

DOKUMENTASI PROGRAM
SISTEM PENGADUAN DAN PELAPORAN PELANGGARAN
DI LINGKUNGAN BADAN PUSAT STATISTIK



POLITEKNIK STATISTIKA STIS
BADAN PUSAT STATISTIK

2024

Daftar Isi

Daftar Isi.....	1
Latar Belakang.....	2
Penjelasan Program.....	5
Proses Bisnis Usulan.....	6
Analisis Kebutuhan Fungsional	7
Arsitektur Sistem Usulan	7
Rancangan Basis Data.....	8
Bahasa Pemrograman dan Tools yang digunakan	10
Lampiran	11

Latar Belakang

Badan Pusat Statistik (BPS) adalah Lembaga Pemerintah Non Departemen di bawah presiden yang berwenang untuk melaksanakan tugas pemerintahan di bidang kegiatan statistik sesuai dengan peraturan perundang-undangan yang berlaku. Untuk melaksanakan tugasnya tersebut, BPS melaksanakan kegiatan survei dan sensus dalam rangka mengumpulkan data yang diperlukan untuk menghitung suatu indikator statistik pemerintahan.

Selain mengumpulkan data melalui sensus dan survei, BPS juga berperan dalam pelayanan publik di bidang statistik. Pelayanan publik di BPS adalah berupa pelayanan statistik terpadu. Pelayanan Statistik Terpadu adalah pemberian pelayanan data dan kegiatan statistik dari beberapa jenis pelayanan yang dilakukan secara terpadu melalui satu pintu oleh satu unit kerja sebagai penanggung jawab. Pelayanan statistik terpadu terdiri dari beberapa layanan, di antaranya layanan perpustakaan, penjualan, konsultasi, dan rekomendasi.

Sebagai institusi yang melaksanakan kegiatan pelayanan publik BPS harus melaksanakan pelayanan dengan sebaik-baiknya. Selain itu, BPS juga harus terus melakukan inovasi dalam meningkatkan kualitas pelayanan publik yang diselenggarakan. Salah satu upaya peningkatan kualitas pelayanan publik, dapat dicapai melalui penyelenggaraan penanganan pengaduan masyarakat.

Berdasarkan UU Nomor 25 Tahun 2009 tentang Pelayanan Publik pasal 8 disebutkan bahwa pengelolaan pengaduan masyarakat merupakan salah satu hal yang harus ada dalam penyelenggaraan pelayanan publik. Hal ini juga disebutkan dalam Perka BPS nomor 78 tahun 2020 pasal 2 yang menyebutkan bahwa pengaduan masyarakat adalah salah satu ruang lingkup dalam pelayanan statistik terpadu di lingkungan BPS RI. Oleh karena itu, untuk melaksanakan amanat peraturan perundang-undangan, BPS RI harus berupaya untuk menyediakan suatu sistem pengaduan masyarakat.

Dengan menimbang permasalahan yang ada, diputuskan bahwa diperlukan pembangunan suatu Sistem Pengaduan dan Pelaporan Pelanggaran dengan harapan dapat menyelesaikan masalah yang dialami. Sistem pengaduan dan pelaporan pelanggaran (whistleblowing system) adalah sistem yang berfungsi untuk menerima, mengelola, menelaah, dan menindaklanjuti pengaduan masyarakat secara daring dengan struktur dan dokumentasi yang baik dan sistematis. Sistem pengaduan dan pelaporan pelanggaran juga harus dapat menyediakan informasi atau laporan yang akurat dan tepat waktu bagi pengambil keputusan untuk ditindaklanjuti.

