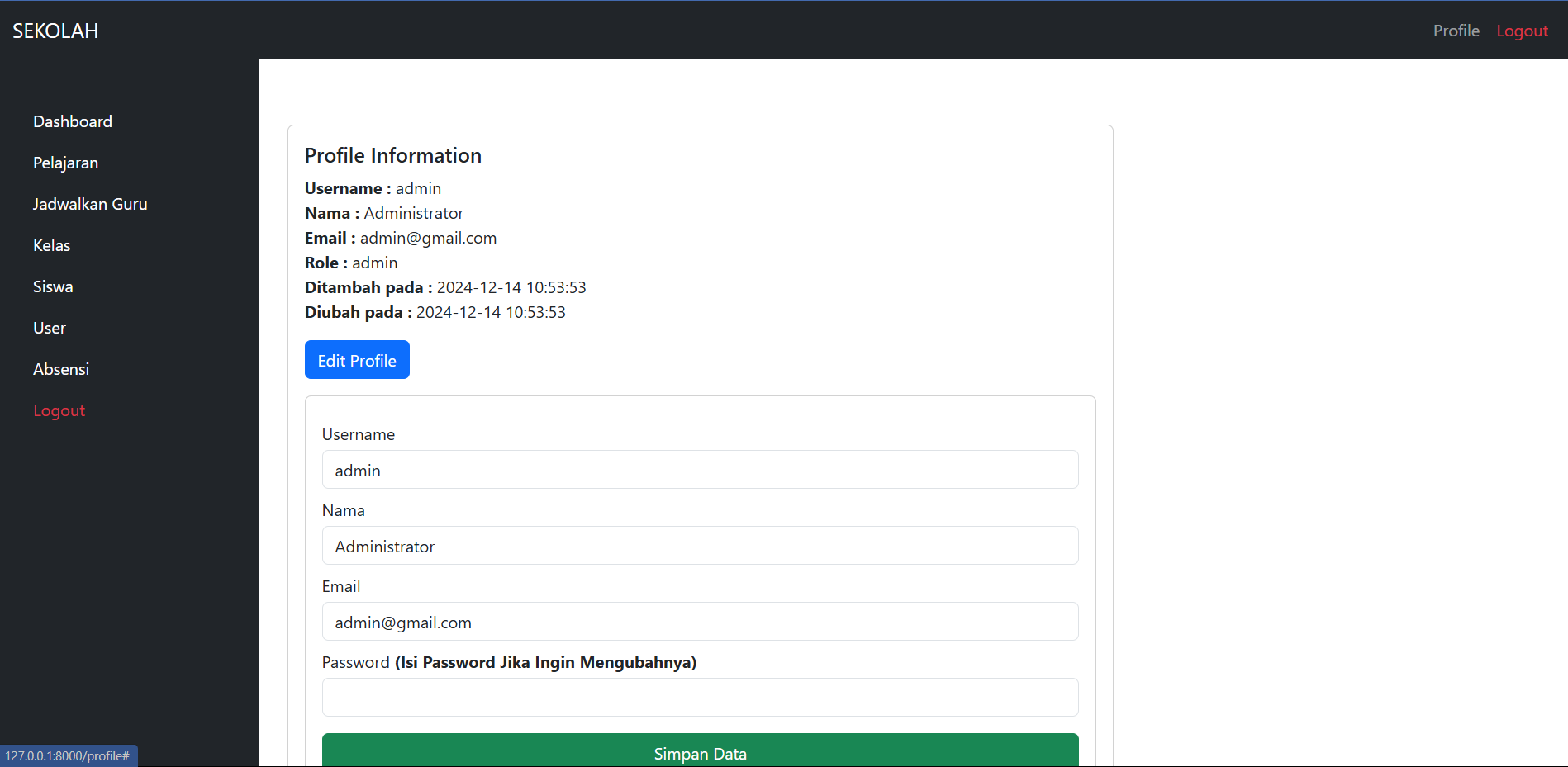
****

**public function index(){**

**return view('profile');**

**}**

**Edit Profil**

****

**public function update(Request $request){**

**$validate = $request->validate([**

**'username' => 'required',**

**'name' => 'required',**

**'email' => 'required|email',**

**'password' => 'nullable',**

**]);**

**if ($request->password) {**

**$validate['password'] = Hash::make($request->password);**

**}**

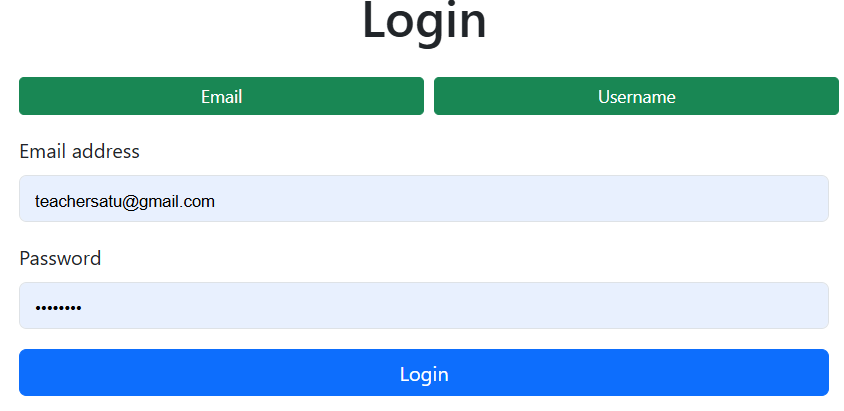
**User::whereId(Auth::user()->id)->update($validate);**

