

# Rancang Bangun Sistem Pendukung Keputusan Berbasis Web untuk Diagnosa Penyakit

Muhammad Yogie Nugroho, Mohammad Nuh, dan Muhammad Yazid  
 Departemen Teknik Biomedik, Institut Teknologi Sepuluh Nopember (ITS)  
*e-mail:* nuh@ee.its.ac.id.

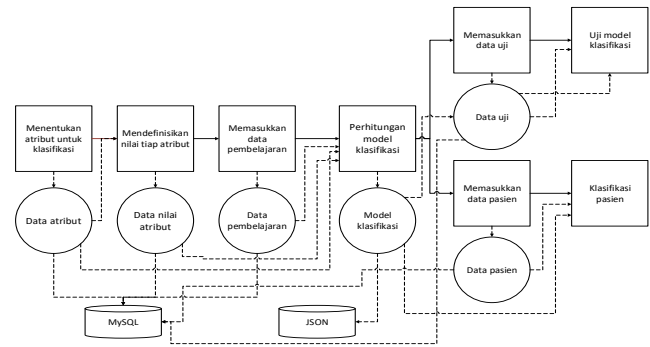
**Abstrak**—Pada beberapa kasus, diagnosa penyakit tidak dapat dilakukan dengan mudah karena beberapa tanda dan gejala mungkin saling beririsan dengan penyakit lain. Sistem pendukung keputusan merupakan salah satu pendekatan yang dapat digunakan untuk membantu dokter dalam pengambilan keputusan diagnosa. Pada penelitian ini dirancang suatu sistem pendukung keputusan yang dapat digunakan untuk mendiganosa suatu penyakit sesuai dengan kebutuhan. Sistem pendukung dirancang agar dapat terintegrasi dengan data pasien sehingga secara praktis dapat digunakan di rumah sakit. Sistem diimplementasikan ke dalam sistem berbasis web dengan bahasa pemrograman PHP dan *Relational Database Management System MySQL*. Jenis sistem pendukung keputusan yang dirancang pada penelitian merupakan *nonknowledge-based sytem* dengan algoritma *machine learning* yang digunakan yaitu *Naive Bayes*. Berdasarkan pengujian performansi, pada saat kondisi normal performansi sistem baik dan tidak terpengaruh jumlah *user* yang melakukan akses. Selain itu tidak terdapat perbedaan performa yang signifikan saat kondisi normal dan pada saat kondisi terdapat lonjakan trafik.

**Kata Kunci**—Sistem Pendukung Keputusan, Diagnosa, *Machine Learning*, *Naive Bayes*

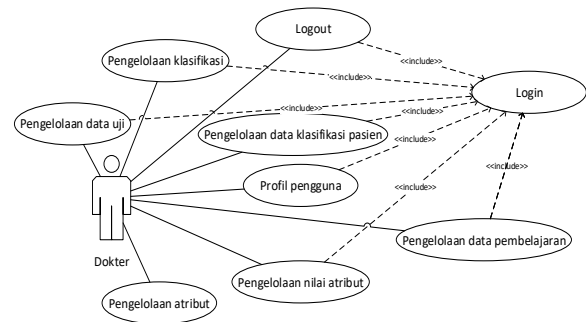
## I. PENDAHULUAN

**K**ESEHATAN merupakan hal yang penting untuk dijaga. Beberapa penyakit dapat menyerang manusia dan menyebabkan terganggunya fungsionalitas tubuh yang berujung pada meningkatkannya resiko kematian. Diagnosa penyakit secara dini disertai penanganan medis yang tepat diharapkan dapat menurunkan resiko kematian. Terlebih lagi dari sisi biaya, pembiayaan perawatan penyakit yang tidak murah dan tingkat keberhasilan pengobatan yang tergantung dari kecepatan penanganan penyakit menjadikan upaya untuk diagnosa pasien secara dini menjadi langkah yang paling tepat. Pada beberapa kasus, diagnosa penyakit tidak dapat dilakukan dengan mudah karena beberapa tanda dan gejala mungkin saling beririsan dengan penyakit lain. Oleh karena itu perlu adanya pendekatan lain untuk membantu mendiagnosa penyakit secara tepat.

