

## BAB III

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Koordinasi

Kedudukan atau jabatan merupakan suatu komponen penting dalam struktur organisasi perusahaan yang menggambarkan posisi seseorang beserta tugas, tanggung jawab, wewenang, dan hak yang melekat padanya. Dalam konteks analisis jabatan, kedudukan dapat dipahami sebagai hasil dari proses, metode, dan teknik pengumpulan data tentang suatu posisi yang kemudian diolah menjadi informasi jabatan yang komprehensif. Pemahaman yang jelas mengenai kedudukan seseorang dalam organisasi menjadi fundamental karena menetapkan batasan peran dan ekspektasi kinerja yang diharapkan dari individu tersebut dalam menjalankan fungsinya dalam perusahaan [3].

Koordinasi, di sisi lain, adalah proses pengintegrasian tujuan dan kegiatan pada satuan-satuan yang terpisah dalam sebuah organisasi untuk mencapai tujuan bersama secara efisien. Kegiatan koordinasi melibatkan berbagai pihak dari tim yang sederajat yang berkolaborasi dengan kesepakatan yang jelas untuk menghindari tumpang tindih pekerjaan atau konflik antar departemen. Melalui koordinasi yang efektif, aliran informasi menjadi lebih terstruktur, pengambilan keputusan menjadi lebih terarah, dan risiko kesalahan dalam pelaksanaan tugas dapat diminimalisir, sehingga organisasi dapat bergerak secara harmonis menuju pencapaian tujuan strategisnya [4].

##### 3.1.1 Kedudukan

Selama pelaksanaan magang di PT IlmuKomputerCom Braindevs Sistema, mahasiswa menempati posisi sebagai *Specialist Internship In Software* yang secara formal termasuk dalam Divisi *System Analysis and Design*. Meskipun secara jabatan merupakan peserta magang, dalam praktiknya mahasiswa menjalankan peran selayaknya *Software Engineer* profesional, terutama dalam konteks pembuatan *website* menggunakan framework Laravel. Kedudukan ini memberikan tanggung jawab yang

signifikan dalam siklus pengembangan *website*, mulai dari perancangan database, pengembangan logika bisnis pada backend, hingga implementasi antarmuka pengguna pada frontend. Dalam proyek BNSP, mahasiswa dipercaya untuk mengerjakan komponen-komponen yang langsung berkontribusi pada produk akhir yang digunakan oleh klien. Posisi ini juga melibatkan interaksi langsung dengan stakeholder dan tim pengembang senior, yang memberikan pengalaman berharga dalam memahami alur kerja profesional dan standar kualitas dalam industri pengembangan perangkat lunak.

### **3.1.2 Koordinasi**

Koordinasi adalah proses penyelarasan aktivitas antar individu atau tim dalam suatu organisasi untuk mencapai tujuan bersama secara efisien dan efektif. Dalam konteks kerja profesional, koordinasi menjadi elemen penting yang memastikan setiap pihak memahami peran, tanggung jawab, dan batasan kerja masing-masing, serta mencegah terjadinya konflik atau tumpang tindih pekerjaan. Koordinasi yang baik menciptakan alur komunikasi yang jelas, mempercepat pengambilan keputusan, dan meningkatkan produktivitas tim secara keseluruhan.

Selama menjalani program magang di PT IlmuKomputerCom Braindevs Sistema, koordinasi antar anggota tim dilakukan dengan memanfaatkan beberapa platform digital yang mendukung kolaborasi. Media utama yang digunakan untuk komunikasi sehari-hari adalah Telegram, yang berfungsi sebagai saluran diskusi informal maupun koordinasi cepat antar anggota tim. Melalui Telegram, mahasiswa dapat menerima arahan serta tugas, berdiskusi mengenai kendala teknis, serta mendapatkan pembaruan informasi secara real-time dari supervisor atau mentor.

Selain itu, perusahaan menggunakan ClickUp sebagai sistem manajemen proyek terpusat. Di platform ini, seluruh aktivitas kerja mahasiswa dicatat secara harian, termasuk progress pekerjaan, task yang sedang dan akan

dikerjakan, serta task yang selesai sudah dikerjakan. ClickUp juga berfungsi sebagai alat dokumentasi kolaboratif dalam tim, di mana mahasiswa dapat menambahkan komentar, mengunggah file terkait pekerjaan, dan mengikuti perkembangan proyek secara keseluruhan.

### 3.2 Tugas dan Uraian Kerja Magang

Pada pelaksanaan program magang yang dilakukan di PT IlmuKomputerCom Braindevs Sistema selama 5 bulan, mahasiswa diberikan kesempatan untuk berkontribusi dalam project *website* Badan Nasional Sertifikasi Profesi yang sudah dijalani dari tahun 2022. Tugas mayoritas yang mahasiswa kerjakan adalah pengembangan atau pembetulan fitur yang sudah ada, dengan kesempatan beberapa kali untuk membuat atau menambahkan fitur yang baru. Tabel 2 dibawah inimenunjukkan kegiatan-kegiatan yang dilakukan oleh mahasiswa sewaktu magang.

Tabel 3.1 Pelaksanaan Tugas Kerja Magang

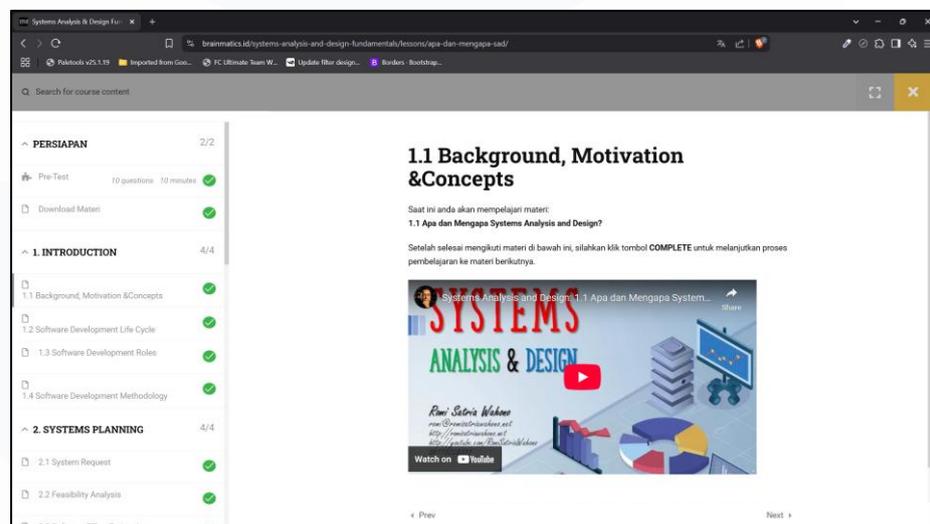
No	Kegiatan	Periode	Tanggal Mulai	Tanggal Selesai
1	Studi dan Peningkatan Kemampuan Teknis	3 Minggu	30 Januari	14 Februari
2	Pengembangan dan Pemeliharaan Halaman Depan dan Halaman Cek Sertifikasi di <i>Website</i> BNSP	4 Minggu	3 Februari	28 Februari
3	Pengembangan Skema dan Pembuatan Fitur Revisi Skema	7 Minggu	5 Februari	21 Maret
4	Pembuatan Halaman List Sertifikasi	14 Minggu	21 Februari	23 Mei
5	Pengembangan Asesor Kompetensi dan Asesi	15 Minggu	11 Februari	23 Mei
6	Pengembangan Profile LSP	17 Minggu	19 Februari	11 Juni
7	Pengembangan Paket Kompetensi	13 Minggu	12 Maret	11 Juni
8	Pengembangan Jadwal Sertifikasi	10 Minggu	5 Maret	20 Mei
9	Pengembangan Fitur Minor	14 Minggu	19 Februari	3 Juni
10	Dokumentasi dan Meeting	13 Minggu	11 Maret	11 Juni

### 3.2.1 Studi dan Peningkatan Kemampuan Teknis

Untuk mendukung pelaksanaan tugas magang, dilakukan beberapa kegiatan untuk meningkatkan kemampuan teknis sesuai dari arahan mentor.

#### 3.2.1.1 Kursus Brainmatics

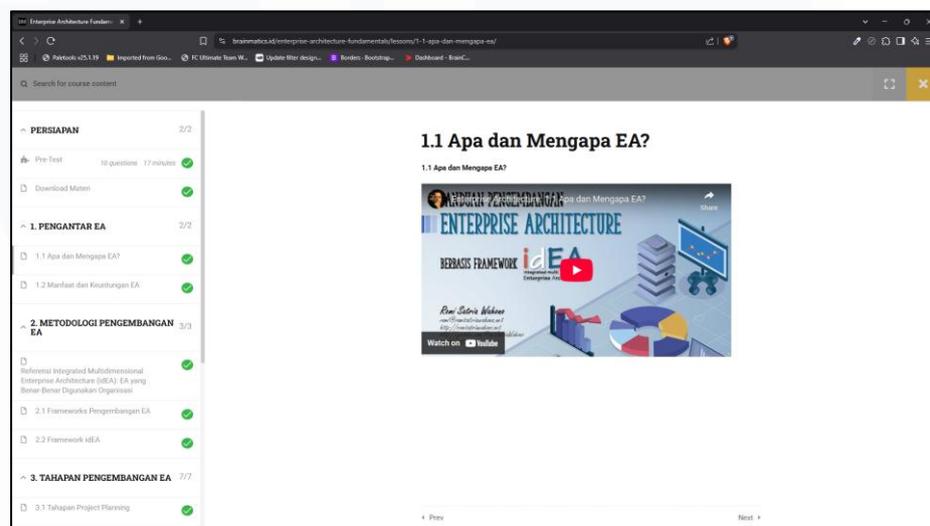
Pada minggu pertama masa magang, mahasiswa ditugaskan untuk mempelajari GIT sebagai pengetahuan dasar untuk berkolaborasi dengan tim dalam proyek BNSP. Selain itu, mahasiswa juga diwajibkan untuk menguasai PHP Laravel, framework yang menjadi dasar pengembangan *website* BNSP. Terakhir, terdapat materi pembelajaran wajib lainnya, yaitu *System Analysis and Design Fundamentals*, *Enterprise Architecture Fundamentals Course*, dan *Data Mining AI Studio Fundamentals*. Kursus-kursus tersebut disediakan secara gratis oleh Brainmatics, perusahaan saudara Braindevs yang berfokus pada penyediaan kursus-kursus teknologi.



Gambar 3.1 System Analysis And Design Course

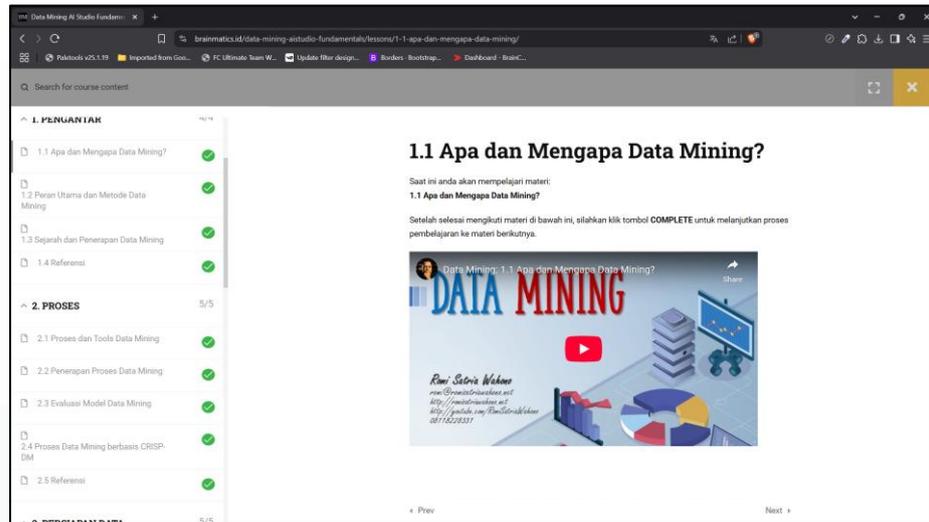
Gambar 3.1 menampilkan halaman *website* kursus yang wajib diselesaikan oleh mahasiswa. Konten kursus ini tersusun secara sistematis, dimulai dari Pre-Test, diikuti dengan Introduction, System Planning, System Analysis, System Design, System Implementation, dan

diakhiri dengan Post-Test. Setiap subjek materi dilengkapi dengan video penjelasan dari YouTube yang dibuat oleh Romi Satria Wahono. Pada tahap akhir, diwajibkan untuk mengerjakan Post-Test dan setelah menyelesaikan seluruh rangkaian materi mahasiswa memperoleh sertifikat yang dapat diunduh. Penyelesaian kursus ini diharapkan dapat meningkatkan pemahaman mahasiswa mengenai System Analysis and Design.



Gambar 3.2 Enterprise Architecture Fundamentals Course

Gambar 3.2 menampilkan halaman untuk kursus kedua yang berjudul Enterprise Architecture Fundamentals. Materi dalam kursus ini disusun secara berurutan, dimulai dari Pre-Test sebagai pengantar awal, kemudian dilanjutkan dengan topik-topik utama seperti Pengantar EA, Metodologi Pengembangan EA, Tahapan Pengembangan EA, dan diakhiri dengan Post-Test untuk mengukur pemahaman akhir mahasiswa. Setiap bagian materi dilengkapi dengan video penjelasan yang dibuat oleh Romi Satria Wahono dan diunggah melalui platform YouTube. Kursus ini dirancang untuk memperluas wawasan mahasiswa terkait dasar-dasar Enterprise Architecture dan penerapannya dalam organisasi.



Gambar 3.3 Data Mining AI Studio Fundamentals

Kursus terakhir yang diikuti oleh mahasiswa berjudul Data Mining AI Studio Fundamentals, sesuai dengan yang ditampilkan pada Gambar 3.3 Materi dalam kursus ini tersusun secara berurutan, dimulai dengan Pre-Test sebagai pengantar, dilanjutkan dengan topik-topik inti seperti Pengantar, Proses, Persiapan Data, Algoritma, Text Mining, Research, dan ditutup dengan sesi Penutupan yang mencakup Post-Test untuk mengevaluasi pemahaman mahasiswa. Setiap modul dilengkapi dengan video penjelasan yang disampaikan oleh Romi Satria Wahono melalui platform YouTube. Kursus ini bertujuan untuk memperkaya pengetahuan mahasiswa mengenai konsep dasar Enterprise Architecture serta penerapannya di lingkungan organisasi. Setelah seluruh materi dan Post-Test diselesaikan, mahasiswa dapat memperoleh sertifikat yang bisa diunduh sebagai bukti kelulusan.

### 3.2.1.2 Pembelajaran Laravel dan Git



Gambar 3.4 Tutorial LARAVEL 11 dari Web Programming UNPAS

Gambar 3.4 menunjukkan playlist video tutorial Laravel 11 dari *channel* YouTube Web Programming UNPAS, yang digunakan sebagai salah satu referensi pembelajaran mandiri selama minggu pertama masa magang. Playlist ini terdiri dari 17 video yang membahas Laravel secara berurutan, dimulai dari pengenalan Laravel, proses instalasi dan konfigurasi, struktur folder, hingga topik lanjutan seperti Eloquent Relationship, Database Seeder, Pagination, dan optimasi performa. Materi yang dibawakan oleh Dosen Universitas Pasundan ini disampaikan secara sistematis dan mudah dipahami, sangat membantu dalam memahami alur kerja Laravel sebagai framework backend berbasis PHP. Setiap judul video menjadi subtopik pembelajaran yang dipelajari secara mendalam untuk memperkuat pemahaman teknis sebelum terlibat langsung dalam pengembangan proyek di perusahaan. Dengan mengikuti tutorial ini, mahasiswa mampu memahami konsep Blade Templating, penggunaan Model dan Migration, serta cara berinteraksi dengan database menggunakan Eloquent ORM yang kemudian diaplikasikan dalam pengembangan berbagai fitur pada *website* proyek BNSP.



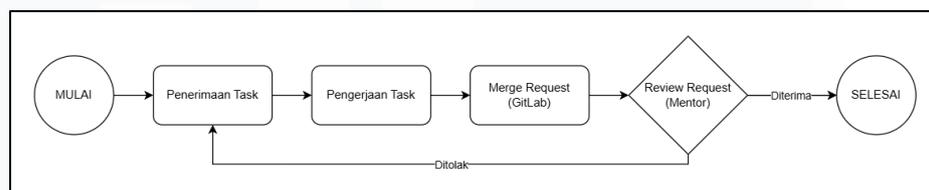
Gambar 3.5 GIT & GITHUB dari Web Programming UNPAS

Gambar 3.5 memperlihatkan playlist video dari YouTuber Web Programming UNPAS yang berfokus pada pembelajaran Git dan GitHub, dua alat penting dalam pengembangan perangkat lunak modern. Playlist ini terdiri dari 14 video yang membahas secara bertahap mulai dari pengenalan Git & GitHub, penggunaan branch, fork, hingga proses merge dan penyelesaian konflik. Materi seperti git remote, gitignore, dan rebase juga turut dijelaskan, memberikan wawasan mendalam mengenai manajemen versi dan kolaborasi dalam proyek tim. Video tutorial ini sangat membantu dalam mempersiapkan mahasiswa menghadapi lingkungan kerja profesional yang menerapkan Git sebagai sistem version control utama, termasuk dalam proyek pengembangan *website* BNSP yang dikerjakan selama masa magang. Melalui pembelajaran dari playlist ini, mahasiswa menjadi lebih siap dalam melakukan branching, menyelesaikan konflik merge, serta memahami alur kerja kolaboratif dengan GitHub secara efisien dan sistematis.

### 3.2.2 Pengembangan dan Pemeliharaan Halaman Depan, Cek Sertifikasi, dan Detail Jadwal Sertifikasi

Pengembangan dan pemeliharaan ini berfokus pada beberapa halaman penting, seperti halaman depan, cek sertifikasi, dan detail jadwal sertifikasi.

