



Penyusunan *Database* Berbasis *Website* untuk Mendeteksi Kerusakan dan Langkah Perbaikan pada Mesin Diesel

Alexander Puruhita Dwinanta Permana^{1*)}, Hartono Yudo¹⁾, Kiryanto¹⁾

¹⁾Laboratorium Teknologi Sistem dan Permesinan Kapal

Departemen Teknik Perkapalan, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Soedarto, SH, Kampus Undip Tembalang, Semarang, Indonesia 50275

^{*)}e-mail : alexanderruru@students.undip.ac.id

Abstrak

Perkembangan teknologi di era globalisasi seperti saat ini, dengan memanfaatkan teknologi sistem informasi dapat membantu dalam kebutuhan akan sistem yang bisa mendeteksi kerusakan mesin diesel secara cepat dan akurat. Tujuan penelitian ini adalah membuat database berbasis website dengan menggunakan algoritma brute force sebagai alternatif pertama dalam proses mendeteksi masalah atau kerusakan yang kemungkinan terjadi pada mesin diesel. Metode yang digunakan pada mesin pencarian penelitian ini adalah algoritma brute force karena kemampuannya dalam melakukan pencarian menyeluruh dan menemukan data yang tepat. Data dari penelitian ini diambil dari wawancara dengan sumber yang sudah berspesialisasi di bidang mesin diesel, dan dari data yang sudah diperoleh akan dimasukkan ke dalam database. Website ini bekerja dengan cara menarik kesimpulan dari data gejala yang di-input oleh user dan memunculkan data Symptoms, Damages, Solutions terkait dengan data gejala yang di-input. Hasil penelitian ini menghasilkan web dengan integrasi database berisikan jumlah gejala sebanyak 9 gejala dan sebanyak 147 jenis kerusakan yang kemungkinan terjadi. Keakuratan dalam menemukan keyword atau kata kunci dari gejala menjadikan pencari kerusakan ini berguna bagi user dalam melakukan perawatan dan perbaikan mesin diesel.

Kata Kunci : Website, Mesin Diesel, Database, Pendeteksi Kerusakan

1. PENDAHULUAN

Mesin yang menjadi daya jual utama kapal tentunya harus dirawat dan dijaga kondisi dan performanya, karena bila kinerja mesin tidak atau kurang efisien tentu tugas niaga kapal menjadi terganggu dan terhambat. Jadi ketika terjadi kerusakan atau sudah jatuh waktu untuk *maintenance* bagi mesin, kapal harus segera diperbaiki dan dilakukan pengecekan pada mesin kapal. Banyak dari kerusakan mesin saat pengoperasian dikarenakan oleh kelalaian pada proses reparasi, yang membuat kegiatan operasional terganggu bahkan terhenti [1]. Maka dari itu proses pengecekan dan identifikasi masalah inilah di mana pakar permesinan (mekanik) diperlukan, karena kerusakan mesin semakin banyak dan beragam terkadang mekanik tidak ditempatkan untuk menyelesaikan masalah tersebut.

Kerusakan pada mesin sering kali disebabkan oleh proses reparasi yang kurang maksimal, pada saat ini galangan kapal atau pemilik kapal sulit untuk mendeteksi kerusakan yang terjadi pada mesin kapal [2]. Biaya juga menjadi salah faktor pertimbangan untuk melakukan perbaikan terhadap kerusakan mesin yang belum dapat bisa dipastikan letak permasalahannya, oleh karena itu diperlukan sistem yang dapat membantu mencari kerusakan yang terjadi pada mesin kapal untuk menghemat biaya dan waktu dalam proses reparasi kapal.

Website mesin pencarian bisa digunakan sebagai alternatif pertama dalam proses pengidentifikasian masalah atau kerusakan yang terjadi pada mesin. Mesin pencarian ini berisikan sistem yang terdapat *database* di dalamnya, di mana *database* tersebut menyimpan pengetahuan seorang pakar yang bisa dipanggil dengan memasukkan fakta-fakta yang terjadi pada objek, maka pengetahuan pakar tersebut akan keluar

sesuai dengan aturan-aturan yang sudah ditetapkan pada mesin pencarian tersebut [3].

Pemindahan pengetahuan pakar dapat dilakukan dengan wawancara atau dari buku yang dibuat oleh seorang pakar. Pada penelitian tentang aplikasi *decision support system* untuk *troubleshooting* pada operasional genset di kapal, mendapatkan hasil media *web* pada penelitian tersebut dapat digunakan untuk mengetahui kerusakan genset terutama pada mesin penggerak generator [4].

Dalam kasus mesin diesel ini, pakar-pakar tersebut ada di galangan atau di antara para *crew* kapal di bagian permesinan dan mekanik-mekanik di bengkel mesin galangan. *Website* mesin pencarian ini dirancang dan dibuat untuk menentukan kerusakan pada mesin kapal dengan memasukkan fakta-fakta mengenai gejala yang ada pada mesin.

Penelitian ini *website* dikembangkan dengan *html* sebagai struktur utama halaman *website*, *css* sebagai bahasa pemrograman yang mengatur gaya dari *website* ini, serta *javascript* sebagai bahasa pemrograman untuk bagian *front end* (bagian luar, *client side*) dan *back end* di mana mengatur masukan atau keluaran dari program. Serta metode yang digunakan dalam fitur pencarian langsung adalah metode algoritma *brute force*.

