**LAPORAN TUGAS BESAR**

**PEMROGRAMAN II / PEMROGRAMAN WEB II**

**SISTEM PENJADWALAN IBADAH GIDI**

202311056

ALFRED DUDE

Kelas B

**PROGRAM STUDI TEKNIK INFORMATIKA**

**UNIVERSITAS SEPULUH NOPEMBER PAPUA**

**2024**

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa atas berkat dan kasih karunia-Nya sehingga kami dapat menyelesaikan laporan proyek sistem yang berjudul "Sistem Penjadwalan Ibadah GIDI" ini dengan baik. Laporan ini dibuat sebagai bagian dari tugas mata kuliah yang kami ambil, sekaligus sebagai bentuk implementasi ilmu yang telah dipelajari selama perkuliahan. Kami menyadari bahwa terselesaikannya laporan ini tidak terlepas dari bimbingan, arahan, dan dukungan dari berbagai pihak. Oleh karena itu, dengan penuh hormat, kami menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

Bapak Ajenkris Yanto Kungkung, S.Kom., M.SI., selaku dosen pengampu mata kuliah, yang telah memberikan arahan, ilmu, serta motivasi kepada kami selama proses penyelesaian laporan ini. Semua pihak yang secara langsung maupun tidak langsung telah membantu dan memberikan dukungan moral maupun material hingga laporan ini selesai. Kami menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi isi maupun penyajiannya. Oleh karena itu, kami terbuka untuk menerima kritik dan saran yang membangun demi perbaikan di masa yang akan datang. Harapan kami, laporan ini dapat bermanfaat dan menjadi referensi bagi pembaca maupun pihak-pihak yang membutuhkannya.

Akhir kata, kami ucapkan terima kasih. Semoga laporan ini dapat memberikan kontribusi positif dalam dunia pendidikan dan pengembangan teknologi informasi, khususnya dalam konteks sistem penjadwalan ibadah GIDI.

Jayapura, 6 Desember 2024

Alfred Dude

**DAFTAR ISI**

[A. Desain Basis Data 6](#_Toc184394692)

[B. Implementasi 7](#_Toc184394693)

[**B.1.** **User** 7](#_Toc184394694)

[B.1.1. Model 7](#_Toc184394695)

[B.1.2. Controller dan View 8](#_Toc184394696)

[B.1.3. View 17](#_Toc184394697)

[**B.2.** **Members** 18](#_Toc184394698)

[B.2.1. Model 18](#_Toc184394699)

[B.2.2. Controller dan View 18](#_Toc184394700)

[B.2.3. View 23](#_Toc184394701)

[**B.3.** **Schedules** 24](#_Toc184394702)

[B.3.1. Model 24](#_Toc184394703)

[B.3.2. Controller dan View 24](#_Toc184394704)

[B.3.3. View 29](#_Toc184394705)

[**B.4.** **Devontion** 30](#_Toc184394706)

[B.4.1. Model 30](#_Toc184394707)

[B.4.2. Controller dan View 30](#_Toc184394708)

[B.4.3. View 34](#_Toc184394709)

[C. Blackbox Testing 36](#_Toc184394710)

[D. Daftar Pustaka 37](#_Toc184394711)

DAFTAR GAMBAR

[Gambar 1. Login 16](#_Toc184395121)

[Gambar 2. Register 17](#_Toc184395122)

[Gambar 3. Index Jemaat 22](#_Toc184395123)

[Gambar 4. Create Jemaat 23](#_Toc184395124)

[Gambar 5. Edit Jemaat 23](#_Toc184395125)

[Gambar 6. Index Schedule Admin Dan Jemaat 28](#_Toc184395126)

[Gambar 7. Create Schedule 29](#_Toc184395127)

[Gambar 8. Edit Schedule 29](#_Toc184395128)

[Gambar 9. Index Devotion 33](#_Toc184395129)

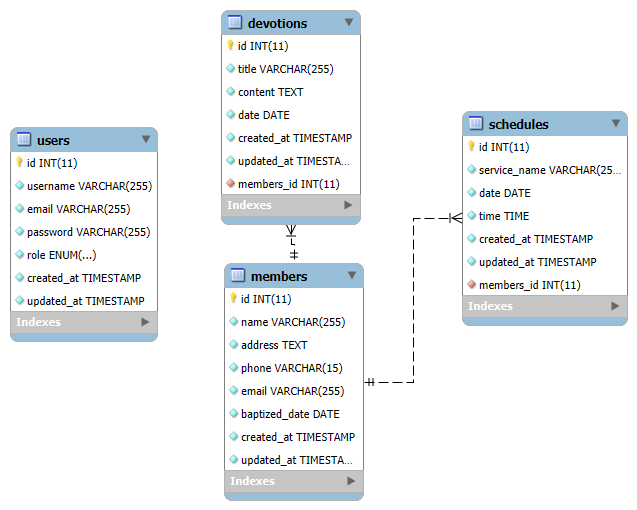
[Gambar 10. Create Devotion 33](#_Toc184395130)

[Gambar 11. Edit Devotion 34](#_Toc184395131)

DAFTAR TABEL

[Table 1. Blackbox 36](#_Toc184395908)

# Desain Basis Data



Gambar 1. Model Database Rekam Medik

Berikut penjelasan tiap tabel dan relasinya:

1. Tabel users

* Kolom:
  + id (Primary Key)
  + username
  + email
  + password
  + role (ENUM, mendeskripsikan peran pengguna)
  + created\_at (Waktu pembuatan data)
  + updated\_at (Waktu terakhir data diperbarui)
* Fungsi: Menyimpan data pengguna yang memiliki akses ke sistem.

Gambar yang Anda lampirkan adalah sebuah diagram Entity Relationship Diagram (ERD) yang mendeskripsikan relasi antar tabel dalam sebuah database. Berikut penjelasan tiap tabel dan relasinya:

* **1. Tabel users**
* **Kolom:**
  + id (Primary Key)
  + username
  + email
  + password
  + role (ENUM, mendeskripsikan peran pengguna)
  + created\_at (Waktu pembuatan data)
  + updated\_at (Waktu terakhir data diperbarui)
* Fungsi: Menyimpan data pengguna yang memiliki akses ke sistem.
* **2. Tabel members**
* **Kolom:**
  + id (Primary Key)
  + name (Nama anggota)
  + address (Alamat anggota)
  + phone (Nomor telepon anggota)
  + email (Email anggota)
  + baptized\_date (Tanggal baptisan)
  + created\_at (Waktu pembuatan data)
  + updated\_at (Waktu terakhir data diperbarui)
* Fungsi: Menyimpan informasi anggota
* **3. Tabel devotions**
* **Kolom:**
  + id (Primary Key)
  + title (Judul renungan)
  + content (Isi renungan)
  + date (Tanggal renungan)
  + created\_at (Waktu pembuatan data)
  + updated\_at (Waktu terakhir data diperbarui)
  + members\_id (Foreign Key, mengacu ke id di tabel members)
* Fungsi: Menyimpan data renungan yang ditulis oleh anggota.
* **4. Tabel schedules**
* **Kolom:**
  + id (Primary Key)
  + service\_name (Nama pelayanan/jadwal)
  + date (Tanggal pelayanan)
  + time (Waktu pelayanan)
  + created\_at (Waktu pembuatan data)
  + updated\_at (Waktu terakhir data diperbarui)
  + members\_id (Foreign Key, mengacu ke id di tabel members)
* Fungsi: Menyimpan data jadwal pelayanan yang terkait dengan anggota.
* **Relasi Antar Tabel**

1. **members → devotions** (One-to-Many)
   * Satu anggota (members) dapat menulis banyak renungan (devotions).
   * Relasi ini direpresentasikan melalui kolom members\_id di tabel devotions.
2. **members → schedules** (One-to-Many)
   * Satu anggota (members) dapat memiliki banyak jadwal pelayanan (schedules).
   * Relasi ini direpresentasikan melalui kolom members\_id di tabel schedules.