#### 3.2.2.1 Pengembangan Halaman Depan



Gambar 3.6 Flowchart Pembuatan atau Pengembangan *Website* BNSP

Gambar 3.6 atas menunjukkan alur kerja dalam proses pembuatan atau pengembangan *website* BNSP. Alur ini digunakan secara konsisten dalam setiap tugas yang dikerjakan oleh mahasiswa magang. Tugas-tugas tersebut dapat diterima melalui berbagai media komunikasi seperti grup Telegram, instruksi langsung dari mentor, atau melalui sistem manajemen tugas seperti ClickUp. Setelah menerima tugas, mahasiswa akan mengerjakannya, kemudian membuat merge request melalui GitLab untuk direview oleh mentor. Apabila terdapat kekurangan, tugas akan dikembalikan untuk diperbaiki. Jika telah sesuai, merge request akan disetujui dan perubahan akan digabungkan ke repositori utama.

```

1 | 1 | @extends('landing-page.templates.master')
2 | 2 | -
3 | 3 | @section('custom-css')
4 | 4 | <style>
5 | 5 |     .title {
6 | 6 |         height: 50px;
7 | 7 |     }
8 | 8 |     .testimonial_item {
9 | 9 |         padding: 0;
10 | 10 |     }
11 | 11 |     @media (max-width: 767px) {
12 | 12 |         .title-limit {
13 | 13 |             font-size: 14px !important;
14 | 14 |         }
15 | 15 |     }
16 | 16 |     @media (min-width: 768px) {
17 | 17 |         .title-limit {
18 | 18 |             font-size: 16px !important;
19 | 19 |         }
20 | 20 |     }
21 | 21 |     @media (min-width: 1200px) {
22 | 22 |         .title-limit {
23 | 23 |             font-size: 18px !important;
24 | 24 |         }
25 | 25 |     }
26 | 26 |     .img_carousel {
27 | 27 |         border-top-left-radius: .25rem;
28 | 28 |         border-top-right-radius: .25rem;
29 | 29 |     }
30 | 30 |     .filter-icon {
31 | 31 |         width: 13px;
32 | 32 |     }
33 | 33 |     .dropdown-menu {
34 | 34 |         width: 180%;
35 | 35 |     }
36 | 36 | </style>
37 | 37 | @endsection
38 | 38 | @section('content')
39 | 39 | <section class="d-flex align-items-center">

```

Gambar 3.7 Menambahkan CSS Serta Menyembunyikan Filter Untuk Konsep Awal Carousel

Gambar 3.7 menampilkan tahap awal pengembangan halaman landing page BNSP, dilakukan penyempurnaan UI dengan menambahkan CSS untuk mempercantik tampilan awal carousel jadwal sertifikasi. Selain itu, bagian filter sementara disembunyikan menggunakan sintaks komentar Blade “`{{{-- --}}`” agar tidak ditampilkan pada tampilan akhir pengguna, namun tetap tersimpan dalam kode untuk keperluan pengembangan selanjutnya. CSS tambahan ini dikirimkan ke bagian “`<head>`” halaman menggunakan mekanisme “`@section('custom-css')`”, yang kemudian diproses oleh template layout utama Laravel. Pendekatan ini memudahkan pengelolaan gaya khusus per halaman tanpa mengganggu tampilan global situs secara keseluruhan.

Setiap halaman dapat mengirimkan custom CSS maupun custom JavaScript masing-masing. Hal tersebut dapat dilakukan karena setiap halaman merujuk pada master blade-nya masing-masing. Pada setiap master blade tersebut, terdapat bagian untuk mengimpor CSS dan JavaScript yang spesifik sesuai dengan kebutuhannya masing-masing, serta penggunaan “`@yield('custom-js')`”, “`@stack('stack-js')`”, “`@yield('custom-css')`”, dan “`@stack('stack-css')`” yang memungkinkan halaman untuk menggunakan bagian khusus sesuai dengan

kebutuhannya. Setelah itu, konten utama halaman akan dimasukkan melalui tag “<main>@yield('content')</main>”.

```
11  
12  
13  
14  
15  
16  
17  
@@ -38,9 +39,18 @@ public function index()  
39  
40  
41  
42 + $kontenList = Konten::query()->limit(6)->get();  
43 + $kontenList = Konten::query()->get();  
44 + $albumList = Album::query()->latest()->limit(4)->get();  
45 + $beritaList = GeneralInformation::query()->latest()->limit(4)->get();  
46 + $sertifikasiliist = JadwalSertifikasiView::query()->active()->limit(20)->get();  
47 $data = [];  
48 $data['title'] = 'BNSP - Badan Nasional Sertifikasi Profesi';  
59  
60  
61
```

Gambar 3.8 Pengambilan Data Sesuai dari Database

Gambar 3.8 menampilkan halaman merge request pada GitLab yang menunjukkan hasil pekerjaan yang telah diselesaikan dalam pengembangan halaman utama *website* BNSP. Tugas yang dijalani yaitu mengubah sistem sebelumnya yang menampilkan data secara statis menjadi dinamis dengan menarik data langsung dari database untuk tiga komponen utama: carousel jadwal sertifikasi, album galeri, dan berita terbaru. Salah satu contoh implementasi yang dilakukan adalah penggunaan fungsi `Carbon::parse($this->tanggal_mulai)->isoFormat('D MMMM YYYY')` untuk menampilkan tanggal jadwal sertifikasi dalam format lokal yang lebih informatif dan mudah dipahami oleh pengguna.



Gambar 3.9 Konsep Carousel dan Filter Jadwal Sertifikasi

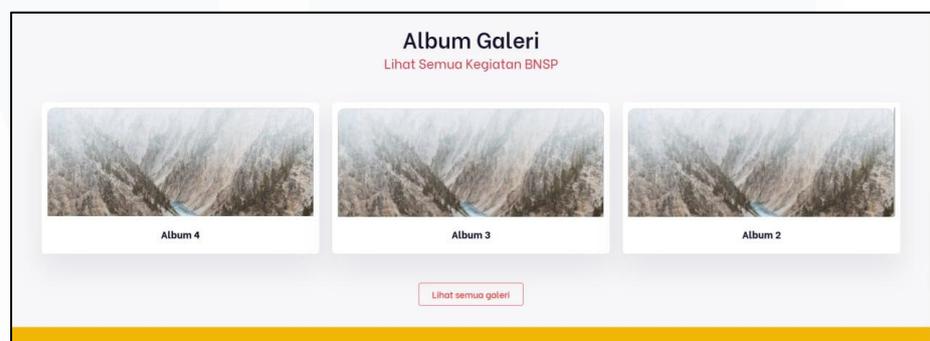
Gambar 3.9 menampilkan konsep awal fitur carousel yang dirancang untuk menayangkan jadwal sertifikasi secara interaktif di halaman utama *website* BNSP. Pada tahap perancangan ini, elemen-elemen visual dasar seperti judul sertifikasi, nama LSP penyelenggara, lokasi, tanggal pelaksanaan, dan tombol Daftar telah diposisikan untuk memberikan gambaran awal kepada tim pengembang. Selain *carousel*, prototipe ini juga memasukkan komponen filter yang memungkinkan pengunjung menyaring jadwal berdasarkan jenis sertifikasi, provinsi, dan bulan pelaksanaan. Integrasi filter di bagian atas carousel diharapkan dapat meningkatkan aksesibilitas informasi, memudahkan pengguna menemukan jadwal yang relevan, serta memperkuat daya tarik visual halaman depan.



Gambar 3.10 Konsep Berita Terbaru

Gambar 3.10 menampilkan konsep Berita Terbaru yang dirancang untuk halaman landing page *website* BNSP. Bagian tersebut

menampilkan empat berita terkini, dengan satu berita utama ditampilkan secara dominan di sisi kiri sebagai fokus utama. Setiap berita dilengkapi dengan judul, tanggal, dan waktu publikasi yang ditampilkan secara jelas. Tata letaknya disusun agar memudahkan pengguna dalam membaca informasi secara cepat dan terstruktur. Di bagian bawah terdapat tombol “Lihat semua berita” yang mengarahkan pengguna ke halaman untuk melihat semua berita BNSP secara lengkap. Bagian berita ini memungkinkan pengguna untuk tetap terinformasi mengenai kegiatan dan perkembangan terbaru di lingkungan BNSP secara efisien.



Gambar 3.11 Konsep Album Galeri

Gambar 3.11 menampilkan konsep awal dari fitur album galeri yang dirancang untuk halaman utama *website* BNSP. Komponen ini dirancang agar mampu menampilkan dokumentasi visual seperti kegiatan pelatihan, sertifikasi, dan acara resmi lainnya. Setiap gambar pada galeri diambil secara dinamis dari database, sehingga dapat dikelola langsung oleh administrator melalui sistem Content Management System (CMS) yang telah disiapkan. Dengan pendekatan ini, admin memiliki fleksibilitas untuk menambah, mengubah, atau menghapus gambar sesuai kebutuhan tanpa perlu mengakses langsung ke dalam kode program. Album galeri ini berfungsi sebagai bukti aktivitas dan kredibilitas lembaga dalam melaksanakan program sertifikasi profesi.



Gambar 3.12 Konsep Media Sosial

Gambar 3.12 menampilkan rancangan awal dari konsep Media Sosial untuk halaman *landing page website* BNSP. Komponen ini menampilkan pilihan platform media sosial seperti Instagram, Facebook, X/Twitter, dan YouTube yang ditampilkan di sisi kanan sebagai tombol navigasi. Jika salah satu platform dipilih, kontennya akan ditampilkan secara besar di sisi tengah halaman. Dalam contoh ini, tampilan utama menampilkan feed Instagram resmi BNSP (@bnspp\_official), termasuk jumlah pengikut, total unggahan, serta beberapa unggahan yang terbaru. Komponen ini dirancang agar informasi dan aktivitas terbaru dari BNSP yang dibagikan melalui media sosial dapat diakses langsung oleh pengunjung situs secara visual dan terintegrasi.

### 3.2.2.2 Pengembangan Halaman Cek Sertifikasi

```

@@ -57,14 +59,24 @@ public function fetchCertificationData(Request $request)

    $data = $this->ps_repository->getAll('eCertification', $request)[0] ?? null;

    $dbDate = \Carbon\Carbon::parse($data->tanggal_cetak);
    $diffYears = \Carbon\Carbon::parse($data->tanggal_expired)->diffInYears($dbDate);

    $result['certification'] = $data ? [
        'asesi' => $data->asesi,
        'nomor_blanko' => $data->nomor_blanko,
        'nomor_sertifikat' => $data->nomor_sertifikat,
        'id_skema' => $data->id_skema,
        'kode_skema' => $data->kode_skema,
        'skema' => $data->skema,
        'unit_skema' => SkemaUnit::where('id_skema', $data->id_skema)->first(),
        'status' => $data->status,
        'lsp' => $data->lsp,
        'tanggal_uji' => $data->tanggal_uji,
        'tanggal_sertifikasi' => $data->tanggal_sertifikasi,
        'tanggal_berakhir' => $data->tanggal_berakhir,
        'status' => $data->status,
        'masa_berlaku' => $diffYears,
        'foto' => $data->foto,
    ] : [];

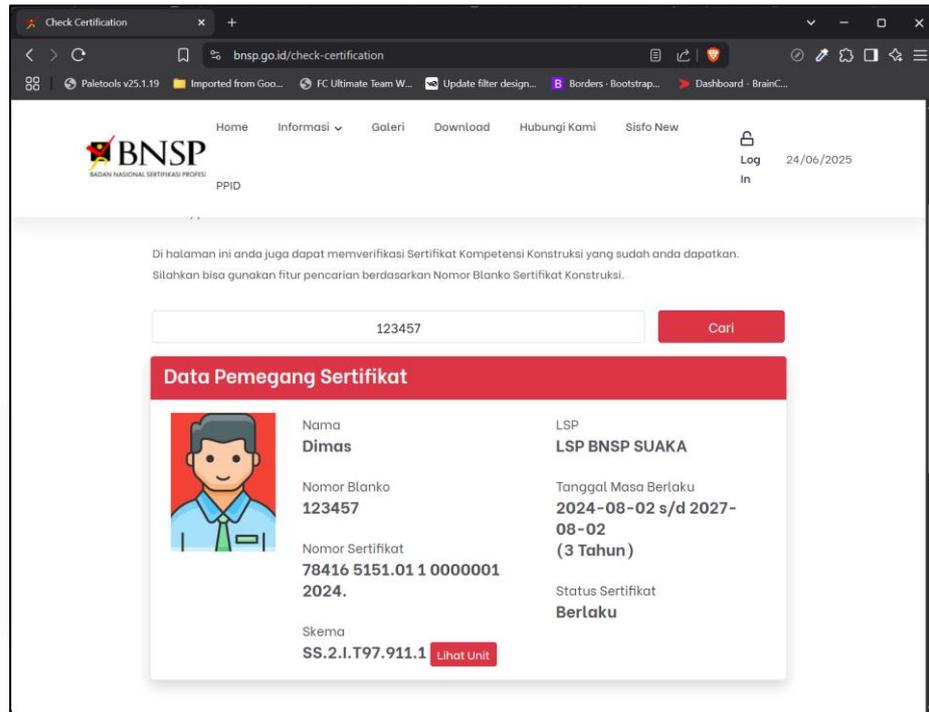
    return CustomResponse::success($result);

```

Gambar 3.13 Kode Pengambilan Data Sertifikasi

Gambar 3.13 menampilkan potongan kode yang digunakan pada halaman Cek Sertifikasi di *website* BNSP, pengguna dapat mencari sertifikat berdasarkan Nomor Sertifikat atau Nomor Blanko. Untuk mendukung fungsionalitas ini, diterapkan proses pengambilan data sertifikasi melalui kode backend. Pada potongan kode ini, data yang ditemukan dikelompokkan dalam struktur array `$result['certification']`, yang mencakup informasi penting seperti nama asesi, masa berlaku sertifikat, hingga foto.

Sebelum data dikembalikan ke frontend, dilakukan pemrosesan terhadap format tanggal. Tanggal cetak sertifikat diparsing terlebih dahulu menggunakan `\Carbon\Carbon::parse($data->tanggal_cetak)` untuk memastikan konsistensi dalam format. Kemudian, selisih tahun antara tanggal cetak dan tanggal kedaluwarsa dihitung menggunakan `diffInYears`, yang menghasilkan informasi masa berlaku sertifikat secara dinamis. Pendekatan ini memungkinkan sistem menampilkan data yang akurat dan terstruktur saat pengguna melakukan pencarian, serta menjaga agar informasi yang diberikan tetap mudah dipahami oleh pengguna.



Gambar 3.14 Halaman Cek Sertifikasi

Gambar 3.14 menampilkan halaman Cek Sertifikasi. Setelah pengguna memasukkan Nomor Sertifikat atau Nomor Blanko dan menekan tombol “Cari”, sistem menampilkan hasil pencarian dalam bentuk data yang terstruktur dan mudah dibaca. Informasi yang ditampilkan mencakup Data Pemegang Sertifikat, yang terdiri dari pas foto, nama lengkap asesi, nomor blanko, nomor sertifikat, nama LSP, masa berlaku sertifikat, serta status sertifikat. Pada tampilan ini, informasi masa berlaku ditampilkan dalam format tanggal awal hingga tanggal akhir (misalnya 2024-08-02 s/d 2027-08-02) disertai perhitungan masa berlaku (misalnya 3 Tahun). Selain itu, pengguna dapat melihat Unit Skema yang dimiliki oleh skema tersebut dengan memilih “Lihat Unit”.

Unit Skema: Housekeeper

Kode Unit	Nama
N.79JPW00.043.2	Mengikuti Prosedur Keselamatan dan Keamanan <i>Follow Safety and Security Procedures</i>
N.79JPW00.058.2	Melakukan Kerjasama Secara Efektif Dengan Pelanggan dan Kolega <i>Work Effectively with Customers and Colleagues</i>
N.79JPW00.061.2	Melakukan Komunikasi Dengan Efektif <i>Communicate Effectively</i>
N.79JPW00.065.2	Melakukan Prosedur Administrasi <i>Perform Administrative Procedures</i>
N.79JPW00.066.2	Mendapatkan Data dari Komputer <i>Obtain Data From Computers</i>
N.79JPW00.130.2	Melakukan Percakapan Dasar Dalam Bahasa Inggris di Tingkat Operasional

Gambar 3.15 Modal Unit Skema

Gambar 3.15 menunjukkan tampilan modal yang muncul ketika pengguna memilih “Lihat Unit” pada halaman cek sertifikasi. Di dalam modal tersebut, ditampilkan daftar unit kompetensi yang termasuk dalam skema sertifikasi, dan disusun dalam bentuk tabel. Tabel ini memuat informasi berupa Kode Unit, Nama Unit dalam Bahasa Indonesia, serta terjemahan Nama Unit dalam Bahasa Inggris. Penyajian data dalam format tabel ini memudahkan pengguna dalam memahami daftar unit kompetensi yang dimiliki oleh skemanya.

### 3.2.2.3 Pengembangan Halaman Detail Jadwal Sertifikasi

```

@foreach ($sertifikasi->kompetensi as $
<tr>
<th scope="row">{{ $loop->iteration
<td>{{ $item->
<td>{{ $item->
}}</td>
</tr>
</tbody>
</table>
</div>
<div class="row" style="margin-top: 50px">
<h5 class="fs-2 mb-3">LSP Pelaksana</h5>
<div class="col-12 col-lg-6 mb-3 mb-lg-0">
<div class="h-100">

</div>
</div>
<div class="col-12 col-lg-6">
<div class="d-flex justify-content-center flex-column">
<h5 class="fw-bold fs-5">{{ $sertifikasi->
}}</h5>
<p class="fs-6">
No. SK Lisensi: {{ $sertifikasi->
}} <br>
No. Lisensi: {{ $sertifikasi->
}} <br>
Jenis: {{ $sertifikasi->
}} <br>
Email: <a href="{{ $sertifikasi->
}}">{{ $sertifikasi->
}}</a> <br>
Website: <a href="{{ $sertifikasi->
}}" target="_blank">{{ $sertifikasi->
}}</a> <br>
Masa Berlaku Sertifikat: {{ $sertifikasi->
}} <br>
</p>
</div>
</div>
</div>
</div>

```

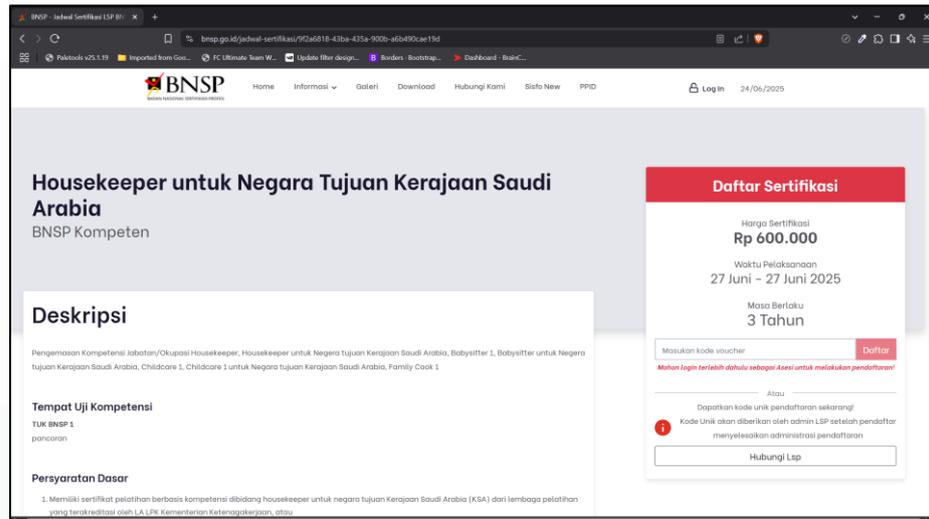
```

class CertificationDetailController extends Controller
{
public function index($id_encrypted)
{
$data = [];
$id = Helper::decryptId($id_encrypted);
$kontenList = Konten::with('dokumen')->where('idkategori', 1)-
>orderBy('tanggal', 'desc')->limit(3)->get();
$sertifikasi = JadwalSertifikasiView::with('lsp')->with('kompetensi')
->find($id);
// dd($sertifikasi->test);
$data['title'] = 'BNSP - Certification Detail';
$data['kontenList'] = $kontenList;
$data['sertifikasi'] = $sertifikasi;
return view('landing-page.certification-detail',$data);
}
}

```

Gambar 3.16 Kode Menampilkan Data Unit Kompetensi dan Informasi LSP

Gambar 3.16 menunjukkan bagian kode yang digunakan untuk menampilkan data pada halaman detail sertifikasi secara dinamis. Sebelumnya, data-data seperti nama sertifikasi, unit kompetensi, dan informasi LSP ditulis secara statis. Mahasiswa diberi tugas untuk mengubahnya agar mengambil langsung dari database berdasarkan jadwal sertifikasi yang sedang dibuka. Data tersebut dikirim dari controller melalui array \$data['sertifikasi'], dan ditampilkan di halaman dengan menggunakan sintaks seperti {{ \$sertifikasi->nama }}. Data yang ditampilkan meliputi nama skema, deskripsi, persyaratan dasar, daftar unit kompetensi, serta informasi lengkap dari LSP yang menyelenggarakan sertifikasi.



