

## BAB III

### PELAKSANAAN KERJA PROFESI

#### 3.1 Bidang Kerja

Praktikan menangani tugas operasional jalan tol selama program kerja profesi di Divisi Operasi. Meskipun berperan sebagai staf IT, praktikan juga terlibat dalam tugas karena IT di divisi ini tidak dipisahkan dari operasi.

Praktikan melakukan berbagai tugas untuk membantu divisi operasi dan pengelolaan IT perusahaan. Pengembangan Website, pengelolaan konten, hingga perbaikan konten website perusahaan. Ini adalah pekerjaan yang dilakukan selama praktik kerja:

- Pengembangan Web Pelaporan:
  - Website ini cepat dan efektif membantu petugas lapangan membuat laporan harian.
  - Mengganti metode pelaporan manual seperti formulir kertas dan Google Form dengan sistem yang lebih terorganisir dan tercatat di database.
  - Memberi manajer alat bantu untuk menilai kinerja karyawan lapangan.
- Pengetahuan Tentang Teknologi Gerbang Tol:
  - Belajar bagaimana mesin tapping dan sensor di gerbang tol berfungsi. Sensor ini mencatat dan mengawasi kendaraan yang melintasi jalan.
  - *Monitoring* kondisi semua teknologi yang ada di tol.
- Perbaikan Konten Website Perusahaan:
  - Membantu kepala divisi dan staf IT mengubah konten situs [jtd.co.id](http://jtd.co.id) dalam bahasa Inggris dan Indonesia.
  - Mengubah data dan menyederhanakan bahasa agar lebih mudah dipahami oleh pengguna.
- *Monitoring* Infrastruktur Tol:
  - Mengawasi tes jalan seperti Pengujian Kekasaran IRI dan kekesatan jalan.
  - Membuat dokumentasi selama proses pengujian.
- Pengelolaan Platform Media Sosial:
  - Mengelola konten Instagram perusahaan untuk meningkatkan interaksi dan membangun citra perusahaan.

- Membuat dan mengedit video, serta mengedit poster untuk kebutuhan media sosial.
- Pembuatan Laporan Untuk Divisi Operasi:
  - Manajer menggunakan data ini untuk menilai kinerja operasional dan membuat strategi untuk meningkatkan layanan.
  - Laporan bulanan mencakup kecelakaan, pelanggaran, total pemasukan tol, dan informasi lainnya.

### 3.2 Pelaksanaan Kerja

Selama pelaksanaan kerja, praktikan belajar cara menggunakan mobile reader di gerbang tol, mendokumentasikan aktivitas pengawasan Instagram JTD, dan membuat beberapa konten video dan foto yang akan ditinjau sebelum diunggah. Praktikan juga akan memantau sistem *real-time* dari pusat komando TIOS dengan ALPR, CCTV, sensor banjir dan AVC untuk memastikan keselamatan jalan. Selain itu, praktisi dapat memperbarui konten situs JTD melalui *dashboard* dinamis tanpa *coding* apapun, termasuk membuat konten berbahasa Inggris setelah melakukan backup data di Excel, dengan hasil akhir ditinjau dan disesuaikan oleh kepala divisi.

Pada tahap pengerjaan website, praktikan melakukan pelaksanaan kerja terdiri dari analisis kebutuhan, perancangan, pencarian desain UI/UX, pengembangan *frontend*, pengembangan *backend*, dan terakhir pengujian. Beberapa *bug* ditemukan selama praktikan mengembangkan website ini, dan setelah diperbaiki, hasil dipresentasikan.

Tabel 3. 1 Alur Kegiatan

Kegiatan	Agustus				September				Oktober			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
Perkenalan pekerjaan	█				█							
memperbarui konten situs JTD		█	█	█	█							
Belajar mobile reader	█				█							
membuat beberapa konten video dan foto di Instagram JTD			█		█							
memantau sistem <i>real-time</i> dari pusat komando TIOS dengan ALPR, CCTV, sensor banjir dan AVC untuk memastikan keselamatan jalan.		█			█							
Analisis Kebutuhan					█							
Perancangan alur					█	█						
Pencarian desain UI/UX							█					
Pengembangan <i>frontend</i>							█	█				
Pengembangan <i>backend</i>									█	█		
Pengujian										█		
Perbaiki <i>bug</i>											█	█

Berdasarkan Tabel 3.1, pengembangan website pelaporan dilakukan oleh praktikan selama 2 bulan. Website tersebut akan terus dikembangkan dari sisi fitur

dan antarmuka oleh tim IT. Pada pelaksanaannya pembimbing eksternal bertanggung jawab memberikan arahan dan memantau setiap pekerjaan praktikan. Praktikan juga melaporkan kemajuan proyek kepada pembimbing eksternal setiap minggunya.

### 3.2.1 Perancangan

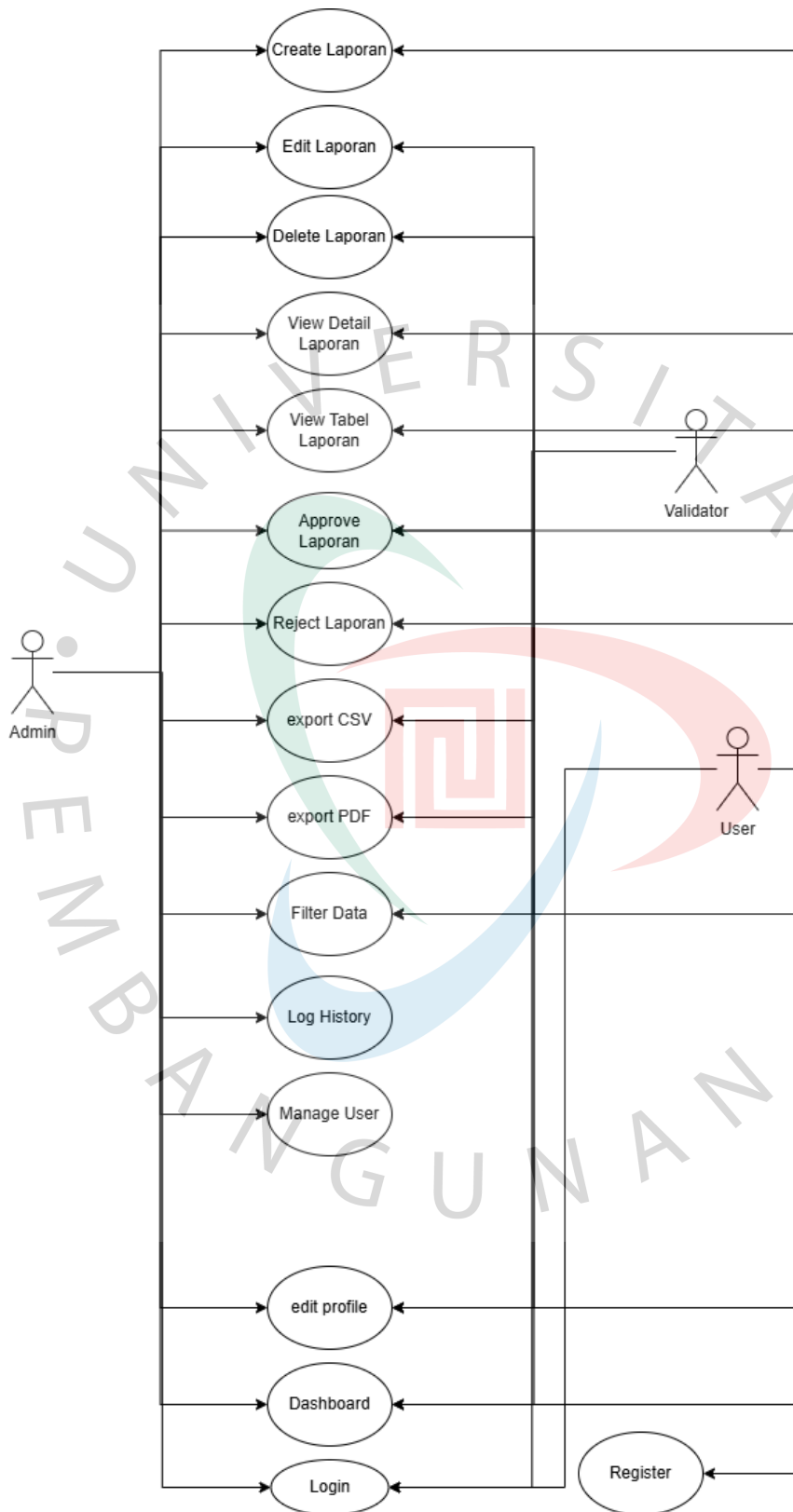
Selama tahap perancangan perangkat lunak, praktikan dan pembimbing bekerja sama untuk merancang alur aplikasi yang disesuaikan dengan kebutuhan pengguna. Praktikan juga membahas persyaratan teknis yang mendukung proses kerja di lapangan. Evaluasi kebutuhan operasional perusahaan menentukan setiap keputusan.

Fitur utama yang disepakati adalah CRUD (Create, Read, Update, Delete). Selain itu, pastikan untuk merencanakan fitur ekspor data ke dalam format CSV dan PDF. Fitur PDF menggunakan template khusus yang sesuai dengan standar laporan perusahaan untuk memastikan laporan yang dihasilkan konsisten.

Untuk keamanan data, pengeditan profil pengguna dan autentikasi akun adalah fitur tambahan. Aktivitas pengguna dicatat dalam fitur *log history*, sedangkan fitur manajemen pengguna membantu administrator mengatur hak akses. Semua fitur dirancang agar aplikasi dapat berjalan dengan baik dan memenuhi kebutuhan manajemen.

Pembimbing meminta perubahan pada alur aplikasi selama proses pengembangan. Awalnya, alur hanya satu arah tanpa proses persetujuan, dan kemudian alur diubah menjadi dua arah dengan mekanisme persetujuan untuk validasi data. Perubahan ini dilakukan untuk meningkatkan akurasi dan transparansi laporan.

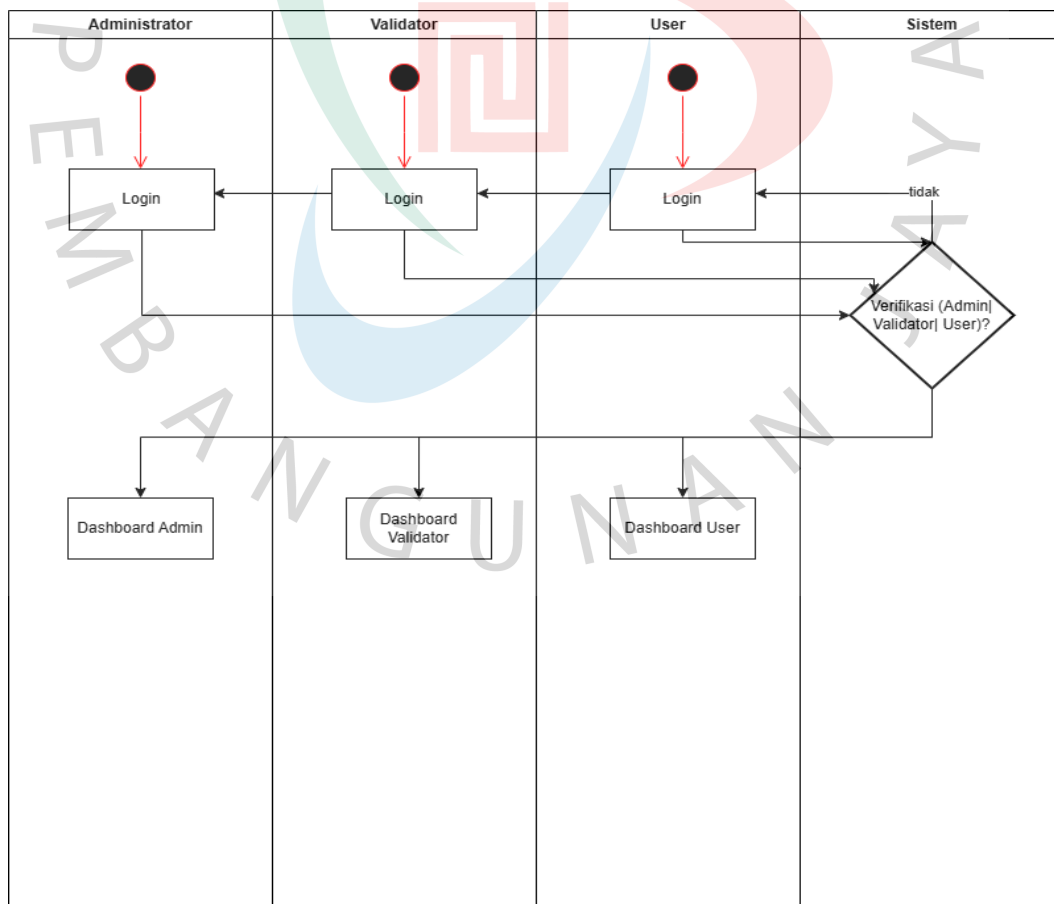
Selain itu, praktikan dan pembimbing melakukan pengujian ulang untuk memastikan bahwa aplikasi berfungsi sesuai dengan kebutuhan baru. Semua fitur ini dirancang untuk membantu pekerja lapangan memberikan laporan dengan cepat, akurat, dan dalam waktu nyata.



