

Pengembangan Aplikasi SIMPEL-DUK Berbasis Web untuk Pengelolaan Laporan (Studi Kasus: Disdukcapil Kabupaten Magetan)

Winda Etika Sari¹, Ronggo Alit²

^{1,2} Teknik Informatika/Teknik Informatika, Universitas Negeri Surabaya

¹windaetika@mhs.unesa.ac.id

²ronggoalit@unesa.ac.id

Abstrak— Pengelolaan laporan harian pelayanan administrasi kependudukan merupakan salah satu kegiatan penting dalam mendukung monitoring, evaluasi, dan pengambilan keputusan pada Dinas Kependudukan dan Pencatatan Sipil (Disdukcapil) Kabupaten Magetan. Namun, proses pelaporan yang masih dilakukan secara manual atau menggunakan media yang tidak terintegrasi menyebabkan berbagai permasalahan, seperti tingginya potensi kesalahan *input* data, duplikasi data, keterlambatan penyusunan laporan, serta kesulitan dalam rekapitulasi layanan dan pengelolaan penggunaan blanko KTP-EL dan blanko KIA. Penelitian ini bertujuan untuk merancang dan mengimplementasikan aplikasi SIMPEL-DUK (Sistem Informasi Pelaporan Pelayanan Dukcapil) berbasis web yang terintegrasi guna mendukung proses pencatatan, pengolahan, rekapitulasi, dan monitoring laporan pelayanan administrasi kependudukan di Disdukcapil Kabupaten Magetan. Fitur utama yang disediakan meliputi autentikasi pengguna, pengelolaan laporan harian, verifikasi laporan, rekapitulasi layanan, pengelolaan penggunaan blanko KTP-EL dan blanko KIA, pencarian data, serta pencetakan laporan. Hasil pengujian menggunakan metode Black Box Testing menunjukkan bahwa seluruh fitur aplikasi berjalan sesuai dengan kebutuhan fungsional yang telah ditetapkan dan berhasil memberikan keluaran yang sesuai pada setiap skenario pengujian. Aplikasi SIMPEL-DUK mampu mengintegrasikan proses pelaporan dalam satu basis data terpusat, mempercepat penyusunan laporan, serta memudahkan monitoring dan evaluasi kinerja pelayanan administrasi kependudukan.

Kata Kunci— SIMPEL-DUK, Sistem Informasi Pelaporan, Administrasi Kependudukan, Pelayanan Dukcapil, Monitoring Layanan.

I. PENDAHULUAN

Perkembangan teknologi informasi telah mendorong transformasi digital pada berbagai sektor, termasuk pemerintahan. Pemanfaatan sistem informasi menjadi salah satu strategi untuk meningkatkan kecepatan, transparansi, dan akurasi pelaporan pelayanan publik. Dalam penyelenggaraan administrasi kependudukan, ketersediaan informasi yang cepat dan akurat menjadi kebutuhan penting karena data pelayanan digunakan sebagai dasar monitoring, evaluasi, serta pengambilan keputusan oleh pimpinan instansi. Oleh sebab itu, proses pelaporan pelayanan administrasi kependudukan perlu didukung oleh sistem yang mampu mengintegrasikan proses pencatatan, pengolahan, penyimpanan, dan penyajian data secara terstruktur.

Disdukcapil Kabupaten Magetan menyelenggarakan pelayanan administrasi kependudukan melalui 19 titik pelayanan yang terdiri atas 17 kecamatan, satu Mall Pelayanan Publik (MPP), dan satu kantor Disdukcapil. Setiap titik pelayanan menghasilkan laporan harian yang harus direkap sebagai bahan monitoring. Berdasarkan hasil observasi, pelaporan masih dilakukan menggunakan *spreadsheet* dan *Google Form* yang belum terintegrasi sehingga operator harus melakukan pencatatan berulang, admin melakukan rekapitulasi manual, perhitungan penggunaan blanko KTP-el dan blanko KIA masih dilakukan secara manual, serta proses monitoring membutuhkan waktu lebih lama. Kondisi ini meningkatkan risiko kesalahan pencatatan, duplikasi data, inkonsistensi data, dan keterlambatan penyampaian informasi kepada pimpinan.

Digitalisasi pelaporan merupakan proses pengalihan pengelolaan laporan dari sistem manual menuju sistem berbasis web sehingga proses pengumpulan, pengolahan, dan analisis data menjadi lebih efektif dan terstruktur [1],[2]. Aplikasi berbasis web dipilih karena dapat diakses melalui browser tanpa instalasi dan mendukung penggunaan pada berbagai perangkat [3],[4]. Pengembangan aplikasi pada penelitian ini menggunakan *framework* Laravel yang menerapkan arsitektur MVC sehingga memudahkan pengembangan aplikasi yang terstruktur, mudah dipelihara, dan dikembangkan [5],[6]. Proses pengembangan sistem menggunakan metode *Waterfall* karena memberikan tahapan yang sistematis mulai dari analisis kebutuhan hingga pengujian [7], sedangkan pengujian fungsional dilakukan menggunakan *Black Box Testing* untuk memastikan seluruh fungsi sistem berjalan sesuai kebutuhan pengguna [8],[9].

Beberapa penelitian terdahulu menunjukkan bahwa sistem pelaporan berbasis web mampu meningkatkan efisiensi pengelolaan data. Rahayuningsih dan Fradinata [10] mengembangkan sistem administrasi pelaporan realisasi SKPD berbasis website yang meningkatkan efisiensi dan akurasi pelaporan, namun belum mendukung pelaporan harian administrasi kependudukan. Safitri dkk. [11] merancang sistem pelaporan perkembangan KUBE berbasis web yang mempercepat dokumentasi laporan, tetapi belum menyediakan rekapitulasi otomatis yang komprehensif. Putra dkk. [12] mengembangkan sistem laporan harian operasional kapal berbasis web yang meningkatkan efektivitas monitoring, tetapi masih terbatas pada sektor pelayaran. Alam dkk. [13] membangun sistem laporan harian kinerja pegawai berbasis

web yang meningkatkan efisiensi monitoring, namun belum mengelola layanan administrasi kependudukan. Harliana dkk. [14] membuktikan bahwa sistem *e-report* berbasis web mampu meningkatkan keteraturan pelaporan, meskipun diterapkan pada bidang pendidikan.

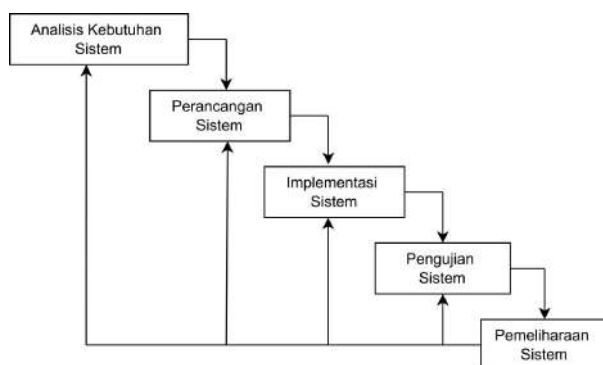
Berdasarkan penelitian-penelitian tersebut dapat disimpulkan bahwa penerapan sistem informasi berbasis web terbukti mampu meningkatkan efisiensi pelaporan pada berbagai bidang. Namun, belum terdapat penelitian yang secara khusus mengintegrasikan pencatatan laporan harian pelayanan administrasi kependudukan, rekapitulasi layanan, pengelolaan penggunaan blanko KTP-el dan blanko KIA, verifikasi laporan, pencarian data, pencetakan laporan, serta monitoring dalam satu sistem terpusat sesuai proses bisnis Disdukcapil. Kesenjangan tersebut menjadi dasar pengembangan aplikasi SIMPEL-DUK.