Gambar 3.17 Halaman Detail Jadwal Sertifikasi

Gambar 3.17 menunjukkan tampilan bagian atas dari halaman detail jadwal sertifikasi yang telah dikembangkan. Halaman ini menyajikan informasi seperti nama sertifikasi, nama LSP penyelenggara, deskripsi, lokasi Tempat Uji Kompetensi (TUK), serta daftar persyaratan dasar. Di sisi kanan, ditampilkan detail harga, tanggal pelaksanaan, serta masa berlaku sertifikasi. Pengguna juga dapat langsung mendaftar melalui form yang tersedia dengan mengisi kode voucher yang diberikan oleh LSP. Terdapat elemen visual berupa latar belakang abu-abu lebar di belakang nama skema dan LSP yang dirancang sebagai tempat untuk menampilkan gambar tampak depan kantor LSP. Namun, karena penempatan gambar ini masih menunggu konfirmasi lebih lanjut dari mentor, maka sementara dibiarkan dalam bentuk placeholder abu-abu.

Data Unit		
13 Unit Kompetensi		
No	Kode Unit	Keterangan
1	T.97PDO00.001.3	Membekali Diri Tentang Kondisi Kerja dan Risiko Kerja
2	T.97PDO00.002.3	Menerapkan Prosedur Keselamatan dan Kesehatan Kerja di Tempat Kerja
3	T.97PDO00.003.3	Melaksanakan Kerjasama di Lingkungan Kerja
4	T.97PDO00.004.3	Mengembangkan Kematangan Emosi dan Motivasi Kerja
5	T.97PDO00.005.3	Melakukan Komunikasi dengan Menggunakan Bahasa Negara Penempatan
6	T.97PDO00.006.3	Menerapkan Prinsip-Prinsip Dasar Pembersihan Lingkungan
7	T.97PDO00.007.3	Mengoperasikan Peralatan Pembersih
8	T.97PDO00.008.3	Membersihkan Ruang Keluarga dan Kamar Tidur
9	T.97PDO00.009.3	Membersihkan Kamar Mandi dan Fasilitas Toilet
10	T.97PDO00.010.3	Membersihkan Area Masak
11	T.97PDO00.011.3	Mencuci Pakaian dan Linen
12	T.97PDO00.012.3	Menyetrika Pakaian dan Linen
13	T.97PDO00.043.3	Menghidangkan Makanan Dan Minuman

Gambar 3.18 List Unit Kompetensi Pada Halaman Detail Jadwal Sertifikasi

Tabel yang ditampilkan pada gambar 3.18 berfungsi untuk menampilkan seluruh unit kompetensi yang menjadi bagian dari skema sertifikasi terkait. Setiap unit menampilkan kode unit dan nama unit dalam Bahasa Indonesia. Tabel ini dikembangkan agar dapat menampilkan data secara dinamis berdasarkan jadwal yang dipilih, sehingga pengguna dapat memahami informasi mengenai unit kompetensi dari skema tersebut secara lengkap dan jelas.

### LSP Pelaksana



**BNSP Kompeten**

No. SK Lisensi: KEP/2218/BNSP/X/2030

No. Lisensi: BNSP-LSP-2368-ID

Jenis: LSP Pihak Ketiga

Email: doni.sdr10091@gmail.com

Website: www.lsp1.smknegeri1bungo.sch.id

Masa Berlaku Sertifikat: 2028-10-20

Status SK Lisensi: Aktif

Gambar 3.19 Informasi LSP Pada Halaman Detail Jadwal Sertifikasi

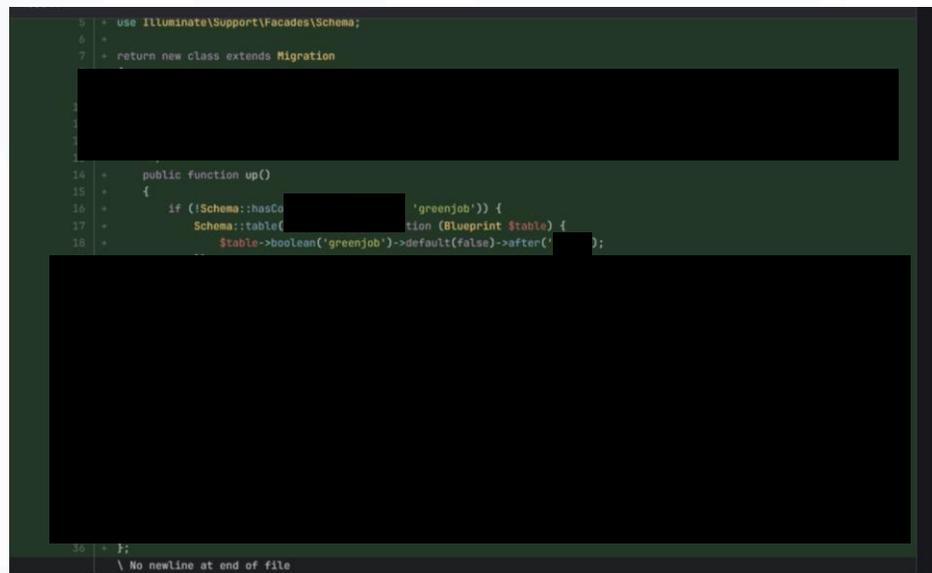
Gambar 3.19 menunjukkan informasi lengkap terkait LSP yang membuka jadwal sertifikasi tersebut. Informasi yang ditampilkan yaitu adalah nama LSP, logo, nomor SK Lisensi, nomor lisensi, jenis LSP, email resmi, alamat *website*, masa berlaku lisensi, serta status legalitas

SK Lisensinya. Tampilan ini bertujuan memberikan transparansi kepada calon peserta sertifikasi, sekaligus meningkatkan kepercayaan terhadap LSP yang bersangkutan.

### 3.2.3 Pengembangan Skema dan Pembuatan Fitur Revisi Skema

Pengembangan pada bagian ini mencakup fitur skema serta penambahan fitur untuk mendukung proses revisi skema secara sistematis.

#### 3.2.3.1 Pengembangan Skema



```
5 + use Illuminate\Support\Facades\Schema;
6 +
7 + return new class extends Migration
8 + {
9 +     public function up()
10 +     {
11 +         Schema::table('blueprint_stable', function (Blueprint $table) {
12 +             $table->boolean('greenjob')->default(false)->after('');
13 +         });
14 +     }
15 + };
16 +
17 +
18 +
19 +
20 +
21 +
22 +
23 +
24 +
25 +
26 +
27 +
28 +
29 +
30 +
31 +
32 +
33 +
34 +
35 +
36 + };
```

Gambar 3.20 Menambahkan Input GreenJob Pada Form Skema

Gambar 3.20 menunjukkan proses penambahan kolom greenjob pada tabel skema melalui fitur migration pada Laravel. Setelah itu, mahasiswa menambahkan input GreenJob pada formulir pembuatan skema serta mengatur pengelolaan datanya agar dapat tersimpan dengan baik ke dalam database. Penambahan ini dilakukan sebagai respon terhadap permintaan agar setiap skema mencantumkan informasi apakah mendukung konsep GreenJob atau tidak. Untuk memenuhi kebutuhan tersebut, tugas yang dilakukan adalah menambahkan checkbox sebagai indikator dukungan terhadap GreenJob, dan ada tautan informasi untuk menjelaskan konsep GreenJob lebih lanjut kepada pengguna. Implementasi ini tidak hanya memenuhi permintaan dari pemerintah,

tetapi juga meningkatkan kepatuhan sistem terhadap standar keberlanjutan dan praktik ramah lingkungan yang sedang digaungkan oleh berbagai institusi nasional.

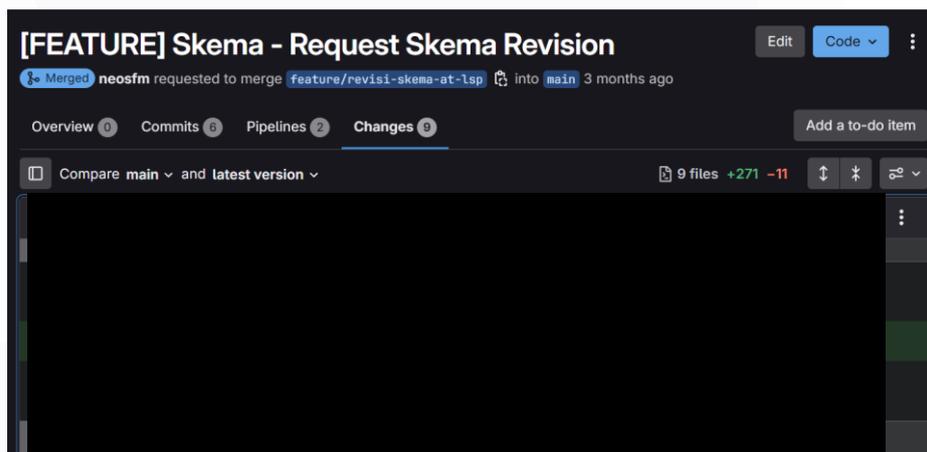


```
103 103      <a href="{{route('skema.master.edit', $item->id)}}" class="btn btn-bsp-edit button-se">
104 104          Ubah
105 105      </a>
106 106      +
107 107      <a href="{{route('skema.master.delete', $item->id)}}" class="btn button-se btn-bsp-delete">
108 108          Hapus
109 109      </a>
110 110  +
111 111  </td>
112 112  </tr>
113 113  @endforeach
114 114  </tbody>
```

Gambar 3.21 Menambahkan Button Untuk Menghapus Skema

Gambar 3.21 menunjukkan penambahan tombol delete pada daftar skema. Fitur ini memungkinkan admin untuk menghapus skema secara langsung dari tampilan list.

### 3.2.3.2 Pembuatan Fitur Revisi Skema



Gambar 3.22 Fitur Revisi Skema

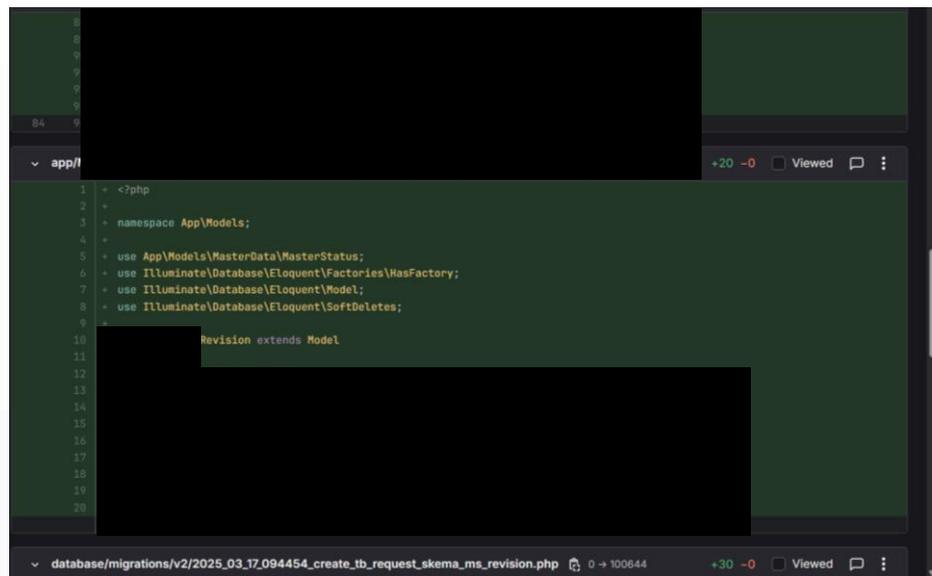
Gambar 3.22 menampilkan fitur penting yang memungkinkan Lembaga Sertifikasi Profesi (LSP) untuk mengajukan revisi terhadap skema yang sudah ada. Alur revisi dimulai ketika LSP mengajukan permohonan revisi, yang secara otomatis mengubah status skema menjadi “revisi”. Selanjutnya, admin akan memvalidasi pengajuan tersebut. Jika revisi disetujui, status skema akan kembali menjadi aktif dan dapat digunakan seperti biasa. Namun apabila ditolak, admin dapat memberikan alasan atau keterangan yang menjelaskan penyebab penolakan. Informasi ini akan ditampilkan kepada pihak LSP agar

mereka dapat memahami kekurangan yang ada, memperbaikinya, dan kembali mengajukan revisi. Siklus ini dapat terus berlanjut hingga pengajuan revisi dinyatakan valid dan diterima oleh admin. Dengan fitur ini, proses revisi skema menjadi lebih terstruktur, transparan, dan terdokumentasi dengan baik.

```
248 +
249 + public function reject
250 +   try {
251 +     $skema = $this->repo->find($id);
252 +     $rejectedStatus = MasterStatus::where('nama', 'Revisi Skema ditolak')->first()->id;
253 +
254 +     if (!isset($skema)) return CustomResponse::notFound();
255 +
256 +     // Update skema status
257 +     $update = $this->repo->update([
258 +       'status' => $rejectedStatus,
259 +     ], $id);
260 +
261 +     // Delete if revision already exist
262 +     $revisionExist =
263 +     if ($revisionExist) {
264 +       $revisionExist->delete();
265 +     }
266 +
267 +     // Save the rejection reason
268 +     $skemaRevision =
269 +     $skemaRevision->idskema_ms = $id;
270 +     $skemaRevision->keterangan_revisi = $request->reason;
271 +     $skemaRevision->save();
272 +
273 +     if($update){
274 +       'Skema - Pengajuan Revisi Skema Ditolak',
275 +       return CustomResponse::update($update);
276 +     }
277 +
278 +     return CustomResponse::success(['message' => 'Pengajuan revisi berhasil ditolak']);
279 +   } catch (\Throwable $th) {
280 +
```

Gambar 3.23 Fungsi Reject Untuk Penolakan Revisi Skema

Gambar 3.23 menampilkan implementasi teknis lanjutan dari fitur revisi skema. Salah satu fungsi penting yang dikembangkan untuk fitur revisi skema adalah fungsi reject, yang menangani proses penolakan revisi skema oleh admin. Fungsi ini mengubah status skema menjadi “revisi ditolak”, menghapus data revisi lama jika ada, menyimpan alasan penolakan, dan yang tak kalah penting, mengirimkan notifikasi ke LSP atau admin melalui repository khusus untuk notifikasi. Dengan sistem ini, setiap pihak terkait, baik LSP maupun admin, akan selalu mendapatkan informasi terkini mengenai status revisi melalui notifikasi yang muncul sesuai perannya masing-masing, sehingga proses koordinasi menjadi lebih efisien dan transparan.

