

Sistem Pakar untuk Mengidentifikasi Kerusakan Mesin Industri Menggunakan Metode *Certainty Factor*

Dedi Suryadi^{a,*}, Rifki Meilianda^a, Ahmad Fauzan Suryono^a, Munadi^b

^aProgram Studi Teknik Mesin, Fakultas Teknik, Universitas Bengkulu
Kampus Kandang Limun Jl. WR. Supratman Bengkulu

^bDepartemen Teknik Mesin, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Sudharto, SH, Tembalang, Semarang 50275

*E-mail: dedi_suryadi@unib.ac.id

Abstrak

Sistem perawatan mesin merupakan suatu hal yang sangat penting di industri untuk memperpanjang umur mesin. Salah satu metode perawatan yang sering digunakan adalah *predictive maintenance* berdasarkan sinyal getaran. Metode ini dapat memprediksi kerusakan mesin berdasarkan sinyal getaran yang timbul, dengan demikian kerusakan parah dapat dihindari. Tetapi metode ini sangat tergantung kepada pakar getaran untuk membaca karakteristik getarannya, sehingga tidak memungkinkan untuk operator biasa tidak bisa berbuat apa-apa apabila para pakar tidak ada. Oleh karena itu, dalam penelitian ini akan merancang sebuah aplikasi sistem pakar berbasis *android* yang dapat mengidentifikasi kerusakan pada mesin industri. Jenis kerusakan yang dapat diidentifikasi dalam aplikasi ini adalah *unbalance*, *misalignment*, kebengkokan poros, kelonggaran, dan kerusakan pada bantalan. Diharapkan aplikasi ini lebih praktis dan dapat digunakan oleh semua operator lapangan untuk mengidentifikasi kerusakan mesin. Metode yang digunakan dalam perancangan aplikasi ini adalah metode *certainty factor*. Untuk mengukur tingkat keakuratan hasil rancangan, aplikasi yang sudah diinstall di *android* dan divalidasi dengan hasil analisis pakar. Proses validasi data aplikasi meliputi pengujian sistem yang dilakukan dengan membandingkan hasil analisa pakar terhadap hasil analisa aplikasi sistem pakar. Pada penelitian ini, data untuk validasi diambil dari data hasil pengujian getaran pada mesin-mesin di industri. Sinyal getaran inilah yang dianalisis jenis kerusakannya oleh pakar dan dirunning oleh aplikasi. Hasil validasi menunjukkan tingkat akurasi aplikasi mencapai 100%.

Kata kunci: *certainty factor*, getaran, *predictive maintenance*, sistem pakar

1. Pendahuluan

Mesin merupakan komponen yang penting dalam proses produksi di dunia industri. Mesin-mesin di industri umumnya bergerak secara rotasi. Mesin-mesin rotasi tersebut rentan mengalami kerusakan. Kerusakan yang terjadi dapat diakibatkan oleh batas usia pemakaian suku cadang. Sebenarnya kerusakan dapat dihindari dengan proses penanganan dan pendeteksian dini kerusakan pada mesin yang dikenal dengan proses *predictive maintenance*. Dalam kegiatan *predictive maintenance* proses analisa kerusakan dapat dilakukan dengan melihat kecenderungan vibrasi yang dialami oleh mesin. Kerusakan yang terjadi umumnya dapat diidentifikasi oleh ahli di bidang vibrasi dengan melihat ciri-ciri kerusakan yang terjadi berdasarkan sinyal getaran. Namun, kelemahan menggunakan jasa ahli dalam menganalisa kerusakan adalah memerlukan waktu yang cukup lama dalam proses analisa. Dengan menggunakan jasa ahli atau analisa secara konvensional, hal ini tentu saja menjadi sebuah kerugian atau kelemahan tersendiri dan berdampak buruk bagi pengeluaran perusahaan terutama waktu.

