Edisi XLIV, Vol 9, No 2, Bulan Juni, Tahun 2024
p-ISSN : 1978-6832
e-ISSN : 2622-5948

PENGEMBANGAN APLIKASI *E-LOGBOOK* BERBASIS *SINGLE-PAGE APPLICATION* DI PERUM LPPNPI CABANG BANDA ACEH

Indra Fadillah Mukti
Politeknik Penerbangan Indonesia Curug
E-mail correspondence : indrafadillahmukti08@gmail.com

Abstrak

Logbook merupakan kegiatan mencatat, mengumpulkan data harian dan laporan dari kegiatan rutin yang terlaksana maupun belum terlaksana. *Logbook* dibagi menjadi dua jenis, yaitu *logbook* konvensional dan *logbook* digital. Di Perum LPPNPI Cabang Banda Aceh, penggunaan *logbook* sudah diterapkan sejak lama guna meningkatkan koordinasi antar pergantian *shift* dinas dan menjadi referensi jika terjadi suatu permasalahan yang sama terhadap peralatan di masa depan. Akan tetapi, teknisi di Perum LPPNPI Cabang Banda Aceh masih menggunakan *logbook* yang bersifat konvensional sehingga sangat beresiko mengalami kerusakan secara fisik, kehilangan dokumen, dan tidak sistematis. Menanggapi hal tersebut, penulis menginisiasi untuk melakukan digitisasi terhadap budaya yang ada melalui pengembangan aplikasi *e-logbook* menggunakan konsep *Single Page Application (SPA)*. Aplikasi ini dikembangkan menggunakan metode *Waterfall Model* dan dibangun menggunakan bahasa pemrograman *HTML*, *CSS*, *Javascript*, dan *PHP* serta menggunakan *framework Laravel*. Aplikasi ini memiliki tiga komponen utama, yaitu *Logbook* harian, *Dashboard* dan riwayat *logbook* serta dilengkapi dengan sistem autentikasi dan otorisasi. Berdasarkan hasil pengujian yang dilakukan, aplikasi ini mampu menjaga keutuhan data, memudahkan teknisi ketika mencari rekam kejadian, teknisi tidak perlu menulis *logbook* secara manual, mengurangi biaya penggunaan kertas dan seluruh data catatan di olah secara otomatis oleh sistem dengan tatanan yang terstruktur.

Kata Kunci: *E-Logbook, Waterfall, Laravel, SPA*

Abstract

A logbook is an activity that involves recording, collecting data, and reports from both completed and pending activities. Logbooks come in two forms: conventional and digital. At Perum LPPNPI Cabang Banda Aceh, logbooks are used to improve coordination among technicians and as a reference for similar issues. The currently used logbook is conventional, which poses the risk of physical damage and lacks systematic organization. In response, the author initiated digitalization by developing an e-logbook using the waterfall method, the Single Page Application (SPA) concept, programming languages such as HTML, CSS, Javascript, PHP, and the Laravel framework. There are three main components: Daily Logbook, Dashboard, and Logbook History, and equipped with authentication and authorization systems. Based on the results, this application can maintain data integrity, help technicians search for records, eliminate manual logbook entries, reduce paper usage, and automate data processing through a structured system.

Keywords*: Online Exam, Platform Monitoring, Web*

PENDAHULUAN

Logbook merupakan suatu kegiatan mencatat, mengumpulkan data harian di sebuah unit layanan dan laporan kegiatan rutin yang dilakukan setiap hari dari kegiatan yang telah terlaksana maupun kegiatan yang belum terlaksana (Febriani & Wahyuni, 2017). Pada umumnya, *logbook* dibagi menjadi dua jenis, yaitu *logbook* konvensional dan *logbook* digital. *Logbook* konvensional merupakan sebuah *logbook* yang penerapannya masih menggunakan kertas sebagai media tulisnya dan pengguna harus mencatat seluruh kegiatan secara manual (Jaya et al., 2023). *Logbook* digital merupakan sebuah *logbook* yang penerapannya memanfaatkan kemajuan teknologi seperti penggunaan sistem informasi berbasis *web* (Nugroho et al., 2016). *Logbook* digital diterapkan dalam bentuk aplikasi perangkat lunak yang dapat diakses melalui media seperti komputer, laptop, dan *smartphone* (Asfariza et al., 2022).

Menurut (McAlpine et al., 2017), 92,5% teknisi menyatakan bahwasannya penggunaan *logbook* sangat penting dan berguna bagi teknisi. Selain itu, melakukan pencatatan kegiatan pemeliharaan pencegahan dan perbaikan fasilitas telekomunikasi penerbangan merupakan suatu kewajiban bagi seorang teknisi telekomunikasi penerbangan (Direktorat Jenderal Perhubungan Udara, 2019). Di Perum LPPNPI Cabang Banda Aceh, penggunaan *logbook* sudah diterapkan sejak lama guna meningkatkan koordinasi antar pergantian *shift* dinas. Selain itu, catatan ini dapat menjadi referensi jika terjadi suatu permasalahan yang sama terhadap peralatan di masa depan. Oleh karena itu, *logbook* merupakan satu di antara fasilitas penunjang yang sangat penting bagi teknisi telekomunikasi penerbangan (Berlian Lindu Agung et al., 2021).