# Implementasi

## **User**

### Model

<?php

namespace App\Models;

use CodeIgniter\Model;

class UserModel extends Model

{

protected $table = 'users';

protected $primaryKey = 'id';

protected $allowedFields = ['username', 'email', 'password', 'role'];

protected $useTimestamps = true;

// Fungsi untuk menemukan user berdasarkan username

public function getUserByUsername($username)

{

return $this->where('username', $username)->first();

}

// Fungsi untuk menambah user baru

public function insertUser($data)

{

return $this->save($data);

}

}

### Controller dan View

1. Controller

<?php

namespace App\Controllers;

use App\Models\UserModel;

use CodeIgniter\Controller;

use CodeIgniter\HTTP\RedirectResponse;

class Auth extends Controller

{

// Halaman Register

public function register()

{

return view('auth/register');

}

// Proses Register

public function doRegister()

{

$validation = \Config\Services::validation();

$data = $this->request->getPost();

// Validasi form

$validation->setRules([

'username' => 'required|min\_length[3]|max\_length[20]|is\_unique[users.username]',

'email' => 'required|valid\_email|is\_unique[users.email]',

'password' => 'required|min\_length[8]',

'role' => 'required|in\_list[admin,jemaat]'

]);

if ($validation->withRequest($this->request)->run()) {

// Hash password

$data['password'] = password\_hash($data['password'], PASSWORD\_DEFAULT);

// Simpan ke database

$userModel = new UserModel();

$userModel->insertUser($data);

// Redirect ke login setelah berhasil register

return redirect()->to('/login');

} else {

// Jika validasi gagal, kembali ke halaman register dengan pesan error

return view('auth/register', ['validation' => $validation]);

}

}

// Halaman Login

public function login()

{

return view('auth/login');

}

// Proses Login

public function doLogin()

{

$data = $this->request->getPost();

$userModel = new UserModel();

// Cari user berdasarkan username

$user = $userModel->getUserByUsername($data['username']);

if ($user && password\_verify($data['password'], $user['password'])) {

// Set session

session()->set([

'user\_id' => $user['id'],

'username' => $user['username'],

'role' => $user['role'],

'is\_logged\_in' => true

]);

// Redirect sesuai role

if ($user['role'] == 'admin') {

return redirect()->to('/dashboard'); // Dashboard Admin

} else if ($user['role'] == 'jemaat') {

return redirect()->to('/dashboard'); // Halaman Jemaat

}

} else {

// Jika login gagal

session()->setFlashdata('error', 'Username atau Password salah');

return redirect()->to('/login');

}

}

// Logout

public function logout()

{

session()->destroy(); // Menghancurkan session

return redirect()->to('/login');

}

}

1. View
   * 1. Login

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Login</title>

<style>

/\* Resetting some default styles \*/

body, h2, p {

margin: 0;

padding: 0;

font-family: 'Arial', sans-serif;

}

/\* Background and main container styling \*/

body {

background-color: #f4f7fa;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

.login-container {

background-color: white;

padding: 30px;

border-radius: 8px;

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

width: 100%;

max-width: 400px;

}

h2 {

text-align: center;

color: #333;

margin-bottom: 20px;

}

/\* Form Styling \*/

label {

display: block;

font-size: 14px;

color: #555;

margin-bottom: 5px;

}

input[type="text"], input[type="password"] {

width: 100%;

padding: 12px;

margin: 8px 0 15px 0;

border: 1px solid #ccc;

border-radius: 4px;

box-sizing: border-box;

font-size: 14px;

}

input[type="text"]:focus, input[type="password"]:focus {

border-color: #6c63ff;

outline: none;

}

button[type="submit"] {

width: 100%;

padding: 14px;

background-color: #6c63ff;

color: white;

font-size: 16px;

border: none;

border-radius: 4px;

cursor: pointer;

transition: background-color 0.3s;

}

button[type="submit"]:hover {

background-color: #5a54e3;

}

/\* Error message styling \*/

.error-message {

color: red;

text-align: center;

margin-top: 10px;

font-size: 14px;

}

/\* Registration link \*/

.register-link {

text-align: center;

margin-top: 15px;

}

.register-link a {

color: #6c63ff;

text-decoration: none;

font-weight: bold;

}

.register-link a:hover {

text-decoration: underline;

}

/\* Responsive design \*/

@media (max-width: 600px) {

.login-container {

padding: 20px;

}

h2 {

font-size: 18px;

}

}

</style>

</head>

<body>

<div class="login-container">

<h2>Login</h2>

<form action="<?= site\_url('/auth/doLogin') ?>" method="post">

<?= csrf\_field() ?>

<label for="username">Username</label>

<input type="text" name="username" id="username" required>

<label for="password">Password</label>

<input type="password" name="password" id="password" required>

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

</form>

<?php if (session()->getFlashdata('error')): ?>

<div class="error-message"><?= session()->getFlashdata('error') ?></div>

<?php endif; ?>

<div class="register-link">

<p>Don't have an account? <a href="<?= site\_url('/register') ?>">Register</a></p>

</div>

</div>

</body>

</html>

* + 1. Register

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Register</title>

<style>

/\* Resetting default styles \*/

body, h2, p {

margin: 0;

padding: 0;

font-family: 'Arial', sans-serif;

}

/\* Background and main container styling \*/

body {

background-color: #f4f7fa;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

.register-container {

background-color: white;

padding: 30px;

border-radius: 8px;

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

width: 100%;

max-width: 400px;

}

h2 {

text-align: center;

color: #333;

margin-bottom: 20px;

}

/\* Form Styling \*/

label {

display: block;

font-size: 14px;

color: #555;

margin-bottom: 5px;

}

input[type="text"], input[type="email"], input[type="password"], select {

width: 100%;

padding: 12px;

margin: 8px 0 15px 0;

border: 1px solid #ccc;

border-radius: 4px;

box-sizing: border-box;

font-size: 14px;

}

input[type="text"]:focus, input[type="email"]:focus, input[type="password"]:focus, select:focus {

border-color: #6c63ff;

outline: none;

}

button[type="submit"] {

width: 100%;

padding: 14px;

background-color: #6c63ff;

color: white;

font-size: 16px;

border: none;

border-radius: 4px;

cursor: pointer;

transition: background-color 0.3s;

}

button[type="submit"]:hover {

background-color: #5a54e3;

}

/\* Error message styling \*/

.error-message {

color: red;

font-size: 12px;

margin-top: -12px;

margin-bottom: 15px;

}

/\* Registration link \*/

.register-link {

text-align: center;

margin-top: 15px;

}

.register-link a {

color: #6c63ff;

text-decoration: none;

font-weight: bold;

}

.register-link a:hover {

text-decoration: underline;

}

/\* Responsive design \*/

@media (max-width: 600px) {

.register-container {

padding: 20px;

}

h2 {

font-size: 18px;

}

}

</style>

</head>

<body>

<div class="register-container">

<h2>Register</h2>

<form action="<?= site\_url('/auth/doRegister') ?>" method="post">

<?= csrf\_field() ?>

<label for="username">Username</label>

<input type="text" name="username" id="username" value="<?= old('username') ?>" required>

<?php if (\Config\Services::validation()->getError('username')): ?>

<div class="error-message"><?= \Config\Services::validation()->getError('username') ?></div>

<?php endif; ?>

<label for="email">Email</label>

<input type="email" name="email" id="email" value="<?= old('email') ?>" required>

<?php if (\Config\Services::validation()->getError('email')): ?>

<div class="error-message"><?= \Config\Services::validation()->getError('email') ?></div>

<?php endif; ?>

<label for="password">Password</label>

<input type="password" name="password" id="password" required>

<?php if (\Config\Services::validation()->getError('password')): ?>

<div class="error-message"><?= \Config\Services::validation()->getError('password') ?></div>

<?php endif; ?>

<label for="role">Role</label>

<select name="role" id="role" required>

<option value="admin" <?= old('role') == 'admin' ? 'selected' : '' ?>>Admin</option>

<option value="jemaat" <?= old('role') == 'jemaat' ? 'selected' : '' ?>>Jemaat</option>

</select>

<?php if (\Config\Services::validation()->getError('role')): ?>

<div class="error-message"><?= \Config\Services::validation()->getError('role') ?></div>