Untuk menanggulangi permasalahan tersebut, pengembangan teknologi sistem pakar merupakan salah satu cara yang efektif. Sistem pakar adalah suatu program komputer cerdas yang menggunakan *knowledge* (pengetahuan) dan prosedur inferensi untuk menyelesaikan masalah yang cukup sulit sehingga membutuhkan seorang yang ahli untuk menyelesaikannya [1]. Sistem pakar secara masif terus dikembangkan untuk dapat mempermudah kerja manusia dalam segala bidang. Beberapa contoh aplikasi dari sistem pakar adalah dengan dikembangkannya sistem pakar untuk mendeteksi kerusakan sepeda motor [2]. Sistem pakar deteksi kerusakan motor dibuat dengan metode *forward chaining* dan berbasis *android* sehingga kerusakan sepeda motor dapat terdeteksi dengan menggunakan aplikasi *android*. Contoh lain pengembangan sistem pakar adalah untuk mendiagnosa gejala kerusakan mesin mobil toyota menggunakan metode *case based reasoning* [3]. Sistem pakar diagnosa kerusakan mobil Toyota dirancang dengan menggunakan metode berbasis kasus dan berbasis *website* sehingga dapat mendiagnosa kerusakan mobil Toyota. Kedua penelitian tersebut membuktikan bahwa sistem pakar dirancang agar mempermudah kerja manusia tak terkecuali untuk mengidentifikasi kerusakan mesin industri.

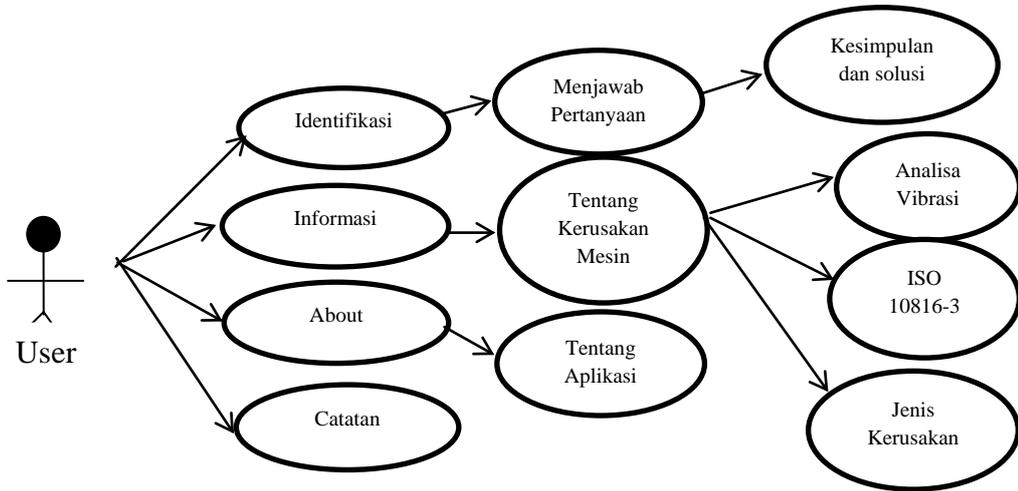
Maka dalam penelitian ini akan dirancang sistem pakar berbasis *android* dengan menggunakan metode *certainty factor*. Metode *certainty factor* menyatakan kepercayaan dalam sebuah kejadian atau fakta berdasarkan bukti atau penilaian pakar. *Certainty factor* menggunakan suatu nilai untuk mengasumsi derajat keyakinan seorang pakar terhadap suatu data [4]. Sehingga, berdasarkan fakta-fakta gejala kerusakan yang ada dimasukkan ke dalam sistem pakar maka

akan didapat kemungkinan dari jenis kerusakan yang terjadi. Dengan adanya sistem pakar untuk identifikasi kerusakan mesin berbasis *android* dinilai dapat menjadi salah satu solusi bagi perusahaan dan teknisi pemula dalam membantu perannya untuk melakukan proses analisa kerusakan lebih efektif dan praktis sehingga tindakan perbaikan dapat segera dilakukan.

2. Metodologi

2.1 Rancangan Sistem

Dalam penelitian ini, sistem dirancang berdasarkan *use case diagram*. *Use case diagram* digunakan untuk memodelkan proses pada aplikasi berdasarkan perspektif pengguna sistem. Diagram ini terdiri atas diagram untuk *use case* dan *actor*. Gambar 1 menunjukkan gambaran *use case diagram* perancangan sistem.



Gambar 1. Use case diagram.

