

Perancangan Aplikasi *Work Order* pada PT. Makmur Bintang Plastindo Berbasis *Web*

M Yunus^{[1]*}, Supardi^[2]
Politeknik Ganesha Medan^{[1],[2]}
Medan, Indonesia

muhammadyunus12321@gmail.com^[1] pardiyasin73@gmail.com^[2]

*Penulis Korespondensi

Diajukan : 06/01/2025

Diterima : 10/01/2025

Dipublikasi : 15/01/2025

Abstract— *This research aims to develop a web-based information system to facilitate maintenance and repair management of machines at PT. Makmur Bintang Plastindo. This system is designed to overcome problems that often arise in maintenance using paper and pen such as delays in reporting damage and difficulty monitoring the status of mechanical work. Through this system, operators, technicians and managers can interact online, update work status and monitor vehicle repair progress in real-time. The research results show that this system increases mechanical work efficiency and makes it easier to manage machine maintenance and repair.*

Keywords— *Web Based Information Systems, Production Efficiency, Work order Applications, Waterfall Method, Information Technology.*

Abstrak— Penelitian ini bertujuan untuk mengembangkan sistem informasi berbasis *web* guna memudahkan manajemen pemeliharaan dan perbaikan mesin di PT. Makmur Bintang Plastindo. Sistem ini dirancang untuk mengatasi masalah yang sering muncul dalam pemeliharaan dengan menggunakan kertas dan pena seperti keterlambatan pelaporan kerusakan dan sulitnya memantau status pekerjaan mekanik. Melalui sistem ini, *operator*, teknisi dan *manager* dapat berinteraksi secara online, memperbarui status pekerjaan serta memantau perkembangan perbaikan kendaraan secara real-time. Hasil penelitian menunjukkan bahwa sistem ini meningkatkan efisiensi kerja mekanik dan memudahkan pengelolaan pemeliharaan dan perbaikan mesin.

Kata Kunci— *Sistem Informasi Berbasis Web, Efisiensi Produksi, Aplikasi Work order, Metode Waterfall, Teknologi Informasi.*

I. PENDAHULUAN

PT. Makmur Bintang Plastindo, sebuah perusahaan manufaktur yang memproduksi karung goni dari biji plastik untuk keperluan pengemasan berbagai bahan pangan seperti tepung, gula dan beras serta pupuk dan pakan ternak masih mengandalkan sistem manual berbasis kertas dan pena dalam proses pengaduan dan pelaporan kerusakan mesin kepada teknisi. Hal ini menyebabkan penanganan memakan waktu yang cukup lama, terutama mengingat luasnya area perusahaan. Di era Revolusi Industri 4.0, di mana kemajuan teknologi telah mengubah berbagai aspek kehidupan, penggunaan metode tradisional seperti ini dinilai kurang efisien. Perusahaan serupa telah beralih ke sistem berbasis teknologi untuk meningkatkan produktivitas dan mengurangi kesalahan, mendorong perlunya perubahan pada PT. Makmur Bintang Plastindo agar lebih efektif dan kompetitif.

Oleh karena itu, sebuah perusahaan harus mampu beradaptasi dan mengikuti perkembangan zaman seperti halnya PT. Makmur Bintang Plastindo yang saat ini berupaya menyesuaikan diri dengan tren teknologi terkini.



PT. Makmur Bintang Plastindo, sebuah perusahaan manufaktur dengan 132 mesin yang digunakan untuk memproduksi karung goni, sering menghadapi berbagai tantangan operasional. Untuk mengatasinya, perusahaan menerapkan formulir *work order* sebagai bagian dari sistem kerja dan terus mengembangkan sistem perintah kerja yang lebih efisien guna mengoptimalkan operasional. Menurut penelitian sebelumnya oleh Andi Saputra, penggunaan sistem perintah kerja mempermudah seluruh kegiatan di perusahaan, terutama dalam hal administrasi (Saputra, Imamuddin, & Sukanto, 2020). Selain itu, Erliza Yubarda juga mengemukakan bahwa untuk mengurangi kesalahan operasional, perusahaan dapat menggunakan dokumen khusus sebagai perintah kepada organisasi untuk menjalankan manajemen dengan lebih efektif (Vella, Yubarda, Jannah, & Surya, 2022).

Penelitian sebelumnya menunjukkan bahwa penggunaan *website* manajemen perintah kerja jauh lebih efektif dibandingkan sistem berbasis kertas seperti yang sedang diterapkan oleh PT. Makmur Bintang Plastindo. Sistem manajemen perintah kerja teknisi di perusahaan tersebut dinilai tidak efektif, karena sering terjadi kesalahan seperti keterlambatan dalam identifikasi masalah. Hal ini disebabkan oleh prosedur yang mengharuskan *operator* melapor ke bagian operasional sebelum mencari teknisi yang memakan waktu lama. Selain itu, penggunaan kertas untuk menyampaikan informasi kepada mekanik dianggap tidak efisien karena kertas mudah rusak yang berpotensi menyebabkan hilangnya informasi penting terkait pekerjaan. Oleh karena itu, perusahaan membutuhkan sistem manajemen perintah kerja otomatis seperti *website* yang akan dikembangkan, untuk memperbaiki sistem lama dan meningkatkan efisiensi operasional di PT. Makmur Bintang Plastindo.