<?php endif; ?>

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

</form>

<div class="register-link">

<p>Already have an account? <a href="<?= site\_url('/login') ?>">Login</a></p>

</div>

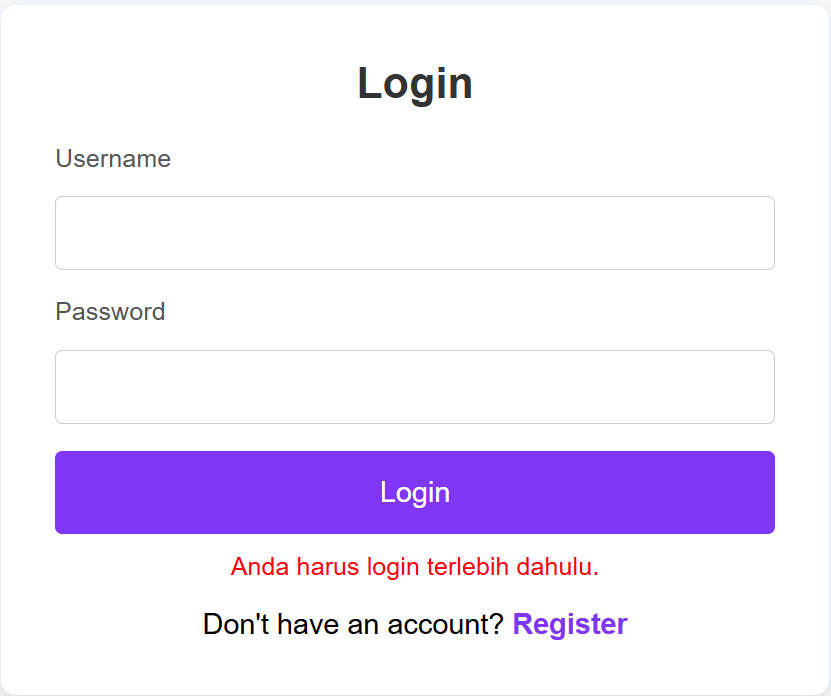
</div>

</body>

</html>

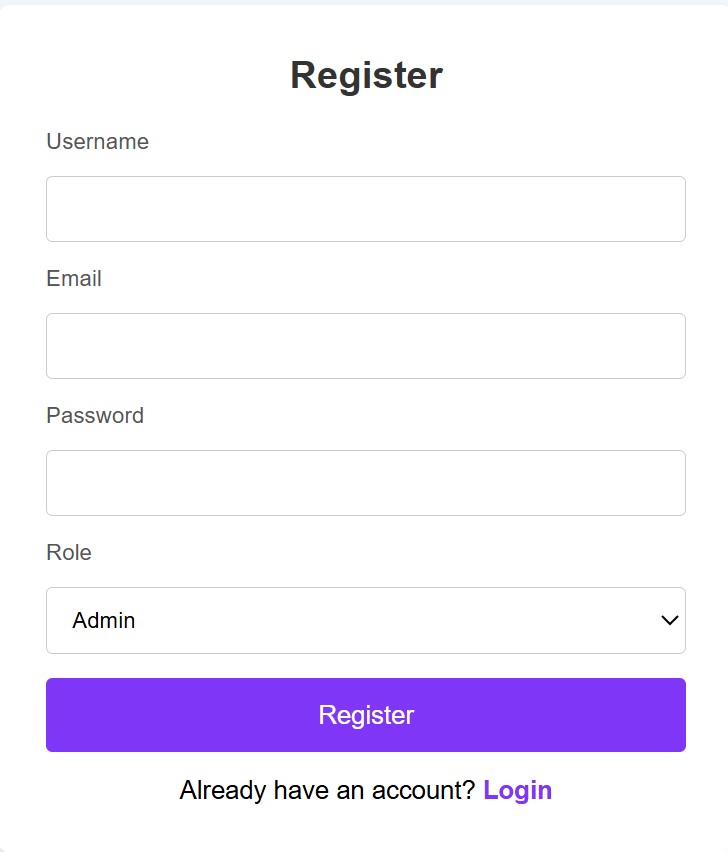
### View

1. **Tampilan Login Admin & Jemaat**



Gambar . Login

1. **Tampilan Register Admin dan Jemaat**

****

Gambar . Register

## **Members**

### Model

<?php

namespace App\Models;

use CodeIgniter\Model;

class MemberModel extends Model

{

protected $table = 'members';

protected $primaryKey = 'id';

protected $allowedFields = ['name', 'address', 'phone', 'email', 'baptized\_date'];

}

### Controller dan View

1. Menampilkan Data
2. Controller

public function \_\_construct()

{

$this->memberModel = new MemberModel();

}

// Menampilkan semua data jemaat

public function index()

{

$data['members'] = $this->memberModel->findAll();

return view('members/index', $data);

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2 class="text-center mb-4">Data Jemaat</h2>

<a href="/members/create" class="btn btn-primary mb-3">Tambah Data Jemaat</a>

<!-- Card untuk Menampilkan Tabel Data Jemaat -->

<div class="card shadow-sm">

<div class="card-header bg-primary text-white">

<h5 class="mb-0">Daftar Jemaat Gereja</h5>

</div>

<div class="card-body">

<table class="table table-striped table-hover">

<thead class="thead-light">

<tr>

<th>No</th>

<th>Nama</th>

<th>Alamat</th>

<th>No. Telepon</th>

<th>Email</th>

<th>Tanggal Baptis</th>

<th>Aksi</th>

</tr>

</thead>

<tbody>

<?php foreach ($members as $key => $member): ?>

<tr>

<td><?= $key + 1 ?></td>

<td><?= esc($member['name']) ?></td>

<td><?= esc($member['address']) ?></td>

<td><?= esc($member['phone']) ?></td>

<td><?= esc($member['email']) ?></td>

<td><?= esc($member['baptized\_date']) ?></td>

<td>

<a href="/members/edit/<?= $member['id'] ?>" class="btn btn-warning btn-sm">Edit</a>

<a href="/members/delete/<?= $member['id'] ?>" class="btn btn-danger btn-sm" onclick="return confirm('Hapus data ini?')">Hapus</a>

</td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

</div>

</div>

</div>

<?= $this->endSection() ?>

1. Menambahkan Data
2. Controller

// Menambahkan data jemaat baru

public function create()

{

return view('members/create');

}

public function store()

{

$this->memberModel->save([

'name' => $this->request->getPost('name'),

'address' => $this->request->getPost('address'),

'phone' => $this->request->getPost('phone'),

'email' => $this->request->getPost('email'),

'baptized\_date' => $this->request->getPost('baptized\_date'),

]);

return redirect()->to('/members');

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2 class="text-center mb-4">Tambah Data Jemaat</h2>

<!-- Card untuk Form Tambah Data -->

<div class="card shadow-sm">

<div class="card-header bg-primary text-white">

<h5 class="mb-0">Form Tambah Data Jemaat</h5>

</div>

<div class="card-body">

<!-- Form Tambah Data -->

<form action="/members/store" method="POST">

<?= csrf\_field() ?>

<div class="mb-3">

<label for="name" class="form-label">Nama Jemaat</label>

<input type="text" class="form-control" id="name" name="name" required>

</div>

<div class="mb-3">

<label for="address" class="form-label">Alamat</label>

<input type="text" class="form-control" id="address" name="address" required>

</div>

<div class="mb-3">

<label for="phone" class="form-label">No. Telepon</label>

<input type="text" class="form-control" id="phone" name="phone" required>

</div>

<div class="mb-3">

<label for="email" class="form-label">Email</label>

<input type="email" class="form-control" id="email" name="email" required>

</div>

<div class="mb-3">

<label for="baptized\_date" class="form-label">Tanggal Baptis</label>

<input type="date" class="form-control" id="baptized\_date" name="baptized\_date" required>

</div>

<button type="submit" class="btn btn-primary">Simpan Data</button>

<a href="/members" class="btn btn-secondary">Kembali</a>

</form>

</div>

</div>

</div>

<?= $this->endSection() ?>

1. Mengubah Data
2. Controller

public function edit($id)

{

$data['member'] = $this->memberModel->find($id);

return view('members/edit', $data);

}

public function update($id)

{

$this->memberModel->update($id, [

'name' => $this->request->getPost('name'),

'address' => $this->request->getPost('address'),

'phone' => $this->request->getPost('phone'),

'email' => $this->request->getPost('email'),

'baptized\_date' => $this->request->getPost('baptized\_date'),

]);

return redirect()->to('/members');

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2 class="text-center mb-4">Edit Data Jemaat</h2>

<!-- Form Edit Data -->

<form action="/members/update/<?= $member['id'] ?>" method="POST">

<?= csrf\_field() ?>

<!-- <input type="hidden" name="\_method" value="PUT"> Method Spoofing untuk PUT -->

<div class="mb-3">

<label for="name" class="form-label">Nama Jemaat</label>

<input type="text" class="form-control" id="name" name="name" value="<?= old('name', $member['name']) ?>" required>

</div>

<div class="mb-3">

<label for="address" class="form-label">Alamat</label>

<input type="text" class="form-control" id="address" name="address" value="<?= old('address', $member['address']) ?>" required>

</div>

<div class="mb-3">

<label for="phone" class="form-label">No. Telepon</label>

<input type="text" class="form-control" id="phone" name="phone" value="<?= old('phone', $member['phone']) ?>" required>

</div>

<div class="mb-3">

<label for="email" class="form-label">Email</label>

<input type="email" class="form-control" id="email" name="email" value="<?= old('email', $member['email']) ?>" required>

</div>

<div class="mb-3">

<label for="baptized\_date" class="form-label">Tanggal Baptis</label>

<input type="date" class="form-control" id="baptized\_date" name="baptized\_date" value="<?= old('baptized\_date', $member['baptized\_date']) ?>" required>

</div>

<button type="submit" class="btn btn-success">Perbarui Data</button>

<a href="/members" class="btn btn-secondary">Kembali</a>

</form>

</div>

<?= $this->endSection() ?>

1. Menghapus Data
2. Controller

// Menghapus data jemaat

public function delete($id)