2.2 Data Gejala Kerusakan

Setelah merancang sistem, langkah selanjutnya adalah mencari data gejala kerusakan mesin untuk beberapa kondisi kerusakan meliputi *unbalance*, *misalignment*, kebengkokan poros, kelonggaran, dan kerusakan pada bantalan. Pada proses ini akan menjelaskan data-data terkait standar vibrasi ISO 10816-3 dengan jenis-jenis kerusakan dan gejala kerusakan yang dialami mesin. Data inilah yang nantinya akan dibuat program dan didahului dengan adanya relasi data antara gejala dan kerusakan yang yang terjadi. Selanjutnya, tentu kita butuh standar sebagai acuan. ISO 10816-3 dijadikan sebagai acuan internasional untuk batas getaran yang diizinkan. Dalam perancangan sistem pakar ini, data ISO 10816-3 akan dijadikan sebagai *database* untuk menentukan apakah level vibrasi dari suatu mesin masih dalam batas izin atau tidak. Gambar 2 menerangkan mengenai standar ISO 10816-3. Data ini akan digunakan dalam pemrograman sebagai *database* batas vibrasi yang diizinkan.

				mm/s	rms
				11	
				7.1	
				4.5	
				3.5	
				2.8	
				2.3	
				1.4	
				0.7	
rigid	flexible	rigid	flexible	Foundation	
pump > 15 kW radial, axial, mixed flow				Machine Type	
integrated driver		external driver		Group	
Group 4		Group 3			
medium sized machines 15 kW < P < 300 kW		large machines 300 kW < P < 50 MW		Machine Type	
motors 160 mm < H < 315 mm		motors 315 mm < H		Group	
Group 2		Group 1			

Gambar 2. ISO 10816-3 velocity [5].

Tahap selanjutnya adalah penentuan indikasi kerusakan pada mesin. Umumnya mesin akan memiliki ciri-ciri atau terdapat gejala tertentu apabila mengalami kerusakan sehingga dari gejala yang timbul dapat teridentifikasi kerusakan yang ada. Tabel 1 menyajikan indikasi atau gejala-gejala kerusakan yang umumnya terjadi pada mesin industri.

Tabel 1. Daftar indikasi kerusakan [6].

Kode	Indikasi Kerusakan
Q1	Getaran dominan di arah aksial
Q2	Getaran dominan di arah radial
Q3	Terdapat puncak di 2x rpm
Q4	Muncul sub harmonik pada 0.5, 1,5, 2.5x rpm
Q5	Muncul puncak 3x rpm
Q6	Spektrum dominan di 1x rpm saat muncul 3x rpm
Q7	Spektrum dominan di 2x rpm saat muncul 3x rpm
Q8	Terdapat puncak pada 1x rpm
Q9	Terdapat <i>sideband</i> pada domain waktu
Q10	Terjadi <i>random vibration</i> pada frekuensi tinggi
Q11	Panas berlebih pada area bantalan
Q12	Suara bising pada bantalan ($> 2gE$)
Q13	Terdapat beda fasa $\pm 180^\circ$
Q14	Beda fasa $\pm 90^\circ$

Setelah gejala dan jenis kerusakan telah didefinisikan maka langkah selanjutnya adalah membuat basis aturan. Adapun basis pengetahuan yang dibuat dapat dilihat pada Tabel 2.

Tabel 2. Basis aturan

No.	Kerusakan	Aturan
1	<i>Unbalance</i>	IF Q2 AND Q8
2	<i>Misalignment</i>	IF Q1 AND Q2 AND Q3 AND Q6 AND Q7 AND Q13
3	<i>Mechanical Looseness</i>	IF Q2 AND Q4 AND Q9
4	<i>Bent Shaft</i>	IF Q1 AND Q3
5	<i>Bearing Defect</i>	IF Q10 AND Q11 AND Q12

2.3 Menentukan Nilai *Certainty Factor* (CF)

Dalam tahapan ini, penentuan nilai *certainty factor* berdasarkan informasi hasil wawancara dengan pakar. Nilai CF didapat dari *qualitatif term* yang diberikan oleh pakar sehingga menjadi nilai CF tertentu. Adapun data penentuan nilai *certainty factor* untuk masing-masing gejala dapat dilihat pada Tabel 3.

Tabel 3. Nilai *certainty factor* setiap gejala

No.	Gejala Kerusakan	Nilai CF
1	Getaran dominan di arah aksial	0.2
2	Getaran dominan di arah radial	0.2
3	Terdapat puncak di 2x rpm	0.8
4	Muncul sub harmonik pada 0.5, 1,5, 2.5x rpm	0.8
5	Muncul puncak 3x rpm	0.8
6	Spektrum dominan di 1x rpm saat muncul 3x rpm	0.8
7	Spektrum dominan di 2x rpm saat muncul 3x rpm	0.8
8	Terdapat puncak pada 1x rpm	0.8
9	Terdapat <i>sideband</i> pada domain waktu	0.6
10	Terjadi <i>random vibration</i> pada frekuensi tinggi	0.4
11	Panas berlebih pada area bantalan	0.2
12	Suara bising pada bantalan ($> 2gE$)	0.6
13	Terdapat beda fasa $\pm 180^\circ$	0.6
14	Beda fasa $\pm 90^\circ$	0.6