II. STUDI LITERATUR

Website

Pada awalnya *website* dikembangkan dengan tujuan untuk menyampaikan informasi kepada orang lain. Sekarang fungsi *website* telah meluas. Pada saat ini *website* digunakan sebagai media dalam mempromosikan produk, jasa, bahkan menunjukkan suatu instansi kepada orang lain. Jika dibedakan dengan media yang lain, seperti brosur, poster, surat kabar dan yang lain. *Website* juga dapat disebut dengan media yang efektif dan efisien. Kelebihan dari *website* adalah daya jangkauan yang luas, informasi tersampaikan dengan cepat dan biaya yang terjangkau. Aplikasi berbasis *web* memuat beberapa kelebihan, sebagai berikut:

1. *Platform* Independen, aplikasi ini dapat di jalankan melalui sistem operasi windows, linux dan Mac OS.
2. Menggunakan aplikasi *web* tidak memerlukan instal pada setiap komputer karena dapat dilakukan dengan *copy script program* ke *server* atau komputer. Jika komputer lain ingin menjalankan cukup dengan membuka alamat yang telah tersimpan melalui *browser*.
3. Aplikasi *web* dapat digunakan dalam jarak jauh tetap dengan bantuan internet.

Work order

Work order adalah dokumen resmi yang dikeluarkan untuk tugas pemeliharaan, perbaikan atau operasional yang merinci pekerjaan yang harus dilakukan, sumber daya yang dibutuhkan serta instruksi penyelesaian. Dokumen ini berfungsi sebagai alat penting untuk mengelola, menetapkan dan memantau aktivitas pemeliharaan, memastikan bahwa tugas-tugas diselesaikan dengan efisien dan sesuai dengan tujuan operasional.

Sebagai elemen inti dalam operasi pemeliharaan, perintah kerja yang diatur dengan baik membantu tim tetap terorganisir, mengatur prioritas pekerjaan dengan efektif dan menyelesaikan tugas dengan lancar. Perintah kerja berfungsi sebagai otorisasi untuk melaksanakan pemeliharaan, perbaikan atau operasional yang telah ditentukan. Perintah kerja bisa dibuat secara manual berdasarkan permintaan dari staf, klien atau penyewa secara otomatis melalui perangkat lunak manajemen perintah kerja (Sidik, 2019).

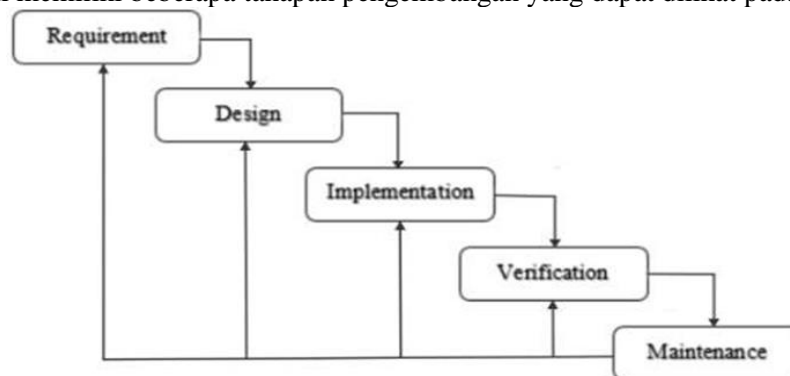
User Interface dan User Experience

User Experience (UX) merupakan proses mendesain suatu produk melalui pendekatan

pengguna produk dengan desain *UX* yang baik akan menciptakan pengalaman yang menyenangkan bagi pengguna saat menggunakan produk Anda. Pengguna jadi mudah dan nyaman saat menggunakan produk. Adapun, komponen *UX* ini meliputi bagaimana fitur-fitur yang disediakan pada produk, struktur desain, navigasi penggunaan produk, aspek visual design dan seluruh aspek interaksi dengan pengguna. *User Interface* adalah bagian dari *UX* yang berupa tampilan *visual design* sebuah sistem. Tampilan tersebut memungkinkan pengguna terhubung dan berinteraksi dengan suatu produk. Selain berfungsi sebagai penghubung, *User Interface* juga berfungsi untuk memperindah tampilan sehingga dapat meningkatkan kepuasan pengguna. Namun, tak hanya harus indah, *User Interface* juga harus mudah digunakan. Beberapa komponen *User Interface* diantaranya adalah komponen tombol, ikon tipografi, tema, layout, animasi yang tampil pada produk dan visual interaktif lainnya. Semua komponen *User Interface* tersebut didesain dengan berfokus pada keindahan dan kemudahan pengguna (Abdussalaam & Ramadhan, 2019).