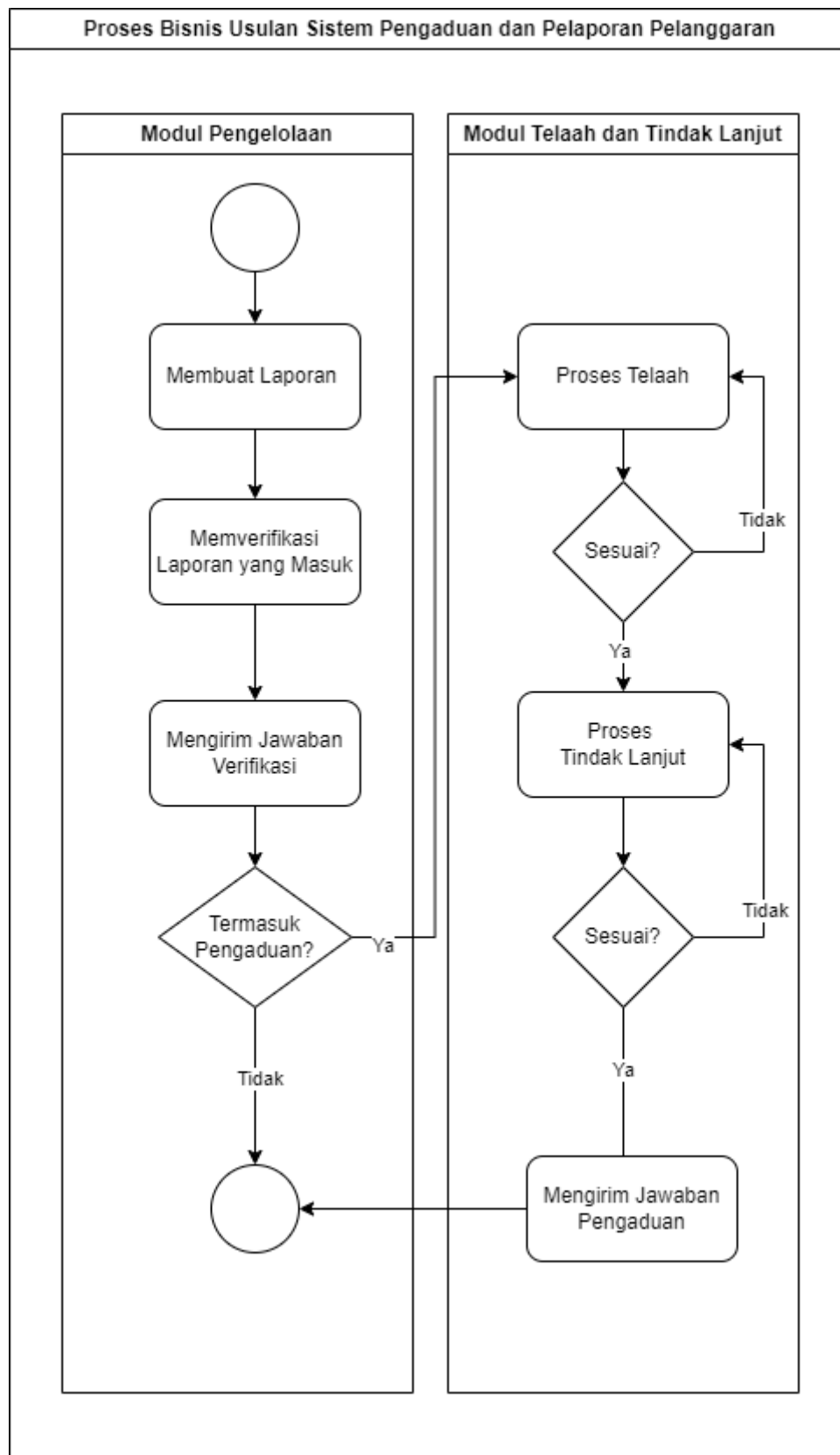
Hal atau aspek yang perlu diperhatikan dalam pembuatan sistem ini adalah komitmen, kerahasiaan, fokus, sosialisasi, dan pengembangan. Hal yang juga menjadi pertimbangan dalam pembuatan sistem ini adalah kerahasiaan, mudah dan cepat, integrasi dan pelaporan, dan monitoring. Sistem pengaduan dan laporan pelanggaran dibuat dan dikembangkan dalam rangka mendorong peran masyarakat, pejabat, serta pegawai di lingkungan BPS dalam upaya pencegahan dan pemberantasan tindak pidana seperti korupsi dan penyalahgunaan wewenang oleh pejabat maupun pegawai. Sistem pengaduan dan laporan pelanggaran juga harus dibuat agar pengelolaan serta tindak lanjut pengaduannya dapat dilakukan secara transparan, akuntabel, dan dapat dipertanggungjawabkan.

Sistem pengaduan dan pelaporan pelanggaran ini berupa aplikasi website yang dibangun menggunakan bahasa pemrograman PHP. Untuk memudahkan pembangunan, digunakan framework Laravel versi 10.0 sebagai framework PHP. Selain itu, digunakan framework CSS yakni tailwind css untuk mengatur tampilan aplikasi, serta framework alpine.js dan choices.js untuk membuat aplikasi menjadi dinamis. Dari sisi basis data, digunakan RDBMS MySql untuk mengatur dan mengelola basis data pada sistem ini.

Penjelasan Program

<i>Business Needs:</i>	<p>Project ini dibangun untuk :</p> <ol style="list-style-type: none"> 1. Menerima pengaduan dan pelaporan pelanggaran dari masyarakat; 2. Menyediakan fitur verifikasi atas laporan yang masuk; 3. Melakukan disposisi pengaduan kepada tim telaah dan operator tindak lanjut; 4. Menyediakan fitur telaah dan tindak lanjut pengaduan; 5. Memeriksa hasil telaah dan tindak lanjut pengaduan; 6. Menyediakan <i>monitoring</i> proses pengelolaan pengaduan dan secara aktual untuk keperluan pimpinan dalam melakukan pengawasan internal.
<i>Business Requirements</i>	
<p>Sistem yang mendukung pengelolaan pengaduan dan pelaporan pelanggaran ini harus memiliki fitur-fitur sebagai berikut :</p> <ol style="list-style-type: none"> 1. Fitur <i>login</i> sesuai dengan <i>role</i> 2. Fitur kelola admin 3. Fitur kelola unit kerja 4. Fitur kelola kategori layanan 5. Fitur kelola tim verifikator 6. Fitur kelola tim telaah 7. Fitur kelola operator 8. Fitur kelola tim pemantau 9. Fitur buat laporan 10. Fitur identifikasi pengaduan 11. Fitur kelola kamus jawaban verifikasi 12. Fitur kirim jawaban verifikasi 13. Fitur disposisi 14. Fitur telaah pengaduan 15. Fitur kirim jawaban tindak lanjut 16. Fitur identifikasi jawaban tindak lanjut 17. Fitur kirim jawaban pengaduan 	
<i>Business Value</i>	
<p>Dengan adanya sistem ini ini diharapkan proses pengelolaan pengaduan dan pelaporan pelanggaran lebih terstruktur dan efisien, dan penugasan kepada pegawai lebih transparan dan merata.</p>	