Logbook yang digunakan oleh teknisi di Perum LPPNPI Cabang Banda Aceh masih bersifat konvensional. *Logbook* tersebut masih menggunakan kertas sebagai media pencatatan sehingga sangat beresiko mengalami kerusakan secara fisik (Hardiansyah et al., 2019). Selain itu, catatan-catatan ini cenderung tidak sistematis dan beresiko hilang jika sudah terlampau lama (Wijaya & Fernandes Andry, 2020).

Pengembangan *logbook* dari konvensional menjadi digital dapat meningkatkan efektifitas penggunaan *logbook* seperti mencari rekam data yang sudah lama, mengurutkan data dan menciptakan tatanan data yang lebih terstruktur (Permana et al., 2023). Oleh karena itu, penulis melakukan digitisasi terhadap budaya yang ada dengan cara pengembangan aplikasi digital melalui pengembangan aplikasi *e-logbook* yang bernama *Digital Record of Event and Management* (DREAM) dengan konsep *Single Page Application* (SPA) di Perum LPPNPI Cabang Banda Aceh.

METODE

Pada penelitian kali ini, penulis menggunakan metode *Waterfall Model* yang memiliki tahapan sebagai berikut (Silvi Purnia et al., 2019)

a. *Requirement analysis and definition*

   Pada bagian ini dilakukan pada data yang tersedia dan mengumpulkan kebutuhan perangkat lunak yang akan dibuat. Bagian ini merupakan tahap awal inisialisasi definisi masalah untuk menyempurnakan teknik dalam proses pengembangan perangkat lunak. Pada akhir tahap analisis, definisi masalah telah diperoleh, disepakati oleh pengguna dan pengembang.

b. *System and Software Design*

   Pada bagian ini dilakukan penyusunan konsep dasar dari sebuah aplikasi yang akan dibuat. Mulai dari desain, logika-logika di dalam aplikasi, dan fitur-fitur yang ada.

c. *Implementation and unit testing*

   Konsep yang sudah dibuat sebelumnya kemudian diimplementasikan menjadi sebuah aplikasi yang utuh. Selain itu, dilakukan juga uji coba terhadap aplikasi berupa pengujian pada masing-masing fitur yang ada.

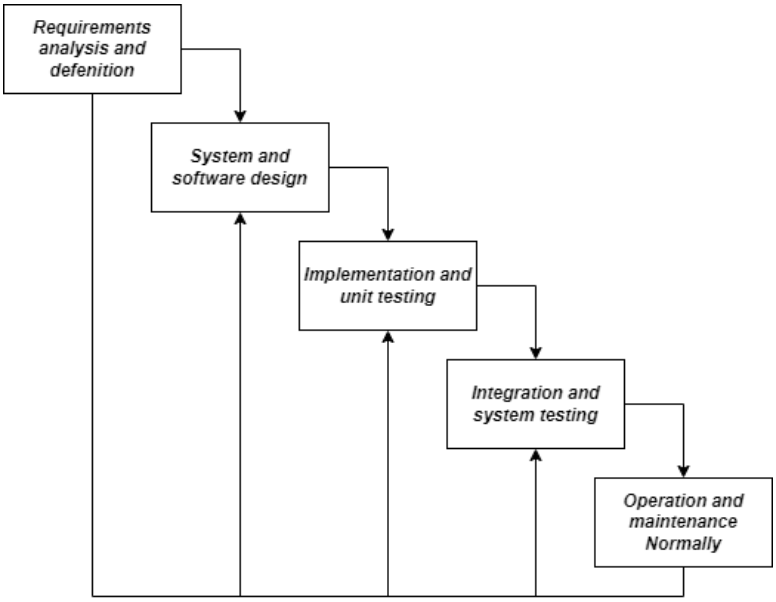
d. *Integration and system testing*

Pada bagian ini dilakukan uji coba terhadap sistem secara menyeluruh. Bagian ini bertujuan untuk memastikan tiap-tiap fitur dan bagian dapat bekerja secara normal serta tidak menghalangi antara yang satu dan lainnya.

e. *Operation and maintenance normally*

Tahapan ini merupakan bagian dimana aplikasi diimplementasikan secara riil dilapangan. Hal ini bertujuan untuk menguji performa dan kualitas aplikasi. Selain itu, tahapan ini juga bertujuan untuk menemukan *bug* yang tidak ditemukan pada tahapan sebelumnya.

Akan tetapi, pada penelitian kali ini penulis hanya akan melakukan sampai pada tahap *integration and system testing* karena aplikasi ini masih memiliki potensi untuk dikembangkan lebih lanjut.



