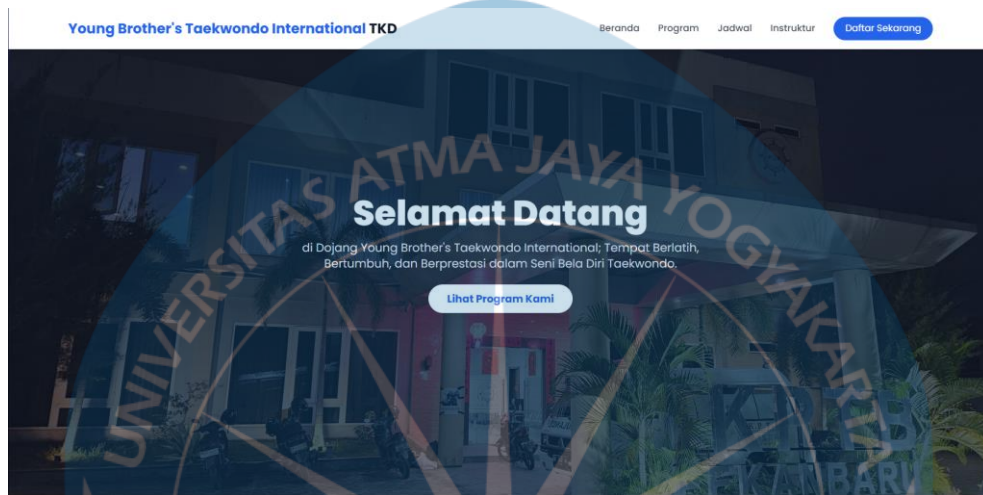


BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

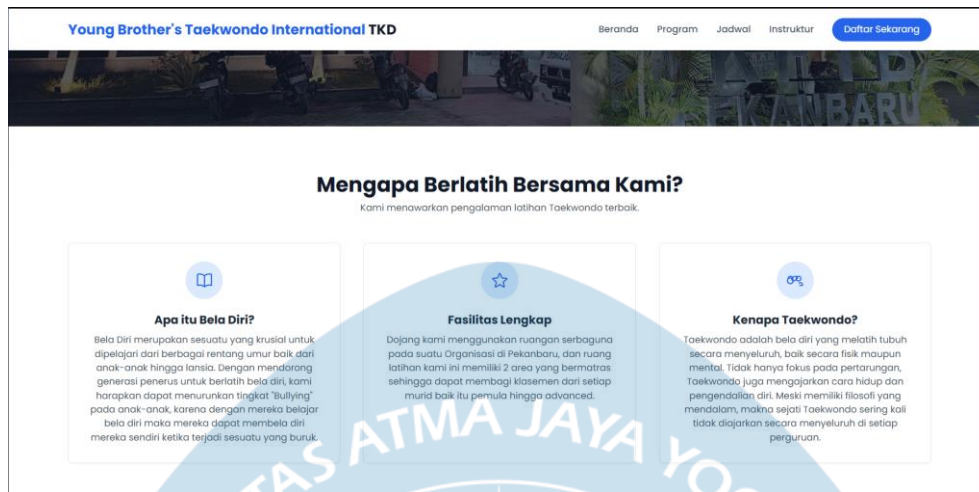
A. Implementasi UI Website

1. Implementasi Landing Page



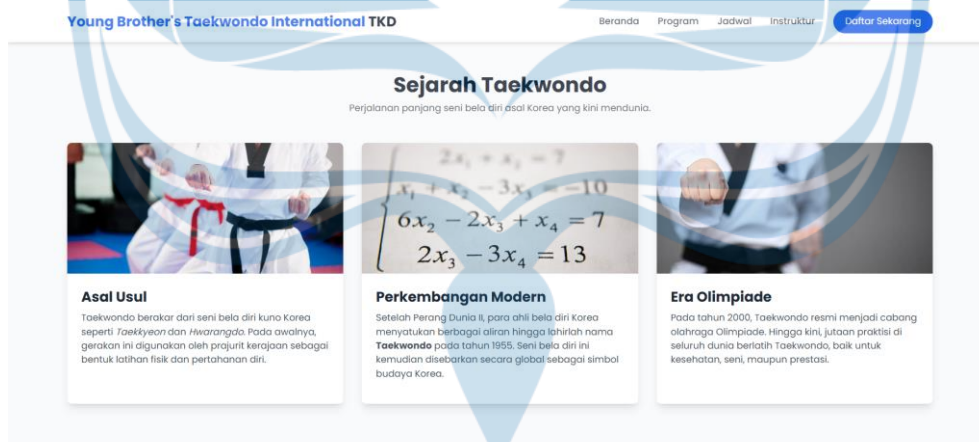
Gambar 5.1 LandingPage 1

Gambar 5.1 merupakan Landing Page pertama yang dari website Young Brother's Taekwondo International. Pengguna akan disambut dengan Welcome dan Foto tempat latihan yang digunakan pada *background*.



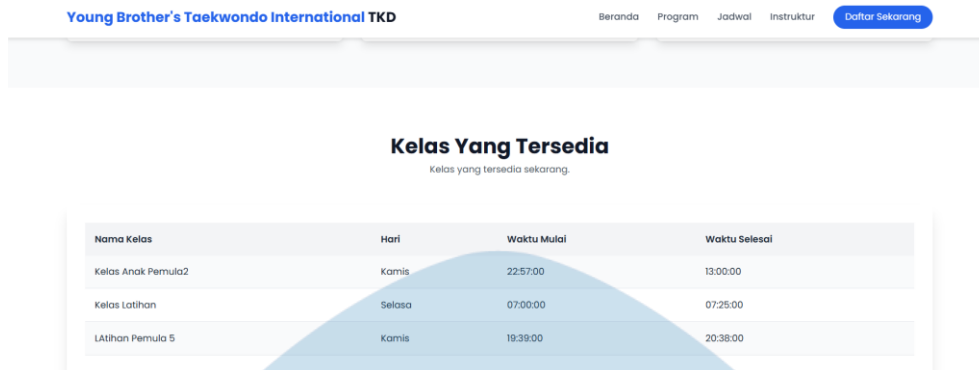
Gambar 5.2 LandingPage 2

Jika pengguna scroll kebawah, pengguna akan ditampilkan dengan contoh proses pelatihan dan penjelasan kenapa Taekwondo itu penting. Informasi yang dipaparkan merupakan hasil wawancara dari salah satu pelatih Taekwondo.



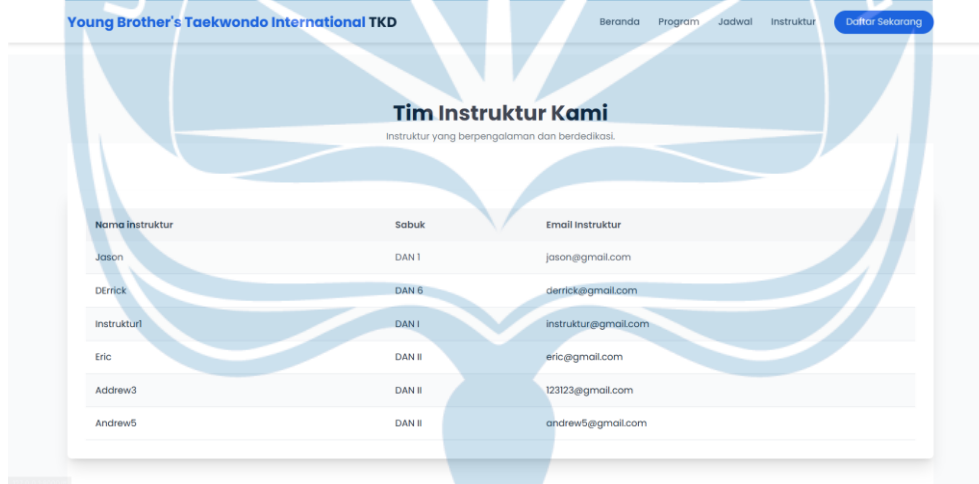
Gambar 5.3 LandingPage 3

Gambar 5.3 menampilkan sejarah, asal-usul dan perkembangan modern pada Olimpiade Taekwondo.



Gambar 5.4 LandingPage 4

Gambar 5.4 menampilkan kelas yang tersedia pada Dojang Young Brother's Taekwondo International.



Gambar 5.5 LandingPage 5

Gambar-Gambar diatas merupakan halaman LandingPage. Halaman muncul ketika User pertama kali masuk kedalam website tersebut. Dalam halaman ini terdapat list kelas dan tim instruktur yang sudah terdaftar pada Dojang melalui website.