Metode Waterfall

Pada penelitian ini, metode pengembangan sistem yang digunakan adalah metode waterfall. Metode waterfall memiliki beberapa tahapan pengembangan yang dapat dilihat pada Gambar 2.1.



Gambar 1. Metode Waterfall

Sumber : A.A Wahid

Tahapan-tahapan dari metode waterfall adalah sebagai berikut :

1. *Requirement*

Pada tahap ini, pengembang sistem perlu berkomunikasi untuk memahami perangkat lunak yang diinginkan oleh pengguna serta batasan-batasannya. Informasi diperoleh melalui wawancara, diskusi atau survei langsung dan kemudian dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

2. *Design*

Pada tahap ini, pengembang merancang sistem yang berfungsi untuk menentukan kebutuhan perangkat keras (*hardware*) dan persyaratan sistem serta membantu mendefinisikan arsitektur keseluruhan sistem.

3. *Implementation*

Pada tahap ini, sistem dikembangkan terlebih dahulu dalam bentuk program kecil yang disebut unit yang kemudian diintegrasikan pada tahap berikutnya. Setiap unit dikembangkan dan diuji fungsionalitasnya melalui proses yang dikenal sebagai *unit testing*.

4. *Verification*

Pada tahap ini, sistem menjalani verifikasi dan pengujian untuk memastikan apakah sistem memenuhi seluruh atau sebagian persyaratan yang telah ditentukan. Pengujian ini dapat dibagi menjadi beberapa kategori: *unit testing* (dilakukan pada modul tertentu dalam kode), *sistem testing* (untuk mengamati reaksi sistem ketika semua modul terintegrasi) dan *acceptance testing* (dilakukan bersama pelanggan untuk memastikan semua kebutuhan mereka terpenuhi).

5. *Maintenance*

Ini adalah tahap akhir dari metode *waterfall*. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya (Umiga, 2022).




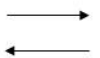
Diagram Alir Data

Diagram Alir Data (*Data Flow Diagram* atau *DFD*) adalah alat visual yang digunakan untuk menggambarkan aliran informasi dalam suatu sistem informasi Diagram Alir Data (*DFD*) mengidentifikasi proses-proses yang terjadi dalam suatu sistem, menggambarkan data yang digunakan dalam proses tersebut serta menunjukkan bagaimana data mengalir melalui berbagai bagian sistem. Berikut penjelasan komponen utama *DFD*:

1. Proses: Merepresentasikan aktivitas yang mengubah *input* menjadi *output*. Setiap proses diberi nama atau nomor, menggambarkan suatu fungsi atau langkah dalam sistem.
2. Aliran Data: Menunjukkan pergerakan data dalam sistem, baik dari proses ke proses dari luar sistem ke dalam, maupun sebaliknya, diwakili oleh panah yang menunjukkan arah aliran.
3. Entitas: Sumber atau tujuan data seperti objek, orang, tempat atau konsep yang menyimpan atau menggunakan data. Entitas digambarkan dengan kotak dan diberi nama sesuai perannya dalam sistem.
4. Penyimpanan Data: Tempat penyimpanan data dalam sistem seperti *database* atau *file* digambarkan dengan dua garis horizontal.
5. Luaran: *Output* yang dihasilkan dari proses, misalnya laporan atau informasi yang diberikan ke entitas atau sistem *eksternal*, digambarkan dengan panah keluar dari proses.
6. *Input*: Data yang diterima oleh suatu proses untuk diolah, bisa berasal dari entitas atau penyimpanan data, digambarkan dengan panah masuk ke proses.

DFD membantu pengembang sistem dan pemangku kepentingan memahami serta menggambarkan aliran data dalam sistem. Diagram ini digunakan dalam perancangan dan analisis sistem serta memfasilitasi komunikasi antar pihak terkait. *DFD* disajikan dalam beberapa tingkatan dari level 0 yang paling umum hingga level n yang lebih rinci, guna menunjukkan detail sistem secara bertahap. Adapun simbol-simbol yang digunakan dalam *DFD* terdapat dalam tabel 2.1 berikut :

Tabel 1. Simbol dalam *DFD*

No.	Simbol	Nama	Fungsi
1.		Proses	Mentransformasikan data secara umum
2.		Entiti luar	Merupakan sumber atau tujuan dari aliran data dari atau ke sistem
3.		Berkas atau tempat penyimpanan	Merupakan komponen untuk menyimpan data/ <i>file</i>
4.		Aliran data	Menunjukkan aliran data dari satu proses ke proses lainnya

Sumber : Jogiyanto Hartono, MBA, Ph.D.(2004:754)

Entity Relationship Diagram

ERD (*Entity Relationship Diagram*) adalah model atau rancangan yang digunakan untuk merancang *database*, sehingga memudahkan dalam menggambarkan data yang memiliki keterkaitan atau hubungan dalam bentuk desain. Melalui diagram ini, struktur *database* yang

dibangun menjadi lebih terorganisir dan tampak lebih jelas.