{

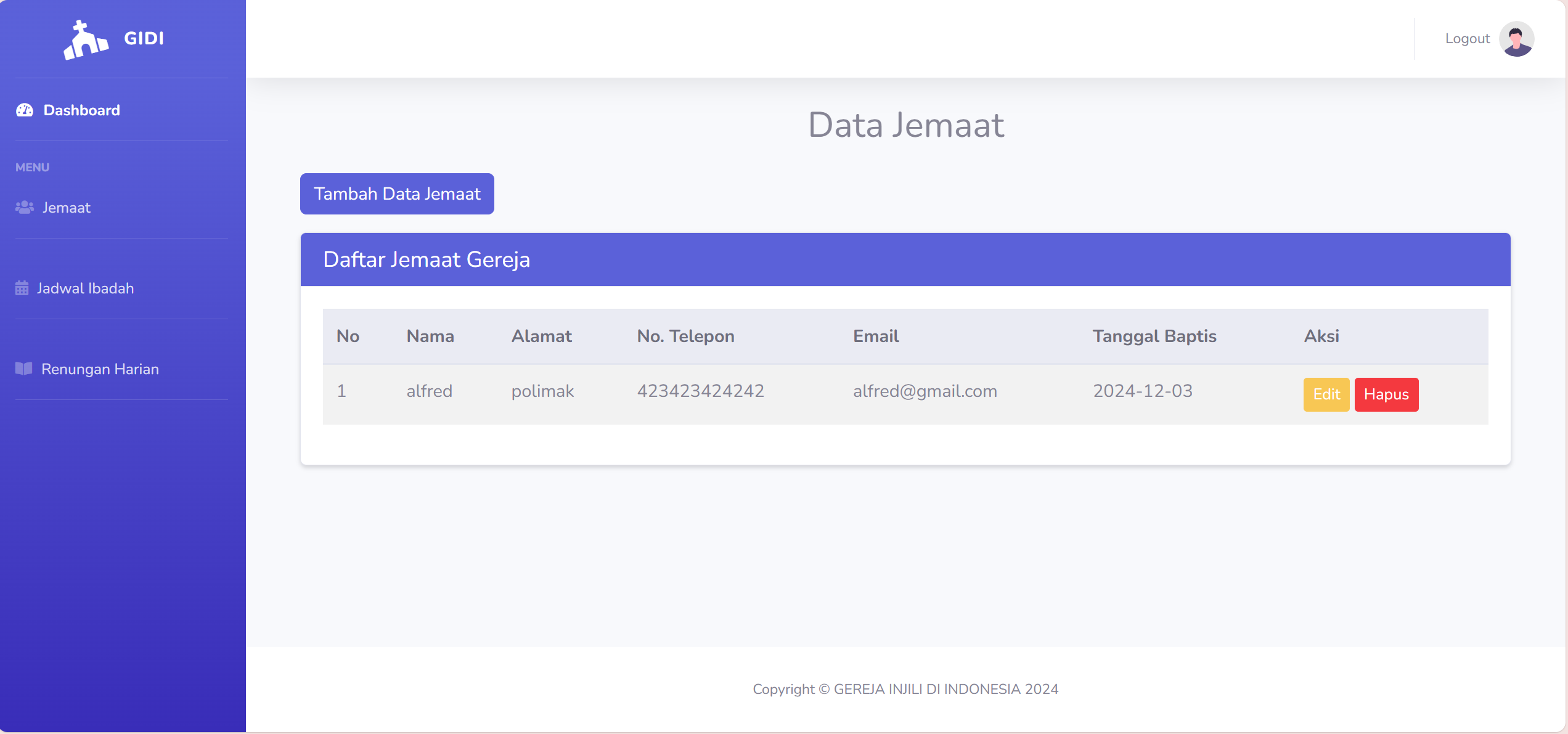
$this->memberModel->delete($id);

return redirect()->to('/members');