Gambar 1. *Waterfall Model*

HASIL DAN PEMBAHASAN

Berikut adalah tahapan pengembangan aplikasi menggunakan metode *waterfall*:

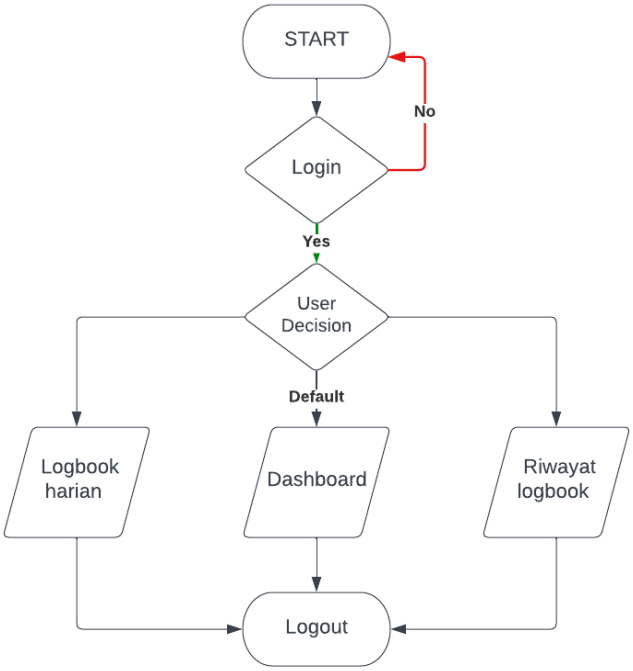
a. *Requirement analysis and defenition*

Pada tahapan ini, penulis menerima permintaan dari teknisi Perum LPPNPI Cabang Banda Aceh untuk membuat sistem yang mampu menggantikan fungsi *logbook* konvensional guna mengurangi penggunaan kertas dan penyimpanan yang lebih sistematis. Oleh karena itu, penulis melakukan studi kasus berupa wawancara kepada teknisi Perum LPPNPI Cabang Banda Aceh guna mengetahui kebutuhan aplikasi yang ingin dibangun. Selain itu, penulis juga mengumpulkan informasi mengenai data apa saja yang tercatat di *logbook* konvensional.

Berdasarkan hasil wawancara dan data yang telah dikumpulkan, penulis menyimpulkan bahwasannya *e-logbook* merupakan sistem yang dapat menjawab kebutuhan teknisi Perum LPPNPI Cabang Banda Aceh. Untuk membangun sistem aplikasi *e-logbook* maka diperlukan bahasa pemrograman *HTML*, *PHP, Javascript,* dan *CSS* (Giovandi & Kusumawati, 2022). Selain itu penulis juga menggunakan *framework Laravel* sebagai penunjang. *Framework* adalah seperangkat *library* yang teroganisir dalam arsitektur untuk memberikan kecepatan, akurasi, kenyamanan dan konsistensi dalam pengembangan aplikasi (Luthfi, 2017). Selain itu, dalam pengembangan aplikasi ini penulis menggunakan metode *Single Page Application* untuk membangun aplikasi *e-logbook. Single Page Application* (SPA) atau Aplikasi Halaman Tunggal adalah aplikasi *web* yang hanya menggunakan satu halaman *web* tapi tetap memiliki tindakan dinamis tanpa harus menyegarkan halaman (*refresh page*) (Belluano, 2018). Selain itu, penggunaan SPA bertujuan untuk menghemat *bandwidth* sehingga memiliki responsivitas tinggi (Ristyabudi & Thamrin, 2016).

b. *System and software design*

Berdasarkan hasil analisis dan kebutuhan yang telah dilakukan, berikut adalah *system design* dari aplikasi yang akan dibuat:



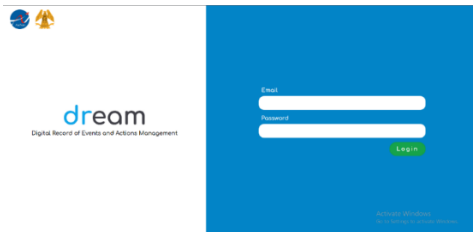
Gambar 2. *Block Diagram* Aplikasi DREAM

Berdasarkan Gambar 2, aplikasi ini memiliki sistem autentikasi dan otorisasi. Sistem autentikasi merupakan sistem yang berfungsi sebagai *security* dari aplikasi yang akan dibangun sehingga aplikasi tidak bisa diakses oleh orang asing. Selain itu, digunakan juga sistem otorisasi sehingga tampilan data yang ditampilkan pada aplikasi akan sesuai dengan otoritas pengguna. Aplikasi ini memiliki tiga komponen utama, yaitu *Logbook Harian*, *Dashboard* dan *Riwayat Logbook*. *Logbook Harian* yang merupakan halaman untuk mengisi kejadian yang terjadi selama dinas dan daftar anggota yang sedang berdinas. *Dashboard* merupakan halaman awal yang menampilkan data kejadian secara umum. *Riwayat Logbook* merupakan halaman yang berfungsi untuk mencari kejadian terdahulu berdasarka hari, tanggal, atau kata kunci terkait kejadian yang ingin dicari. Selain itu, pada halaman ini penulis juga dapat mencetak *logbook* menjadi *file* dengan format *pdf*.

c. *Implementation and unit testing*

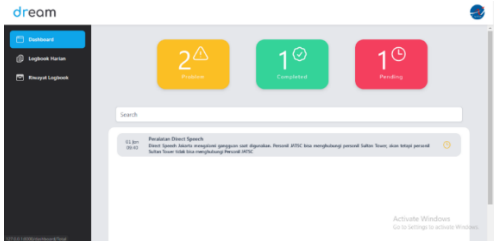
Berdasarkan *software design* yang telah dibuat, selanjutnya tiap-tiap bagian tersebut diimplementasikan menjadi sebuah aplikasi dengan menggunakan bahasa pemrograman. Bagian yang pertama adalah halaman *login*. Pada halaman ini pengguna harus memasukkan *email* dan *password* untuk menggunakan aplikasi. Data

yang dimasukkan pengguna akan mengalami proses autentikasi oleh sistem. Jika autentikasi berhasil, pengguna akan memasuki halaman *Dashboard*.