Beberapa penelitian terdahulu yang menggunakan algoritma *brute force* dalam sebuah sistem pencarian, dalam penelitian mengenai implementasi algoritma *brute force* pada sistem pertanian di balai desa. Perancangan sebuah sistem informasi berbasis *website* memudahkan *user* dalam mendapatkan informasi tentang pertanian yang diperlukan, dan penerapan algoritma *brute force* terbukti dapat menampilkan hasil pencarian yang diinginkan [5].

Penelitian lain mengenai pencarian berita pada *website* portal berita menggunakan algoritma *brute force*. Pencarian pada mesin ini akan menampilkan hasil pencarian berdasarkan *keyword* atau fakta yang ingin dicari dan akan memunculkan kalimat yang paling banyak dicari oleh orang-orang berdasarkan *website*. Sejumlah berita akan masuk ke dalam mesin, ini karena *keyword* harus dicocokkan menggunakan algoritma *brute force*, di mana algoritma ini akan membandingkan pola karakter atau teks. Hasil dari pencocokannya akan dibandingkan dengan hasil pencari sehingga pengguna dapat dengan mudah menemukan kata kunci atau *keyword* dan informasi yang mereka cari di *website* berita informasi [6].

Berdasarkan penelitian tentang aplikasi sistem informasi *smart register online* berbasis android menggunakan algoritma *brute force*. Mendapatkan hasil bahwa menggunakan algoritma *brute force* ini

berhasil, pengujiannya sesuai dengan keinginan dan layak digunakan dalam fase pendaftaran calon mahasiswa baru [7].

Berdasarkan landasan tersebut, penelitian ini akan berfokus pada kerusakan yang kemungkinan terjadi pada mesin diesel dengan tujuan membuat sistem ini sebagai langkah awal untuk membantu memperbaiki mesin diesel tanpa perlu bantuan teknisi profesional karena *user* dapat mengetahui secara langsung dan *realtime* permasalahan yang terjadi pada mesin kapal.

2. METODE

Penelitian ini diawali dengan pembuatan *database* sebagai gambaran data apa saja yang diperlukan dan menjadi topik pembahasan pada saat wawancara dengan para pakar. *Database* ini yang akan menjadi penampung dari data-data hasil dari wawancara dengan para pakar. Selanjutnya, dibuat *website* sebagai penghubung *database* dengan *user* dalam penggunaan sistem pendeteksi kerusakan. Penelitian ini menggunakan bahasa *Hyper Text Markup Language* (HTML), *Cascading Style Sheet* (CSS), dan *Javascript* yang merupakan elemen pendukung pembuatan *website* dan *javascript* untuk bahasa pemrograman bagian *backend* (*server side*).

Penelitian ini menggunakan algoritma *string matching* yaitu algoritma *brute force* sebagai metode yang dipakai *search engine* dalam mencari data. Penerapan algoritma *brute force* terletak pada proses sistem pendeteksi kerusakan ini dalam mencari kerusakan berdasarkan gejala yang di-*input* oleh *user* ke dalam fitur *search* sistem. Hal ini dikarenakan menurut banyak peneliti *string matching* adalah pencarian sebuah pola pada sebuah teks, *string matching* sendiri merupakan algoritma yang ditujukan untuk mencocokkan *sub string* dan *string* utama [8].

2.1. Algoritma Brute Force

Algoritma digunakan untuk menggambarkan metode pemecahan yang efektif, terbatas, determinis, dan cocok untuk sebuah implementasi dalam sebuah program [9].

Metode yang dipilih dalam penelitian ini adalah metode algoritma *brute force* dalam proses pencarian data di dalam *database*. Algoritma *brute force* ini merupakan pendekatan secara langsung dalam memecahkan permasalahan berdasarkan pernyataan dalam permasalahan dan arti cara yang dilibatkan. Algoritma *brute force* digunakan untuk mengecek pada setiap kedudukan karakter dalam teks mulai dari yang paling awal hingga terakhir. Algoritma ini dapat melakukan penyelesaian

masalah dengan mudah, kompleks, sederhana, dan jelas. Algoritma *brute force* digunakan dalam penelitian ini dikarenakan algoritma ini merupakan pendekatan *straightforward* (secara yang langsung) dan jelas untuk memecahkan masalah dan biasanya didasarkan pada *problem statement*. Penggunaan algoritma *brute force* pada penelitian ini berfokus pada bagaimana algoritma ini dapat mencari dan memunculkan data-data berdasarkan *keyword* yang dimasukkan dari *database* yang ada [10].

Penerapan algoritma *brute force* pada penelitian ini ada pada bagian *search engine* dari *website*, di mana algoritma *brute force* diperlukan dalam memeriksa data gejala pada data SDS yang akan diperlihatkan, sehingga *user* tidak perlu untuk menuliskan keseluruhan data gejala yang ingin dicari, hanya perlu memasukkan kata kunci atau *keyword*.

2.2. Website

Penelitian ini menggunakan *website* sebagai sarana bagi para *user* untuk bisa menggunakan sistem pencarian ini. Bahasa yang digunakan untuk membuat *web* ini adalah, *Hyper Text Markup Language* (HTML), *Cascading Style Sheet* (CSS), dan *Javascript* yang merupakan elemen pendukung pembuatan *website* dan *javascript* untuk bahasa pemrograman bagian *backend* (*server side*).