Berdasarkan permasalahan tersebut, penelitian ini difokuskan pada bagaimana merancang dan mengimplementasikan aplikasi pelaporan harian berbasis web yang terintegrasi untuk mendukung pencatatan, rekapitulasi, perhitungan penggunaan blanko, serta monitoring pelayanan administrasi kependudukan di Disdukcapil Kabupaten Magetan. Penelitian ini bertujuan merancang dan membangun aplikasi SIMPEL-DUK berbasis web menggunakan Laravel dengan metode *Waterfall* sehingga mampu mengintegrasikan proses pelaporan harian, pengelolaan data pelayanan, rekapitulasi, pengelolaan blanko, dan penyajian informasi secara *real-time*. Kontribusi penelitian ini adalah menghasilkan aplikasi yang sesuai dengan kebutuhan operasional Disdukcapil Kabupaten Magetan serta dapat meningkatkan efisiensi pengelolaan laporan harian dan kualitas informasi sebagai dasar pengambilan keputusan.

II. METODE PENELITIAN

A. Model Waterfall

Penelitian ini merupakan penelitian rekayasa perangkat lunak yang bertujuan merancang, membangun, dan menguji aplikasi SIMPEL-DUK berbasis web untuk mengintegrasikan pengelolaan laporan harian pelayanan administrasi kependudukan. Pengembangan sistem menggunakan model *Waterfall* karena menyediakan tahapan yang sistematis mulai dari analisis kebutuhan, perancangan, implementasi, hingga pengujian. Adapun tahapan model *waterfall* dapat dilihat pada Gbr. 1 berikut ini:



Gbr. 1 Model *Waterfall*

B. Proses Bisnis Diusulkan

Proses bisnis sistem pelaporan menggunakan SIMPEL-DUK diawali ketika operator pelayanan administrasi kependudukan mengakses aplikasi melalui halaman *login* dengan memasukkan *username* dan *password*. Sistem kemudian melakukan validasi data autentikasi dengan membandingkannya terhadap data pengguna yang tersimpan pada *database*. Apabila data yang dimasukkan tidak sesuai, sistem akan menampilkan pesan kesalahan dan meminta operator untuk melakukan *login* kembali. Sebaliknya, apabila data valid, sistem akan memberikan hak akses sesuai peran pengguna dan menampilkan halaman utama operator.

Setelah berhasil masuk ke dalam aplikasi, operator melakukan *input* laporan harian pelayanan administrasi kependudukan yang meliputi jumlah pelayanan KTP-el, KK, KIA, surat pindah, serta berbagai jenis akta pencatatan sipil sesuai dengan pelayanan yang telah dilaksanakan. Selanjutnya sistem melakukan validasi terhadap data laporan untuk memastikan seluruh data wajib telah terisi dan sesuai dengan aturan yang ditetapkan, seperti jumlah pelayanan tidak bernilai negatif dan penggunaan blanko tidak melebihi stok yang tersedia. Apabila ditemukan kesalahan, sistem akan menampilkan pesan kesalahan sehingga operator dapat memperbaiki data yang dimasukkan. Jika seluruh data telah valid, sistem menyimpan laporan ke dalam *database* dengan status menunggu verifikasi dan menampilkan notifikasi bahwa laporan berhasil disimpan.

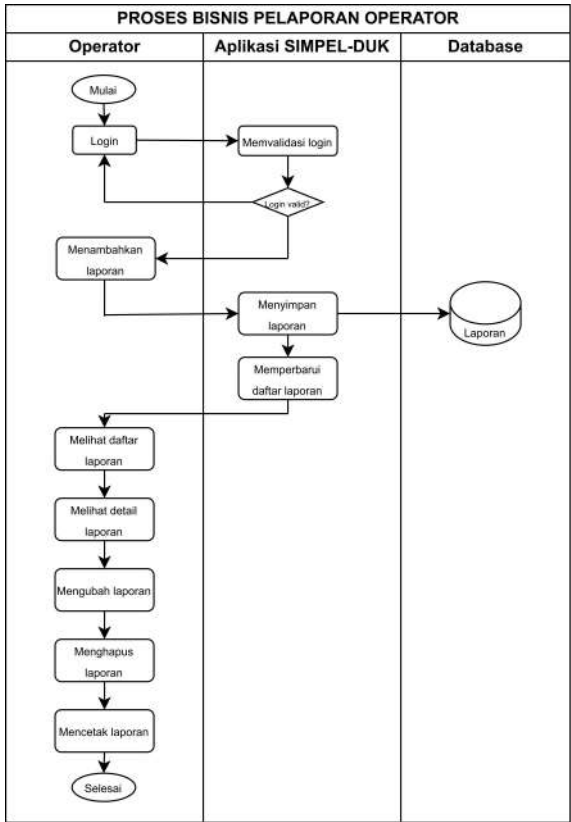
Laporan yang telah dikirim kemudian masuk ke daftar verifikasi admin. Admin melakukan *login* menggunakan akun administrator untuk mengakses menu verifikasi dan pengelolaan sistem. Setelah berhasil masuk, admin memeriksa kesesuaian data laporan dengan kondisi pelayanan yang sebenarnya. Apabila ditemukan kesalahan atau data belum lengkap, admin dapat menolak laporan sehingga operator dapat melakukan perbaikan dan mengirimkannya kembali. Sebaliknya, apabila laporan telah sesuai, admin memverifikasi laporan sehingga statusnya berubah menjadi terverifikasi dan perubahan tersebut disimpan ke dalam *database*.

Setelah laporan disetujui, sistem secara otomatis memperbarui stok blanko dengan mengurangi jumlah blanko yang digunakan pada laporan sehingga informasi stok dapat dipantau secara *real-time*. Selain itu, admin juga dapat melakukan pengelolaan blanko, seperti menambah stok, mendistribusikan blanko ke kecamatan, melakukan koreksi stok, serta melihat riwayat perubahan. Seluruh aktivitas tersebut disimpan ke dalam *database* sebagai histori penggunaan dan distribusi blanko.