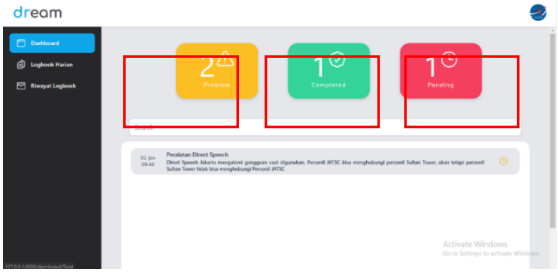
Gambar 3. Halaman *Login*

Halaman yang kedua adalah halaman *dashboard*. Terdapat empat unsur utama di dalam halaman *dashboard*, yaitu total permasalahan, jumlah permasalahan yang telah selesai, jumlah permasalahan yang belum selesa dan tabel yang menyajikan data kejadian. Secara *default* data yang disajikan di dalam tabel adalah data dari permasalahan yang belum selesai. Hal ini dilakukan agar personel teknik akan langsung mengetahui daftar permasalahan apa saja yang perlu dikerjakan.



Gambar 4. Halaman *Dashboard*

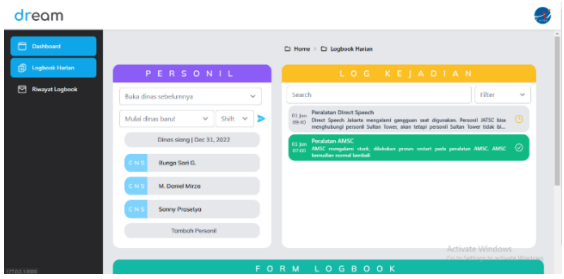
Terdapat empat unsur utama di dalam halaman *dashboard*, yaitu total permasalahan, jumlah permasalahan yang telah selesai, jumlah permasalahan yang belum selesa dan tabel yang menyajikan data kejadian. Secara *default* data yang disajikan di dalam tabel adalah data dari permasalahan yang belum selesai. Hal ini dilakukan agar personel teknik akan langsung mengetahui daftar permasalahan apa saja yang perlu dikerjakan.



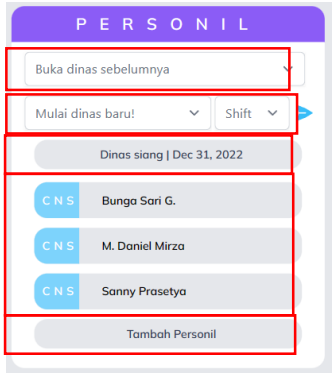
Gambar 5. Fitur Halaman Dashboard

Masing-masing *item* yang diberi kotak merah berfungsi sebagai penyaji data sekaligus tombol. Ketika pengguna mengklik *item problem* maka sistem akan secara otomatis menyajikan seluruh permasalahan baik yang sudah selesai maupun yang belum selesai di dalam tabel. Jika pengguna mengklik *item completed* sistem akan secara otomatis menyajikan data permasalahan yang telah selesai saja di dalam tabel. Jika pengguna mengklik *item pending* maka sistem akan secara otomatis menyajikan data permasalahan yang belum selesai saja di dalam tabel.

Di sisi sebelah kiri terdapat menu navigasi yang memudahkan pengguna untuk berpindah halaman. Terdapat pilihan *Dashboard* untuk menuju ke menu *dashboard*, *Logbook Harian* untuk menuju menu *logbook* harian dan *Riwayat Logbook* untuk menuju menu riwayat. Halaman *Logbook Harian* merupakan halaman tempat pengguna mencatat kegiatan. Di sisi sebelah kiri terdapat menu navigasi yang memudahkan pengguna untuk berpindah halaman. Terdapat pilihan *Dashboard* untuk menuju ke menu *dashboard*, *Logbook Harian* untuk menuju menu *logbook* harian dan *Riwayat Logbook* untuk menuju menu riwayat. Halaman ini memiliki 3 komponen utama, yaitu *Personel*, *Log Kejadian* dan *Form Logbook*.

