

Sistem Pengarsipan Data Berbasis Web Pada Kelurahan Oetete, Kota Kupang Menggunakan *Framework Laravel*

Gede Abdi Candrawira Wicaksana¹, Evangs Mailoa²

^{1,2}*Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Salatiga*

¹672019053@student.uksw.edu, ²evangs.mailoa@uksw.edu

ABSTRAK

Penelitian ini mengambil studi kasus pada kelurahan Oetete, Kota Kupang, Nusa Tenggara Timur. Permasalahan yang dihadapi adalah dokumen arsip yang rentan mengalami kerusakan, tercecer atau hilang, serta waktu yang dibutuhkan untuk melakukan pengarsipan yang memakan waktu kurang lebih empat hari karena memiliki tujuh belas kategori dokumen dan pengarsipan data yang dilakukan masih menggunakan media cetak atau kertas. Juga faktor lain seperti hama rayap dan tempat penyimpanan arsip yang kurang memadai. Luaran dari penelitian ini adalah sebuah sistem pengarsipan dokumen berbasis web yang dibuat menggunakan bahasa pemrograman PHP dan framework laravel. Dengan adanya sistem pengarsipan berbasis web ini, diharapkan dapat membantu masalah pengarsipan data pada Kelurahan Oetete, Kota Kupang, Nusa Tenggara Timur.

Kata kunci : *Arsip digital, Laravel, Sistem pengarsipan*

ABSTRACT

This research takes a case study at Oetete, Kupang City, East Nusa Tenggara. The problems faced are archival documents which are susceptible to damage, being scattered or lost, as well as the time required to archive which takes approximately four days because there are seventeen document categories and data archiving is carried out using print or paper media. There are also other factors such as termites and inadequate archive storage. The output of this research is a web-based document archiving system created using the PHP programming language and the Laravel framework. With this web-based archiving system, it is hoped that it can help with data archiving problems in Oetete, Kupang City, East Nusa Tenggara.

Keywords: *Archiving system, Digital archive, Laravel*

1. PENDAHULUAN

Arsip merupakan catatan dan atau keterangan yang dibuat dan diterima oleh perusahaan atau lembaga pemerintahan dalam pelaksanaan kegiatannya baik tertulis ataupun terekam dalam bentuk atau media apapun [1]. Sebagai sumber informasi, arsip dapat membantu ingatan seseorang mengenai sebuah peristiwa tertentu, sehingga dalam praktiknya arsip

dapat digunakan oleh pimpinan sebuah lembaga dalam pengambilan keputusan saat terjadi permasalahan yang serupa dengan isi arsip, yang membuat arsip harus dijaga agar tidak hilang atau mengalami kerusakan.

Berdasarkan pertimbangan diatas, sistem informasi pengarsipan dalam bentuk digital sangat diperlukan sebagai solusi pengarsipan data yang masih dilakukan menggunakan media cetak. Selain itu,

pengarsipan data akan menjadi lebih cepat dan efisien dengan menggunakan media elektronik atau digital [2].

Arsip digital merupakan kumpulan data yang disimpan dalam bentuk kode-kode biner yang dapat diolah menggunakan bantuan alat komputasi seperti komputer sehingga data dapat digunakan [3].

Pengelolaan arsip digital dapat meliputi tahap penyimpanan dan penemuan kembali arsip. Pengarsipan digital merupakan sebuah proses pengelolaan arsip yang dimulai dari tahap alih media arsip hingga pada penataan arsip dalam media yang baru [4].

Adapun beberapa penelitian yang relevan dan memiliki keterkaitan dengan penelitian ini. Penelitian pertama yang relevan dengan penelitian ini adalah penelitian yang berjudul “Rancang Bangun Sistem Informasi Arsip Berbasis Web Menggunakan *Framework Codeigniter*” yang menghasilkan sebuah sistem berbasis web yang dapat mengelola pencatatan surat masuk, surat keluar, pencarian surat dan penyimpanan arsip secara digital [2].

