

## Perancangan Sistem Pengarsipan Surat dan Penugasan Berbasis *Web* Menggunakan *Framework* Laravel

Yuven Prisma Dyarenggasto <sup>1\*</sup>, Nina Setiyawati <sup>2</sup>

<sup>1,2</sup> Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.

*Email:* 672019041@student.uksw.edu <sup>1\*</sup>, nina.setiyawati@uksw.edu <sup>2</sup>

### Histori Artikel:

*Dikirim* 6 November 2023; *Diterima dalam bentuk revisi* 25 November 2023; *Diterima* 10 Desember 2023; *Diterbitkan* 10 Januari 2024. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Indonesia Banda Aceh.

### Abstrak

Palang Merah Indonesia yang selanjutnya disingkat PMI adalah sebuah organisasi perhimpunan nasional di Indonesia yang bergerak ada bidang sosial dan kemanusiaan. Sistem pengarsipan surat yang digunakan masih dilakukan secara manual dengan cara pencatatan pada suatu buku, sedangkan untuk sistem penugasan sendiri sudah dilakukan secara online melalui web Palang Merah Indonesia sendiri. Dengan menggunakan metode lama tersebut, dinilai kurang efektif karena akan memakan waktu yang banyak serta mudahnya kehilangan data. Oleh karena itu, diperlukan sebuah sistem informasi berbasis web yang dirasa mempermudah pegawai dalam melakukan pengarsipan surat. Sehingga dalam penelitian ini dibangun dengan menggunakan framework Laravel. Framework Laravel digunakan karena dinilai mudah dipahami serta memiliki fitur-fitur yang memudahkan developer dalam pembangunan aplikasi tersebut. Penelitian ini bertujuan untuk mempermudah pegawai dalam melakukan pengarsipan surat, penugasan serta pelaporan dari selesainya penugasan pada Palang Merah Indonesia Kota Salatiga. Hasil penelitian menunjukkan bahwa aplikasi ini dapat digunakan untuk meningkatkan kinerja pegawai.

**Kata Kunci:** Laravel Website; Pengarsipan Surat; Penugasan; Pelaporan.

### Abstract

The Indonesian Red Cross, hereinafter abbreviated as PMI, is a national association organization in Indonesia that is active in the social and humanitarian fields. The letter filing system used is still done manually by recording it in a book, while the assignment system itself has been done online via the Indonesian Red Cross's own website. Using the old method, it is considered less effective because it will take a lot of time and it is easy to lose data. Therefore, we need a web-based information system that is felt to make it easier for employees to do letter filing. So in this study, it was built using the Laravel framework. The Laravel framework is used because it is considered easy to understand and has features that make it easier for developers to build the application. This study aims to facilitate employees in filing letters, assignments, and reporting the completion of assignments at the Indonesian Red Cross in Salatiga City. The research results show that this application can improve employee performance.

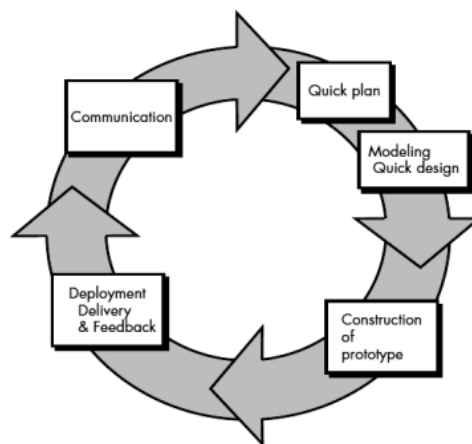
**Keyword:** Laravel; Website; Mail Archiving; Assignment; Reporting.

## 1. Pendahuluan

Surat merupakan salah satu sarana komunikasi tertulis yang digunakan untuk menyampaikan informasi bukti konkrit akan suatu kejadian [1]. Surat terbagi menjadi dua, yaitu surat masuk dan surat keluar. Surat masuk merupakan surat yang diterima suatu instansi dari luar instansi yang bertujuan memberikan informasi tertentu. Sedangkan surat keluar merupakan surat yang dibuat oleh suatu instansi untuk dikirimkan kepada instansi-instansi lainnya [2]. Arsip merupakan kumpulan dari semua surat-surat yang dibuat dan diterima dimana arsip ini terjadi karena didalamnya terdapat beberapa aksi seperti mencatat, menyimpan, serta juga mengolah dari surat-surat yang ada. Pengarsipan dapat dikatakan baik ketika dalam waktu tertentu, surat yang dibutuhkan dapat ditemukan dengan sangat mudah [3]. Dalam penelitian ini kasus yang diangkat adalah pada hal administrasi sistem pengarsipan surat masih dilakukan secara manual yaitu dengan menuliskan surat masuk dan keluar pada buku arsip.

Proses manual tersebut dinilai riskan akan terjadinya kehilangan data. Sedangkan untuk melakukan penugasan dan pelaporan sudah dilakukan secara online namun pada aplikasi Sistem Informasi Administrasi dan Manajemen Operasi (SIAMO). Karena berada pada aplikasi yang berbeda, penugasan dan pelaporan dinilai kurang efisien dan membutuhkan satu aplikasi yang dapat mencakup dari pengarsipan, penugasan, dan pelaporan. Berdasarkan latar belakang yang ada, penelitian ini dilakukan dengan tujuan untuk merancang sistem pengarsipan surat dan penugasan berbasis *web* menggunakan framework Laravel. Penggunaan *framework* Laravel dinilai mempermudah dalam pengembangan karena hanya perlu membuat satu master *template blade* saja untuk menggunakan di semua halaman atau fitur yang dibangun [4]. Harapan dibangunnya sistem ini bertujuan agar pengelolaan surat dan penugasan dapat dilakukan dengan lebih efisien dan efektif. Perancangan aplikasi ini dilakukan untuk mempermudah staf sumber daya manusia dalam melakukan pengarsipan surat dan penugasan dalam satu *web* yang sama.