2.4 Implementasi dan Pengujian Sistem

Dalam tahap ini akan dilakukan pengujian dan validasi hasil analisa aplikasi dengan hasil analisa ahli. Hal ini dilakukan untuk menentukan tingkat akurasi aplikasi yang telah dirancang. Selain itu, validasi juga dilakukan pada metode yang digunakan dalam perancangan aplikasi ini. Proses validasi metode *certainty factor* adalah dengan membandingkan hasil nilai CF aplikasi dengan nilai CF perhitungan manual. Validasi ini bertujuan untuk menguji dan

memastikan bahwa algoritma yang disusun pada program telah sesuai dan benar. Adapun langkah-langkah perhitungan metode *certainty factor* yaitu [7]:

1. Menentukan nilai CF Pakar {CF(H)} dan CF pengguna {CF(E)} yang didapat dari hasil wawancara pakar pada Tabel 2.
2. Perhitungan nilai CF dilakukan dengan mengalikan nilai CF Pakar dengan nilai CF pengguna.
3. Mengkombinasikan setiap 2 nilai CF sesuai dengan aturan yang ada. Dan apabila terdapat lebih dari 2 nilai CF, maka dilakukan pengkombinasian nilai CF tersebut.
4. Mengalikan hasil nilai CF kombinasi dengan 100% untuk mendapat jawaban dalam bentuk persentase.

3. Hasil dan Pembahasan

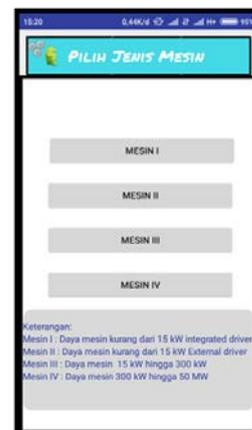
Setelah tahap perancangan dan proses pembuatan koding maka perlu dilakukan implementasi sistem. Dalam sistem pakar perlu dilakukan proses implementasi sistem untuk mengetahui apakah sistem berjalan dengan baik. Adapun hasil dari perancangan dari sistem pakar identifikasi kerusakan adalah sebagai berikut:

3.1 Hasil Tampilan *Layout* Layar Utama

Pada tampilan *layout* menu utama terdiri dari 4 pilihan menu, yaitu menu MULAI IDENTIFIKASI untuk memulai tahap proses analisa vibrasi, menu INFORMASI untuk menampilkan informasi mengenai ISO 10816-3, analisa vibrasi dan jenis kerusakan berdasarkan karakteristik getaran, menu ABOUT untuk menampilkan tentang aplikasi, menu CATATAN untuk membuat catatan sebelum dan sesudah proses analisa. Tampilan *layout* menu utama dapat dilihat pada Gambar 3.



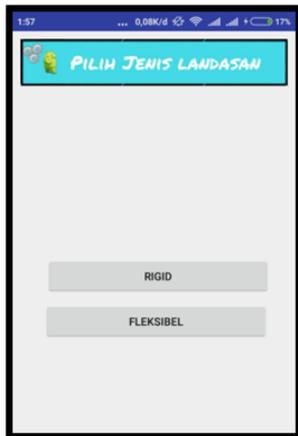
Gambar 3. Tampilan *layout* halaman utama.



Gambar 4. Tampilan menu pilih jenis mesin.

Dalam menu identifikasi terdapat beberapa proses atau tahapan, diantaranya pemilihan jenis mesin, pemilihan jenis landasan, input nilai vibrasi dan pilih menu gejala. Proses pertama dalam tahapan menu identifikasi yang akan dilakukan adalah memilih jenis mesin yang akan dianalisa kerusakannya. Dalam menu ini terdapat 4 pilihan jenis mesin, di mana mesin jenis pertama adalah mesin dengan daya kurang dari 15 kW *integrated driver* (penggerak terhubung langsung dengan mesin). Sedangkan untuk pilihan mesin 2 merupakan mesin dengan daya kurang dari 15 kW *external driver* (penggerak tidak terhubung langsung ke mesin). Pilihan mesin 3 ialah mesin dengan kapasitas daya 15 kW sampai dengan 300 kW. Untuk pilihan mesin 4 adalah mesin dengan kapasitas daya 300 kW sampai dengan 50 MW. Tampilan menu pilih jenis mesin dapat dilihat pada Gambar 4.