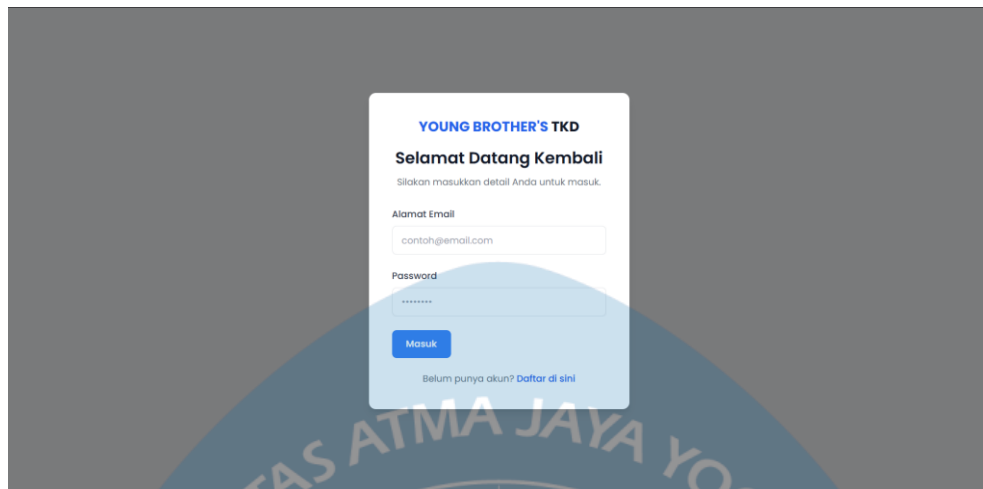
```

<body class="bg-gray-50 text-gray-800">
  <header id="header" class="bg-white shadow-md fixed top-0 left-0 right-0 z-50 transition-all duration-300">
    <div class="container mx-auto px-6 py-4 flex justify-between items-center">
      <a href="#" class="text-2xl font-bold text-gray-900">
        <span class="text-blue-600">Young Brother's Taekwondo International</span> TKD
      </a>
      <nav class="hidden md:flex space-x-8 items-center">
        <a href="#home" class="text-gray-600 hover:text-blue-600 font-medium">Beranda</a>
        <a href="#program" class="text-gray-600 hover:text-blue-600 font-medium">Program</a>
        <a href="#jadwal" class="text-gray-600 hover:text-blue-600 font-medium">Jadwal</a>
        <a href="#instruktur" class="text-gray-600 hover:text-blue-600 font-medium">Instruktur</a>
      </nav>
      @auth
        @if(Auth::user()->role == 'admin')
          <a href="{{ route('admin.dashboard') }}" class="text-gray-600 hover:text-blue-600 font-medium">Profile</a>
        @elseif(Auth::user()->role == 'instruktur')
          <a href="{{ route('instruktur.absensi.index') }}" class="text-gray-600 hover:text-blue-600 font-medium">Profile</a>
        @elseif(Auth::user()->role == 'anggota')
          <a href="{{ route('member.profile.index') }}" class="text-gray-600 hover:text-blue-600 font-medium">Profile</a>
        @endif
        <form action="{{ route('logout') }}" method="POST">
          @csrf
          <button type="submit" class="bg-red-500 text-white px-5 py-2 rounded-full hover:bg-red-600 transition duration-300">Logout</button>
        </form>
      @endauth
      @guest
        <a href="{{ route('login') }}" class="bg-blue-600 text-white px-5 py-2 rounded-full hover:bg-blue-700 transition duration-300">Daftar Sekarang</a>
      @endguest
    </div>
    <button id="mobile-menu-button" class="md:hidden">
      <svg xmlns="http://www.w3.org/2000/svg" fill="none" viewBox="0 0 24 24" stroke-width="1.5" stroke="currentColor" class="w-6 h-6">
        <path stroke-linecap="round" stroke-linejoin="round" d="M3.75 6.75 12 0 20.25 6.75" />
      </svg>
    </button>
  </div>

```

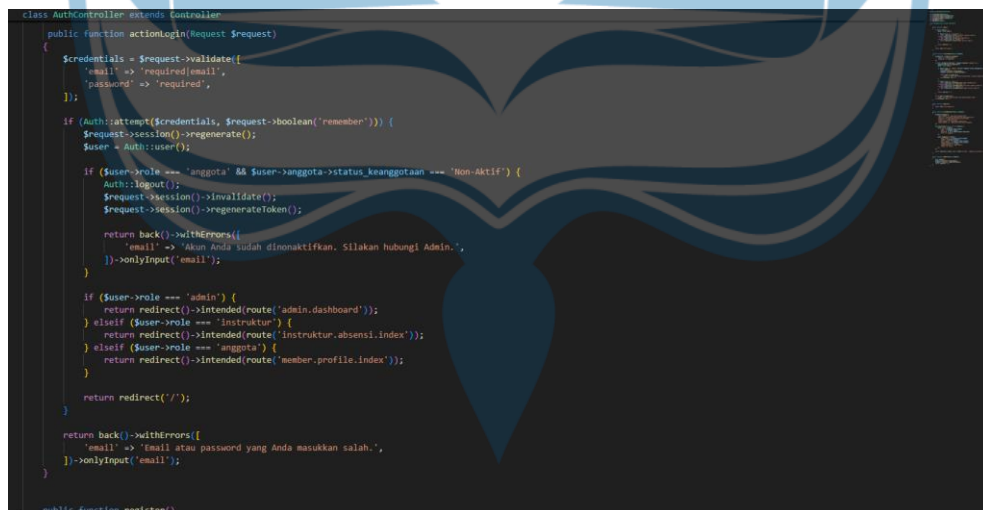
Gambar 5.6 Kode Navbar LandingPage

Gambar 5.6 merupakan Kode untuk Navbar yang terdapat di Tampilan LandingPage, jadi disini terdapat beberapa *Authentication check* dimana Website akan cek role apa yang dimiliki oleh user, jika User memiliki role instruktur, maka ketika User klik Profile, Profile tersebut akan menunjukkan instruktur Dashboard yang telah dibuat. Sama seperti Admin untuk Student tetapi Student akan dialihkan ke Profile diri dan Admin akan dialihkan ke Admin Dashboard dimana ia dapat mengakses semua fungsi Admin. Dan yang tentu kode lainnya seperti kode front-end dengan CSSnya. Tidak hanya role dari user, jika user belum autentikasi atau belum login, Navbar akan ganti Tombol Profile menjadi Login



Gambar 5.7 Tampilan LoginPage

Tombol Login akan muncul pada Navbar ketika User belum melakukan Autentikasi Login. Gambar 5.7 merupakan tampilan dari Halaman Login pada Website. Dalam login ini User dapat menggunakan email dan password yang sudah didaftarkan pada Halaman Register.



Gambar 5.8 Back-end Login

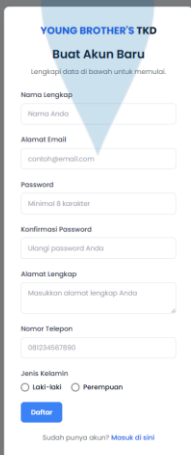
Gambar 5.8 merupakan back-end dari Halaman Login ini dan fungsi `actionLogin()`. `actionLogin()` menerima data nama dan password yang telah diinput pada form LoginPage, Website akan memvalidasi dari data tersebut apakah nama dan password yang diinput terdapat pada database. Jika tidak maka User akan mendapatkan pesan

error bahwa email atau password tersebut salah. Jika benar maka User akan dialihkan kepada Landing Page dan Tombol Login akan digantikan menjadi Profile.

Dalam Kode tersebut terdapat :

```
        If ($user->role === 'admin') {  
Return redirect()-  
>intended(route('admin.dashboard'));  
        } else if($user->role === 'instruktur') {  
Return redirect()-  
>intended(route('instruktur.absensi.index'));  
        } else if($user->role === 'anggota') {  
Return redirect()-  
>intended(route('member.profile.index'))
```

Bagian ini akan cek apakah user yang terdaftar merupakan sebuah *Admin* atau Instruktur berdasarkan role yang sudah didaftarkan sejak awal. User dengan role admin akan di hard code didalam, role instruktur yang nanti juga bisa diurus oleh admin kedepannya.



The image shows a registration form for 'YOUNG BROTHER'S TKD'. The form is titled 'Buat Akun Baru' and includes the following fields and options:

- Nama lengkap**: Text input field.
- Alamat Email**: Text input field with a placeholder 'contoh@email.com'.
- Password**: Text input field with a note 'Minimal 8 karakter'.
- Konfirmasi Password**: Text input field with a note 'Ulangi password Anda'.
- Alamat lengkap**: Text input field with a note 'Masukkan alamat lengkap Anda'.
- Nomor Telepon**: Text input field with a placeholder '081234567890'.
- Jenis Kelamin**: Radio buttons for 'Laki-laki' and 'Perempuan'.
- Daftar**: Blue button to submit the form.
- Sudah punya akun? Masuk di sini**: Link at the bottom.

Gambar 5.9 Tampilan Register Page

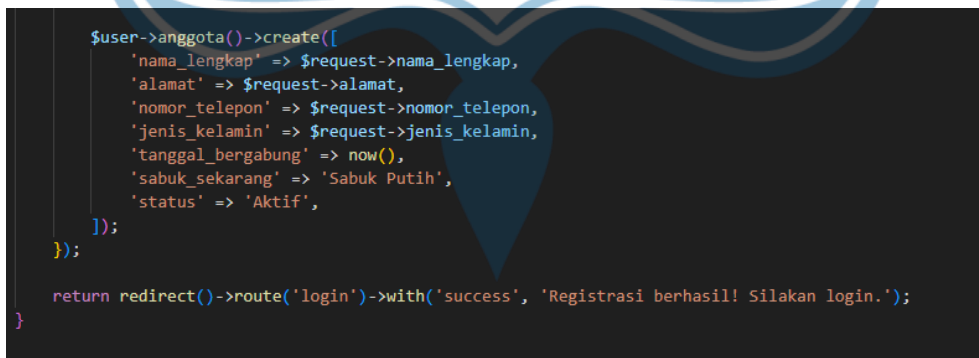
Gambar 5.9 merupakan tampilan dari Register Page dimana User dapat menginput data sesuai dengan yang dibutuhkan oleh form dari website. Register page melakukan API POST untuk menerima data yang telah diinput oleh user. Website akan memvalidasi data-data yang telah diinput



```
public function actionRegister(Request $request)
{
    $request->validate([
        'nama_lengkap' => 'required|string|max:255',
        'email' => 'required|string|email|max:255|unique:users',
        'password' => 'required|string|min:8|confirmed',
        'alamat' => 'required|string',
        'nomor_telepon' => 'required|string|max:15',
        'jenis_kelamin' => 'required|in:Laki-laki,Perempuan',
    ]);

    DB::transaction(function () use ($request) {
        $user = User::create([
            'name' => $request->nama_lengkap,
            'email' => $request->email,
            'password' => Hash::make($request->password),
            'role' => 'anggota',
        ]);
    });
}
```

Gambar 5.10 Gambar Back-end Register Bagian 1



```
$user->anggota()->create([
    'nama_lengkap' => $request->nama_lengkap,
    'alamat' => $request->alamat,
    'nomor_telepon' => $request->nomor_telepon,
    'jenis_kelamin' => $request->jenis_kelamin,
    'tanggal_bergabung' => now(),
    'sabuk_sekarang' => 'Sabuk Putih',
    'status' => 'Aktif',
]);

return redirect()->route('login')->with('success', 'Registrasi berhasil! Silakan login.');
```