## 2. Metode Penelitian



Gambar 1. Metode *Prototype* (Roger S. Pressman, 2012)

Pada penelitian ini menggunakan metode *prototype*. Metode *prototype* adalah suatu proses yang menjaankan komunikasi antara *developer* yang adalah peneliti dan *client* yang dalam penelitian ini adalah staf bidang sumber daya manusia PMI Kota Salatiga. Bukan menyajikan bentuk asli dari sistem namun metode ini berperan penting untuk menyajikan gambaran global dari sistem yang akurat kepada *client* [5][6][7]. Dalam metode ini baik *developer* maupun *client* diuntungkan karena dalam pembuatan aplikasi, komunikasi berlangsung secara terus menerus hingga pembuatan aplikasi selesai dibuat sesuai dengan kebutuhan daripada *client* sendiri, dan apabila dalam pembuatan aplikasi ada ketidaksesuaian harapan

dari *client*, pihak *developer* dapat memperbaiki sesuai dengan permintaan *client* dengan baik. Metode ini memiliki beberapa tahapan yang memiliki fungsi sendiri-sendiri dari tiap tahapannya. Fungsi dari tahapan-tahapan tersebut ada pada penjelasan berikut:

1) *Communication*

Merupakan tahap awal dalam memulai metode *prototype*. *Communication* yang dimaksud adalah untuk mengidentifikasi permasalahan sebelumnya serta mendapatkan informasi penting lainnya untuk pembuatan sistem. Tahap ini dilakukan secara wawancara kepada staf bidang sumber daya manusia PMI Kota Salatiga. Dalam wawancara tersebut mencangkup tujuan dari pembuatan aplikasi secara keseluruhan.

2) *Quick Plan*

Merupakan tahapan untuk menentukan pengguna yang akan terlibat dalam penggunaan sistem, serta melakukan spesifikasi untuk menentukan kebutuhan dari pengembangan sistem agar dapat sesuai dengan yang diharapkan. Kegiatan yang dilakukan pada tahap ini adalah perancangan awal bentuk dari tampilan, perangkat lunak, serta *layout* yang akan di bangun nantinya.

3) *Modelling Quick Design*

Melakukan tahap prancangan *usecase diagram*, *activity diagram*, *class diagram* serta *database* pada sistem sesuai analisis yang sudah dilakukan sebelumnya. Perancangan ini dilakukan dengan efektif mungkin agar kebutuhan sesuai dengan yang diharapkan.

4) *Construction of Prototype*

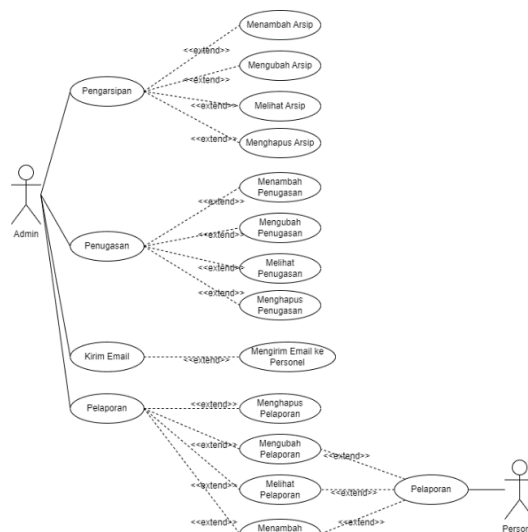
Selanjutnya pembangunan sistem akan dimulai pada tahapan ini. Pembangunan sistem ini dilakukan sesuai dengan tujuan umum dan rancangan aplikasi serta aspek-aspek data yang telah dikumpulkan pada tahapan sebelum-sebelumnya. Dengan demikian, diharapkan penyusunan sistem dapat segera terselesaikan dan akan lebih cepat mendapatkan *feedback* terkait sistem yang dibangun.

5) *Deployment Delivery & Feedback*

Tahap ini merupakan tahapan terakhir dalam metode *prototype*. Sistem yang telah dibangun, akan diserahkan kepada *client* untuk mendapatkan *feedback* dari hasil pembangunan sistem. Dimana *feedback* tersebut akan digunakan untuk acuan perbaikan dari sistem yang kurang sesuai dengan kebutuhan. Dengan dilakukannya evaluasi oleh *client*, tahap *communication* kembali terjadi yang dilanjutkan oleh tahapan-tahapan lainnya hingga *client* puas terhadap aplikasi yang dibangun.