}

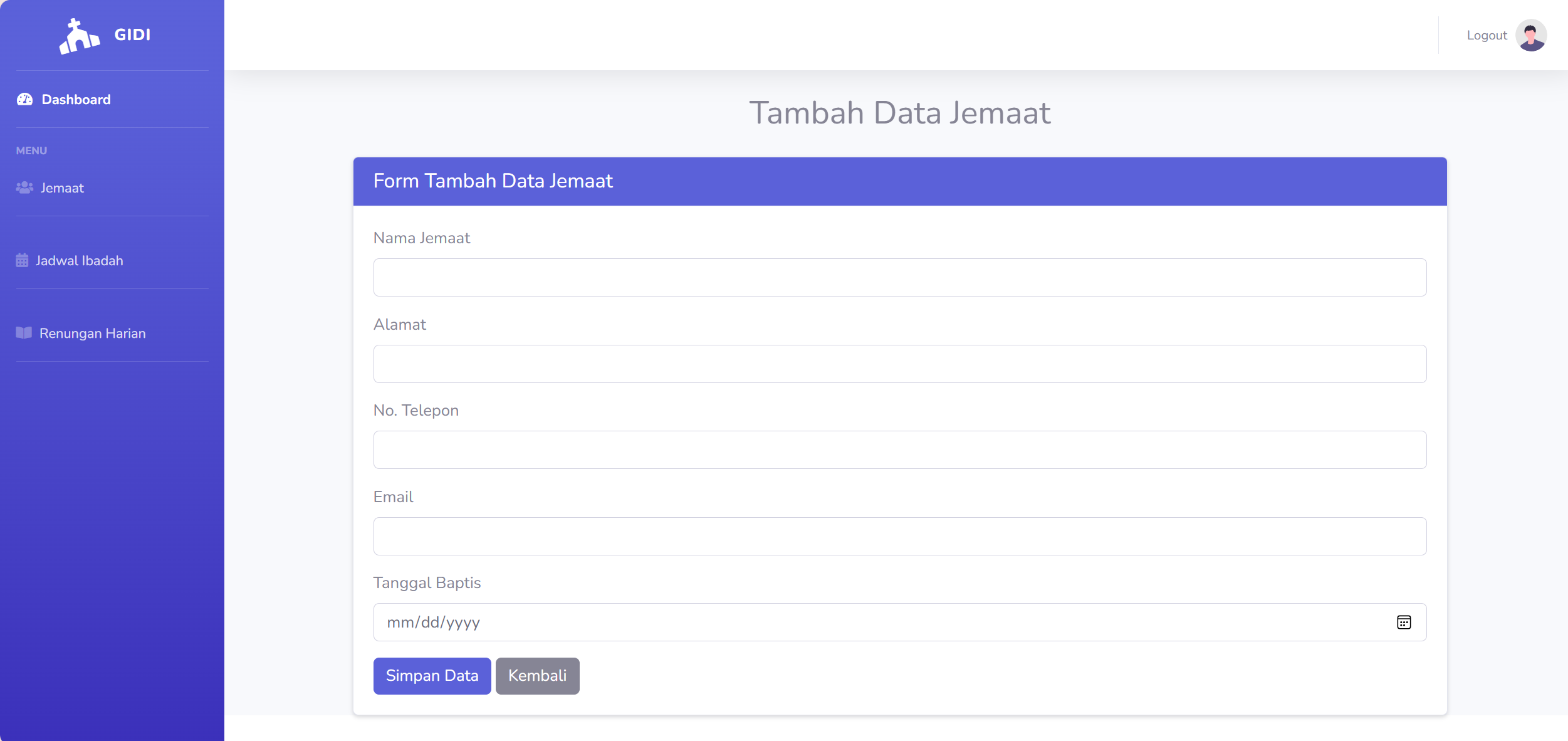
### View

1. **Index Jemaat**

****

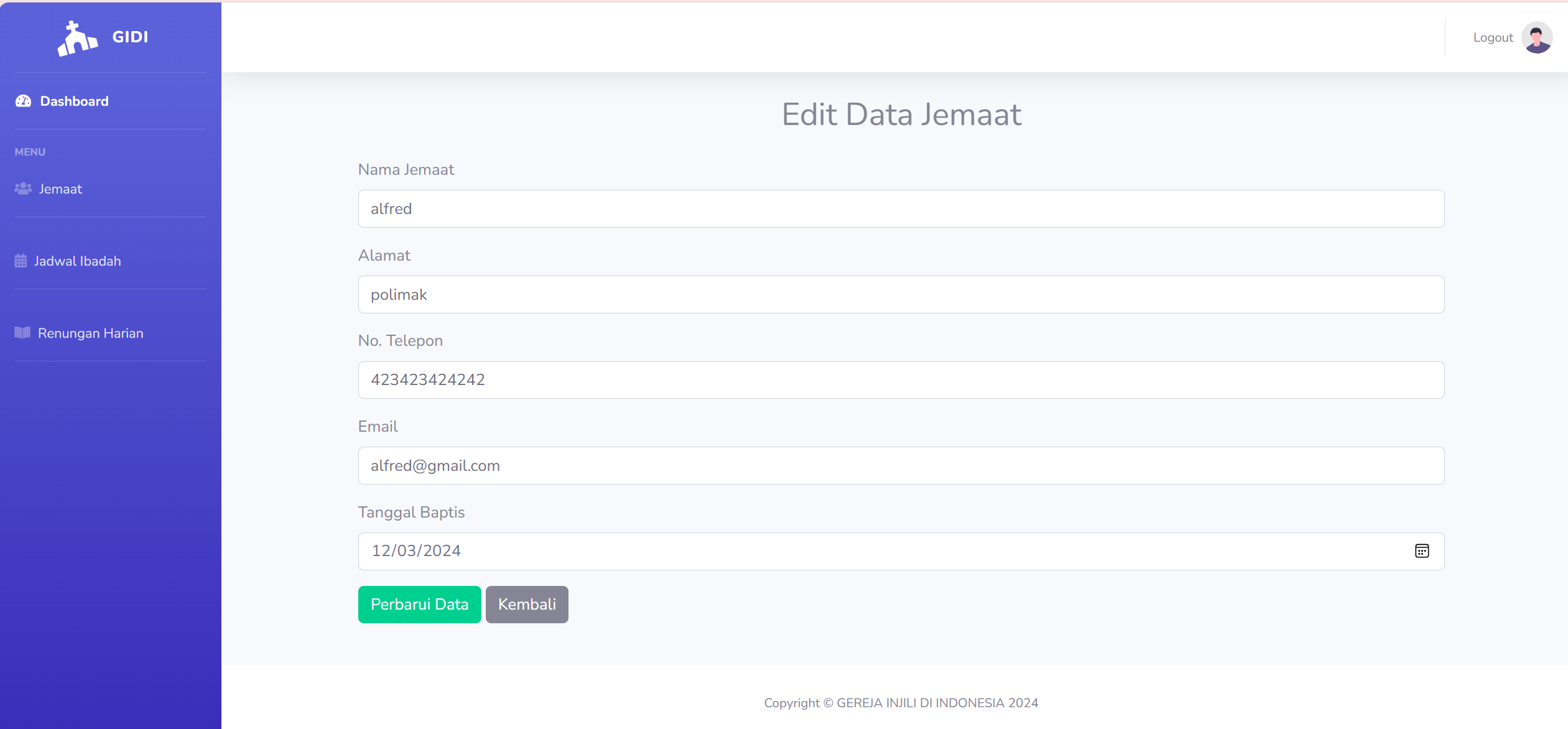
Gambar . Index Jemaat

1. **Tambah Jemaat**



Gambar . Create Jemaat

1. **Ubah Jemaat**



Gambar . Edit Jemaat

## **Schedules**

### Model

<?php

namespace App\Models;

use CodeIgniter\Model;

class ScheduleModel extends Model

{

protected $table = 'schedules';

protected $primaryKey = 'id';

protected $allowedFields = ['service\_name', 'date', 'time'];

public function getNextSchedule()

{

return $this->where('date >=', date('Y-m-d'))->orderBy('date', 'asc')->first();

}

}

### Controller dan View

1. Menampilkan Data
2. Controller

public function \_\_construct()

{

$this->scheduleModel = new ScheduleModel();

}

// Menampilkan semua jadwal

public function index()

{

$data['schedules'] = $this->scheduleModel->findAll();

return view('schedules/index', $data);

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2 class="text-center mb-4">Jadwal Ibadah</h2>

<!-- Card for Jadwal Ibadah Table -->

<div class="card shadow-lg">

<div class="card-header bg-gradient text-white">

<h5 class="mb-0">Daftar Jadwal Ibadah Gereja</h5>

</div>

<div class="card-body">

<div class="d-flex justify-content-between mb-4">

<a href="/schedules/create" class="btn btn-primary">Tambah Jadwal Ibadah</a>

<!-- Search Bar -->

<div class="w-50">

<input type="text" id="search" class="form-control" placeholder="Cari Jadwal Ibadah...">

</div>

</div>

<!-- Table -->

<div class="table-responsive">

<table class="table table-hover table-striped">

<thead class="thead-light">

<tr>

<th>No</th>

<th>Nama Ibadah</th>

<th>Tanggal</th>

<th>Waktu</th>

<th>Aksi</th>

</tr>

</thead>

<tbody id="schedule-list">

<?php foreach ($schedules as $key => $schedule): ?>

<tr>

<td><?= $key + 1 ?></td>

<td><?= $schedule['service\_name'] ?></td>

<td><?= $schedule['date'] ?></td>

<td><?= $schedule['time'] ?></td>

<td>

<a href="/schedules/edit/<?= $schedule['id'] ?>" class="btn btn-warning btn-sm">Edit</a>

<a href="/schedules/delete/<?= $schedule['id'] ?>" class="btn btn-danger btn-sm" onclick="return confirm('Hapus jadwal ini?')">Hapus</a>

</td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

</div>

<!-- No data message -->

<div id="no-data" class="text-center text-danger" style="display: none;">

<p>Data jadwal ibadah tidak ditemukan.</p>

</div>

</div>

</div>

</div>

<!-- Javascript to filter search -->

<script>

document.getElementById('search').addEventListener('input', function() {

filterTable();

});

function filterTable() {

let searchQuery = document.getElementById('search').value.toLowerCase();

let rows = document.querySelectorAll('#schedule-list tr');

let found = false;

rows.forEach(row => {

let name = row.cells[1].innerText.toLowerCase();

let date = row.cells[2].innerText.toLowerCase();

let time = row.cells[3].innerText.toLowerCase();

if (name.includes(searchQuery) || date.includes(searchQuery) || time.includes(searchQuery)) {

row.style.display = '';

found = true;

} else {

row.style.display = 'none';

}

});

document.getElementById('no-data').style.display = found ? 'none' : 'block';

}

</script>

<?= $this->endSection() ?>

1. Menambahkan Data
2. Controller

public function create()

{

return view('schedules/create');

}

public function store()

{

$this->scheduleModel->save([

'service\_name' => $this->request->getPost('service\_name'),

'date' => $this->request->getPost('date'),

'time' => $this->request->getPost('time'),

]);

return redirect()->to('/schedules');

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2>Tambah Jadwal Ibadah</h2>

<form action="/schedules/store" method="post">

<div class="form-group">

<label>Nama Ibadah</label>

<input type="text" name="service\_name" class="form-control" required>

</div>

<div class="form-group">

<label>Tanggal</label>

<input type="date" name="date" class="form-control" required>

</div>

<div class="form-group">

<label>Waktu</label>

<input type="time" name="time" class="form-control" required>

</div>

<button type="submit" class="btn btn-success">Simpan</button>

<a href="/schedules" class="btn btn-secondary">Kembali</a>

</form>

</div>

<?= $this->endSection() ?>

1. Mengubah Data
2. Controller

public function edit($id)

{

$data['schedule'] = $this->scheduleModel->find($id);

return view('schedules/edit', $data);

}

public function update($id)

{

$this->scheduleModel->update($id, [

'service\_name' => $this->request->getPost('service\_name'),

'date' => $this->request->getPost('date'),

'time' => $this->request->getPost('time'),

]);

return redirect()->to('/schedules');

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2>Edit Jadwal Ibadah</h2>

<form action="/schedules/update/<?= $schedule['id'] ?>" method="post">

<div class="form-group">

<label>Nama Ibadah</label>

<input type="text" name="service\_name" class="form-control" value="<?= $schedule['service\_name'] ?>" required>

</div>

<div class="form-group">

<label>Tanggal</label>

<input type="date" name="date" class="form-control" value="<?= $schedule['date'] ?>" required>

</div>

<div class="form-group">

<label>Waktu</label>

<input type="time" name="time" class="form-control" value="<?= $schedule['time'] ?>" required>

</div>

<button type="submit" class="btn btn-success">Simpan Perubahan</button>

<a href="/schedules" class="btn btn-secondary">Kembali</a>

</form>

</div>

<?= $this->endSection() ?>

1. Menghapus Data
2. Controller

public function delete($id)

{

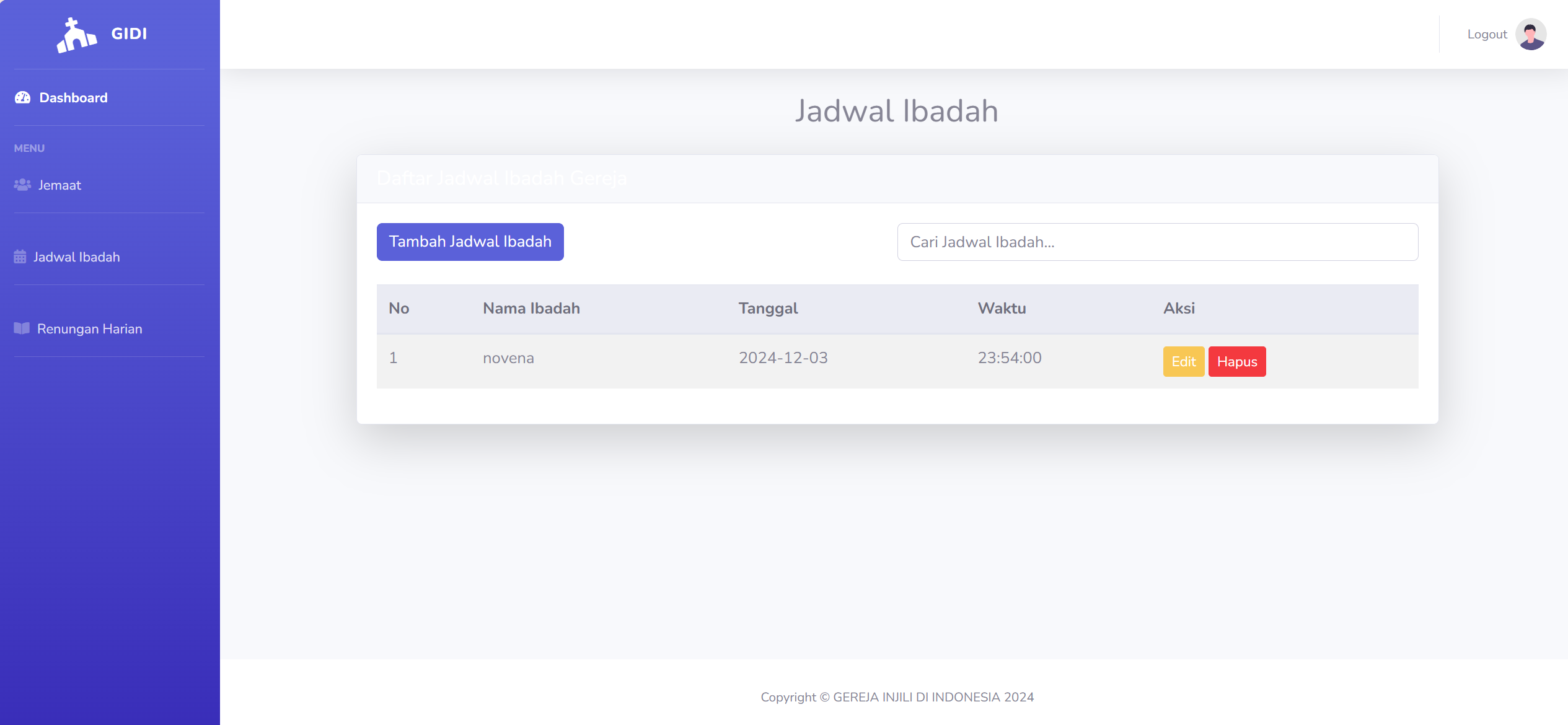
$this->scheduleModel->delete($id);