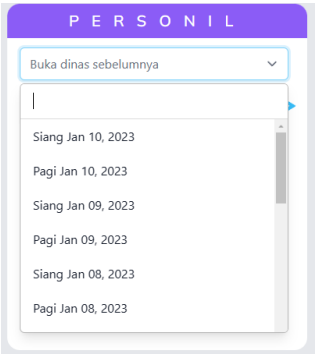


Gambar 6. Halaman *Logbook* Harian



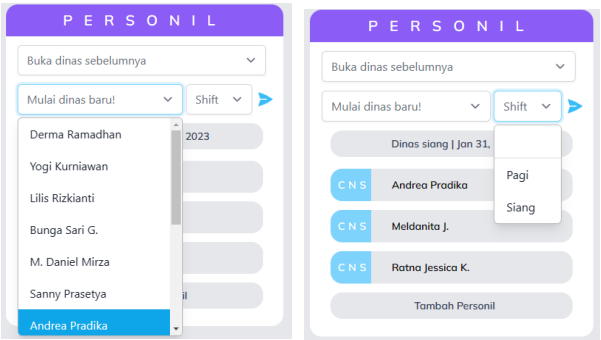
Gambar 7. Tampilan Komponen Personel

Record dinas merupakan suatu fitur yang memungkin pengguna untuk mengakses dinas-dinas sebelumnya. Pengguna cukup mengklik dinas yang ingin di akses atau bisa juga dengan cara mengetikkan tanggal atau *shift* yang diinginkan.



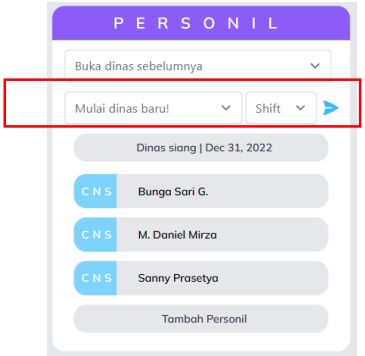
Gambar 8. Tampilan *Record* Dinas

Form Session merupakan sebuah *form* jika teknisi ingin memulai dinas. Bagian ini terdiri dari tiga bagian, yaitu daftar nama teknisi, *shift* dan tombol *submit*.



Gambar 9. Tampilan *Form Session* Personel

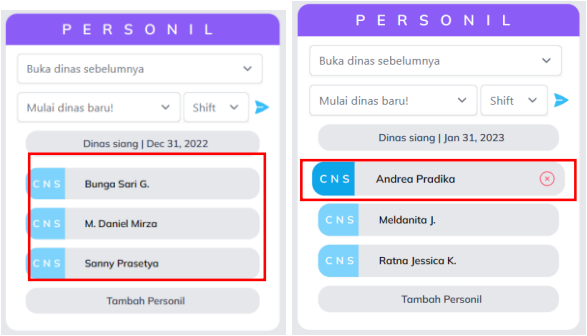
Selanjutnya terdapat bagian keterangan dinas. Bagian ini merupakan bagian yang menampilkan informasi berupa *shift* dinas dan tanggal dinas.



Gambar 10. Tampilan Keterangan Dinas Personel

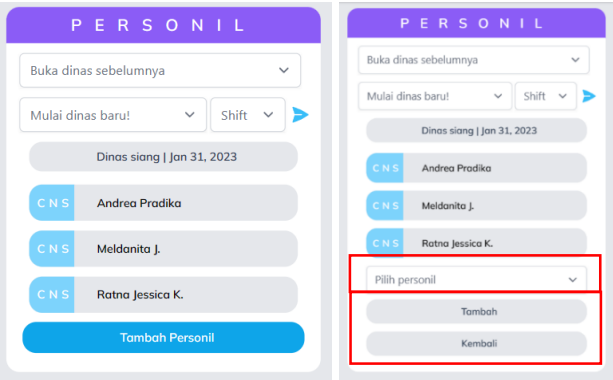
Daftar personel dinas merupakan bagian yang menampilkan daftar personel yang sebelumnya telah di *submit*. Selain menampilkan data personel, pengguna juga dapat menghapus personel dengan cara mengarahkan kursor ke nama personiel terpilih

hingga muncul tanda silang berwarna merah lalu mengklik tanda tersebut seperti gambar di atas.



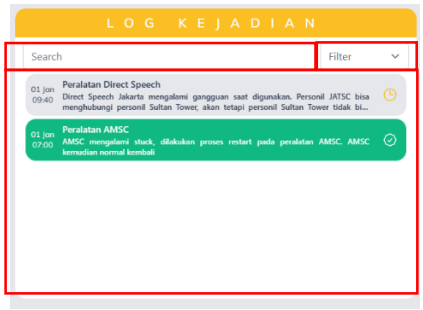
Gambar 11. Tampilan Daftar Personel

Tambah personel merupakan fitur yang berfungsi untuk menambahkan personil baru pada dinas yang sudah di buat. Arahkan *cursor* ke tambah personil lalu klik. Setelah itu akan muncul menu *dropdown* untuk memilih personil, tombol tambah personil dan tombol kembali.



Gambar 12. Tampilan Tambah Personel

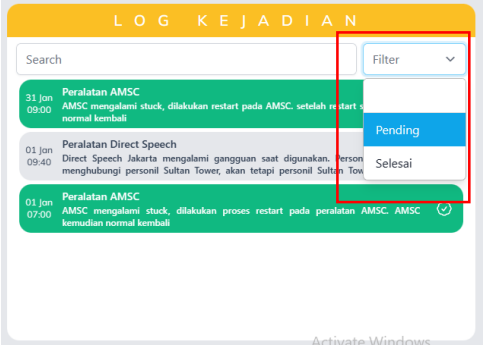
Komponen yang selanjutnya adalah *Log* Kejadian. Komponen ini merupakan komponen yang menampilkan permasalahan yang telah selesai dan belum selesai.