### 3. Hasil dan Pembahasan

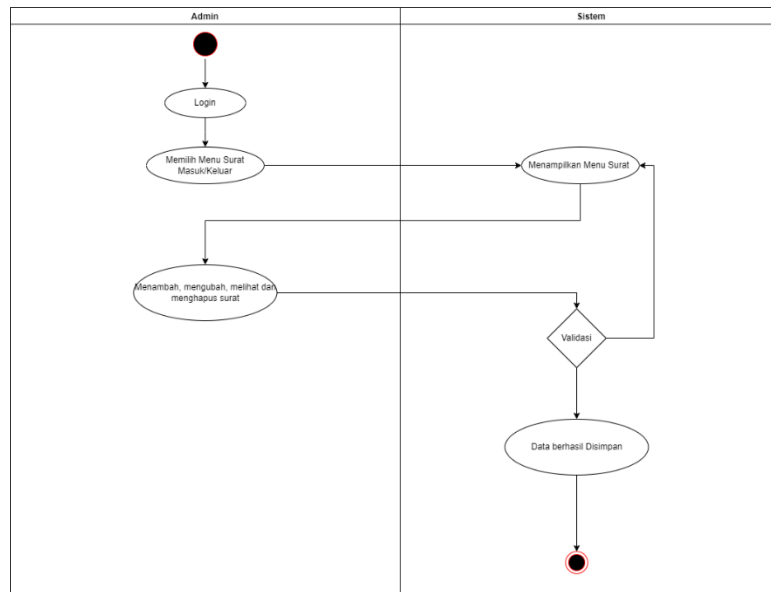
#### 3.1 Use Case Diagram



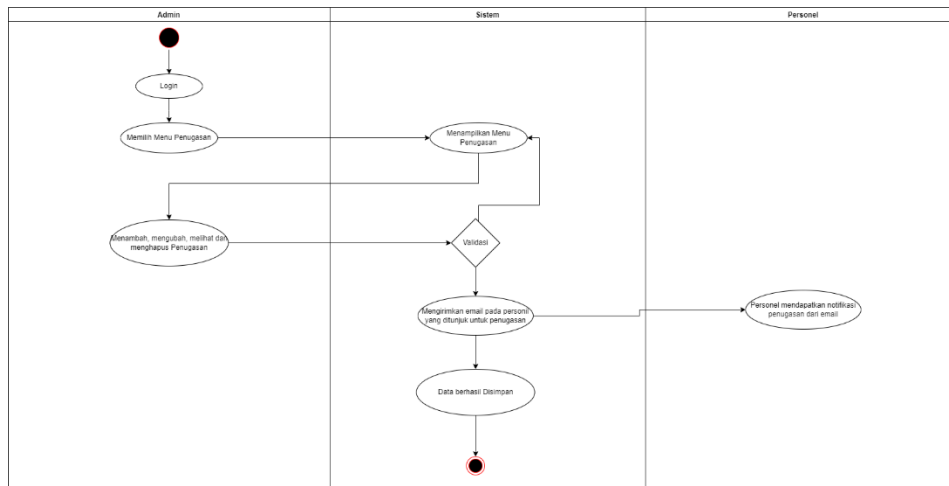
Gambar 2. Use Case Diagram

Use Case Diagram merupakan uraian dari hubungan aktor dengan sistem[8]. Diagram ini mampu digunakan untuk mengetahui apa saja fungsi dan siapa saja yang dapat menggunakan fungsi tersebut[9]. Pada use case ini, terdapat dua aktor yaitu *admin* dan *personel*. *Admin* merupakan staf pada bidang Sumber Daya Manusia (SDM) yang dapat mengakses semua fitur yang ada. Sedangkan untuk *personel* adalah orang yang ditugaskan dalam penugasan serta hanya dapat mengakses halaman pelaporan.