Proses Bisnis Usulan



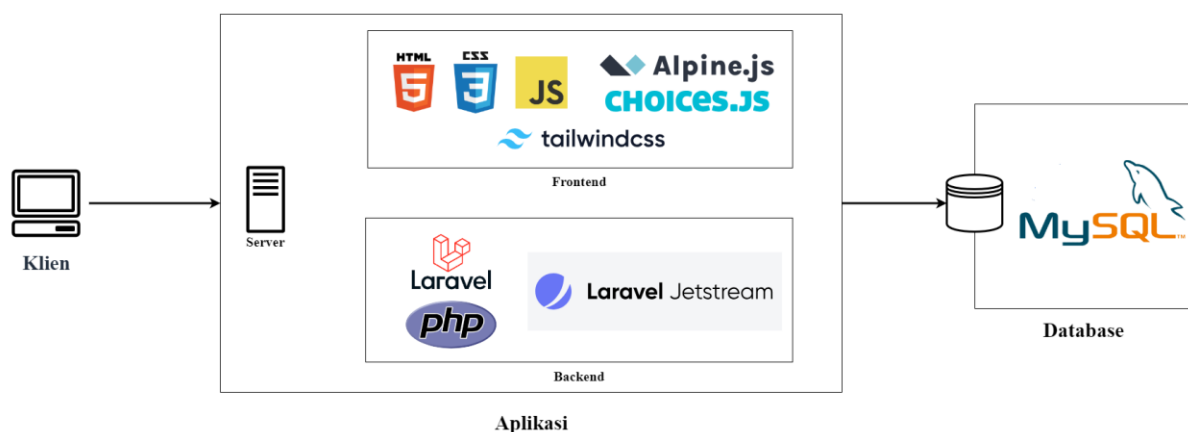
Gambar 1. Proses Bisnis Sistem

Pada proses bisnis usulan melibatkan dua modul, yaitu modul pengelolaan dan modul telaah dan tindak lanjut. Modul pengelolaan digunakan untuk proses penerimaan laporan, verifikasi laporan, dan pengiriman jawaban verifikasi. Modul telaah dan tindak lanjut digunakan untuk proses telaah pengaduan, tindak lanjut pengaduan, dan pengiriman jawaban pengaduan. Proses bisnis sistem usulan dapat dilihat pada Gambar 1.

Analisis Kebutuhan Fungsional

Pada sistem ini mencakup fitur masuk sesuai peran, fitur untuk mengelola pengguna, fitur buat laporan, fitur identifikasi pengaduan, fitur kelola kamus jawaban verifikasi, fitur disposisi, fitur isi kertas kerja telaah, fitur kirim jawaban tindak lanjut, fitur identifikasi jawaban tindak lanjut, dan fitur kirim jawaban pengaduan.