Gambar 13. Tampilan *Log* Kejadian

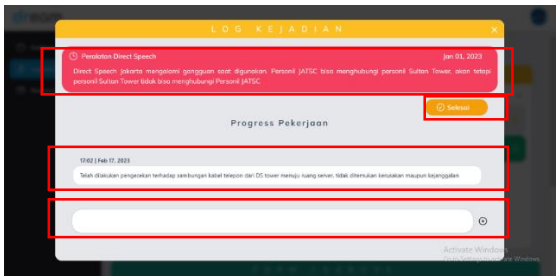
Pada menu ini terdapat tombol *search* untuk mencari permasalahan dan tombol *filter* untuk menampilkan permasalahan berdasarkan kategori yang di pilih. Terdapat dua

kategori yang dapat dipilih, yaitu *Selesai* untuk menampilan masalah yang sudah selesai saja dan *Pending* untuk menampilkan permasalahan yang belum selesai saja.



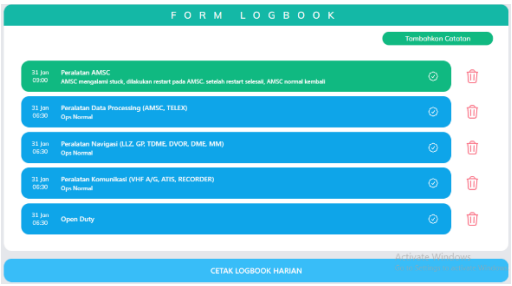
Gambar 14. Tampilan *Filter Log* Kejadian

Tiap-tiap *item* permasalahan merupakan sebuah *button* yang dapat di klik. Ketika *item* permasalahan di klik, maka akan menampilkan progress pekerjaan dari seluruh dinas yang melakukan pekerjaan terhadap permasalahan tersebut.



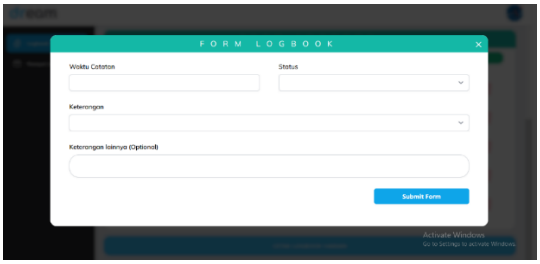
Gambar 15. Tampilan *Progress* Pekerjaan

Gambar di atas merupakan tampilan jika mengklik *item* permasalahan. Pada bagian ini terdapat permasalahan awal, tombol untuk menyatakan bahwasannya permasalahan telah selesai, *record* terhadap *progress* pekerjaan masalah terkait, dan *form* untuk menambahkan *progress* pekerjaan.



Gambar 16. Tampilan *Form Logbook*

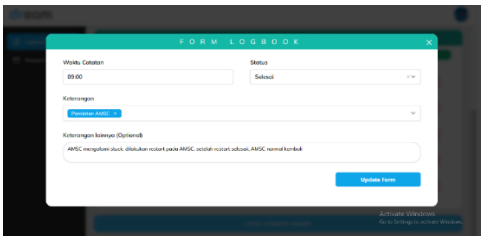
Klik tombol *Tambahkan Catatan* untuk menambahkan catatan. Setelah di klik, maka akan muncul tampilan seperti di bawah ini.



Gambar 17. Tampilan isi *Form Logbook*

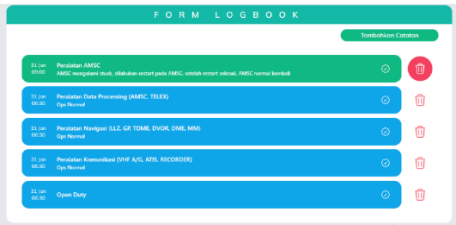
Pada tampilan tersebut terdapat empat kolom, yaitu waktu catatan untuk memilih kapan kejadian yang dicatat terjadi, status kejadian, keterangan dan keterangan lainnya. Klik *Submit Form* untuk menyimpan catatan yang ingin ditambahkan.

Jika ingin mengedit *form* yang telah disimpan, klik catatan yang ingin di edit dibagian daftar catatan. Setelah di klik, maka halaman akan menampilkan *form logbook*. Setelah muncul tampilan seperti pada Gambar 3.17, ubah data yang ingin anda ubah. Setelah selesai mengubah data, klik tombol *Update Form* untuk menyimpan data yang telah diubah.



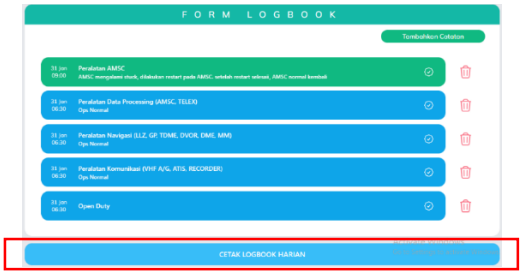
Gambar 18. Tampilan Edit *Form*

Jika ingin menghapus *form* yang telah disimpan, klik simbol tempat sampah yang berada di sebelah kanan item yang ingin di hapus dari bagian daftar catatan. Setelah di klik, maka catatan akan dihapus oleh sistem.