Gambar 3. 1 Diagram UseCase Website pelaporan

*Use case diagram* adalah alat penting dalam pemodelan perangkat lunak yang menggambarkan semua fungsi yang akan dilakukan oleh sistem dan berfungsi sebagai representasi visual interaksi antara pengguna dan sistem. Digunakan untuk memahami fungsi apa yang ada dalam sistem dan siapa yang dapat mengakses fungsi tersebut. Untuk membuat *use case diagram* yang baik, Anda harus melakukan analisis dan penulisan kebutuhan yang mendalam (Taufan et al., 2022).

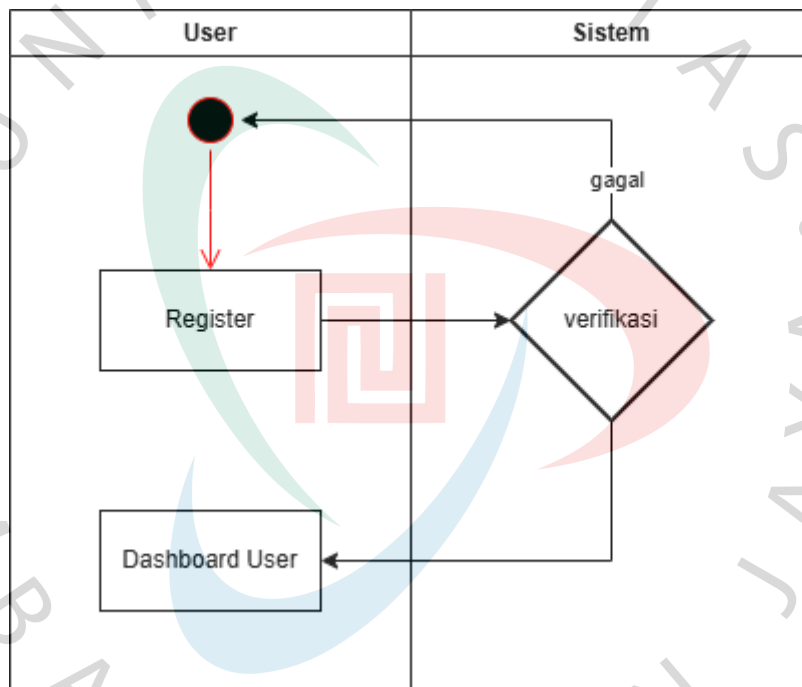
Gambar 3.1 menunjukkan versi terbaru dari use case, yang menggabungkan tiga peran utama yaitu, *user*, *validator*, dan administrator dengan alur dua arah. Dalam proses pelaporan, manajer bertindak sebagai *validator*. Selain itu, use case tersebut menjelaskan tugas dan tugas yang dapat dilakukan oleh setiap peran sistem yang dikembangkan oleh praktikan dan pembimbing. Praktikan menggunakan use case di atas sebagai referensi untuk membuat diagram aktivitas.



**Gambar 3. 2 Diagram Activity Login**

Gambar 3.2 menyajikan diagram aktivitas (*activity diagram*) untuk proses *login* setiap pengguna di website pelaporan. Diagram dimulai dengan tampilan antarmuka *login*, di mana sistem mengidentifikasi pengguna sebagai *administrator*, *validator*, atau *user*.

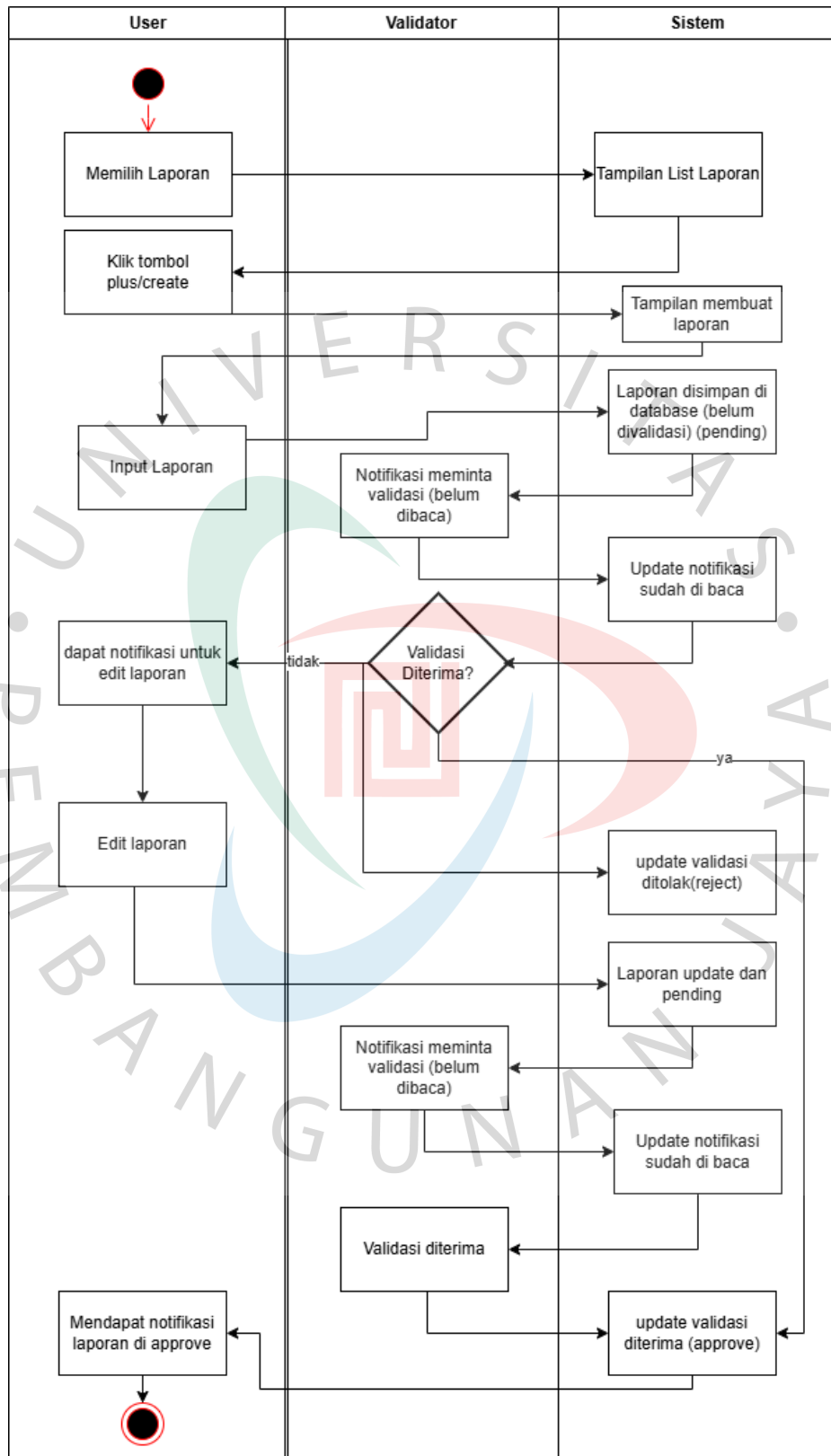
Sistem akan mengembalikan pengguna ke antarmuka *login* jika *email* atau *password* salah, jika berhasil, sistem akan menentukan peran pengguna dan mengarahkan mereka ke *dashboard* yang sesuai. *Administrator* akan masuk ke *dashboard administrator*, *validator* akan masuk ke *dashboard validator*, dan *user* akan masuk ke *dashboard user*.



Gambar 3. 3 Diagram Activity Register

Berdasarkan Gambar 3.3, proses registrasi akun *user* digambarkan dalam diagram di atas. Proses dimulai di antarmuka registrasi, di mana sistem memverifikasi data yang dimasukkan. Jika verifikasi tidak berhasil, *user* akan dikembalikan ke antarmuka registrasi. Namun, setelah semuanya berjalan dengan baik, pengguna akan diarahkan ke *dashboard* pengguna.

Karena akun dengan *role validator* harus dibuat oleh admin, registrasi ini hanya berlaku untuk *role user*. Sementara itu, akun admin dibuat secara otomatis melalui seed saat website pertama kali diluncurkan.



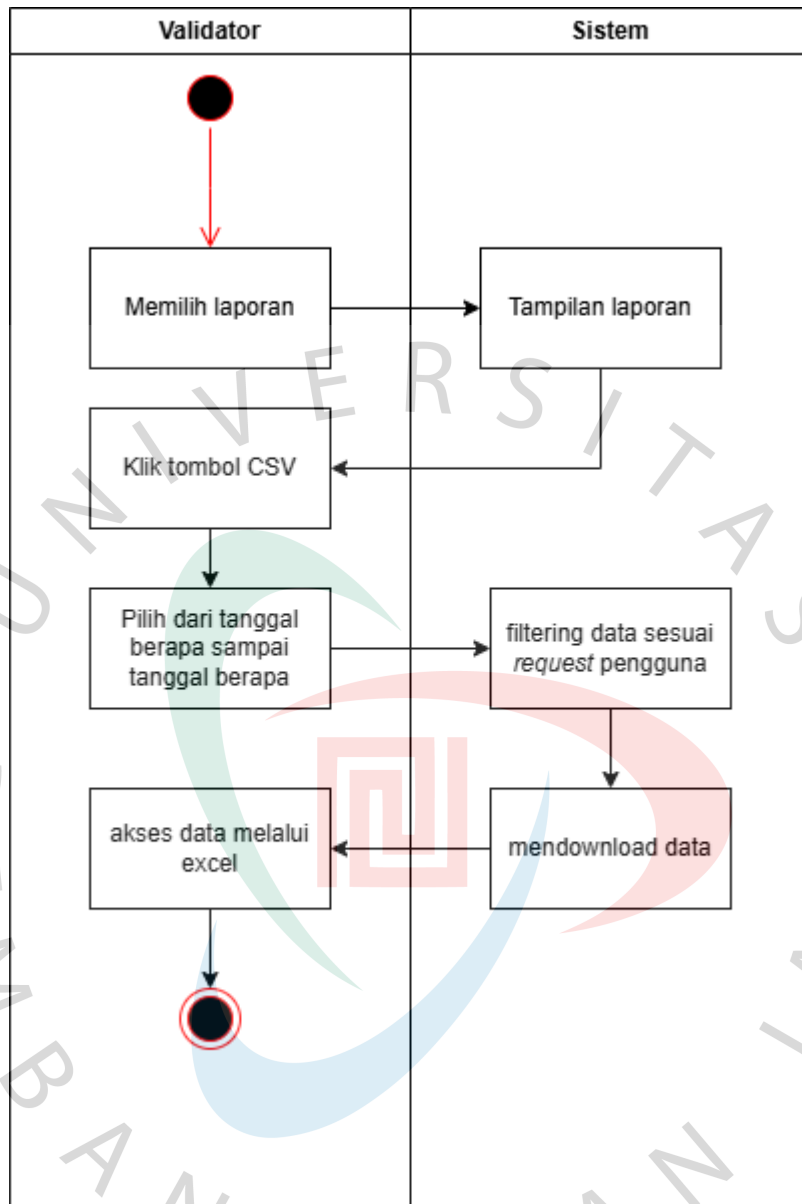
Gambar 3. 4 Diagram Activity Proses Pembuatan Laporan Hingga Tahap Persetujuan



Gambar 3.4 menunjukkan proses utama dalam website, yaitu alur pembuatan laporan. Setelah masuk ke *dashboard*, *user* dapat menemukan menu laporan di sidebar yang sesuai dengan jenis pekerjaan mereka. Setelah memilih menu ini, *user* akan dibawa ke halaman yang menampilkan daftar laporan dalam bentuk tabel.

Klik tombol "+" untuk membuat laporan baru, yang akan membawa *user* ke antarmuka pembuatan laporan. Setelah laporan dimasukkan, *validator* akan menerima pemberitahuan untuk memeriksanya.

*Validator* akan menyetujui (*approve*) laporan jika tidak ada kesalahan, dan *user* akan menerima notifikasi persetujuan. Jika laporan ditolak (*reject*), *user* akan menerima notifikasi penolakan. Selain itu, notifikasi ini memungkinkan *user* mengedit laporan yang ditolak. Ketika *user* mengklik notifikasi, *user* akan dibawa ke antarmuka edit laporan. Laporan kembali ke proses validasi setelah perubahan selesai. Pada tahap ini, proses selesai jika laporan disetujui.