Arsitektur Sistem Usulan

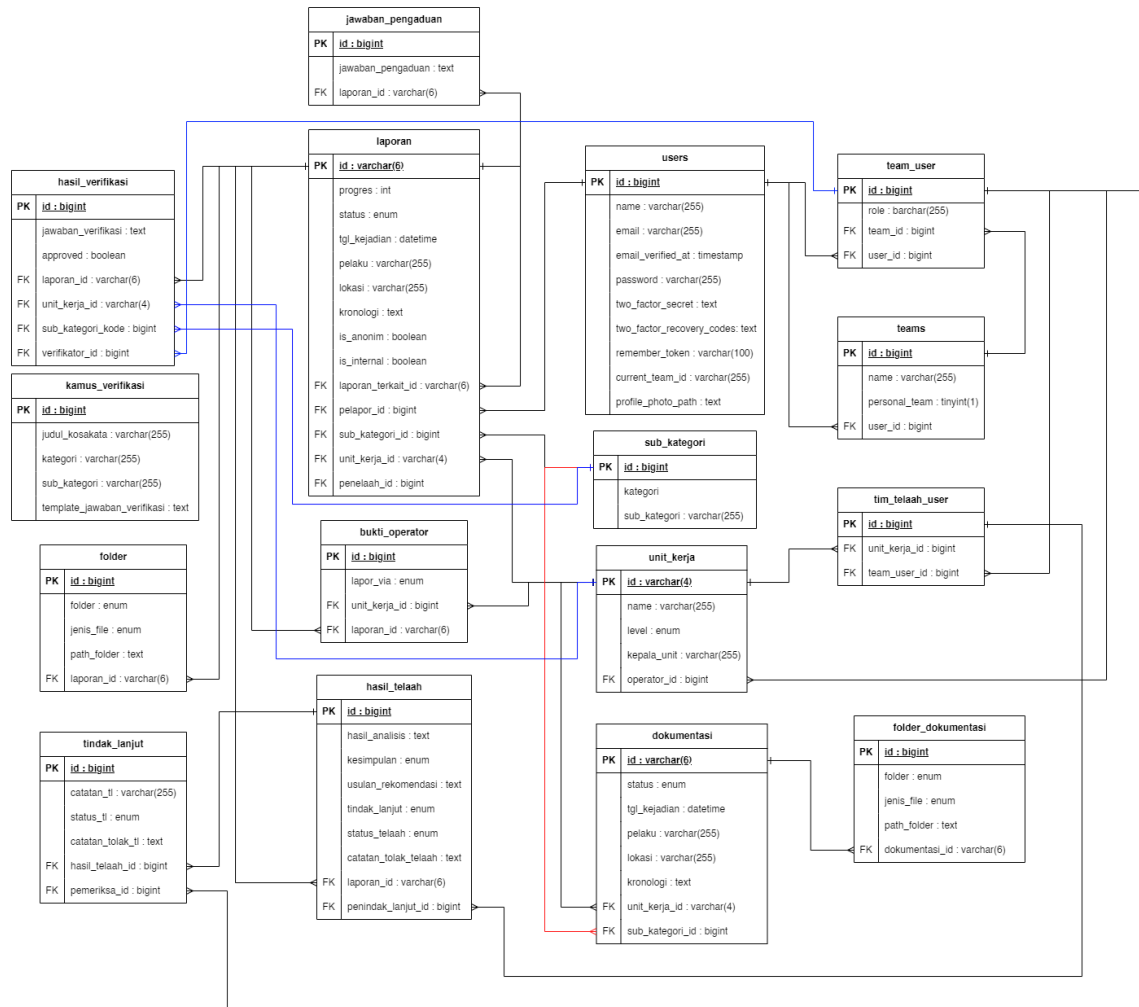


Gambar 2. Arsitektur Sistem Usulan

Sistem yang dibangun merupakan sistem informasi berbasis web sesuai dengan kebutuhan non fungsional dari sistem. Sistem dibangun berbasis web dengan tujuan agar sistem dapat akses dan digunakan dengan mudah oleh pengguna tanpa perlu memasang pada perangkatnya, melainkan cukup mengakses melalui *web browser* yang tersambung dengan internet. Sistem dibangun dengan secara keseluruhan dengan bagian *front-end* yang dibangun menggunakan html, css, javascript dan tailwind css yang dikelola oleh *view* Laravel, sedangkan untuk bagian *back-end* menggunakan *framework* Laravel dan DBMS *MySQL*. Rancangan arsitektur pada sistem usulan dapat dilihat pada Gambar 2.

Rancangan Basis Data

Pada sistem ini pemodelan dan perancangan basis data diterjemahkan dalam bentuk *entity relationship diagram*. ERD pada sistem yang dibangun dapat dilihat pada Gambar 3. Berikut merupakan tautan untuk tampilan ERD yang lebih jelas dan lengkap. [s.stis.ac.id/ERD Pengaduan](https://s.stis.ac.id/ERD_Pengaduan)



Gambar 3. Entity Relationship Diagram

Tabel 1. Daftar entitas dan deskripsi entitas sistem usulan

No.	Nama Entitas	Deskripsi
(1)	(2)	(3)
1.	users	Entitas yang digunakan untuk menyimpan informasi dasar terkait pengguna seperti nama, email, dan sebagainya.
2.	teams	Entitas yang digunakan untuk menyimpan informasi tim yang ada pada sistem yang dibangun seperti nama tim.
3.	team_user	Entitas yang digunakan untuk menyimpan informasi keanggotaan pengguna ke suatu tim termasuk role.
4.	tim_telaah_user	Entitas yang digunakan untuk menyimpan informasi keanggotaan pengguna ke suatu tim telaah.
5.	unit_kerja	Entitas yang digunakan untuk menyimpan informasi unit kerja di lingkungan Badan Pusat Statistik.
6.	laporan	Entitas yang digunakan untuk menyimpan informasi mengenai laporan seperti tanggal kejadian, pelaku, kronologi, dan sebagainya.
7.	sub_kategori	Entitas yang digunakan untuk menyimpan informasi sub kategori pada masing-masing kategori laporan.