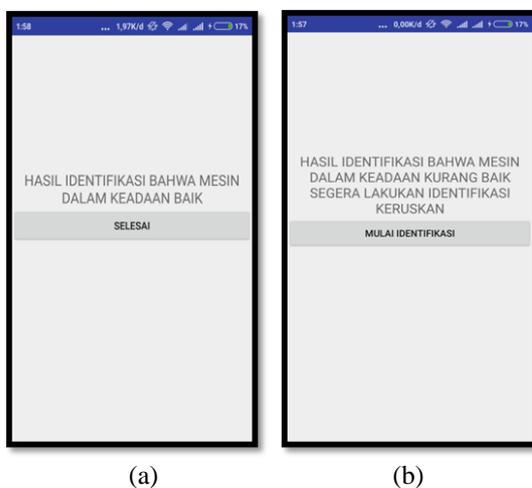
Tahapan kedua yang harus dilakukan adalah memilih jenis landasan mesin. Dalam tahap ini terdapat dua pilihan jenis landasan mesin, yaitu *rigid* atau kaku dan landasan fleksibel. Gambar 5 menampilkan tampilan *layout* menu pilih jenis landasan. Proses selanjutnya ialah menginputkan nilai vibrasi dari mesin yang akan dianalisa. Pada tahapan ini, nilai vibrasi yang diinputkan berada dalam satuan *velocity* (mm/s). Gambar 6 menunjukkan tampilan *layout* input nilai vibrasi. Selanjutnya, setelah nilai vibrasi diinputkan maka akan keluar hasil apakah nilai vibrasi tersebut berada dalam batas yang diizinkan sesuai dengan tipe dan landasan mesin berdasarkan ISO 10816-3. Jika nilai vibrasi berada di bawah level vibrasi yang diizinkan maka tahapan proses analisa dianggap selesai. Dapat dilihat pada Gambar 7 (a). Apabila nilai vibrasi yang telah diinputkan berada di atas level vibrasi yang diizinkan maka akan dilanjutkan ke tahap proses identifikasi kerusakan berdasarkan gejala-gejala yang ada. Gambar 7 (b) menunjukkan *layout* jika level vibrasi di atas batas yang diizinkan. Berdasarkan nilai vibrasi yang telah diinputkan sebelumnya, jika nilai vibrasi di atas level yang diizinkan maka tahapan selanjutnya adalah memilih gejala kerusakan yang ada. Menu tampilan pilih gejala kerusakan dapat dilihat pada Gambar 8.



Gambar 5. Tampilan menu pilih jenis landasan.

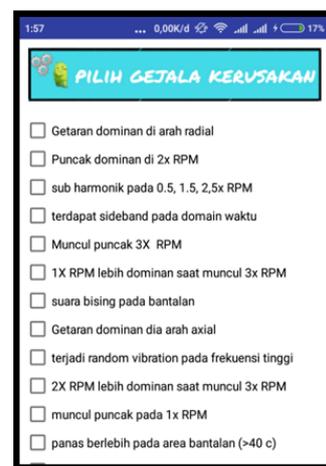


Gambar 6. Tampilan *layout* input nilai vibrasi.



(a) (b)

Gambar 7. (a) Tampilan *layout* nilai vibrasi aman dan (b) Tampilan *layout* nilai vibrasi tidak aman



Gambar 8. Tampilan *layout* pilih gejala kerusakan

3.2 Hasil Pengujian Sistem

Aplikasi yang sudah dibuat diuji coba dengan menggunakan data pengujian getaran mesin di industri. Di sini, proses analisa membutuhkan spesifikasi mesin yang digunakan. Adapun data spesifikasi mesin dan hasil pengukuran *overall vibration* yang akan dianalisa dapat dilihat pada Tabel 4.