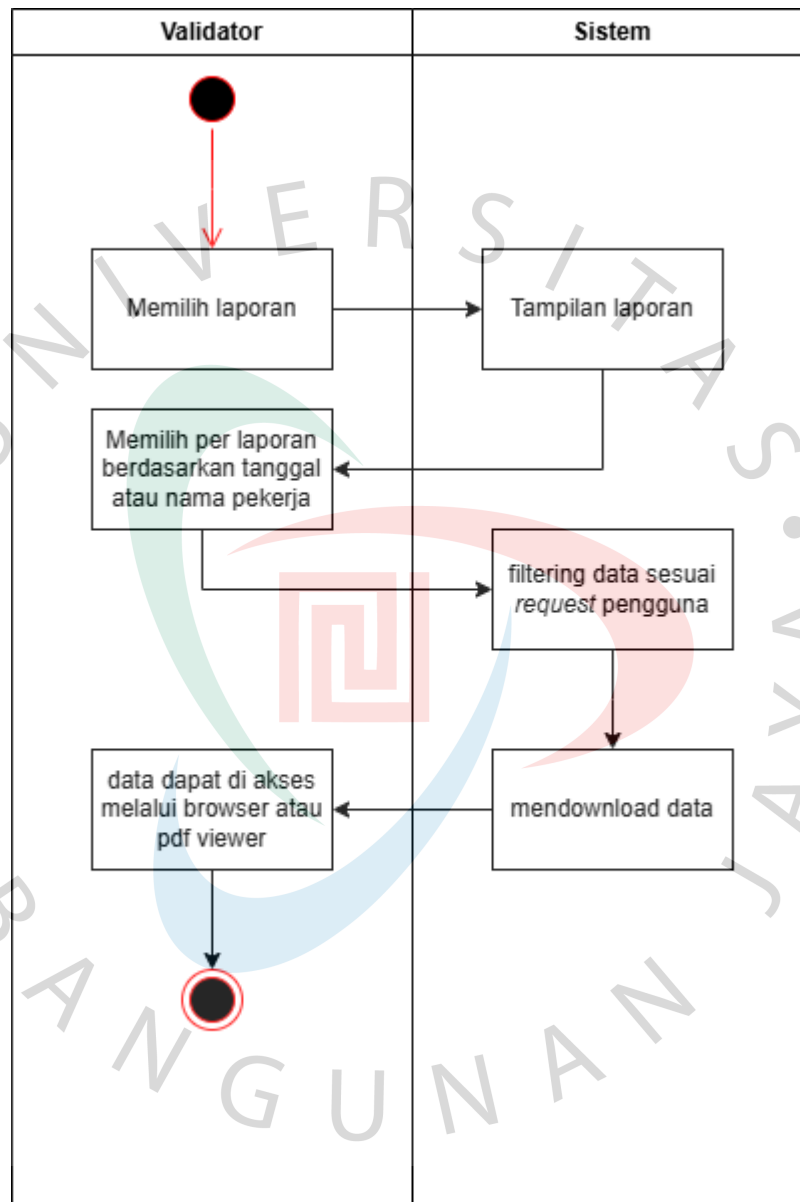


Gambar 3. 5 Diagram Activity Fitur Ekspor CSV

Gambar 3.5 menjelaskan alur pengunduhan data dalam format CSV secara terstruktur. Proses dimulai dengan admin atau *validator* memilih laporan dari daftar yang tersedia. Setelah memilih laporan, pengguna akan diarahkan ke antarmuka daftar laporan dengan tombol CSV.

Setelah itu, sistem meminta rentang tanggal dari pengguna setelah mengklik tombol CSV. Setelah rentang dimasukkan, sistem memfilter data sesuai kriteria yang dimasukkan.

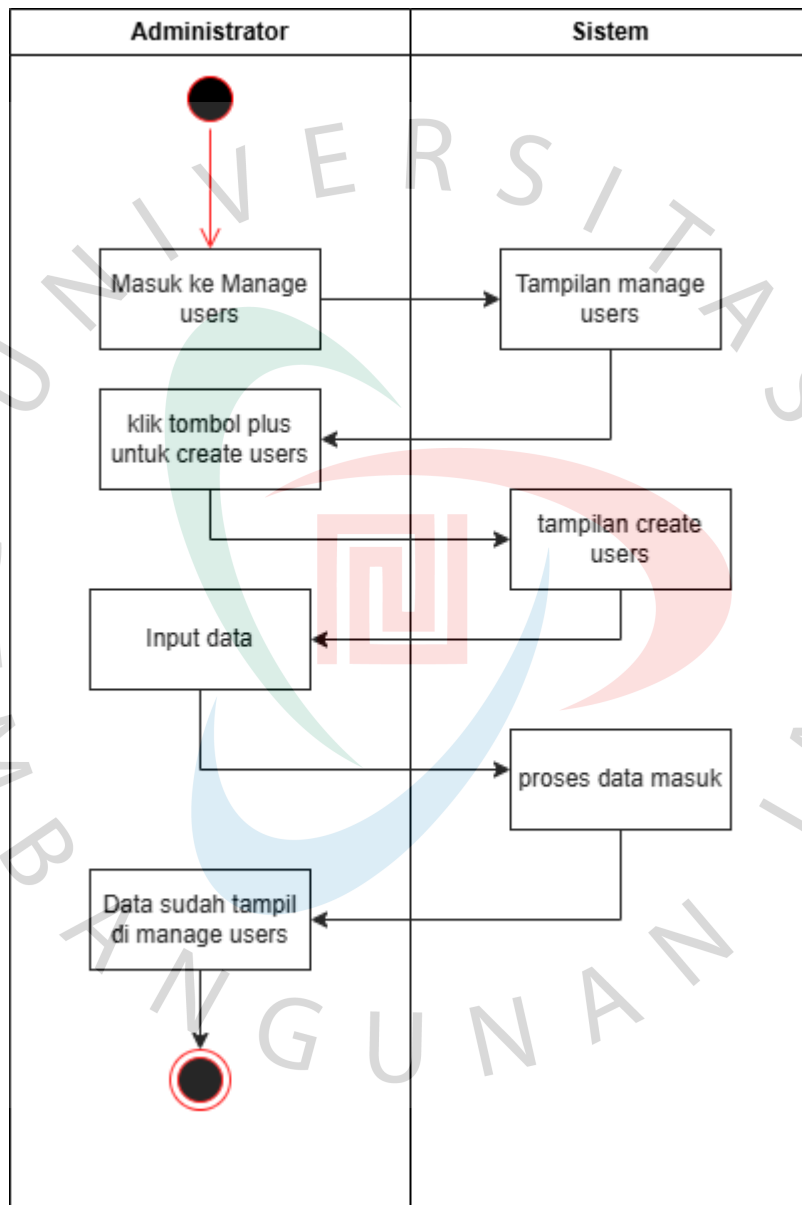
Selanjutnya, sistem memproses data yang telah difilter dan mengunduhnya dalam format CSV. File hasil unduhan dapat dibuka dan dikelola menggunakan aplikasi seperti Microsoft Excel untuk analisis dan kebutuhan lainnya.



Gambar 3. 6 Diagram Activity Fitur Ekspor PDF

Alur ekspor laporan ke dalam format PDF untuk masing-masing data dijelaskan pada Gambar 3.6. Proses dimulai dengan *validator* memilih laporan untuk diekspor. Setelah memilih, *validator* diarahkan ke antarmuka daftar laporan, di mana *validator* memilih nama laporan untuk diekspor ke PDF.

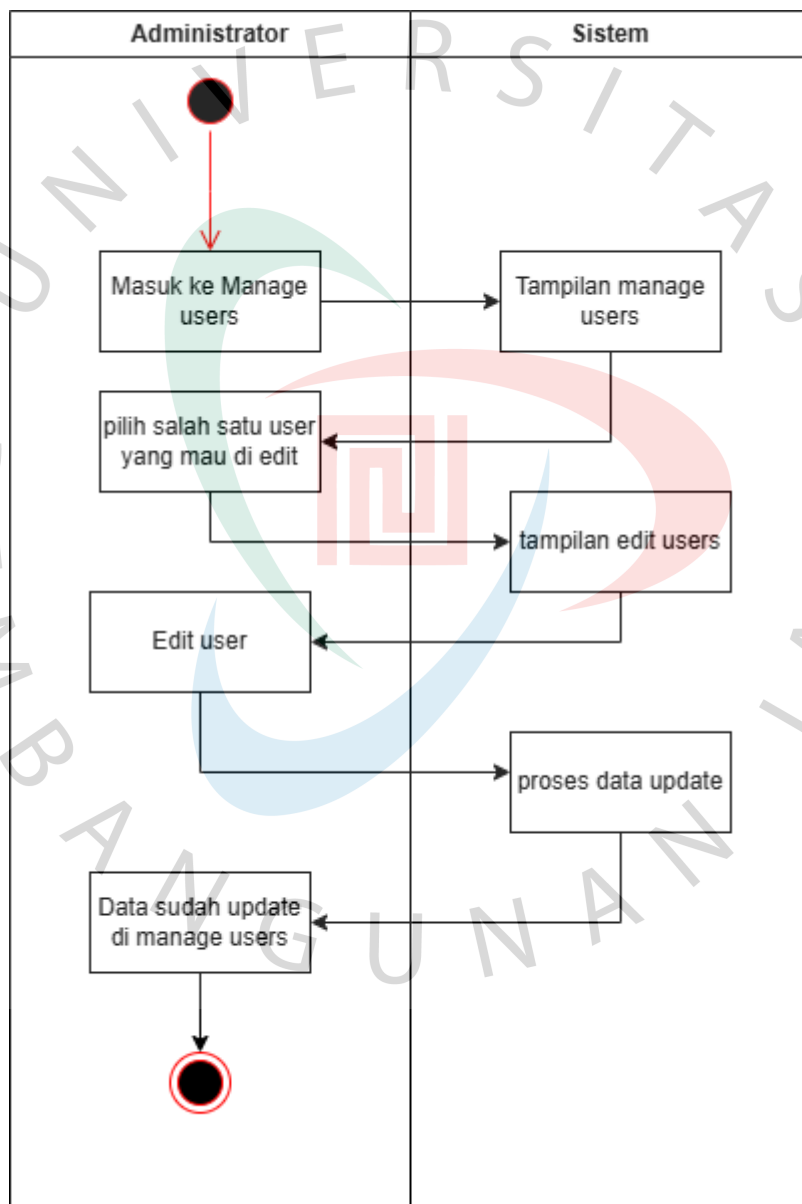
Ini tidak mendukung ekspor massal, ekspor PDF hanya dapat dilakukan untuk satu data. Setelah memilih data, klik tombol "Document" untuk mengunduh file PDF. File ini dapat dibuka menggunakan editor PDF setelah laporan selesai diisi dan melakukan proses tanda tangan.



**Gambar 3. 7 Diagram Activity Manage User (Create User)**

Gambar 3.7 menunjukkan diagram aktivitas proses *manage users*, yang dimulai dengan proses pembuatan *user* baru. Untuk memulai, admin harus memilih menu *manage user* di *dashboard*. Kemudian, dia akan dibawa ke antarmuka *manage user* dengan mengklik tombol *Plus* atau *Create*.

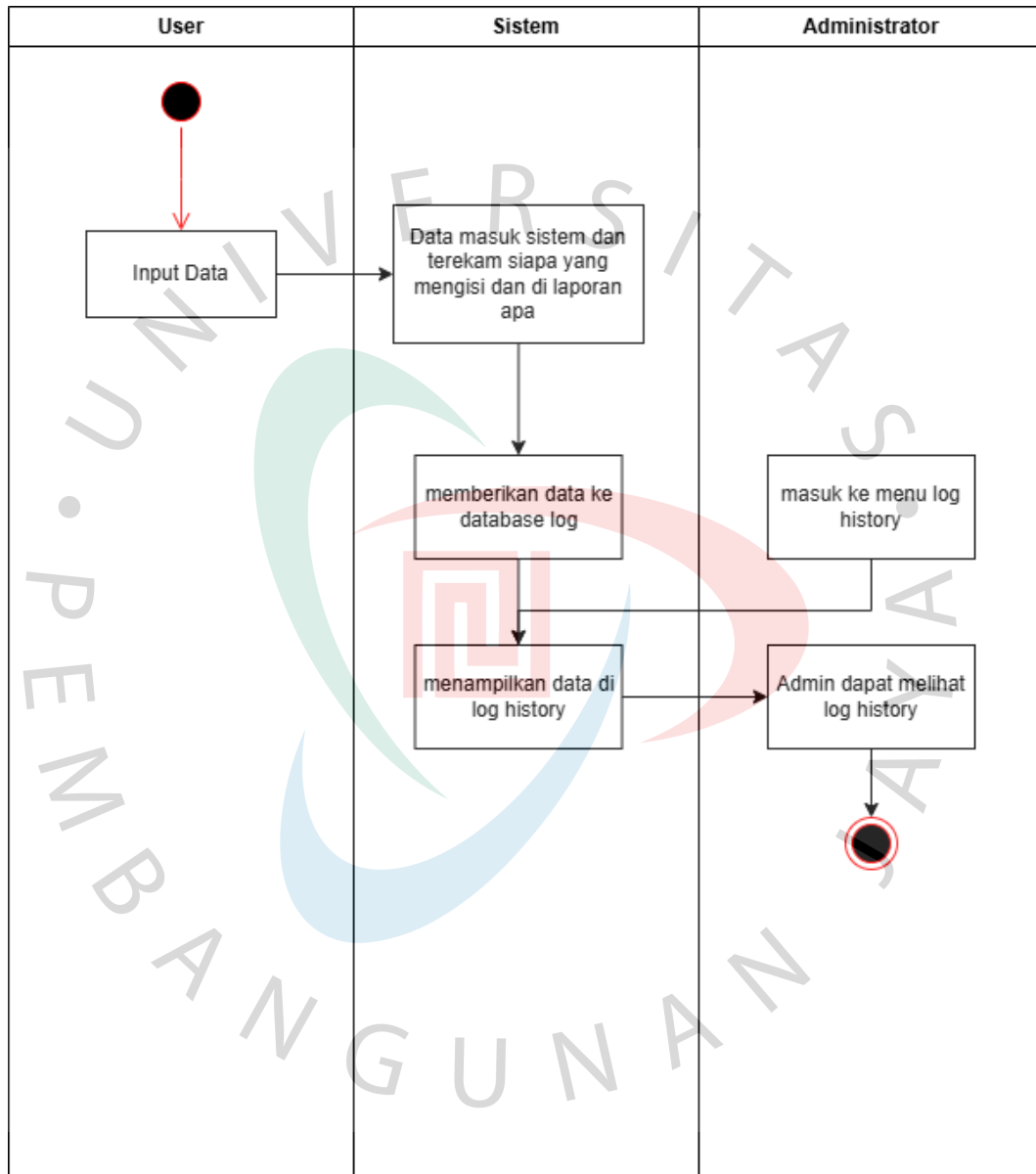
Setelah memilih opsi tersebut, admin akan masuk ke antarmuka Buat Pengguna, di mana mereka dapat mengisi informasi seperti nama, *role*, jabatan, nomor induk karyawan, *password*, dan informasi lainnya. Setelah selesai mengisi, data pengguna baru akan disimpan ke sistem. Selanjutnya, admin akan kembali ke antarmuka *manage user*, di mana data *user* yang baru dibuat akan muncul dalam daftar *user*.