8.	hasil_verifikasi	Entitas yang digunakan untuk menyimpan informasi hasil verifikasi dari sebuah laporan seperti jawaban verifikasi, verifikator, dan sebagainya.
9.	kamus_verifikasi	Entitas yang digunakan untuk menyimpan informasi kamus jawaban verifikasi seperti judul kosakata, template jawaban verifikasi, dan sebagainya.
10.	hasil_telaah	Entitas yang digunakan untuk menyimpan informasi hasil telaah seperti hasil analisis, usulan rekomendasi, penelaah, dan sebagainya.
11.	tindak_lanjut	Entitas yang digunakan untuk menyimpan informasi hasil tindak lanjut pengaduan seperti penindaklanjutan, catatan tindak lanjut, dan sebagainya.
12.	folder	Entitas yang digunakan untuk menyimpan informasi lokasi penyimpanan dokumen-dokumen yang terkait dengan suatu pengaduan.
13.	bukti_operator	Entitas yang digunakan untuk menyimpan informasi mengenai bukti operator seperti operator unit, lapor via, dan sebagainya.
14.	dokumentasi	Entitas yang digunakan untuk menyimpan informasi mengenai dokumentasi penanganan pelanggaran disiplin.
15.	folder_dokumentasi	Entitas yang digunakan untuk menyimpan informasi lokasi penyimpanan dokumen-dokumen yang terkait dengan suatu dokumentasi penanganan pelanggaran disiplin.
16.	jawaban_pengaduan	Entitas yang digunakan untuk menyimpan informasi mengenai jawaban pengaduan yang dikirimkan kepada pelapor.

Bahasa Pemrograman dan *Tools* yang digunakan

Bahasa Pemrograman

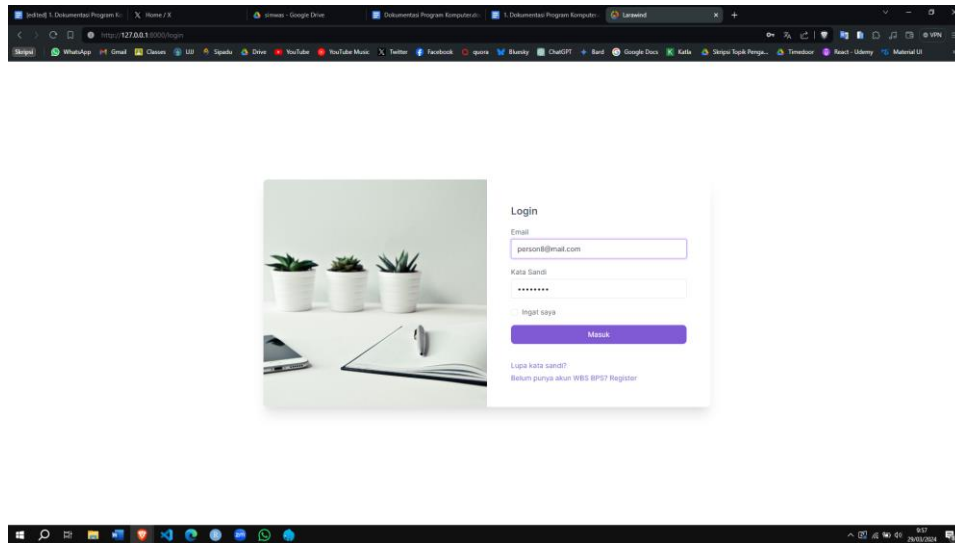
1. PHP 8.1
2. *Framework* Laravel 10
3. *Framework* Tailwind CSS
4. DBMS Mysql

Tools

1. *System Design*: Draw.io
2. *System Development*: Visual Studio Code

Lampiran

Lampiran 1. Contoh Fitur Login



Gambar 4. Fitur Login dengan Akun WBS

Untuk masuk ke dalam aplikasi, pengguna harus login terlebih dahulu. Tersedia tiga opsi login yakni login dengan SSO BPS, Akun Google, dan Akun WBS. Pada Gambar 4. ditampilkan login dengan akun WBS. Pengguna dapat melakukan registrasi terlebih dahulu

melalui menu yang telah disediakan. Apabila sudah memiliki akun WBS, pengguna dapat memasukkan email dan password untuk masuk ke dalam aplikasi.

Lampiran 2. Contoh Fitur Menambah Laporan

The screenshot displays the 'Buat Laporan' (Create Report) form within the WBS application. The form is titled 'Buat Laporan' and includes a 'Simpan Draft' (Save Draft) button and a 'Submit' button. The form contains several sections: 'Apakah anda ingin melaporkan sebagai anonim?' (Do you want to report anonymously?), 'Apakah Anda melaporkan sebagai Internal atau eksternal BPS?' (Are you reporting as Internal or External BPS?), 'Laporan Terkait (Optional)' (Related Report), '1. Dugaan pelanggaran apa yang akan anda laporkan?' (What violation do you suspect?), '2. Siapa pihak yang terlibat dalam dugaan pelanggaran?' (Who are the parties involved in the suspected violation?), '3. Kapan terjadinya pelanggaran tersebut?' (When did the violation occur?), '4. Dimana lokasi kejadian pelanggaran tersebut?' (Where did the violation occur?), '5. Satuan/Unit kerja yang terkait dengan pelanggaran ini' (Which unit/department is related to this violation?), '6. Upload bukti pelanggaran tersebut (Optional)' (Upload evidence of the violation (Optional)), and '7. Kronologi kejadian' (Chronology of the incident). The form also includes a 'Pilih tanggal kejadian' (Select incident date) field and a 'Pilih lokasi kejadian' (Select incident location) field. The bottom section is a rich text editor for the 'Kronologi kejadian' (Chronology of the incident).