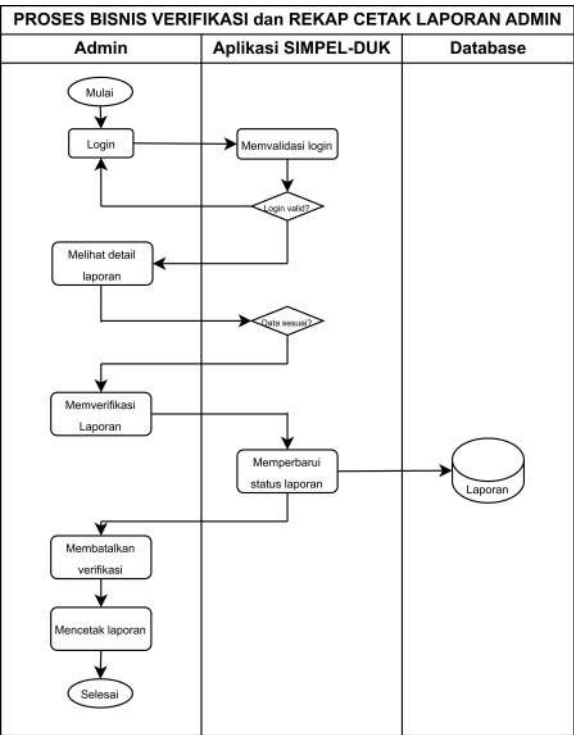
Tahap berikutnya adalah proses rekapitulasi laporan. Sistem mengambil seluruh laporan yang telah berstatus terverifikasi sesuai periode yang dipilih, kemudian mengolah data menjadi informasi rekapitulasi berupa jumlah pelayanan per jenis dokumen, jumlah pelayanan per kecamatan, penggunaan blanko, dan statistik pelayanan. Hasil rekapitulasi dapat dicetak dalam dokumen PDF sebagai bahan pelaporan, evaluasi kinerja, serta pendukung pengambilan keputusan.

Setelah seluruh proses selesai, operator maupun admin dapat melakukan *logout*. Sistem akan menghapus sesi

otentikasi yang sedang aktif dan mengarahkan pengguna kembali ke halaman *login* sehingga keamanan data dan hak akses pengguna tetap terjaga. Alur visual dari proses bisnis aplikasi SIMPEL-DUK dapat dilihat pada Gbr. 2 dan Gbr. 3 berikut ini:



Gbr. 2 Proses Bisnis Pelaporan



Gbr. 3 Proses Bisnis Verifikasi dan Rekap Cetak Laporan

C. Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam proses pengembangan dan uji coba fitur baru adalah sebagai berikut:

Processor : Intel Core i5-11400H
RAM : 16 GB
Penyimpanan : SSD 512 GB
Sistem Operasi : Windows 11 Home 64-bit

D. Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk menunjang pengembangan aplikasi adalah Visual Studio Code, Apache Web Server, PHP, Laravel, Database MySQL, Xampp, Browser.

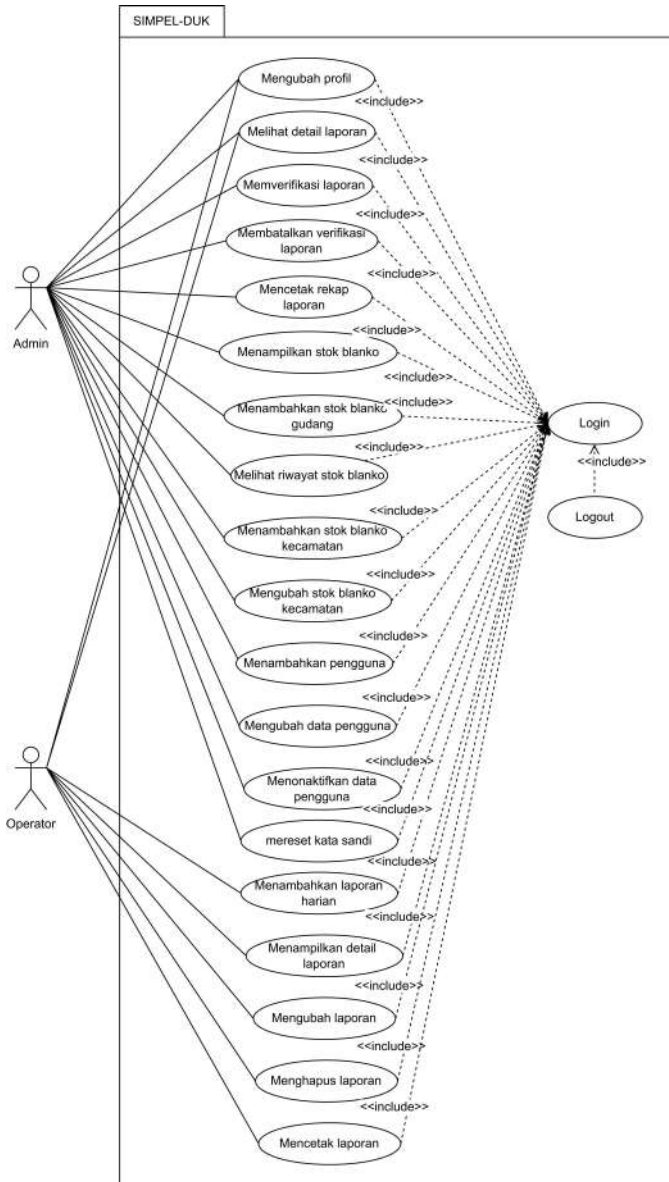
E. Kebutuhan Fungsional

Adapun kebutuhan yang harus dipenuhi oleh aplikasi SIMPEL-DUK adalah sebagai berikut:

1. Sistem mampu melakukan autentikasi pengguna (*login*) sesuai hak akses admin dan operator.
2. Sistem mampu mengelola data laporan harian administrasi kependudukan, meliputi penambahan, perubahan, penghapusan, penampilan detail, serta pencetakan laporan.
3. Sistem mampu memverifikasi dan membatalkan verifikasi laporan harian.
4. Sistem mampu menghasilkan dan mencetak rekapitulasi laporan berdasarkan periode yang dipilih.
5. Sistem mampu mengelola data stok blanko gudang dan stok blanko kecamatan, termasuk penambahan, perubahan, serta menampilkan riwayat stok blanko.
6. Sistem mampu mengelola data pengguna, meliputi penambahan, perubahan, dan penghapusan pengguna.
7. Sistem mampu mengakhiri sesi penggunaan aplikasi (*logout*).

F. Use Case

Use case diagram aplikasi SIMPEL-DUK dapat dilihat pada Gbr. 4. Diagram tersebut melibatkan dua aktor, yaitu admin dan operator, yang memiliki hak akses berbeda sesuai dengan tugas dan tanggung jawabnya. Sebelum mengakses sistem, kedua aktor harus melakukan autentikasi melalui proses *login*. Operator memiliki hak akses untuk mengelola laporan harian pelayanan administrasi kependudukan, meliputi menambahkan, mengubah, menghapus, melihat detail, dan mencetak laporan harian. Selain itu, operator dapat mengubah data profil serta mengakhiri sesi penggunaan aplikasi (*logout*). Sementara itu, admin memiliki hak akses yang lebih luas, yaitu memverifikasi dan membatalkan verifikasi laporan, mencetak rekapitulasi laporan, mengelola stok blanko gudang dan kecamatan, mengelola data pengguna, melihat detail laporan, mengubah data profil, serta mengakhiri sesi penggunaan aplikasi (*logout*). Dengan adanya pembagian hak akses tersebut, setiap aktor hanya dapat menjalankan fungsi yang sesuai dengan kewenangannya sehingga keamanan dan pengelolaan data dalam aplikasi SIMPEL-DUK dapat terjaga dengan baik.