Gambar 3. 8 Diagram Activity Manage User (Edit User)

Gambar 3.8 menunjukkan diagram aktivitas untuk proses mengedit user. Proses ini serupa dengan pembuatan user baru, tetapi admin harus memilih salah

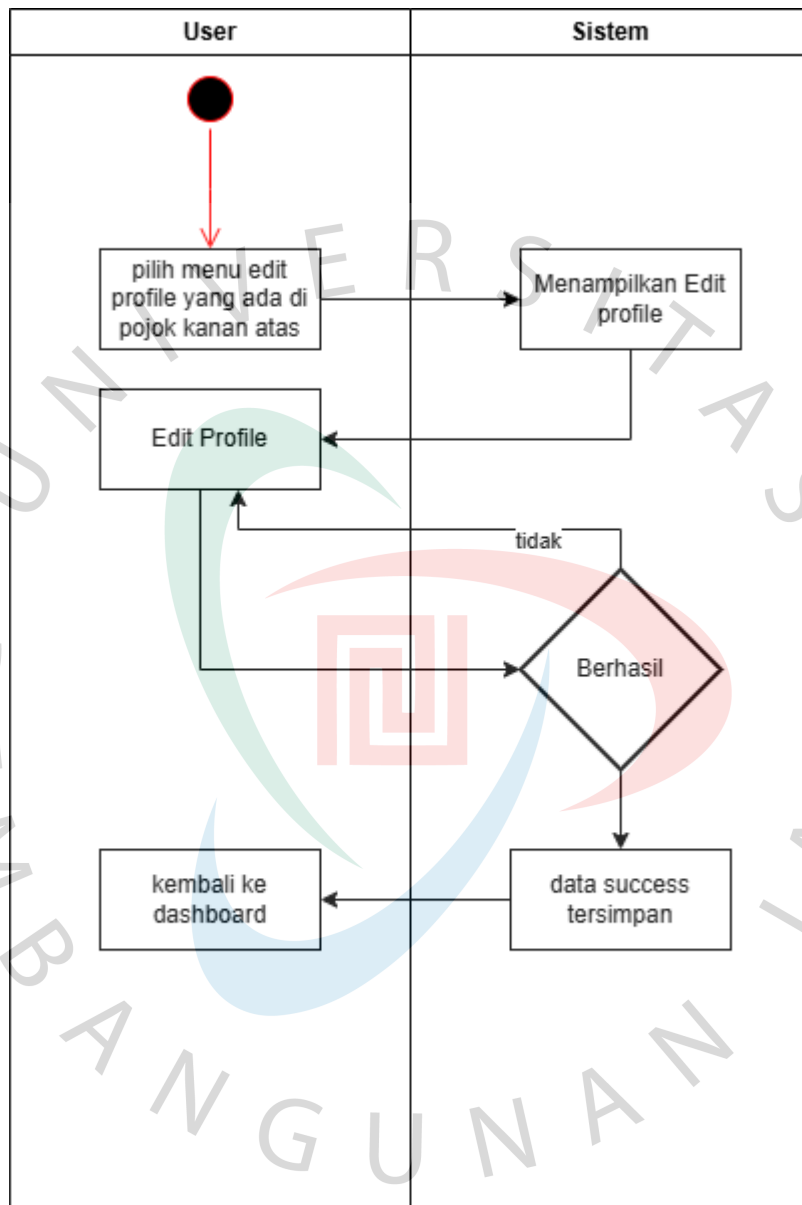
satu *user* yang ingin diperbarui. Setelah memilih, admin akan dibawa ke antarmuka *edit user*. Setelah melakukan perubahan, data *user* akan diperbarui di sistem, dan administrator akan kembali ke antarmuka *manage user*.



**Gambar 3. 9 Diagram Activity Log History**

Gambar 3.9 menunjukkan diagram aktivitas proses *Log History*. Proses ini berjalan bersamaan dengan penginputan data, tetapi dipisahkan dari diagram untuk membuatnya lebih mudah dipahami. Ketika pengguna memasukkan data, proses dimulai. Sistem menyimpan dan secara otomatis merekam aktivitas pengguna, yang kemudian disimpan di database *Log History*. Setelah data

disimpan, admin dapat mengakses dan melihat *log history* melalui menu *Log History* di *dashboard*.



**Gambar 3. 10 Diagram Activity Edit Profile**

Proses mengedit profil digambarkan dalam diagram aktivitas di Gambar 3.10. Proses ini dimulai di *dashboard* dengan mengklik tombol di pojok kanan atas yang menampilkan nama pengguna. Kemudian dropdown menu muncul, dan pengguna memilih opsi Edit Profile. Setelah mengedit profil, data disimpan di sistem, dan pengguna kembali ke *dashboard*.

### 3.2.2 Implementasi

Pada tahap implementasi, praktikan menggunakan perangkat lunak dengan spesifikasi sebagai berikut.

Tabel 3. 2 Tabel Spesifikasi Perangkat Lunak

No	Nama Perangkat Lunak	Spesifikasi Perangkat Lunak
1	Operating System	Windows 10
2	Framework	Laravel 10
3	Database	MySQL
4	IDE	Visual Studio Code

Berdasarkan Tabel 3.2, Windows 10 adalah sistem operasi terbaru yang memungkinkan pengguna melakukan berbagai tugas dengan lebih canggih dan memiliki antarmuka yang berbeda dibandingkan versi sebelumnya, seperti Windows XP dan Windows 7. Studi ini menekankan bahwa sistem operasi sekolah harus diperbarui untuk mendukung belajar mengajar yang lebih baik (Dalimunthe et al., 2020).

Laravel 10 adalah versi terbaru dari *framework* PHP Laravel, yang dirilis untuk membantu pengembang membuat aplikasi web yang lebih mudah dengan menggunakan arsitektur Model-View-Controller (MVC). *Framework* ini memiliki fitur canggih seperti routing, middleware, dan ORM (Eloquent), yang membantu pengembang membuat aplikasi yang terstruktur dengan baik. Laravel 10 berfokus pada kemudahan penggunaan, fleksibilitas desain, dan komunitas yang aktif (Renaldo Prasena & Sama, 2020).

Dalam penelitian ini, kami menggunakan MySQL, salah satu sistem manajemen basis data relasional (RDBMS) yang paling populer, untuk membandingkan kinerja empat basis data yang berbeda. MySQL mendukung berbagai aplikasi web dan sistem informasi dengan kecepatan, fleksibilitas, dan kemudahan. (Kleweka et al., 2020).

Editor teks Visual Studio Code yang sederhana ini disediakan oleh Microsoft dan kompatibel dengan berbagai sistem operasi seperti Linux, Mac, dan Windows. Editor mendukung berbagai bahasa pemrograman seperti JavaScript, TypeScript, dan Node.js, serta bahasa lain melalui plugin yang dapat diunduh dari pasar Visual Studio Code, seperti PHP, C++, C#, Python, dan Java. (Ningsih



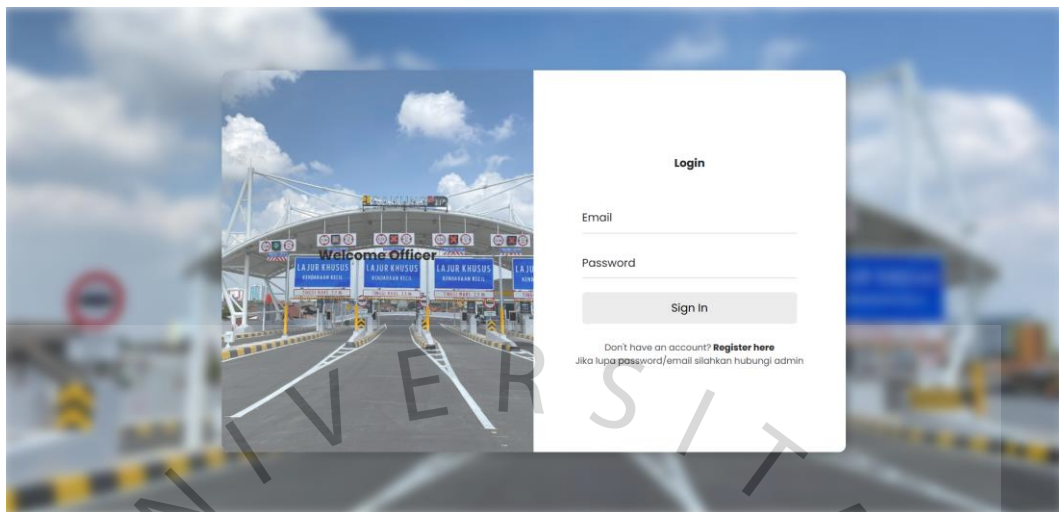
Kherina Surya et al., 2022). Spesifikasi perangkat keras yang digunakan untuk mendukung praktikan pembuatan website adalah sebagai berikut.

**Tabel 3. 3 Tabel Spesifikasi Perangkat Keras**

No	Nama Perangkat Keras	Spesifikasi Perangkat Keras
1	Processor	AMD Ryzen 7 3700U with Radeon Vega Mobile Gfx 2.30 GHz
2	GPU	AMD Radeon™ RX Vega 10 Graphics
3	RAM	16 GB

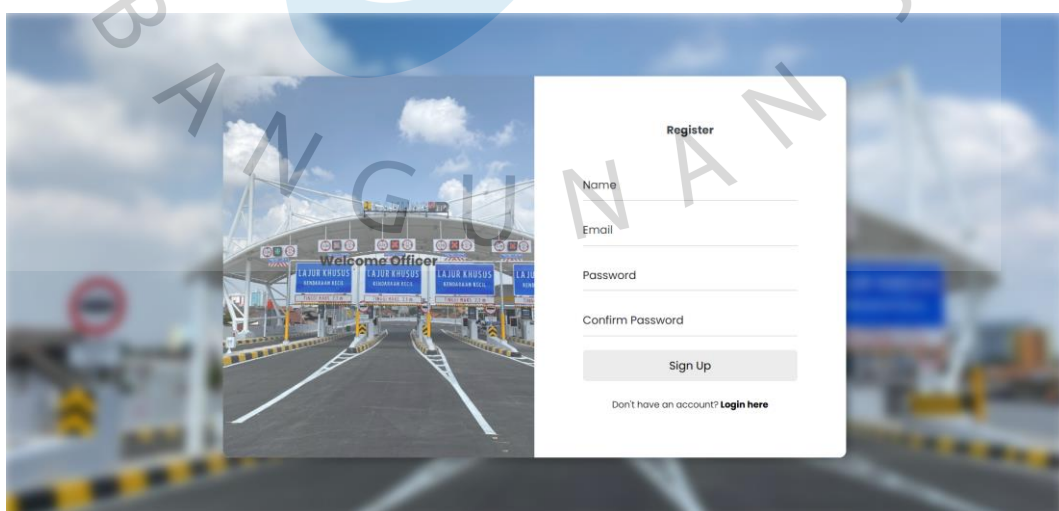
Berdasarkan Tabel 3.2, praktikan dapat dengan mudah mengerjakan pengembangan situs web dengan spesifikasi perangkat yang memadai.

Penggunaan *Framework* Laravel dipilih karena hampir semua website PT Jakarta Tollroad Development menggunakannya. Selain itu, penggunaan *Framework* Laravel bertujuan untuk membuat pengembangan di masa mendatang lebih mudah. Dengan fiturnya, *framework* ini meningkatkan efisiensi pembuatan aplikasi web. Praktikan juga memilih MySQL sebagai basis data karena mudah digunakan dan berintegrasi dengan Laravel.