Gambar 5.11 Gambar Back-end Register Bagian 2

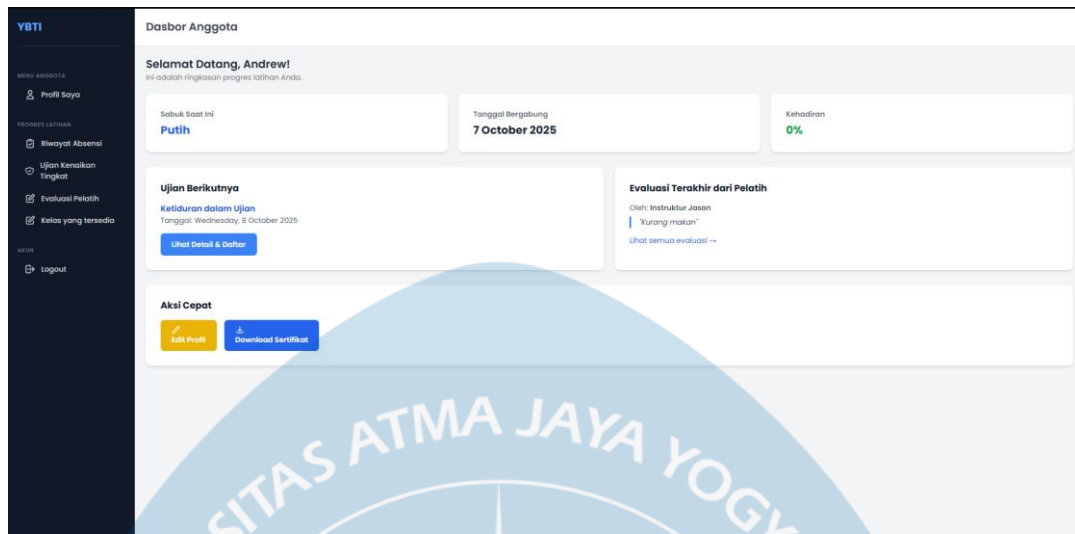
Gambar 5.10 dan Gambar 5.11 menunjukkan proses back-end yang memvalidasi data-data yang telah diinput oleh user melalui front-end. Proses validasi ini terjadi pada fungsi `actionRegister()` yang dibuat dalam `AuthController`. Sebelum data tersebut disimpan dalam Database,

Website akan melakukan proses validasi untuk memastikan bahwa data tersebut sudah sesuai dengan ketentuan yang telah ditetapkan. Validasi yang ditetapkan seperti:

1. Nama harus berupa string dan tidak boleh kosong
2. Email harus dalam format email dan harus unik sehingga tidak adanya 2 email yang sama pada database/
3. Password wajib diisi dan harus minimal memiliki panjang 8 karakter
4. nomor_telepon, jenis_kelamin dan alamat juga berupa varchar.

Setelah datanya berhasil divalidasi sistem akan membuat akun baru sesuai dengan data yang diinput di database, password yang dimasukkan oleh user akan di hash langsung dengan `Hash::make()` untuk meningkatkan keamanan data. Tanggal_bergabung dibuat untuk mencatat kapan user melakukan registrasi, tanggal_bergabung akan diisi secara langsung dengan waktu saat ini yaitu `now()`. Status akun akan secara default diatur menjadi “Aktif”.

Setelah seluruh proses pendaftaran register berhasil, sistem akan memberi notifikasi bahwa akun tersebut telah berhasil dibuat dengan `Session::flash`, User akan diarahkan kembali ke halaman Registrasi dengan notifikasi bahwa registrasi telah berhasil dan user dapat login dengan akun tersebut.



Gambar 5.12 Tampilan Profile Page dari Anggota

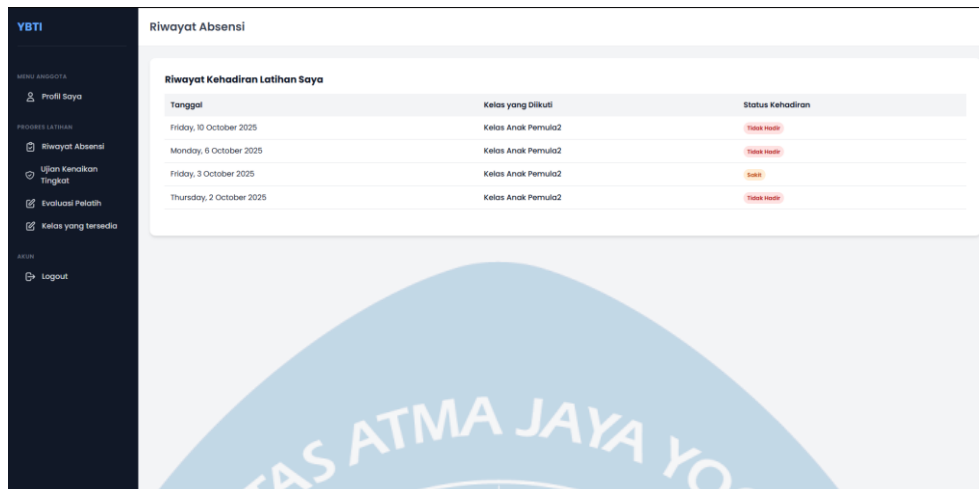
Gambar 5.12 merupakan tampilan profile user ketika user berhasil login kedalam sistem dengan role “anggota”. Tampilan ini merupakan bagian dari profile sendiri dari user yang otomatis menyesuaikan fitur-fitur yang tersedia sesuai dengan role user dalam sistem. Dalam tampilan berikut user memiliki role student atau siswa dan beberapa fitur yang dapat diakses oleh user.

1. Unduh Profil

User memiliki fitur untuk mengunduh profile dalam bentuk PDF, fitur ini memberi User untuk menyimpan data-data penting pribadi, tingkat sabuk dan catatan kegiatan yang mereka telah lakukan.

2. Lihat Kelas yang tersedia

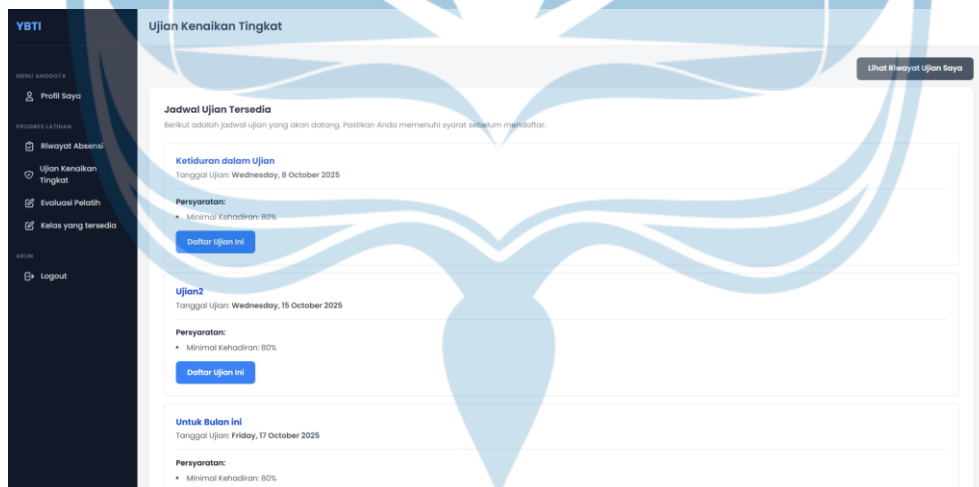
User dapat melihat jadwal Latihan atau Kelas yang tersedia pada kedepannya. Jadwal ini mencakup waktu pelaksanaan dan deskripsi kegiatan kelas yang tersedia. Fitur ini membantu untuk murid mengetahui kapan kapan harus hadir dalam latihan yang jadwalnya telah ditentukan.



Gambar 5.13 Tampilan Riwayat Absensi

3. Lihat Riwayat Absensi

Salah satu fitur yang dapat diakses oleh anggota adalah melihat Riwayat absensi untuk mengecek absensi kelas yang diikuti.



Gambar 5.14 Tampilan Ujian

4. Lihat Ujian yang tersedia

Fitur ini dapat digunakan untuk melihat ujian peningkatan sabuk yang tersedia atau dibuka, Anggota dapat mendaftarkan dirinya untuk ujian tersebut dengan menekan tombol Daftar.



Gambar 5.15 Tampilan Evaluasi dari Instruktur

5. Lihat Evaluasi dari Instruktur

Fitur ini dapat digunakan untuk melihat evaluasi yang diberikan dari Instruktur lengkap dengan tanggalnya.

Seluruh fitur yang terintegrasi dalam halaman ini profile dengan tujuan sederhana tetapi fungsional sehingga memudahkan user dalam mengakses dan mengelola data. Dengan fitur-fitur yang disediakan dalam profile ini, website tidak hanya berfungsi untuk mengolah data murid tetapi juga sebagai platform pengembangan diri dengan mencatat dan mendokumentasi progres diri.

SERTIFIKAT PELATIHAN

Dengan ini menyatakan bahwa:

Andrew

Telah berpartisipasi aktif dan menyelesaikan program pelatihan Taekwondo di **Young Brother's Taekwondo International (YBTI)** dengan tingkatan sabuk sekarang:

Sabuk Putih

Pekanbaru, 2 October 2025

29 October 2025

Jason Davento Ong
Pelatih YBTI

Gambar 5.16 Gambar Hasil Unduh Sertifikat

Gambar 5.16 merupakan hasil tampilan ketika sertifikat telah berhasil diunduh oleh user dengan unduh “Download” yang disediakan pada profile user. Fitur ini dibuat dengan tujuan supaya user memiliki bukti bahwa mereka telah berlatih dalam dojang tersebut. Ketika user menekan tombol “Download”

1. Sabuk

Dokumen tersebut menampilkan sabuk user yang terakhir kali tercapai. Ini menunjukkan bahwa sudah berapa jauh perjalanan user di dunia Taekwondo.