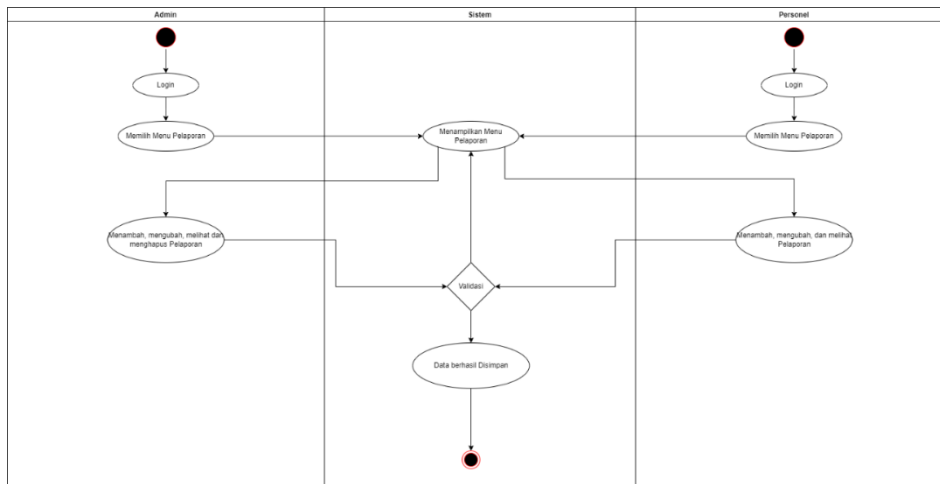
### 3.2 Activity Diagram



Gambar 3. Activity Diagram Pengarsipan Surat



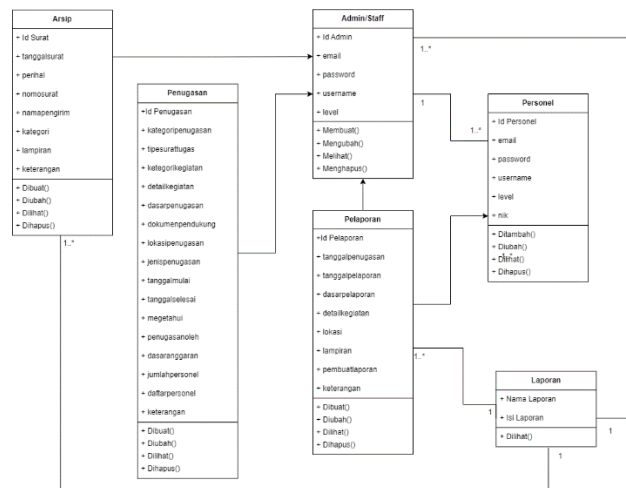
Gambar 4. Activity Diagram Penugasan



Gambar 5. Activity Diagram Pelaporan

Activity Diagram merupakan alur kerja sistem yang digambarkan secara visual untuk mengetahui yang terjadi dalam aplikasi ini. Diagram ini mirip dengan flowchart karena baik activity diagram maupun flowchart sendiri merupakan wujud model dari satu aktifitas ke aktifitas lainnya, perbedaannya ialah flowchart untuk menggambarkan alur kerja sistem sedangkan activity diagram untuk menggambarkan aktivitas yang dapat dilakukan oleh aktor [8][10]. Dalam aplikasi ini, Gambar 3 merupakan activity diagram untuk menu pengarsipan baik surat masuk maupun keluar yang dapat dilakukan oleh admin. Pada Gambar 4 merupakan activity diagram untuk fitur penugasan yang hanya bisa dilakukan oleh admin. Sedangkan pada Gambar 5 adalah fitur pelaporan yang dapat dilakukan oleh admin dan personel. Pada fitur ini yang membedakan adalah pada menu action menghapus hanya bisa dilakukan oleh admin saja, dan pada fitur input, view, dan update bisa dilakukan baik admin maupun personel.

### 3.3 Class Diagram



Gambar 6. Class Diagram

Class diagram merupakan gambaran struktur dari pendefinisian semua kelas, selain itu juga merupakan inti pengembangan dan desain dari sistem yang dibangun [11][12] serta membantu untuk memvisualisasikan detail struktur antar kelas dimana pada setiap kelas itu sendiri memiliki 3 komponen penting yaitu attribute, operation, dan name[8]. Gambar 6 merupakan class diagram pada aplikasi ini. Total class pada pembangunan aplikasi ini ada 6 class. Dalam pembangunan aplikasi ini Admin melakukan kelola data dengan menambahkan data-data yang ada pada atribut class arsip,

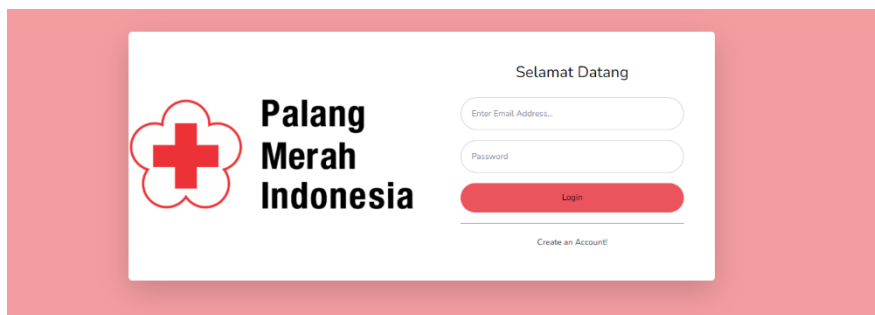
penugasan, pelaporan, dan personel. Isi *class* tersebut dapat ditambah, diubah, dilihat, dan dihapus. Personel melakukan proses penambahan data pelaporan dengan menambahkan atribut-atribut yang ada di *class* pelaporan, isi *class* tersebut dapat ditambah, diubah, dilihat, namun tidak dapat menghapus data yang telah ditambahkan. Pelaporan hanya dapat dihapus oleh admin.

### 3.4 Framework Laravel

Dalam perancangan aplikasi, *template blade* yang dimiliki oleh Laravel sangat membantu proses perancangan aplikasi ini. *Template blade* ini terdiri dari satu file yang berisi *master* dari HTML dan file lain yang berisi halaman yang akan ditampilkan. Di dalam *template* ini hanya membutuhkan satu *master template* yang dapat digunakan di lebih dari satu halaman[4]. Dalam menjalankan *framework* Laravel, terdapat *composer* yang dapat digunakan untuk menginstall *library*, salah satunya Mail. Laravel Mail memudahkan untuk mengirimkan email dalam jumlah besar dengan cara yang efisien. Dalam penggunaannya, Laravel Mail dapat diakses melalui API atau melalui fungsi-fungsi bawaan seperti `Mail::to()` dan `Mail::send()`. Contoh penggunaannya dapat dilihat pada Kode Program 1.

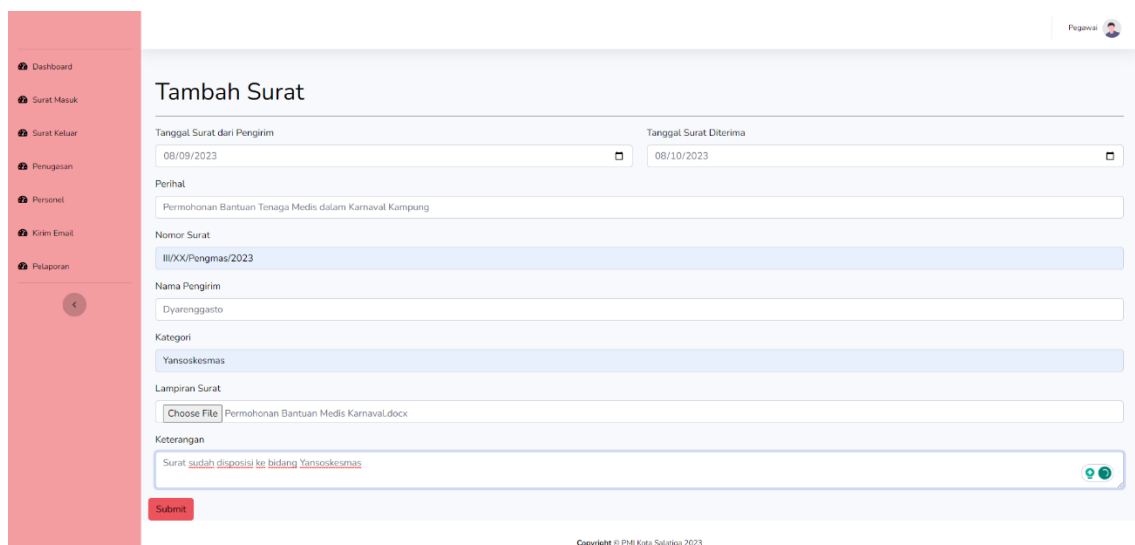
### 3.5 Hasil Penelitian

Pada tahap selanjutnya adalah hasil dari perancangan sistem pengarsipan surat dan penugasan berbasis *website* menggunakan *framework* Laravel yang akan ditunjukkan pada Gambar 7.