Gambar 3. 11 Tampilan *Login*

Halaman *login* sistem, yang berfungsi sebagai akses website, digambarkan dalam Gambar 3.11. Pada halaman ini, pengguna diminta untuk mengakses *dashboard* yang sesuai dengan *role* pengguna dengan memasukkan *email* dan *password* pengguna. Halaman *login* ini memiliki validasi untuk mencegah kesalahan input, seperti memasukkan *email* atau *password* yang salah. Selain itu, bagi pengguna yang belum memiliki akun, ada tautan ke halaman registrasi. Namun, sistem ini tidak memiliki fitur "lupa *password*". Pengguna dapat menghubungi kantor atau admin untuk mendapatkan bantuan jika mereka lupa *email* atau *password*.

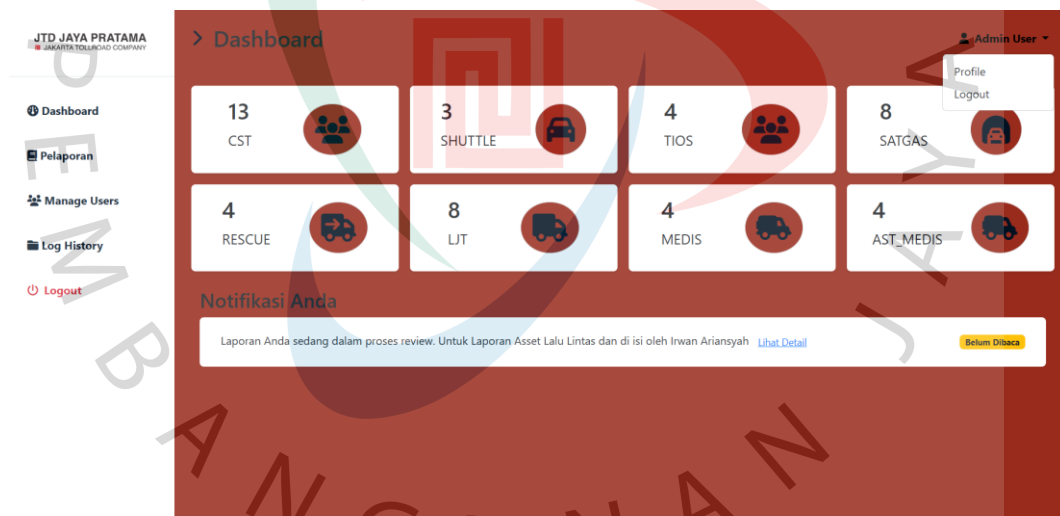


Gambar 3. 12 Tampilan *Register*

Halaman yang dapat digunakan oleh pengguna yang belum memiliki akun ditunjukkan pada Gambar 3.12. Pengguna diminta untuk mengisi *email*, nama, *password*, dan konfirmasi *password* pada halaman ini. Setelah proses registrasi selesai, sistem secara otomatis memberikan peran (*role*) sebagai pengguna kepada pengguna baru.

Halaman ini juga memiliki validasi untuk mencegah kesalahan input selama proses pengisian data, seperti memastikan bahwa *password* cocok dengan *password* dan konfirmasi *password*. Pengguna akan diarahkan ke *dashboard* yang sesuai dengan perannya setelah mereka mendaftar.

Meskipun demikian, halaman ini tidak menyediakan opsi pengaturan peran (*role*) tambahan, seperti admin atau *validator*, karena pendaftaran peran tersebut hanya dapat dilakukan oleh admin. Tujuan dari proses ini adalah untuk menjaga keamanan dan kontrol akses sistem.

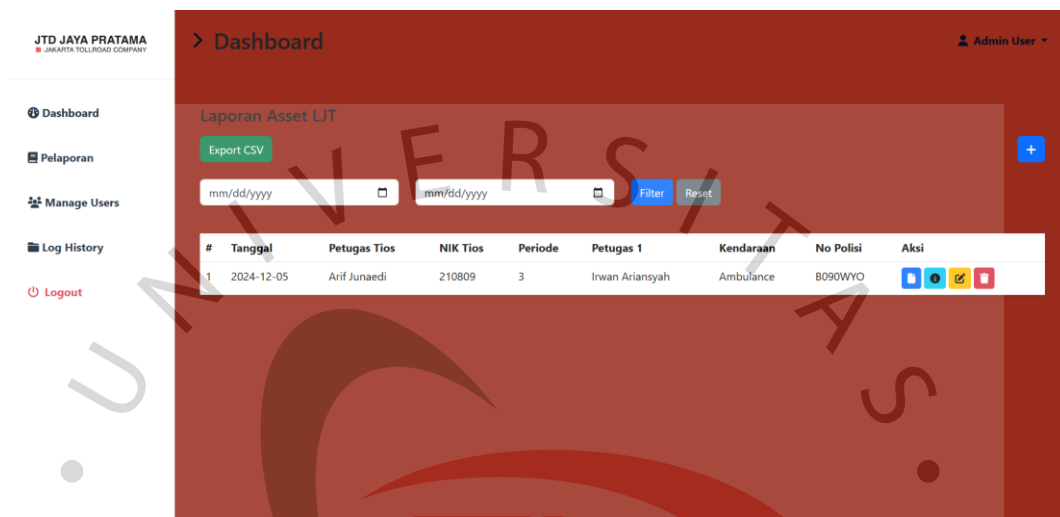


Gambar 3. 13 Tampilan *Dashboard Admin*

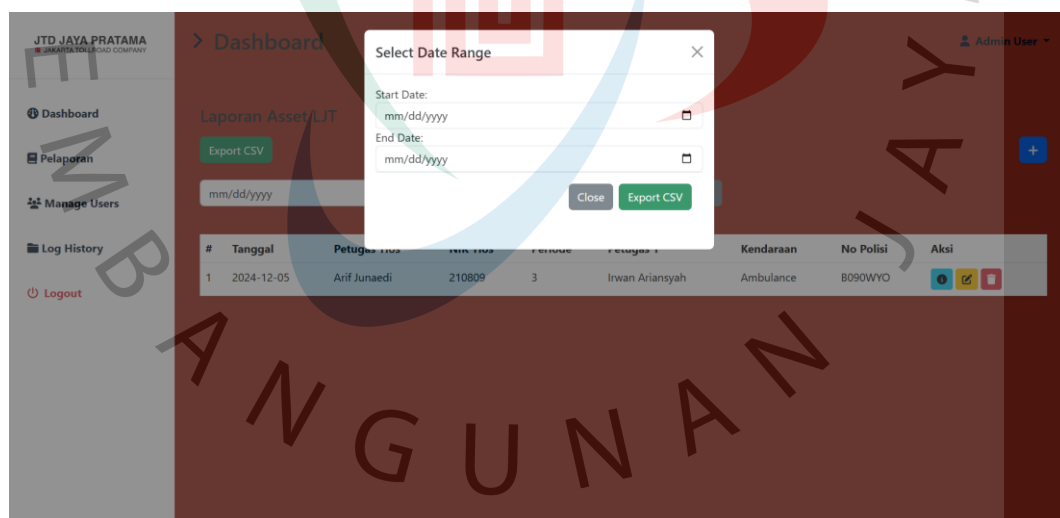
Gambar 3.13 menunjukkan *dashboard* utama admin, yang berfungsi sebagai pusat kontrol sistem. *Dashboard* ini menampilkan informasi penting seperti jumlah pegawai lapangan, notifikasi laporan terbaru, dan status laporan yang memerlukan tindak lanjut, yang dimaksudkan untuk membantu administrator memantau aktivitas sistem dengan lebih baik.

Menu yang berkaitan dengan pengelolaan sistem, seperti menu pelaporan, manajemen pengguna, log catatan, dan *logout*, tersedia di sidebar. Menu-menu ini

memudahkan administrator untuk mengakses berbagai fungsi utama sistem. Selain itu, terdapat menu untuk *logout* dan edit profil (*user login*) di bagian kanan atas *dashboard*, yang membuat lebih mudah bagi administrator untuk melakukan edit mandiri.



Gambar 3. 14 Tampilan List Laporan



Gambar 3. 15 Fitur Export CSV

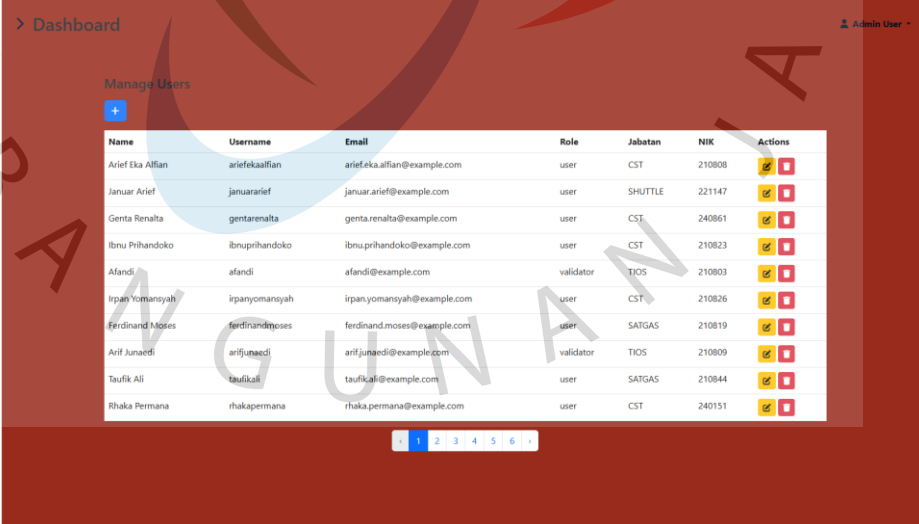
Daftar laporan yang dibuat khusus untuk admin ditampilkan pada Gambar 3.14 dan Gambar 3.15. Tampilan ini memiliki banyak fitur, seperti ekspor CSV, pembuatan laporan baru, filter laporan, tabel laporan, dan ekspor PDF. Selain itu, ada opsi untuk melihat detail laporan, mengeditnya, dan menghapusnya. Dengan fitur-fitur ini, manajer dapat mengelola laporan dengan lebih mudah.





















Untuk meningkatkan produktivitas dan memastikan bahwa data yang relevan dapat diakses dengan mudah, fitur filter laporan pada tampilan ini memungkinkan admin menemukan data dengan lebih cepat berdasarkan rentang waktu tertentu.

Admin memiliki kontrol penuh atas laporan, sebagai pengguna dengan akses tertinggi. Mereka dapat menyetujui (approve) atau menolak (reject) laporan yang masuk, serta mengedit atau menghapus laporan jika diperlukan. Selain itu, manajer memiliki kemampuan untuk membuat laporan baru untuk memenuhi kebutuhan operasional tertentu.

Antarmuka ini memungkinkan administrator untuk mengelola sistem pelaporan dengan lebih efisien, memastikan bahwa semua data disimpan dengan aman dan dapat diakses kapan saja.

- Proses penggunaan fitur export CSV digambarkan pada Gambar 3.16. Proses dimulai dengan menentukan rentang tanggal, memilih tanggal awal dan akhir yang diperlukan. Setelah rentang tanggal ditentukan, sistem akan secara otomatis mengunduh file CSV yang berisi data yang memenuhi kriteria tersebut.




Name	Username	Email	Role	Jabatan	NIK	Actions
Arief Eka Alfian	ariefekaalfian	arief.eka.alfian@example.com	user	CST	210808	 
Januar Arief	januararief	januar.arief@example.com	user	SHUTTLE	221147	 
Genta Renalta	gentarenalta	genta.renalta@example.com	user	CST	240861	 
Ibnu Prihandoko	ibnuprihandoko	ibnu.prihandoko@example.com	user	CST	210823	 
Afandi	afandi	afandi@example.com	validator	TIOS	210803	 
Irpan Yomansyah	irpanyomansyah	irpan.yomansyah@example.com	user	CST	210826	 
Ferdinand Moses	ferdinandmoses	ferdinand.moses@example.com	user	SATGAS	210819	 
Arif Junaedi	arifjunaedi	arif.junaedi@example.com	validator	TIOS	210809	 
Taufik Ali	taufikali	taufik.ali@example.com	user	SATGAS	210844	 
Rhaka Permana	rhakapermana	rhaka.permana@example.com	user	CST	240151	 

**Gambar 3. 16 Tampilan *Manage Users***

Gambar 3.16 menunjukkan tampilan Manajemen Pengguna, yang memungkinkan administrator untuk memantau profil setiap pengguna yang

terdaftar di website ini. Administrator dapat menambahkan pengguna baru, mengubah data pengguna, atau menghapus akun pengguna yang tidak lagi aktif.

Praktikan menambahkan fitur navigasi dan membatasi tampilan menjadi 10 pengguna per halaman. Selain itu, pencarian data lebih mudah dan meningkatkan efisiensi pengelolaan agar data lebih terorganisir.