Gambar 5. Fitur Menambah Laporan

Untuk menambah laporan, pengguna dapat masuk ke menu buat laporan. Selanjutnya pengguna dapat mengisi formulir yang ditampilkan pada Gambar 5. Pengguna harus melengkapi beberapa informasi seperti tanggal kejadian, pelaku, kronologi, dan informasi lainnya sebelum menekan tombol tambah untuk mengirim data ke sistem. Tersedia juga fitur draft untuk menyimpan isian sementara, sebelum dikirimkan ke sistem secara utuh.

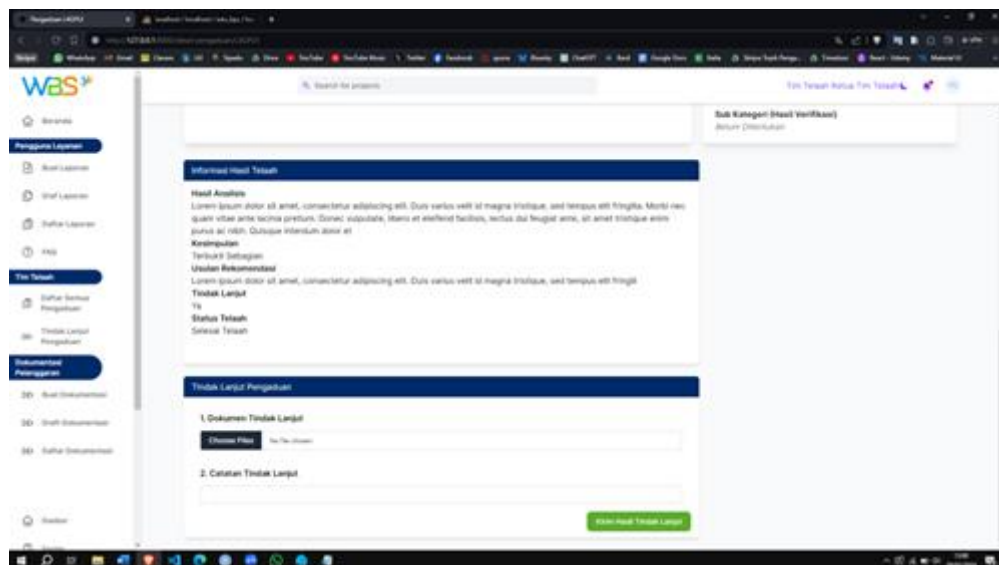
Lampiran 3. Contoh Fitur Telaah Pengaduan

The screenshot displays the 'Kartus Kerja Telaah' (Case Review Card) form within the WBS application. The form is titled 'Kartus Kerja Telaah' and includes a 'Simpan Draft' (Save Draft) button and a 'Submit' button. The form contains several sections: '1. Tambahkan Bukti Dukung (Optional)' (Add supporting evidence (Optional)), '2. Hasil Analisis' (Analysis Results), '3. Lampiran Hasil Analisis (Optional)' (Attachment of Analysis Results (Optional)), '4. Kesimpulan' (Conclusion), '5. Usulan Rekomendasi' (Recommendation), '6. Tindak Lanjut' (Follow-up), '7. Draft Surat Jawaban ke Pelapor' (Draft response letter to the reporter), and '8. Draft Surat Jawaban ke Manajemen' (Draft response letter to management). The form also includes a 'Pilih tanggal kejadian' (Select incident date) field and a 'Pilih lokasi kejadian' (Select incident location) field. The bottom section is a rich text editor for the 'Kronologi kejadian' (Chronology of the incident).

Gambar 6. Fitur Telaah Pengaduan

Proses telaah pengaduan diakomodasi dengan formulir kertas kerja telaah yang ditampilkan pada Gambar 6. Formulir ini terdiri atas beberapa isian yang harus diisi setiap selesai melakukan proses telaah. Isian tersebut diantaranya adalah hasil analisis beserta dokumen, tambahan bukti dukung, usulan rekomendasi, kesimpulan telaah, keputusan tindak lanjut, dan draft jawaban untuk pelapor dan manajemen.

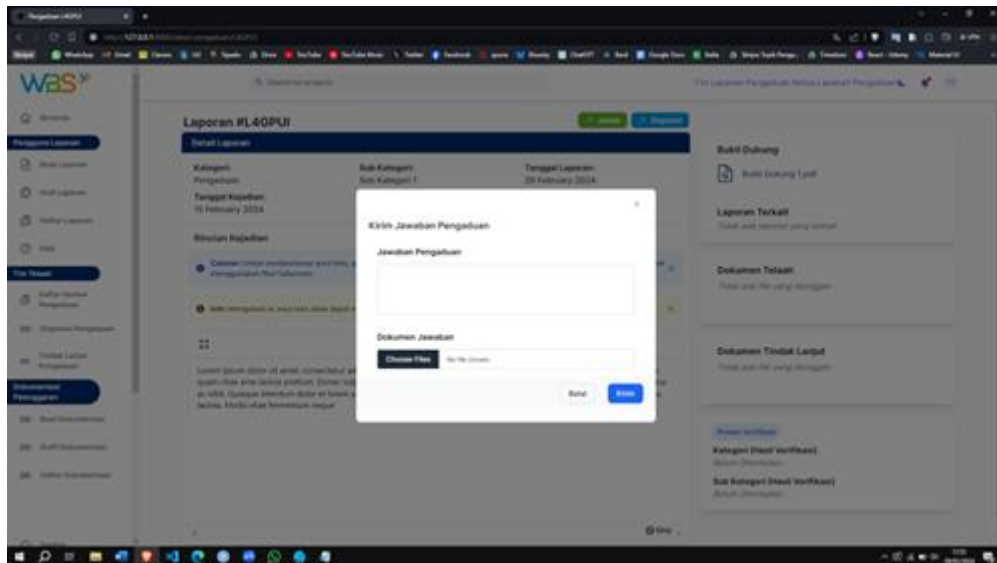
Lampiran 4. Contoh Fitur Tindak Lanjut Pengaduan