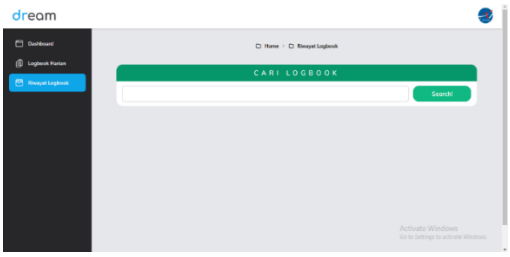
Gambar 19. Tampilan Hapus *Form*

Fitur yang terakhir adalah cetak *logbook*. Pengguna dapat menyimpan catatan yang telah dibuat kedalam bentuk *file* berformat *pdf*. Hal ini sangat berguna jika pengguna ingin menggunakan *soft copy* dari catatan yang telah dibuat.



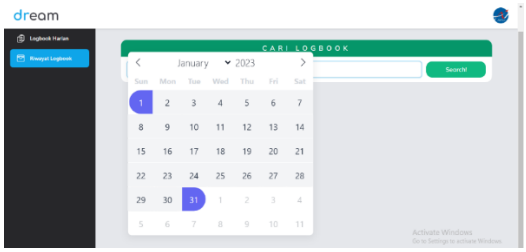
Gambar 20. Tampilan Cetak *Logbook*

Halaman yang terakhir adalah halaman riwayat *logbook*. Halaman *Riwayat Logbook* meruapakan halaman tempat pengguna mengakses *logbook* terdahulu yang pernah disimpan. Selain itu, halaman riwayat juga menyediakan fitur cetak *logbook*, hapus *logbook* dan daftar *logbook*.



Gambar 21. Tampilan Utama Riwayat *Logbook*

Untuk memulai pencarian, klik kolom *search* dan pilih tanggal *logbook* yang anda lihat. Aplikasi DREAM menyediakan fitur pengguna untuk memilih tanggal berdasarkan *range* yang dipilih. Fitur ini memungkinkan pengguna untuk mengakses dan melakukan aksi seperti cetak atau hapus data lebih dari satu dalam satu waktu.



Gambar 22. Tampilan Fitur *Range* Pada Pilih Riwayat

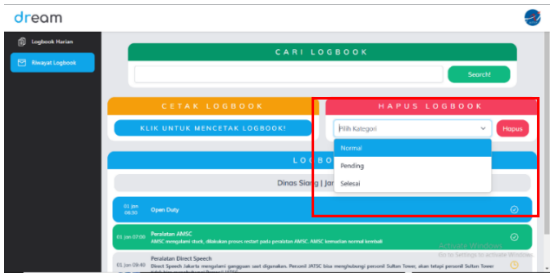
Setelah memilih tanggal dari *logbook* yang ingin diakses, klik tombol *search*. Setelah itu maka sistem akan secara otomatis mencari data yang sesuai dengan kriteria tanggal yang telah dipilih.

Gambar 23. Tampilan Hasil Pencarian

Jika ingin mencetak *form* berdasarkan *range* tanggal yang dipilih, klik tombol *Klik Untuk Mencetak Logbook* yang berada di sisi kiri. Setelah di klik, maka sistem akan secara otomatis mencetak seluruh catatan yang berada di dalam *range* tanggal.

Gambar 24. Tampilan Mencetak Riwayat

Jika ingin menghapus *form* yang telah disimpan, klik tombol Hapus yang berada di sisi sebelah kanan. Pilih kategori yang ingin di hapus. Pengguna dapat memilih lebih dari satu kategori. Setelah di klik, maka catatan akan dihapus oleh sistem berdasarkan *range* tanggal dan kategori yang telah dipilih.



Gambar 25. Tampilan Hapus Riwayat

d. *Integration and system testing*

Komponen-komponen utama yang telah dibuat sebelumnya kemudian digabungkan menjadi suatu aplikasi yang diberi nama *Digital Record of Event and Management* (DREAM). Aplikasi ini kemudian dijalankan pada server lokal dan dilakukan pengujian aplikasi. Berikut adalah tabel validasinya:

Tabel 1. Pengujian Sistem Aplikasi

No	HALAMAN	BAGIAN	FITUR	KETERANGAN
1	*Login*	Input *username* dan *password*	Autentikasi dan otorisasi	Normal
2	*Dashboard*	Total Permasalahan	Menampilkan total permasalahan	Normal
		Jumlah permasalahan yang telah selesai	Menampilkan jumlah permasalahan yang telah selesai	Normal