Berikut penjelasan tentang komponen-komponen *ERD* beserta simbol-simbolnya:

1. Entitas (*Entity*)

Entitas adalah objek atau konsep yang penting bagi sistem dan memiliki data yang perlu disimpan. Entitas bisa berupa orang, benda, tempat atau konsep yang relevan. Simbolnya digambarkan dengan persegi panjang atau kotak. Nama entitas biasanya ditulis di dalam kotak tersebut. Contoh entitas: "Mahasiswa," "Produk," "Karyawan."

2. Hubungan/Relasi (*Relationship*)

Hubungan menggambarkan bagaimana dua atau lebih entitas saling berhubungan dalam sistem. Relasi ini menghubungkan entitas dan menunjukkan bagaimana data antar entitas saling terkait. Simbolnya digambarkan dengan bentuk belah ketupat (diamond) dan diberi nama yang menunjukkan tipe relasi, misalnya "Mengikuti" (antara Mahasiswa dan Kelas) atau "Memiliki" (antara Karyawan dan Proyek).

3. Atribut (*Attribute*)

Atribut adalah karakteristik atau properti yang menjelaskan entitas. Setiap entitas memiliki atribut yang menggambarkan detail tentang entitas tersebut. Contoh atribut untuk entitas "Mahasiswa" bisa berupa NIM, Nama dan Alamat. Simbol atribut digambarkan dengan elips (oval) yang terhubung ke entitas yang dimaksud.


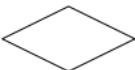



4. Garis Relasi (*Relationship Line*)

Garis relasi menghubungkan entitas dengan relasi atau entitas dengan atribut. Garis ini menunjukkan adanya keterkaitan antara entitas dan hubungan dalam diagram. Simbolnya digambarkan dengan garis lurus yang menghubungkan persegi panjang (entitas) dengan belah ketupat (relasi) atau elips (atribut) dengan entitas.

5. Entitas Lemah (*Weak Entity*)

Entitas lemah adalah entitas yang tidak dapat berdiri sendiri dan bergantung pada entitas lain untuk eksis. Entitas lemah biasanya tidak memiliki atribut kunci sendiri dan memerlukan "entitas kuat" untuk identifikasinya. Simbolnya digambarkan dengan entitas lemah digambarkan dengan persegi panjang ganda (*double rectangle*) dan relasi dengan entitas kuat dihubungkan melalui belah ketupat dengan garis ganda.

Tabel 2. Simbol dalam *ERD*

No	Simbol	Nama	Keterangan
1		Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber daya atau transaksi yang <i>field-fieldnya</i> dipergunakan dalam aplikasi program
2		Hubungan atau Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya
3		Atribut	Atribut adalah karakteristik dari sebuah entitas
4		Garis Relasi	Menunjukkan hubungan (keterkaitan) antar entitas
5		Entitas Lemah	Entitas yang kemunculannya tergantung dari entitas lain yang lebih kuat

Sumber : W. W. Widiyanto

WWW (World Wide Web)

Menurut Virna Miftahuljannah *World Wide Web* (WWW) adalah salah satu koleksi dokumen HTML milik pribadi atau perusahaan yang berisi informasi dan berada dalam *Web Server* dan dapat diartikan sebagai kumpulan dari halaman situs yang terangkum dalam sebuah *domain* atau *subdomain* dan berada didalam.

PHP (Hypertext Preprocessor)

Menurut Solichin, *PHP* merupakan salah satu bahasa pemrograman berbasis *web* yang ditulis oleh dan untuk pengembang *web*. *PHP* pertama kali dikembangkan oleh Rasmus Lerdorf, seorang pengembang *software* dan anggota tim *Apache* dan dirilis pada akhir tahun 1994. *PHP* dikembangkan dengan tujuan awal hanya untuk mencatat pengunjung pada *website* pribadi Rasmus Lerdorf *PHP* merupakan Bahasa pemrograman berbasis *web* yang dibuat secara khusus untuk membangun aplikasi berbasis *web*. Selain tersedia secara gratis, *PHP* juga mudah dipelajari oleh siapapun .

MySQL

Menurut (Elgamar, 2020), *MySQL* menjadi *database* yang paling populer saat sekarang ini *MySQL* merupakan *database* yang memiliki tiga tipe data bersifat relasional yang berarti *MySQL* memiliki cara dalam menyimpan datanya dalam berbentuk tabel-tabel yang saling terhubung. *Database My Structure Language (MySQL)* berfungsi dalam mengelola *database* menggunakan bahasa Struktur *Query Language (SQL)* .