return redirect()->to('/schedules');

}

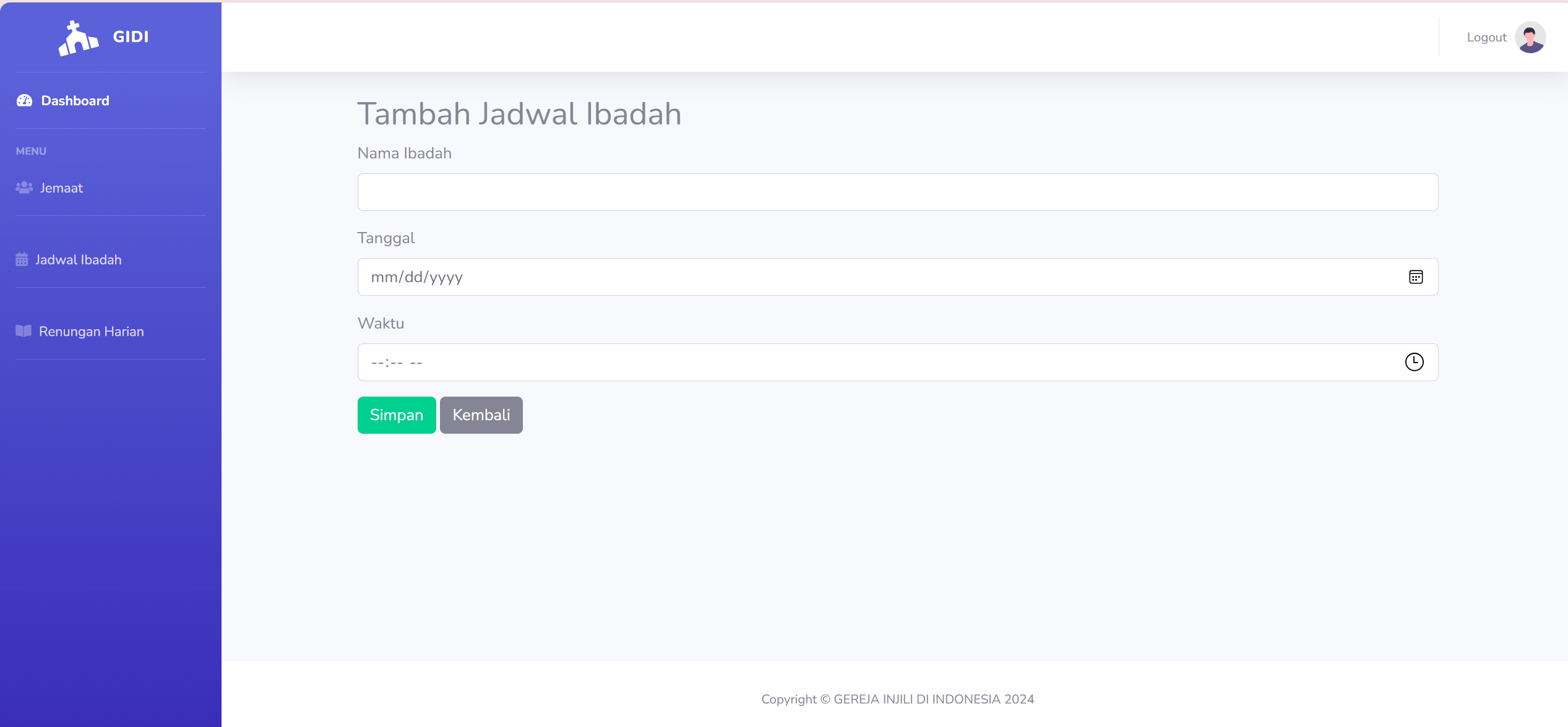
### View

1. **Index Schedule Admin dan Jemaat**

****

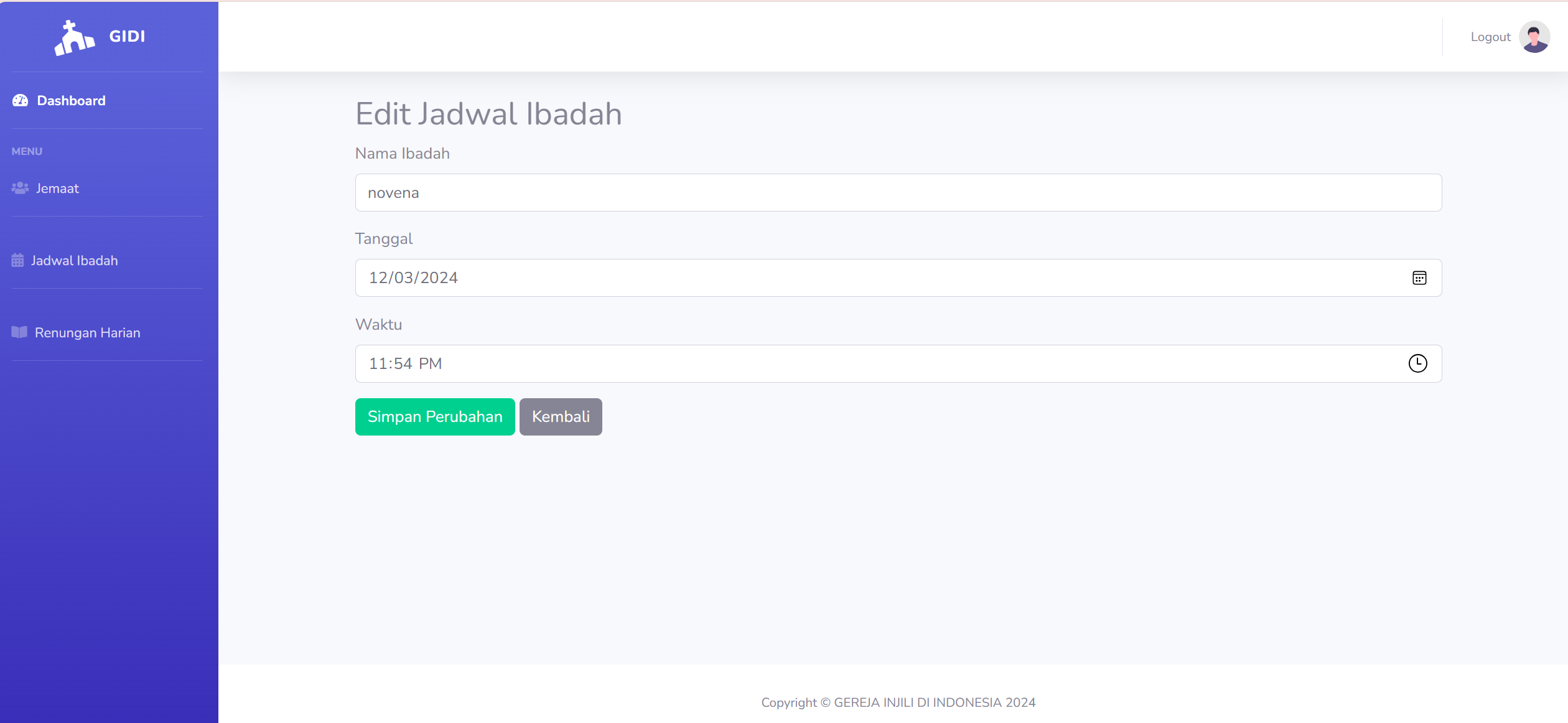
Gambar . Index Schedule Admin Dan Jemaat

1. **Tambah Schedule**

****

Gambar . Create Schedule

1. **Ubah Schedule**

****

Gambar . Edit Schedule

## **Devontion**

### Model

<?php

namespace App\Models;

use CodeIgniter\Model;

class DevotionModel extends Model

{

protected $table = 'devotions';

protected $primaryKey = 'id';

protected $allowedFields = ['title', 'content', 'date'];

public function getLatestDevotion()