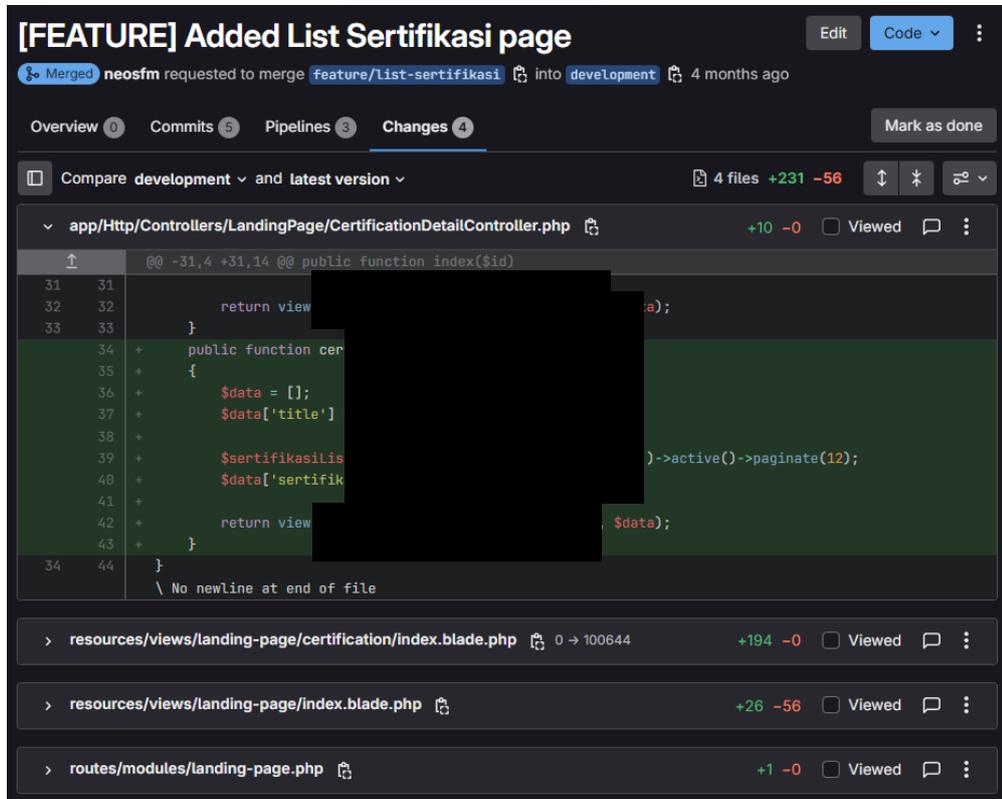


```
1 + <?php
2 +
3 + namespace App\Models;
4 +
5 + use App\Models\MasterData\MasterStatus;
6 + use Illuminate\Database\Eloquent\Factories\HasFactory;
7 + use Illuminate\Database\Eloquent\Model;
8 + use Illuminate\Database\Eloquent\SoftDeletes;
9 +
10 + Revision extends Model
11 +
12 +
13 +
14 +
15 +
16 +
17 +
18 +
19 +
20 +
```

Gambar 3.24 Model Revisi Skema

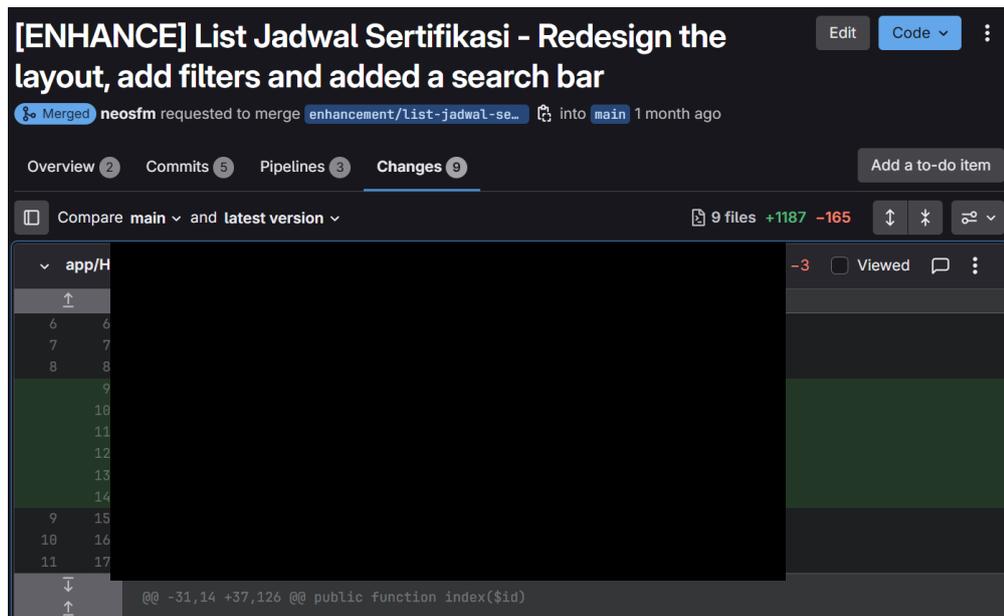
Gambar 3.24 menggambarkan pengembangan lanjutan dari fitur revisi skema yang memerlukan penyesuaian pada struktur data sistem. Dalam implementasinya, dibuat sebuah tabel baru yang berfungsi untuk menampung data pengajuan revisi dari LSP, termasuk keterangan alasan revisi. Tabel ini terhubung langsung dengan data skema utama agar riwayat revisi dapat dikelola dengan baik. Selain itu, ditambahkan juga status baru, yaitu "Revisi Skema Ditolak", ke dalam tabel master status. Status ini digunakan untuk menandai skema yang revisinya tidak disetujui, sehingga sistem dapat menampilkan informasi status secara akurat dan konsisten pada seluruh antarmuka yang relevan. Pembaruan ini menjadi bagian penting dalam mendukung alur pengajuan revisi yang lebih lengkap dan terstruktur.

### 3.2.4 Pembuatan Halaman List Sertifikasi



Gambar 3.25 Merge Request Pembuatan Konsep Awal Halaman List Sertifikasi

Gambar 3.25 menunjukkan implementasi konsep awal terhadap fitur baru berupa halaman “List Sertifikasi” pada sistem. Mahasiswa menambahkan method `certification()` pada Controller yang bertugas mengambil data sertifikasi aktif dari model list jadwal sertifikasi beserta relasinya dengan LSP yang berkaitan, lalu meneruskannya ke view halaman list sertifikasi. Tampilan halaman tersebut menampilkan daftar sertifikasi secara terstruktur dan dipaginasi per-12 item untuk konsep awal. Selain itu, mahasiswa juga menambahkan rute baru pada file untuk mengarahkan URL ke method `certification()` tersebut. Tentunya tombol navigasi juga ditambahkan agar dapat menuju ke halaman ini, ditambahkan pada halaman utama yang merujuk ke rute baru tersebut. Dengan adanya fitur ini, pengguna publik dapat dengan mudah melihat daftar jadwal sertifikasi yang tersedia tanpa harus login terlebih dahulu.



Gambar 3.26 Merge Request Fitur List Sertifikasi

Gambar 3.26 menunjukkan kontribusi saya dalam pengembangan halaman list sertifikasi, di mana saya diberikan kesempatan untuk mengerjakan fitur dengan ruang lingkup kode yang cukup besar, terlihat dari perubahan sebanyak +1187 dan -165 baris kode. Fitur ini memiliki peran penting dalam sistem, karena jika halaman landing page hanya menampilkan beberapa jadwal sertifikasi dalam bentuk carousel, maka halaman ini menyajikan daftar lengkap seluruh jadwal yang tersedia. Di dalamnya terdapat fungsionalitas pagination, filter berdasarkan berbagai parameter, serta fitur search untuk memudahkan pengguna menemukan jadwal sertifikasi yang relevan dengan cepat dan efisien.

```

$data] = $request->get( // Nama LSP, Jenis LSP, Nama Skema, Nama Unit Skema, Tanggal Mulai, KBLI, KBJI
$data] = $request->get( ? '' // Provinsi
$data] = $request->get( '' // Jenis LSP
$data] = $request->get( ? '' // Kementerian
$data] = $request->get( ? '' // Harga Minimum
$data] = $request->get( '' // Harga Maksimum
$data] = $request->ge ?? '' // Tanggal Mulai
$data] = $request->get( '' // Tanggal Selesai

$data] = $this->jadwal Search($data);

```

Gambar 3.27 Controller Untuk Mengelola Filter Pada Halaman List Sertifikasi

Gambar 3.27 menunjukkan bagian controller yang digunakan untuk menangani permintaan filter pada halaman list sertifikasi. Mahasiswa diminta

untuk mengelola berbagai parameter pencarian, seperti nama LSP, jenis LSP, nama skema, unit kompetensi, provinsi, kementerian, kisaran harga (minimum dan maksimum), serta rentang tanggal pelaksanaan sertifikasi. Setiap parameter diambil dari request pengguna menggunakan method `$request->get()` dan disimpan ke dalam variabel array `$data` untuk kemudian diproses lebih lanjut. Data yang telah dikumpulkan akan diteruskan ke fungsi pencarian pada service layer (`$this->jadwalX->XSearch($data)`), yang bertanggung jawab melakukan pencarian dan pemfilteran data berdasarkan kriteria yang telah ditentukan. Proses ini dirancang agar sistem dapat menampilkan daftar jadwal sertifikasi yang relevan dan sesuai dengan kebutuhan pengguna.

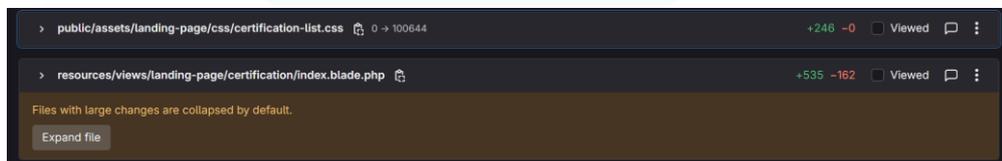
```
$query->where(function($mainQuery) use ($data) {
    // Search filter
    if (!empty($data)) {
        $mainQuery->where(function($searchQuery) use ($data) {
            $searchQuery->whereHas('skema', function($sq) { // Nama Skema
                $sq->where('nama', $data['nama']);
            })
            ->orWhereHas('lsp', function($sq) { // Nama LSP
                $sq->where('nama', $data['nama']);
            })
            ->orWhereHas('unit_kompetensi', function($sq) { // KBLI
                $sq->where('unit_kompetensi', $data['unit_kompetensi']);
            })
            ->orWhereHas('jenis_lsp', function($sq) { // KBJI
                $sq->where('jenis_lsp', $data['jenis_lsp']);
            })
            ->orWhereHas('unit_skema', function($sq) { // Unit Skema
                $sq->where('unit_skema', $data['unit_skema']);
            })
        });
    }

    // Province filter
    if (!empty($data['provinsi'])) {
        $provinces = $mainQuery->whereHas('provinsi', function($sq) {
            $sq->where('provinsi', $data['provinsi']);
        });
        $mainQuery->whereHas('provinsi', function($sq) {
            $sq->where('provinsi', $data['provinsi']);
        });
    }

    // LSP Type filter
    if (!empty($data['jenis_lsp'])) {
        $mainQuery->whereHas('jenis_lsp', function($sq) {
            $sq->where('jenis_lsp', $data['jenis_lsp']);
        });
    }
});
```

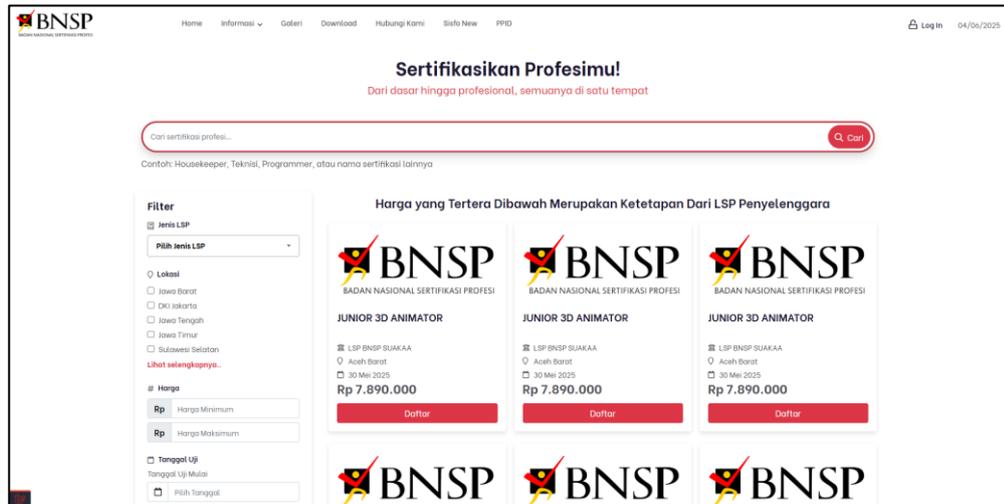
Gambar 3.28 Kode Filter Jadwal Sertifikasi

Gambar 3.28 menampilkan bagian backend dari halaman list sertifikasi yang menangani proses pengambilan dan penyaringan data jadwal sertifikasi yang sedang aktif. Fungsi yang dikembangkan bertujuan untuk menampilkan data berdasarkan kondisi real-time dari database, serta memberikan respons terhadap permintaan filter dari pengguna. Berbeda dengan halaman utama yang hanya menampilkan beberapa data secara terbatas melalui carousel, halaman ini menyediakan sistem filter yang jauh lebih kompleks dan fleksibel. Adapun filter yang tersedia mencakup: kata kunci pencarian (nama skema, LSP, KBLI, KBJI, unit skema), provinsi atau kota lokasi LSP, jenis LSP, kementerian, rentang harga, dan rentang tanggal pelaksanaan sertifikasi. Seluruh data hasil filter ditampilkan secara terstruktur dengan sistem pagination, sehingga pengguna dapat menavigasi dan menemukan jadwal yang sesuai dengan kebutuhan mereka secara efisien. Fitur ini memperkuat fungsionalitas halaman dan memberikan pengalaman pengguna yang lebih baik dalam menjelajahi layanan sertifikasi yang ditawarkan.



Gambar 3.29 Index Blade dan CSS Halaman List Sertifikasi

Gambar 3.29 memperlihatkan perubahan signifikan yang saya lakukan pada halaman indeks dari list sertifikasi. Perubahan tercatat sebanyak +532 dan -162 baris kode pada blade dan +246 pada CSS yang digunakan pada halaman tersebut, yang menunjukkan proses redesign mendalam pada struktur dan tampilan halaman. Desain ulang ini bertujuan untuk menyelaraskan tampilan dengan identitas visual BNSP, terutama dengan penerapan tema warna merah sebagai warna utama. Hasil akhirnya adalah halaman yang lebih modern, responsif, dan mudah dinavigasi, sekaligus meningkatkan pengalaman pengguna dalam menjelajahi jadwal sertifikasi yang tersedia.



Gambar 3.30 Halaman List Sertifikasi

Gambar 3.30 menampilkan hasil akhir dari implementasi halaman list sertifikasi yang telah saya kembangkan secara menyeluruh. Tampilan halaman menunjukkan tata letak yang terstruktur dengan baik, di mana bagian kiri terdapat panel filter dan di bagian atas menampilkan search bar yang memungkinkan pengguna untuk menyaring data berdasarkan berbagai parameter yang telah diimplementasikan sebelumnya. Di bagian tengah halaman menampilkan daftar jadwal sertifikasi dalam format yang rapi dan informatif, memudahkan pengguna untuk melihat detail setiap sertifikasi yang tersedia. Bagian bawah halaman dilengkapi dengan sistem pagination yang memungkinkan navigasi antar halaman data dengan lancar. Salah satu aspek penting dari pengembangan ini adalah penerapan desain responsif yang memastikan halaman dapat diakses dengan optimal pada berbagai perangkat, termasuk *mobile device*. Implementasi *mobile-friendly* ini menjamin pengalaman pengguna yang konsisten baik saat diakses melalui desktop maupun smartphone, sehingga meningkatkan aksesibilitas dan kemudahan penggunaan platform sertifikasi BNSP bagi seluruh kalangan pengguna.

### 3.2.5 Pengembangan Asesor Kompetensi dan Asesi

Bagian ini mencakup pengembangan fitur yang berkaitan dengan asesor kompetensi dan asesi dalam *website* BNSP.

### 3.2.5.1 Pengembangan Asesor Kompetensi

```
Route::middleware(['authenticated', 'role: Koordinator Sertifikasi Kompetensi'])->group(function() {
    // Route Select All Profile Member
    Route::get("/delete-meds", function() {
    Route::get('check-nik', function() {
```

Gambar 3.31 Perbaikan Hak Akses untuk Pencarian NIK oleh Koordinator Sertifikasi

Gambar 3.31 menunjukkan perbaikan pada hak akses untuk fitur pencarian NIK pada halaman edit data asesor oleh user dengan level Koordinator Sertifikasi Kompetensi. Sebelumnya, fitur pencarian NIK yang mengambil data dari sistem SIAPKerja tidak dapat diakses oleh user Koordinator Sertifikasi Kompetensi, karena level tersebut belum dimasukkan ke dalam middleware yang mengatur hak akses terhadap rute check-nik.

```
<option value="">-- Pilih Provinsi --</option>
<option value="default">-- Pilih Provinsi --</option>
@foreach ($provinsi ?? [] as $item)
    <option
+1015,7 @@
    url:
    succ
    HideLoader();
    ${id}.html('').html('<option>-- Pilih Kabupaten/Kota --</option>');
    ${id}.html('').html('<option value="default">-- Pilih Kabupaten/Kota --</option>');
    let selected = '';
+1446,7 @@
    });
    $('#${id_kota}').html('<option value="">-- Pilih Kabupaten/Kota --</option>');
    $('#${id_kota}').html('<option value="default">-- Pilih Kabupaten/Kota --</option>');
    $('#${id_kota}').select2({
        width: '100%',
    }).val('default');
+1815,12 @
cript>
document.getElementById("form_askom").addEventListener("submit", function(event){
    event.preventDefault()
    if (qSelect('#kode_propinsi_ktp').value === "default") {
        return swalAlert('Harap Pilih Provinsi Terlebih Dahulu!');
    }
    if (qSelect('#kab_ktp').value === "default") {
        return swalAlert('Harap Pilih Kabupaten/Kota Terlebih Dahulu!');
```

Gambar 3.32. Perbaikan Select Provinsi dan Kabupaten/Kota Setelah Pencarian NIK

Gambar 3.32 menunjukkan perbaikan yang dilakukan untuk mengatasi bug yang menyebabkan opsi provinsi tidak terpilih secara otomatis setelah melakukan pencarian NIK, sehingga tampilan form menjadi tidak bagus karena opsi yang tidak terpilih mengakibatkan tinggi input select terlalu kecil. Solusinya adalah menetapkan opsi default secara eksplisit agar select option kembali terisi dengan benar.

```
</select>
</div>
<div class="col-md-6 mb-3">
  <label class="form-label">Foto: <small class="text-danger">*</small></label>
  <input type="text" value="" />
  <label class="form-label">Upload Foto Dengan Background Merah: <small class="text-danger">*</small></label>
  <input
    type="file"
    name="foto_users"
  />
</div>
```

Gambar 3.33 Perubahan Label Input Foto pada Formulir Asesor Kompetensi

Gambar 3.33 menunjukkan perubahan minor pada label input untuk mengunggah foto dalam formulir Asesor Kompetensi. Sebelumnya, label hanya bertuliskan “Foto”, namun diubah menjadi “Upload Foto Dengan Background Merah” untuk memberikan instruksi yang lebih jelas kepada pengguna. Perubahan ini bertujuan agar calon asesor memahami standar foto yang diminta sistem sejak awal, sehingga meminimalkan kesalahan pengunggahan dan memastikan konsistensi data visual di seluruh platform.



```

function chooseAsesorNotValid(id) {
  Swal.fire({
    title: "Pastikan Asesor Kompetensi yang anda pilih sudah benar!",
    text: "Apakah Anda yakin ingin menambahkan Asesor Kompetensi ini?",
    icon: "warning",
    showCancelButton: true,
    confirmButtonText: "Ya",
    cancelButtonText: 'Tidak',
  }).then((result) => {
    if (result.isConfirmed) {
      ShowLoader();
      $.ajax({
        type: "POST",
        url: route('asesor_kompetensi.storeAsesorRequest'),
        data: {
          asesor: id,
          lsp: '{ app('session')->get('user')->idlsp }',
          _token: '{ csrf_token() }'
        },
        success: function(response){
          if (response.status == 'success') {
            HideLoader();
            Swal.fire('Berhasil!', 'Asesor Kompetensi Berhasil Ditambahkan', 'success').then((result) => {
              if (result.isConfirmed) {
                window.location.href = route('asesor_kompetensi.edit', {id: response.data.id});
              }
            });
          } else {
            HideLoader();
            Swal.fire('Gagal', 'Data Gagal Disimpan', 'error');
          }
        },
        error: function(err) {
          Swal.fire('Gagal', 'Ada Error Pada Sistem! Silahkan Coba Lagi Nanti', 'error');
          HideLoader();
        }
      });
    }
  });
}