Website ini dirancang sebagai *search engine* untuk mencari kerusakan yang kemungkinan terjadi pada mesin diesel. Tujuan dari penelitian ini adalah bagaimana *user* bisa dengan mudah mencari kerusakan yang kemungkinan terjadi pada mesin diesel mereka dengan memasukkan *keyword* berupa gejala atau *symptoms* yang terlihat pada mesin diesel. Di dalam *website* juga ada fitur untuk menambahkan data SDS (*Symptom, Damage, Solution*) yang dapat dilakukan oleh admin.

2.3. Mesin Diesel dan Sistem Tambahan

Penelitian ini berfokus pada kerusakan yang terjadi pada mesin diesel beserta sistem dan peralatan tambahan yang membantu dalam pengoperasian. Sistem tambahan tersebut meliputi, sistem minyak pelumas, sistem *start*, sistem pendingin, sistem bahan bakar, dan sistem *intake* dan *exhaust*.

2.4. Objek Penelitian

Objek pada penelitian ini adalah mekanik dan pakar permesinan di galangan PT. Yasa Wahana Tirta Samudera. Data juga diambil dari KKM dan

orang permesinan di kapal *tugboat* Patra Tunda 3153.

2.5. Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan di galangan PT. Yasa Wahana Tirta Samudera di Semarang. Data tersebut dikumpulkan melalui wawancara dengan para pihak dari instansi di atas yang memiliki keahlian atau spesialisasi di bagian permesinan. Data yang dikumpulkan berupa data SDS (*Symptom, Damage, Solution*) seperti, kerusakan yang biasa terjadi pada mesin diesel, gejala yang dapat dilihat bila terjadi kerusakan tersebut, solusi yang dapat dilakukan untuk menanggulangnya, serta kondisi optimal dari bagian yang mengalami kerusakan. Data pendukung juga dibutuhkan untuk melengkapi data primer, seperti jurnal, buku, artikel, *software*, serta data-data dari penelitian sebelumnya untuk mempelajari permasalahan dan solusinya.

Berdasarkan data yang sudah terkumpul, dibuat sebuah *database* yang berisikan beberapa tabel yang akan dimasukkan nilai dari data yang sudah terkumpul. Berikut adalah keterangan tentang tabel yang ada di dalam *database* :

Tabel 1. Kerusakan_2tak

No	Nama Kolom	Tipe	Panjang Karakter
1	<i>Id</i>	<i>Int</i>	11
2	Gejala	<i>Varchar</i>	191
3	Kerusakan	<i>Varchar</i>	191
4	Solusi	<i>Varchar</i>	191
5	Bagian	<i>Varchar</i>	191

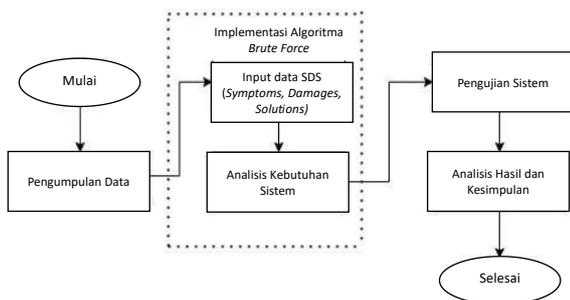
Tabel 2. Kerusakan_4tak

No	Nama Kolom	Tipe	Panjang Karakter
1	<i>Id</i>	<i>Int</i>	11
2	Gejala	<i>Varchar</i>	191
3	Kerusakan	<i>Varchar</i>	191
4	Solusi	<i>Varchar</i>	191
5	Bagian	<i>Varchar</i>	191

Tabel 1 dan 2. kolom terdiri dari *id*, gejala, kerusakan, solusi, dan bagian yang rusak. *Id* merupakan *primary key* yang digunakan sebagai pembeda yang akan disimpan, Gejala berisikan fakta tentang gejala yang terlihat dari mesin yang rusak, Kerusakan berisikan data kerusakan apa yang terjadi ketika gejala terkait terlihat, solusi berisi langkah yang perlu dilakukan ketika gejala tersebut terjadi dan terdapat juga kondisi prima dari bagian yang rusak tersebut, terakhir Referensi yang berisi data tentang sumber tertulis yang dapat menjadi referensi utama dari tindakan SDS tersebut.

2.6. Pengolahan Data

Perancangan *website* pada penelitian ini menggunakan *waterfall model*, di mana setiap langkah diselesaikan sepenuhnya terlebih dahulu secara bertahap sebelum melanjutkan ke langkah selanjutnya. Pengembangan *website* menggunakan model *waterfall* karena sesuai dengan langkah yang akan dilakukan dimana proses tersebut mencakup, analisis, desain atau perancangan, pengodean, dan pengujian. Berikut adalah gambaran proses pengembangan *website* dengan menggunakan *waterfall model* :



Gambar 1. Model *Waterfall* yang digunakan

- a) Analisis
Pengembang harus mengetahui apa saja yang akan dibutuhkan oleh pengguna *website*. Sehingga pada proses ini pengembang harus paham hal-hal apa saja yang berguna dan diperlukan oleh pengguna *website* nanti.
- b) Desain
Tahap selanjutnya desain. Proses ini mengharuskan pengembang untuk memikirkan dan merancang desain dari *website*, sehingga tampilannya dapat memudahkan *user* dalam menggunakan *website*.
- c) Coding
Proses selanjutnya dalam model ini adalah pembuatan kode *website* secara mekanis. Tahap ini mengharuskan pengembang untuk melakukan penulisan *coding*, pada penelitian ini menggunakan *HTML*, *CSS*, dan *javascript* agar perancangan fungsi *website* sesuai apa yang dianalisis, dan tampilannya sesuai dengan yang didesain.
- d) Pengujian
Tahap terakhir ini adalah tahap yang menentukan keberhasilan keseluruhan model. Pengujian ini dilakukan untuk mengetahui hasil akhir dari segala proses

sebelumnya apakah sudah sesuai dengan yang diharapkan atau tidak.