		Jumlah permasalahan yang belum selesai	Menampilkan jumlah permasalahan yang belum selesai	Normal
		Search bar	Mencari kejadian berdasarkan kata kunci kejadian	Normal
		Tabel permasalahan	Menyajikan data kejadian	Normal
3	*Logbook* Harian	Personel	*Record* dinas	Normal
			Form session	Normal
			Keterangan dinas	Normal
			Daftar personel dinas	Normal
			Hapus personel	Normal
			Tambah personel	Normal
		Log Kejadian	*Search bar*	Normal
			Filter	Normal
			Open progress pekerjaan	Normal
			Tambah *record progress*	Normal
			Status *progress*	Normal
			Update status *progress*	Normal
		Form Logbook	Tambah *Form*	Normal
			Edit *Form*	Normal
			Tampilkan *Form*	Normal
			Hapus *Form*	Normal
			Cetak *Logbook* Harian	Normal
4	Riwayat *Logbook*	Cari *Logbook*	Mencari *logbook* berdasarkan *range* tanggal	Normal
		Daftar *Logbook*	Menampilkan daftar *logbook* hasil pencarian	Normal
		Cetak *Logbook*	Mencetak *logbook* yang telah dipilih dalam bentuk *file* pdf	Normal
		Hapus *Logbook*	Menghapus *logbook* yang telah dipilih berdasarkan kategori	Normal

PENUTUP

Kesimpulan

Berdasarkan hasil data penelitian yang telah dilakukan, berikut adalah kesimpulannya:

1. Aplikasi *logbook* ini memanfaatkan metode *single page application* sehingga pengolahan data dapat dilakukan dengan cepat oleh sistem.

2. Kelebihan dari aplikasi *logbook* digital berbasis *single page application* ini adalah *logbook* tersimpan lebih aman, memudahkan teknisi ketika mencari rekam kejadian, teknisi tidak perlu menulis *logbook* secara manual, mengurangi biaya penggunaan kertas dan seluruh data catatan di olah secara otomatis oleh sistem dengan tatanan yang terstruktur.
3. Aplikasi yang dibuat dapat berjalan dengan baik dan sesuai dengan *requirement analysis*.

DAFTAR PUSTAKA

Asfariza, A., Fitriadi, R., & Setiawan, P. (2022). Perancangan Sistem Digital Log Book Untuk Penggunaan dan Perawatan Peralatan Laboratorium Fakultas Teknik Berbasis Website. *Simposium Nasional RAPI XX*.

Belluano, P. L. L. (2018). Pengembangan Single Page Application Pada Sistem Informasi Akademik. *ILKOM Jurnal Ilmiah*, *10*(1).

Berlian Lindu Agung, M., Pambudiyanto, N., & Diana Puspita, R. (2021). Rancangan E-Logbook Berbasis Web dan Database Untuk Teknisi Telekomunikasi Dan Navigasi Udara. *Seminar Nasional Inovasi Teknologi Penerbangan (SNITP)*.

Direktorat Jenderal Perhubungan Udara. (2019). KP 35 Tentang Pedoman Teknis Operasional Peraturan Keselamatan Penerbangan Sipil Bagian 171-12 (Advisory Circular Part 171-12) Prosedur Pemeliharaan dan Pelaporan Fasilitas Telekomunikasi Penerbangan. *Direktorat Jenderal Perhubungan Udara*.

Febriani, O. M., & Wahyuni, T. (2017). *Perancangan Sistem E-Document Administrasi Logbook Penelitian Pada Unit Layanan di Bandar Lampung*.

Giovandi, K., & Kusumawati, A. (2022). Aplikasi E-Commerce Berbasis Web pada PT. Triwarga Dian Sakti. *Jurnal Mahasiswa Institut Teknologi Dan Bisnis Kalbis*, *8*(1).

Hardiansyah, R., Erlan Afiuddin, A., & Khoirul Hasin, M. (2019). Perancangan dan Pembuatan Sistem Informasi Penyimpanan Data Limbah B3 Menggunakan Metode Personal Extreme Programming (PXP) di Industri Asam Fosfat. *National Conference Proceeding on Waste Treatment Technology*.

Jaya, D. K., Susilowati, L. E., & Akhdiyat, H. R. (2023). SISTEM INFORMASI QUICK RESPONSE CODE (QR CODE) SEDERHANA UNTUK MENGOPTIMALKAN INVENTARISASI PEMINJAMAN DAN PENGGUNAAN ALAT-ALAT LABORATORIUM DI JURUSAN ILMU TANAH, UNIVERSITAS MATARAM. *Jurnal Abdi Insani*, *10*(2), 771–783. https://doi.org/10.29303/abdiinsani.v10i2.951

Luthfi, F. (2017). Penggunaan Framework Laravel Dalam Rancang Bangun Modul Back-End Artikel Website Bisnisbisnis.ID. In *JISKa* (Vol. 2, Issue 1).

McAlpine, H., Cash, P., & Hicks, B. (2017). The role of logbooks as mediators of engineering design work. *Design Studies*, *48*, 1–29. https://doi.org/10.1016/j.destud.2016.10.003

Nugroho, H., Darmawan, A., & Sufyan, A. (2016). PERANCANGAN SISTEM INFORMASI ELEKTRONIK LOG BOOK PENANGKAPAN IKAN BERBASIS WEB. *Jurnal Kelautan Nasional*, *11*(1), 53. https://doi.org/10.15578/jkn.v11i1.6066

Permana, A. A., Fadillah, B., & Taufiq, R. (2023). Penggunaan Metode V-Model Untuk merancang Sistem Informasi E-Logbook Berbasis Website. *Jurnal Minfo Polgan*, *12*(2), 30–2023. https://doi.org/10.33395/jmp.v12i2.12347