2. Tanda Tangan dari Pemilik YBTI

Dengan adanya ini, user dapat menggunakan hasil download ini sebagai bukti partisipasi dalam pelatihan bela diri untuk keperluan pribadi ataupun sekedar sebagai dokumentasi perkembangan diri.

```

<script>
  document.getElementById('downloadCard').addEventListener('click', function() {
    const toHide = document.querySelectorAll('.no-export, .hide-on-download');
    const exportTitle = document.getElementById('export-title');

    toHide.forEach(el => el.style.display = 'none');

    exportTitle.style.display = 'block';

    setTimeout(() => {
      html2canvas(document.getElementById('downloadContent')).then(canvas => {
        const link = document.createElement('a');
        link.download = 'ProfileSummary.png';
        link.href = canvas.toDataURL('image/png');
        link.click();
        toHide.forEach(el => el.style.display = '');
        exportTitle.style.display = 'none';
      });
    }, 100);
  });
</script>

```

Gambar 5.17 Controller untuk Download Profile

Pada Gambar 5.17 menjelaskan proses fitur download profile ketika user menekan tombol Download pada halaman profile. Proses ini dilakukan menggunakan library yang tersedia yaitu *Laravel-dompdf* dimana library tersebut mengizinkan website untuk mengubah elemen HTML dan mengubahnya menjadi PDF atau format gambar lain secara otomatis, script ini akan dijalankan ketika user menekan tombol Download.

```

public function index()
{
    $anggota = Auth::user()->anggota;

    $tanggalMulaiHitung = $anggota->tanggal_bergabung ?? now();

    // cek total hadir kelas
    $jumlahHadir = $anggota->absensi()
        ->where('status', 'Hadir')
        ->where('tanggal_absensi', '>=', $tanggalMulaiHitung)
        ->count();

    $totalSesiDiabsen = $anggota->absensi()
        ->where('tanggal_absensi', '>=', $tanggalMulaiHitung)
        ->count();

    if ($totalSesiDiabsen == 0) {
        $persentaseHadir = 0;
    } else {
        $persentaseHadir = ($jumlahHadir / $totalSesiDiabsen) * 100;
    }

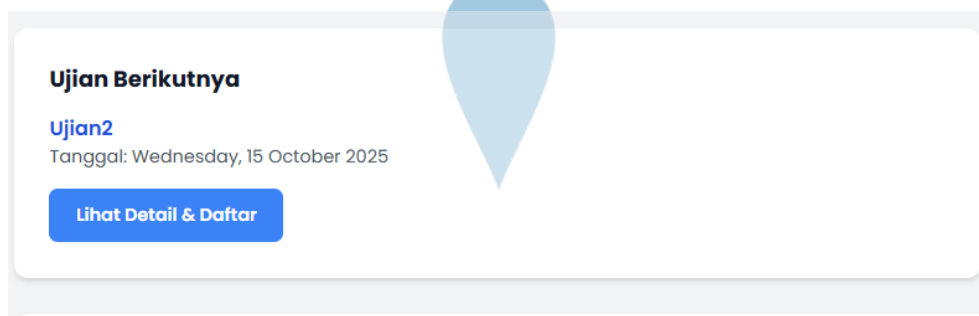
    //ambil evaluasi
    $evaluasiTerakhir = $anggota->evaluasi()->with('instruktur')->latest()->first();
    //ambil ujian berikutnya
    $ujianBerikutnya = $anggota->ujian()->where('tanggal_ujian', '>=', now())->orderBy('tanggal_ujian', 'asc')->first();

    return view('member.profile.index', compact('anggota', 'evaluasiTerakhir', 'ujianBerikutnya', 'persentaseHadir'));
}

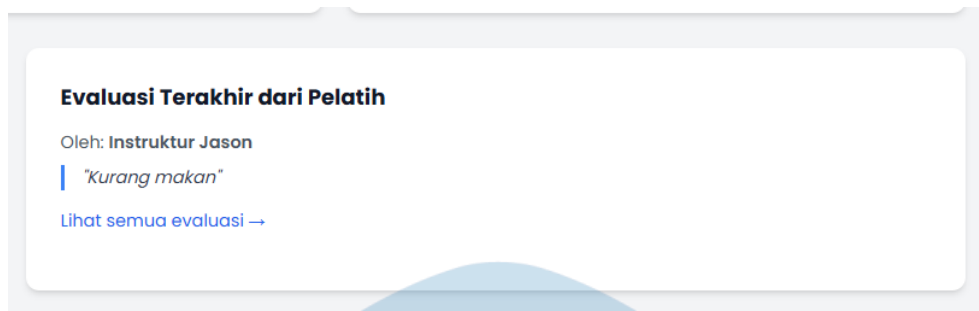
```

Gambar 5.18 Tampilan Profile Controller

Gambar 5.18 menampilkan bagian dari salah satu struktur backend website yaitu ProfileController. Fungsi dari controller ini adalah untuk penghubung utama antara View Profile dengan database yang khususnya pada Profile user atau member.profile.index. Pada kode yang ditampilkan. Controller ini bertanggung jawab untuk mengambil data user yang sedang login dengan `$users = Auth::user();` Setelah itu sistem mengambil semua data Evaluasi dan Ujian yang berhubungan dengan user yang telah login dengan query `$evaluasiTerakhir = $anggota->evaluasi()-<with('instruktur')` dan `$ujianBerikutnya = UjianKenaikanTingkat::where(kondisi);` dan diurutkan dari yang paling baru. Sistem juga mengambil seluruh data evaluasi dan ujian untuk menampilkan informasi ujian yang tersedia dengan keterangan Ujian Berikutnya. Secara keseluruhan ProfileController merupakan salah satu controller yang penting untuk profile untuk menghubungkan table user, UjianKenaikanTingkat dan Evaluasi dengan view member.profile.index.



Gambar 5.19 Tampilan Ujian Berikutnya Pada Profile

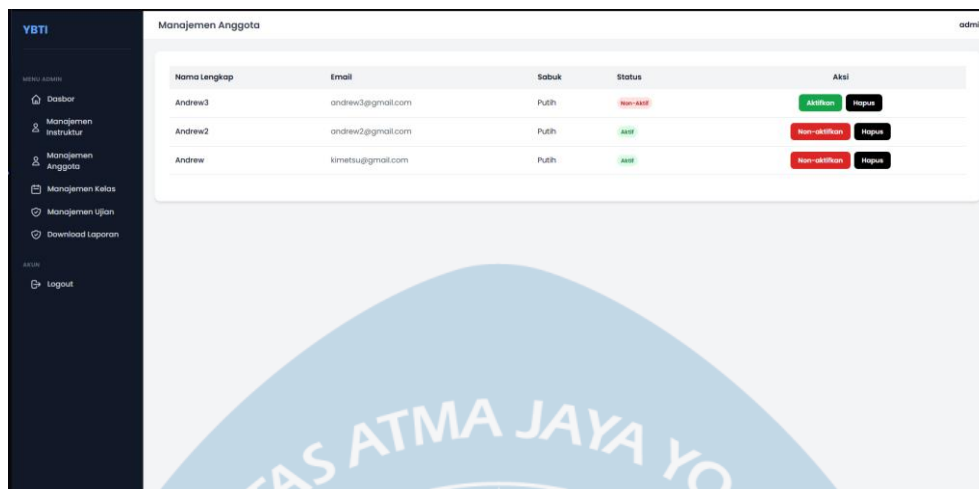


Gambar 5.20 Tampilan Evaluasi Terbaru

Gambar 5.20 menampilkan dimana member dapat melihat evaluasi terbaru dari Instruktur yang menampilkan lengkap dengan nama dan tanggalnya.



Gambar 5.21 Tampilan Front-end Admin Dashboard



Gambar 5.22 Tampilan Manajemen Anggota

Gambar 5.21 dan Gambar 5.22 Menampilkan tampilan front-end untuk Admin Dashboard dimana ketika user yang login memiliki role admin, maka user akan dialihkan ke page ini. Dashboard ini berfungsi sebagai tempat dimana Admin untuk mengelola seluruh data yang terdapat pada Website seperti contoh mengatur data murid, non-aktifkan akun murid, meningkatkan tingkat sabuk murid melalui ujian, menghapus profile murid serta melihat kelas yang pernah diadakan dengan tanggal dan murid yang berpartisipasi.

```

public function indexAdmin()
{
    $instrukturList = Instruktur::with('user')->latest()->paginate(10);
    return view('admin.management.instrukturAdmin.index', compact('instrukturList'));
}

```

Gambar 5.23 Gambar Back-end AdminPage Bagian 1

Potongan kode pada Gambar 5.23 merupakan bagian dari fungsi yang terdapat pada AdminController. Fungsi pertama yang terdapat pada Controller adalah fungsi indexAdmin. Fungsi tersebut bertanggung jawab untuk mengambil semua data dari table Anggota,

```

public function store(Request $request)
{
    $request->validate([
        'nama_lengkap' => 'required|string|max:255',
        'email' => 'required|string|email|max:255|unique:users',
        'password' => 'required|string|min:8|confirmed',
        'tingkat_dan' => 'required|string|max:255',
    ]);

    DB::transaction(function () use ($request) {
        $user = User::create([
            'name' => $request->nama_lengkap,
            'email' => $request->email,
            'password' => Hash::make($request->password),
            'role' => 'instruktur', // untuk instruktur saja
        ]);

        $user->instruktur()->create([
            'nama_lengkap' => $request->nama_lengkap,
            'tingkat_dan' => $request->tingkat_dan,
        ]);
    });

    return redirect()->route('admin.instruktur.index')->with('success', 'Instruktur baru berhasil ditambahkan.');
```

Gambar 5.24 Gambar Back-end AdminPage Bagian 2

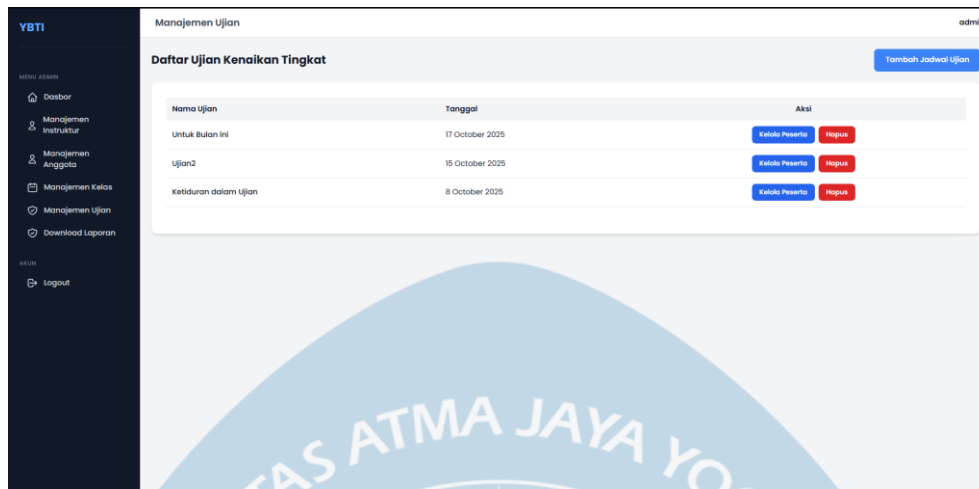
Fungsi kedua pada back-end di AdminController adalah store(). Fungsi ini digunakan menyimpan data instruktur yang ditambah oleh Admin. Formulir yang dibutuhkan untuk menambahkan instruktur baru adalah nama_lengkap, email, password, role instruktur yang dihubungkan kepada table User, kemudian dihubungkan kembali pada tabel Instruktur.

```

public function destroy(User $user)
{
    if ($user->role !== 'instruktur') {
        return redirect()->route('admin.instruktur.index')->with('error', 'Hanya instruktur yang dapat dihapus.');
```