{

return $this->orderBy('date', 'desc')->first();

}

}

### Controller dan View

1. Menampilkan Data
2. Controller

public function \_\_construct()

{

$this->devotionModel = new DevotionModel();

}

// Menampilkan semua renungan

public function index()

{

$data['devotions'] = $this->devotionModel->findAll();

return view('devotions/index', $data);

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2>Renungan Harian</h2>

<a href="/devotions/create" class="btn btn-primary mb-3">Tambah Renungan</a>

<table class="table table-bordered table-hover">

<thead class="thead-dark">

<tr>

<th>#</th>

<th>Judul</th>

<th>Isi</th>

<th>Tanggal</th>

<th>Aksi</th>

</tr>

</thead>

<tbody>

<?php foreach ($devotions as $key => $devotion): ?>

<tr>

<td><?= $key + 1 ?></td>

<td><?= $devotion['title'] ?></td>

<td><?= substr($devotion['content'], 0, 50) . '...' ?></td>

<td><?= $devotion['date'] ?></td>

<td>

<a href="/devotions/edit/<?= $devotion['id'] ?>" class="btn btn-warning btn-sm">Edit</a>

<a href="/devotions/delete/<?= $devotion['id'] ?>" class="btn btn-danger btn-sm" onclick="return confirm('Hapus renungan ini?')">Hapus</a>

</td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

</div>

<?= $this->endSection() ?>

1. Menambahkan Data
2. Controller

public function create()

{

return view('devotions/create');

}

public function store()

{

$this->devotionModel->save([

'title' => $this->request->getPost('title'),

'content' => $this->request->getPost('content'),

'date' => $this->request->getPost('date'),

]);

return redirect()->to('/devotions');

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2>Tambah Renungan</h2>

<form action="/devotions/store" method="post">

<div class="form-group">

<label>Judul</label>

<input type="text" name="title" class="form-control" required>

</div>

<div class="form-group">

<label>Isi</label>

<textarea name="content" class="form-control" rows="5" required></textarea>

</div>

<div class="form-group">

<label>Tanggal</label>

<input type="date" name="date" class="form-control" required>

</div>

<button type="submit" class="btn btn-success">Simpan</button>

<a href="/devotions" class="btn btn-secondary">Kembali</a>

</form>

</div>

<?= $this->endSection() ?>

1. Mengubah Data
2. Controller

public function edit($id)

{

$data['devotion'] = $this->devotionModel->find($id);

return view('devotions/edit', $data);

}

public function update($id)

{

$this->devotionModel->update($id, [

'title' => $this->request->getPost('title'),

'content' => $this->request->getPost('content'),

'date' => $this->request->getPost('date'),

]);

return redirect()->to('/devotions');

}

1. View

<?= $this->extend('index') ?>

<?= $this->section('content') ?>

<div class="container mt-4">

<h2>Edit Renungan</h2>

<form action="/devotions/update/<?= $devotion['id'] ?>" method="post">

<div class="form-group">

<label>Judul</label>

<input type="text" name="title" class="form-control" value="<?= $devotion['title'] ?>" required>

</div>

<div class="form-group">

<label>Isi</label>

<textarea name="content" class="form-control" rows="5" required><?= $devotion['content'] ?></textarea>

</div>

<div class="form-group">

<label>Tanggal</label>

<input type="date" name="date" class="form-control" value="<?= $devotion['date'] ?>" required>

</div>

<button type="submit" class="btn btn-success">Update</button>

<a href="/devotions" class="btn btn-secondary">Kembali</a>

</form>

</div>

<?= $this->endSection() ?>

1. Menghapus Data
2. Controller

public function delete($id)

{

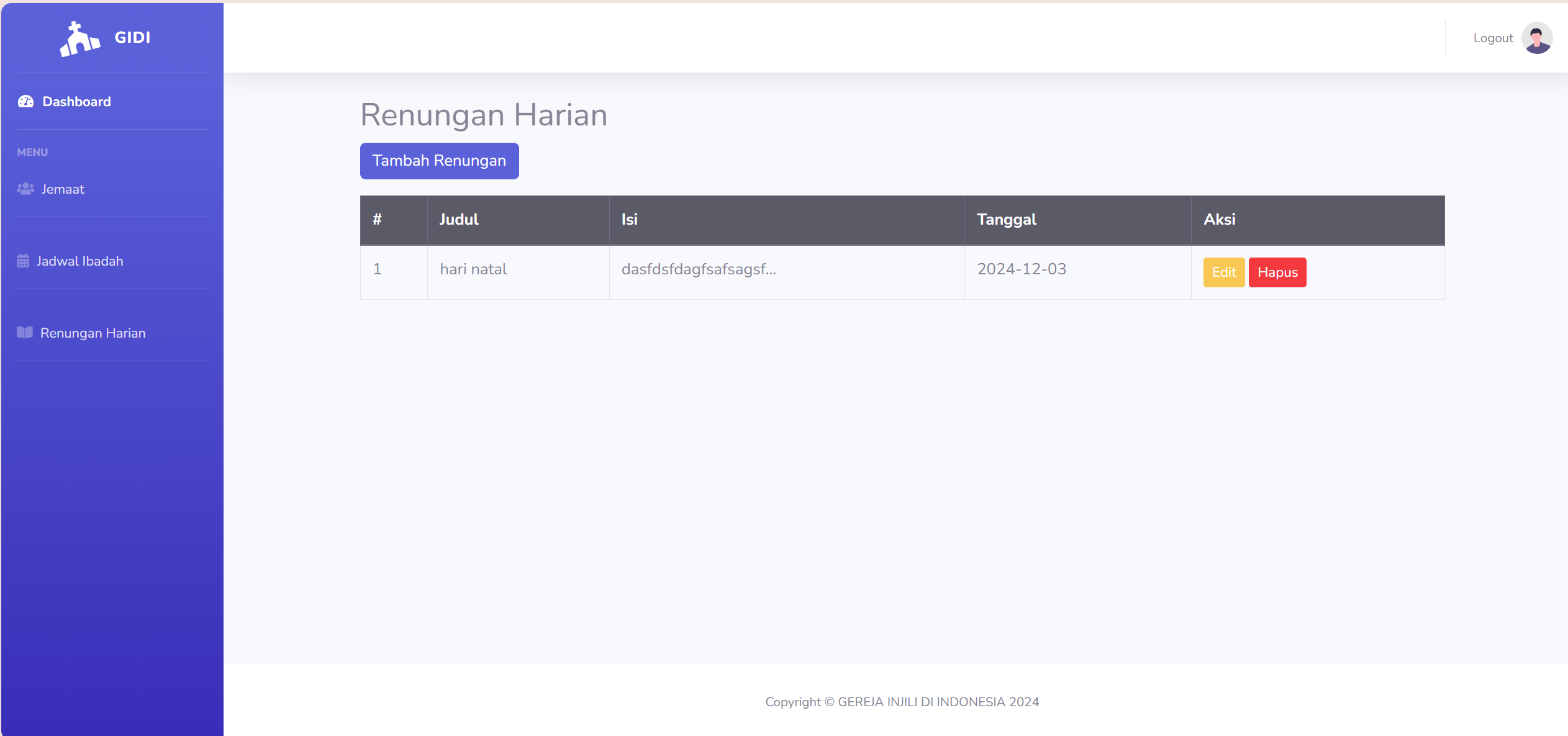
$this->devotionModel->delete($id);

return redirect()->to('/devotions');