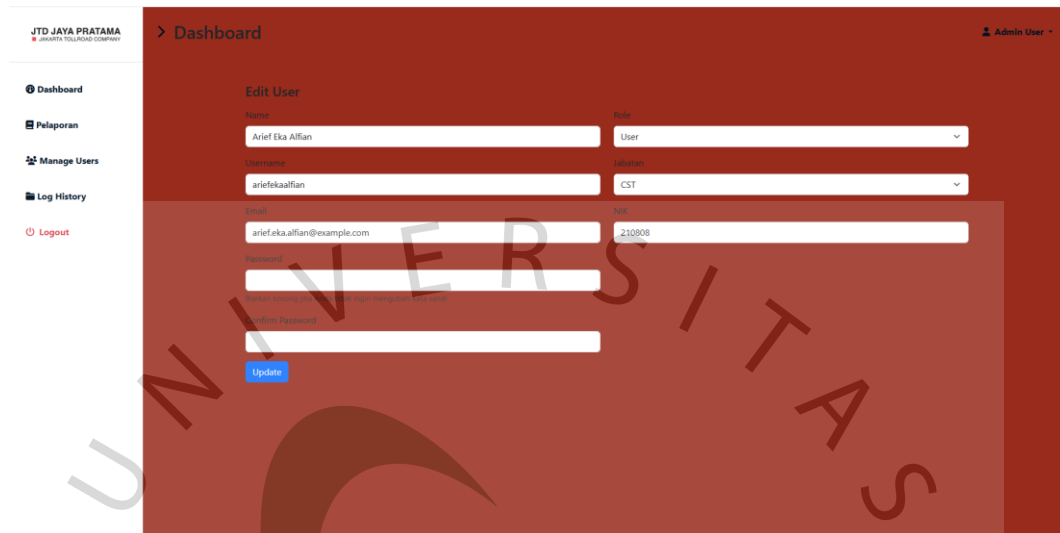
The image shows a web application interface with a dark red theme. On the left is a sidebar menu with options: Dashboard, Pelaporan, Manage Users, Log History, and Logout. The main content area is titled 'Dashboard' and contains a form titled 'Add New User'. The form has two columns of input fields. The first column contains: 'Name' (text input), 'Username' (text input), 'Email' (text input), 'Password' (password input), and 'Confirm Password' (password input). The second column contains: 'Role' (dropdown menu with 'User' selected), 'Job' (dropdown menu with 'SPV\_CST' selected), and 'NIK' (text input). A blue 'Submit' button is located at the bottom left of the form area.

**Gambar 3. 17 Tampilan Create User**

Tampilan *Create User* dapat digunakan untuk menambahkan pengguna baru dengan akses yang diatur oleh admin, seperti yang ditunjukkan pada Gambar 3.17. Kolom isian seperti nama, *role*, *username*, jabatan, *email*, NIK, kata sandi, dan konfirmasi kata sandi dapat ditemukan di halaman ini.

Setelah proses pembuatan selesai, administrator akan secara otomatis diarahkan kembali ke halaman Manajemen Pengguna. Di sana, pengguna yang baru dibuat akan ditampilkan dalam daftar dan dapat langsung menggunakan akun

mereka untuk *login* ke sistem. Fitur ini dimaksudkan untuk mempermudah pengelolaan akun baru sambil mempertahankan struktur dan keamanan data.



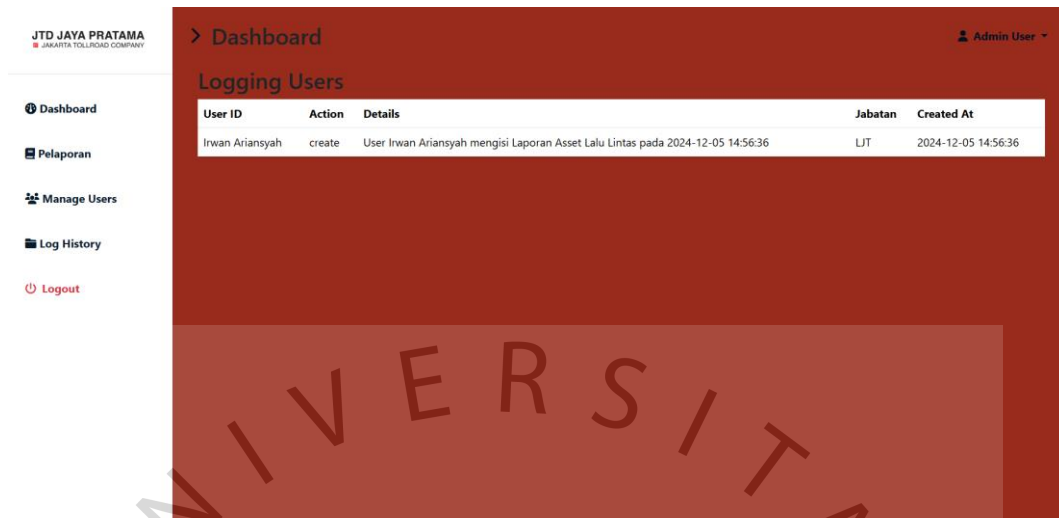
The screenshot shows a web dashboard with a dark red theme. On the left is a sidebar menu with options: Dashboard, Pelaporan, Manage Users, Log History, and Logout. The main content area is titled 'Dashboard' and contains an 'Edit User' form. The form has the following fields: Name (Arief Eka Alfian), Role (User), Username (ariefekaalfian), Password (CST), Email (ariefekaalfian@example.com), and Phone (210808). There is an 'Update' button at the bottom of the form. A large watermark 'UNIVERSITAS' is visible across the center of the image.

**Gambar 3. 18 Tampilan Edit User**

Antarmuka *Edit User* memungkinkan administrator mengubah peran pengguna, seperti dari pengguna biasa menjadi *validator* atau admin, seperti yang ditunjukkan pada Gambar 3.18. Selain itu, antarmuka ini juga digunakan untuk memperbarui jabatan pengguna yang dipromosikan.

Proses pengeditan dilakukan untuk memastikan bahwa data pengguna sesuai dan akurat. Antarmuka ini juga digunakan untuk memperbaiki data pengguna yang salah yang dimasukkan saat pendaftaran melalui halaman *Register*. Setelah pengeditan dan penyimpanan data selesai dilakukan oleh administrator, perubahan akan secara otomatis tercatat di sistem.

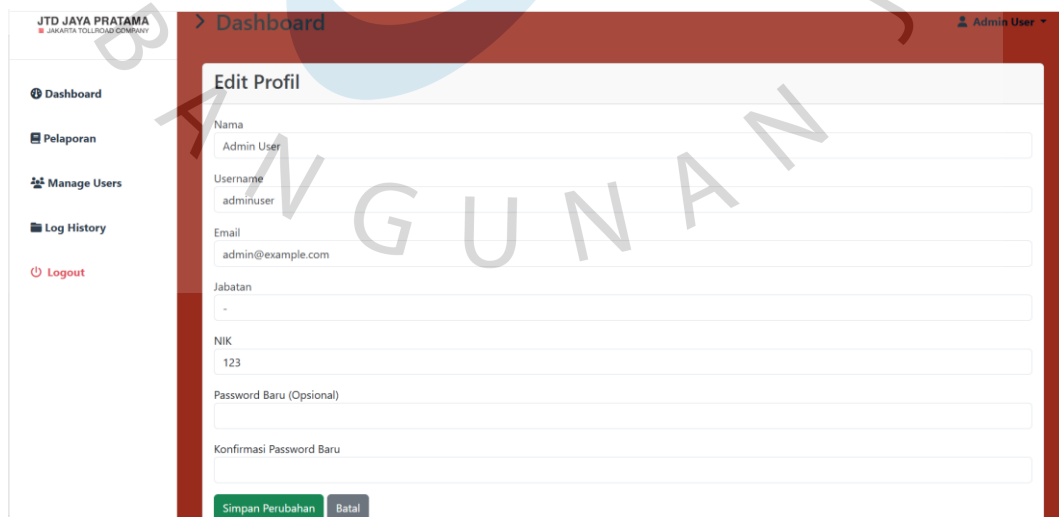
Setelah itu, administrator akan diarahkan kembali ke halaman Manajemen Pengguna. Di sana, administrator akan melihat pengguna yang telah diubah. Ini adalah fitur yang memudahkan administrator untuk memastikan bahwa data pengguna selalu akurat dan terkini.



**Gambar 3. 19 Tampilan *Log History***

Antarmuka *Log History* yang dimaksudkan untuk membantu manajer memantau interaksi pengguna selama mengakses website ditampilkan pada Gambar 3.19. Informasi penting seperti nama pengguna, jenis tindakan, detail tindakan, jabatan pengguna, dan tanggal kejadian disajikan di tampilan ini.

Fitur ini sangat bermanfaat untuk melacak aktivitas pengguna secara menyeluruh dan memastikan keamanan dan akurasi data. Dengan adanya *Log History*, administrator dapat dengan mudah menemukan perubahan atau aktivitas yang mencurigakan untuk ditindaklanjuti jika diperlukan.

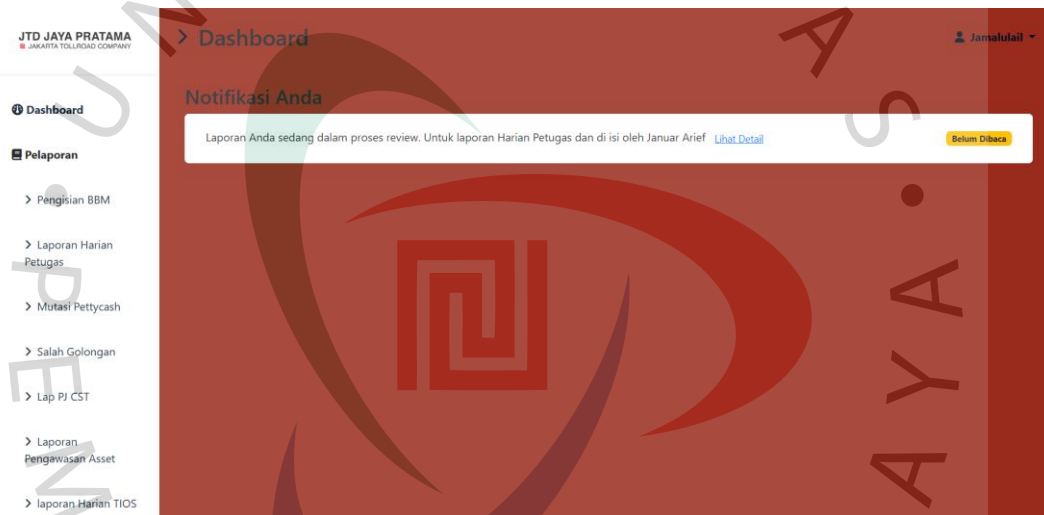


**Gambar 3. 20 Tampilan *Edit Profile***



Antarmuka untuk mengedit profil terdapat pada Gambar 3.21. Semua jenis pengguna, termasuk pengguna admin, *validator*, dan pengguna biasa, dapat mengakses fitur ini. Pengguna dapat memperbarui informasi penting seperti jabatan, nama, *username*, *email*, dan NIK di *menu* ini, serta mengatur ulang *password* baru jika diperlukan.

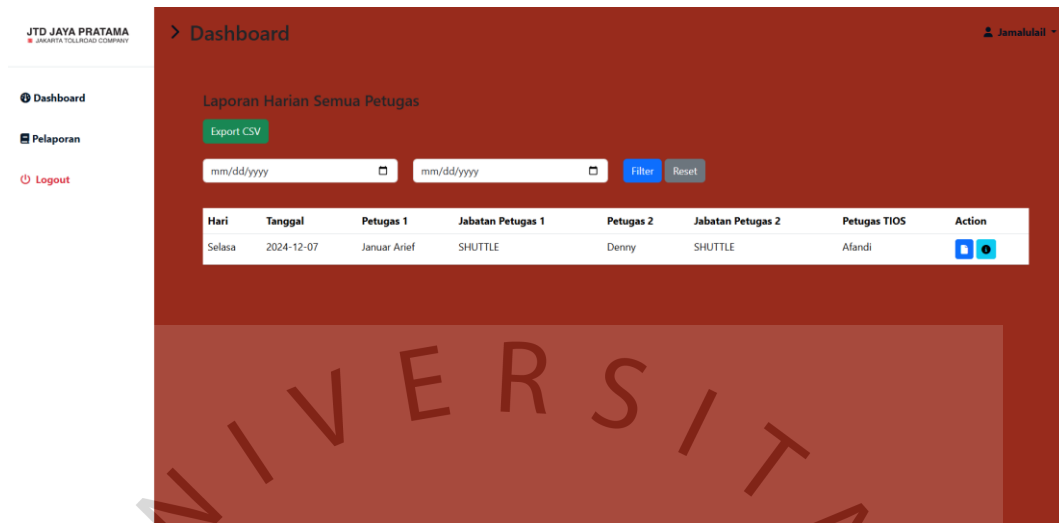
Setelah proses pengeditan dan input data selesai, sistem akan secara otomatis memperbarui data. Setelah itu selesai, pengguna akan dikembalikan ke halaman *dashboard*. Fitur ini dirancang untuk memungkinkan pengguna mengelola data pribadi mereka sendiri.