```

Gambar 3.34 Penggantian Alert Biasa Menjadi SweetAlert pada Proses Penambahan Asesor Kompetensi

Gambar 3.34 menunjukkan peningkatan tampilan UI pada proses penambahan Asesor Kompetensi. Sebelumnya, sistem menggunakan fungsi alert() standar dari JavaScript yang tampilannya sudah jadul dan terlalu sederhana. Kini, alert tersebut diganti dengan SweetAlert (Swal), yang menampilkan dialog konfirmasi dengan desain lebih modern dan interaktif. Dialog ini memberikan peringatan visual yang lebih jelas kepada pengguna sebelum menambahkan asesor, serta notifikasi sukses atau gagal yang lebih mudah dipahami. Peningkatan ini bertujuan untuk meningkatkan pengalaman pengguna dan mengurangi risiko kesalahan klik.

```

Swal.fire({
  icon: 'success',
  title: 'Success',
  text: 'Data berhasil disimpan'
}, {
  text: 'Data berhasil disimpan',
  showConfirmButton: false,
  timer: 2000,
}).then(() => {
  const data = response.data;

  @if (in_array(app('session')->get('role'), ['LSP'])) {
    window.location.href = route('asesor.kompetensi');
  } @else {
    window.location.href = route('asesor.kompetensi.validate', {id: data.id});
  } @endif
});
},
error: function(xhr) {

```

Gambar 3.35 *Redirect* Otomatis Berdasarkan Peran Setelah Menyimpan Data Asesor Kompetensi

Gambar ini menunjukkan logika *redirect* yang berbeda tergantung pada peran pengguna setelah berhasil menyimpan perubahan pada Asesor Kompetensi. Setelah data berhasil disimpan dan notifikasi berhasil ditampilkan menggunakan SweetAlert, pengguna diarahkan ke halaman yang sesuai. Jika pengguna berasal dari peran LSP, maka akan diarahkan kembali ke halaman daftar Asesor Kompetensi. Namun, jika pengguna memiliki hak akses sebagai validator (misalnya Koordinator Sertifikasi atau Super Admin), maka akan diarahkan ke halaman khusus untuk memvalidasi data asesor. Pengaturan ini bertujuan untuk mempercepat alur kerja sesuai tanggung jawab masing-masing peran.

```
app/Http/Requests/ Request.php
@@ -40,7 +40,8 @@ public function rules()
40     $rules = [
41         [REDACTED]
42     ];
43     +
44     + 'profile.nik' => 'required|numeric',
45     ];
46     [REDACTED] {
47     }

@@ -69,6 +70,7 @@ public function messages()
70     [REDACTED]
71     [REDACTED]
72     [REDACTED]
73     + 'profile.nik.required' => 'Silahkan input NIK terlebih dahulu.',
74     ];
75     }
76 }
```

Gambar 3.36 Penambahan Validasi NIK pada Formulir Asesor Kompetensi

Gambar 3.36 menunjukkan penambahan aturan validasi pada proses penyimpanan data Asesor Kompetensi. Salah satu penyesuaian penting adalah penambahan validasi di dalam FormRequest terhadap input NIK agar tidak kosong dan harus berupa angka. Validasi ini ditambahkan dalam method rules() untuk memastikan bahwa data yang dikirimkan memenuhi format yang benar. Selain itu, pesan kesalahan khusus juga ditambahkan pada method messages() untuk memberikan pesan yang lebih informatif kepada pengguna. Perubahan ini bertujuan meningkatkan kualitas dan kelengkapan data asesor yang didaftarkan melalui sistem.

```

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::table('...', function (Blueprint $table) {
            $table->text('revision')->after('...')->nullable();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::table('...', function (Blueprint $table) {
            $table->dropColumn('revision');
        });
    }
};

```

Gambar 3.37 Penambahan Kolom Catatan Revisi Asesor Kompetensi pada Database

Gambar 3.37 menunjukkan penambahan kolom baru pada tabel asesor yang digunakan untuk menyimpan catatan revisi apabila seorang asesor ditolak dalam proses validasi. Perubahan ini dilakukan melalui migration, dengan menambahkan kolom bernama revision bertipe teks dan bersifat opsional. Kolom ini bertujuan menampung keterangan alasan revisi agar dapat ditampilkan kembali dan menjadi acuan perbaikan bagi pihak LSP atau pihak terkait lainnya.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

```

@if ((isset($revision_disabled) != true)
<div class="col-12 mt-3">
  <div class="form-check mb-3">
    <input class="form-check-input" type="checkbox" id="needRevision">
    <label class="form-check-label" id="revisionLabel" for="needRevision">
      Revisi Asesor Kompetensi ini
    </label>
  </div>
  <div class="form-group d-none" id="descriptionContainer">
    <label for="revision">Deskripsi Revisi</label>
    <textarea class="form-control" id="revision" name="revision" rows="3"></textarea>
  </div>
</div>
@endif
@if (in_array(session('user')->
  @if (isset($data->revision)
    <div class="alert alert-danger">
      Keterangan Revisi: <b>{{ $data->revision }}</b>
    </div>
  @endif
  @if (isset($data->revision) && in_array(
    <div class="alert alert-danger">
      Keterangan Revisi Sebelumnya: <b>{{ $data->revision }}</b>
    </div>
  @endif
@endif
@endif

```

Gambar 3.38 Penambahan Input Catatan Revisi Pada Formulir Validasi Asesor Kompetensi

Gambar 3.38 menampilkan penyesuaian pada formulir validasi Asesor Kompetensi. Ketika asesor ditolak atau diberi status revisi, sistem akan memunculkan input tambahan berupa textarea untuk mengisi alasan atau catatan revisi. Fitur ini hanya dapat digunakan oleh pengguna dengan hak akses tertentu. Selain itu, jika status asesor sudah dalam kondisi revisi, sistem juga menampilkan catatan revisi sebelumnya agar dapat dijadikan referensi oleh validator atau pihak terkait.

```

// Remove revision if data is correct
if(array_key_exists('revision', $data) && is_null($data['revision'])) {
  $this->model
  ->where('id', $id)
  ->update(['revision' => null]);
}

```

Gambar 3.39 Penghapusan Otomatis Catatan Revisi Jika Status Sudah Valid

Gambar 3.39 menjelaskan proses otomatisasi yang terjadi saat data asesor diperbarui. Jika status validasi asesor telah berubah ke kondisi valid, maka sistem secara otomatis akan menghapus konten dari kolom revision di database. Tujuannya adalah untuk menjaga kebersihan

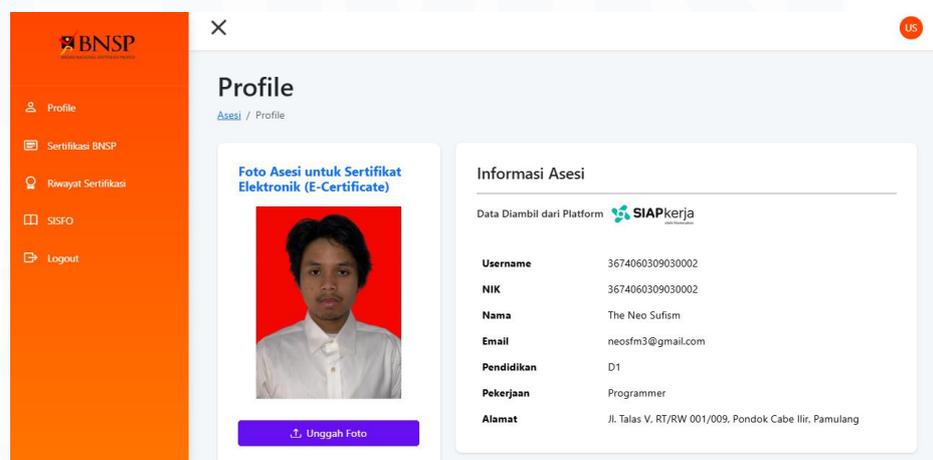
data dan memastikan bahwa catatan revisi hanya disimpan selama proses perbaikan berlangsung, tidak permanen jika sudah tidak relevan lagi.

### 3.2.5.2 Pengembangan Asesi

```
<li>
  <a href="{{ route(' [REDACTED] '}}'" class="nav-link">
    <i class="bi bi-card-heading"></i>
    <span class="menu-text">Sertifikasi BNSP</span>
  </a>
</li>
<li class="active">
  <a href="{{ route(' [REDACTED] '}}'" class="nav-link">
    <i class="bi bi-award"></i>
    <span class="menu-text">Sertifikasi</span>
    <span class="menu-text">Riwayat Sertifikasi</span>
  </a>
</li>
```

Gambar 3.40 Kode Penambahan Menu Pada Sidebar Asesi

Gambar 3.40 menunjukkan penambahan menu baru bernama "Sertifikasi BNSP" yang mengarahkan pengguna ke halaman list sertifikasi pada *website* BNSP. Sebelumnya, asesi hanya memiliki akses ke menu "Sertifikasi" atau lebih tepatnya riwayat sertifikasi yang pernah diikuti. Dengan adanya menu baru ini, pengguna kini dapat dengan mudah menjelajahi dan mendaftarkan jadwal sertifikasi aktif tanpa keluar dari sistem asesi. Implementasi ini bertujuan untuk meningkatkan kenyamanan dan aksesibilitas bagi pengguna dalam menemukan informasi jadwal sertifikasi terkini.



Gambar 3.41 Halaman Asesi

Gambar 3.41 menampilkan tampilan halaman Asesi, di mana secara default pengguna diarahkan ke menu "Profile". Menu ini memiliki informasi pribadi asesi serta menyediakan fitur untuk mengunggah pas foto, yang nantinya akan digunakan pada sertifikat elektronik. Selain itu, terdapat menu "Riwayat Sertifikasi" yang memungkinkan asesi untuk melihat seluruh sertifikasi yang telah atau sedang diikuti, termasuk akses untuk melihat sertifikat digital yang telah diterbitkan. Di sidebar, selain menu yang telah disebutkan, tersedia juga menu "Sertifikasi BNSP" yang mengarahkan ke halaman list sertifikasi yang sedang aktif, menu "SISFO" untuk asesi yang juga berperan sebagai asesor kompetensi agar dapat mengakses sistem SISFO yang baru, serta menu "Logout" untuk keluar dari akun. Tampilan ini dirancang agar asesi dapat dengan mudah mengelola data dan akses sertifikasinya dalam satu tempat yang terpusat.

```
body: body,  
variableMessage: 'data',  
withDefaultAlertAfterSuccessRequest: false,  
callbackAfterResponse: response => {  
  if(response.status === 'success') {  
    location.replace(`${ route('landing-page.certification-list') }`)  
    Swal.fire({  
      icon: 'success',  
      title: 'Berhasil',  
      text: response.message,  
      showConfirmButton: false,  
      timer: 2000  
    }).then(() => {  
      location.replace(`${ route('asesi.sertifikasi') }`)  
    });  
  } else {  
    Swal.fire({  
      icon: 'error',  
      title: 'Gagal',  
      text: response.message,  
    });  
  }  
}
```

Gambar 3.42 Pendaftaran Sertifikasi Berhasil

Gambar 3.42 menunjukkan potongan kode yang digunakan untuk mengarahkan asesi ke halaman “Riwayat Sertifikasi” setelah proses pendaftaran berhasil. Tujuan dari pengalihan ini adalah agar asesi dapat

langsung melihat status dari sertifikasi yang telah mereka daftarkan, apakah telah diterima, ditolak, atau masih menunggu proses validasi.

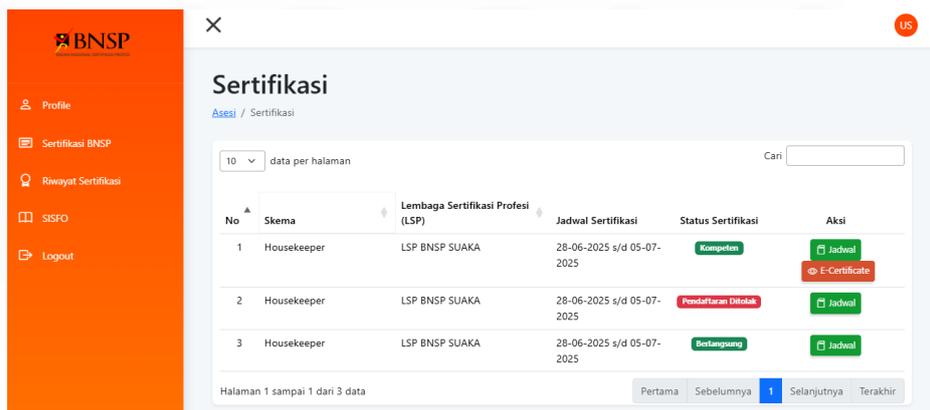
```

if ($row->is_certificate_issued == 1) {
    $html = '<span class="badge bg-success">Kompeten</span>';
} elseif ($row->is_approved == null) {
    $html = '<span class="badge bg-warning">Menunggu Persetujuan</span>';
} elseif ($row->is_approved == 0) {
    $html = '<span class="badge bg-danger">Pendaftaran Ditolak</span>';
} elseif ($row->is_approved == 1 && $row->jadwal->status == 2302) {
    $html = '<span class="badge bg-warning">Terjadwal</span>';
} elseif ($row->is_approved == 1 && $row->jadwal->status == 2303) {
    $html = '<span class="badge bg-success">Berlangsung</span>';
} elseif (($row->is_attend == 0 || $row->is_pass == 0) && $row->jadwal->status != 2302 && $row->jadwal->status != 2306) {
    $html = '<span class="badge bg-danger">Tidak Hadir | Belum Kompeten</span>';
} elseif ($row->is_pass == 0 && ($row->jadwal->status == 2306 || $row->jadwal->status == 2311)) {
    $html = '<span class="badge bg-dark">Belum Kompeten</span>';
} elseif ($row->is_attend == 1 && $row->is_pass == 1 && ($row->jadwal->status == 2304 || $row->jadwal->status == 2306)) {
    $html = '<span class="badge bg-success">Menunggu Penerbitan Sertifikat</span>';
} elseif ($row->getDataAndCheckIfCertificateExists('nama') && $row->jadwal->status == 2311) {
    $html = '<span class="badge bg-danger">Sertifikat Dicabut</span>';
}
}

```

Gambar 3.43 Status Sertifikasi pada Halaman Riwayat Sertifikasi

Kode pada Gambar 3.43 mendefinisikan berbagai status yang dapat dimiliki oleh sebuah sertifikasi dalam sistem. Terdapat sembilan status yang akan ditampilkan, yaitu Kompeten, Menunggu Persetujuan, Pendaftaran Ditolak, Terjadwal, Berlangsung, Tidak Hadir | Belum Kompeten, Belum Kompeten, Menunggu Penerbitan Sertifikat, dan Sertifikat Dicabut. Status ini ditampilkan pada halaman Riwayat Sertifikasi untuk memberikan informasi yang jelas terkait proses dan hasil sertifikasi yang diikuti oleh asesi.



Gambar 3.44 Halaman Riwayat Sertifikasi

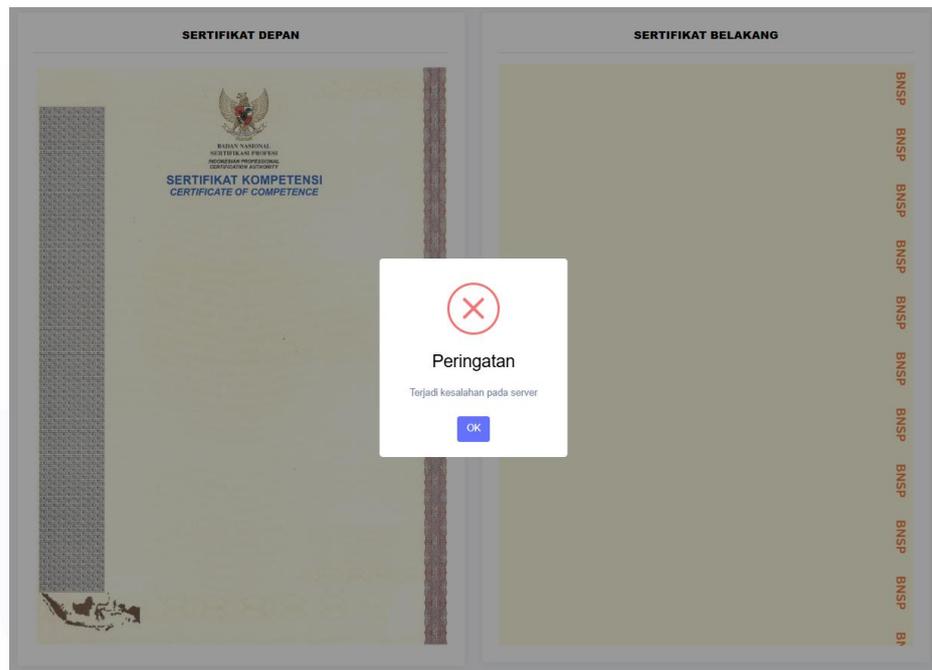
Gambar 3.44 menampilkan halaman Riwayat Sertifikasi, tempat asesi dapat melihat semua jadwal sertifikasi yang telah mereka ikuti dalam bentuk tabel. Informasi yang disajikan meliputi nama skema,

Lembaga Sertifikasi Profesi (LSP), jadwal sertifikasi, status sertifikasi, serta kolom aksi. Jika sertifikasi sudah mencapai status “Kompeten”, maka tombol E-Certificate akan muncul agar asesi dapat mengakses dan mengunduh sertifikat mereka dalam bentuk digital.

```
} catch (\Exception $e) {  
    if (isset($dataSertifikat) && empty($dataSertifikat->ketua_lsp)) {  
        return CustomResponse::error('Ketua LSP tidak ditemukan');  
    }  
    if(!isset($asesi)) {  
        return CustomResponse::error('Asesi tidak ditemukan');  
    }  
    if(!isset($dataSertifikat)) {  
        return CustomResponse::error('Sertifikat tidak ditemukan');  
    }  
  
    return CustomResponse::error('Terjadi kesalahan pada server');  
}
```

Gambar 3.45. Snippet Kode Penanganan Error

Gambar 3.45 memperlihatkan potongan kode yang bertanggung jawab terhadap penanganan berbagai kondisi kesalahan saat mengambil data sertifikasi. Beberapa pesan error yang ditangani antara lain adalah "Ketua LSP tidak ditemukan", "Asesi tidak ditemukan", "Sertifikat tidak ditemukan", serta "Terjadi kesalahan pada server". Penanganan ini penting untuk memberikan pesan yang tepat dan jelas kepada pengguna saat terjadi kegagalan sistem.