Gambar 7. Fitur Tindak Lanjut Pengaduan

Proses ini dilakukan oleh operator tindak lanjut di setiap unit kerja. Formulir tindak lanjut terdiri dari dua isian yakni catatan tindak lanjut serta dokumen tindak lanjut. Untuk hasil tindak lanjut, juga tersedia fitur pemeriksaan berjenjang yakni pemeriksaan oleh tim pemantau, dilanjutkan dengan pemeriksaan oleh tim layanan pengaduan. Tampilan proses tindak lanjut dapat dilihat pada Gambar 7.

Lampiran 5. Contoh Fitur Kirim Jawaban Pengaduan



Gambar 8. Fitur Kirim Jawaban Pengaduan

Langkah terakhir dalam proses telaah dan tindak lanjut pengaduan adalah proses kirim jawaban pengaduan seperti yang ditampilkan pada Gambar 8. Proses ini dapat dilakukan oleh user dengan role ketua layanan pengaduan. Jawaban yang dikirimkan berupa teks dan dokumen jawaban. Setelah jawaban berhasil dikirim, pengaduan terkait akan ditandai selesai.

Lampiran 6. Kode Untuk Menambah Pengaduan

```
public function tambahLaporan(Request $request)
{
    // dd($request->all());

    if ($request->status == 'Draf Laporan') {
        $request->validate([
            'kategori' => 'required',
            'pelaku' => 'required',
        ]);

        if (!isset($request->id)) {
```

```
// dd($request->id);

do {

    $randomString = Str::upper(Str::random(6));

} while (LaporanModel::where('id', $randomString)-
>exists());

if (isset($request->tgl_kejadian)) {

    $date_kejadian = date('Y-m-d H:i:s',
strtotime($request->tgl_kejadian));

} else {

    $date_kejadian = null;

}

$laporan = LaporanModel::create([

    'id' => $randomString,

    'pelapor_id' => Auth::id(),

    'laporan_terkait_id' => $request-
>laporan_terkait_id,

    'unit_kerja_kode' => $request->unit_kerja,

    'sub_kategori_id' => $request->kategori,

    'progres' => '0',

    'status' => $request->status,

    'tgl_kejadian' => $date_kejadian,

    'pelaku' => $request->pelaku,

    'lokasi' => $request->lokasi,

    'kronologi' => $request->kronologi,

    'is_anonim' => $request->is_anonim,
```



```
        'is_internal' => $request->is_internal,

        'created_at' => date("Y-m-d H:i:s"),

        'updated_at' => date("Y-m-d H:i:s"),

    ]->save();

    return redirect()->route('draf-laporan', ['id' =>
$randomString])->with('message', 'Draf laporan berhasil ditambahkan');

    } else {

        // dd($request->lokasi);

        if (isset($request->tgl_kejadian)) {

            $date_kejadian = date('Y-m-d H:i:s',
strtotime($request->tgl_kejadian));

        } else {

            $date_kejadian = null;

        }

        $laporan = LaporanModel::find($request->id)

        ->update([

            'pelapor_id' => Auth::id(),

            'laporan_terkait_id' => $request->
>laporan_terkait_id,

            'unit_kerja_kode' => $request->unit_kerja,

            'sub_kategori_id' => $request->kategori,

            'progres' => '0',

            'status' => $request->status,

            'tgl_kejadian' => $date_kejadian,

            'pelaku' => $request->pelaku,

            'lokasi' => $request->lokasi,
```

```
        'kronologi' => $request->kronologi,

        'is_anonim' => $request->is_anonim,

        'is_internal' => $request->is_internal,

        'updated_at' => date("Y-m-d H:i:s"),

    ]);

    $msg = 'Draf laporan #' . $request->id . ' berhasil
diperbarui';

    return redirect()->route('draf-laporan', ['id' =>
$request->id])->with('message', $msg);

}

}else if ($request->status == 'Proses Verifikasi') {

    $request->validate([

        'kategori' => 'required',

        'pelaku' => 'required',

        'tgl_kejadian' => 'required',

        'lokasi' => 'required',

        'unit_kerja' => 'required',

        'status' => 'required',

        'is_anonim' => 'required',

        'is_internal' => 'required',

        'files.*' => 'max:102400',

        'files' => 'max:2',

    ]);
```

```
        if (!isset($request->id)) {

            do {

                $randomString = Str::upper(Str::random(6));

            } while (LaporanModel::where('id', $randomString)-
>exists());

            $date_kejadian = date('Y-m-d H:i:s', strtotime($request-
>tgl_kejadian));

            $laporan = LaporanModel::create([

                'id' => $randomString,

                'pelapor_id' => Auth::id(),

                'laporan_terkait_id' => $request-
>laporan_terkait_id,

                'unit_kerja_kode' => $request->unit_kerja,

                'sub_kategori_id' => $request->kategori,

                'progres' => '1',