3. HASIL DAN PEMBAHASAN

Pembuatan penelitian ini mengambil data dari beberapa sumber atau pakar di kalangan PT. Yasa Wahana Tirta Samudera, serta beberapa bengkel yang berspesialisasi pada mesin diesel. Data tersebut merupakan data mengenai SDS mesin diesel yang berupa, data kerusakan yang biasa terjadi pada mesin diesel, gejala yang terlihat pada mesin yang mengalami kerusakan tersebut, solusi dari kerusakan tersebut, serta kondisi optimal dari bagian yang mengalami kerusakan di atas.

3.1. Data Kerusakan Kapal

Penelitian mengenai Penyusunan *Database* Berbasis *Website* Untuk Mendeteksi Kerusakan dan Langkah Perbaikan Pada Mesin Diesel ini menemukan fakta bahwa kerusakan umum yang biasa terjadi pada bagian mesin penggerak utama kapal, dan analisis kerusakannya dikaitkan, dibandingkan, dan dijelaskan kerusakan dan kondisi yang ada dengan teori dan *manual book* mesin kapal. Penelitian ini mengaitkan mesin diesel kapal *tugboat* Patra Tunda 3153 yang sedang bersandar di kalangan PT. Yasa Wahana Tirta Samudera. Berikut adalah tabel jumlah gejala kerusakan serta solusi yang ditemukan :

Tabel 3. Jenis Gejala dan Jumlah Solusi

Gejala Kerusakan	Jumlah Solusi Perbaikan
Start Fail	15 solusi
Mesin Tiba-Tiba Mati	15 solusi
Mesin <i>Overheat</i> Peningkatan Suhu Mesin	21 solusi
Menghasilkan Suara Tidak Normal	22 solusi
Gas Buang Tidak Normal	12 solusi
Konsumsi Bahan Bakar Berlebih	14 solusi
Penunjuk Minyak Lumas Tidak Normal	5 solusi
Penurunan Efisiensi Mesin	23 solusi
Putaran Mesin (RPM) Tidak Normal	20 solusi
Jumlah Total Solusi Perbaikan	147 solusi

3.2. Alur Pencarian

Alur pencarian pada *website* ini dirancang sedemikian rupa agar *user* dapat mencari kerusakan dari mesin diesel dengan meng-*input* gejala yang terlihat. *User* dapat melakukan pencarian tanpa perlu melakukan pendaftaran atau masuk menggunakan *e-mail* dan sudah bisa melakukan pencarian. Pada *website* ini terdapat dua fitur dalam melakukan pendeteksian kerusakan, yaitu dengan fitur *search* dan fitur pemilihan gejala. Berikut adalah alur untuk melakukan pencarian kerusakan dengan fitur *search* :

1. *User* Memilih Tipe Mesin
User dapat langsung menggunakan *search engine* tanpa perlu mendaftar. Dari beranda *website*, *user* bisa memilih tipe mesin (2 Tak atau 4 Tak) apa yang ingin dicari data SDS-nya.
2. Mulai Mencari Kerusakan
Setelah memilih tipe mesin, *user* diberikan sebuah *form* untuk memasukkan data gejala apa yang terlihat pada mesin diesel.
3. Memilih Data yang Cocok
Halaman pencarian dilanjutkan ketika *user* telah memasukkan data gejala, dan akan muncul berbagai data yang sesuai dengan apa yang telah dimasukkan oleh *user*. Data SDS akan muncul setelah *user* memilih data yang kemungkinan berkaitan dengan kerusakan yang terjadi pada mesin diesel yang sedang diamati *user*. Halaman ini akan menunjukkan data SDS dari gejala apa yang sudah dimasukkan *user* di halaman sebelumnya.

Berikut adalah alur untuk melakukan pencarian kerusakan dengan fitur pemilihan gejala :

1. *User* Memilih Tipe Mesin
User dapat langsung menggunakan *search engine* tanpa perlu mendaftar. Dari beranda *website*, *user* bisa memilih tipe mesin (2 Tak atau 4 Tak) apa yang ingin dicari data SDS-nya.
2. Mencari dengan Memilih Gejala
Setelah memilih tipe mesin, *user* diberikan sejumlah pilihan gejala-gejala yang kemungkinan terjadi pada mesin diesel.

3. Percabangan Sistem

Dari gejala yang dipilih, *user* dapat lanjut dengan memilih bagian dari sistem mana yang kemungkinan terjadi kerusakan.

4. Memilih Data yang Cocok

Halaman pencarian dilanjutkan, akan muncul berbagai data yang sesuai dengan apa yang telah dipilih oleh *user*. Data SDS akan muncul setelah *user* memilih data yang kemungkinan berkaitan dengan kerusakan yang terjadi pada mesin diesel yang sedang diamati *user*. Halaman ini akan menunjukkan data SDS dari gejala apa yang sudah dimasukkan *user* di halaman sebelumnya.