Penelitian kedua berjudul “Perancangan Sistem Informasi Penyimpanan Data Transaksi Simpan Pinjam Pada Koperasi Berbasis Web” yang membahas mengenai perancangan sebuah sistem untuk mengintegrasikan informasi transaksi dari koperasi yang berbasis web yang berguna sebagai media penyimpanan data koperasi untuk memperlancar proses transaksi mengurangi resiko kehilangan data [5].

Penelitian ketiga berjudul “Implementasi Framework Laravel Pada Sistem Informasi Penyewaan Kamera (Studi Kasus Di Rumah Kamera Semarang)” yang membahas mengenai

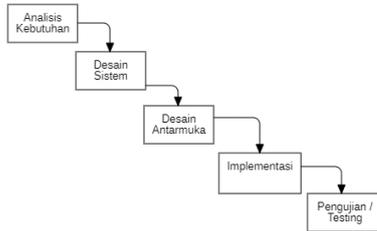
sistem informasi pada tempat penyewaan kamera berbasis web dengan tujuan untuk memudahkan pelanggan dalam proses penyewaan dan melihat jadwal penyewaan kamera serta memudahkan petugas penyewaan dalam mengelola data penyewaan [6].

Penelitian berjudul Sistem Pengarsipan Data Berbasis Web Pada Kelurahan Oetete, Kota Kupang Menggunakan Framework Laravel mengambil studi kasus pada daerah Nusa Tenggara Timur, yaitu pada kelurahan Oetete, Kota Kupang. Luaran yang ingin dicapai adalah sistem berbasis web yang digunakan untuk mengatasi masalah pengarsipan data yang ada pada kelurahan tersebut.

Berdasarkan wawancara dengan pihak terkait, permasalahan yang dihadapi adalah dokumen arsip yang rentan mengalami kerusakan, tercecer atau hilang, serta waktu yang dibutuhkan untuk melakukan pengarsipan yang memakan waktu kurang lebih empat hari karena dokumen yang perlu diarsipkan mempunyai tujuh belas kategori dokumen yang mana proses pengarsipan data-data tersebut masih dilakukan menggunakan media cetak atau kertas. Adapun masalah lainnya seperti tempat penyimpanan arsip yang kurang terawat dan rayap yang dapat merusak dokumen karena terbuat dari bahan kertas.

Dengan adanya penelitian ini, diharapkan pengarsipan data pada Kelurahan Oetete, Kota Kupang dapat dilakukan secara lebih terstruktur dan tersistem. Serta dapat mempermudah dan mempersingkat proses pengarsipan data yang diharapkan kurang dari sebagaimana proses pengarsipan ketika masih menggunakan media cetak.

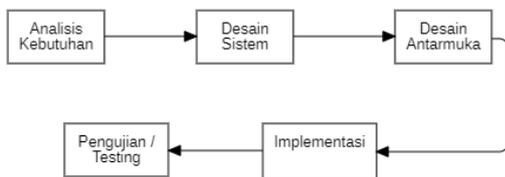
2. METODE PENELITIAN



Gambar 1. Metode Waterfall

Penelitian ini menggunakan pendekatan waterfall. Pendekatan waterfall merupakan pendekatan Software Development Life Cycle (SDLC) untuk pengembangan perangkat lunak yang dilakukan secara sistematis dengan beberapa tahapan dimulai dari tahap perencanaan, analisa, desain dan implementasi pada sistem yang dikerjakan. Tahapan-tahapan pada metode waterfall antara lain adalah requirement analysis, design, development, testing, dan maintenance [7].

Berdasarkan pendekatan waterfall tersebut, tahapan penelitian yang digunakan pada penelitian ini dibagi menjadi lima tahapan antara lain analisis kebutuhan, desain sistem, desain antarmuka, implementasi, dan pengujian yang dapat dilihat pada Gambar 2.