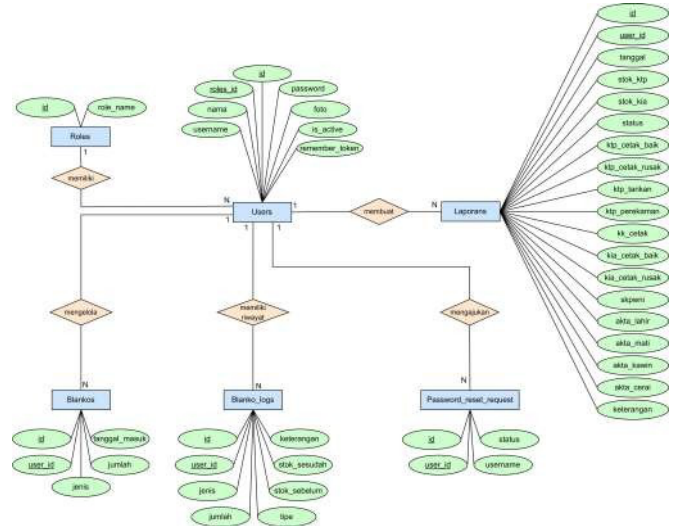
Gbr. 4 Diagram *Use Case*

G. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) digunakan untuk menggambarkan struktur basis data serta hubungan antar entitas pada aplikasi SIMPEL-DUK. Berdasarkan hasil perancangan, terdapat enam entitas utama, yaitu Roles, Users, Laporan, Blankos, Blanko_logs, dan Password_reset_request. Entitas Roles digunakan untuk menyimpan data peran pengguna dan memiliki hubungan *one-to-many* dengan entitas Users, sehingga satu peran dapat dimiliki oleh beberapa pengguna. Entitas Users berfungsi sebagai pusat pengelolaan data pengguna dan memiliki relasi dengan beberapa entitas lain, seperti Laporan, Blankos, Blanko_logs, dan Password_reset_request.

Entitas `Laporans` digunakan untuk menyimpan data laporan harian pelayanan administrasi kependudukan yang dibuat oleh operator. Entitas `Blankos` digunakan untuk mengelola data stok blanko, sedangkan `Blanko_logs` berfungsi untuk mencatat riwayat perubahan stok blanko yang terjadi pada sistem.

Selain itu, entitas `Password_reset_request` digunakan untuk menyimpan data pengajuan reset kata sandi oleh pengguna. Perancangan basis data ini bertujuan untuk menjaga integritas data, mengurangi duplikasi data, serta mendukung proses pengelolaan laporan pelayanan, pengelolaan stok blanko, pencatatan aktivitas, dan manajemen pengguna secara terintegrasi. ERD aplikasi SIMPEL-DUK ditunjukkan pada Gbr. 5.



Gbr. 5 *Entity Relationship Diagram* (ERD)

III. HASIL DAN PEMBAHASAN

A. Implementasi Source Code Sistem

Tahapan implementasi *source code* pada aplikasi SIMPEL-DUK dilakukan dengan menerjemahkan hasil perancangan sistem ke dalam bentuk program menggunakan framework Laravel dan database MySQL. Implementasi ini meliputi pengembangan fitur autentikasi pengguna (*login*), pengelolaan laporan harian, verifikasi laporan, kelola blanko serta rekapitulasi dan pencetakan laporan sesuai dengan proses bisnis yang telah dirancang sebelumnya.

1. Source Code Fitur Login

```
public function store(LoginRequest $request): RedirectResponse
{
    $request->authenticate();

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

    return redirect()->intended(route('dashboard', absolute: false));
}
```

Gbr. 6 *Source Code Method store()*

Method store() berfungsi untuk menerima data *login* yang dikirimkan pengguna melalui objek *LoginRequest* dan melakukan proses autentikasi dengan mencocokkan *username* dan *password* dengan data yang tersimpan pada basis data menggunakan perintah `$request->authenticate()`. Apabila autentikasi berhasil, sistem akan membuat session baru menggunakan `$request->session()->regenerate()` untuk meningkatkan keamanan, kemudian mengarahkan pengguna ke halaman utama sistem. Sebaliknya, jika autentikasi gagal, pengguna akan dikembalikan ke halaman *login* disertai pesan kesalahan.

2. Source Code Fitur Laporan Harian

```
public function store(Request $request)
{
    $user = Auth::user();
    $stockKtp = $this->getAvailableStock($user, 'ktp');
    $stockKia = $this->getAvailableStock($user, 'kia');

    $validator = Validator::make($request->all(), [
        'tanggal' => 'required',
        'ktp_cetak_baik' => 'required|integer|min:0',
        'ktp_cetak_rusak' => 'required|integer|min:0',
        'ktp_tarikan' => 'required|integer|min:0',
        'ktp_perekaman' => 'required|integer|min:0',
        'kk_cetak' => 'required|integer|min:0',
        'kia_cetak_baik' => 'required|integer|min:0',
        'kia_cetak_rusak' => 'required|integer|min:0',
        'akta_lahir' => 'required|integer|min:0',
        'akta_nati' => 'required|integer|min:0',
        'akta_kawin' => 'required|integer|min:0',
        'akta_cerai' => 'required|integer|min:0',
        'skpmi' => 'required|integer|min:0',
        'keterangan' => 'nullable|string',
    ], [
        'tanggal.unique' => 'Laporan untuk tanggal ini sudah dibuat.',
    ]);

    if ($validator->fails()) {
        return redirect()->back()->withErrors($validator)->withInput();
    }

    $laporan = create([...
    ]);

    return redirect()->route('operator.index')->with('success', 'Laporan berhasil ditambahkan');
}
```

Gbr. 7 Source Code Method store()

```
public function print(Laporan $laporan)
{
    if ($laporan->user_id != Auth::id() && Auth::user()->role_id != 2) {
        abort(403);
    }

    // Hanya laporan yang sudah diverifikasi yang boleh dicetak
    if ($laporan->status != 'verified') {
        return redirect()->back()->with('error', 'laporan belum diverifikasi dan tidak dapat dicetak.');
```

Gbr. 8 Source Code Method print()

Method store() digunakan untuk menerima dan menyimpan data laporan harian yang dikirimkan oleh operator melalui form tambah laporan, dengan terlebih dahulu melakukan validasi terhadap seluruh data masukan untuk memastikan kesesuaian format data serta mencegah pengguna membuat lebih dari satu laporan pada tanggal yang sama. Sementara itu, method print() digunakan untuk mencetak laporan harian ke dalam format PDF menggunakan library DomPDF, dengan memastikan bahwa hanya laporan yang telah berstatus terverifikasi yang dapat dicetak. Setelah proses pencetakan berhasil dilakukan, sistem akan mengunduh dokumen PDF dengan nama dokumen yang disesuaikan dengan tanggal laporan yang dicetak.