3.3. Alur Memasukkan Data

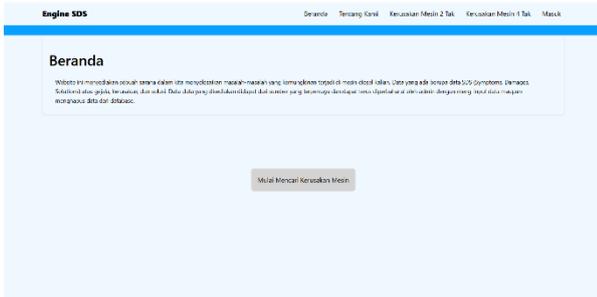
Alur memasukkan data ini hanya bisa diakses oleh admin yang sudah mendaftar ke sistem *website*. Alur memasukkan data ini memiliki fitur yang serupa dengan alur pencarian data, dibedakan oleh siapa yang bisa mengakses fitur ini yaitu admin. Berikut adalah alur untuk memasukkan data SDS ke dalam *database* :

1. Daftar Untuk Menjadi Admin
Tidak sembarang orang bisa diizinkan dalam proses memasukkan data SDS ke dalam *database* untuk menghindari hal yang tidak diinginkan. Hanya para admin yang sudah terdaftar dapat memasukkan data SDS ke dalam *database*.
2. Memilih Tipe Mesin
Admin memilih tipe mesin apa yang akan dimasukkan data SDS-nya.
3. Memasukkan Data SDS
Langkah selanjutnya adalah memasukkan data SDS yang sudah disiapkan ke dalam setiap *form* yang disediakan, lalu menekan tombol "*submit*". Data yang dimasukkan sudah masuk ke dalam *database* sesuai dengan tipe mesin apa yang dipilih.

3.4. Desain Website

Desain *website database* ini menjadi parameter dalam menentukan bagus atau tidaknya *website* yang kita kembangkan. Jika dari desain yang dipakai tidak mudah untuk digunakan oleh *user*, tentu akan menjadi nilai negatif dimata *user*. Penelitian ini menggunakan bahasa pemrograman *HTML*, *CSS*, dan *javascript*. Berikut detail desain dari setiap halaman dari *website* :

3.4.1. Fitur Beranda



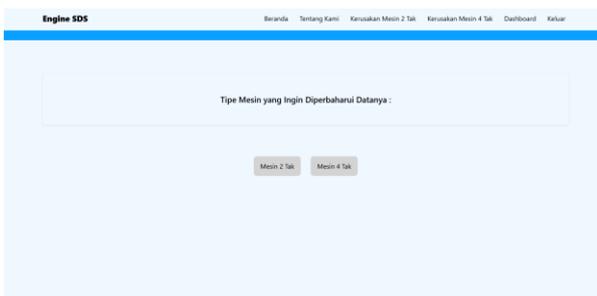
Gambar 2. Halaman Beranda Website

Gambar 2. menunjukkan halaman beranda website, berisikan penjelasan tentang fungsi website dan tombol untuk memulai mencari kerusakan yang terjadi. Gambar 2. juga menunjukkan pada bagian atas setiap halaman website terdapat baris navigasi untuk memudahkan user berpindah halaman dengan cepat dan ringkas.

3.4.2. Fitur Memilih Tipe Mesin



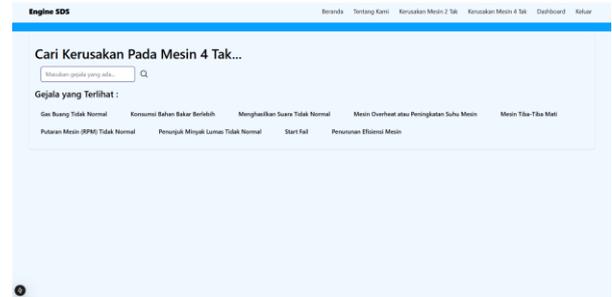
Gambar 3. Halaman Memilih Tipe Mesin (Mencari Kerusakan)



Gambar 4. Halaman Memilih Tipe Mesin (Memasukkan Data SDS)

Gambar 3 dan 4. menunjukkan bagaimana user nanti dapat memilih tipe mesin apa yang akan dicari kerusakannya (gambar 3.) atau memasukkan data dari kerusakan, gejala, dan solusi yang baru (gambar 4.).

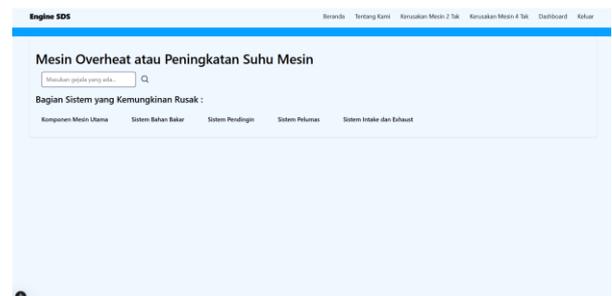
3.4.3. Fitur Pencarian Kerusakan



Gambar 5. Halaman Pencarian Kerusakan Mesin

Gambar 5. menunjukkan halaman di mana user dapat memulai pencarian kerusakan pada mesin. Bagian "Masukkan gejala yang ada..." diisi dengan gejala yang terlihat pada mesin diesel. Sistem melakukan pencarian dengan algoritma brute force pada fitur search engine. Di bawah kolom pencarian di halaman pencarian kerusakan mesin (gambar 5.). juga terdapat pilihan untuk fitur pemilihan gejala, yang terdapat berbagai gejala kerusakan yang kemungkinan terjadi.