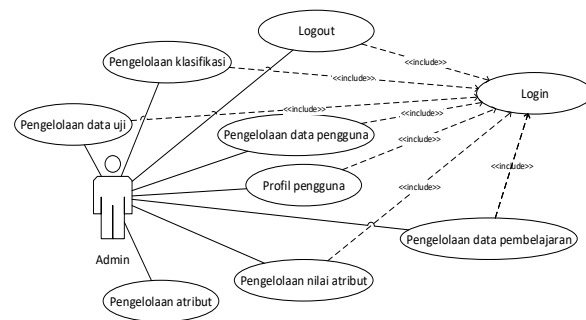
Sistem pendukung keputusan telah digunakan selama beberapa dekade dan berpotensi secara signifikan meningkatkan perawatan dan keselamatan pasien [1]. Dalam rangka meningkatkan perawatan dan keselamatan pasien tersebut, sistem pendukung keputusan telah dibuat untuk berbagai kebutuhan klinis, salah satunya digunakan untuk membantu dokter dalam pengambilan keputusan diagnosa. Pada beberapa penelitian terkait, sistem pendukung keputusan yang dirancang dapat digunakan untuk membantu diagnosa berbagai macam penyakit kronis seperti batu ginjal



Gambar 1. Gambaran Umum Sistem

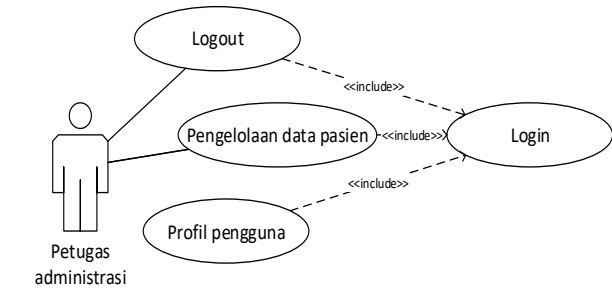


Gambar 2. Use Case Dokter

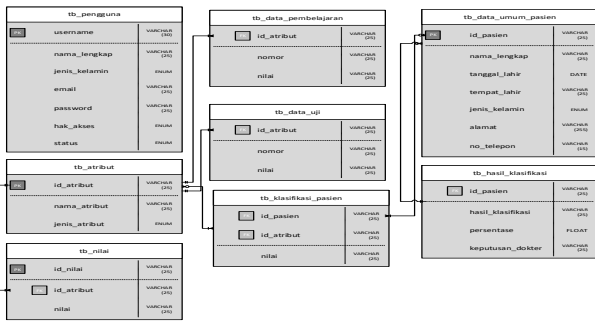


Gambar 3. Use Case Admin

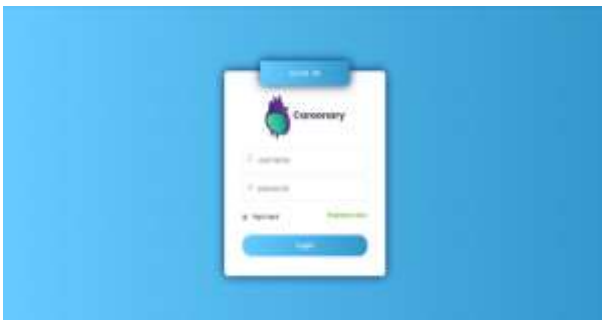
[2], hepatitis [3], serta jantung koroner [4]. Namun, umumnya sistem pendukung keputusan yang dirancang hanya dapat digunakan untuk membantu diagnosa suatu penyakit yang spesifik. Selain itu tidak terdapat mekanisme yang mengatur akses ke dalam sistem pendukung keputusan dan integrasi dengan data pasien. Pada penelitian ini, dirancang sistem pendukung keputusan berbasis web yang tidak hanya dapat digunakan untuk membantu diagnosa penyakit secara spesifik. Sistem pendukung keputusan yang dirancang nantinya dapat digunakan untuk membantu diagnosa berbagai penyakit, bergantung pada kebutuhan pengguna atau instansi di rumah sakit [5], [6]. Sistem