3. Source Code Fitur Verifikasi Laporan

```
public function laporanVerify(Laporan $laporan)
{
    DB::transaction(function () use ($laporan) {
        $laporan->update(['status' => 'verified']);

        $user = $laporan->user;
        $adminId = auth()->id();

        // Kurangi stok KTP
        $ktpJumlah = $laporan->ktp_jumlah;
        if ($ktpJumlah > 0) {
            $this->kurangiStokBlanko($user, 'ktp', $ktpJumlah, $adminId,
                'Penggunaan blanko KTP berdasarkan laporan diverifikasi');
        }

        // Kurangi stok KIA
        $kiaJumlah = $laporan->kia_jumlah;
        if ($kiaJumlah > 0) {
            $this->kurangiStokBlanko($user, 'kia', $kiaJumlah, $adminId,
                'Penggunaan blanko KIA berdasarkan laporan diverifikasi');
        }
    });

    return redirect()->route('admin.laporan.index')->with('success',
        'Laporan berhasil diverifikasi');
```

Gbr. 9 Source Code Method laporanVerify()

Method laporanVerify() digunakan untuk melakukan verifikasi terhadap laporan harian yang diajukan oleh operator dengan menjalankan seluruh proses menggunakan DB::transaction() agar perubahan data dilakukan secara konsisten dan dapat dibatalkan apabila terjadi kegagalan. Setelah status laporan diubah menjadi terverifikasi, sistem akan menghitung penggunaan blanko KTP-el dan blanko KIA berdasarkan data laporan, kemudian secara otomatis mengurangi stok blanko serta mencatat riwayat perubahan stok ke dalam riwayat blanko. Apabila seluruh proses berhasil dilakukan, sistem akan mengarahkan admin kembali ke halaman verifikasi laporan dan menampilkan notifikasi bahwa laporan berhasil diverifikasi.

```
public function laporanUnverify(Laporan $laporan)
{
    DB::transaction(function () use ($laporan) {
        $laporan->update(['status' => 'pending']);

        $user = $laporan->user;
        $adminId = auth()->id();

        // Tambah kembali stok KTP
        $ktpJumlah = $laporan->ktp_jumlah;
        if ($ktpJumlah > 0) {
            $this->tambahStokBlanko($user, 'ktp', $ktpJumlah, $adminId,
                'Pembatalan verifikasi laporan - blanko KTP dikembalikan');
        }

        // Tambah kembali stok KIA
        $kiaJumlah = $laporan->kia_jumlah;
        if ($kiaJumlah > 0) {
            $this->tambahStokBlanko($user, 'kia', $kiaJumlah, $adminId,
                'Pembatalan verifikasi laporan - blanko KIA dikembalikan');
        }
    });

    return redirect()->route('admin.laporan.index')->with('success',
        'Verifikasi laporan dibatalkan');
```

Gbr. 10 Source Code Method laporanUnverify()

Method laporanUnverify() digunakan untuk membatalkan verifikasi laporan yang sebelumnya telah disetujui oleh admin dengan menjalankan seluruh proses menggunakan DB::transaction() untuk menjaga konsistensi data. Setelah status laporan diubah kembali menjadi pengajuan, sistem akan mengembalikan jumlah blanko KTP-el dan blanko KIA yang sebelumnya telah dikurangi ke stok kecamatan serta mencatat perubahan tersebut ke dalam riwayat blanko. Setelah seluruh proses selesai, sistem akan menampilkan notifikasi bahwa pembatalan verifikasi laporan berhasil dilakukan.

4. Source Code Fitur Kelola Blanko

```
public function store(Request $request)
{
    // Pastikan user_id selalu dalam bentuk array untuk mendukung bulk and single
    if ($request->has('user_id') && !is_array($request->user_id)) {
        $request->merge(['user_id' => [$request->user_id]]);
    }

    $request->validate([...
    ]);

    $userIds = $request->user_id;
    $countUsers = count($userIds);
    $jumlahPerUser = $request->jumlah;
    $totalJumlah = $countUsers * $jumlahPerUser;

    $warehouseStock = Blanko::whereNull('user_id')
        ->where('jenis', $request->jenis)
        ->sum('jumlah');

    if ($warehouseStock < $totalJumlah) {
        // ...
    }

    DB::transaction(function () use ($request, $userIds, $jumlahPerUser, $warehouseStock) {
        // ...
    });

    $successMsg = $countUsers > 1 ? 'Berhasil menambahkan ' . $countUsers . ' jenis blanko.' : 'Berhasil menambahkan 1 jenis blanko.';
    return redirect()->route('admin.blanko.dashboard', ['jenis' => $request->jenis])
        ->with('success', $successMsg);
}
```

Gbr. 11 Source Code Method store()

Method store() pada Gbr. 11 digunakan untuk mendistribusikan blanko dari gudang kepada operator kecamatan. Sistem akan mengurangi stok gudang dan menambahkan stok pada kecamatan tujuan sesuai jumlah blanko yang didistribusikan, kemudian mencatat proses tersebut pada riwayat blanko.

5. Source Code Fitur Rekap dan Cetak Laporan

```
public function printRekap(Request $request)
{
    $request->validate([
        'tanggal' => 'required|date',
        'tipe' => 'required|in:harian,bulanan',
    ]);

    $date = Carbon::createFromFormat('d F Y', $request->tanggal);
    $query = Laporan::where('status', 'verified');

    if ($request->tipe == 'harian') {
        $query->whereDate('tanggal', $date);
        $periode = 'Tanggal ' . $date->translatedFormat('d F Y');
    } else {
        $query->whereMonth('tanggal', $date->month)
            ->whereYear('tanggal', $date->year);
        $semester = ($date->month <= 6) ? 'I' : 'II';
        $periode = 'Bulan ' . $date->translatedFormat('F Y') . ' periode ' . $semester;
    }

    $laporan = $query->get();

    if ($laporan->isEmpty()) {
        if ($request->ajax()) {
            return response()->json(['error' => 'Tidak ada laporan terverifikasi pada periode tersebut.', 404]);
        }
        return redirect()->back()->with('error', 'Tidak ada laporan terverifikasi pada periode tersebut.')->with('message', 'Gagal');
    }

    $rekap = [
        // ...
    ];

    $pdf = Pdf::loadView('admin.laporan.print_rekap', compact('rekap'));
    return $pdf->download('rekap_' . $request->tipe . '_' . $date->format('Y-m-d') . '.pdf');
}
```

Gbr. 12 Source Code Method printRekap()

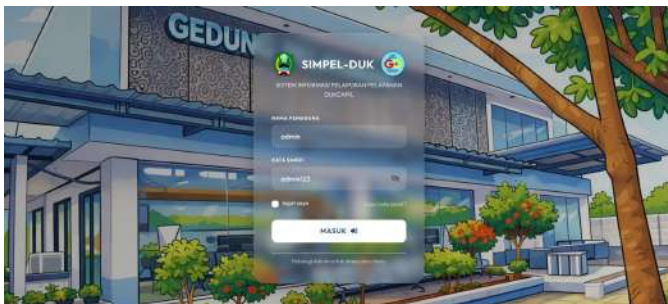
Method printRekap() digunakan untuk menghasilkan laporan rekapitulasi pelayanan administrasi kependudukan dalam format PDF berdasarkan laporan yang berstatus terverifikasi. Sistem mengambil data sesuai dengan kategori laporan harian atau bulanan. Selanjutnya, sistem menghitung total seluruh jenis pelayanan dari seluruh kecamatan dan mengirimkan hasil rekapitulasi tersebut ke template laporan menggunakan library DomPDF untuk menghasilkan dokumen PDF. Setelah proses selesai, dokumen laporan akan diunduh secara otomatis oleh admin.