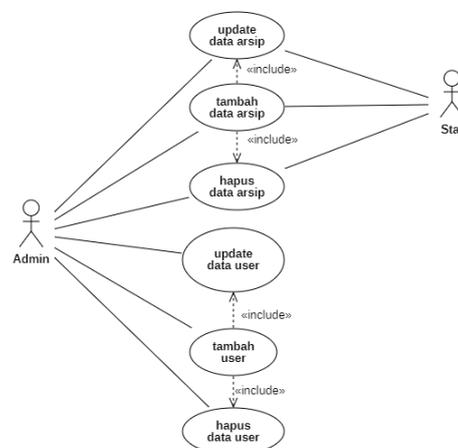


Gambar 2. Tahapan penelitian

Sebelum mengembangkan perangkat lunak, pengembang perlu mengetahui

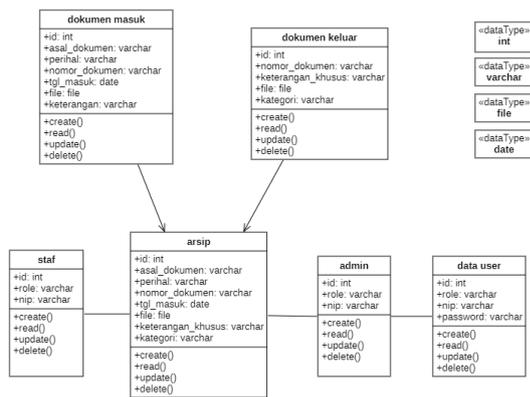
spesifikasi kebutuhan yang dibutuhkan untuk melakukan proses pengembangan perangkat lunak [7]. Studi literatur dilakukan untuk mencari referensi yang relevan dengan penelitian ini [8]. Serta dilakukan wawancara untuk mengetahui dan menganalisis spesifikasi kebutuhan diperlukan untuk mengembangkan sistem sesuai kebutuhan pihak Kelurahan Oetete. Tahap berikutnya dilakukan proses desain sistem meliputi pembuatan UML (Unified Modeling Language) yang berfungsi untuk merincikan sistem yang dibuat dari berbagai aspek [9]. Setelah melakukan perancangan system, dilakukan proses perancangan antarmuka aplikasi. Tahap keempat dilakukan proses implementasi yang mengubah hasil desain system dan antarmuka pada tahap kedua dan ketiga kedalam bentuk kode dengan menggunakan bahasa pemrograman PHP, framework Laravel dan basis data MySQL. Pada tahap kelima, dilakukan pengujian sistem secara menyeluruh untuk mengidentifikasi kemungkinan adanya kegagalan atau kesalahan pada sistem [7].

3. HASIL DAN PEMBAHASAN



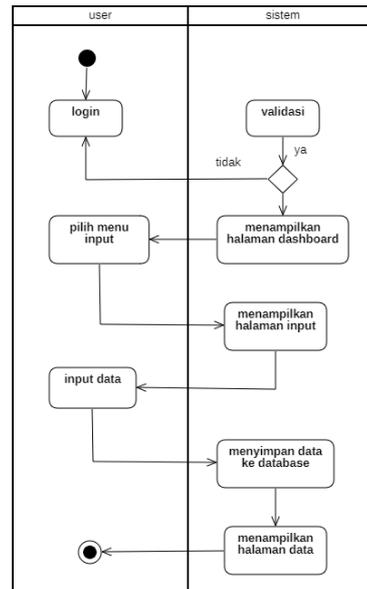
Gambar 3. Usecase Diagram Admin

Gambar 3 menunjukkan usecase diagram yang digunakan. Sistem dibangun dengan menerapkan dua macam aktor antara lain admin dan staf. Admin merupakan sebuah akun yang memiliki hak akses untuk seluruh data yang ada didalam aplikasi ini. Admin dapat melakukan tambah, hapus, dan update data arsip maupun user yang ada didalam sistem. Sedangkan staf memiliki hak akses yang lebih terbatas dibandingkan admin yang mana hanya terbatas pada tambah, hapus, dan update data arsip. Adapun class diagram yang dapat dilihat pada **Gambar 4**.