Gambar 3.46. Halaman Sertifikat Elektronik dengan Peringatan Error

Gambar 3.46 memperlihatkan tampilan halaman Sertifikat Elektronik ketika terjadi kegagalan saat proses pengambilan data. Dalam contoh yang ditampilkan, muncul pesan error “Terjadi kesalahan pada server” yang menandakan bahwa proses render data tidak berhasil dilakukan. Pesan ini disesuaikan dengan kondisi error yang ditangani di sisi backend agar pengguna memahami bahwa terjadi kendala teknis.

### 3.2.6 Pengembangan Profile LSP

```
resources/views/components/profile/index.blade.php
@@ -36,6 +36,24 @@
36 36
37 37     @extends('master')
38 38     @section('title', 'Profile')
39 + @push('style')
40 + <style>
41 +     @media (min-width: 1600px) {
42 +         .col-xxxl-1 {
43 +             flex: 0 0 auto;
44 +             width: 8.33%;
45 +         }
46 +         .col-xxxl-7 {
47 +             flex: 0 0 auto;
48 +             width: 58.33%;
49 +         }
50 +         .col-xxxl-8 {
51 +             flex: 0 0 auto;
52 +             width: 66.67%;
53 +         }
54 +     }
55 + </style>
56 + @endpush
```

Gambar 3.47 Kode CSS Media Query untuk No Sertifikat Lisensi LSP and No SK LSP Terakhir Pada Profile

Gambar 3.47 menunjukkan penambahan aturan media query dalam halaman profil LSP. Tujuan dari penambahan ini adalah untuk meningkatkan responsivitas tampilan elemen input “No. Sertifikat Lisensi LSP” dan “No. SK LSP Terakhir” pada layar dengan resolusi sangat besar (minimal 1600px). Media query ini memperkenalkan kelas-kelas baru seperti `.col-xxxl-1`, `.col-xxxl-7`, dan `.col-xxxl-8` yang mengatur lebar kolom dengan presisi pada perangkat berlayar lebar. Dengan demikian, tampilan form tetap proporsional dan mudah dibaca, terutama pada perangkat dengan layar besar, tanpa mengganggu tata letak di resolusi standar. Perubahan ini bertujuan menjaga kenyamanan pengguna dan memastikan elemen input tampil dengan benar, karena sebelumnya kolom input tersebut terpotong dan tidak sesuai dengan desain yang diharapkan.

```

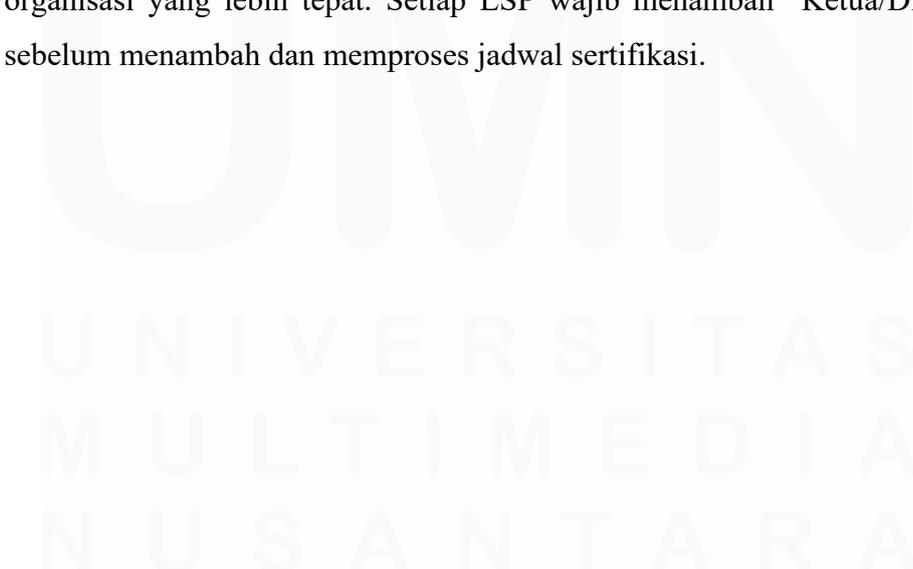
  app/Models/Profile/LSP.php
  @@ -155,7 +155,7 @@ public function sdm_lsp()
  155 155
  156 156     public function
  157 157     {
  158 -     $data =
  158 +     $data =
  159     return $data->nama ?? null;
  160     }
  161 161

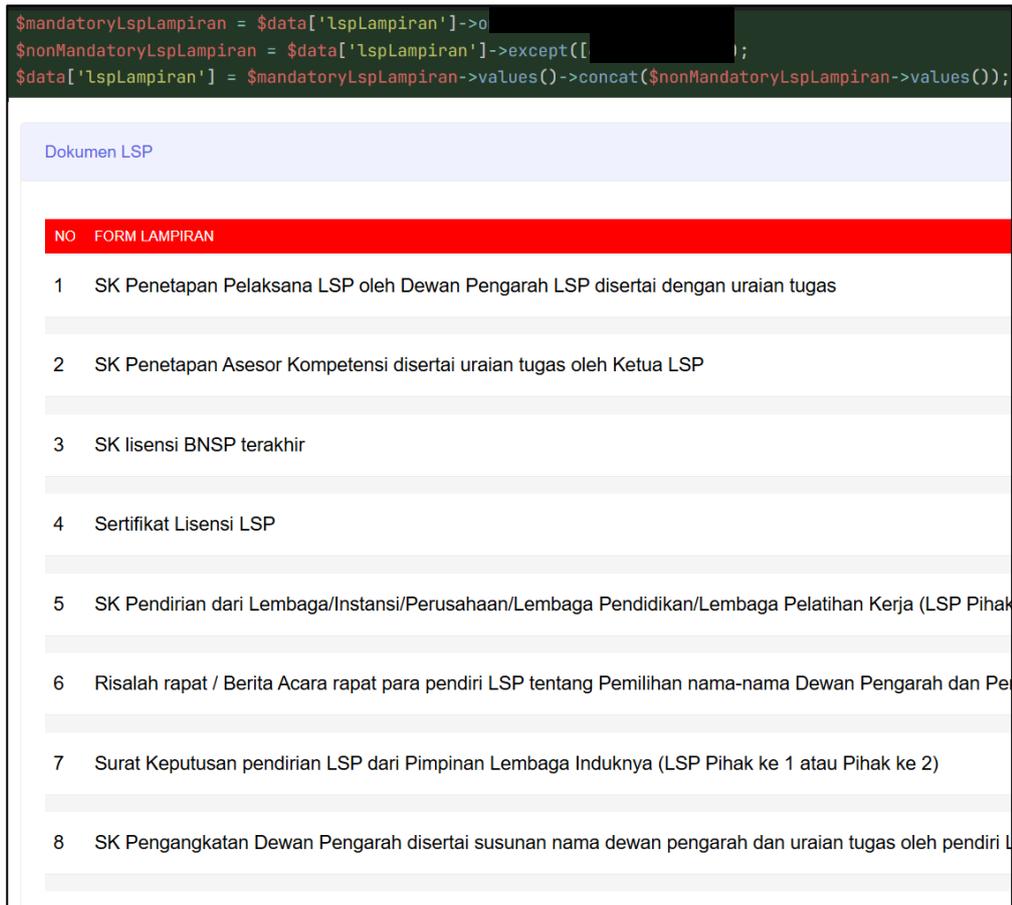
  app/Models/LSP.php
  @@ -167,7 +167,7 @@ public function sdm_lsp()
  167 167
  168 168     public function
  169 169     {
  170 -     $data =
  170 +     $data =
  171     return $data->nama ?? null;
  172     }
  173 173

```

Gambar 3.48 Merevisi SDM yang Wajib Didaftarkan Pada Profile LSP

Gambar 3.48 menampilkan revisi yang dilakukan pada halaman profil SDM terkait perubahan kewajiban penambahan struktur organisasi LSP. Perubahan tersebut mengubah terminologi dari "Ketua Pelaksana" menjadi "Pengarah Ketua/Direktur" sesuai dengan kebutuhan regulasi dan struktur organisasi yang lebih tepat. Setiap LSP wajib menambah "Ketua/Direktur" sebelum menambah dan memproses jadwal sertifikasi.





Gambar 3.49 Perubahan Posisi Lampiran Dokumen Wajib Pada Profile LSP

Gambar 3.49 menampilkan kode untuk memisahkan dokumen lampiran wajib dari seluruh daftar lampiran, lalu menggabungkannya kembali dengan dokumen non-wajib sehingga dokumen wajib tampil di urutan awal. Tujuannya agar dokumen penting lebih mudah diakses dan tidak terlewat saat proses pengisian data dari LSP.

```
> app/Http/Controllers/ProfileController.php
> app/Models/Profile/LSP.php
> app/Models/RequestLspProfileRevision.php
database/migrations/v2/2025_05_28_152647_add_column 0 → 100644
1 + <?php
2 +
3 + use Illuminate\Database\Migrations\Migration;
4 + use Illuminate\Database\Schema\Blueprint;
5 + use Illuminate\Support\Facades\Schema;
6 +
7 + return new class extends Migration
8 + {
9 +     /**
10 +      * Run the migrations.
11 +     */
12 +     public function up(): void
13 +     {
14 +         Schema::table('...', function (Blueprint $table) {
15 +             $table->string('document', 255)->nullable()->after('...');
16 +         });
17 +     }
18 +
19 +     /**
20 +      * Reverse the migrations.
21 +     */
22 +     public function down(): void
23 +     {
24 +         Schema::table('...', function (Blueprint $table) {
25 +             $table->dropColumn('document');
26 +         });
27 +     }
28 + };
```

Gambar 3.50. Penambahan Relasi dan Kolom Dokumen SK Askom pada Model dan Database

Gambar 3.50 memperlihatkan penambahan fitur pada halaman Profile LSP untuk mendukung penampilan dokumen SK Penetapan Asesor Kompetensi (SK Askom) pada proses validasi profil LSP. Perubahan mencakup penambahan kolom ke dalam tabel permintaan revisi profil LSP melalui proses migration. Selain itu, ditambahkan relasi di model untuk dapat mengakses data dokumen tersebut, serta fungsi untuk mengambil URL file agar dapat digunakan di halaman secara langsung. Ini memungkinkan sistem untuk menampilkan dokumen SK Askom terbaru, baik dari dokumen asli maupun hasil revisi.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

```

@if($isp: [1])
<div class="col-lg-12">
<div class="forms-sample mb-3">
<label class="form-label">No SK LSP terakhir</label>
<div class="form-control bg-light">{{ $isp: [1] }}
</div>
</div>
@if(session('user'))
<a target="_blank" href="{{ $isp: [1] }}" class="btn btn-sm btn-bnsp-preview">Lihat Sertifikat Lisensi LSP</a>
<a target="_blank" href="{{ $isp: [1] }}" class="btn btn-sm btn-bnsp-preview mt-3 mt-lg-0">Lihat SK lisensi BNSP LSP terakhir</a>
<a target="_blank" href="{{ $isp: [1] }}" class="btn btn-sm btn-bnsp-preview mt-3 mt-lg-0">Lihat SK Penetapan Asesor Kompetensi LSP terakhir</a>
</div>
@endif
@endif

```

Gambar 3.51 Penambahan Tombol Lihat Dokumen SK Askom di Halaman Validasi Profil LSP

Gambar 3.51 menunjukkan penyesuaian pada sisi *front-end*, tepatnya di bagian validasi data revisi profil LSP. Ditambahkan sebuah tombol baru “Lihat SK Penetapan Asesor Kompetensi LSP terakhir” yang akan membuka dokumen SK Askom pada tab baru. Tombol ini hanya muncul bagi pengguna yang memiliki level di luar LSP, seperti validator atau admin BNSP. Penambahan tombol ini bertujuan untuk memudahkan proses verifikasi dokumen penting yang dibutuhkan saat melakukan validasi profil LSP, tanpa harus mencarinya secara manual dalam daftar lampiran dokumen.

### 3.2.7 Pengembangan Paket Kompetensi

Pengembangan ini berfokus pada penyempurnaan fitur terkait paket kompetensi, termasuk penambahan versi, pencatatan revisi saat validasi, serta penyesuaian minor lainnya.

#### 3.2.7.1 Penambahan Input Versi Paket Kompetensi

```

public function rules()
{
    return [
        $rules = [
            // ...
        ]
    ];
}

if (app('session')->has('nama')) {
    $rules['version'] = [
        'required',
        new PaketKompetensiVersionRule($this->nama)
    ];
}

return $rules;
}

public function messages()
{
    return [
        'version.required' => 'No. Versi harus diisi',
    ];
}

```

Gambar 3.52 Validasi Form Request untuk Versi Paket Kompetensi

Gambar 3.52 menunjukkan penambahan aturan validasi pada sisi server untuk memastikan bahwa kolom versi pada input paket kompetensi tidak kosong saat data dikirimkan. Validasi ini ditambahkan melalui form request guna menjaga integritas data dan mencegah input yang tidak lengkap.

```
app/Rules/PaketKompetensiVersionRule.php 0 → 100644
1 + <?php
2 +
3 + namespace App\Rules;
4 +
5 + use App\Models\Skema;
6 + use Closure;
7 + use Illuminate\Contracts\Validation\ValidationRule;
8 +
9 + class PaketKompetensiVersionRule implements ValidationRule
10 + {
11 +     /**
12 +      * Run the validation rule.
13 +      *
14 +      * @param \Closure(string): \Illuminate\Translation\PotentiallyTranslatedString $fail
15 +      */
16 +     protected $model, $name, $excludeId;
17 +     public function __construct($name, $excludeId = null)
18 +     {
19 +         $this->name = $name;
20 +         $this->excludeId = $excludeId;
21 +     }
22 +
23 +     public function validate(string $attribute, mixed $value, Closure $fail): void
24 +     {
25 +         $model = new Skema();
26 +         $query = $model
27 +             ->whereNull('deleted_by')
28 +             ->whereNull('deleted_at')
29 +             ->where(
30 +                 ->where(
31 +
32 +
33 +
34 +
35 +
36 +         if ($query->exists()) {
37 +             $fail('No versi paket kompetensi yang anda inputkan sudah digunakan! Mohon cek ulang kembali!');
38 +         }
39 +     }
40 + }
```

Gambar 3.53 Custom Rule untuk Versi Paket Kompetensi

Gambar 3.53 menunjukkan implementasi custom rule yang dibuat untuk memastikan bahwa versi paket kompetensi yang diinputkan oleh pengguna belum pernah digunakan sebelumnya untuk nama paket yang sama. Aturan ini memeriksa apakah kombinasi nama dan versi paket sudah ada dalam data aktif, dan jika ditemukan duplikat, sistem akan menolak input tersebut dengan menampilkan pesan kesalahan. Pendekatan ini digunakan untuk menjaga agar tidak terjadi konflik data akibat versi yang sama digunakan lebih dari satu kali dalam entitas yang serupa.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



kompetensi. Selain itu, ditampilkan notifikasi peringatan apabila input versi dikosongkan, sehingga pengguna tidak bisa melanjutkan sebelum mengisi nilai tersebut.

```
function limitNumVersion(input)
{
    if (input.value > 99) input.value = 99;
}

function isNumberKey(evt) {
    var charCode = (evt.which) ? evt.which : evt.keyCode
    if (charCode > 31 && (charCode < 48 || charCode > 57))
        return false;
    return true;
}

$('#version').on('blur', function() {
    let value = $(this).val();
    if (value.length === 1) {
        $(this).val('0' + value);
    }
});
```

Gambar 3.56 Pembatasan Input Versi Hanya Berupa Angka

Gambar 3.56 menunjukkan implementasi validasi tambahan di sisi tampilan untuk memastikan bahwa input versi hanya dapat diisi dengan angka. Ini dilakukan berguna untuk mencegah kesalahan input dari sisi pengguna dan memperkuat kontrol data sejak awal sebelum disimpan ke sistem.

### 3.2.7.2 Penambahan Catatan Revisi Ketika Paket Kompetensi Sedang Divalidasi

```

const catatanField = $('#catatan');
const originalValue = catatanField.val();

$('#status').on('change', function() {
    const status = $(this).val();

    if (status == 2196) {
        catatanField.prop('readonly', false);
    } else {
        catatanField.prop('readonly', true);
    }

    if (status == 2195 || status == 2199) {
        catatanField.val('');
        $('input[name="catatan_validation"]').val(true);
    } else {
        $('input[name="catatan_validation"]').val(false);
        catatanField.val(originalValue);
    }
});

```

Gambar 3.57 Penambahan Kolom Catatan Revisi Pada Form Validasi Paket Kompetensi

Sebelumnya, input untuk mengisi catatan revisi sebenarnya sudah tersedia, namun disembunyikan oleh pengembang. Oleh karena itu, untuk menambahkan fitur ini, mahasiswa terlebih dahulu memunculkan kembali kolom tersebut. Gambar 3.57 menunjukkan penyesuaian lebih lanjut, yaitu kolom catatan hanya dapat diisi ketika status validasi berada pada kondisi revisi tertentu dan oleh pengguna dengan level akses khusus seperti Koordinator Sertifikasi atau Super Admin. Selain itu, ditambahkan skrip JavaScript yang mengatur agar kolom tersebut otomatis aktif saat status diubah ke revisi, dan kembali nonaktif ketika status berubah. Tujuannya adalah agar catatan hanya ditulis saat diperlukan, serta mencegah catatan lama tertimpa atau tersimpan secara tidak sengaja.

```

if($data['catatan_validation'] === 'true') {
    $store->catatan = null;
}

$this->storeSkemaMasterJenisLsp($data, $store);
$store->save();
if (isset($store->id)) {

```

Gambar 3.58 Logika Backend Untuk Menghapus Catatan Revisi Jika Status Tidak Relevan

Gambar 3.58 memperlihatkan potongan kode pada bagian backend yang bertugas menghapus isi catatan revisi apabila status validasi tidak lagi menunjukkan kondisi revisi. Hal ini dilakukan agar catatan lama tidak tersimpan saat validasi berpindah ke status yang sudah final. Validasi ini ditandai dengan flag “catatan\_validation” yang dikirim dari sisi client. Dengan pendekatan ini, sistem menjaga agar data catatan hanya tersimpan ketika sesuai dengan alur proses validasi yang berlaku.