B. Implementasi Aplikasi Web SIMPEL-DUK

Fitur yang dikembangkan memenuhi kebutuhan fungsional yang sudah dijelaskan sebelumnya pada tahapan perancangan sistem yaitu kebutuhan fungsional.

1. Fitur Login

Fitur login pada aplikasi SIMPEL-DUK merupakan halaman yang digunakan pengguna sebelum dapat mengakses aplikasi. Pengguna diminta memasukkan informasi akun yang telah terdaftar pada aplikasi berupa nama pengguna dan kata sandi. Setelah proses autentikasi berhasil dilakukan, sistem akan mengarahkan pengguna menuju halaman utama sesuai dengan hak akses yang dimiliki. Gbr. 13 berikut ini tampilan fitur login.



Gbr. 13 Fitur Login

2. Fitur Laporan Harian

Fitur laporan harian digunakan oleh operator kecamatan untuk mencatat jumlah pelayanan administrasi kependudukan yang telah dilaksanakan setiap hari. Data laporan yang diinputkan nantinya akan digunakan oleh admin sebagai bahan monitoring dan rekapitulasi laporan. Implementasi fitur laporan harian terdiri dari dua halaman utama, yaitu halaman Daftar Laporan pada Gbr. 14 dan halaman Tambah Laporan Gbr. 15. Berikut ini tampilan fitur laporan harian.

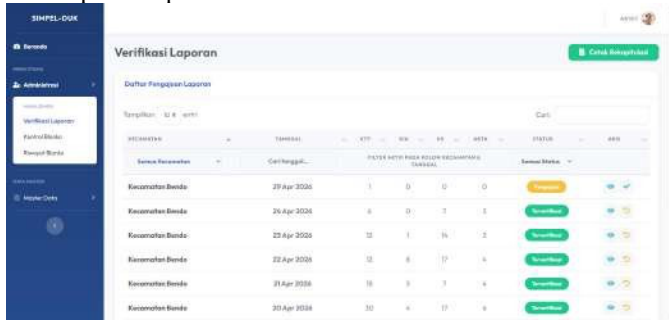
Gbr. 14 Daftar Laporan

Gbr. 15 Tambah Laporan

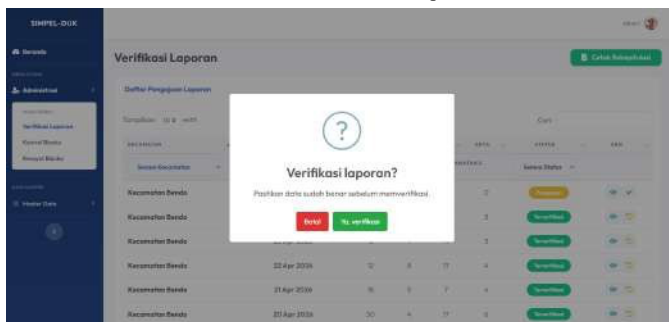
3. Fitur Verifikasi Laporan

Fitur verifikasi laporan digunakan oleh admin untuk memeriksa dan memvalidasi laporan harian yang diajukan oleh operator kecamatan sebelum digunakan sebagai data rekapitulasi pelayanan administrasi kependudukan. Pada halaman ini, sistem menampilkan daftar laporan beserta informasi jumlah pelayanan yang telah dilakukan, seperti KTP-el, KK, KIA, SKPWNI, dan berbagai jenis akta pencatatan sipil, serta menyediakan fitur pencarian dan penyaringan berdasarkan kecamatan maupun status laporan untuk memudahkan pengelolaan data. Selain itu, sistem menampilkan pesan konfirmasi sebelum proses verifikasi

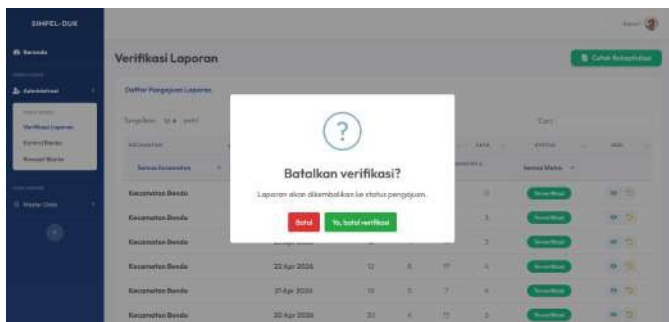
dilakukan guna memastikan data telah diperiksa dan dinyatakan benar. Tampilan fitur verifikasi laporan dapat dilihat pada implementasi berikut :



Gbr. 16 Fitur Verifikasi Laporan



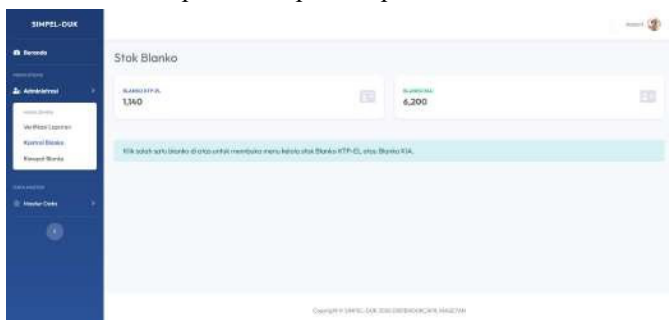
Gbr. 17 Konfirmasi Verifikasi Laporan



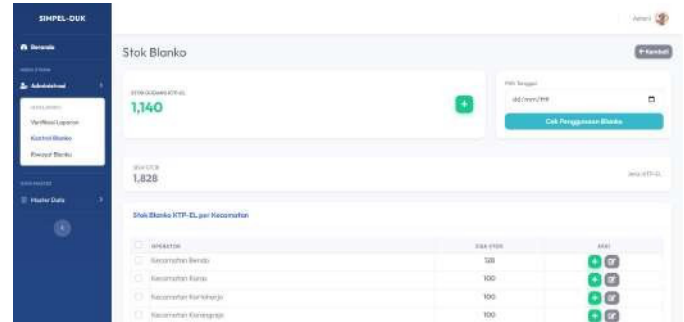
Gbr. 18 Konfirmasi Batalan Verifikasi Laporan