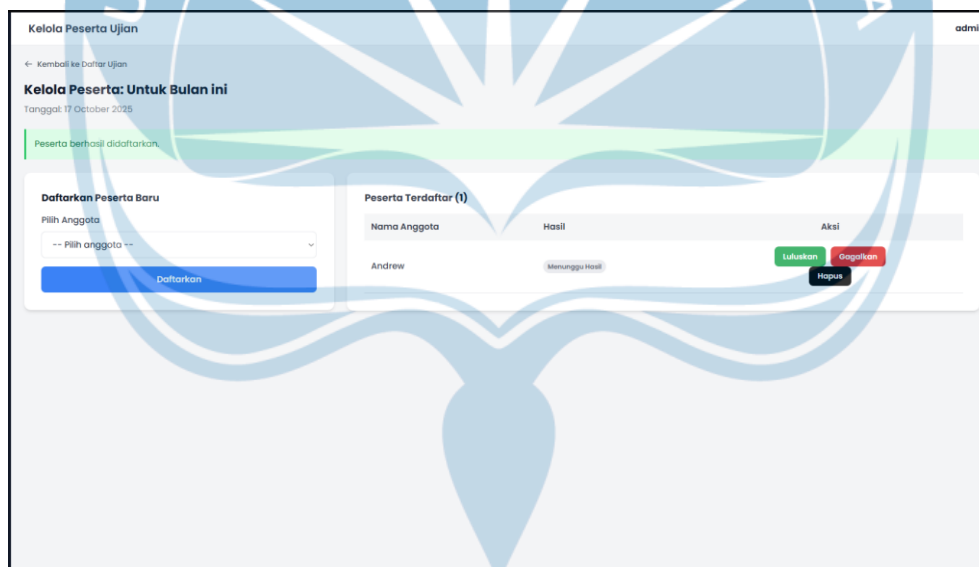
Gambar 5.25 Gambar Back-end AdminPage Bagian 3

Fungsi Destroy adalah fungsi dimana sistem akan menghapus data instruktur yang berada dalam tabel Instruktur. Jika data instruktur tersebut ditemukan, maka akan melakukan proses *delete* sesuai dengan idnya.



Gambar 5.26 Tampilan Daftar Ujian Yang Tersedia

Gambar 5.26 merupakan tampilan dari total ujian yang sudah diselenggarakan.



Gambar 5.27 Tampilan Daftar Peserta Ujian

Gambar 5.27 merupakan tampilan dari dashboard daftar peserta ujian yang telah terdaftar

```

public function show(UjianKenaikanTingkat $ujian)
{
    $pendaftaranList = $ujian->pendaftaran()->with('anggota')->get();

    $anggotaTersedia = Anggota::where('status_keanggotaan', 'Aktif')->whereHas('user', function ($query) { $query->where('role', 'anggota'); })->whereDoesntHave('pendaftaranUjian', function ($query) { $query->where('id_ujian', $ujian->id_ujian); })->orderBy('nama_lengkap')->get();

    return view('admin.management.ujianAdmin.show', compact('ujian', 'pendaftaranList', 'anggotaTersedia'));
}

public function daftarkanPeserta(Request $request, UjianKenaikanTingkat $ujian)
{
    $request->validate([
        'id_anggota' => 'required|exists:anggota,id_anggota'
    ]);

    $sudahDaftar = PendaftaranUjian::where('id_anggota', $request->id_anggota)
        ->where('id_ujian', $ujian->id_ujian)
        ->exists();

    if ($sudahDaftar) {
        return back()->with('error', 'Anggota tersebut sudah terdaftar pada ujian ini.');
```

Gambar 5.28 Back-end Proses Menampilkan Ujian

Gambar 5.28 *back-end* proses dimana tampilan ujian akan ditampilkan pada fungsi show dan daftarkan peserta adalah fungsi untuk mendaftarkan member tersebut dalam ujian.

```

public function updateHasil(Request $request, PendaftaranUjian $pendaftaran)
{
    // dd($request->all());
    $request->validate(['hasil_ujian' => 'required|in:Lulus,Gagal']);

    $pendaftaran->update(['hasil_ujian' => $request->hasil_ujian]);

    if ($request->hasil_ujian == 'Lulus') {
        $anggota = $pendaftaran->anggota;
        $sabukBerikutnya = $this->getSabukBerikutnya($anggota->sabuk_saat_ini);

        $anggota->update([
            'sabuk_saat_ini' => $sabukBerikutnya,
            'tanggal_ujian_terakhir' => $pendaftaran->ujian->tanggal_ujian_terakhir,
        ]);
    }

    return back()->with('success', 'Hasil ujian anggota berhasil diperbarui.');
```

```

private function getSabukBerikutnya($sabukSekarang)
{
    $urutanSabuk = [
        'Putih' => 'Kuning',
        'Kuning' => 'Kuning Strip',
        'Kuning Strip' => 'Hijau',
        'Hijau' => 'Hijau Strip',
        'Hijau Strip' => 'Biru',
        'Biru' => 'Biru Strip',
        'Biru Strip' => 'Merah',
        'Merah' => 'Merah Strip 1',
        'Merah Strip 1' => 'Merah Strip 2',
        'Merah Strip 2' => 'DAN I',
    ];

    return $urutanSabuk[$sabukSekarang] ?? $sabukSekarang;
}
```

Gambar 5.29 Back-end Proses Kenaikan Sabuk

Tambah Kelas Baru

Formulir Kelas Baru

Nama Kelas	Instruktur Penanggung Jawab	
Contoh: Kelas Anak Pemula	Pilih Instruktur	
Hari Latihan	Waktu Mulai	Waktu Selesai
Selasa	08:00 - 09:00	09:00 - 10:00
Simpan Kelas	Batal	

Gambar 5.30 Tampilan Fitur Tambah Kelas

Gambar 5.30 merupakan tampilan modal untuk tambah kelas baru, tampilan ini akan muncul ketika user menekan button Tambah Kelas. Modal tersebut ditampilkan dengan form yang akan diisi oleh user dengan kegiatan kelas dan pada hari apa kelas akan diadakan. Setelah user menginput data, Sistem akan mengirim sebuah *request* kepada backend untuk memvalidasi data yang telah diinput. Data yang divalidasi adalah nama kelas harus string, day harus Selasa Kamis Sabtu, memilih instruktur dengan *dropdown* dan waktu mulai dan selesai dengan *dropdown*.

Isi Absensi

Absensi: Kelas Anak Pemula2
Jadwal: Kamis, 22:57

Tanggal Absensi
10/08/2025

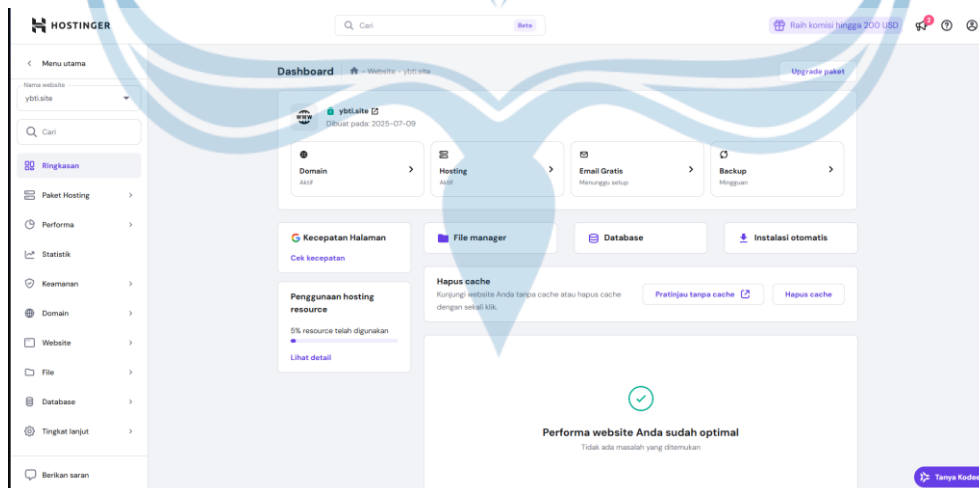
Nama Anggota	Status Kehadiran
Andrew	☒ Hadir ☐ Izin ☐ Sakit ☐ Tidak Hadir
Andrew2	☒ Hadir ☐ Izin ☐ Sakit ☐ Tidak Hadir

[Simpan Absensi](#) [Batal](#)

Gambar 5.31 Tampilan view Absensi

Gambar 5.31 merupakan tampilan jika instruktur telah melakukan presensi untuk murid, tombol presensi hanya bisa di tekan oleh user untuk sekali. Setelah ditekan, data-data berupa `id_kelas`, `id_user` dan tanggal absensi akan disimpan didalam tabel Absensi.

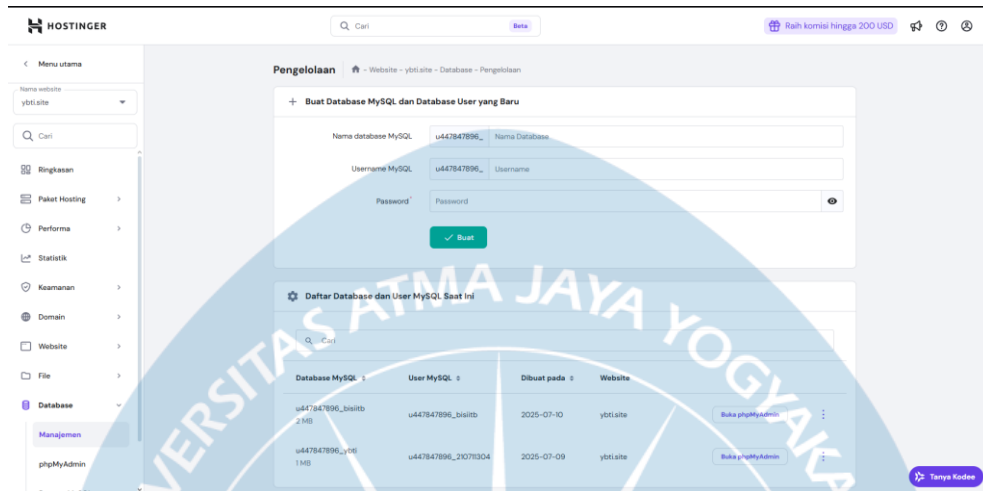
B. Implementasi Hosting Website



Gambar 5.32 Tampilan Hostinger untuk Website

Tujuan dari pembuatan website ini adalah untuk mempromosikan bela diri yang terdapat pada Pekanbaru dan memberi Dojang Young Brother's

Taekwondo International sebuah wadah untuk mengelola data-data yang diperlukan untuk murid secara mudah.