### 3.2.7.3 Penambahan dan Penyesuaian Minor

```

success: function(response){
    if (response.status == 'success') {
        body_attr_skk_unit = '';
        if (response.data.length === 0) {
            body_attr_skk_unit = `
                <tr>
                    <td colspan="3" align="center">Tidak ada Unit SKK</td>
                </tr>
            `;
        } else {

```

Gambar 3.59 Penambahan Label Informasi Saat Unit SKK Kosong

Gambar 3.59 menunjukkan perbaikan tampilan pada halaman input paket kompetensi, khususnya bagian unit SKK. Sebelumnya, jika data unit SKK tidak tersedia, halaman akan tampak kosong tanpa penjelasan. Perubahan ini menambahkan label atau pesan informatif "Tidak ada Unit SKK" yang ditampilkan secara terpusat pada tabel ketika data kosong. Penambahan ini bertujuan agar pengguna tidak bingung saat data tidak

muncul, dan mengetahui bahwa memang tidak ada data yang tersedia, bukan karena kesalahan sistem.

```
// IS EXISTING FILTER NAMA
if (
    $data = $data->where(function($q) {
        $q->where('nama', 'LIKE',
            ->orWhere('english_name', 'LIKE',
        });
    });
}
```

Gambar 3.60 Penyesuaian Fitur Pencarian dengan Dukungan Nama Bahasa Inggris

Gambar 3.60 memperlihatkan peningkatan pada fitur pencarian data paket kompetensi agar lebih fleksibel dan mendukung penggunaan dua bahasa. Sebelumnya, pencarian hanya terbatas pada nama dalam Bahasa Indonesia. Setelah ditingkatkan, sistem kini dapat mencari berdasarkan nama dalam Bahasa Indonesia maupun Bahasa Inggris dengan menggunakan kondisi logika “OR”. Perubahan ini memudahkan pengguna yang terbiasa dengan istilah internasional dan meningkatkan aksesibilitas data paket kompetensi lintas bahasa.

### 3.2.8 Pengembangan Jadwal Sertifikasi

```
resources/views/components/jadwal-sertifikasi/edit.blade.php
721 + $(function() {
722 +     generateDataTable({
723 +         selector: $('#table_asesi'),
724 +         url: route('...', ...),
725 +         data: function(d) {
726 +             d.{{ ... }};
727 +         },
728 +         columns: [
729 +             {
730 +                 data: null,
731 +                 searchable: false,
732 +                 render: function (data, type, row, meta) {
733 +                     return meta.row + meta.settings._iDisplayStart + 1;
734 +                 },
735 +                 width: ...,
736 +                 class: ...,
737 +             },
738 +             {
739 +                 data: ...,
740 +             },
741 +             {
742 +                 data: ...,
743 +             },
744 +             {
745 +                 data: ...,
746 +             },

```

Gambar 3.61 Menampilkan Table Asesi Menggunakan Datatable Pada Halaman Jadwal Sertifikasi

Pada Gambar 3.61, ditampilkan potongan kode JavaScript yang digunakan untuk menginisialisasi tabel menggunakan DataTables dalam halaman ubah atau detail Jadwal Sertifikasi dari sisi admin LSP. Fungsi generateDataTable() dipanggil dengan parameter selector, url, dan data yang disesuaikan berdasarkan id dari jadwal tersebut. Data tersebut kemudian diambil dari controller yang merujuk kepada service datatable itu sendiri untuk ditampilkan ke dalam baris-baris tabel. Penggunaan DataTables memudahkan dalam mengelola banyak data karena mendukung fitur pencarian, pagination, dan pemrosesan di sisi server, yang membuat tampilan tetap ringan dan responsif. Salah satu kolom juga dirancang untuk secara otomatis menampilkan nomor urut baris.

```

public function datatableAsesi($request)
{
    $query = $this->repository->
    $jadwal = $this->repository->
    $statuses = $this->

    return DataTables::of($query)
        ->addColumn(
            $photoUrl)
        ->addColumn(
            return '
                <center>';
        })
        ->addColumn(
            return e
        })
        ->addColumn(
            return e
        })
        ->addColumn(
            return e
        })
        ->addColumn(
            return e
        })
        ->addColumn(
            $asessor
            return '
        })
        ->addColumn(
            return v

```

Gambar 3.62 Fungsi Datatable Asesi

Gambar 3.62 menunjukkan implementasi fungsi backend yang bertugas untuk menangani permintaan data dari DataTables di sisi frontend. Fungsi ini memanggil metode pada repository untuk mengambil data Asesi yang relevan dengan jadwal\_id tertentu, kemudian mengembalikannya dalam bentuk yang dapat diproses oleh plugin DataTables. Semua data yang ditampilkan telah melalui proses validasi dan sanitasi untuk menjaga keamanan dan konsistensi data. Penggunaan fungsi pembantu seperti `getDataAndCheckIfCertificateExists()` juga memastikan bahwa hanya data yang valid dan memiliki sertifikat yang ditampilkan ke dalam tabel.

```

+ <?php
+ namespace App\Rules;
+
+ use Closure;
+ use Illuminate\Contracts\Validation\ValidationRule;
+
+ class ██████████ Rule implements ValidationRule
+ {
+     /**
+      * Run the validation rule.
+      *
+      * @param \Closure(string): \Illuminate\Translation\PotentiallyTranslatedString $fail
+      */
+     public function validate(string $attribute, mixed $value, Closure $fail): void
+     {
+         if ($value ██████████
+             $fail('Masih Ada Asesi Yang Menunggu Persetujuan, Silahkan untuk menyetujui atau menolak asesi tersebut!');
+         }
+     }
+ }

```

Gambar 3.63 Validasi Status Asesi Pada Jadwal Sertifikasi

Gambar 3.63 menunjukkan implementasi validation rule khusus untuk memvalidasi status asesi dalam proses transisi Jadwal Sertifikasi dari status draft menuju berjalan. Sebelumnya, sistem masih memperbolehkan perubahan status jadwal menjadi berjalan meskipun terdapat asesi yang belum disetujui atau ditolak. Hal ini berisiko menyebabkan ketidaksesuaian data.

Sebagai solusinya, mahasiswa menambahkan kelas validasi khusus. Kelas ini memeriksa nilai input, dan jika ditemukan adanya asesi dengan status “Menunggu Persetujuan”, maka validasi akan gagal dan menampilkan pesan kesalahan yang informatif kepada pengguna: "Masih Ada Asesi Yang Menunggu Persetujuan, Silahkan untuk menyetujui atau menolak asesi tersebut!".

```

resources/views/components/jadwal-sertifikasi/create.blade.php
@@ -81,14 +81,20 @@
81 81 <div class="col-6 mt-3">
82 82 <div class="form-group">
83 83 <label for="">Tanggal Uji dari<span class="text-danger">*</span></label>
84 84 <input type="date" name="tanggal_mulai" id="tanggal_mulai" onchange="setDateFinish()" class="form-
85 85 <div class="input-group flatpickr">
86 86 <span class="input-group-text input-group-addon bg-transparent calendar-border"><i class="fa fa-
calendar fw-normal" aria-hidden="true"></i></span>
87 87 <input type="text" name="tanggal_mulai" id="tanggal_mulai" onchange="setDateFinish()" class="date-
88 88 <small>*Input Jadwal Maksimal H-1 Pelaksanaan Uji Kompetensi</small>
89 89 </div>
90 90 </div>
91 91 <div class="col-6 mt-3">
92 92 <div class="form-group">
93 93 <label for="">Tanggal Sampai<span class="text-danger">*</span></label>
94 94 <input type="date" name="tanggal_selesai" id="tanggal_selesai" class="form-control" @if
95 95 <div class="input-group flatpickr">
96 96 <span class="input-group-text input-group-addon bg-transparent calendar-border"><i class="fa fa-
calendar fw-normal" aria-hidden="true"></i></span>
97 97 <input type="text" name="tanggal_selesai" id="tanggal_selesai" class="date-filter-border form-
98 98 </div>
99 99 </div>

```

Gambar 3.64 Penyesuaian Input Tanggal dengan Flatpickr

Pada Gambar 3.64, memperlihatkan penyesuaian elemen input tanggal yang sebelumnya menggunakan tipe date standar dari HTML, lalu diganti dengan plugin Flatpickr agar lebih intuitif, menarik dan mudah digunakan. Komponen ini memunculkan kalender secara langsung saat memilih tanggal, meningkatkan pengalaman user. Selain itu, gaya visualnya juga lebih seragam di berbagai *browser*. Selain halaman pembuatan jadwal sertifikasi, Flatpickr juga digunakan pada halaman edit jadwal sertifikasi.

```

resources/views/components/jadwal-sertifikasi/absensi-asesi.blade.php
+89 -50
85 + </div>
86 + @if($jadwal_asesi)
87 + $status = $status['Revisi Penerbitan E Sertifikat']
88 + <div class="col-xxl-2">
89 + <input
90 + class="form-check-input"
91 + type="checkbox"
92 +
93 +
94 +
95 +
96 +
97 +
98 +
99 +
100 +
101 +
102 +
103 + </div>
104 + @endif
105 + </div>

```

Gambar 3.65 Penambahan Checkbox Kehadiran dan Kompetensi Asesi Pada Jadwal Sertifikasi Berjalan

Kode pada Gambar 3.65 menunjukkan penambahan fitur untuk mencatat kehadiran dan hasil asesmen asesi secara langsung di halaman jadwal sertifikasi. Checkbox ditampilkan jika status jadwal telah mencapai tahap "Jadwal Sertifikasi Berjalan" atau "Revisi Penerbitan E Sertifikat". Checkbox ini memungkinkan admin untuk mencatat siapa saja yang tidak hadir dalam pelaksanaan uji kompetensi. Selain itu, pada bagian lain juga ditambahkan opsi untuk menentukan status Kompeten atau Belum Kompeten bagi setiap asesi, sebagai bagian dari proses evaluasi hasil asesmen. Fitur ini mempercepat proses pencatatan dan meminimalkan kesalahan data.

### 3.2.9 Pengembangan Fitur-fitur Minor

```

@@ -52,7 +52,7 @@ public function tests(): array
-
+ 'lisensi' => 'nullable',
+ 'lisensi' => 'required',

@@ -89,6 +89,7 @@ public function messages()

    'mulai_berlaku.date' => 'Format tanggal mulai berlaku tidak valid.',
    'expired.date' => 'Format tanggal kadaluarsa tidak valid.',
    'email_komunikasi_lsp.email' => 'Format email komunikasi LSP tidak valid.',
+    'lisensi.required' => 'No. Sertifikat Lisensi LSP wajib diisi.',
    ];
}

es/views/components/user/form.blade.php
@@ -164,10 +164,10 @@
<h1 class="card-title">DATA LISENSI LSP</h1>
<div class="row">
<div class="row mb-3 col-12">
<label class="col-sm-12 col-form-label">No. Sertifikat Lisensi LSP</label>
+ <label class="col-sm-12 col-form-label">No. Sertifikat Lisensi LSP<span class="text-danger">*</span></label>
<div class="col-sm-12">
<input type="text" name="no_sertifikat_lisensi" id="no-sertifikat-l
placeholder="No. Sertifikat Lisensi" @if (isset($is_edit)) v
placeholder="No. Sertifikat Lisensi" required @if (isset($is
" @endif>
</div>
</div>

```

Gambar 3.66 Kode Mewajibkan “No. Sertifikat Lisensi LSP”

Gambar 3.66 menunjukkan penyesuaian validasi data pada proses pembuatan akun LSP baru dalam sistem. Validasi dilakukan dengan menambahkan aturan agar input “No. Sertifikat Lisensi LSP” wajib diisi oleh admin, sebagai upaya untuk meningkatkan kelengkapan dan akurasi data registrasi. Validasi ini diterapkan di dua sisi, yaitu di bagian front-end (Blade) dan pada back-end (Form Request).

```

388 388
389 389
390 390
391 +
392 +         myModel._element.addEventListener('shown.bs.modal', function () {
393 +             const intro = introJs();
394 +             intro.setOptions({
395 +                 showBullets: false,
396 +                 nextLabel: 'Selanjutnya',
397 +                 prevLabel: 'Sebelumnya',
398 +                 doneLabel: 'Dipahami',
399 +                 steps: [{
400 +                     element: '#tb_lsp',
401 +                     title: 'Tabel TUK',
402 +                     intro: 'Data TUK yang tersedia di Kabupaten/Kota dan Kecamatan yang dipilih',
403 +                     position: 'bottom'
404 +                 }, {
405 +                     element: '.btn-bnsp-add',
406 +                     title: 'Pembuatan TUK Baru',
407 +                     intro: 'Jika TUK yang dipilih tidak tersedia, maka Anda dapat membuat TUK baru',
408 +                     position: 'left'
409 +                 }
410 +             ]
411 +             intro.start();
412 +         }, { once: true });
391 413     }
392 414 }
393 415 });

```

Gambar 3.67 IntroJS

Gambar 3.67 menunjukkan implementasi IntroJS pada halaman pembuatan lokasi TUK. Fitur ini digunakan untuk membimbing pengguna melalui proses pemilihan TUK yang tersedia, dan memberikan penjelasan saat opsi pembuatan TUK baru diperlukan jika data belum tersedia.

```

app/Http/Repository/MasterData/ProfileUserRepository.php
@@ -99,6 +99,7 @@ public function getAll_2($mode = "datatable", $request)
99 99     $v->no_urut = $no_urut;
100 100     $v->id_encrypted = Crypt::encrypt($v->id);
101 101     $v->nama_peran = Helper::getPeranByJabatanName($v->idjabatan);
102 +     $v->sisnaker_verified = '<center> . view('components.profile.profile-verified', ['item' => $v])-
>render() . '</center>';
102 103     }
103 104     array_push($result, $v);
104 105 }

```

```

resources/views/components/profile/profile-verified.blade.php

```

Gambar 3.68 Menandai Status Terverifikasi Dari SIAPKerja

Gambar 3.68 menunjukkan implementasi fitur verifikasi yang saya kembangkan untuk menampilkan status terverifikasi dari platform SIAPKerja. Fitur ini berfungsi sebagai penanda visual berupa ikon centang yang muncul

ketika data seseorang telah terdaftar dan terverifikasi di sistem SIAPKerja. Penerapan fitur verifikasi ini saya integrasikan pada dua area penting dalam sistem, yaitu pada daftar asesor kompetensi dan daftar SDM (Sumber Daya Manusia) yang terdapat di halaman profil LSP. Dengan adanya penanda verifikasi ini, pengguna dapat dengan mudah mengidentifikasi asesor atau SDM yang telah melewati proses validasi data melalui SIAPKerja, sehingga meningkatkan tingkat kepercayaan terhadap kualifikasi dan kredibilitas individu tersebut. Implementasi fitur ini tidak hanya memperkuat aspek transparansi dalam sistem sertifikasi, tetapi juga menjadi kewajiban yang baru terhadap setiap LSP, yaitu memastikan asesor kompetensi atau SDM mereka telah terdaftar di SIAPKerja.

```

<x-nav-link route= " title="Data TUK" />
<x-nav-link route= ="database" title="Asesor Kompetensi" />
<x-nav-link route= sertifikat" icon="award" title="Pemegang Sertifikat" />
<x-nav-link route= sertifikat" icon="database" title="Surveilan Pemegang Sertifikat" />
<x-nav-link route= "database" title="Paket Kompetensi" />
<x-nav-link route= ="database" title="Asesor Kompetensi" />
@endif

@if (in_array($usr-
@if (!session()->get( ) && !in_array($usr-
<li class="nav-item nav-category">Lisensi</li>
<x-nav-link route= "file-plus" title="Permohonan" />
@if (in_array($usr-
<li class="nav-item nav-category">Permohonan</li>
@if (
<x-nav-link route= "-plus" title="Lisensi Awal" />
@endif
@if (in_array($usr-
<x-nav-link route= icon="calendar" title="Jadwal Asesmen" />
<x-nav-link route= icon="calendar" title="Jadwal Witness" />
@endif
@endif
@if ($usr-
<x-nav-link route= e" title="Validasi Skema" param="is_validasi=1" checker="{ request()->qu
<x-nav-link route= book" title="Validasi Paket Kompetensi" />
<x-nav-link route= ="list" title="Validasi Profil LSP" />
@endif
@if (in_array($usr->
@if (!session()->
<li class="nav-item nav-category">Lisensi </li>
<x-nav-link route= "file-plus" title="Permohonan" />

```

Gambar 3.69 Perubahan Hak Akses Setiap Menu Pada Sidebar

Gambar ini menunjukkan proses penyesuaian hak akses pengguna terhadap menu pada sidebar aplikasi berdasarkan level pengguna yang sedang login. Setiap level pengguna diberikan akses khusus untuk melihat dan mengakses menu yang relevan, seperti Validasi Skema, Jadwal Witness, Permohonan Lisensi Awal, dsb. Pengaturan ini dilakukan melalui logika kondisi if di dalam Blade template Laravel, guna memastikan bahwa hanya

pengguna dengan otorisasi tertentu yang dapat mengakses menu tertentu, sehingga meningkatkan keamanan dan menjaga konsistensi peran masing-masing pengguna dalam sistem.

```

resources/views/components/surveilan/surveilan_pemegang_sertifikat.blade.php
@@ -51,6 +51,7 @@
51 51 <div class="modal-dialog modal-lg">
52 52 <div class="modal-content">
53 53 <form id="PA_form_upload" method="post" enctype="multipart/form-data">
54 54 + @csrf
54 55 <div class="modal-header">
55 56 <h5 class="modal-title" id="ModalSurveilLabel">Update</h5>
56 57 <button type="button" class="btn-close" data-bs-dismiss="modal"
@@ -294,6 +295,7 @@ function saveUploadSurveilan() {
294 295     var pekerjaan_lainnya = $('#pekerjaan_lainnya').val();
295 296     var instansi = $('#instansi').val();
296 297     var status = $('input[name="status"]:checked').val();
298 298 +     var csrfToken = document.querySelector('input[name="_token"]').value;
297 299
298 300     var formData = new FormData();
299 301
@@ -305,7 +307,7 @@ function saveUploadSurveilan() {
305 307     formData.append('pekerjaan_lainnya', pekerjaan_lainnya);
306 308     formData.append('instansi', instansi);
307 309     formData.append('kompeten', status);
308 308 -     formData.append('_token', '{{ csrf_token() }}');
310 310 +     formData.append('_token', csrfToken);
309 311
310 312     $.ajax({
311 313         type: "POST",