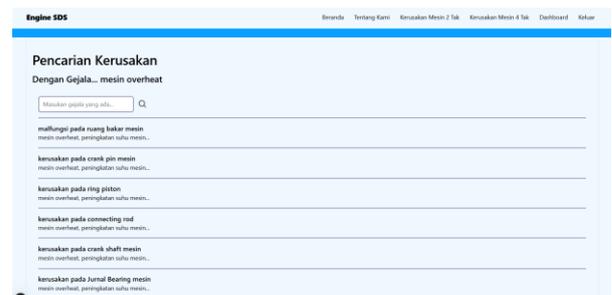
3.4.4. Fitur Cabang Pemilihan Gejala



Gambar 6. Halaman Pencarian Kerusakan Mesin

Gambar 6. Menunjukkan halaman untuk memilih percabangan sistem pembantu mesin yang kemungkinan terjadi kerusakan ketika menggunakan fitur pemilihan gejala, yang di mana pilihan tersebut akan memunculkan bagian-bagian dari sistem pembantu mesin.

3.4.5. Fitur Hasil Pencarian



Gambar 7. Halaman Hasil Pencarian

Gambar 7. memperlihatkan halaman hasil dari pencarian yang dimasukkan *user*, dengan kata kunci “mesin” mengeluarkan gejala-gejala kerusakan yang mengandung kata ”mesin” di dalamnya. Data-data yang keluar pada saat ini diambil dari *database* dengan fitur pemanggilan pada bagian *backend*.

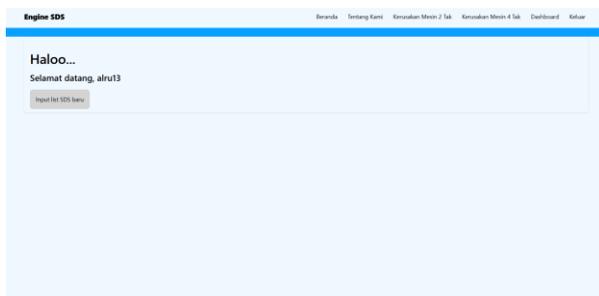
3.4.6. Fitur Detail Data SDS



Gambar 8. Halaman Detail Data SDS

Gambar 8. memperlihatkan halaman detail data SDS yang di dalamnya terdapat data, Gejala Kerusakan, Kemungkinan Letak Kerusakan dan Solusi yang diambil dari *database* dengan fitur pemanggilan pada bagian *backend*.

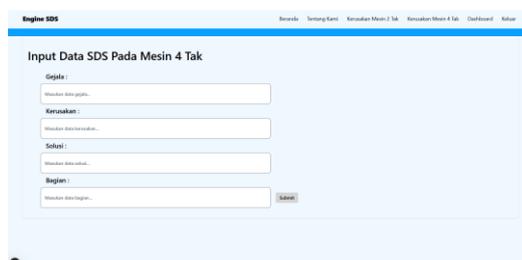
3.4.7. Fitur Dashboard Admin



Gambar 9. Halaman Dashboard Admin

Gambar 9. menunjukkan halaman *dashboard* admin, di mana admin dapat menambahkan data SDS mesin saat menekan tombol “Input List SDS Baru” maka halaman akan dialihkan ke halaman fitur memilih tipe mesin untuk memasukkan data (gambar 4.).

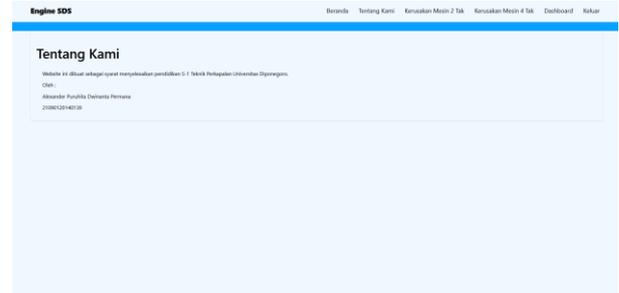
3.4.8. Fitur Pemasukan Data SDS Admin



Gambar 10. Halaman Memasukkan Data SDS Oleh Admin

Gambar 10. memperlihatkan halaman di mana admin dapat memasukkan data SDS ke dalam *database*. Admin memasukkan semua data SDS ke dalam *form* yang tersedia, Gejala, Kerusakan, Solusi, dan Referensi lalu menekan tombol “Submit” untuk memasukkan data yang sudah siap ke dalam *database*.

3.4.9. Fitur Halaman Tentang Kami



Gambar 11. Halaman Tentang Kami

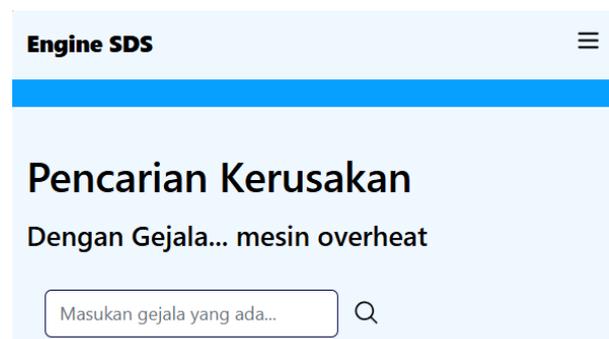
Gambar 11. memperlihatkan halaman yang berisikan alasan *website* ini dibuat dan juga perkenalan singkat dengan pengembang *website*.