**return redirect()->back();**

**}**

**Implementasi User Guru**

**Halaman Login**

****

**public function showLoginForm(){**

**return view('auth.login');**

**}**

**public function login(Request $request){**

**$request->validate([**

**'username' => 'nullable',**

**'email' => 'nullable|email',**

**'password' => 'required',**

**]);**

**if ($request->username) {**

**$attempt = $request->only('username', 'password');**

**} else {**

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

**}**

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

**$request->session()->regenerate();**

**return redirect()->intended('/');**

**}**

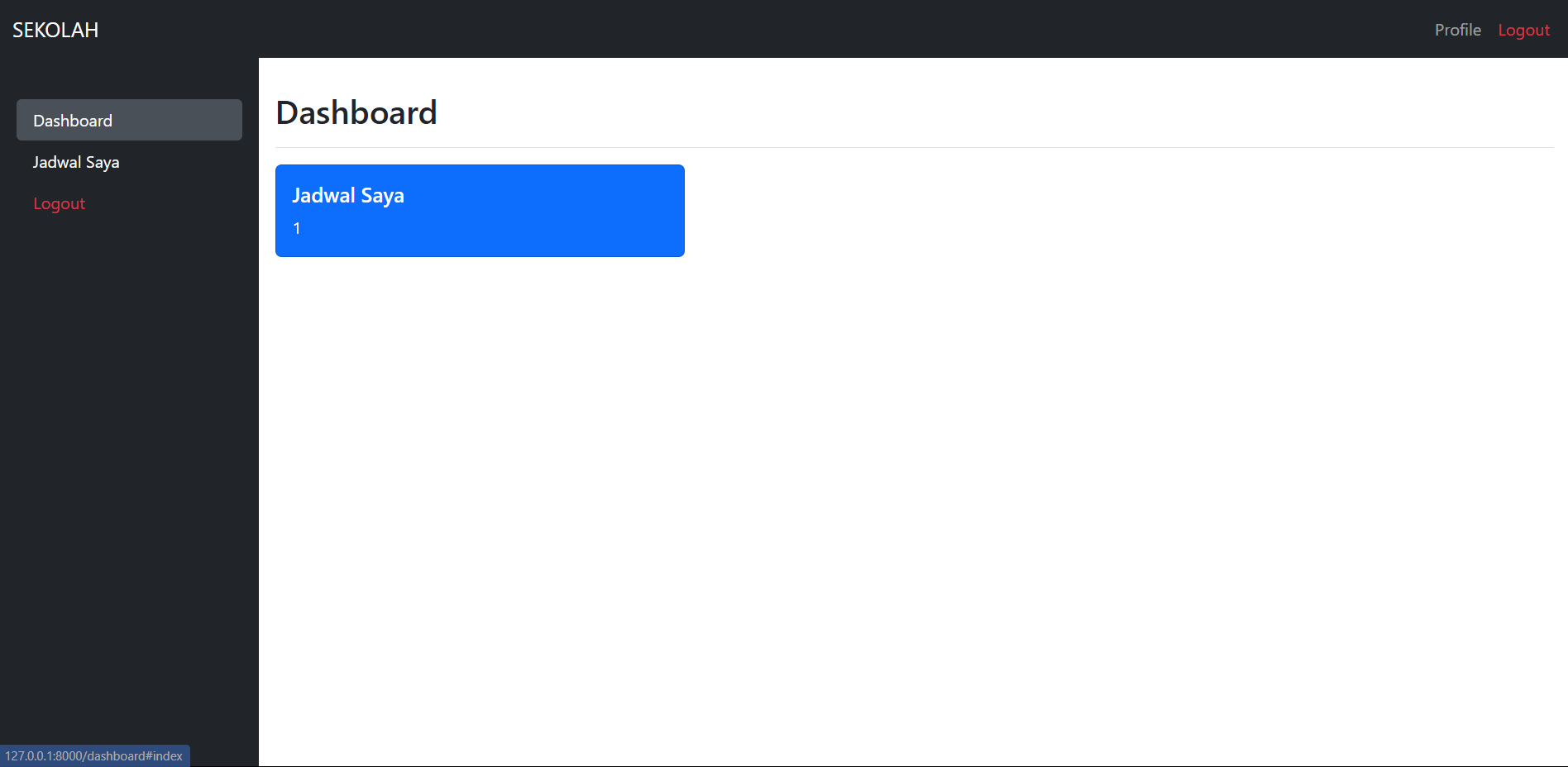
**return back()->withErrors([**

**'email' => 'The provided credentials do not match our records.',**

**]);**

**}**

**Halaman Dashboard**

****

**public function index() {**

**$data['siswa'] = Siswa::count();**

**$data['kelas'] = Kelas::count();**

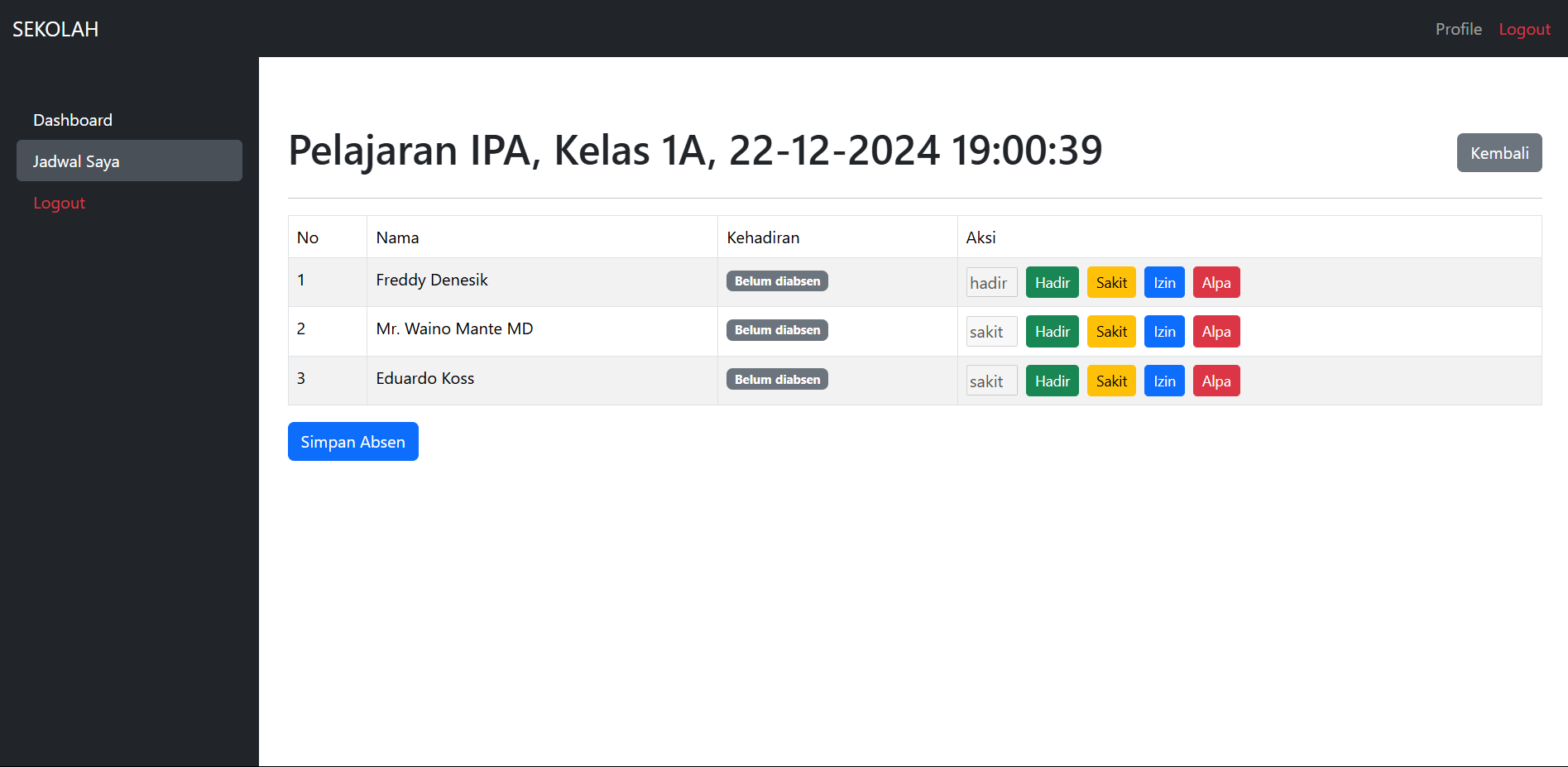
**$data['guru'] = User::where('role', 'guru')->count();**