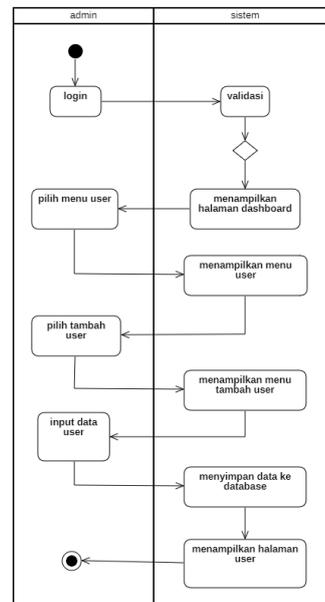


Gambar 4. Class Diagram

Class diagram merupakan jenis diagram yang memperlihatkan class-class yang terdapat dalam sistem yang memiliki atribut dan metode-metodenya. Dimana model ini merepresentasikan hubungan antar class seperti inheritance (pewarisan) dan association (asosiasi). Class diagram berfungsi untuk memudahkan proses perancangan dan pengembangan sistem, serta dapat membantu proses dokumentasi sistem [10]. Class diagram pada penelitian ini dapat pada **Gambar 4**.



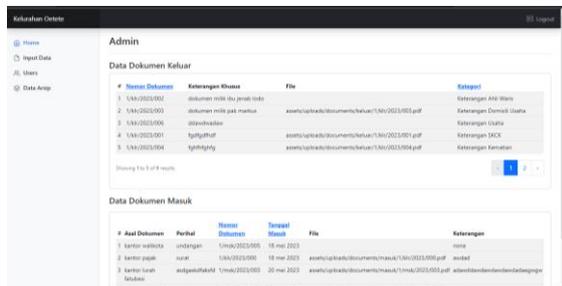
Gambar 5. Activity Diagram Input Data arsip



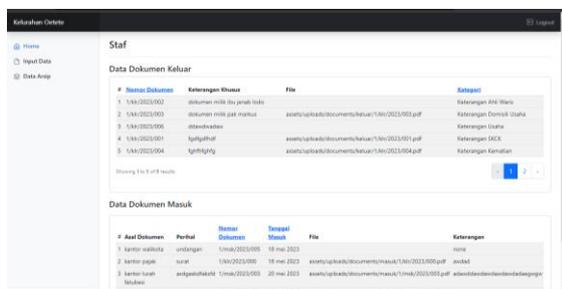
Gambar 6. Activity Diagram Input Data user

Selain use case diagram dan class diagram, juga terdapat activity diagram seperti yang dapat dilihat pada **Gambar 5** dan **Gambar 6**. Activity diagram merupakan aktivitas atau alur kerja dalam suatu sistem. Activity diagram juga dapat

digunakan untuk mengelompokan alur dari tampilan didalam sebuah sistem [11].



Gambar 7. Tampilan Dashboard Admin



Gambar 8. Tampilan dashboard Staf

Tampilan dashboard untuk admin dapat dilihat pada **Gambar 7** dan tampilan dashboard untuk staf dapat dilihat pada **Gambar 8**. Tampilan dashboard admin dan staf menampilkan tabel berisi data arsip yang terakhir ditambahkan oleh pengguna. Terdapat perbedaan pada sidebar dimana admin dapat mengakses menu users sedangkan staf tidak karena pembatasan hak akses berdasarkan role yang telah ditentukan.

Kode program 1. Kode program untuk menampilkan data pada tabel

```
public function datas(Request $request)
{
    $data = DataModel::sortable()->paginate(5)->onEachSide(1);
}
```

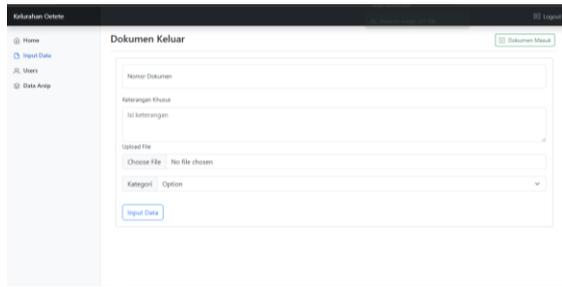
```
return view('admin.datas-admin',
compact('data'), [
'title' => 'Data Arsip',
'name' => 'Kelurahan Oetete',
'desc' => 'Arsip Dokumen Keluar',
'button' => 'Data Arsip'
])->with([
'data' => $data,
]);
}
```