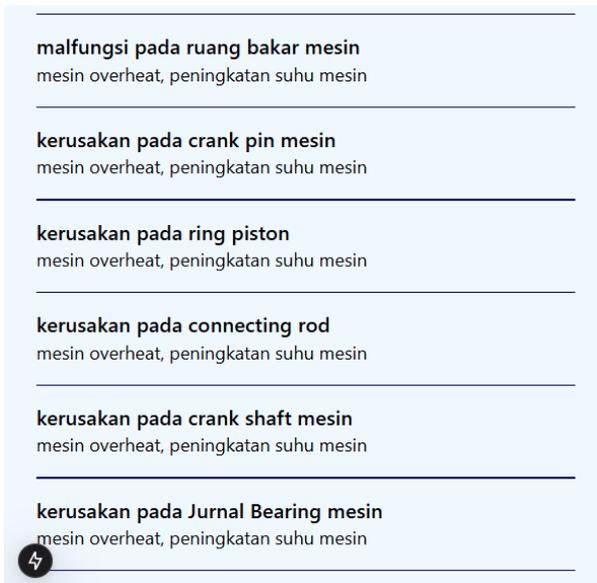
3.5. Hasil Pengujian Pencarian Kerusakan

Pengujian dalam penelitian ini dilakukan dengan dua cara, yang pertama dengan langsung memasukkan gejala apa yang terlihat pada mesin diesel, atau cara kedua dengan mencari secara bertahap dari memilih gejala apa yang terlihat, lalu memilih sistem yang kemungkinan terjadi kerusakan, terakhir terlihat kira-kira bagian dari sistem tersebut mana yang kemungkinan terjadi kerusakan dengan gejala yang dipilih tersebut. Berikut hasil dari kedua cara pencarian kerusakan :

3.5.1. Hasil Pengujian Fitur Search Engine

Pengujian dilakukan dengan menyimulasikan proses pencarian kerusakan yang akan dilakukan oleh *user*. *User* memasukkan data gejala kerusakan dengan *keyword* “mesin *overheat*”.





Gambar 12. Hasil Uji Coba Algoritma *Brute Force*

Gambar 12. menunjukkan proses pencarian kerusakan dengan memunculkan data kerusakan dan gejala mesin sesuai dengan *keyword* “mesin *overheat*” berdasarkan gejala yang dialami mesin diesel. *List* yang dimunculkan oleh *search engine* (gambar 12.) menunjukkan data kerusakan dengan *font* yang ditulis *bold* dan data gejala yang terletak dibawahnya. Proses selanjutnya yang perlu dilakukan adalah memastikan masalah apa yang sebenarnya terjadi dengan parameter kerusakan serta solusi yang sudah disediakan oleh *website*.

3.5.2. Hasil Pengujian Fitur Pemilihan Gejala

Pengujian dilakukan dengan menyimulasikan proses pencarian kerusakan yang akan dilakukan oleh *user*. Untuk pengujian sistem pakar ini dimulai dengan *user* memilih pilihan yang terdapat gambar 5. di halaman pencarian kerusakan.



Gambar 13. Halaman Gejala “Mesin *Overheat*”

Gambar 13. menjabarkan halaman gejala kerusakan yang terlihat berupa “Mesin *Overheat*, Peningkatan Suhu Mesin”, dari gejala tersebut memunculkan kemungkinan kerusakan terjadi pada sistem-sistem pembantu mesin.



Gambar 14. Halaman Hasil Uji Coba Sistem Pakar

Gambar 14. menunjukkan hasil dari pencarian setelah memilih sistem apa yang kemungkinan mengalami kerusakan, dan memunculkan bagian-bagian dari sistem tersebut yang memiliki kemungkinan mengalami kerusakan. Gambar 14. *user* memilih kerusakan yang terjadi pada “Komponen Mesin Utama”, jadi *website* mengeluarkan hasil bagian yang kemungkinan terjadi pada bagian di komponen mesin utama. Proses selanjutnya yang perlu dilakukan adalah memastikan masalah apa yang sebenarnya terjadi dengan parameter kerusakan serta solusi yang sudah disediakan oleh *website*.

3.5.3. Performa *Website*

Dari kedua hasil pengujian, *website* sudah berhasil dalam memunculkan data kemungkinan kerusakan yang terjadi dari data gejala yang di-*input* oleh *user*.



Gambar 15. Fitur Pencarian *Web Portal* Pencarian Berita [6]

Seperti pada penelitian tentang pencarian berita pada *web portal* menggunakan algoritma *brute force string matching*, hasil dari penelitian ini menunjukkan algoritma *brute force* dapat mencocokkan *string* namun *web portal* pencarian berita memiliki kekurangan di bagian aksesibilitas pengguna, di mana pengguna tidak mendapat rekomendasi pencarian dari teks yang di-*input* seperti pada gambar 15. [6].