**$data['jadwal'] = Jadwal::where('id\_user', Auth::user()->id)->count();**

**return view('dashboard', $data);**

**}**

**Halaman Jadwal Saya**

****

**public $title = 'Jadwal';**

**public $view = 'guru.jadwal.';**

**public $route = 'guru.jadwal.';**

**public function show($id){**

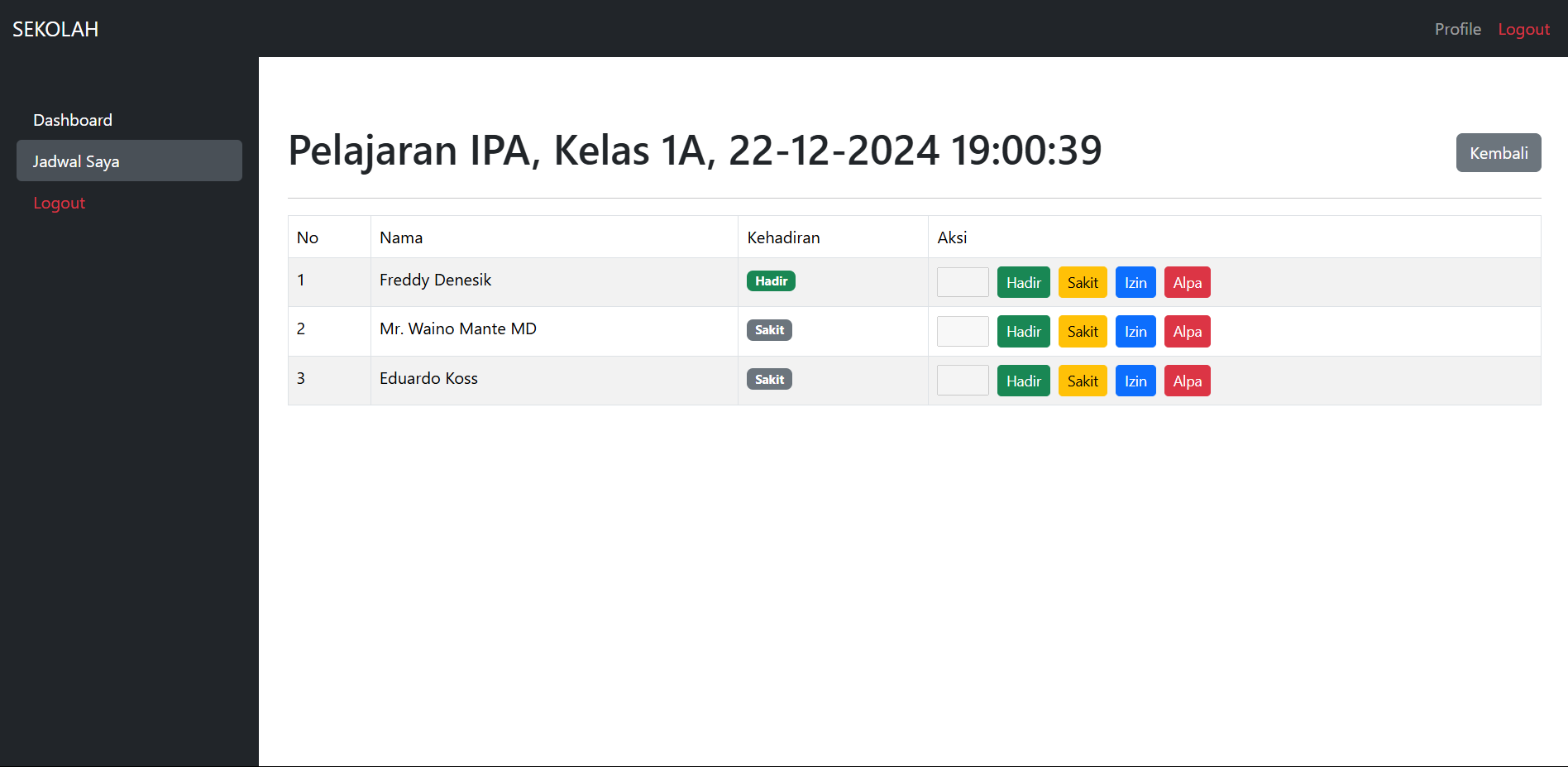
**$data['title'] = $this->title;**

**$data['jadwal'] = Jadwal::find($id);**

**return view($this->view.'show', $data);**

**}**

**Halaman Jadwal Saya Setelah di Simpan**

****

**public function store(Request $request){**

**$data['title'] = $this->title;**

**$validate = $request->validate([**

**'id\_siswa.