HTML (Hypertext Markup Language)

HTML merupakan singkatan *Hypertext Markup Language* yaitu bahasa standar *web* yang dikelola penggunaannya oleh *W3C (World Wide Web Consortium)* berupa tag-tag yang menyusun setiap elemen dari *website*. *HTML* berperan sebagai penyusun struktur halaman *website* yang menempatkan setiap elemen *website* layout yang diinginkan. *HTML* biasanya disimpan dalam sebuah file berekstensi *.html*. Untuk mengetikkan skrip *HTML* dapat menggunakan *text editor* seperti *Notepad* sebagai bentuk paling sederhana atau *text editor* khusus yang dapat mengenali setiap unsur skrip *HTML* dan menampilkannya dengan warna yang berbeda sehingga mudah di baca seperti *Notepad++*, *Sublime Text* dan masih banyak lagi aplikasi lain yang sejenisnya

Aplikasi

Menurut (Dongoran et al., 2020) Aplikasi merupakan program yang dapat dipakai oleh pengguna buat melaksanakan berbagai macam tugas secara khusus, contohnya seperti untuk dokumentasi, editing gambar ataupun juga dalam pembuatan laporan. Serta aplikasi yang dibangun oleh orang lain ataupun *programmer* yang mempunyai tujuan tertentu buat melaksanakan tugas-tugas tertentu. Sedangkan menurut (Novendri et al, 2019) yang berpendapat bahwa aplikasi ialah program siap digunakan yang teruntuk malakukan sesuatu guna bagi pengguna. Sistem aplikasi merupakan bagian kelas dari fitur lunak (*software*) yang menggunakan keahlian komputer secara langsung guna melakukan suatu tugas yang diidamkan oleh *user*. Umumnya ketimbang dengan fitur lunak sistem yang mengintegrasikan bermacam keahlian komputer, tetapi tidak secara langsung mengaplikasikan keterampilan tersebut buat mengerjakan suatu tugas yang menguntungkan pengguna (Ikhsanudin, 2022).

Database

Database merupakan kumpulan informasi yang disimpan di dalam komputer yang secara sistematis agar dapat diperiksa menggunakan suatu program komputer. Secara konsep *Database* berupa kumpulan dari data-data yang membentuk file yang antara satu dan lainnya terhubung dengan tatacara tertentu untuk membentuk data baru.

III. METODE

Metode Analisis Data

Adapun metode analisis data yang dibuat oleh penulis pada penelitian yang berjudul PERANCANGAN APLIKASI *WORK ORDER* PADA PT. MAKMUR BINTANG PLASTINDO

BERBASIS *WEB* melalui aplikasi ini di harapkan dapat meningkatkan kinerja dan komunikasi tim teknisi dalam melakukan perbaikan pada mesin di PT. Makmur Bintang Plastindo.

Dengan adanya suatu Perancangan Aplikasi *WORK ORDER* ini, peneliti melakukan metode analisis data sebagai perancangan programnya yang berbasiskan *web*.

1. Pada tahap awal, peneliti melakukan observasi terhadap formulir *work order* dan proses kerja teknisi dalam menangani perbaikan mesin, untuk mengevaluasi apakah formulir tersebut efektif digunakan di lapangan. Berdasarkan wawancara dengan para teknisi, mereka mengeluhkan bahwa kertas surat perintah kerja sering rusak dan prosesnya dianggap terlalu memakan waktu.
2. Pada tahap kedua, peneliti akan mengelompokkan dan menyusun data berdasarkan bagian-bagian yang relevan, khususnya pada tim teknisi. Setelah data terkumpul, peneliti akan melakukan analisis untuk memastikan perencanaan proses perancangan aplikasi *work order* dapat dilakukan dengan baik. Aplikasi berbasis *web* ini diharapkan berguna bagi tim teknisi dalam menangani perbaikan, sehingga proses perbaikan menjadi lebih mudah, terorganisir dan menghindari terjadinya tumpang tindih pekerjaan antar teknisi.