Gambar 5.33 Tampilan Database Website di Hostinger

Gambar 5.33 Menampilkan tampilan halaman pengelolaan data base untuk website yang dihosting pada platform Hostinger. User dapat membuat database baru dan mengelolanya.

C. Pengujian Fungsionalitas Perangkat Lunak

Pengujian ini dilakukan dengan metode *Black box testing* dimana yang diharapkan dari test ini merupakan hasil atau output dari aplikasi apakah sesuai dengan yang dijelaskan atau ekspektasi. Alasan untuk menggunakan metode *Black box* dikarenakan fokus pengujian berada pada fungsionalitas, sehingga tidak mengetahui kode program, hanya dilakukan dari sudut pandang pengguna untuk memastikan bahwa fitur sudah berjalan sudah sesuai dengan kebutuhan dan keinginannya Tabel 5.1 berikut akan menunjukkan hasil pengujian

Table 5.1 Tabel Hasil Pengujian Website

No.	Deskripsi	Prosedur Pengujian	Masukan	Hasil yang Diinginkan	Hasil yang Didapatkan	Kesimpulan
1	Pengujian Register	1. Memasukkan data-data yang dibutuhkan untuk register seperti nama, email, password, nomor telepon, alamat, jenis kelamin. 2. Menekan tombol <i>Register</i>	1. Username: Andrew 2. Email: andrew@gmail.co 3. Password: 123123 4. No. Telp: 123123 5. Alamat: Jalan Kemarin 6. Jenis Kelamin: Laki-laki	Pengguna berhasil register dan dapat melakukan login	Pengguna berhasil register dan dapat melakukan login	Sesuai
2	Pengujian Login	1. Memasukkan data yang diminta seperti Name dan Password 2. Menekan tombol Login	1. Name: Andrew 2. Password:123123 3. Menekan tombol Login	Pengguna akan diarahkan ke LandingPage dengan tombol Login digantikan dengan Profile	Pengguna akan diarahkan ke LandingPage dengan tombol Login digantikan dengan Profile	Sesuai
3	Pengujian navigasi Landing Page	Menekan tulisan “Young Brother’s Taekwondo International” dan menavigasi sekitarnya.	Tidak ada masukan	Penguna diarahkan ke landing page dan dapat menavigasinya.	Penguna diarahkan ke landing page dan dapat menavigasinya.	Sesuai
4	Pengujian melakukan <i>edit</i> pada Profile	1. Menekan tombol Tambah Edit Profile di Profile 2. Menggantikan anny sesuai dengan keinginan 3. Menekan tombol	1. Nama: Andrew2 2. Nomor Telepon: 0888 3. Alamat Lengkap: Jalan Hari ini 4. Jenis Kelamin: Laki-laki	Pengguna dapat <i>edit</i> profilnya sesuai dengan keinginan	Pengguna dapat <i>edit</i> profilnya sesuai dengan keinginan	Sesuai

No.	Deskripsi	Prosedur Pengujian	Masukan	Hasil yang Diinginkan	Hasil yang Didapatkan	Kesimpulan
		Simpan Perubahan				
5.	Pengujian melakukan pendaftaran mandiri dalam Ujian	1. Menekan tombol “Daftar Ujian Ini”	Tidak ada masukan	Pengguna dapat mendaftarkan dirinya dalam ujian, jika syarat ujian telah terpenuhi.	Pengguna dapat mendaftarkan dirinya dalam ujian, jika syarat ujian telah terpenuhi.	Sesuai
6.	Pengujian melakukan pendaftaran pada kelas yang tersedia.	1. Menekan tombol “Daftar Kelas ini	Tidak ada masukan	Pengguna dapat mendaftarkan dirinya dalam kelas yang diinginkan.	Pengguna dapat mendaftarkan dirinya dalam kelas yang diinginkan.	Sesuai
7.	Pengujian logout pada profile	1. Menekan tombol Logout pada profile	Tidak ada masukan	Pengguna dapat logout dari profile.	Pengguna dapat logout dari profile	Sesuai
8.	Pengujian mengisi absensi	1. Menekan tombol “Isi Absensi” pada kelas yang diinginkan. 2. Tanggal Absensi dan Status Kehadiran disesuaikan.	Tidak ada masukan	Pengguna dapat mengisi absensi pada kelas yang diinginkan.	Pengguna dapat mengisi absensi pada kelas yang diinginkan.	Sesuai
9.	Pengujian mengisi evaluasi untuk murid yang diinginkan.	1. Menekan tombol “Lihat Detail & Evaluasi” pada murid yang diinginkan.	1. Evaluasi: Tendangannya kurang kuat.	Pengguna sukses mengisi evaluasi untuk murid yang diinginkan.	Pengguna sukses mengisi evaluasi untuk murid yang diinginkan.	Sukses
10.	Pengujian menambah instruktur.	1. Menekan tombol “Tambah Instruktur”	1. Nama Lengkap: Derrick 2. Password: 123123123 3. Alamat Email: derrick@gmail.com	Pengguna sukses menambahkan Instruktur baru.	Pengguna sukses menambahkan Instruktur baru.	Sukses
11.	Pengujian menghapus	1. Menekan tombol	Tidak ada masukan	Pengguna sukses	Pengguna sukses	Sukses

No.	Deskripsi	Prosedur Pengujian	Masukan	Hasil yang Diinginkan	Hasil yang Didapatkan	Kesimpulan
	data instruktur yang diinginkan	Hapus pada Instruktur yang diinginkan.		menghapuskan data instruktur yang diinginkan.	menghapuskan data instruktur yang diinginkan.	
12.	Pengujian pengelolaan data anggota.	1. Menekan tombol Aktifkan atau Non-aktifkan jika ingin mengubah status anggota. 2. Menekan tombol hapus jika ingin menghapus data anggota.	Tidak ada masukan	Pengguna sukses mengelola data anggota.	Pengguna sukses mengelola data anggota.	Sukses
13.	Pengujian menambah kelas baru.	1. Menekan tombol Tambah Kelas Baru 2. Mengisi formulir sesuai dengan data yang diinginkan. 3. Menekan tombol Simpan Kelas.	1. Nama Kelas: Ketiduran. 2. Hari Latihan: Selasa. 3. Pilih Instruktur: Derrick 4. Waktu Mulai: 07.45 PM. 5. Waktu Selesai: 08.46 PM.	Pengguna sukses menambah kelas baru.	Pengguna sukses menambah kelas baru.	Sesuai
14.	Pengujian menghapus kelas yang diinginkan.	1. Menekan tombol Hapus pada kelas yang ingin dihapus.	Tidak ada masukan	Pengguna sukses menghapus data kelas.	Pengguna sukses menghapus data kelas.	Sesuai
15.	Pengujian menambah ujian baru	1. Menekan tombol Tambah Ujian	1. Nama Ujian: Ujian1. 2. Tanggal Pelaksanaan: 6 Oktober 2025	Pengguna dapat menambah ujian baru.	Pengguna dapat menambah ujian baru.	Sesuai

No.	Deskripsi	Prosedur Pengujian	Masukan	Hasil yang Diinginkan	Hasil yang Didapatkan	Kesimpulan
		2. Mengisi data Ujian sesuai dengan formulir. 3. Menekan tombol Simpan Jadwal Ujian				
16.	Pengujian menghapus ujian yang diinginkan	1. Menekan tombol Hapus pada ujian yang diinginkan.	Tidak ada masukan	Pengguna dapat menghapus data ujian.	Pengguna dapat menghapus data ujian.	Sesuai
17.	Pengujian menambah instruktur dalam sebuah kelas	1. Menekan tombol icon tambah instruktur 2. Menekan nama instruktur yang ingin dimasukkan 3. Menekan tombol simpan	Tidak ada masukan	Pengguna dapat menambah instruktur dalam sebuah kelas yang diinginkan	Pengguna dapat menambah instruktur dalam sebuah kelas yang diinginkan	Sesuai
18.	Pengujian menambah murid dalam sebuah kelas	1. Menekan tombol tambah murid 2. Menekan nama murid yang ingin dimasukkan 3. Menekan tombol simpan	Tidak ada masukan	Pengguna dapat menambah murid dalam sebuah kelas yang diinginkan	Pengguna dapat menambah murid dalam sebuah kelas yang diinginkan	Sesuai
19.	Pengujian mengelola data peserta dalam ujian.	1. Menekan tombol Kelola Peserta. 2. Memilih anggota yang ingin	Tidak ada masukan	Pengguna dapat mengelola data peserta ujian.	Pengguna dapat mengelola data peserta ujian.	Sesuai

No.	Deskripsi	Prosedur Pengujian	Masukan	Hasil yang Diinginkan	Hasil yang Didapatkan	Kesimpulan
		dimasukkan dengan <i>dropdown</i> . 3. Peserta dapat Meluluskan ,Gagalkan dan Hapus data peserta dari Ujian.				



D. Hasil Pengujian Terhadap Pengguna

Kuesioner ini dibuat sesuai dengan format *User Acceptance Testing* dengan tujuan untuk menguji apakah fitur yang disediakan dan fungsinya sudah berjalan sesuai dengan deskripsi dan apakah tampilan sudah menarik dan nyaman digunakan oleh pengguna.