\*' => 'required|exists:siswa,id',**

**'desc.\*' => 'nullable',**

**'id\_jadwal' => 'required|exists:jadwals,id',**

**'status.\*' => 'required|in:hadir,alpa,izin,sakit',**

**]);**

**$validate['date'] = Carbon::now();**

**foreach ($request->id\_siswa as $key => $value) {**

**Absensi::create([**

**'id\_siswa' => $value,**

**'id\_jadwal' => $request->id\_jadwal,**

**'date' => $validate['date'],**

**'status' => $request->status[$key],**

**]);**

**}**

**Jadwal::whereId($request->id\_jadwal)->update([**

**'status' => 'selesai'**

**]);**

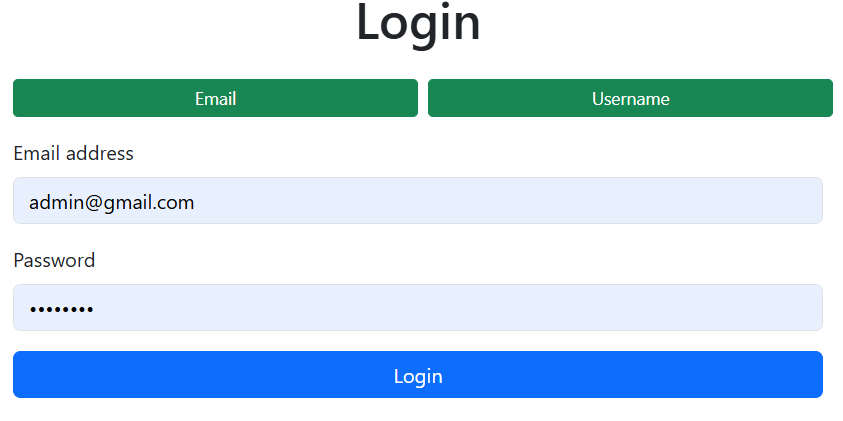