Analisa Sistem Yang Berjalan

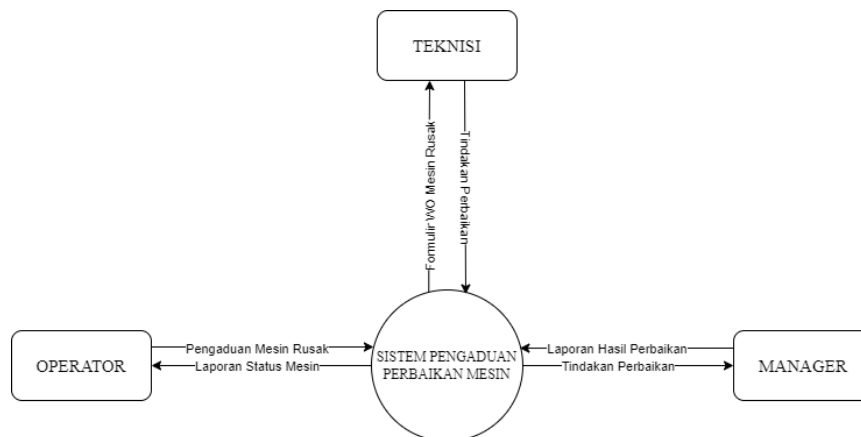
Prosedur pengolahan data

Adapun metode analisis data yang dibuat oleh penulis pada penelitian yang berjudul PERANCANGAN APLIKASI *WORK ORDER* PADA PT. MAKMUR BINTANG PLASTINDO BERBASIS *WEB* sebagai berikut :

Dengan adanya suatu perancangan aplikasi *work order* proses produksi ini, peneliti melakukan metode analisis data sebagai perancangan programnya yang berbasiskan *web*.

Diagram konteks

Diagram konteks adalah suatu gambaran keseluruhan dari proses sistem secara garis besar berdasarkan prosedur kerja yang ada dalam sistem yang dapat dilihat pada gambar 3.6 dibawah ini.



Gambar 2. Diagram Konteks Sistem Yang Sedang Berjalan

Diagram konteks yang ditampilkan adalah representasi tingkat tinggi dari sistem "*Work order* berbasis *web*" yang menunjukkan hubungan antara sistem pengaduan menggunakan formulir *work order* yang sedang berjalan dengan entitas *eksternal* (aktor) yang berinteraksi dengannya.

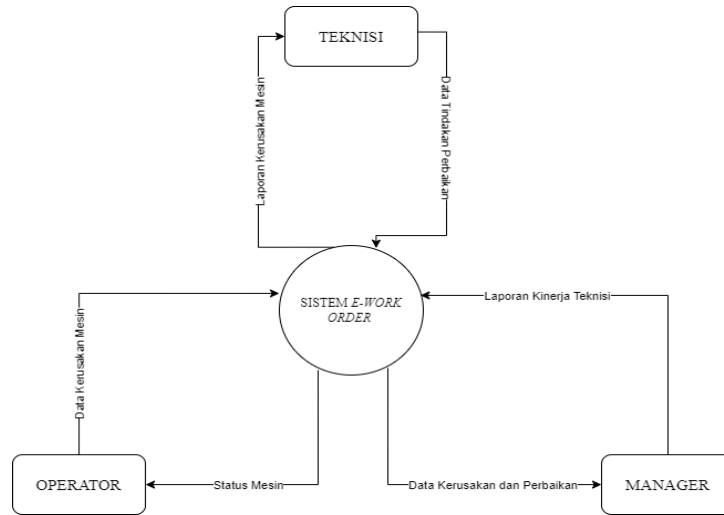
Perancangan Sistem

Perancangan Aplikasi merupakan suatu persiapan atau perbaikan dari sistem yang sedang berjalan dan sebagai tindak lanjut dari penyelesaian masalah berdasarkan hasil evaluasi sistem. Sistem yang dirancang merupakan usulan perancangan sistem untuk memperbaiki proses pengaduan dan perbaikan pada PT. Makmur Bintang Plastindo.

Data flow diagram

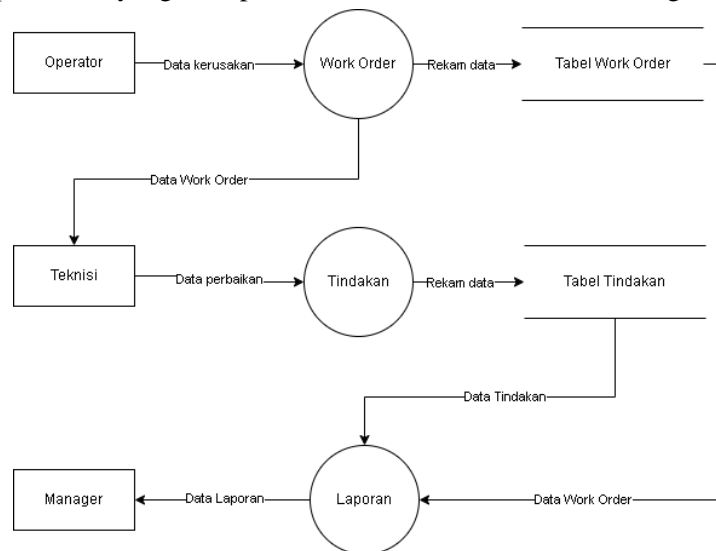
Data Flow Diagram adalah pembuatan model yang memungkinkan profesional sistem untuk

menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur secara komputerisasi. Data Flow Diagram (DFD) level 0 perancangan *work order* pada PT. Makmur Bintang Plastindo berbasis *web* dapat dilihat pada gambar 3.7 di bawah ini.