Kode program 2. Kode program untuk menampilkan data pada tabel

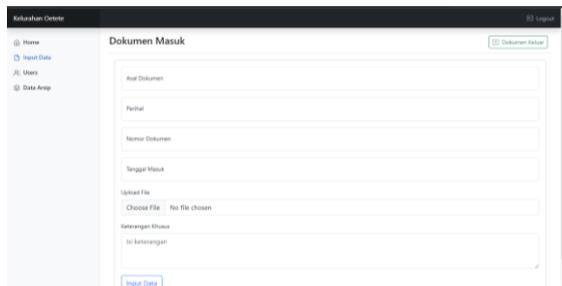
```
public function show(data_masuk $data_masuk, Request $request)
{
    $data_masuk =
data_masuk::sortable()->paginate(5)->onEachSide(1);

return view('admin.datas-admin-masuk', compact('data_masuk'), [
'title' => 'Data Arsip',
'name' => 'Kelurahan Oetete',
'desc' => 'Arsip Dokumen Masuk',
'button' => 'Data Arsip'
])->with([
'data' => $data_masuk,
]);
}
```

Kode program 1 dan **kode program 2** menunjukkan kode yang berfungsi untuk menampilkan data pada tabel. Dimana data diambil dari table data_masuk dan table data_keluar. Data yang diambil dari database disimpan kedalam variable \$data pada baris 3. Kode pada baris 5-12 berfungsi untuk melakukan return ke halaman view agar data dapat tampil seperti pada **Gambar 7** dan **Gambar 8**.



Gambar 9. Halaman input dokumen keluar



Gambar 10. Halaman input dokumen masuk

Input data terbagi menjadi dua bagian yakni input dokumen masuk dan input dokumen keluar. Berdasarkan **Gambar 9** data yang perlu diisi untuk memenuhi form antara lain nomor dokumen, keterangan khusus, file dokumen dalam format pdf, dan kategori dokumen yang akan dikeluarkan oleh pihak Kelurahan. Berbeda dengan dokumen keluar. Dokumen masuk memiliki beberapa data yang berbeda dengan dokumen keluar. Diantaranya ada perihal dan tanggal masuk seperti yang dapat dilihat pada **Gambar 10**.

Kode program 3. Kode program untuk input data dokumen keluar

```
public function store(Request $request)
{
    $request->validate([
        'nomor_dokumen' =>
        'required|max:100',
        'keterangan_khusus' => 'max:255',
        'file' => 'mimes:pdf',
        'kategori' => 'required'
```

```
]);

if ($request->file('file')) {
    $file = $request->file('file');
    $file-
    >move(public_path('assets/uploads/docum
ents/keluar'), $request->nomor_dokumen .
    '.' . $file->getClientOriginalExtension());
}

$data = [
    'nomor_dokumen' => $request-
    >nomor_dokumen,
    'keterangan_khusus' => $request-
    >keterangan_khusus,
    'kategori' => $request->kategori,
    'file' => $request->file('file') ?
    'assets/uploads/documents/keluar/' .
    $request->nomor_dokumen . '.' . $file-
    >getClientOriginalExtension() : null,
];

if (DataModel::create($data)) {
    return redirect('/input')-
    >with('success', 'Berhasil menambah data');
}
}
```