4. Fitur Kelola Blanks

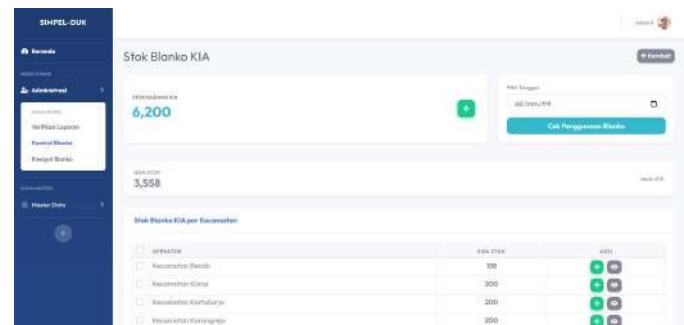
Fitur kelola blanks digunakan oleh admin untuk memantau, mendistribusikan, dan mengelola persediaan blanks KTP-el dan blanks KIA pada gudang dan setiap kecamatan. Melalui fitur ini, admin dapat mengetahui jumlah stok blanks yang tersedia dan menambah stok blanks pada gudang pusat maupun pada masing-masing kecamatan sehingga distribusi blanks dapat dilakukan secara lebih terkontrol. Tampilan fitur kelola blanks dapat dilihat pada implementasi berikut :



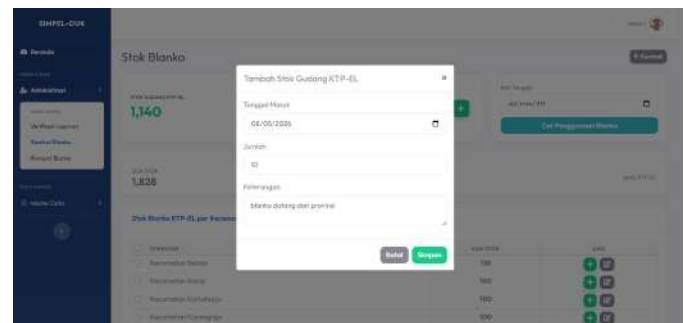
Gbr. 19 Fitur Kelola Blanks



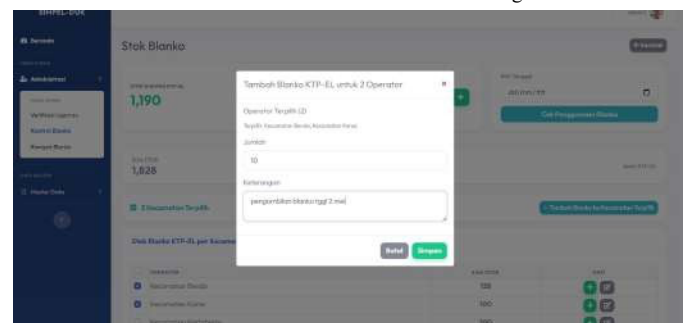
Gbr. 20 Kelola Blanks KTP-el



Gbr. 21 Kelola Blanks KIA



Gbr. 22 Menambah Stok Blanks Gudang

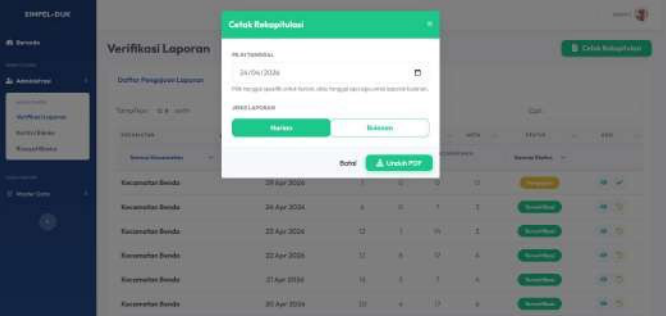


Gbr. 23 Menambah Stok Blanks Kecamatan

5. Fitur Rekap dan Cetak Laporan

Fitur rekapitulasi dan cetak laporan digunakan oleh admin untuk menghasilkan laporan pelayanan administrasi kependudukan berdasarkan data laporan harian yang telah diverifikasi dari seluruh kecamatan. Sistem menyediakan rekapitulasi berdasarkan periode harian maupun bulanan guna memudahkan proses monitoring dan evaluasi pelayanan, serta memungkinkan admin untuk mengunduh hasil rekap laporan dalam dokumen PDF yang dapat digunakan sebagai bahan pelaporan kepada pimpinan maupun sebagai dasar dalam pengambilan keputusan dan evaluasi kinerja pelayanan. Gbr.

24 dan Gbr. 25 berikut ini tampilan fitur rekap dan cetak laporan.



Gbr. 24 Fitur Rekap dan Cetak Laporan



Gbr. 25 Hasil Dokumen Rekap Laporan

C. Pengujian Fitur Baru

Tahapan terakhir setelah implementasi *source code* dan fitur aplikasi *web* adalah melakukan pengujian pada sistem aplikasi menggunakan metode *black box testing*. Hasil pengujian semua fitur pada aplikasi SIMPEL-DUK dapat dilihat pada tabel I berikut ini :

TABEL I
PENGUJIAN FITUR PADA APLIKASI SIMPEL-DUK

No	Fungsi Fitur	Hasil yang Diharapkan	Hasil Uji
1.	Login	Sistem dapat melakukan autentikasi pengguna menggunakan <i>username</i> dan <i>password</i> serta mengarahkan pengguna ke halaman sesuai hak aksesnya.	Berhasil
2.	Laporan Harian	Sistem dapat menambahkan, menampilkan, mengubah, menghapus, dan mencetak laporan harian sesuai dengan data yang dimasukkan oleh operator.	Berhasil
3.	Verifikasi Laporan	Sistem dapat melakukan verifikasi maupun pembatalan verifikasi laporan serta memperbarui status laporan dan stok blanko secara otomatis.	Berhasil
4.	Rekap dan Cetak Laporan	Sistem dapat menghasilkan rekapitulasi laporan berdasarkan periode tertentu dan mencetaknya dalam format PDF.	Berhasil
5.	Kelola Blanko	Sistem dapat menambah, mengurangi, mendistribusikan, serta menampilkan stok blanko dan riwayat perubahan stok.	Berhasil
6.	Manajemen	Sistem dapat menambah,	Berhasil

No	Fungsi Fitur	Hasil yang Diharapkan	Hasil Uji
	Pengguna	mengubah, serta menonaktifkan data pengguna sesuai kebutuhan.	
7.	Kelola Profil	Sistem dapat menampilkan dan memperbarui informasi profil pengguna.	Berhasil
8.	Reset Kata Sandi	Sistem dapat menerima, menyetujui, maupun menolak permintaan reset kata sandi pengguna.	Berhasil
9.	Logout	Sistem dapat mengakhiri sesi pengguna dan mengarahkan pengguna kembali ke halaman login.	Berhasil

Setelah semua fitur aplikasi SIMPEL-DUK dilaksanakan, seluruh hasil pengujian kemudian dirangkum untuk mengetahui tingkat keberhasilan aplikasi SIMPEL-DUK dalam memenuhi kebutuhan fungsional sistem. Rekapitulasi ini menunjukkan jumlah skenario pengujian yang berhasil maupun gagal berdasarkan hasil pengujian menggunakan metode *Black Box Testing*. Rekapitulasi hasil pengujian disajikan pada Tabel II.