                'status' => $request->status,

                'tgl_kejadian' => $date_kejadian,

                'pelaku' => $request->pelaku,

                'lokasi' => $request->lokasi,

                'kronologi' => $request->kronologi,

                'is_anonim' => $request->is_anonim,

                'is_internal' => $request->is_internal,

                'created_at' => date("Y-m-d H:i:s"),

                'updated_at' => date("Y-m-d H:i:s"),

            ]);
```

```
$laporan->save();

if ($request->hasFile('files')) {

    $counter = 1;

    $tanggalSekarang = date('dmY');

    $nama_folder =
'storage/Pengaduan/' . $randomString . '_' . $tanggalSekarang . '/';

    $path_folder =
'Pengaduan/' . $randomString . '_' . $tanggalSekarang . '/';

    foreach ($request->file('files') as $file) {

        $filename = 'Bukti Dukung ' . $counter . '_' .
$randomString . '_' . Str::random(10) . '.' . $file-
>getClientOriginalExtension();

        $file->move($nama_folder, $filename); // Simpan
file ke folder1/folder2

        $counter++;

    }

    FolderModel::create([

        'folder' => 'Pengaduan',

        'jenis_file' => 'Bukti Pendukung',

        'path_folder' => $path_folder,

        'laporan_id' => $randomString,

    ]->save());

}
```

```
        return redirect()->route('buat-laporan') -
>with('message', 'Laporan berhasil ditambahkan');

    } else {

        $date_kejadian = date('Y-m-d H:i:s', strtotime($request-
>tgl_kejadian));

        $laporan = LaporanModel::find($request->id)

        ->update([

            'pelapor_id' => Auth::id(),

            'laporan_terkait_id' => $request-
>laporan_terkait_id,

            'unit_kerja_kode' => $request->unit_kerja,

            'sub_kategori_id' => $request->kategori,

            'progres' => '1',

            'status' => $request->status,

            'tgl_kejadian' => $date_kejadian,

            'pelaku' => $request->pelaku,

            'lokasi' => $request->lokasi,

            'kronologi' => $request->kronologi,

            'is_anonim' => $request->is_anonim,

            'is_internal' => $request->is_internal,

            'updated_at' => date("Y-m-d H:i:s"),

        ]);
```

```
        if ($request->hasFile('files')) {

            $counter = 1;

            $tanggalSekarang = date('dmY');

            $nama_folder = 'storage/Pengaduan/'.$request->id.'_'.$tanggalSekarang.'/';

            $path_folder = 'Pengaduan/'.$request->id.'_'.$tanggalSekarang.'/';

            foreach ($request->file('files') as $file) {

                $filename = 'Bukti Dukung ' . $counter . '._'.
                $request->id . '._'.Str::random(10) . '._' . $file->getClientOriginalExtension();

                $file->move($nama_folder, $filename);

                $counter++;

            }

            FolderModel::create([

                'folder' => 'Pengaduan',

                'jenis_file' => 'Bukti Pendukung',

                'path_folder' => $path_folder,

                'laporan_id' => $request->id,

            ]->save());

        }

        return redirect()->route('buat-laporan')->with('message', 'Laporan berhasil ditambahkan');

    }
}
```



Lampiran 7. Kode Untuk Kirim Jawaban Pengaduan

```
public function jawab_pengaduan(Request $request)
{
    // dd($request->all());
    $request->validate([
        'jawaban_pengaduan' => 'required',
    ]);

    DB::table('tbl_laporan')
        ->where('id', $request->laporan_id)
        ->update([
            'progres' => '14',
            'status' => 'Selesai',
            'updated_at' => now(),
        ]);

    DB::table('tbl_jawaban_pengaduan')->insert([
        'laporan_id' => $request->laporan_id,
        'jawaban_pengaduan' => $request->jawaban_pengaduan,
        'created_at' => now(),
        'updated_at' => now(),
    ]);

    if ($request->hasFile('dokumen_jawaban')) {
        $tanggalSekarang = date('dmY');
        $nama_folder = 'storage/JawabanPengaduan/' . $request->laporan_id . '_' . $tanggalSekarang . '/';
        $path_folder = 'JawabanPengaduan/' . $request->laporan_id . '_' . $tanggalSekarang . '/';
        $file = $request->file('dokumen_jawaban');
        $extension = $file->getClientOriginalExtension(); // Get the file extension
        $filename = 'Dokumen Jawaban Pengaduan ' . $request->laporan_id . '_' . Str::random(10) . '.' . $extension; // Append the extension
        $file->move($nama_folder, $filename);
    }

    return redirect()->route('detail-pengaduan', $request->laporan_id);
}
```