Kode program 4. Kode program untuk input data dokumen masuk

```
public function store(Request $request)
{
    $request->validate([
        'asal_dokumen' =>
        'required|max:255',
        'perihal' => 'required|max:255',
        'nomor_dokumen' =>
        'required|max:100',
        'file' => 'mimes:pdf',
        'tgl_masuk' => 'required|max:100',
        'keterangan' => 'required'
    ]);

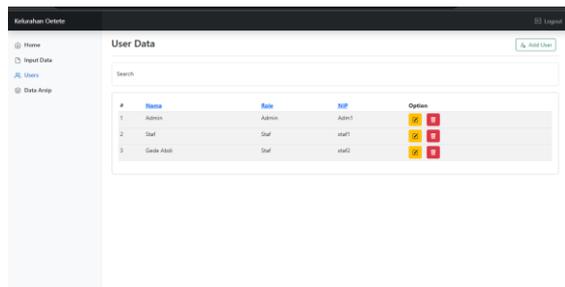
    if ($request->file('file')) {
        $file = $request->file('file');
```

```

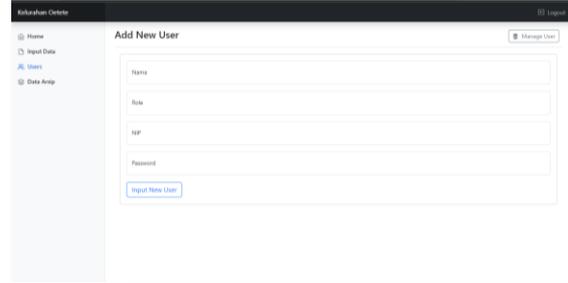
$file-
>move(public_path('assets/uploads/docum
ents/masuk'), $request->nomor_dokumen .
'.' . $file->getClientOriginalExtension());
}
$data = [
'asal_dokumen' => $request-
>asal_dokumen,
'perihal' => $request->perihal,
'nomor_dokumen' => $request-
>nomor_dokumen,
'tgl_masuk' => $request-
>tgl_masuk,
'keterangan' => $request-
>keterangan,
'file' => $request->file('file') ?
'assets/uploads/documents/masuk/'
.$request->nomor_dokumen . '.' . $file-
>getClientOriginalExtension() : null,
];

if (data_masuk::create($data)) {
return redirect('/input-dok-masuk')-
>with('success', 'Berhasil menambah data');
}
}
    
```

Kode program 3 dan **kode program 4** memiliki fungsi yang sama yaitu untuk menambahkan data baru ke database. Variable \$data menyimpan data yang telah di input oleh user. Kemudian model melakukan create data yang disimpan didalam \$data dan mengembalikan response 'Berhasil menambahkan data'.



Gambar 11. Halaman data user



Gambar 12. Halaman tambah user

Menu user hanya dapat diakses oleh Admin. Pada menu users berisi tabel untuk menampilkan data user seperti pada **Gambar 11**. Selain itu, juga terdapat mesin pencari dan tombol untuk beralih halaman ke menu tambah user baru seperti pada **Gambar 12**.

Kode program 5. Kode program untuk menampilkan data user

```

public function index(Request $request)
{
    $cari = $request->query('cari');

    if (!empty($cari)) {
        $data = User::sortable()
            ->where('users.NIP', 'like', '%' .
            $cari . '%')
            ->orWhere('users.role', 'like', '%' .
            $cari . '%')
            ->orWhere('users.name', 'like',
            '%' . $cari . '%')
            ->paginate(5)->onEachSide(1);
    } else {
        $data = User::sortable()-
        >paginate(5)->onEachSide(1);
    }

    return view('admin.users-admin',
    compact('data'), [
        'title' => 'Users',
        'name' => 'Kelurahan Oetete',
        'desc' => 'User Data',
        'button' => 'Users'
    ])->with([
    ]
    )
    
```

```
'data' => $data,
'cari' => $cari
]);
}
```

Kode program 5 menunjukkan kode untuk menampilkan data user pada table. Baris 3-13 merupakan algoritma dari mesin pencari dan fitur sorting data. Baris 15-23 berfungsi untuk return data user yang sudah tersimpan di database ke halaman view.