TABEL II
HASIL PENGUJIAN BLACK BOX TESTING

Keterangan	Jumlah
Jumlah fitur yang diuji	9
Jumlah fitur dengan status berhasil	9
Jumlah fitur dengan status gagal	0
Tingkat keberhasilan	100%

IV. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa penelitian ini berhasil merancang dan membangun aplikasi SIMPEL-DUK berbasis web untuk mendukung pengelolaan laporan harian pelayanan administrasi kependudukan di Disdukcapil Kabupaten Magetan. Pengembangan sistem dilakukan menggunakan metode *Waterfall* yang meliputi tahapan analisis kebutuhan, perancangan menggunakan UML dan ERD, implementasi, serta pengujian sistem. Hasil implementasi menghasilkan aplikasi yang mampu mengintegrasikan proses autentikasi pengguna, pengelolaan laporan harian, verifikasi laporan, rekapitulasi pelayanan, pengelolaan blanko KTP-el dan KIA, manajemen pengguna, serta pencetakan laporan dalam satu sistem yang terpusat.

Berdasarkan hasil pengujian menggunakan metode *Black Box Testing*, seluruh fitur utama aplikasi telah berjalan sesuai dengan kebutuhan fungsional yang ditetapkan dengan tingkat keberhasilan pengujian mencapai 100% dari 9 fitur pengujian yang dilakukan. Dengan adanya aplikasi SIMPEL-DUK, proses pengelolaan laporan harian, verifikasi, rekapitulasi, pengelolaan penggunaan blanko, serta monitoring pelayanan administrasi kependudukan dapat dilakukan secara lebih terstruktur, terintegrasi, dan efisien sehingga mendukung penyajian informasi dan pengambilan keputusan di Disdukcapil Kabupaten Magetan.

REFERENSI

- [1] I. R. A. Nugroho, A. A. Putrawan, and I. Suasnawa, "Digitalisasi Laporan Kinerja Program Studi Dengan Instrumen Sembilan Kriteria Di Politeknik Negeri Bali I," *J. Teknol. Inf. dan Komput.*, pp. 482–488, 2023..
- [2] D. P. Anggraeni, "Pengembangan Sistem Informasi Laporan Keuangan Digital Berbasis Web untuk Efisiensi Manajemen Keuangan," *J. Penelit. Sist. Inf.*, vol. 1, no. 1, pp. 17–22, 2023, doi: 10.54066/jpsi.v1i1.2629.
- [3] Suherman, M. Villa Waru, and Nurnaningsih, "Perancangan Aplikasi Pendataan Bantuan Langsung Tunai (BLT) Berbasis Web Pada Kantor Desa Maccile Kecamatan Lalabata Kabupaten Soppeng," *J. Ilm. Sist. Inf. dan Tek. Inform.*, vol. 5, no. 2, pp. 117–124, 2022, doi: 10.57093/jisti.v5i2.136.
- [4] S. Pulungan, M. Fakhriza, and A. M. Harahap, "Sistem Pakar Mendiagnosa Penyakit Kanker Nasofaring Sejak Dini Menggunakan Metode Dempster Shafer Berbasis Web," *J. Ilm. Sist. Inf. dan Ilmu Komput.*, vol. 3, no. 2, pp. 59–86, 2023, doi: 10.55606/juisik.v3i2.486.
- [5] F. Maulana, A. Abdussomad, I. Kurniawan, and A. Wibowo, "Implementasi Sistem Informasi Jadwal Bimbingan Perkawinan Pada Bimas Islam Kementerian Agama Kabupaten Karawang," *Simpatik J. Sist. Inf. dan Inform.*, vol. 5, no. 1, pp. 63–70, 2025, doi: 10.31294/simpatik.v5i1.8895.
- [6] Mohammad Bayu Anggara, Aulia Hanifa Ardita, and Rudi Hermawan, "Rancang Bangun Aplikasi Simpan Pinjam Koperasi Menggunakan Framework Laravel (Studi Kasus : Koperasi Mekar Mandiri Sejahtera)," *J. Inform.*, vol. 11, pp. 68–74, 2024.
- [7] D. Murdiani and M. Sobirin, "Perbandingan Metodologi Waterfall Dan RAD Dalam Pengembangan Sistem Informasi," *JINTEKS (Jurnal Inform. Teknol. dan Sains)*, vol. 4, no. 4, pp. 302–306, 2022, [Online]. Available: <https://www.jurnal.uts.ac.id/index.php/JINTEKS/article/download/2008/1065>.
- [8] I. K. Putrawan et al., "Pengukuran Efisiensi Qr Code pada Sistem Presensi Karyawan The Seminyak Beach Resort and Spa," *J. Nas. Komputasi dan Teknol. Inf.*, vol. 8, no. 1, pp. 542–551, 2025, doi: 10.32672/jnkti.v8i1.8272.
- [9] Abi Mayu Abdi, "Tinjauan Langkah Pengujian Perangkat Lunak Dengan Metode Black Box Menggunakan Teknik Equivalence Partitions," *JISCO (Journal Inf. Syst. Comput. ISSN)*, vol. 2, no. 2, pp. 59–65, 2024.
- [10] P. A. Rahayuningsih and F. Fradinata, "Sistem Informasi Administrasi Pelaporan Realisasi Skpd Berbasis Website Pada Bappeda Kabupaten Kubu Raya," *J. Inform. Kaputama*, vol. 6, no. 2, 2022.
- [11] A. B. Y. Gatot Safitri, Ifan Junaedi, "Sistem Informasi Berbasis Web Tentang Pelaporan Perkembangan Kelompok Usaha Bersama (Kube)," *J. Manaj. Inform. Jakarta*, vol. 2, pp. 64–76, 2022.
- [12] Gustian Rama Putra, Agung Fajar Ilmiyono, Raid Rafif, and Dinar Munggaran Akhmad, "Design and Implementation of a Web-Based Vessel Daily Report Information System for Optimizing Operational Efficiency and Accounting in the Shipping Industry," *J-KOMA J. Ilmu Komput. dan Apl.*, vol. 7, no. 2, pp. 9–20, 2025, doi: 10.21009/jkoma.v7i2.02.
- [13] A. Alam, Imilda, and T. Iqbal, "Sistem Informasi Laporan Harian Kinerja Pegawai Kontrak berbasis Web Pada Kantor Satpol PP dan WH Aceh," *J. Sist. Komput.*, vol. 3, no. 2, pp. 65–73, 2023, doi: 10.35870/siskom.v3i2.795.
- [14] F. D. Laksana and R. Suwanda, "Design of a Web-Based Self-Reporting Information System for Teacher Professional Education," *J. Minfo Polgan*, vol. 13, no. 1, pp. 1085–1091, 2024, doi: 10.33395/jmp.v13i1.13932.