**return redirect()->back();**

**}**

#### **5. Langkah- langkah menjalankan aplikasi**

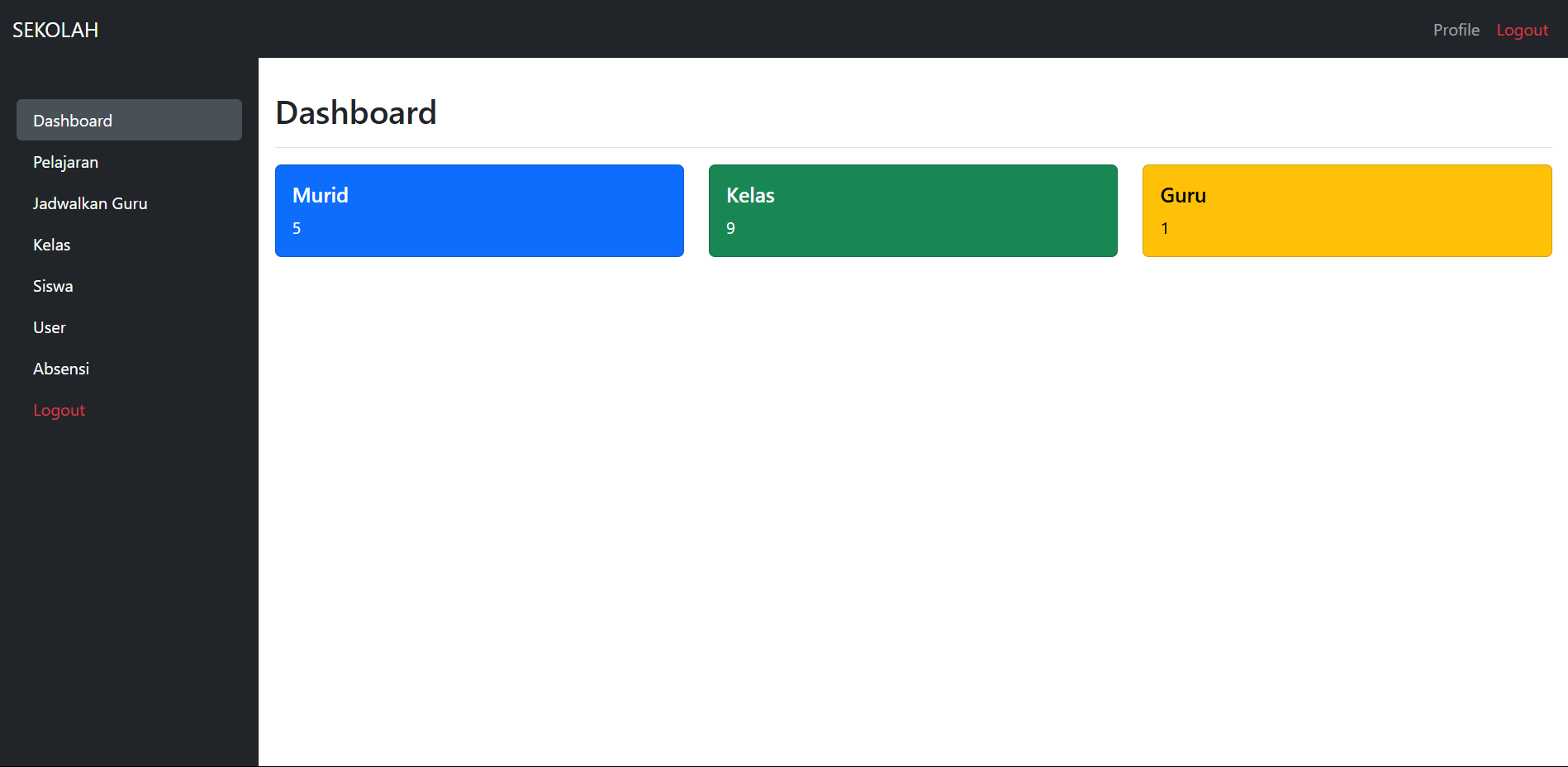
1. Buka xampp atau laragon, kemudian klik start pada MySQL dan Apache di xampp, klik start all jika menggunakan laragon.

2. Buka web browser dan ketikkan *http://127.0.0.1:8000/login* agar dapat mengakses halaman login.



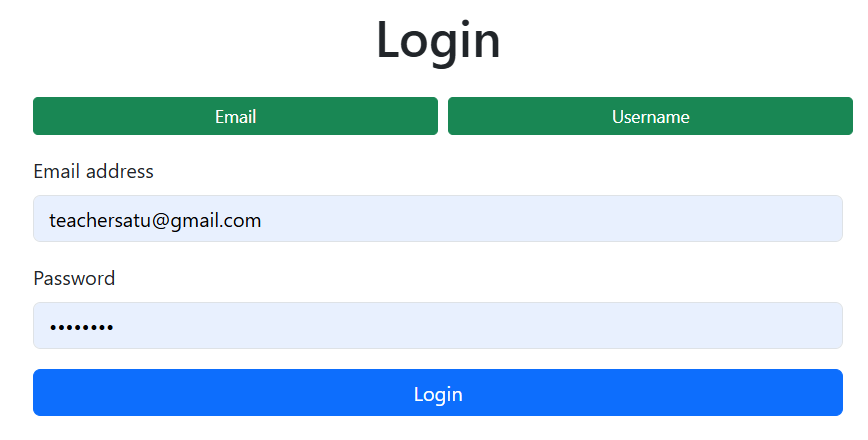
3. Melakukan login admin dengan memasukkan username admin@gmail.com dan password “12345678”.