Gambar 3. 21 Tampilan *Dashboard Validator*

Untuk mempermudah proses pengecekan dan validasi laporan, *dashboard* khusus untuk *validator* ditunjukkan pada Gambar 3.21. Setiap laporan yang diajukan oleh pengguna di *dashboard* ini akan langsung muncul pada bagian notifikasi ini, sehingga *validator* tidak perlu mencari laporan secara manual. Bagian notifikasi ini juga menampilkan laporan yang memerlukan tindakan penting, yaitu proses *approve* atau *reject*.

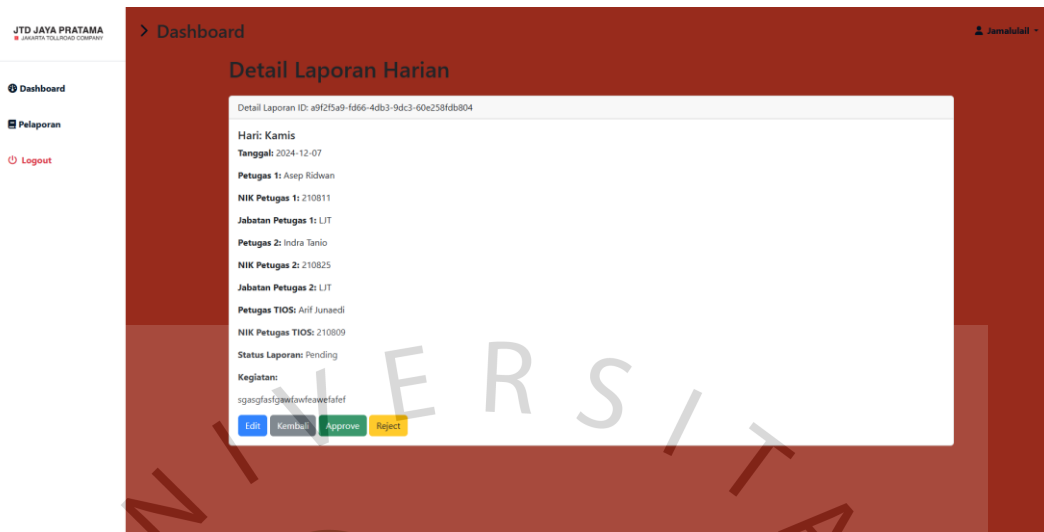
Untuk melanjutkan ke halaman detail laporan, *validator* hanya perlu mengklik tombol "Lihat Detail" pada notifikasi. Notifikasi tersebut akan secara otomatis hilang dari daftar setelah diklik, menandakan bahwa laporan tersebut telah dibaca dan ditindaklanjuti. Fitur ini dimaksudkan untuk meningkatkan produktivitas dan memastikan bahwa *validator* dapat segera memproses laporan yang masuk.



**Gambar 3. 22** Tampilan List Laporan *Validator*

Daftar laporan yang telah dikirim untuk divalidasi oleh *validator* ditampilkan pada Gambar 3.22. Sesuai dengan peran *validator*, tampilan ini dirancang dengan fitur terbatas. Hanya dengan menggunakan *validator*, hanya dapat mengekspor data dalam format CSV dan PDF, melakukan filter sesuai dengan kriteria tertentu, dan melihat detail laporan.

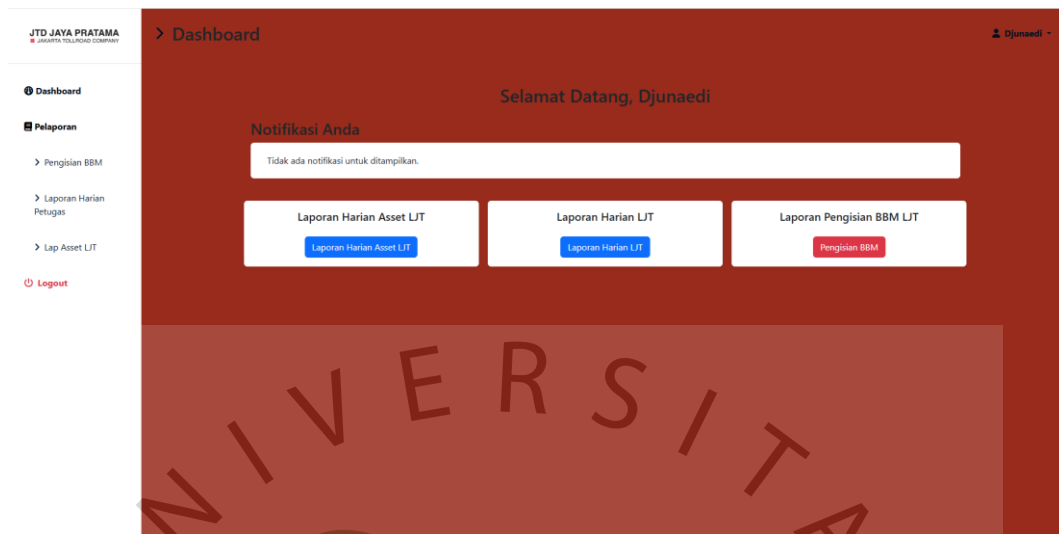
Berbeda dengan tampilan admin, *validator* tidak dapat membuat atau menghapus laporan yang sudah ada. Tampilan ini dibuat untuk menjaga pembagian tanggung jawab antara admin, *validator*, dan *user* biasa sehingga proses pelaporan dan validasi menjadi lebih terstruktur dan efisien. Tampilan ini juga membantu *validator* fokus pada tugas utamanya, yaitu memverifikasi dan meninjau laporan yang masuk.



**Gambar 3. 23 Tampilan Detail Laporan Validator**

Gambar 3.23 menunjukkan antarmuka detail laporan yang menampung semua data yang dimasukkan oleh pengguna biasa atau karyawan lapangan. Data tidak sama dalam setiap laporan. Tampilan ini dibuat agar *validator* dapat melihat semua informasi yang diperlukan untuk memvalidasi laporan secara menyeluruh. *Validator* dapat melihat setiap elemen laporan, termasuk detail khusus yang dimasukkan oleh pengguna.

Selain itu, tampilan ini menawarkan fitur untuk melakukan tindakan seperti mengedit, menolak, atau memverifikasi laporan. *Validator* juga dapat memilih opsi "kembali" untuk kembali ke daftar laporan tanpa melakukan perubahan. Dengan fitur ini, proses validasi laporan menjadi lebih terorganisir dan efisien, memastikan bahwa laporan yang diterima sesuai dengan standar dan persyaratan perusahaan.

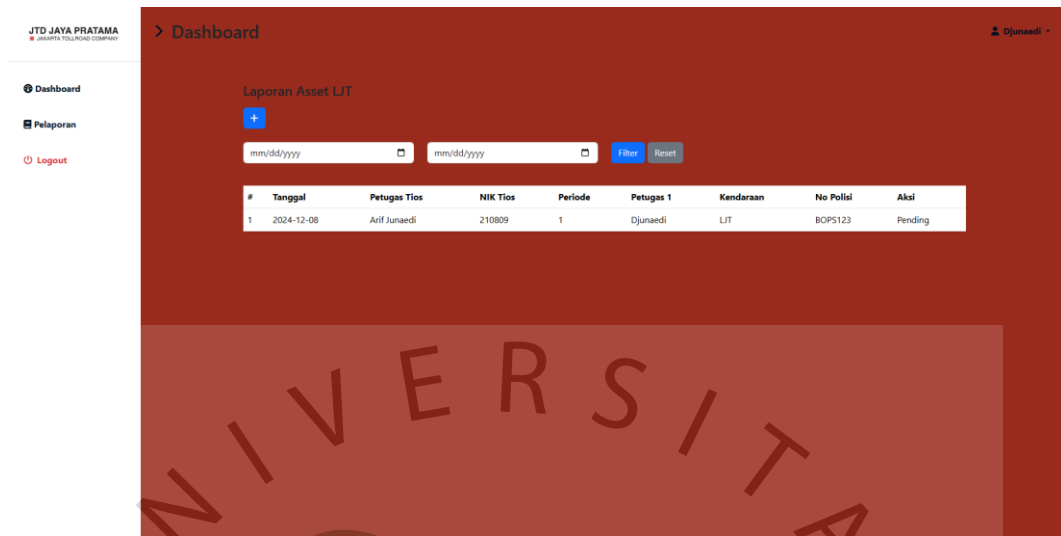


**Gambar 3. 24 Tampilan Dashborad User**

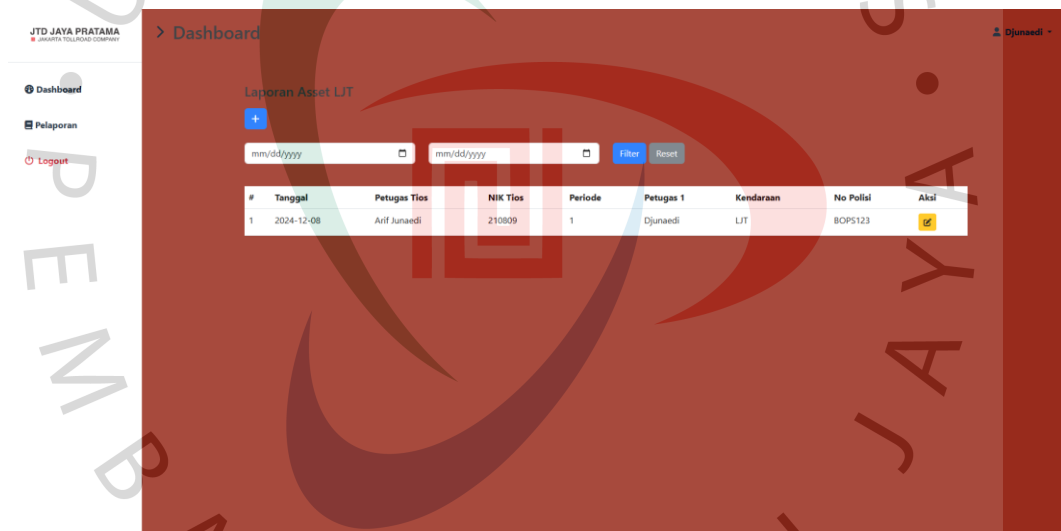
Gambar 3.24 merupakan *Dashboard* yang ditujukan untuk *user* karyawan. Tampilan ini mencakup fitur-fitur yang berguna termasuk notifikasi, yang memberi tahu pengguna tentang status laporan yang telah mereka siapkan, seperti apakah laporan tersebut telah diterima atau ditolak oleh *validator*. Jika laporan ditolak, pengguna akan diminta untuk mengubah dan memperbaruinya sehingga *validator* dapat memvalidasinya kembali.

Antarmuka ini juga memudahkan navigasi dengan menyertakan hyperlink "Lihat Detail" di setiap notifikasi. Hal ini memungkinkan pengguna untuk menavigasi langsung ke halaman edit laporan tanpa membuka daftar laporan terlebih dahulu. Hal itu, memungkinkan pengguna untuk menyempurnakan laporan mereka dengan mudah.

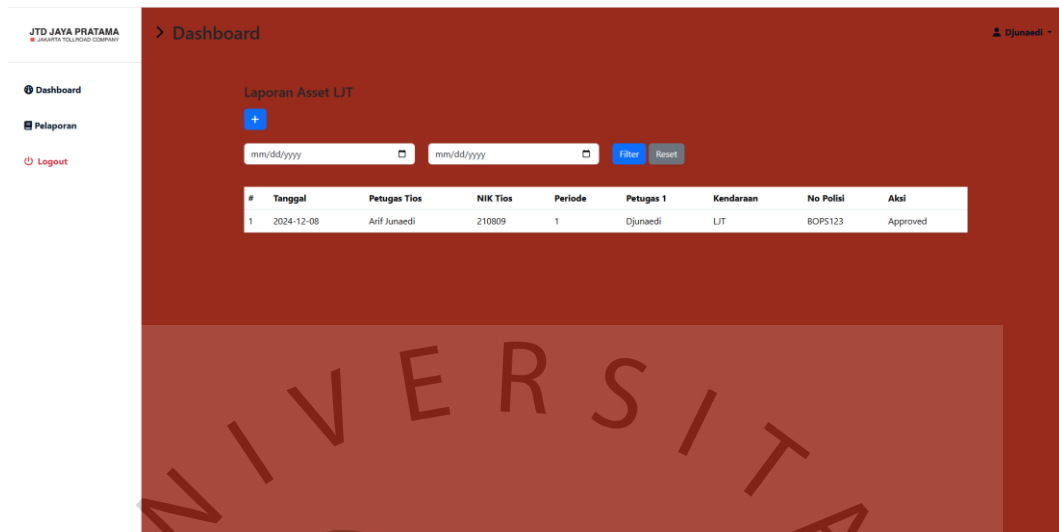
*Dashboard* juga menampilkan daftar laporan yang dapat diisi oleh *user* umum. Tergantung pada posisi atau jabatan mereka di kantor, masing-masing pengguna memiliki jumlah laporan yang berbeda. Sebagai contoh, seorang karyawan di layanan LJT (Lalu Lintas Jalan Tol) hanya dapat mengisi tiga jenis laporan. Ini juga ditunjukkan pada sidebar, yang menunjukkan bahwa hanya tiga laporan yang dapat diakses. Dengan desain ini, laporan dapat diakses sesuai dengan kebutuhan dan tanggung jawab sesuai perusahaan.