Kode program 6. Kode program untuk menambah data user

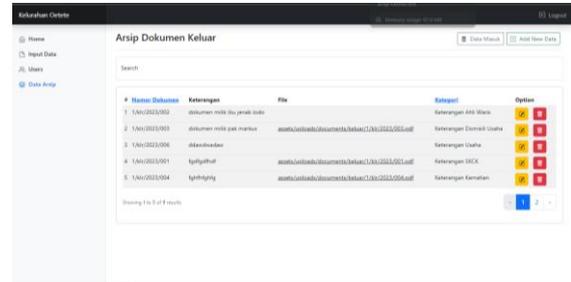
```
public function store(Request $request)
{
    $request->validate([
        'name' => 'required|max:255',
        'role' => 'required',
        'NIP' => 'required|unique:users',
        'password' => 'required|min:5|max:255',
        'remember_token'
    ]);

    $data = [
        'name' => Str::title($request->name),
        'role' => $request->role,
        'NIP' => $request->NIP,
        'password' => bcrypt($request->password),
        'remember_token' => Str::random(60)
    ];

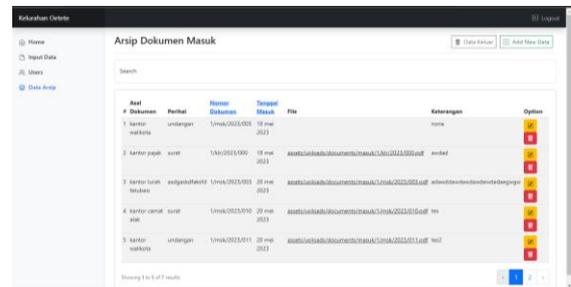
    if (User::create($data)) {
        return redirect('/users');
    }
}
```

Kode program 6 menampilkan kode untuk menambahkan data baru ke database. \$data menyimpan data yang telah di input oleh admin. Kemudian model User melakukan create data yang disimpan

didalam variabel \$data dan menyimpannya kedalam database.



Gambar 13. Halaman data arsip keluar



Gambar 14. Halaman data arsip masuk

Data yang telah diarsipkan akan disimpan kedalam database. Untuk mengakses data tersebut dapat dilakukan melalui menu arsip yang mana berfungsi menampilkan data yang sudah tersimpan didalam database seperti yang dapat dilihat pada **Gambar 13** dan **Gambar 14**.

Kode program 7. Kode program untuk menampilkan data arsip keluar

```
public function datas(Request $request)
{
    $cari = $request->query('cari');

    if (!empty($cari)) {
        $data = DataModel::sortable()
        -
        >where('data_models.nomor_dokumen',
        'like', '%' . $cari . '%')
    }
}
```

```
-
<orWhere('data_models.kategori', 'like', '%'
. $cari . '%')
->paginate(5)->onEachSide(1);
} else {
    $data = DataModel::sortable()-
>paginate(5)->onEachSide(1);
}

return view('admin.datas-admin',
compact('data'), [
    'title' => 'Data Arsip',
    'name' => 'Kelurahan Oetete',
    'desc' => 'Arsip Dokumen Keluar',
    'button' => 'Data Arsip'
])->with([
    'data' => $data,
    'cari' => $cari
]);
}
```

Kode program 7 menunjukkan kode untuk menampilkan data arsip keluar pada table seperti yang dapat dilihat pada Gambar 13. Baris 3-12 merupakan algoritma dari mesin pencari dan fitur sorting data. Baris 14-22 berfungsi untuk return data arsip keluar yang sudah tersimpan di database ke halaman view.

Kode program 8. Kode program untuk menampilkan data arsip masuk

```
public function show(data_masuk
$data_masuk, Request $request)
{
    $cari = $request->query('cari');

    if (!empty($cari)) {
        $data_masuk =
data_masuk::sortable()
-
>where('data_masuks.nomor_dokumen',
'like', '%' . $cari . '%')
```

```
-
<orWhere('data_masuks.tgl_masuk', 'like',
'%' . $cari . '%')
->paginate(5)->onEachSide(1);
} else {
    $data_masuk =
data_masuk::sortable()->paginate(5)-
>onEachSide(1);
}

return view('admin.datas-admin-
masuk', compact('data_masuk'), [
    'title' => 'Data Arsip',
    'name' => 'Kelurahan Oetete',
    'desc' => 'Arsip Dokumen Masuk',
    'button' => 'Data Arsip'
])->with([
    'data' => $data_masuk,
    'cari' => $cari
]);
}
```

Kode program 8 menunjukkan kode untuk menampilkan data arsip masuk pada table seperti yang dapat dilihat pada Gambar 14. Baris 3-12 merupakan algoritma dari mesin pencari dan fitur sorting data. Baris 14-22 berfungsi untuk return data arsip masuk yang sudah tersimpan di database ke halaman view.