Gambar 16. Fitur Pencarian Kerusakan

Pada penelitian ini, *user* diberikan dua opsi dalam mencari kerusakan mesin diesel. Gambar 16. menunjukkan terdapat fitur pilihan gejala yang terlihat pada mesin diesel yang memudahkan *user* dalam melakukan pencarian kerusakan pada mesin

diesel. Memberikan fitur pilihan gejala ini juga mengurangi kemungkinan kesalahan pengetikan dalam proses pendeteksian kerusakan.

Hasil dari mesin penelusuran penelitian ini dalam mencocokkan *string* sudah berhasil dan dapat memunculkan data yang sesuai dengan *string* yang di-*input* oleh *user*. Untuk kelebihan dari algoritma *brute force* sendiri terletak pada kesederhanaannya dalam proses pencocokan *string*, yang membuatnya dapat melakukan pencocokan dengan cepat dan akurat. Namun, kelemahannya terletak pada bila data yang di-*input* terdapat kekeliruan kemungkinan terjadi *error* sangat besar, dan algoritma *brute force* kurang cocok untuk menangani data berskala, karena proses pencocokannya dilakukan satu persatu.

4. KESIMPULAN

Berdasarkan hasil dari penelitian penyusunan *database* berbasis *website* untuk mendeteksi kerusakan dan langkah perbaikannya pada mesin diesel dapat diambil kesimpulan bahwa perancangan sistem informasi berbasis *website* dapat memudahkan *user* untuk mencari kerusakan yang kemungkinan terjadi pada mesin diesel yang sedang *user* amati. *Website* ini juga bisa diakses oleh teknisi dan mekanik serta admin yang terdaftar. Melalui penerapan metode algoritma *brute force* ini perancangan *website* berhasil menampilkan data SDS yang sekiranya diinginkan saat proses pencarian kerusakan mesin diesel.

Pada beberapa jenis kerusakan pada mesin diesel, terdapat beberapa hal yang sama yang diulang pada tiap-tiap jenis kerusakan yang kemungkinan terjadi.

Hasil dari penelitian ini berhasil mengembangkan *database* berisikan data dengan jumlah kerusakan sebanyak 147 (seratus empat puluh tujuh) jenis kerusakan pada mesin diesel. Keakuratan algoritma *brute force* dalam mencari *keyword* atau kata kunci dari data gejala yang dimasukkan menjadikan *search engine* ini alat yang berguna bagi mekanik dan teknisi dalam melakukan perawatan dan perbaikan pada mesin diesel.

UCAPAN TERIMA KASIH

Penulis ingin berterima kasih banyak kepada dosen pembimbing I dan dosen pembimbing II yang telah membimbing, memberi arahan, dan masukan selama penyusunan jurnal ini. Galangan PT. Yasa Wahana Tirta Samudera dan bengkel-bengkel mesin diesel di daerah Semarang yang sudah bersedia memberikan data untuk pengerjaan penelitian ini.

DAFTAR PUSTAKA

- [1] A. Maternová, M. Materna, A. Dávid, A. Török, and L. Švábová, "Human Error Analysis and Fatality Prediction in Maritime Accidents," *J Mar Sci Eng*, vol. 11, no. 12, Dec. 2023, doi: 10.3390/jmse11122287.
- [2] G. Vizentin, G. Vukelic, L. Murawski, N. Recho, and J. Orovic, "Marine propulsion system failures—a review," Sep. 01, 2020, *MDPI AG*. doi: 10.3390/jmse8090662.
- [3] D. Fitriati and I. Gibran, "Sistem Pakar Diagnosis Penyakit Meningitis Menggunakan Metode Forward Chaining," *Jurnal Sistem Informasi, Teknologi Informasi dan Komputer*, vol. 12, no. 1, pp. 46–50, 2021, [Online]. Available: <https://jurnal.umj.ac.id/index.php/just-it/index>
- [4] U. Budiarto, "Aplikasi Decision Support System Untuk Troubleshooting Pada Operasional Genset Di Kapal," *KAPAL*, vol. 5, no. 3, Oct. 2008.
- [5] Ajeng Tri Rahayu, Mukhtar Hanafi, and Maimunah, "Implementasi Algoritma Brute Force Pada Sistem Pertanahan di Balai Desa," *Journal of Computer System and Informatics (JoSYC)*, vol. 4, no. 3, pp. 710–721, May 2023, doi: 10.47065/josyc.v4i3.3520.
- [6] G. H. Fandi Nainggolan, S. Andryana, and dan Aris Gunaryati, "Pencarian Berita Pada Web Portal Menggunakan Algoritma Brute Force String Matching," *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 06, no. 1, pp. 1–10, Jun. 2021.
- [7] Dimas Dandy Aryarajendra Suprpto, Fauziah, Iskandar Fitri, and Nur Hayati, "Pengembangan Aplikasi Sistem Informasi Smart Register Online Berbasis Android Menggunakan Algoritma BruteForce," *Edumatic: Jurnal Pendidikan Informatika*,

vol. 4, no. 1, pp. 47–56, May 2020, doi:
10.47065/josyc.v4i3.3520.

- [8] Munir Rinaldi, *Strategi Algoritma*. Bandung: Institut Teknologi Bandung, 2009.
- [9] R. Sedgewick and K. Wayne, *Algorithms, 4th Edition*: Boston: Addison-Wesley, 2011. [Online]. Available: <https://books.google.co.id/books?id=GhsNBQAAQBAJ>
- [10] C. Charras and T. Lecroq, *Handbook of Exact String Matching Algorithms*. London: King's College Publications, 2004.