Gambar 3. 25 Tampilan List Laporan User



Gambar 3. 26 Tampilan List Laporan User (Kondisi Reject)



**Gambar 3. 27 Tampilan List Laporan User (Kondisi approve)**

Daftar laporan yang dibuat khusus untuk pengguna umum dapat dilihat pada Gambar 3.25 hingga Gambar 3.27. Tampilan ini memiliki tabel daftar laporan, fitur filter tabel, dan tombol untuk membuat laporan baru. Dalam situasi seperti ini, status laporan masih berada dalam tahap *pending*, yang menunjukkan bahwa *validator* belum memverifikasinya.

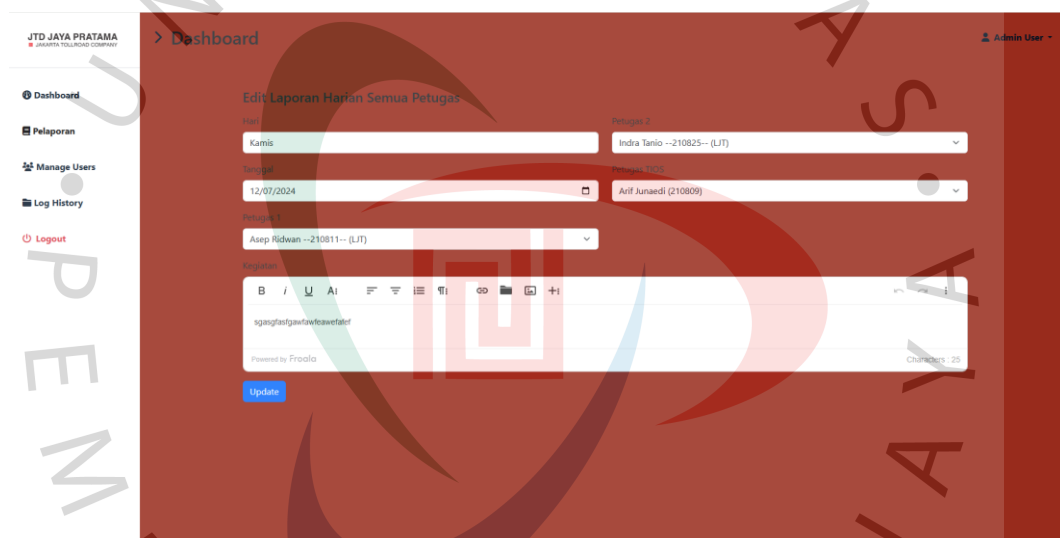
Seperti yang ditunjukkan pada Gambar 3.27 dan 3.28, jika laporan telah divalidasi, statusnya akan berubah sesuai dengan hasil validasi. Laporan yang diterima akan memiliki status *approved*, sedangkan laporan yang ditolak akan memiliki ikon edit. Pengguna dapat langsung memperbaiki laporan mereka tanpa kembali ke halaman lain dengan ikon ini. Setelah Anda mengklik "Lihat Detail" di notifikasi, fitur ini menjadi opsi alternatif.

Hal yang perlu diperhatikan adalah bahwa tabel ini dirancang untuk menampilkan hanya laporan yang dibuat oleh *user* yang sedang *login*. setiap *user* memiliki tabel terpisah yang berisi laporan yang mereka buat sendiri, tanpa data dari *user* lain. Desain ini menjaga privasi dan memudahkan pengguna mengelola laporan mereka.



Hanya *user* umum dan admin yang dapat membuat laporan ini. Setelah laporan selesai diisi, data akan otomatis masuk ke sistem, dan pengguna akan diarahkan kembali ke halaman list laporan. Di sana, laporan terbaru akan langsung muncul dan memiliki status *pending*, yang menunjukkan bahwa laporan sedang menunggu validasi oleh *validator*.

Selain meminimalkan kesalahan input, antarmuka ini dirancang untuk menjadi mudah digunakan dan mengurangi kesalahan input. Dengan berbagai tipe field dan navigasi yang jelas, proses pelaporan menjadi lebih terorganisir dan efisien.



Gambar 3. 30 Tampilan Edit Laporan

Tampilan edit yang digunakan untuk memperbarui laporan yang telah ditolak oleh *validator* dapat dilihat pada Gambar 3.30. Tampilan ini dibuat untuk digunakan oleh admin maupun *user* umum.

Bagi *user* umum, fitur ini ditujukan untuk merevisi laporan yang telah ditolak yang didasarkan pada input *validator*. Sementara itu, administrator dapat menggunakan tampilan ini untuk memperbarui laporan yang sangat penting atau memerlukan tindakan segera.

Tampilan edit laporan memiliki field yang mirip dengan tampilan *create* laporan agar pengguna tetap familiar dengan proses input data. Setelah diperbarui, laporan akan kembali ke sistem dan muncul di list laporan. Laporan yang diedit oleh *user* umum akan melalui proses validasi ulang oleh *validator*,



sedangkan laporan yang diedit oleh administrator tidak memerlukan validasi lebih lanjut karena admin memiliki wewenang untuk langsung memvalidasi laporan. Dengan desain ini, proses revisi laporan menjadi lebih sistematis dan efisien sambil mengurangi kesalahan input yang mungkin terjadi.

### 3.2.3 Pengujian

Pengujian *Black-Box* adalah cara untuk menguji perangkat lunak tanpa melihat struktur kode internalnya. Tujuan utama pengujian ini adalah memastikan perangkat lunak beroperasi sesuai dengan spesifikasi dan memenuhi kebutuhan pengguna. Proses pengujian dilakukan dengan merancang kasus uji berdasarkan persyaratan aplikasi dan menemukan kesalahan seperti fungsi yang hilang, masalah antarmuka, atau kesalahan struktur data. Pengujian ini sangat penting untuk menilai kualitas, keandalan, dan keefektifan aplikasi (Efendi, 2021). Tabel pengujian kotak hitam dapat ditemukan di table 3.4.

Tabel 3. 4 Tabel Pengujian Kotak Hitam

No	Fungsi yang diuji	Skenario	Input	Hasil yang diharapkan	status
1	<i>Login</i>	Input <i>email</i> dan <i>password</i> valid	<i>Email:</i> valid@example.com, <i>Password:</i> password123	Pengguna diarahkan ke <i>dashboard</i> sesuai <i>role</i>	Berhasil
		Input <i>email/password</i> salah	<i>Email:</i> invalid@example.com, <i>Password:</i> wrongpass	Sistem menampilkan pesan " <i>Email</i> atau <i>password</i> salah"	Berhasil
2	Registrasi	Input data valid	Nama, <i>Email</i> , <i>Password</i> , Konfirmasi <i>Password</i>	Akun baru dibuat, <i>role</i> default: <i>user</i> , diarahkan ke <i>login</i>	Berhasil
		Input <i>password</i> tidak cocok	<i>Password:</i> pass123, Konfirmasi <i>Password:</i> pass321	Sistem menampilkan pesan " <i>Konfirmasi password</i> tidak cocok"	Berhasil
3	<i>Dashboard Admin</i>	Memuat data <i>dashboard</i>	Tidak ada	<i>Dashboard</i> menampilkan statistik terbaru	Berhasil
4	<i>Dashboard Validator</i>	Memuat data <i>dashboard</i>	Tidak ada	<i>Dashboard</i> menampilkan	Berhasil

				statistik terbaru	
5	<i>Dashboard User</i>	Memuat data <i>dashboard</i>	Tidak ada	<i>Dashboard</i> menampilkan statistik terbaru	Berhasil
6	Membuat Laporan	Input semua field laporan	Data lengkap: teks, dropdown, gambar	Laporan disimpan dengan status " <i>Pending</i> "	Berhasil
7		Input data tidak lengkap	Field wajib kosong	Sistem menampilkan pesan error " <i>Field tidak boleh kosong</i> "	Berhasil
8		Input Petugas hanya 1	Tidak mengisi Petugas 2	Laporan berhasil disimpan tanpa mengisi petugas 2	Berhasil
9	Validasi Laporan	<i>Validator approve</i> laporan	Status laporan: <i>Pending</i>	Status laporan berubah menjadi " <i>Approved</i> "	Berhasil
10		<i>Validator reject</i> laporan	Status laporan: <i>Pending</i>	Status laporan berubah menjadi " <i>Rejected</i> ", <i>user</i> menerima notifikasi	Berhasil
11	<i>Manage Users</i>	Tambah pengguna baru	Data pengguna valid	Pengguna baru ditambahkan ke sistem	Berhasil
12		Tambah pengguna dengan <i>email</i> sudah digunakan	<i>Email:</i> used@example.com	Sistem menampilkan pesan error " <i>Email sudah terdaftar</i> "	Berhasil
13		Input data tidak lengkap	Field wajib kosong	Sistem menampilkan pesan error " <i>Field tidak boleh kosong</i> "	Berhasil
14	Edit Profil	Perbarui data profil	Data baru: nama, <i>email</i> , <i>password</i>	Profil berhasil diperbarui,	Berhasil

				diarahkan ke <i>dashboard</i>	
15	Ekspor Data	Ekspor CSV untuk rentang tanggal	Tanggal: 01/01/2024 - 31/01/2024	File CSV berhasil diunduh	Berhasil
16		Ekspor PDF untuk laporan tertentu	Pilih laporan	File PDF berhasil diunduh	Berhasil
17	Notifikasi	Klik tombol Lihat Detail dengan kondisi berhasil di <i>approve</i>	Tidak ada	Masuk ke dalam tampilan detail	Berhasil
18		Klik tombol Lihat Detail dengan kondisi berhasil di <i>reject</i>	Tidak ada	Masuk ke dalam tampilan edit	Berhasil
19		Klik tombol Lihat Detail dengan kondisi <i>pending</i>	Tidak ada	Masuk ke dalam tampilan detail	Berhasil
20	<i>Logout</i>	Klik tombol <i>Logout</i>	Tidak ada	Sesi berakhir, diarahkan ke halaman <i>login</i>	Berhasil

### 3.3 Kendala yang Dihadapi

Selama menjalani kerja profesi di PT Jakarta Tollroad Development, praktikan menghadapi beberapa kendala sebagai berikut.

1. Kurangnya pengetahuan tentang prosedur operasional jalan tol. Sangat sering menemukan istilah baru di jalan raya, seperti ALPR, TIOS, dan lainnya.
2. Kurangnya pemahaman mengenai teknologi sistem tol, seperti sensor, mobile reader, dan gardu.
3. Praktikan membutuhkan waktu lebih lama untuk mempelajari fitur *approve* dan *reject*, atau yang dikenal sebagai ticketing system, dan menerapkannya dalam Laravel.
4. Praktikan membutuhkan waktu lebih lama untuk memahami materi karena ada perbedaan antara pekerjaan dan materi yang dipelajari di kampus.

### 3.4 Cara Mengatasi Kendala

Selama magang di PT Jakarta Tollroad Development, berikut adalah cara-cara praktis untuk mengatasi masalah:

1. Berkonsultasi dengan pembimbing tentang prosedur operasional dan istilah yang sering digunakan saat bekerja di PT Jakarta Tollroad Development.
2. Praktikan berdiskusi dengan pembimbing untuk mendapatkan penjelasan tentang teknologi sistem tol.
3. Memanfaatkan dokumentasi dan forum online serta berdiskusi dengan pembimbing tentang sistem ticketing yang sedang dibuat.
4. berdiskusi dengan karyawan PT Jakarta Tollroad Development dan mencari informasi tambahan tentang pekerjaan jalan tol.

### **3.5 Pembelajaran yang Diperoleh dari Kerja Profesi**

Selama menjalani kerja profesi di PT Jakarta Tollroad Development, praktikan memperoleh berbagai pembelajaran yang berharga, di antaranya:

1. Pemahaman tentang Operasional Jalan Tol, dengan memahami prosedur operasional jalan tol, yang mencakup teknologi dan sistem seperti ALPR dan TIOS, pengetahuan praktikan tentang industri jalan tol dan standar operasional yang diterapkan.
2. Pengembangan Kemampuan Teknis dalam Sistem Ticketing. Praktikan belajar tentang konsep dan implementasi sistem ticketing, terutama fitur *approve* dan *reject*. Selama proses ini, mereka belajar bagaimana menggunakan Laravel untuk menerapkan logika ticketing system dengan efektif.
3. Peningkatan Keterampilan dalam Teknologi Tol. Praktikan mempelajari berbagai teknologi yang digunakan dalam sistem jalan tol, seperti sensor, pembaca ponsel, dan sistem gardu. Pengetahuan ini membantu mereka memahami lebih baik bagaimana teknologi digunakan dalam operasi jalan tol.
4. Pengembangan Soft Skill dalam Kolaborasi dan Pemecahan Masalah. Praktikan banyak berinteraksi dengan pembimbing dan karyawan lain ketika menghadapi tantangan operasi dan teknis. Ini meningkatkan kemampuan mereka untuk berkomunikasi, bekerja sama, dan memecahkan masalah dengan efektif di tempat kerja.