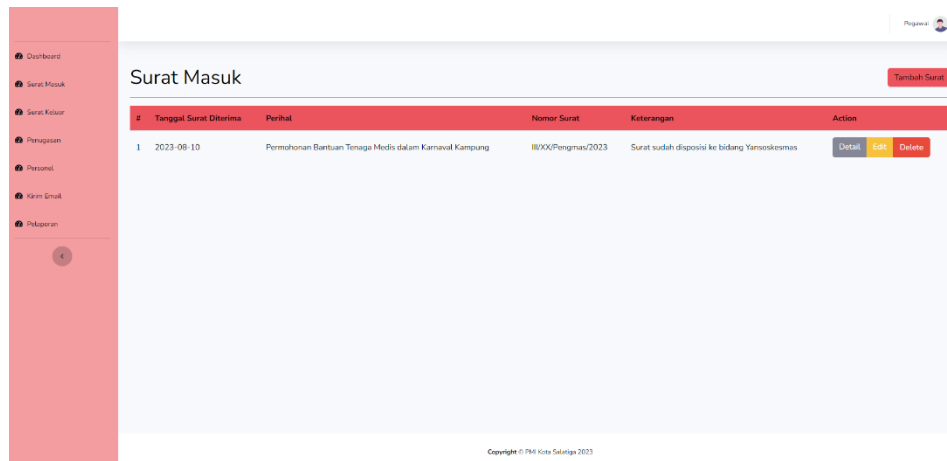
Gambar 7. Halaman Login

Pada gambar 7 merupakan tampilan login yang memiliki form *username* dan *password*. Untuk *login* ke sistem, baik admin maupun personel wajib mengisi *username* dan *password* sesuai dengan yang di daftarkan pada saat melakukan pendaftaran.



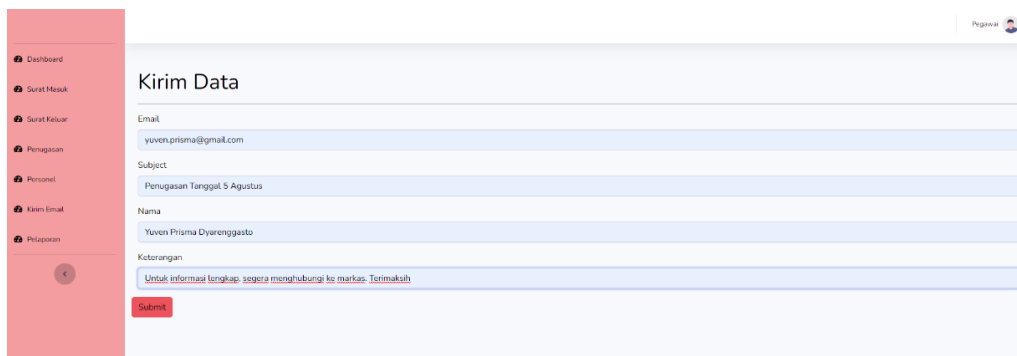
Gambar 8. Tampilan Form Tambah Surat Masuk

Gambar 8 merupakan tampilan ketika pegawai akan melakukan *input* surat masuk. Pada halaman ini, pegawai juga dapat menambahkan file pendukung terkait surat permohonan dari pihak pemohon. Apabila pegawai sudah menekan tombol *submit*, maka secara otomatis, sistem akan menampilkan halaman data surat seperti yang dapat dilihat pada Gambar 9.



Gambar 9. Tampilan Data Surat Masuk

Pada Gambar 9 diatas, terlihat bahwa surat telah berhasil ditambahkan. Selanjutnya pada kolom *action*, *admin* dapat melakukan aksi *view*, *edit*, dan *delete* apabila diperlukan. Fitur yang ada di gambar 8 dan 9 sama dengan fitur pada menu surat keluar, penugasan, dan pelaporan. Pada tampilan ini dapat dijelaskan bahwa mengacu pada tujuan utama pembuatan aplikasi ini telah selesai. Dimana pengarsipan yang dilakukan secara manual telah diselesaikan dengan pengarsipan secara *online* ini. Data yang sebelumnya masih disimpan secara manual dan beresiko hilang, sekarang *admin* tidak perlu khawatir lagi karena data yang tersimpan akan lebih aman.