```

Gambar 3.70 Perbaikan Input Data Surveil pada LSP

Gambar di atas menunjukkan perbaikan pada fitur input data surveilan untuk LSP. Sebelum dilakukan perbaikan, mahasiswa melakukan pengujian terhadap fitur tersebut dan menemukan bahwa data surveilan tidak berhasil disimpan ke dalam sistem. Setelah ditelusuri, penyebabnya adalah token CSRF (Cross-Site Request Forgery) yang tidak terbaca dengan benar pada proses pengiriman data melalui AJAX. Perbaikan dilakukan dengan memastikan bahwa token CSRF diambil langsung dari elemen input bernama `_token` pada form, kemudian dimasukkan secara ke dalam objek `formData` yang dikirimkan melalui AJAX. Dengan penambahan ini, sistem berhasil memverifikasi permintaan sebagai permintaan yang valid, dan data surveilan kini dapat disimpan ke dalam database dengan baik. Perubahan ini meningkatkan keamanan dan memastikan keandalan proses input data pada fitur tersebut.

```
resources/views/components/pengajuan-blanko/jadwal/form.blade.php +8 -2 Viewed
@@ -81,14 +81,20 @@
81 81 <div class="col-6 mt-3">
82 82 <div class="form-group">
83 83 <label for="">Tanggal Uji dari<span class="text-danger">*</span></label>
84 84 <input type="date" name="tanggal_dari" id="tanggal_dari" onchange="setDateFinish()" class="form-
84
85 <span class="input-group-text input-group-addon bg-transparent" style="border:1px solid #aeaeae
!important"><i class="fa fa-calendar fw-normal" aria-hidden="true"></i></span>
86 <input type="text" name="tanggal_dari" id="tanggal_dari" onchange="setDateFinish()" class="date-
86
87 </div>
85 88 <small>*Input Jadwal Maksimal H-1 Pelaksanaan Uji Kompetensi</small>
86 89 </div>
87 90 </div>
88 91 <div class="col-6 mt-3">
89 92 <div class="form-group">
90 93 <label for="">Tanggal Sampai<span class="text-danger">*</span></label>
91 <input type="date" name="tanggal_sampai" id="tanggal_sampai" class="form-control" @if (app('session')-
94
95 <span class="input-group-text input-group-addon bg-transparent" style="border:1px solid #aeaeae
!important"><i class="fa fa-calendar fw-normal" aria-hidden="true"></i></span>
96 <input type="text" name="tanggal_sampai" id="tanggal_sampai" class="date-filter-border form-
97
```

Gambar 3.71 Perubahan Input Menjadi Menggunakan Flatpickr Pada Pengajuan Blanko

Gambar 3.71 menunjukkan peningkatan tampilan input tanggal pada formulir pengajuan blanko. Elemen `<input type="date">` diganti dengan komponen Flatpickr untuk memberikan UI kalender yang lebih interaktif dan ramah pengguna. Perubahan ini bertujuan meningkatkan kenyamanan input serta konsistensi tampilan antar *browser*. Tidak hanya digunakan pada form pengajuan blanko, Flatpickr juga digunakan pada halaman jadwal asesmen.



```
}  
$("#tanggal_dari").flatpickr({  
  altInput: true,  
  altFormat: "Y-m-d",  
  minDate: "today",  
});  
$("#tanggal_sampai").flatpickr({  
  altInput: true,  
  altFormat: "Y-m-d",  
  minDate: "today",  
});
```

Gambar 3.72 Perbaikan Backdate Pada Jadwal Asesmen & Blanko Umum

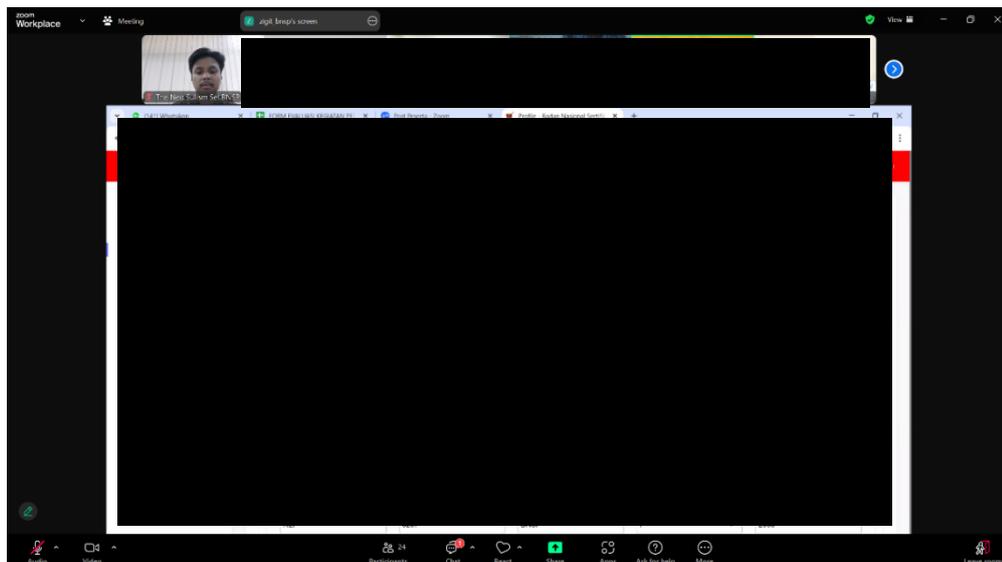
Gambar 3.72 menunjukkan perbaikan validasi tanggal pada halaman Jadwal Asesmen dan Blanko Umum, di mana sebelumnya pengguna masih dapat memilih tanggal di masa lalu (*backdate*). Setelah diperbaiki, sistem kini memastikan bahwa tanggal yang diinput minimal adalah hari ini, guna menjaga konsistensi dan keaslian jadwal yang diajukan.

### 3.2.10 Dokumentasi dan Meeting



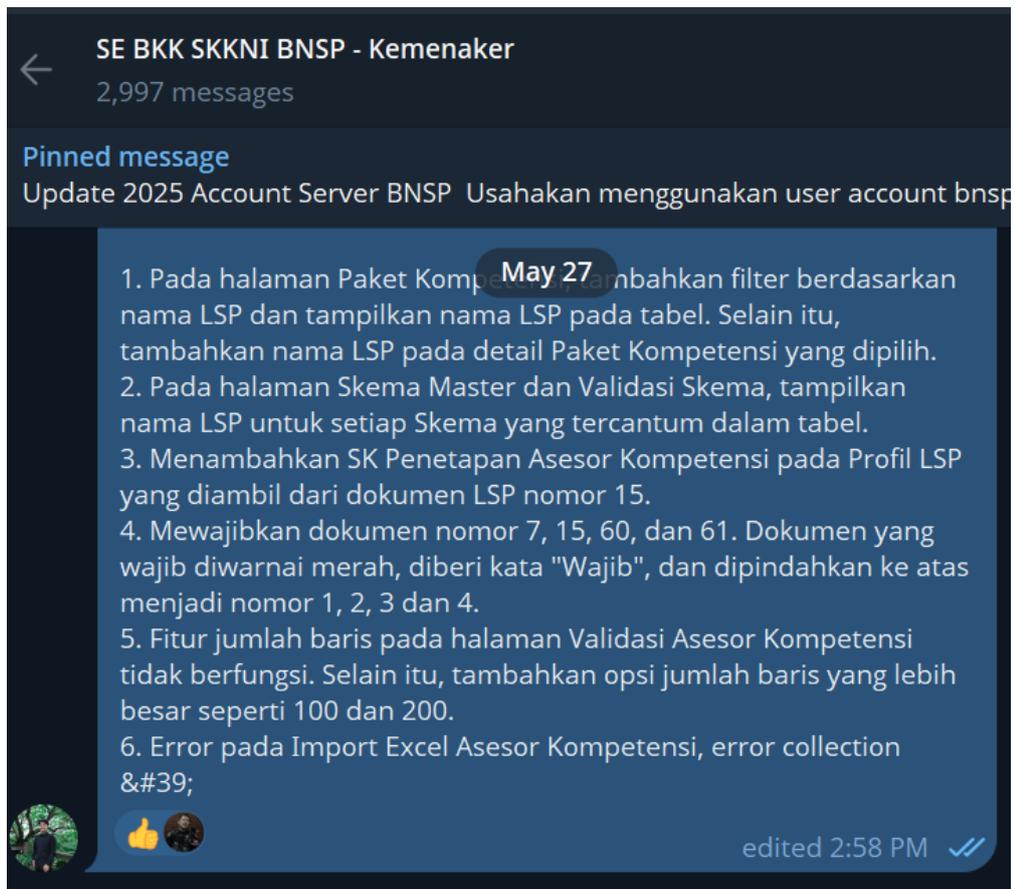
Gambar 3.73 ToT Dengan 6 LSP Secara Langsung

Gambar 3.73 menampilkan proses Training of Trainer (ToT) yang dilakukan secara langsung bersama enam Lembaga Sertifikasi Profesi (LSP). Pelatihan ini berfokus pada penggunaan sistem informasi terbaru, yaitu SISFO New, yang dirancang untuk menggantikan sistem SISFO lama. Materi pelatihan dimulai dengan proses login menggunakan username dan password baru yang telah disediakan, kemudian dilanjutkan dengan panduan mengganti password agar masing-masing LSP dapat menggunakan kata sandi sendiri. Setelah itu, peserta diajarkan cara mengisi profil LSP mereka, membuat jadwal sertifikasi baru, dan bagaimana asesi dapat mendaftar pada jadwal tersebut. Selama proses pelatihan berlangsung, beberapa kendala teknis seperti error atau bug ditemukan. Seluruh permasalahan tersebut dicatat oleh tim Braindevs untuk ditindaklanjuti setelah sesi ToT berakhir. Namun, jika terdapat kendala yang dapat segera diselesaikan, perbaikan dilakukan langsung pada saat itu juga.



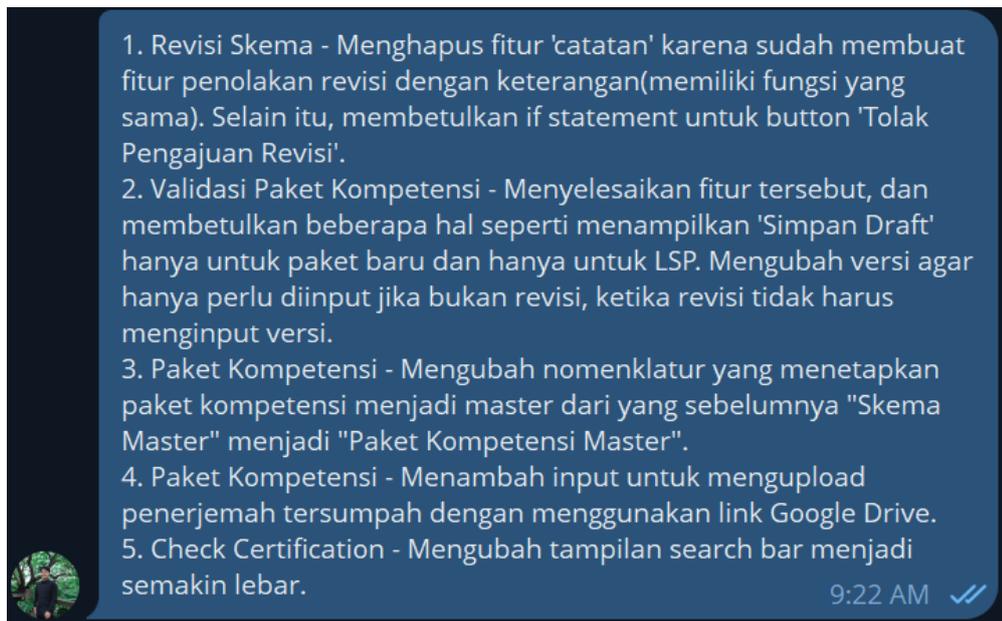
Gambar 3.74 ToT Dengan LSP Menggunakan Zoom

Gambar 3.74 menampilkan proses Training of Trainer (ToT ) yang dilakukan secara daring melalui platform Zoom dengan melibatkan lebih banyak Lembaga Sertifikasi Profesi (LSP) dibandingkan sesi sebelumnya. Materi yang disampaikan mirip dengan ToT tatap muka, yaitu penggunaan sistem SISFO New sebagai pengganti sistem lama. Peserta pelatihan diajarkan mulai dari proses login dengan akun yang baru disediakan, mengganti password, melengkapi profil LSP, hingga membuat jadwal sertifikasi yang nantinya dapat didaftarkan oleh para asesi. Perbedaannya adalah pada ToT kali ini peserta diajarkan untuk membuat Paket Kompetensi dan Skema yang baru. Selain itu, peserta diwajibkan untuk mendaftarkan anggota SDM LSP di halaman profil. Terakhir, peserta juga diberikan informasi bahwa setiap asesor kompetensi yang didaftarkan wajib memiliki akun SIAPKerja. Selama sesi berlangsung, beberapa kendala teknis seperti error dan bug juga ditemukan. Tim Braindevs mencatat setiap permasalahan yang muncul untuk ditindaklanjuti setelah pelatihan berakhir, namun tetap berupaya menyelesaikan kendala secara langsung apabila memungkinkan selama sesi Zoom berlangsung.



Gambar 3.75 Notulensi Perbaikan atau Penambahan Fitur dari Hasil Meeting yang Dikirim ke Group Tim Pengembang BNSP

Gambar 3.75 menunjukkan dokumentasi notulensi hasil rapat yang berisi rangkuman berbagai kendala, kebutuhan penambahan fitur, serta revisi yang harus dilakukan dalam sistem. Mahasiswa berperan aktif mencatat poin-poin penting selama proses diskusi berlangsung. Setelah rapat telah selesai dan notulensi selesai disusun, dokumen tersebut dikirimkan ke grup Telegram internal tim pengembang BNSP agar seluruh anggota tim mendapatkan informasi yang sama, serta dapat menindaklanjuti sesuai tanggung jawab masing-masing. Pendekatan ini membantu menjaga koordinasi dan memastikan tindak lanjut atas perbaikan dilakukan secara sistematis.



Gambar 3.76 Notulensi Perbaikan atau Penambahan Fitur dari Hasil Meeting yang Dikirim ke Mentor

Gambar 3.76 menunjukkan bahwa tidak semua notulensi langsung dikirim ke grup pengembang. Pada beberapa kesempatan, mentor meminta agar mahasiswa mencatat hasil diskusi terlebih dahulu dan mengirimkannya secara pribadi kepada beliau. Notulensi ini kemudian akan ditinjau, dirangkum, dan digabungkan dengan catatan rekan lain oleh mentor sebelum disampaikan secara resmi ke grup tim. Meskipun alur distribusinya berbeda, konteks informasi yang dicatat tetap sama, yaitu sebagai acuan revisi dan pengembangan lebih lanjut pada sistem informasi BNSP.

### 3.3 Kendala yang Ditemukan

Selama proses magang dalam project BNSP, mahasiswa menghadapi berbagai tantangan yang berkaitan dengan kemampuan teknis, pemahaman terhadap alat kolaborasi, serta kesiapan dalam menghadapi situasi profesional di dunia kerja nyata. Beberapa kendala tersebut berdampak langsung pada kecepatan adaptasi dan kelancaran pengerjaan tugas yang diberikan.

1. Mahasiswa mengalami kendala karena belum memiliki pengetahuan dan pengalaman sama sekali dalam menggunakan Laravel. Hal ini

menyebabkan mahasiswa tidak dapat langsung berkontribusi pada proyek saat awal magang dan perlu waktu untuk menyesuaikan diri.

2. Mahasiswa belum pernah menggunakan Git sebelumnya, termasuk konsep dasar seperti commit, push, pull, merge, serta praktik penggunaan GitLab dalam lingkungan kolaboratif. Hal ini menyulitkan mahasiswa dalam memahami alur kerja proyek dan menyebabkan banyak kesalahan teknis saat mencoba berkontribusi.
3. Mahasiswa mengalami kebingungan dalam memahami alur pengiriman data dari form HTML Blade menggunakan JavaScript/jQuery dan AJAX. Hal ini disebabkan karena sebelumnya tidak pernah mendapatkan pembelajaran praktik mengenai interaksi data dinamis di sisi klien.
4. Mahasiswa merasa tidak percaya diri saat berhadapan langsung dengan klien, terutama ketika ditanya mengenai kendala teknis pada sistem. Mahasiswa cenderung hanya mencatat dan menyerahkan penjelasan kepada mentor, dan merasa canggung ketika harus menjawab secara langsung.
5. Dalam satu kesempatan, mahasiswa harus menghadiri pertemuan dengan klien tanpa ditemani rekan atau mentor karena beberapa senior sedang sakit. Kondisi ini menimbulkan tekanan dan kecemasan tinggi karena mahasiswa belum terbiasa menjawab langsung permasalahan yang dihadapi klien.

### **3.4 Solusi atas Kendala yang Ditemukan**

Untuk mengatasi berbagai kendala yang dihadapi selama magang, mahasiswa berinisiatif mencari solusi secara mandiri maupun dengan bimbingan mentor. Solusi-solusi ini diterapkan guna meningkatkan pemahaman teknis, memperbaiki cara kerja, dan menyesuaikan diri dengan lingkungan kerja profesional.

1. Untuk mengatasi ketidaktahuan terhadap Laravel, mahasiswa mengambil inisiatif untuk mempelajari ulang konsep dasar PHP melalui playlist video di YouTube sebelum mempelajari Laravel itu sendiri. Setelah memahami PHP, mahasiswa mempelajari dokumentasi dan tutorial Laravel untuk memahami struktur MVC dan praktik penggunaannya dalam proyek magang.

2. Mahasiswa mempelajari Git secara bertahap dengan menonton video tutorial dan membaca dokumentasi. Proses belajar dilakukan melalui trial and error, termasuk memahami pembuatan pesan commit yang baik, serta proses merge dan cherry-pick pada GitLab. Kesalahan yang terjadi dijadikan pembelajaran untuk memperkuat pemahaman ke depannya.
3. Dalam menghadapi kebingungan pada penggunaan JavaScript dan jQuery, mahasiswa langsung mencari solusi secara daring, terutama melalui StackOverflow. Mahasiswa belajar bagaimana mengirim form menggunakan AJAX dari Blade Laravel dan bagaimana data tersebut diproses di sisi controller dan database. Metode pembelajaran ini dilakukan secara langsung saat menghadapi kasus nyata (on-the-spot learning).
4. Untuk mengatasi kesulitan dalam komunikasi, mahasiswa mencoba memahami pertanyaan klien dengan lebih baik dan mencatat poin penting saat pertemuan. Meskipun masih belum berani menjawab secara langsung, mahasiswa mulai belajar dari cara mentor menjelaskan agar bisa perlahan-lahan membangun kepercayaan diri.
5. Ketika harus menghadiri meeting sendiri, mahasiswa menjalani situasi tersebut sebaik mungkin dengan mencatat semua poin penting dan mencoba menjawab pertanyaan semampunya. Meskipun belum menemukan solusi langsung terhadap rasa gugup, mahasiswa menjadikan pengalaman ini sebagai pembelajaran mental untuk situasi serupa di masa mendatang.