4. Jika sudah berhasil login maka akan menampilkan halaman dashboard dan dapat melakukan pengelolaan data.

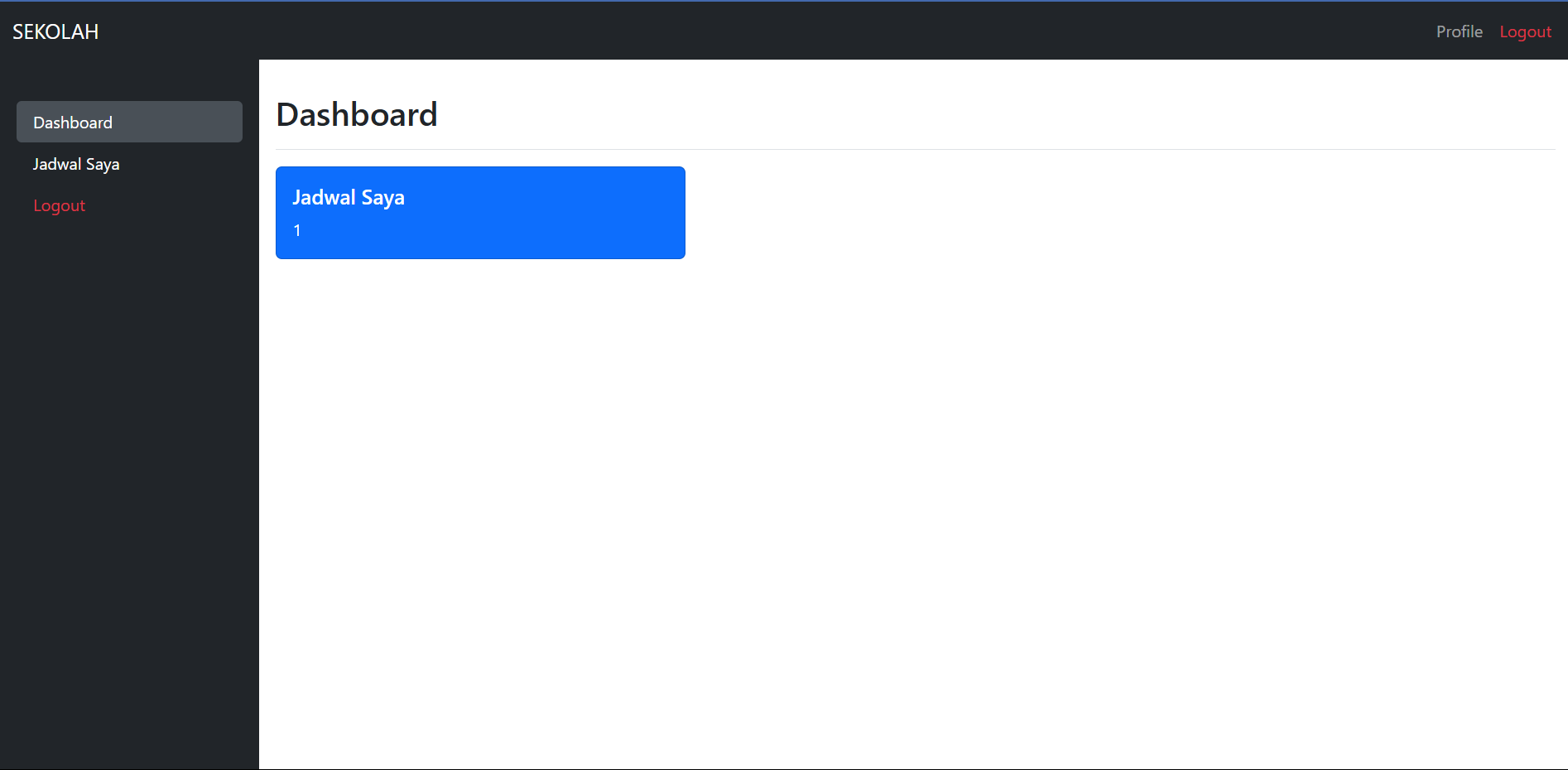


5. Buka web browser dan ketikkan *http://127.0.0.1:8000/login* agar dapat mengakses halaman login.

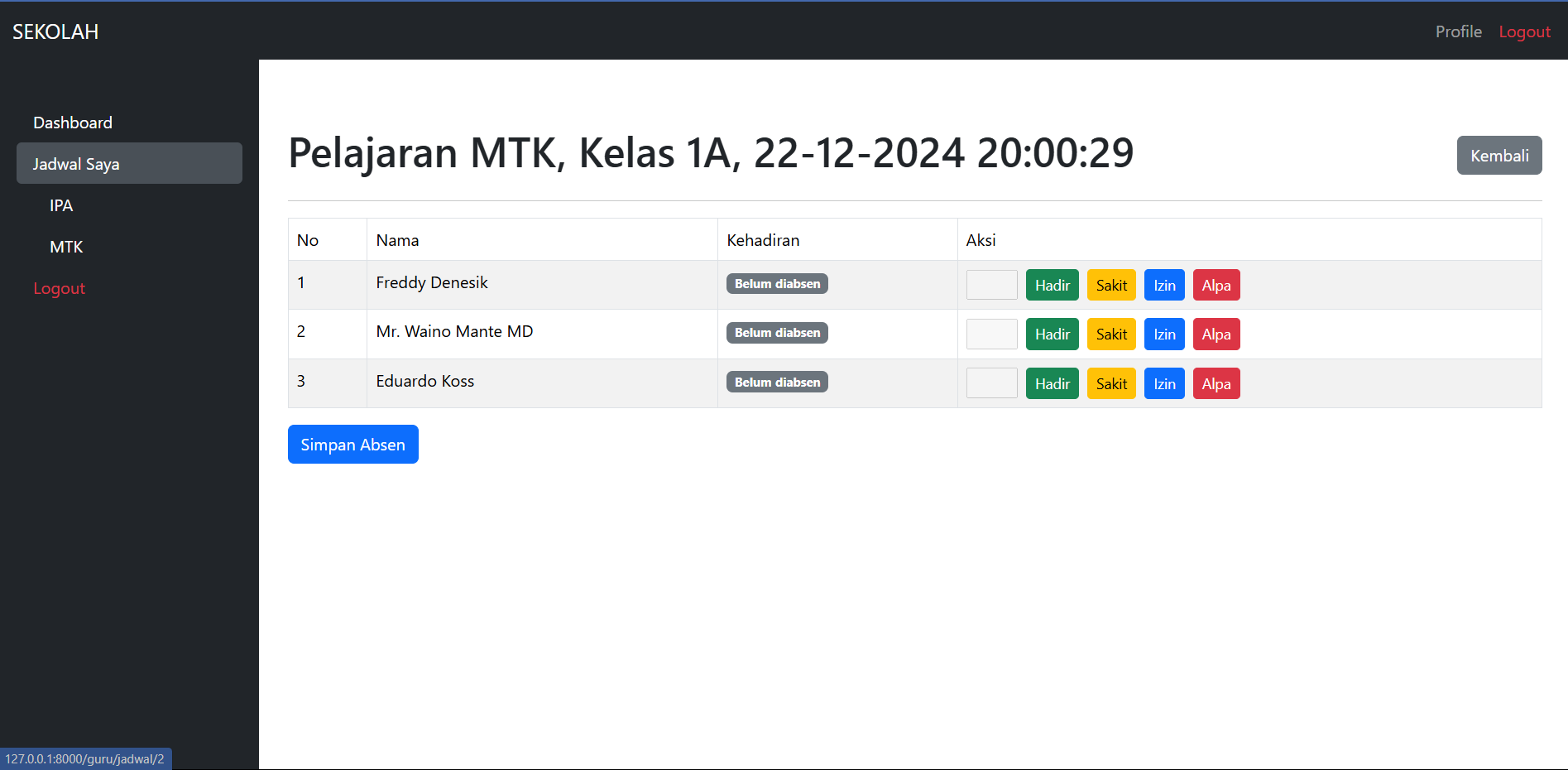
6. Melakukan login guru dengan memasukkan username user dan password “12345678” bawaan yang sudah dibuat oleh admin, password bisa diubah ketika sudah login sebagai guru.



7. Jika sudah berhasil login maka akan menampilkan halaman dashboard dan dapat melihat statistik berapa jumlah mapel untuk guru tersebut.



8. Klik Jadwal Saya untuk menampilkan jadwal-jadwal mapel yang sudah ditetapkan oleh admin untuk guru tersebut.

****

Setelah memilih mapel akan ditampilkan siswa-siswa pada kelas tersebut dan guru dapat mengisi absensi pada mapel tersebut.

9. Klik Simpan Absen maka selanjutnya data absen tersbut akan terkirim ke admin

#### **6. Kesimpulan dan Saran**

1. **Kesimpulan:**

Proyek pengembangan aplikasi web absensi siswa berhasil menyederhanakan proses pencatatan kehadiran secara otomatis, cepat, dan efisien dibandingkan metode manual. Aplikasi ini memungkinkan guru dan sekolah memantau kehadiran siswa secara real-time, mengelola data dengan lebih terstruktur, mudah diakses, serta mengurangi kesalahan pengolahan data. Secara keseluruhan, aplikasi ini meningkatkan efisiensi administrasi sekolah dan menyediakan data yang transparan serta akurat.

1. **Saran:**

Pengembangan selanjutnya dapat mencakup integrasi dengan sistem LMS, penambahan fitur geolokasi atau QR code untuk akurasi absensi, serta laporan analisis kehadiran untuk mendukung pengambilan keputusan. Versi mobile, fitur notifikasi, dan pengingat juga dapat meningkatkan aksesibilitas, kenyamanan, serta keterlibatan orang tua dalam pemantauan kehadiran siswa.

### 