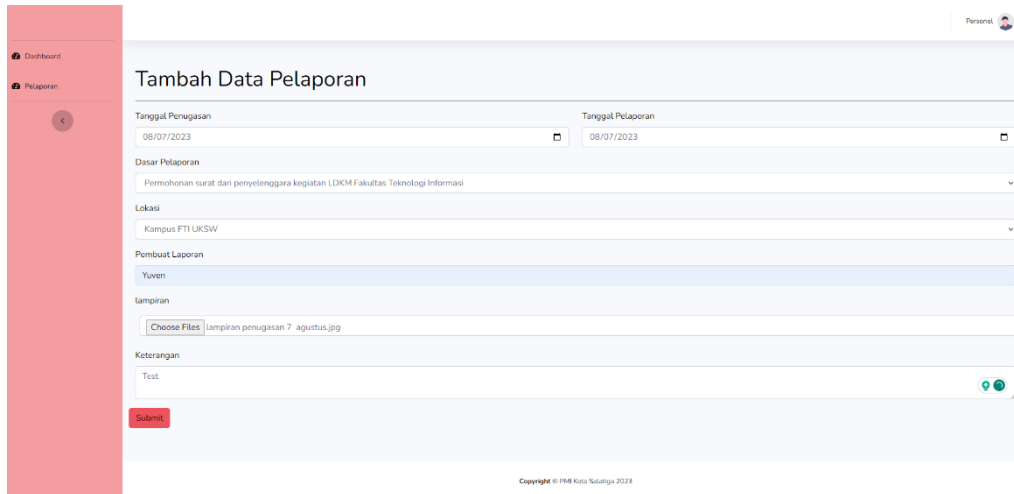
Gambar 10. Menu Kirim Email

Gambar 10 adalah tampilan form untuk pegawai saat mengirimkan email kepada personel yang bertugas. Fitur ini hanya dapat dikirimkan oleh pegawai saja dan tidak berlaku sebaliknya. Selanjutnya setelah pegawai mengirimkan email kepada personel, personel akan menerima email mereka.



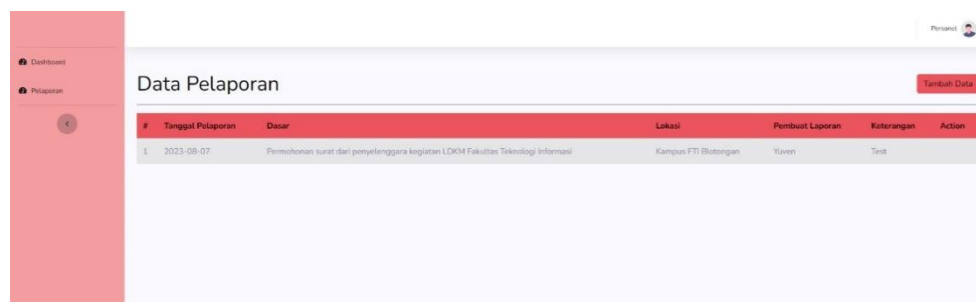
Gambar 11. Tampilan Email yang diterima Personel

Gambar 11 menunjukkan bahwa email yang dikirimkan pegawai telah diterima oleh personel. Selanjutnya apabila personel telah menjalankan tugas yang diberikan, personel akan melakukan penambahan pelaporan.



Gambar 12. Tampilan Menu Pelaporan

Pada gambar 12 menampilkan *form* pelaporan yang dapat diisi oleh personel dan juga pegawai. Apabila sudah selesai melakukan pelaporan, data akan muncul halaman data pelaporan Gambar 13.



#	Tanggal Pelaporan	Dasar	Lokasi	Pembuat Laporan	Keterangan	Action
1	2023-08-07	Permohonan surat dari penyelenggara kegiatan LDKM Fakultas Teknologi Informasi	Kampus FTI Blottangan	Yuveni	Test	

Gambar 13. Tampilan Data Pelaporan

Gambar 13 merupakan tampilan ketika pelapor telah melakukan pelaporan. Tampilan ini sama dengan tampilan yang ada pada pegawai, namun yang membedakan yaitu pada pegawai terdapat fitur *edit*, *view*, dan *delete* sedangkan personel hanya terdapat fitur *edit*, dan *view*. Adapun Kode Program *Mengirim email* sebagai berikut:

```
$request->validate ([
    'email' => 'required|email',
    'subject' => 'required',
    'nama' => 'required',
    'keterangan' => 'required'
]);

$data = [
    'email' => $request->email,
    'subject' => $request->subject,
    'nama' => $request->nama,
    'keterangan' => $request->keterangan
];
Mail::to($data['email'])->send(new MailNotify($data));
```

Pada Kode Program diatas merupakan *Controller* untuk mengirim surat. Dapat dilihat pada baris ke-15 bahwa dengan menambahkan library “Mail::to(\$data['email']->send(new MailNotify(\$data));” dimana pada *syntax* tersebut email akan otomatis tertuju sesuai dengan yang telah ditambahkan pada form kirim email pada Gambar 11. Adapun Kode Program *Menampilkan Data Surat Masuk* sebagai berikut:

```
<div class="row">
  <div class="col mb-3">
    <label class="form-label">Tanggal Surat Diterima</label>
    <input type="datetime" name="tanggalsuratditerima" class="form-control"
placeholder="Tanggal Surat Diterima" value="{{ $suratmasuk->
tanggalsuratditerima }}" readonly>
  </div>
  <div class="col mb-3">
    <label class="form-label">Tanggal Surat dari Pengirim</label>
    <input type="datetime" name="tanggalsuratpengirim" class="form-control"
placeholder="Tanggal Surat dari Pengirim" value="{{ $suratmasuk->
tanggalsuratpengirim }}" readonly>
  </div>
</div>
<div class="row">
  <div class="col mb-3">
    <label class="form-label">Tanggal Surat Diterima</label>
    <input type="datetime" name="tanggalsuratditerima" class="form-control"
placeholder="Tanggal Surat Diterima" value="{{ $suratmasuk->
tanggalsuratditerima }}" readonly>
  </div>
  <div class="col mb-3">
    <label class="form-label">Tanggal Surat dari Pengirim</label>
    <input type="datetime" name="tanggalsuratpengirim" class="form-control"
placeholder="Tanggal Surat dari Pengirim" value="{{ $suratmasuk->
tanggalsuratpengirim }}" readonly>
  </div>
</div>
```

Kode Program diatas merupakan potongan kode program untuk menampilkan data surat masuk. Dalam menampilkan data surat masuk, data akan diambil dari *database* yang ada sesuai dengan data yang ditambah saat menambahkan data surat masuk. Data yang diambil dari *database* nantinya akan muncul pada tabel yang sudah disiapkan.