Gambar 3. DFD level 0

Berikut adalah alur data yang dihasilkan oleh DFD level 0. *Operator* mengirimkan pengaduan *work order* beserta foto kerusakan ke sistem. Permintaan ini diterima oleh teknisi yang kemudian mengelola data order tersebut. Teknisi menerima pengaduan tentang kerusakan mesin, melakukan perbaikan, memperbarui status mesin dan mengirimkan informasi terbaru kembali ke sistem. *Operator*, teknisi dan *manager* akan menerima pembaruan status terkini mengenai mesin yang terkait. Setelah perbaikan selesai, *operator* dapat kembali menjalankan mesin untuk memproduksi karung goni. Selanjutnya, *manager* dapat membuat laporan berdasarkan data perbaikan yang terdapat dalam sistem untuk disusun sebagai laporan bulanan.



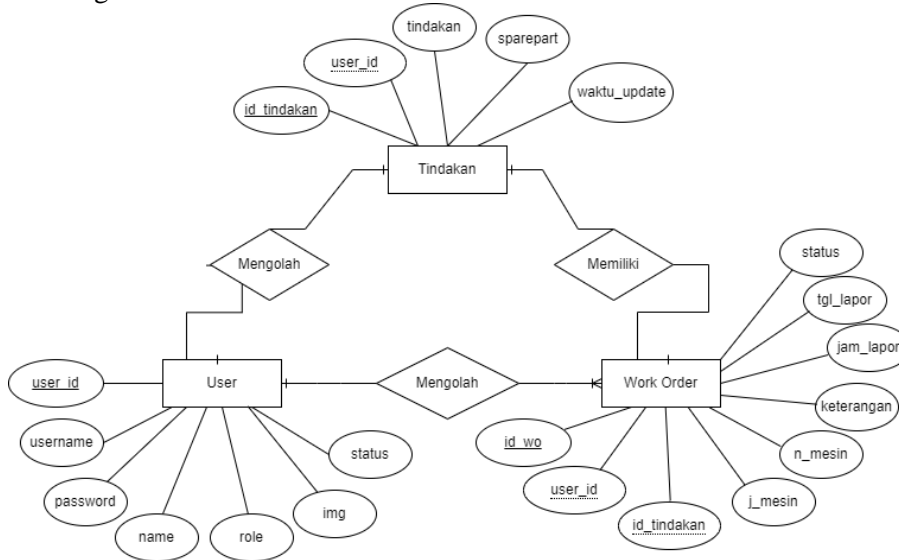
Gambar 4. DFD Level 1

Dalam proses pengaduan, *operator* mengirimkan data kerusakan melalui formulir *work order* pada aplikasi, di mana data tersebut disimpan dalam tabel *work order* di *database*. Tugas kemudian diteruskan ke teknisi dalam bentuk formulir *work order* yang menjadi acuan teknisi untuk melakukan perbaikan pada mesin yang rusak. Setelah perbaikan selesai, teknisi mengirimkan data tindakan perbaikan yang direkam dalam tabel tindakan di *database*. Laporan yang memuat informasi dari tabel *work order* dan tabel tindakan kemudian dikirimkan ke *manager* sebagai bukti kerusakan dan perbaikan yang telah dilakukan. Hal ini memudahkan

manager untuk memantau kinerja teknisi dan menyusun laporan bulanan yang akan dikirimkan ke atasan.

Entity-Relationship Diagram

ERD (*Entity-Relationship Diagram*) adalah alat visual yang digunakan dalam perancangan basis data untuk menggambarkan struktur dan hubungan antar entitas dalam sistem. ERD membantu perancang basis data untuk memodelkan data yang akan disimpan serta bagaimana data tersebut saling berinteraksi.



Gambar 5. Entity-Relationship Diagram

IV. HASIL DAN PEMBAHASAN

Analisa Kebutuhan Sistem

Berdasarkan hasil analisis dan perancangan yang telah dilakukan, diperlukan implementasi sistem informasi yang telah dirancang untuk menguji dan mengevaluasi komponen-komponen utama yang dikembangkan. Implementasi ini berfungsi sebagai tolok ukur keberhasilan serta alat untuk menguji kinerja dan efektivitas program yang telah dibuat. Proses implementasi sistem mencakup penerapan secara menyeluruh, baik dari aspek perangkat keras maupun perangkat lunak, guna memastikan sistem dapat berfungsi dengan optimal sesuai dengan kebutuhan.

Analisa Perancangan Sistem

Setelah desain sistem selesai, langkah selanjutnya adalah mengembangkan kode program sesuai dengan rancangan yang telah dibuat. Dalam implementasinya, penelitian ini memanfaatkan XAMPP versi 5.6.40 sebagai server lokal serta menggunakan Visual Studio Code sebagai perangkat lunak text editor untuk penulisan kode.