4. KESIMPULAN

Berdasarkan latar belakang dan rumusan masalah Sistem Pengarsipan Data Berbasis Web Menggunakan framework Laravel, dapat diambil kesimpulan bahwa dengan adanya sistem pengarsipan data ini, diharapkan dapat membantu masalah pengarsipan data pada Kelurahan Oetete. Saran dari penulis untuk lebih memperhatikan proses analisis kebutuhan

sehingga tidak terjadi banyak perubahan pada sistem yang sedang dibuat. Juga diharapkan adanya penelitian lebih lanjut untuk pengembangan sistem ini kedepannya.

DAFTAR PUSTAKA

- [1] K. Bagi *et al.*, “Kontribusi Pengelolaan Arsip Dalam Proses Pengambilan,” *J. Ilmu Perpust.*, vol. 9, no. 1, p. 25, 2020.
- [2] N. Firdaus and D. Irfan, “Rancang Bangun Sistem Informasi Arsip Berbasis Web Menggunakan Framework Codeigniter,” *Voteteknika (Vocational Tek. Elektron. dan Inform.)*, vol. 8, no. 1, p. 44, 2020, doi: 10.24036/voteteknika.v8i1.107759.
- [3] I. P. Sari, I. H. Batubara, and P. P. Hariani, “Perancangan Sistem Informasi Pengelolaan Arsip Digital Berbasis Web untuk Mengatur Sistem Kearsipan di SMK Tri Karya,” 2022.
- [4] N. Putu and E. Merliana, “yang dapat diakses oleh siapa saja. Kesiapan penerapan arsip digital perlu diperhatikan untuk pengelolaannya agar arsip dapat tertata dengan baik. Adanya arsip digital yang diterapkan dalam Lembaga pendidikan akan mendukung proses pembelajaran dalam mempe,” no. 3, pp. 141–152, 2021.
- [5] B. A. E. Praditya, Y. Nataliani, and P. F. Tanaem, “Perancangan sistem informasi penyimpanan data koperasi berbasis web,” *Aiti*, vol. 18, no. 1, pp. 34–53, 2021, doi: 10.24246/aiti.v18i1.34-53.
- [6] D. Purnama Sari and R. Wijanarko, “Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus di Rumah Kamera Semarang),” *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 2, no. 1, p. 32, 2020, doi: 10.36499/jinrpl.v2i1.3190.
- [7] Lp2m.uma, “Metode Waterfall – Definisi dan Tahap-tahap Pelaksanaannya,” *Universitas Medan Area*, 2022. <https://lp2m.uma.ac.id/2022/06/07/metode-waterfall-definisi-dan-tahap-tahap-pelaksanaannya/>
- [8] D. Aipina and H. Witriyono, “PEMANFAATAN FRAMEWORK LARAVEL DAN FRAMEWORK BOOTSTRAP PADA PEMBANGUNAN APLIKASI,” vol. 18, no. 1, pp. 36–42, 2022.
- [9] E. B. Kristianto *et al.*, “Perancangan Website Dinas Pemberdayaan Masyarakat dan Pemerintahan Kampung (DPMPK) Pegunungan Bintang Menggunakan Framework Laravel,” vol. 18, no. 2, pp. 139–157, 2021.
- [10] Meilinaeka, “Class Diagram adalah, Temukan Pengertiannya di Sini,” *telkom university*, 2023. <https://it.telkomuniversity.ac.id/class-diagram-adalah/>
- [11] F. Natacia and E. Mailoa, “Perancangan Aplikasi Jaringan Dokumentasi dan Informasi Hukum Berbasis Web Menggunakan Framework Laravel,” vol. 6, pp. 1616–1628, 2022.