Pengujian atau *User Testing* dilakukan dengan membeirkan kuesioner dan sekaligus memantau secara daring ataupun chat langsung. Kuesioner ini dibagikan kepada orang yang memiliki latar belakang dari Pekanbaru maupun dari daerah lain. Pengujian ini dilakukan dengan 3 akun yang berbeda yaitu akun dengan role admin, instruktur dan akun sendiri yang harus dilakukan dengan register sendiri. Responden telah disebarkan kepada total 31 Responden. Kuesioner berikut memiliki 3 tahap yaitu tahap testing Admin, tahap Testing Guru dan tahap Testing User dimana responden diminta untuk melakukan semuanya. Pertanyaan yang terdapat pada kuesioner apat dijawab dengan bobot atau penilaian sebagai berikut:

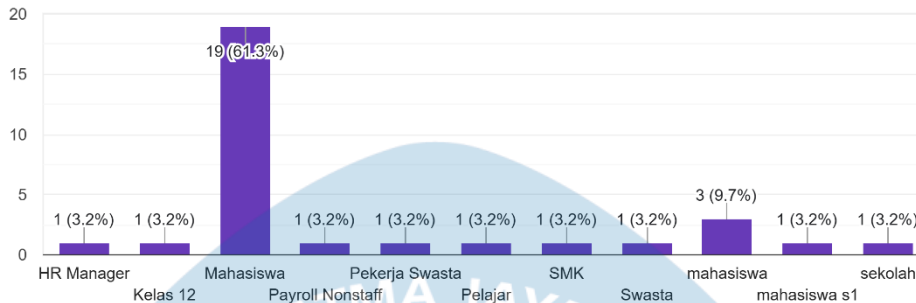
1. Sangat Setuju (SS)
2. Setuju (S)
3. Sedang (SE)
4. Tidak Setuju (TS)
5. Sangat Tidak Setuju (ST)

Tabel 5.2 Hasil Pengujian Terhadap Responden

Pertanyaan	Opsi				
	ST	TS	SE	S	SS
Apakah Anda dapat menavigasi website dengan mudah?			5	14	12
Apakah deskripsi untuk setiap bela diri mudah dimengerti?			4	14	13
Menurut Anda, apakah tampilan website menarik untuk mata Anda?			5	17	9
Apakah fitur-fitur yang disediakan sudah sesuai untuk Website Bela Diri sebagai murid?			1	15	15
Seberapa puas Anda terhadap fitur-fitur User yang tersedia di dalam website bela diri ini?			1	17	13
Apakah fitur-fitur yang disediakan sudah sesuai untuk Website Bela Diri sebagai Admin?			1	18	12
Seberapa puas Anda terhadap fitur-fitur Admin yang tersedia di dalam website bela diri ini?			2	17	12
Apakah fitur-fitur yang disediakan sudah sesuai untuk Website Bela Diri sebagai Instruktur atau Guru?			2	19	10
Seberapa puas Anda terhadap fitur-fitur Instruktur yang tersedia di dalam website bela diri ini?			3	19	9

Pekerjaan

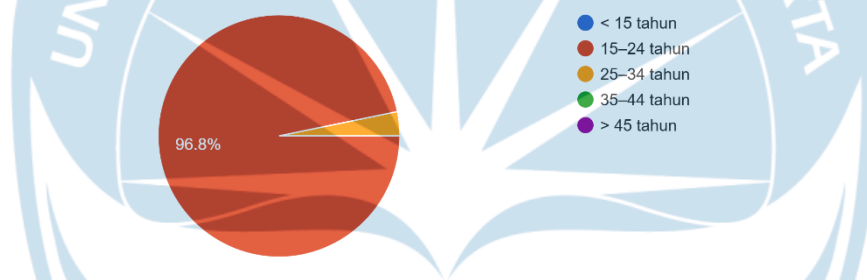
31 responses



Gambar 5.34 Persentase Pekerjaan dari Responden

Usia

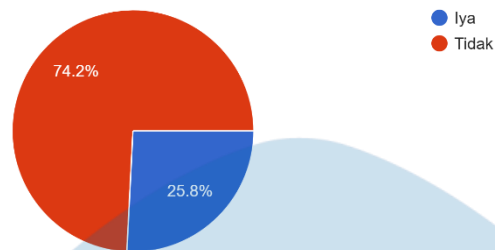
31 responses



Gambar 5.35 Rentang Usia Responden

Gambar 5.34 dan Gambar 5.35 menunjukkan bahwa rata-rata umur responden yang telah mengisi kuesioner ini berada 15-24 tahun, dengan persentase hanya satu yang mengisi 25-34 tahun, menunjukkan bahwa yang mengisi responden rata-rata merupakan remaja, dengan gambar 5.36 menunjukkan dengan yang mengisi responden rata-rata merupakan mahasiswa.

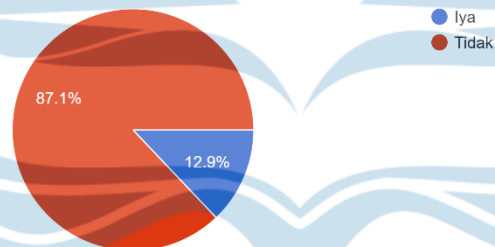
Apakah Anda memiliki pengalaman dalam berlatih bela diri?
31 responses



Gambar 5.36 Tampilan Responden Berpengalaman Bela Diri

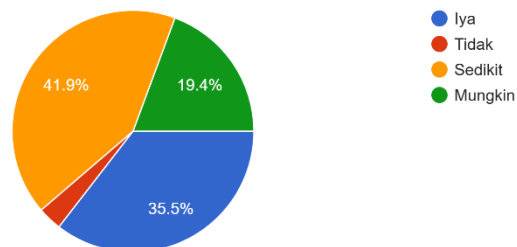
Dengan gambar chart ini menunjukkan latar belakang 74.2% responden tidak memiliki pengalaman dalam bela diri dan 25.8% responden memiliki pengalaman dalam bela diri.

Apakah Anda menekuni latihan bela diri?
31 responses



Gambar 5.37 Gambar Penekunan Bela Diri Responden

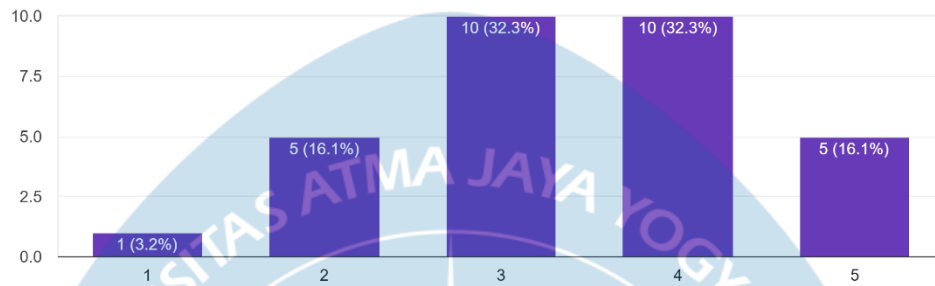
Apakah Anda mengerti apa yang dimaksud dengan seni bela diri?
31 responses



Gambar 5.38 Gambar Pengertian Responden Terhadap Bela Diri

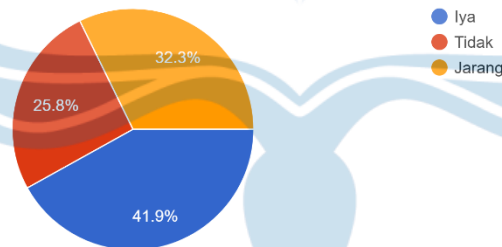
Gambar 5.44 dan Gambar 5.45 merupakan pertanyaan umum untuk mengukur seberapa paham penduduk Pekanbaru terhadap Bela Diri.

Seberapa sering Anda melihat atau mendengar informasi tentang bela diri?
31 responses



Gambar 5.39 Gambar Frekuensi Responden dalam Mendengar tentang Bela Diri

Apakah Anda tahu ada perguruan bela diri di Pekanbaru?
31 responses

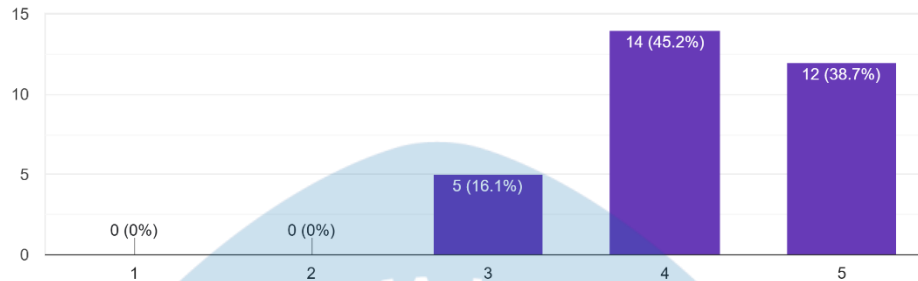


Gambar 5.40 Tingkat Pengetahuan Responden tentang Perguruan Bela Diri di Pekanbaru

Gambar 5.39 dan Gambar 5.40 menggambarkan seberapa banyak responden mendengarkan berita ataupun informasi mengenai bela diri, informasi seperti apakah ada perguruan bela diri di Pekanbaru dan informasi lainnya. Berikut Kuesioner akan melanjutkan ke tahap pertanyaan mengenai *user-acceptance* terhadap website

Apakah Anda dapat menavigasi website dengan mudah?

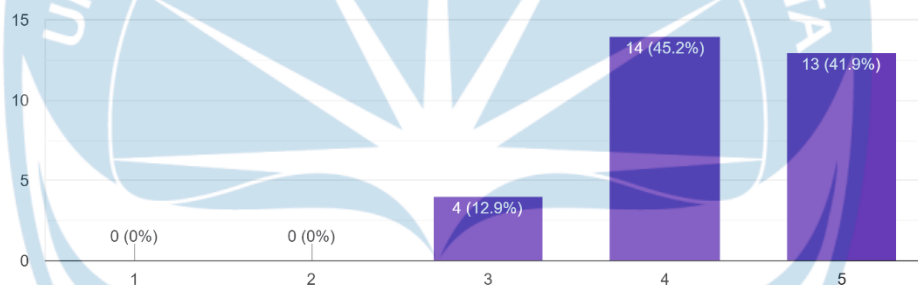
31 responses



Gambar 5.41 Pertanyaan Mengenai Tampilan Website Bagian Pertama

Apakah deskripsi untuk setiap bela diri mudah dimengerti?