Tabel 1. *Black Box Testing*

Fungsi	Kondisi	Output yang diharapkan	Output yang dihasilkan	Status pengujian
Login	Username dan Password benar	Login berhasil	Login berhasil	Valid
	Username dan Password salah	Login gagal	Login gagal	Valid
Menambah data surat masuk	Mengisi semua form	Data berhasil ditambah	Data berhasil ditambah	Valid
	Tidak memasukkan lampiran surat	Data gagal ditambahkan	Data gagal ditambahkan	Valid
	Mengisi semua form	Data berhasil ditambah	Data berhasil ditambah	Valid

---

Menambah data surat keluar	Tidak memasukkan lampiran surat	Data gagal ditambahkan	Data gagal ditambahkan	Valid
Menambah data penugasan	Mengisi semua form	Data berhasil ditambah	Data berhasil ditambah	Valid
	Tidak memasukkan lampiran surat	Data gagal ditambahkan	Data gagal ditambahkan	Valid
Menambah data personel	Mengisi semua form	Data berhasil ditambah	Data berhasil ditambah	Valid
	Terdapat form yang tidak diisi	Data gagal ditambahkan	Data gagal ditambahkan	Valid
Menambahkan data pelaporan	Mengisi semua form	Data berhasil ditambah	Data berhasil ditambah	Valid
	Terdapat form yang tidak diisi	Data gagal ditambahkan	Data gagal ditambahkan	Valid
Mengirim email	Email valid	Email berhasil dikirim	Email berhasil dikirim	Valid
	Email invalid	Email gagal dikirim	Email gagal dikirim	Valid
Menampilkan data	Menampilkan data surat masuk, surat keluar, pelaporan, dan personel yang sudah ditambahkan	Data berhasil ditampilkan	Data berhasil ditampilkan	Valid
Mengubah data	Mengubah data surat masuk, surat keluar, pelaporan, dan personel yang sudah ditambahkan	Data berhasil diubah	Data berhasil diubah	Valid
Menghapus data	Menghapus data surat masuk, surat keluar, pelaporan, dan personel yang sudah ditambahkan	Data berhasil dihapus	Data berhasil dihapus	Valid

---

*Black Box Testing* merupakan suatu pengujian terhadap sistem yang berguna untuk mencari kesalahan fungsi dari sistem yang dibangun. Pengujian sangat penting dilakukan karena akan dapat memastikan apakah kualitas dari perangkat lunak maupun sistem yang dibangun dapat berjalan dengan baik atau tidaknya[13]. Dalam pengujian ini, data yang diambil adalah secara acak, dengan tujuan untuk mendapatkan hasil yang diharapkan. Sebagai contoh, apabila ketika menambahkan data dan tidak dapat ditambahkan kedalam *database* maka dianggap salah, namun apabila data dapat ditambahkan kedalam *database* maka dianggap benar [14].

Tabel 2. *User Acceptance Test*

Pernyataan	STS	STS	N	S	SS	Hasil Perhitungan Skala Likert	Interpretasi Skala Likert
1 Aplikasi mudah digunakan baik staf maupun personel				7	3	86%	Sangat Puas
2 Menu Pengarsipan Surat dinilai sangat membantu meringankan pekerjaan admin staf SDM			1	6	3	84%	Sangat Puas
3 Menu Penugasan mempermudah staf dalam memberikan penugasan personel				7	3	86%	Sangat Puas
4 Menu Pelaporan mempermudah personel untuk membuat laporan setelah melaksanakan tugas				6	4	88%	Sangat Puas
5 Data yang ditampilkan tersaji dengan benar dan sesuai				6	4	88%	Sangat Puas

*User Acceptance Test* adalah suatu pengujian langsung oleh pengguna yang berinteraksi langsung dengan sistem untuk mengetahui apakah fungsi telah dari sistem telah berjalan sesuai kebutuhan[15]. Pengujian *User Acceptance Test* memiliki tujuan memastikan solusi yang dibangun dari permasalahan awal dan diaplikasikan kedalam aplikasi ini dapat menyelesaikan permasalahan *client*[16]. Pengujian ini dilakukan dengan menyebarkan 10 kuisioner ke kantor Palang Merah Indonesia Kota Salatiga yang diantaranya diisi oleh 5 orang pegawai dan 5 orang relawan. Setelah mendapatkan jawaban dari para *responden*, penghitungan dilakukan dengan menggunakan skala likert. Table 2 menjelaskan bahwa solusi yang telah dibangun melalui pembangunan *web* ini mendapatkan nilai sangat puas di semua pernyataan.

#### 4. Kesimpulan

Berdasarkan penelitian yang sudah dilakukan, sistem pengarsipan surat dan penugasan dapat mempermudah pegawai dalam melakukan pengarsipan surat dan penugasan personel, serta personel yang mendapat penugasan dapat melakukan pelaporan hasil penugasan mereka dengan mudah. Selain itu, sistem ini dapat mengurangi resiko akan hilangnya data surat maupun pelaporan yang telah dibuat. Tabel 2 juga telah menjelaskan bahwa interpretasi dari hasil skala likert menunjukkan kepuasan dari pembangunan aplikasi ini. Terbukti bahwa interpretasi dari 5 pernyataan mendapatkan hasil sangat puas yang dimana pembangunan aplikasi ini dapat dinyatakan berhasil. Pembangunan aplikasi dilakukan menggunakan *framework* Laravel yang dipermudah dengan penggunaan *template blade* untuk membuat tampilan dan *package* Mail untuk pengiriman email dari menu penugasan. Untuk penelitian ke depan, aplikasi bisa dikembangkan dalam *platform mobile* untuk meningkatkan kemudahan dari penggunaan aplikasi ini.