Hasil Dan Pembahasan

Tahapan implementasi antarmuka sistem melibatkan pembuatan tampilan antarmuka sesuai dengan desain yang telah dirancang sebelumnya.

Implementasi Aplikasi

Implementasi aplikasi *work order* pada PT. Makmur Bintang Plastindo disambut dengan hangat oleh divisi terkait, hal ini melibatkan beberapa langkah penting untuk memastikan keberhasilan penggunaan sistem. Proses ini dimulai dengan instalasi aplikasi di infrastruktur IT perusahaan, di mana tim pengembang melakukan pengujian fungsionalitas untuk memastikan semua fitur berjalan dengan baik. Setelah aplikasi siap, dilakukan pelatihan untuk pengguna utama yang mencakup operator, teknisi dan *manager*, agar mereka memahami cara menggunakan sistem dengan efektif.

Dalam rangka memastikan *stakeholder* memahami manfaat dan fungsi aplikasi, peneliti bekerja sama dengan pihak PT. Makmur bintang Plastindo untuk mengadakan sesi penyuluhan. Dalam sesi ini, *manager* diperkenalkan dengan antarmuka pengguna dan fitur-fitur penting dari aplikasi seperti pengelolaan pengaduan, pembaruan status perbaikan dan pelaporan. *Manager* juga diajarkan cara mengakses laporan yang dihasilkan oleh sistem, sehingga mereka dapat mengambil keputusan berdasarkan data yang akurat dan terkini.

Setelah mengikuti sesi penyuluhan, *manager* diharapkan dapat mengadopsi dan menerapkan aplikasi dalam rutinitas sehari-hari. Penerimaan program oleh *manager* sangat penting, karena mereka akan menjadi salah satu pengguna utama yang bertanggung jawab untuk mengawasi proses perbaikan dan pemeliharaan. Dengan pemahaman yang baik mengenai aplikasi, *manager* dapat mengoptimalkan penggunaan sistem untuk meningkatkan efisiensi operasional perusahaan. Selain itu, dukungan dari *manager* akan memotivasi anggota tim lainnya untuk aktif menggunakan aplikasi dan berkontribusi dalam proses manajemen pemeliharaan yang lebih baik.

V. KESIMPULAN

Kesimpulan yang dapat diambil oleh penulis dalam penyusunan Tugas Akhir ini adalah sebagai berikut :

1. Perancangan aplikasi *work order* berbasis *web* di PT. Makmur Bintang Plastindo berhasil meningkatkan efisiensi dalam proses perbaikan mesin. Sistem ini mengotomatiskan alur kerja yang sebelumnya dilakukan secara manual dengan kertas dan pena, sehingga mempermudah pelaporan kerusakan, mempercepat proses tindak lanjut oleh teknisi dan memperjelas alur komunikasi antar tim yang secara keseluruhan meningkatkan produktivitas dan efektivitas kerja.
2. Dengan menggunakan sistem berbasis *web*, waktu penanganan pengaduan dan pelaporan kerusakan mesin berkurang secara signifikan. Sistem ini memungkinkan pemantauan status perbaikan secara real-time yang memudahkan *operator*, teknisi dan *manager* untuk mengelola perbaikan, terutama di area perusahaan yang luas. Hasilnya, proses perbaikan menjadi lebih terkoordinasi dan responsif.

VI. REFERENSI

- Abdussalaam, F., & Ramadhan, M. M. (2019). Perancangan sistem informasi work order dengan metode iteratif menggunakan framework codeigniter (Studi Kasus: CV Sirna Miskin Bandung). *Jurnal E-Komtek (Elektro-Komputer-Teknik)*, 35-48.
- Ikhsanudin, M. (2022). *Aplikasi Penjualan Sparepart Motor pada Toko MN Motor berbasis Website*. Doctoral dissertation, Prodi Teknik Informatika.
- Saputra, A., Imamuddin, A., & Sukamto, P. (2020). RANCANG BANGUN APLIKASI SISTEM PENJUALAN CASE STUDY: PT. X. *Infotech: Jurnal Informatika Teknologi*, 78-86.
- Sidik, A. (2019). *Teori, strategi dan evaluasi merancang website dalam perspektif desain*. Kalimantan: Repository Universitas Islam Kalimantan.
- Umiga, M. (2022). Perancangan user interface (UI) dan user experience (UX) aplikasi e-learning studi kasus SMK N Jenawi dengan pendekatan user centered design. *Jurnal Cakrawala Informasi*, 56-62.
- Vella, C. D., Yubarda, E., Jannah, M., & Surya, C. (2022). SURAT PERINTAH KERJA KARYAWAN RIGLESS PADA PT. PRAKARSA PRAMANDITA BERBASIS WEB. *JSR: Jaringan Sistem Informasi Robotik*, 269-283.