Tabel 4. Nilai *overall vibration*

No.	Parameter	Satuan	Nilai
1	Daya motor	Kw	15
2	Vibration		
	Vertikal	mm/s	6,8
	Horizontal	mm/s	4
	Noisy	gE	0,28
3	Temperatur	°C	36

Respons getaran dalam domain frekuensi disajikan dalam Gambar 9. Berdasarkan nilai *Overall Vibration* dan Gambar 9, pakar menyimpulkan bahwa mesin mengalami kerusakan jenis *Unbalance*. Dari data hasil analisa tersebut diketahui bahwa kerusakan yang terjadi adalah *unbalance* dan untuk menguji sistem maka perlu dilakukan analisa menggunakan aplikasi. Dari data yang tersedia, proses pertama yang dilakukan dalam tahapan analisa adalah memilih jenis mesin, kemudian memilih jenis landasan mesin dan menginputkan nilai vibrasi. Dalam kasus ini, jenis mesin yang dipilih adalah jenis mesin 2 dengan rincian mesin dengan daya 15 kW *external driver* kemudian dengan jenis landasan *rigid*. Nilai vibrasi yang diinputkan adalah 6.8 mm/s. Dari hasil identifikasi nilai vibrasi didapatkan bahwa kondisi vibrasi mesin di atas batas yang diizinkan maka perlu dilakukan analisa lanjutan. Pada tahapan pilih gejala, gejala yang dipilih berupa adanya vibrasi dominan di arah radial dengan 1x RPM. Ciri tersebut berdasarkan pengamatan secara langsung berdasarkan data yang tersedia.

kerusakan pada bantalan dengan nilai *certainty factor* 76. Sedangkan pada kasus 4 didapatkan hasil analisa *expert* kerusakan berupa *looseness bearing* dan untuk hasil analisa aplikasi menunjukkan adanya kemungkinan *mechanical looseness* dengan nilai CF 84, *unbalance, misalignment* dengan nilai CF 20 dan *bearing defect* dengan nilai CF 60. Berdasarkan hasil analisa aplikasi tersebut, menunjukkan adanya beberapa gejala lain yang memungkinkan. Hanya saja dalam kasus 4 ini jenis kerusakan *mechanical looseness* kemungkinan besar terjadi karena nilai CF sebesar 84 atau kemungkinan kerusakan 84%.

Jenis kerusakan *mechanical looseness* dan *bearing defect* teridentifikasi berdasarkan hasil analisa aplikasi dengan masing-masing nilai *certainty factor* 80 untuk *mechanical looseness* dan 76 untuk *bearing defect*. Sedangkan hasil analisa *expert* menunjukkan adanya kerusakan *looseness bearing*. Sehingga antara hasil *expert* dan aplikasi sistem pakar tidak berbeda. Hanya saja pada aplikasi mengidentifikasi adanya kerusakan pada bagian *bearing*. Dari ke 5 data yang telah di uji, pada aplikasi menunjukkan kesamaan hasil analisa dengan analisa *expert* (dapat dilihat pada Tabel 5). Akan tetapi, pada hasil analisa aplikasi menunjukkan adanya beberapa kerusakan-kerusakan lain yang teridentifikasi. Hal ini tentu saja membutuhkan keputusan dari teknisi yang melakukan analisa. Sehingga proses perbaikan yang tepat dapat dilakukan.

4. Kesimpulan

Dari hasil perancangan sistem pakar identifikasi kerusakan mesin berbasis *android* dapat ditarik kesimpulan bahwa aplikasi sistem pakar yang dirancang mampu mengidentifikasi kerusakan mesin berdasarkan karakteristik getaran yang menyatakan akurasi aplikasi mencapai 100%. Sehingga aplikasi sistem pakar ini dapat digunakan sebagai alat bantu para teknisi dalam proses analisa vibrasi.

Referensi

1. Kusriani, S., 2008, "*Sistem Pakar Teori dan Aplikasi*," Andi Publisher, Yogyakarta.
2. Putu, I.P.W., 2015, "*Sistem Pakar untuk Mendeteksi Kerusakan Sepeda Motor Berbasis Android*," Konferensi Nasional Sistem dan Informatika, STMIK STIKOM Bali.
3. Abdullah, D., Azmi, K., 2016, "*Sistem Pakar Mendiagnosa Gejala Kerusakan Mesin Mobil Toyota Menggunakan Metode Case Based Reasoning*," *Reserachgate*.
4. Durkin, J., 1994, "*Expert Systems Design and Development*," Prentice Hall.
5. Technical Associates of Charlotte, 2001, "*Vibration Analysis Module*," North Carolina, USA.
6. Girdhar, P., Scheffer, C., 2004, "*Machinery Vibration Analysis & Predictive Maintenance*," Netherlands, IDC Technologies.
7. Hasibuan, N.A., Sunandar, H., Alas, S., Suginam, 2017, "*Sistem Pakar Mendiagnosa Penyakit Kaki Gajah Menggunakan Metode Certainty Factor*," *Jurnal Riset Sistem Informasi dan Teknik Informatika*, 2(1): 29-39.