}

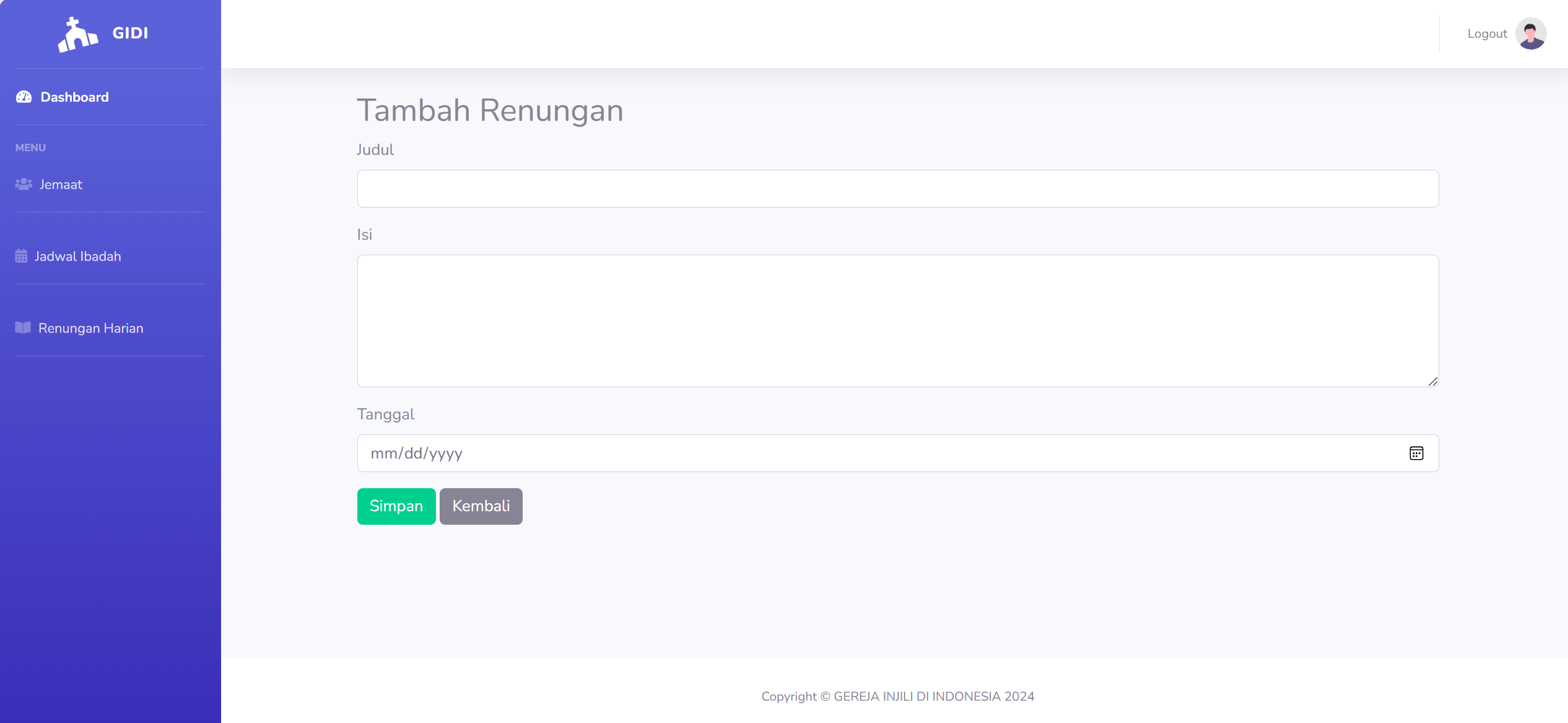
### View

1. **Index Devotion**

****

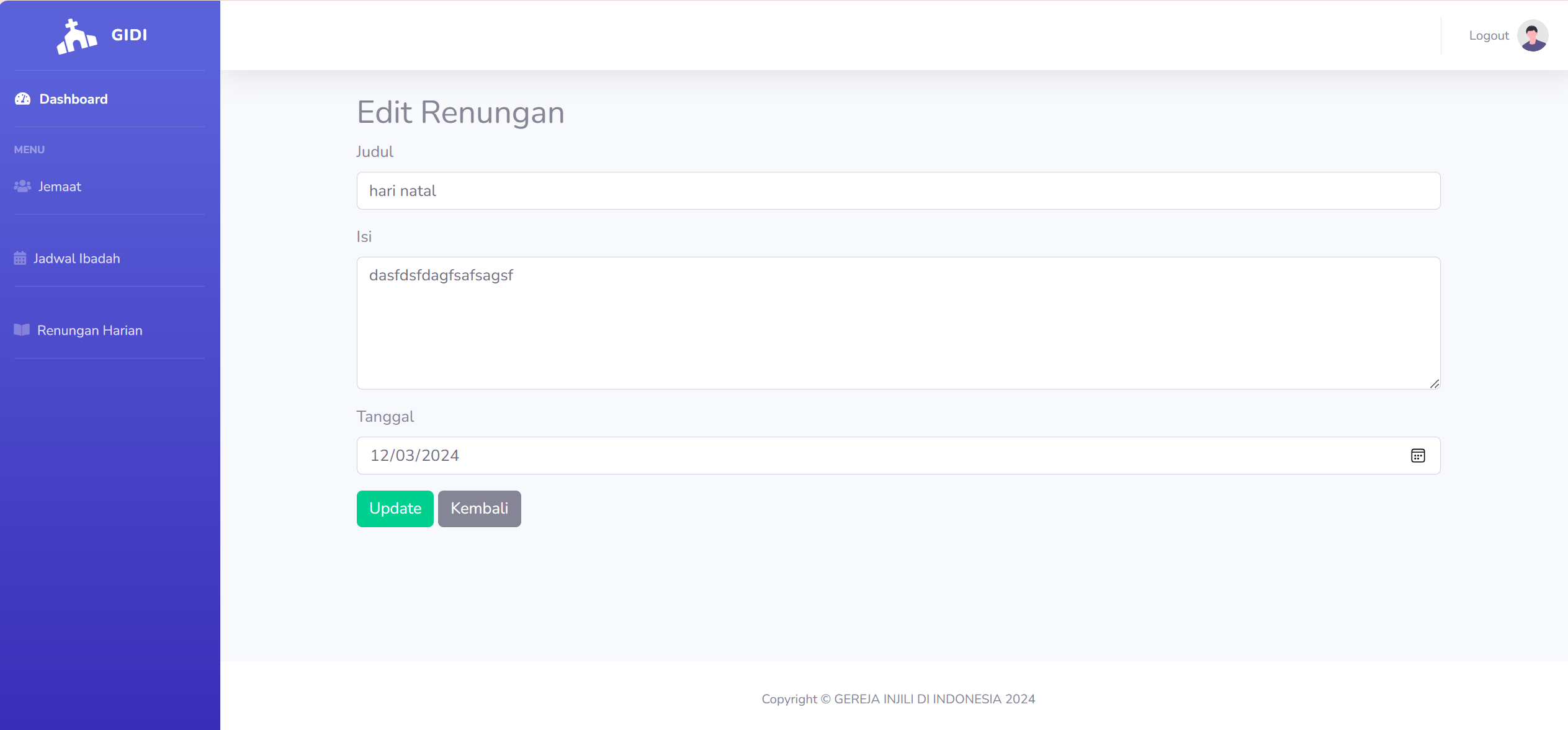
Gambar . Index Devotion

1. **Tambah Devotion**

****

Gambar . Create Devotion

1. **Ubah Devotion**

****

Gambar . Edit Devotion

# Blackbox Testing

Berisi testing dari fungsionalitas yang ada pada aplikasi berikut format tabelnya

Table . Blackbox

| **Fungsionalitas** | **Skenario Uji** | **Hasil yang diharapkan** | **Hasil Pengujian** | **Keterangan** |
| --- | --- | --- | --- | --- |
| Menambah data Jemaat | Tidak mengisi inputan apapun dan mengklik tombol simpan | Data tidak tersimpan dan muncul pesan error | Data tidak tersimpan dan muncul pesan error | Valid/Sesuai |
| Mengisi seluruh inputan dan mengklik tombol simpan | Data tersimpan dan muncul pesan berhasil | Data tersimpan dan muncul pesan berhasil | Valid/Sesuai |
| Menghapus data Jemaat | Melakukan klik pada tombol hapus yang berada pada daftar Jemaat | Data Terhapus dan tampil pesan berhasil | Data terhapus dan pesan berhasil tampil | Valid/Sesuai |
| Ubah data Jemaat | mengubah data inputan dengan mengklik tombol ubah/simpan | Data yang Tersimpan akan berubah dan tampil pesan data berhasil diperbahurui | Data yang Tersimpan akan berubah dan tampil pesan data berhasil diperbahurui | Valid/Sesuai |
| Menambah data Schedule | Tidak mengisi inputan apapun dan mengklik tombol simpan | Data tidak tersimpan dan muncul pesan error | Data tidak tersimpan dan muncul pesan error | Valid/Sesuai |
| Mengisi seluruh inputan dan mengklik tombol simpan | Data tersimpan dan muncul pesan berhasil | Data tersimpan dan muncul pesan berhasil | Valid/Sesuai |
| Menghapus data Schedule | Melakukan klik pada tombol hapus yang berada pada daftar Schedule | Data Terhapus dan tampil pesan berhasil | Data terhapus dan pesan berhasil tampil | Valid/Sesuai |
| Ubah data Schedule | mengubah data inputan dengan mengklik tombol ubah/simpan | Data yang Tersimpan akan berubah dan tampil pesan data berhasil diperbahurui | Data yang Tersimpan akan berubah dan tampil pesan data berhasil diperbahurui | Valid/Sesuai |
| Menambah data devotion | Tidak mengisi inputan apapun dan mengklik tombol simpan | Data tidak tersimpan dan muncul pesan error | Data tidak tersimpan dan muncul pesan error | Valid/Sesuai |
| Mengisi seluruh inputan dan mengklik tombol simpan | Data tersimpan dan muncul pesan berhasil | Data tersimpan dan muncul pesan berhasil | Valid/Sesuai |
| Menghapus data devotion | Melakukan klik pada tombol hapus yang berada pada daftar devotion | Data Terhapus dan tampil pesan berhasil | Data terhapus dan pesan berhasil tampil | Valid/Sesuai |
| Ubah data devotion | mengubah data inputan dengan mengklik tombol ubah/simpan | Data yang Tersimpan akan berubah dan tampil pesan data berhasil diperbahurui | Data yang Tersimpan akan berubah dan tampil pesan data berhasil diperbahurui | Valid/Sesuai |

# Daftar Pustaka

Link Repository: