

APLIKASI PENGAJUAN UJI PRODUK PADA BALAI STANDARISASI DAN PELAYANAN JASA INDUSTRI MEDAN MENGGUNAKAN METODE FAST

Muhammad Rifky Farid, Marina Elsera, Nurjamiyah

Sistem Informasi, Universitas Harapan Medan

Jalan HM. Joni No.70 C, Teladan Baru, Kec. Medan Kota, Kota Medan, Sumatera Utara 20216

faridsx8@gmail.com

ABSTRAK

Penelitian ini bertujuan untuk merancang dan mengimplementasikan aplikasi pengajuan uji produk di Balai Standarisasi dan Pelayanan Jasa Industri (BSPJI) Medan menggunakan metode *FAST (Framework for the Application of System Thinking)*. Aplikasi berbasis web ini dikembangkan untuk mengatasi berbagai kendala dalam proses manual, seperti keterlambatan, kesalahan pengisian data, dan kurangnya transparansi. Dengan aplikasi ini, pelaku industri dapat mengajukan uji produk secara online dengan lebih cepat, efisien, dan terstruktur. Sistem ini memungkinkan pelanggan mengajukan permohonan, yang kemudian diproses oleh admin untuk menentukan tarif. Tarif yang telah disetujui oleh pelanggan akan dilanjutkan ke proses penugasan kalibrasi oleh petugas terkait. Setiap akun dalam sistem memiliki tugas yang terorganisasi, sehingga alur kerja menjadi lebih jelas dan efisien. Pengembangan sistem mengikuti tahapan metode FAST, mulai dari investigasi awal, analisis masalah, perancangan kebutuhan, hingga konstruksi dan pengujian aplikasi. Aplikasi ini mengotomatisasi tahapan penting, seperti pemeriksaan awal dokumen, pengujian produk, dan evaluasi hasil uji. Fitur pelacakan status pengajuan secara real-time meningkatkan transparansi proses bagi pelaku industri. Diharapkan, implementasi sistem ini mampu meningkatkan efisiensi operasional di BSPJI Medan serta memberikan kepercayaan lebih terhadap produk yang telah diuji sesuai standar.

Kata kunci: Pengujian, Kalibrasi, Produk FAST

1. PENDAHULUAN

Dalam era globalisasi dan kompetisi pasar yang semakin ketat, kualitas produk menjadi faktor kunci yang menentukan daya saing industri. Badan Standarisasi dan Kebijakan Jasa Industri berperan penting dalam memastikan bahwa produk yang beredar di pasar memenuhi standar kualitas yang telah ditetapkan. Proses pengajuan uji produk yang efektif dan efisien sangat diperlukan untuk mendukung industri dalam mencapai standar kualitas tersebut. Balai Standarisasi dan Pelayanan Jasa Industri Medan (BSPJI Medan) adalah sebuah unit pelaksana teknis yang berada ke dalam Badan Standarisasi dan Kebijakan Jasa Industri, Kementerian Perindustrian. [1]

Sebagai unit kerja industri BSPJI Medan mempunyai tugas pokok untuk melakukan kegiatan standarisasi, serta sertifikasi dalam bidang industri. Tugas pokok ini sesuai dengan kebijakan teknis yang dikeluarkan oleh Kepala Badan Standarisasi dan Kebijakan Jasa Industri, Kementerian Perindustrian. Pentingnya peran BSPJI Medan ini adalah memastikan bahwa prosedur sistem dan layanan teknis yang diberikan, khususnya dalam bidang pengujian laboratorium sesuai dengan standar standar yang berlaku. Layanan pengujian di BSPJI Medan antara lain pengujian kimia, fisika, dan mikrobiologi pada bahan dan produk.

Proses pengajuan uji produk di Badan Standarisasi dan Kebijakan Jasa Industri melibatkan beberapa tahapan utama. Tahapan ini dimulai dengan pengajuan permohonan, di mana pelaku industri mengisi formulir permohonan dan melampirkan dokumen pendukung. Setelah itu, dilakukan

pemeriksaan awal terhadap kelengkapan dokumen oleh petugas. Produk yang diajukan kemudian diuji di laboratorium sesuai dengan standar yang berlaku, dan hasil uji dievaluasi oleh tim ahli untuk memastikan kesesuaian dengan standar. Jika produk memenuhi kriteria, akan diterbitkan sertifikat yang menyatakan bahwa produk tersebut telah lulus uji.

Namun, proses manual dalam pengajuan uji produk sering menghadirkan sejumlah tantangan. Beberapa masalah yang muncul meliputi waktu proses yang lama, potensi kesalahan dalam pengisian data, efisiensi yang rendah, kesulitan dalam melacak dokumen pengajuan, serta kurangnya transparansi yang menghambat pelaku industri dalam memantau status pengajuan secara real-time. Untuk mengatasi kendala ini, diperlukan solusi inovatif berupa perancangan aplikasi pengajuan uji produk berbasis web menggunakan metode *FAST (Framework for The Application of System Thinking)*. Aplikasi ini menawarkan berbagai keunggulan, seperti kemudahan pengajuan secara online, otomatisasi proses pemeriksaan awal, pengujian, dan evaluasi hasil uji, serta fitur validasi data yang dapat mengurangi kesalahan. Adanya *role* akun yang memiliki tugas masing-masing untuk melakukan proses secara realtime mulai dari admin yang menugaskan petugas untuk mengerjakan kalibrasi dari pengajuan produk dari pemohon atau pelaku industri. Dengan pendekatan ini, proses pengajuan uji produk dapat dilakukan lebih cepat, efisien, dan transparan, sehingga mendukung pelaku industri dalam memenuhi standar kualitas secara optimal.

Adapun penelitian terdahulu yang berkaitan dengan penelitian ini dilakukan oleh [2] hasil

penelitian yang dilakukan tentang sistem informasi manajemen perpustakaan di SMA 1 Negeri Katon. Dengan menggunakan metode *FAST (Framework for The Application of System Thinking)*, dapat disimpulkan bahwa aplikasi sistem informasi manajemen perpustakaan dibangun dengan tujuan untuk membantu pihak perpustakaan daerah dengan membuat database berbasis PHP dan *database* berbasis MySQL.

Penelitian terdahulu yang dilakukan oleh [3] Dalam penelitian ini, penulis menggunakan model *FAST* yang merupakan kerangka kerja yang cerdas dalam pendekatan istematis yang digunakan untuk memecahkan masalah yang kompleks di berbagai jenis proyek dan memiliki strategi yang baik. Tahapan dalam metode *FAST* terdiri dari *scope definition, problem analysis, requirement analysis, logical design, decision analysis, physical design, construction & testing, dan instalation and delivery.*

Terakhir penelitian terdahulu dilakukan oleh [4] Metode yang digunakan adalah metode berorientasi objek dengan menggunakan alat bantu pemodelan *Unified Modelling Language (UML)*, seperti *usecase diagram*, dan *sequence diagram* untuk visualisasi sistemnya yang dilakukan untuk penerapan pengembangan menggunakan model *FAST*, metode berorientasi objek dan diagram-diagram yang tersedia didalam *UML*.

2. TINJAUAN PUSTAKA

2.1. Metode FAST

Metode Pengembangan Sistem *FAST (Framework for the Application of Systems Thinking)* adalah pendekatan yang digunakan untuk mengembangkan sistem informasi dengan menekankan pada kecepatan dan efisiensi untuk menghasilkan solusi yang tepat waktu dengan memanfaatkan prinsip-prinsip berpikir sistematis, iterasi cepat, dan keterlibatan aktif dari pengguna. [2] Metode pengembangan sistem *FAST* memiliki aspek-aspek utama seperti pendekatan berpikir sistematis dalam mengidentifikasi masalah, iterasi proses yang cepat sehingga lebih sesuai dengan kebutuhan pengguna, keterlibatan aktif pengguna dalam proses pengajuan uji produk, focus pada kecepatan dan efisiensi tanpa mengurangi kualitas, menggunakan alat dan teknik modern untuk pengembang lebih responsive terhadap perubahan kebutuhan atau kondisi pasar, mengevaluasi dan perbaikan berkelanjutan serta menerapkan fleksibilitas dan adaptabilitas.

2.2. MySQL

MySQL adalah sistem manajemen basis data relasional (*RDBMS open-source*) yang paling populer di dunia. MySQL didasarkan pada bahasa pemrograman *Structured Query Language (SQL)*, yang merupakan standar industri untuk mengelola dan memanipulasi basis data relasional. MySQL memiliki arsitektur *client-server* yang terdiri dari server MySQL yang mengelola semua akses basis data dan berbagai

klien yang mengirimkan permintaan *SQL* ke server. Klien MySQL dapat berupa aplikasi yang ditulis dalam berbagai bahasa pemrograman seperti PHP, Java, Python, dan banyak lagi, yang mengakses basis data untuk mendapatkan atau menyimpan data sesuai kebutuhan.[5] Selain itu, integrasi yang mudah dengan berbagai alat pengembangan dan manajemen basis data juga meningkatkan daya tarik MySQL sebagai solusi basis data yang komprehensif.[6]

2.3. Laravel

Laravel adalah sebuah *framework* aplikasi *web open-source* yang ditulis dalam bahasa pemrograman PHP. *Framework* ini dirancang untuk memudahkan pengembangan aplikasi *web* dengan menyediakan berbagai fitur dan alat yang kuat namun sederhana, serta mempromosikan praktik pengembangan yang baik seperti pola desain *MVC (Model-View-Controller)* agar lebih terstruktur dan efisien.[7]

Laravel juga menawarkan *ORM (Object-Relational Mapping)* yang kuat bernama Eloquent. Eloquent memungkinkan pengembang untuk berinteraksi dengan basis data menggunakan objek PHP, mengurangi kompleksitas dalam menulis kueri *SQL* langsung.[8]

Laravel juga dikenal karena *system template Blade*-nya yang mendukung tampilan (*template inheritance*) dan komponen tampilan (*view components*), juga membantu dalam membangun tata letak yang konsisten dan mudah dipelihara. Selain itu, Laravel juga mendukung berbagai sistem basis data populer seperti MySQL, PostgreSQL, dan SQLite, serta memiliki dukungan yang kuat untuk integrasi dengan sistem-sistem eksternal dan layanan web lainnya.[8]

2.4. Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak paradigmatis berorientasi objek. Pemodelan sebenarnya digunakan untuk menyederhanakan masalah yang kompleks dengan cara yang membuatnya lebih mudah untuk dikuasai dan dipahami.[9]

2.5. Diagram Use Case

Use case diagram memodelkan perilaku sistem informasi yang akan dibuat. Kasus penggunaan menggambarkan interaksi antara satu atau lebih aktor dan sistem informasi yang akan didirikan. Ada dua hal utama dalam *use case* yaitu definisi *actor* dan *use case*.

2.6. Activity Diagram

Diagram aktivitas menggambarkan alur kerja (*workflow*) atau aktivitas suatu sistem dari proses bisnis, yang perlu diperhatikan di sini adalah diagram aktivitas merepresentasikan aktivitas sistem, bukan apa yang dilakukan pelaku, sehingga aktivitas tersebut dapat dilakukan oleh sistem.

2.7. Class Diagram

Class diagram bisa diartikan hubungan antar kelas dan penjelasan detail tiap kelas di dalamnya. Diagram kelas menunjukkan kelas dan batasan yang sesuai dengan objek yang terkait.

2.8. PHP

PHP, singkatan dari *Hypertext Preprocessor*, adalah bahasa pemrograman *scripting open-source* yang digunakan terutama untuk pengembangan aplikasi *web* dan dapat disematkan ke dalam kode HTML. PHP dirancang khusus untuk pengembangan web dinamis dan interaktif, dengan cara mengolah kode pada sisi server, yang berarti hasil dari eksekusi kode PHP ditampilkan dalam bentuk HTML yang dapat dikenali oleh *browser web*. [10]

PHP dapat dijalankan di berbagai *platform* sistem operasi, termasuk Windows, Linux, macOS, dan *platform* lainnya. [11]

2.9. CSS

CSS (*Cascading Style Sheets*) adalah bahasa *stylesheet* yang digunakan untuk mendesain dan mengatur tampilan halaman *web*. [12] Salah satu keuntungan utama dari penggunaan CSS adalah kemampuannya untuk memisahkan konten dari presentasi. Ini berarti bahwa perubahan pada tampilan situs web dapat dilakukan di satu *file* CSS tanpa perlu mengubah konten HTML. Hal ini meningkatkan efisiensi dan konsistensi dalam desain *web*, serta memudahkan proses pemeliharaan dan pembaruan situs *web*. [13]

CSS didukung oleh semua *browser modern*, membuatnya menjadi standar yang sangat kompatibel untuk pengembangan *web*. Dengan kemampuan untuk mengontrol tampilan dan tata letak halaman web secara efisien, CSS membantu pengembang menciptakan situs web yang tidak hanya fungsional tetapi juga menarik secara visual. Ini memungkinkan fleksibilitas dalam desain dan memastikan pengalaman pengguna yang konsisten di berbagai perangkat dan platform. [14]

3. METODE PENELITIAN

3.1. Metode Pengembangan Perangkat Lunak

Berikut adalah tahapan menggunakan metode FAST (*Framework for the Application of Systems Thinking*) untuk tata kelola kegiatan karyawan dalam lingkungan kantor:

- a. *Preliminary Investigation* (Investigasi Awal)
Mengidentifikasi masalah atau kebutuhan terkait tata kelola kegiatan karyawan di lingkungan kantor.
- b. *Problem Analysis* (Analisis Masalah)
Data terkait kegiatan karyawan dikumpulkan melalui observasi, wawancara, dan survei.
- c. *Requirements Analysis* (Analisis Kebutuhan)
Pada tahap ini, kebutuhan pengguna dan sistem didokumentasikan secara rinci. Kemudian, kebutuhan yang dikumpulkan divalidasi untuk memastikan kesesuaiannya.

- d. *Logical Design* (Desain Logis)

Alur kerja dan proses tata kelola kegiatan karyawan yang baru dirancang, serta antarmuka pengguna yang dirancang untuk memastikan kemudahan dalam penggunaan sistem.

- e. *Decision Analysis* (Analisis Keputusan)

Berbagai alternatif Solusi untuk membangun sistem dievaluasi dipilih berdasarkan analisis biaya-manafaat dan kriteria lainnya.

- f. *Physical Design* (Desain Fisik):

Arsitektur teknis dari sistem, yang mencakup perangkat keras, perangkat lunak, dan jaringan, dirancang. Struktur dan skema *database* yang akan digunakan dirancang, untuk memastikan sistem yang dirancang sesuai dengan spesifikasi kebutuhan.

- g. *Construction and Testing* (Konstruksi dan Pengujian)

Sistem dikembangkan sesuai dengan desain yang telah dibuat. Setiap komponen atau modul sistem diuji secara menyeluruh.

- h. *Implementation* (Implementasi)

Data dipindahkan dari sistem lama ke sistem yang baru untuk memastikan semua informasi yang relevan tersedia dalam sistem yang baru. Setelah semua persiapan selesai, sistem baru diluncurkan secara resmi dan mulai digunakan oleh seluruh pengguna.

3.2. Analisis Sistem

Pengelolaan kegiatan karyawan memiliki peran krusial dalam memastikan produktivitas dan kualitas layanan dalam sebuah organisasi. Karyawan adalah aset utama yang perlu mendapatkan tujuan yang jelas, umpan balik yang konstruktif, dan dukungan berkelanjutan. Namun, proses pengembangan dan pelatihan karyawan yang berkelanjutan sering kali diabaikan, sehingga karyawan menjadi kurang relevan dan kompetitif di tengah perubahan era bisnis yang dinamis. Tanpa tata kelola yang baik, kinerja karyawan tidak dapat optimal, menghambat tercapainya hasil yang diinginkan. Penggunaan Microsoft Office Excel secara manual untuk mengelola kegiatan pengujian menyebabkan berbagai masalah operasional. Pendataan jadwal kalibrasi yang memakan waktu lama mengakibatkan ketidakefisienan dan kesalahan. Selain itu, pengolahan data produk secara manual sering kali menghasilkan data yang tidak akurat, mengganggu operasional dan layanan BSPJI Medan. Metodologi FAST akan membantu mengidentifikasi masalah utama dalam pengelolaan kegiatan karyawan dan merumuskan solusi komprehensif untuk meningkatkan produktivitas dan kinerja perusahaan.

3.3. Perancangan Sistem Use Case Diagram

Diagram *use case* untuk sistem kalibrasi menggambarkan interaksi antara dua aktor utama, yaitu *Admin* dan *Pelanggan*, dengan berbagai fitur dalam sistem. Aktor *Admin* memiliki akses ke fitur login. Setiap *sub-fitur* dihubungkan ke fitur *login*

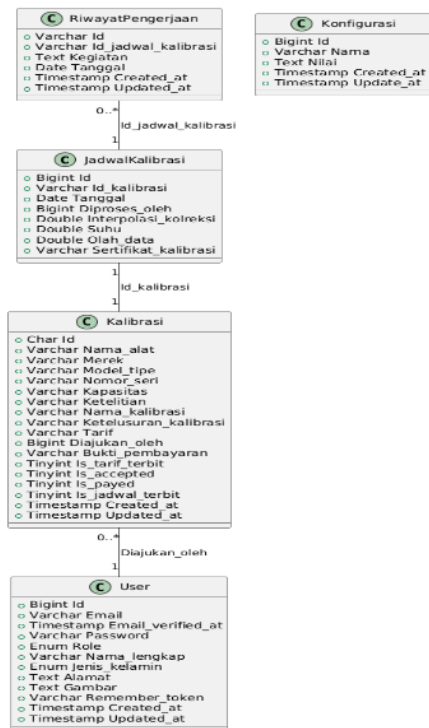
menggunakan hubungan *include*, yang menunjukkan bahwa fitur-fitur tersebut hanya dapat diakses setelah *Admin* melakukan *login*. Aktor *Pelanggan* memiliki dua fitur utama yang dapat diakses, yaitu fitur *register* dan fitur *login*. Setelah pengguna *login*, mereka memiliki akses ke *sub-fitur* seperti pengajuan kalibrasi, jadwal kalibrasi, dan ubah *password*, yang juga dihubungkan ke fitur *login* melalui hubungan *include*. Hubungan *include* dalam diagram ini menunjukkan bahwa setelah melakukan *login*, baik *Admin* maupun *Pelanggan* dapat mengakses fitur-fitur yang relevan dengan peran mereka.



Gambar 1. Use Case Diagram

3.4. Perancangan Sistem Class Diagram

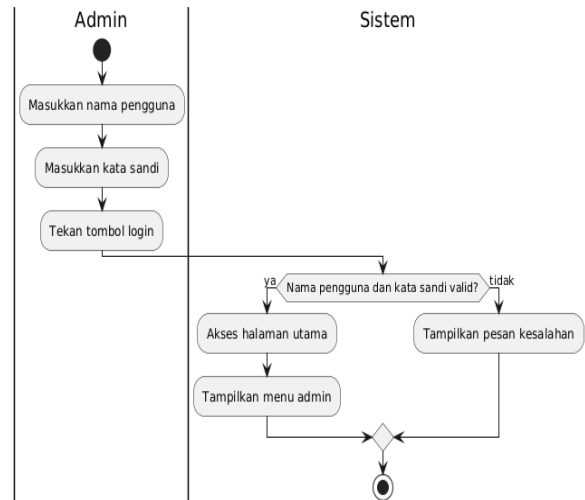
Class diagram yang menggambarkan sistem kalibrasi terdiri dari beberapa entitas utama yang terhubung dengan berbagai relasi, seperti gambar 2 berikut.



Gambar 2. Class Diagram

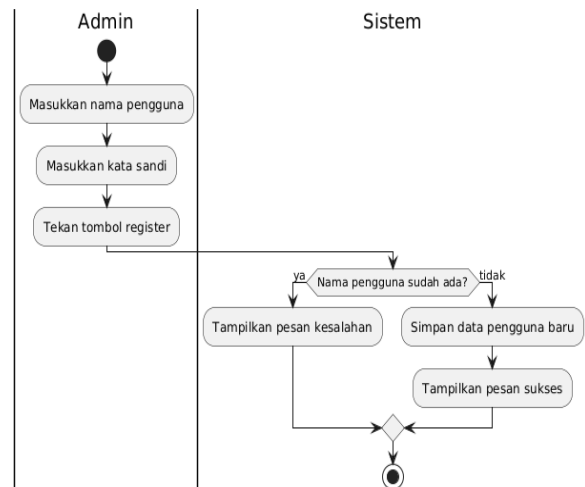
3.5. Activity Diagram

Rangkaian kegiatan pada setiap terjadi *event* sistem digambarkan pada *activity* diagram, seperti diagram aktivitas *login* menggambarkan proses *login* untuk seorang *admin*.



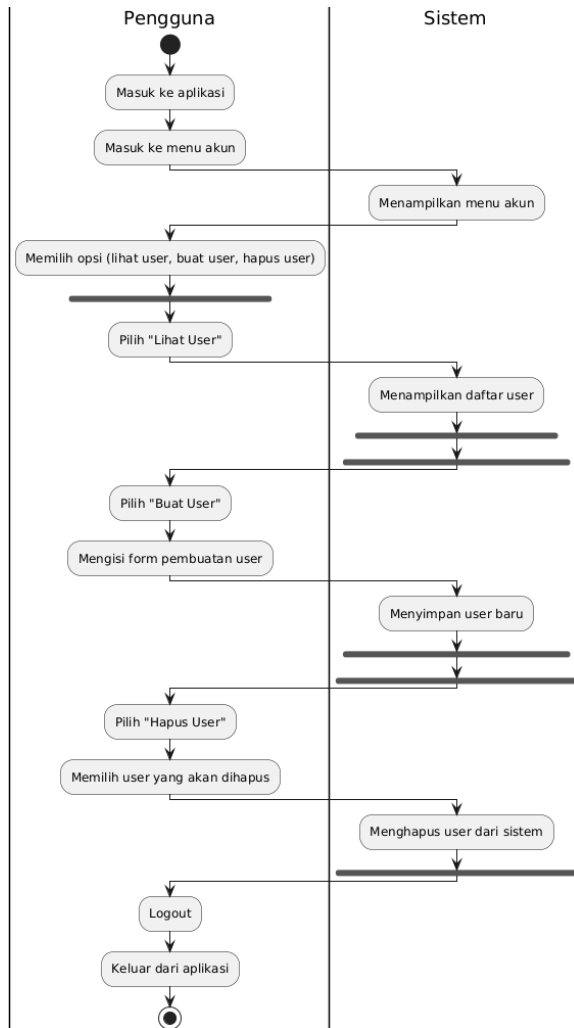
Gambar 3. Activity Diagram Login Admin

Diagram ini membantu memvisualisasikan langkah-langkah yang terlibat dalam proses *login* dari perspektif *admin* dan sistem. Diagram aktivitas untuk proses registrasi menggambarkan langkah-langkah yang diambil oleh *admin* dan sistem.



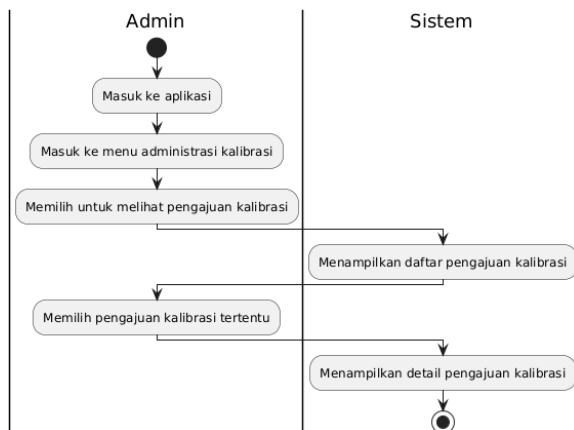
Gambar 4. Activity Diagram Register

Proses ini berakhir setelah sistem memberikan respon yang sesuai berdasarkan hasil validasi nama pengguna. *Activity* diagram ini menggambarkan proses interaksi antara pengguna dan sistem dalam menu akun aplikasi.



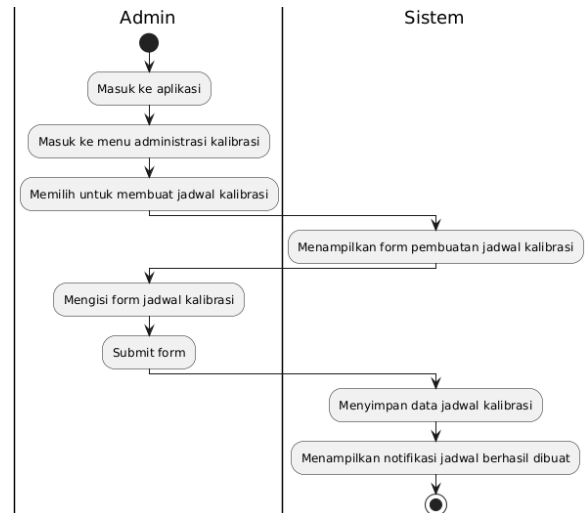
Gambar 5. Activity Diagram Menu Akun

Diagram ini menggunakan *swimlane* untuk memisahkan tindakan antara Pengguna dan Sistem, sehingga memberikan visualisasi yang jelas tentang siapa yang melakukan setiap aktivitas dan bagaimana interaksi antara pengguna dan sistem terjadi dalam proses pengelolaan akun. Activity diagram ini menggambarkan proses melihat pengajuan kalibrasi oleh *admin* dalam sistem.



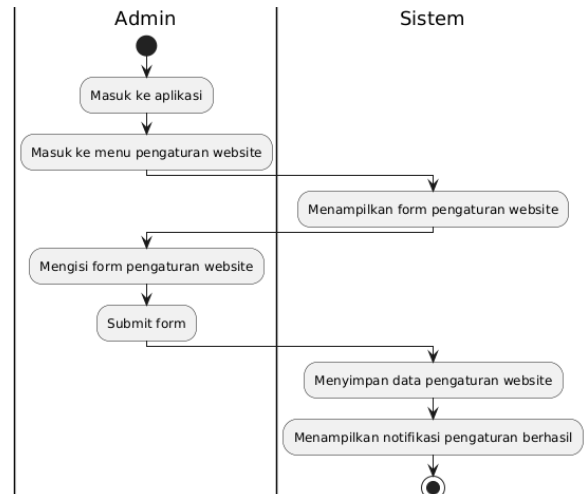
Gambar 6. Activity Diagram Pengajuan Kalibrasi

Diagram ini menggunakan *swimlane* untuk memisahkan tindakan antara *Admin* dan Sistem, memberikan visualisasi yang jelas tentang siapa yang melakukan setiap aktivitas dan bagaimana interaksi antara *admin* dan sistem terjadi dalam proses melihat pengajuan kalibrasi. Activity diagram ini menggambarkan proses yang dilakukan oleh *admin* untuk membuat jadwal kalibrasi dalam sistem.



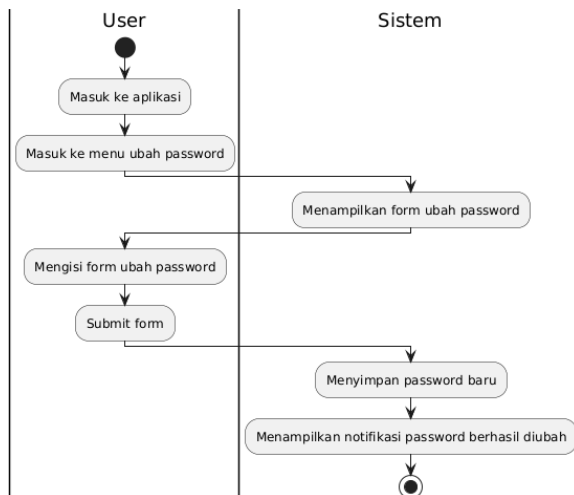
Gambar 7. Activity Diagram Jadwal Kalibrasi

Diagram ini menggunakan *swimlane* untuk memisahkan aktivitas yang dilakukan oleh *Admin* dan Sistem, memberikan visualisasi yang jelas tentang peran masing-masing dalam proses pembuatan jadwal kalibrasi. Activity diagram ini menggambarkan proses pengaturan *website* oleh *admin* dalam sistem.



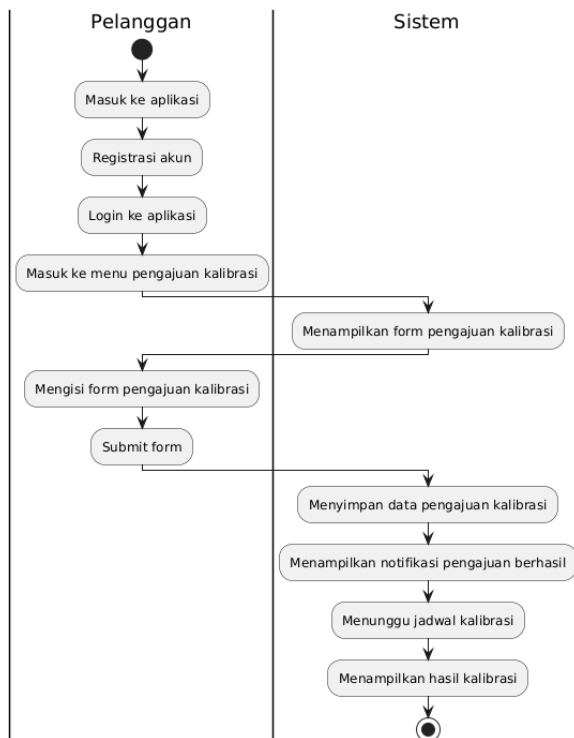
Gambar 8. Activity Diagram Pengaturan Website

Diagram ini menggunakan *swimlane* untuk memisahkan aktivitas yang dilakukan oleh *admin* dan sistem, memberikan visualisasi yang jelas tentang peran masing-masing dalam proses pengaturan *website*. Proses dimulai dengan pengguna mengakses Menu Perhitungan.



Gambar 9. Activity Diagram Ubah Password

Setelah itu, sistem secara otomatis melakukan perhitungan berdasarkan data yang tersedia. Activity diagram ini menggambarkan proses pengajuan kalibrasi oleh pelanggan dalam system.

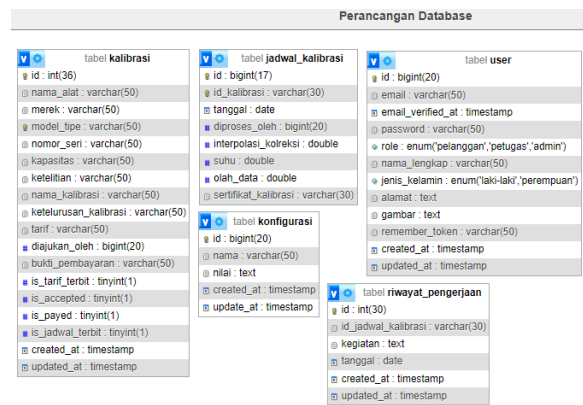


Gambar 10. Activity Diagram Pengajuan Kalibrasi Pelanggan

Diagram ini menggunakan swimlane untuk memisahkan aktivitas yang dilakukan oleh pelanggan dan sistem, memberikan visualisasi yang jelas tentang peran masing-masing dalam proses pengajuan kalibrasi.

3.6. Perancangan Database

Perancangan database digunakan untuk menentukan nilai atau type data suatu atribut, pada file yang terdapat pada database.



Gambar 11. Perancangan Database

Perancangan database juga digunakan untuk memudahkan penyimpanan data yang sesuai dengan kelompok dari data atau informasi.

4. HASIL DAN PEMBAHASAN

4.1. Hasil Implementasi Sistem

Setelah penelitian selesai, tahap berikutnya adalah mengimplementasikan sistem. Sistem ini mencakup beberapa halaman, masing-masing dengan fungsinya sendiri. Berikut adalah halaman-halaman yang ditampilkan:

4.2. Halaman Login

Berikut merupakan tampilan halaman login pada gambar 12 yaitu sebagai berikut ini.

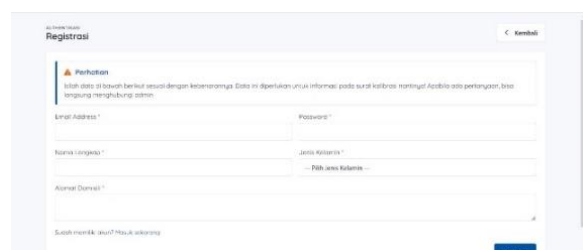


Gambar 12. Halaman Login

Pada gambar 12 menunjukkan sebuah halaman login yang digunakan untuk mengakses sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).

4.3. Halaman Register

Berikut merupakan tampilan halaman register pada gambar 13 yaitu sebagai berikut ini.

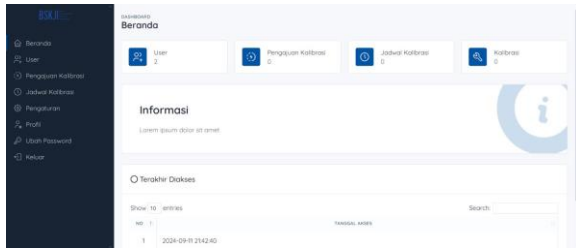


Gambar 13. Halaman Register

Pada gambar 13 menampilkan halaman registrasi untuk membuat akun baru dalam sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).

4.4. Halaman Dashboard

Berikut merupakan tampilan halaman *dashboard* pada sistem pada gambar 14 yaitu sebagai berikut.

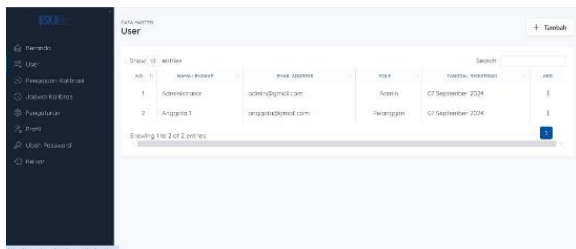


Gambar 14. Halaman User

Pada gambar 14 menampilkan *dashboard* dari sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).

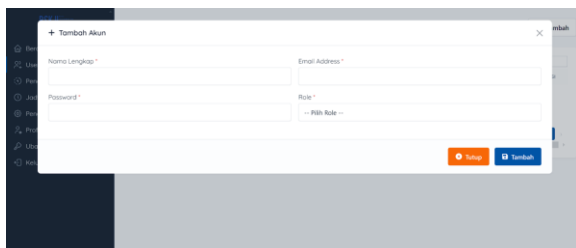
4.5. Halaman User

Berikut merupakan tampilan halaman *user* pada gambar 15 yaitu sebagai berikut.



Gambar 15. Halaman Data User

Pada gambar 15 menampilkan halaman Data *Master User* dari sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).

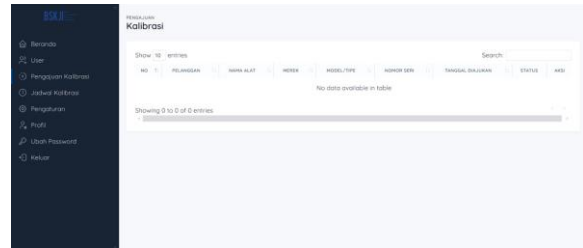


Gambar 16. Tambah User

Pada gambar 16 menunjukkan formulir Tambah Akun yang digunakan untuk menambahkan pengguna baru ke dalam sistem.

4.6. Halaman Pengajuan Kalibrasi

Berikut merupakan tampilan halaman pengajuan kalibrasi pada gambar 17 yaitu sebagai berikut.

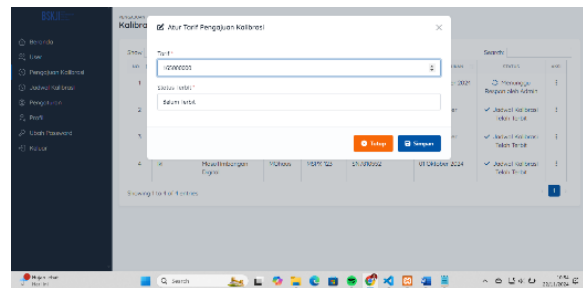


Gambar 17. Halaman Pengajuan Kalibrasi

Pada gambar 17 menampilkan halaman Pengajuan Kalibrasi dalam sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).

4.7. Halaman pengaturan Tarif Kalibrasi

Berikut merupakan tampilan halaman pengaturan tarif pada gambar 18 sebagai berikut.

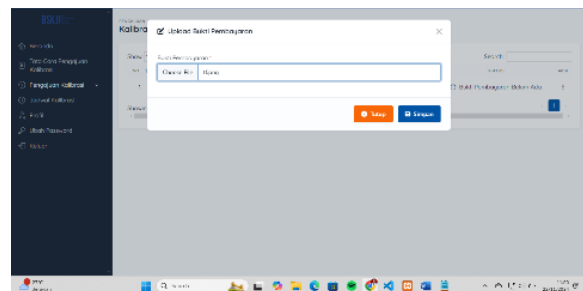


Gambar 18. Halaman Pengaturan Tarif Kalibrasi

Pada gambar 18 proses ini dilakukan oleh *admin* untuk pengajuan unit kalibrasi nya.

4.8. Halaman Upload Bukti Pembayaran

Berikut merupakan tampilan halaman *upload* bukti pembayaran pada gambar 19 berikut.

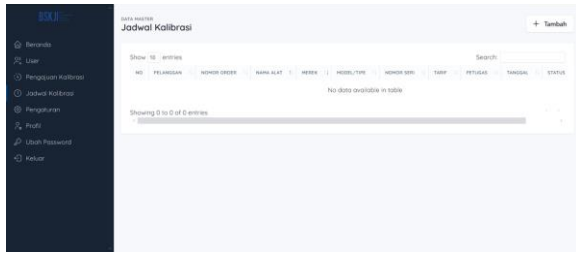


Gambar 19. Halaman Upload Bukti Pembayaran

Pada gambar 19 merupakan halaman *upload* bukti pembayaran dari tarif yang sudah disetujui oleh pelanggan.

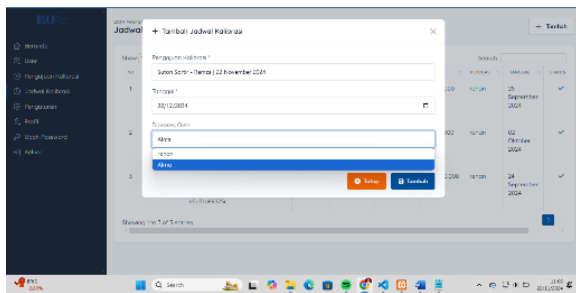
4.9. Halaman Jadwal Kalibrasi

Berikut merupakan tampilan halaman jadwal kalibrasi pada gambar 20 yaitu sebagai berikut :



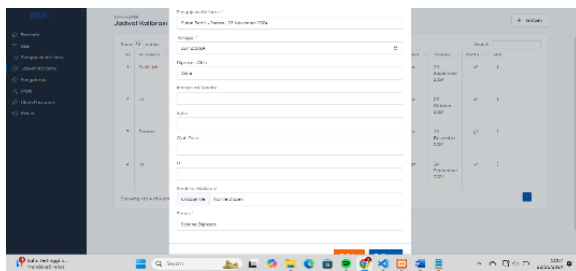
Gambar 20. Halaman Jadwal Kalibrasi

Pada gambar 20 menampilkan halaman Jadwal Kalibrasi dalam sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).



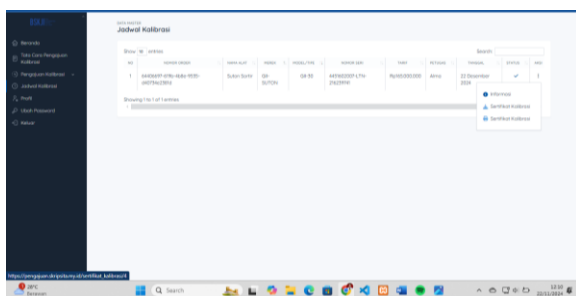
Gambar 21. Tambah Jadwal Kalibrasi

Pada gambar 21 menunjukkan formulir Tambah Jadwal Kalibrasi serta penyerahan petugas kalibrasi pengajuan kalibrasi.



Gambar 22. Detail Kalibrasi

Pada gambar 22 menampilkan form detail kalibrasi oleh petugas kalibrasi.

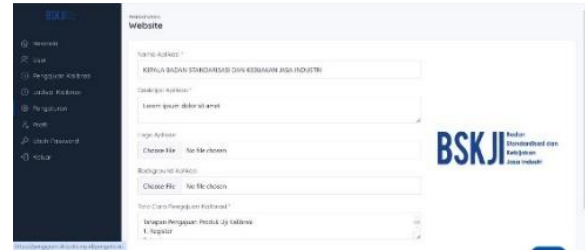


Gambar 23. Output Kalibrasi

Pada gambar 23 menampilkan *output* atau hasil dari kalibrasi merupakan sertifikat kalibrasi yang sudah di *upload* oleh petugas kalibrasi yang bisa di *download* serta di *print*.

4.10. Halaman Pengaturan

Berikut merupakan tampilan pengaturan pada gambar 24 yaitu sebagai berikut.

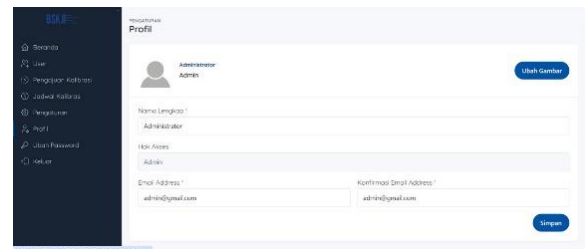


Gambar 24. Halaman Pengaturan

Pada gambar 24 menampilkan halaman Pengaturan *Website* dalam sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).

4.11. Halaman Profil

Berikut merupakan tampilan halaman profil pada gambar 25 yaitu sebagai berikut.



Gambar 25. Halaman Profil

Pada gambar 25 menampilkan halaman Pengaturan Profil dalam sistem Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI).

4.12. Pembahasan

Bab ini membahas hasil dan implementasi dari sistem yang telah dirancang untuk Badan Standardisasi dan Kebijakan Jasa Industri (BSKJI). Beberapa halaman utama yang dijelaskan dalam bab ini meliputi halaman *login*, registrasi, *dashboard*, data *user*, pengajuan kalibrasi, jadwal kalibrasi, pengaturan, dan profil. Masing-masing halaman memiliki fungsi spesifik yang membantu pengguna atau administrator dalam mengelola data dan aktivitas terkait proses kalibrasi alat industri. Halaman *login* berfungsi untuk mengotentikasi pengguna terdaftar, sedangkan halaman registrasi memungkinkan pendaftaran pengguna baru. *Dashboard* memberikan ringkasan aktivitas, seperti jumlah pengguna dan status pengajuan kalibrasi, serta navigasi cepat ke berbagai fitur penting. Halaman data *user* menampilkan daftar pengguna yang terdaftar, sementara pengajuan kalibrasi dan jadwal kalibrasi mempermudah pengguna dalam mengajukan dan memantau jadwal kalibrasi alat. Selain itu, halaman pengaturan memungkinkan *administrator* untuk menyesuaikan tampilan dan informasi aplikasi, seperti logo dan latar belakang. Halaman profil memungkinkan pengguna untuk memperbarui

informasi pribadi mereka. *Role* akun pelanggan akan melakukan pengajuan uji produk yang selanjutnya akan di proses oleh *admin* mulai dari pengaturan tarif kalibrasi serta proses kalibrasi nya untuk diproses jadwal kalibrasi nya. Setelah dilakukan pengaturan tarif, pelanggan akan menerima apakah tarif sesuai jika sudah maka terima dan *upload* bukti pembayaran untuk dilakukan proses penjadwalan kalibrasi dari admin yang akan memberikan pengajuan produk yang tarif nya sudah di terima dan kapan dilaksanakannya jadwal kalibrasi kepada petugas yang akan melakukan kalibrasi produk dan *upload* bukti sertifikasi kalibrasi. Setelah proses selesai maka pelanggan dapat melihat detail kalibrasi dan *download* sertifikat kalibrasi untuk pengajuan produk.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan bahwa, implementasi aplikasi pengajuan uji produk di BSPJI Medan menggunakan metode *FAST* terbukti mempercepat proses pengajuan, dengan mengotomatisasi berbagai tahapan penting seperti pemeriksaan awal, pengujian, dan evaluasi hasil uji, sehingga mampu mengurangi potensi kesalahan dalam pengisian data dan mempercepat waktu pemrosesan, juga meningkatkan transparansi dalam proses pengajuan uji produk, karena pengguna dapat melacak status permohonan mereka secara *real-time*, yang sebelumnya sulit dilakukan dengan proses manual dikarenakan setiap *role* yang memiliki tugas masing masing seperti *role* akun pelanggan akan melakukan pengajuan uji produk yang selanjutnya akan di proses oleh *admin* mulai dari pengaturan tarif kalibrasi serta proses kalibrasi nya untuk diproses jadwal kalibrasi nya. Setelah dilakukan pengaturan tarif, pelanggan akan menerima apakah tarif sesuai jika sudah maka terima dan *upload* bukti pembayaran untuk dilakukan proses penjadwalan kalibrasi dari admin yang akan memberikan pengajuan produk yang tarif nya sudah di terima dan kapan dilaksanakannya jadwal kalibrasi kepada petugas yang akan melakukan kalibrasi produk dan *upload* bukti sertifikasi kalibrasi. Setelah proses selesai maka pelanggan dapat melihat detail kalibrasi serta *download* sertifikat kalibrasi untuk pengajuan produk, sehingga industri dapat mengajukan permohonan secara *online* dengan lebih efisien Saran yang dapat diberikan yaitu untuk pengembangan fitur tambahan seperti notifikasi otomatis yang dapat membantu meningkatkan pengalaman pengguna dan efisiensi lebih lanjut.

DAFTAR PUSTAKA

- [1] A. Batary Citta, "Pengembangan SumberDayaManusia Dalam Mendukung Transformasi Transportasi Laut Dalam Era Revolusi Industri 4.0," *J. Soc. Sci. Res.*, vol. 3, no. 4, pp. 6184–6197, 2023.
- [2] M. P. Sari, S. Setiawansyah, and ..., "Perancangan Sistem Informasi Manajemen Perpustakaan Menggunakan Metode Fast (Framework for the Application System Thinking)(Studi Kasus: Sman 1 Negeri ..., " ... *Dan Sist. Inf.*, vol. 2, no. 2, pp. 69–77, 2021, [Online]. Available: <https://jim.teknokrat.ac.id/index.php/sisteminformasi/article/view/1136%0Ahttps://jim.teknokrat.ac.id/index.php/sisteminformasi/article/viewFile/1136/361>
- [3] S. S. Sundari, A. Supriatman, E. Dewi, S. Mulyani, R. Prasetya, and Y. Teuku, "Perancangan E-commerce pada Unit Kerja BUMDes Desa Tanjungsari Berbasis Web Menggunakan Metode FAST (Framework for the Application of System Thinking)," pp. 424–429, 2024.
- [4] K. W. Haryanto, M. F. Roji, D. Fitri, and P. Online, "Pakaian Menggunakan Metode Framework for," vol. 16, no. 1, pp. 264–271, 2024.
- [5] Romadloni, N. Tri, and M. Miswanto, *Sistem Basis Data: Teori Dan Praktikum*. PT. Sonpedia Publishing Indonesia, 2024.
- [6] S. Sutedi *et al.*, *Buku Ajar Sistem Basis Data*. PT. Sonpedia Publishing Indonesia, 2024.
- [7] Arya Hafizh Tofani and Fadelis Sukya, "Sistem Informasi Manajemen Kegiatan UKM English Club PSDKU Polinema Di Kediri Berbasis Framework Laravel," *J. Inform. dan Multimed.*, vol. 14, no. 2, pp. 15–22, 2023, doi: 10.33795/jim.v14i2.1131.
- [8] Kurniawan *et al.*, *BELAJAR WEB PROGRAMMING: Referensi Pengenalan Dasar Tahapan Belajar Pemrograman Web Untuk Pemula*. PT. Sonpedia Publishing Indonesia, 2023.
- [9] F. R. W. Siregar, I. Lubis, and Maulana, "Sistem informasi Inventory Iparepart Motor Pada Bengkel Al Motor," *J. Nas. Teknol. Komput.*, vol. 3, no. 4, pp. 297–313, 2023, doi: 10.61306/jnastek.v3i4.113.
- [10] N. Harani and A. Sunandhar, *Aplikasi Prospek Sales Menggunakan Codeigniter*. Kreatif, 2020.
- [11] Rachmad, Y. Edhie, R. Dewantara, S. Junaidi, M. Firdaus, and S. Sulistianto, *Mastering Cloud Computing (Foundations and Applications Programming)*. PT. Sonpedia Publishing Indonesia, 2023.
- [12] Komaran, R. Mahya, R. Andarsyah, M. Lubis, and R. M. Awangga, *Langkah Mudah Membuat Dashboard Rest API*. Penerbit Buku Pedia, 2023.
- [13] Mukhlis *et al.*, *BUKU AJAR PEMROGRAMAN WEB 1*. PT. Sonpedia Publishing Indonesia, 2023.
- [14] Putri, Ardiva, and R. M. Awangga, *Membangun Frontend Dan Backend Packages Dengan Golang"Studi Kasus Sistem Administrasi"*. Penerbit Buku Pedia, 2023.