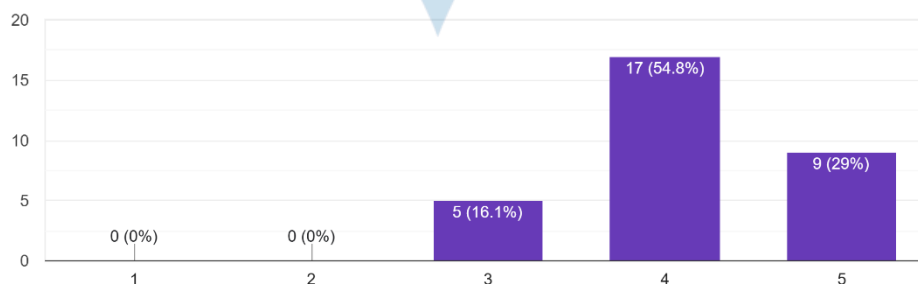
31 responses



Gambar 5.42 Pertanyaan Mengenai Tampilan Website Bagian Kedua

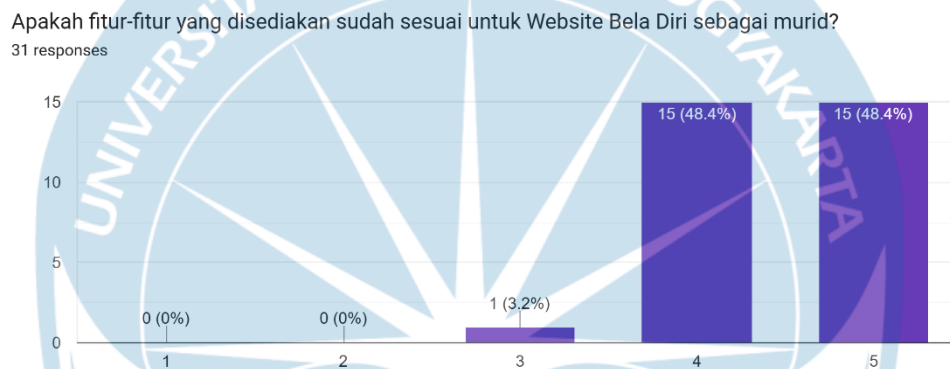
Menurut Anda, apakah tampilan website menarik untuk mata Anda?

31 responses

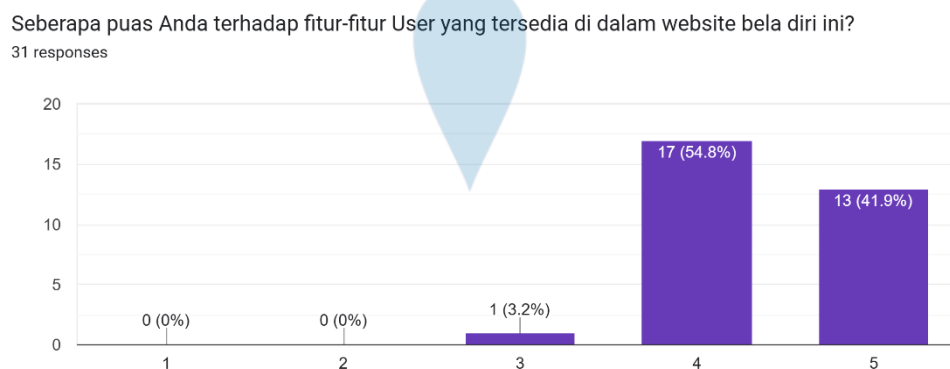


Gambar 5.43 Pertanyaan Mengenai Tampilan Website Bagian Ketiga

Gambar 5.41, Gambar 5.42 dan Gambar 5.43 merupakan pertanyaan mengenai tampilan website apakah sudah sesuai dengan respondennya dan tanggapan responden mengenai tampilan antarmuka website, penilaian tersebut terkait dengan tata letak, warna ataupun kemudahan untuk bernavigasi disekitar website. Pertanyaan ini disajikan karena tampilan pertama dari sebuah website adalah tampilan yang paling penting. Saran dan komentar yang diberikan oleh pengguna adalah mungkin bisa dirapikan lagi.



Gambar 5.44 Pertanyaan Mengenai Kepuasan User Terhadap Fitur Murid Bagian Pertama

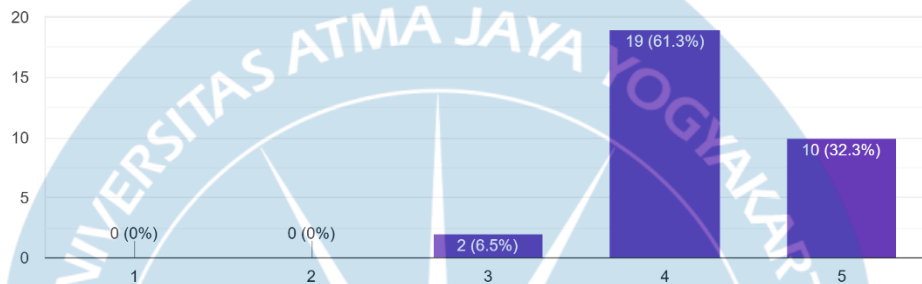


Gambar 5.45 Pertanyaan Mengenai Kepuasan User Terhadap Fitur Murid Bagian Kedua

Gambar 5.44 dan Gambar 5.45 menunjukkan bahwa mayoritas responden merasa fitur yang sudah tersedia dalam website sudah sesuai dengan penjelasan fitur-fitur murid yang diberikan sebelum melakukan pengujian.

Apakah fitur-fitur yang disediakan sudah sesuai untuk Website Bela Diri sebagai Instructor atau Guru?

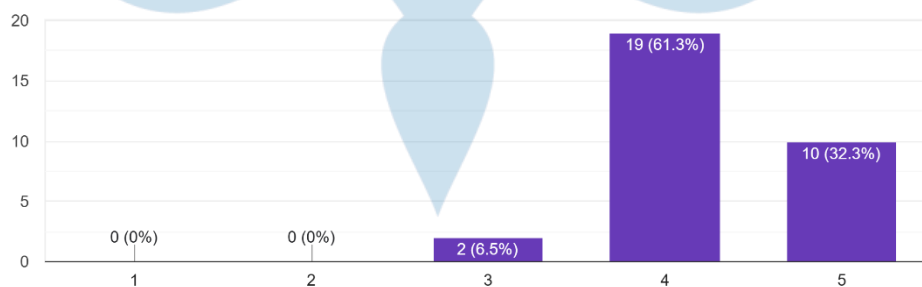
31 responses



Gambar 5.46 Pertanyaan Mengenai Kepuasan User Terhadap Fitur Role Guru Bagian Pertama

Apakah fitur-fitur yang disediakan sudah sesuai untuk Website Bela Diri sebagai Instructor atau Guru?

31 responses

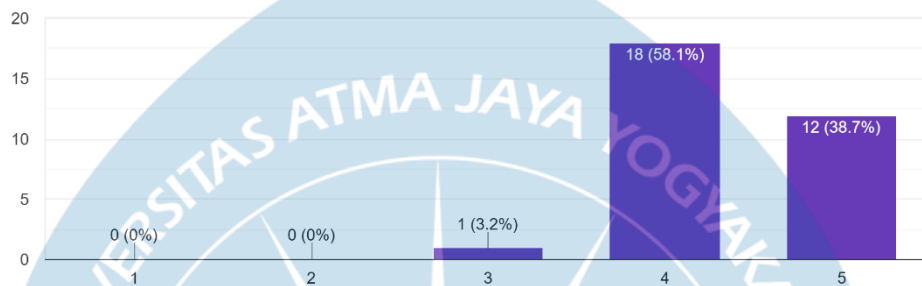


Gambar 5.47 Pertanyaan Mengenai Kepuasan User Terhadap Fitur Role Guru Bagian Kedua

Gambar 5.46 dan Gambar 5.47 merupakan hasil pertanyaan mengenai kepuasan dan tanggapan responden terhadap fitur-fitur yang hanya dapat diakses oleh akun dengan role instruktur. Akun dengan role instruktur

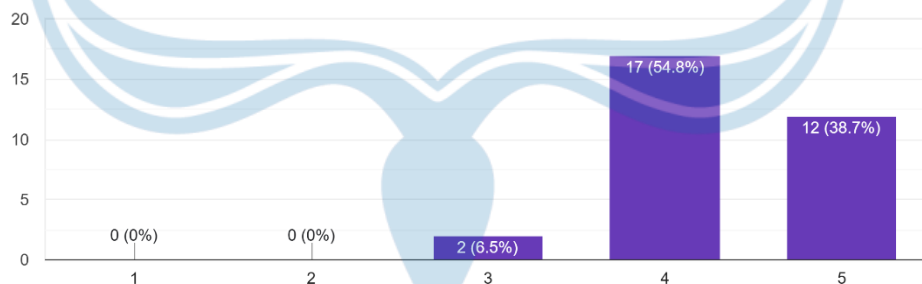
sudah disediakan oleh adminnya langsung, secara langsung digantikan rolenya menjadi instruktur dari database Hostinger.

Apakah fitur-fitur yang disediakan sudah sesuai untuk Website Bela Diri sebagai Admin?
31 responses



Gambar 5.48 Pertanyaan Mengenai Kepuasan User Terhadap Fitur Role Admin Bagian Pertama

Seberapa puas Anda terhadap fitur-fitur Admin yang tersedia di dalam website bela diri ini?
31 responses

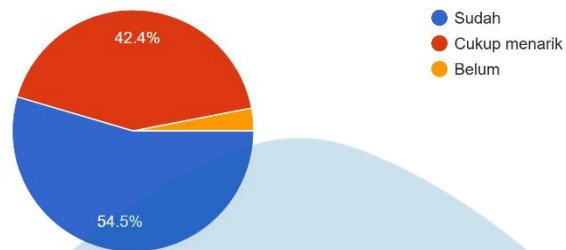


Gambar 5.49 Pertanyaan Mengenai Kepuasan User Terhadap Fitur Role Admin Bagian Kedua

Gambar 5.48 dan Gambar 5.49 menunjukkan kepuasan responden terhadap fitur yang disediakan untuk admin dalam website, seperti kenaikan dan penurunan sabuk, absensi, aktifkan dan menonkatifkan akun. Semua fitur sudah diuji oleh responden.

Menurut Anda, Apakah Websitenya sudah cukup menarik?

33 responses



Gambar 5.50 Pertanyaan Penutup Kuesioner

Kuesioner ditutup dengan beberapa pertanyaan umum seperti pada Gambar 5.50. menunjukkan bahwa tidak semua responden sudah menanggapi website tersebut sebagai website yang menarik, terdapat beberapa responden yang menanggapinya cukup menarik dan 1% belum menarik dengan alasan tidak terlalu rapi dan terdapat beberapa elemen yang tidak memiliki fungsionalitas.