#### 5. Daftar Pustaka

- [1] Hendriyani, M., Sugiyono, S., & Riyanto, A. (2023). PROSEDUR PENANGANAN SURAT MASUK DAN SURAT KELUAR PADA SUBBAGIAN TATA USAHA DI PUSAT JASA KEARSIPAN ARSIP NASIONAL REPUBLIK INDONESIA. *KOMPLEKSITAS: JURNAL ILMIAH MANAJEMEN, ORGANISASI DAN BISNIS*, 12(1), 8-14. DOI: <https://doi.org/10.56486/kompleksitas.vol12no1.311>.

- [2] Suherman, Y. (2017). Sistem informasi kearsipan tata kelola surat pada kantor inspeksi BRI kota Padang. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 1(1), 26-33. DOI: <https://doi.org/10.29207/resti.v1i1.7>.
- [3] Simanjuntak, N. J., Suryadi, S., & Silaen, G. J. (2017). Sistem Pengarsipan Surat Bagian Organisasi Dan Tatalaksana Pada Kantor Bupati Labuhanbatu Berbasis Web. *Informatika*, 5(3), 26-36. DOI: <https://doi.org/10.36987/informatika.v5i3.733>.
- [4] Setiyawati, N., & Kesowo, S. T. B. (2017). Pembangunan Aplikasi Pelaporan Kecelakaan Lalu Lintas Berbasis Web Menggunakan Framework Laravel. *Jurnal Sistem Informasi Indonesia*, 2(1).
- [5] Ichwani, A., Anwar, N., Karsono, K., & Alrifqi, M. (2021). Sistem Informasi Penjualan Berbasis Website dengan Pendekatan Metode Prototype. *Prosiding Sisfotek*, 5(1), 1-6.
- [6] Pradipta, A. A., Prasetyo, Y. A., & Ambarsari, N. (2015). Pengembangan Web E-Commerce Bojana Sari Menggunakan Metode Prototype. *eProceedings of Engineering*, 2(1).
- [7] Fenando, F. (2020). Implementasi E-Commerce Berbasis Web pada Toko Denia Donuts Menggunakan Metode Prototype. *JUSIFO (Jurnal Sistem Informasi)*, 6(2), 66-77. DOI <https://doi.org/10.19109/jusifo.v6i2.6532>.
- [8] Yulianti, M. (2023). SISTEM INFORMASI PENDAFTARAN PESERTA DIDIK BARU (PPDB) SMK IPTEK TANGSEL BERBASIS WEB DENGAN METODE WATERFALL. *LOGIC: Jurnal Ilmu Komputer dan Pendidikan*, 1(3), 485-495.
- [9] R. Hafsa, E. Arib, dan N. Maulana, "PERANCANGAN SISTEM INFORMASI MANAJEMEN INVENTORI DAN PENJUALAN PADA PERUSAHAAN PT. INHUTANI V DOI: <https://doi.org/10.30656/prosisko.v10i2.7001>.
- [10] Trilaksono, R. J. (2022). Rancang Bangun Aplikasi Kasir Berbasis Offline Pada Primkop Kartika Viyata Virajati. *ULIL ALBAB: Jurnal Ilmiah Multidisiplin*, 1(4), 857-865.
- [11] Ikhlas, M. (2018). Analisa Dan Perancangan Sistem Informasi Transaksi Dan Persediaan Pada Toko Bangunan UD. Romi Padang Menggunakan Bahasa Pemrograman PHP Dan Database MySQL. *Jurnal KomTekInfo*, 5(2), 51-62. DOI: <https://doi.org/10.35134/komtekinfo.v5i2.24>.
- [12] Putra, F. D., Riyanto, J., & Zulfikar, A. F. (2020). Rancang Bangun Sistem Informasi Manajemen Aset pada Universitas Pamulang Berbasis WEB. *Journal of Engineering, Technology, and Applied Science (JETAS)*, 2(1), 32-50. DOI: <https://doi.org/10.36079/lamintang.jetas-0201.93>.
- [13] Pallas, D. K. (2021). Black box testing aplikasi point of sales post. *Kurawal-Jurnal Teknologi, Informasi dan Industri*, 4(1), 1-16.
- [14] Ichsanudin, M. N., Yusuf, M., & Suraya, S. (2022). PENGUJIAN FUNGSIONAL PERANGKAT LUNAK SISTEM INFORMASI PERPUSTAKAAN DENGAN METODE BLACK BOX TESTING BAGI PEMULA. *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer*, 1(2), 1-8. DOI: <https://doi.org/10.55123/storage.v1i2.270>.



- [15] Wahyudi, I., Fahrullah, F., Alameka, F., & Haerullah, H. (2023). ANALISIS BLACKBOX TESTING DAN USER ACCEPTANCE TESTING TERHADAP SISTEM INFORMASI SOLUSIMEDSOSKU. *JURNAL TEKNO SAINS KODEPEN A*, 4(1), 1-9. DOI: <https://doi.org/10.54423/jtk.v4i1.54>.
- [16] Bangkalang, D. H. (2019). Pembangunan Aplikasi Monitoring Kinerja Surveyor Produk Kredit Bank XYZ Berdasarkan Tahapan Engineering Development. *CESS (Journal of Computer Engineering, System and Science)*, 4(2), 258-265. DOI: <https://doi.org/10.24114/cess.v4i2.13856>.