Gambar 4. Use Case Petugas Administrasi



Gambar 5. Rancangan database sistem pendukung keputusan



Gambar 6. Antarmuka menu login

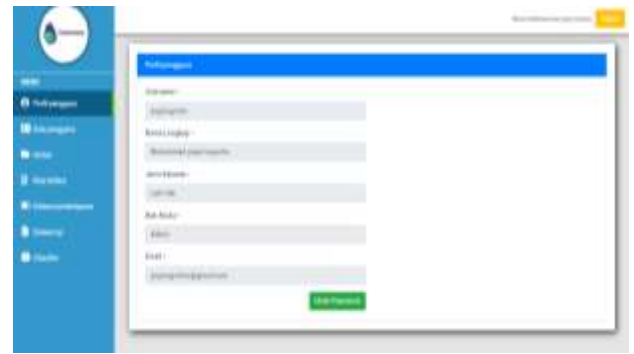
pendukung keputusan juga dirancang agar dapat terintegrasi dengan data pasien sehingga secara praktis dapat digunakan di rumah sakit.

## II. PERANCANGAN SISTEM

### A. Gambaran Umum Cara Kerja Sistem

Jenis sistem pendukung keputusan yang dirancang pada penelitian merupakan *nonknowledge-based system* dengan algoritma *machine learning* yang digunakan yaitu Naive Bayes. Rancangan *nonknowledge-based system* digunakan karena pengimplementasian rancangan *knowledge-based system* hanya bisa digunakan secara spesifik ke suatu penyakit. Selain itu, algoritma Naive Bayes dipilih untuk mengatasi kekurangan dari rancangan *nonknowledge-based system* yang umumnya tidak dapat memberikan informasi yang berguna bagi pengguna akibat dari penggunaan algoritma *machine learning* yang sulit diobservasi hubungan masukan dan keluarannya. Sistem pendukung keputusan pada tugas akhir ini selanjutnya diimplementasikan dalam bahasa pemrograman PHP dan RDBMS MySQL.

Secara umum sistem pendukung keputusan yang dibuat digambarkan oleh Gambar (1). Blok proses pertama yaitu menentukan atribut yang dibutuhkan untuk mendiagnosa penyakit. Blok proses selanjutnya yaitu menentukan nilai-nilai yang mungkin dari masing-masing atribut tersebut.



Gambar 7. Antarmuka menu profil pengguna



Gambar 8. Antarmuka menu data umum pasien



Gambar 9. Antarmuka menu data pengguna

Setelah pendefinisian atribut dan nilai atribut dilakukan, data pembelajaran perlu dimasukkan ke dalam sistem sebagai acuan proses *learning* oleh Naive Bayes. Hasil keluaran dari proses *learning* adalah model klasifikasi yang didapatkan dari perhitungan matematis algoritma Naive Bayes. Model klasifikasi yang didapatkan dari proses *learning* selanjutnya digunakan untuk mengklasifikasikan data pasien. Hasil dari proses *learning* dapat diuji tingkat akurasi berdasarkan suatu data uji yang dimasukkan ke dalam sistem. Proses *learning* dalam sistem diatur oleh pengguna yang memiliki otoritas. Model klasifikasi dapat diperbarui seiring dengan bertambahnya data pembelajaran. Data-data mengenai atribut, nilai atribut, data pasien, data pembelajaran, dan data uji disimpan dalam database MySQL sedangkan model klasifikasi disimpan dalam JSON oleh sistem.

### B. Perancangan Use Care Diagram

*Use case* diagram menggambarkan fungsionalitas dari keseluruhan sistem pendukung keputusan [5], [7], [8]. Pengguna sistem serta fungsi-fungsi yang dapat dikerjakan oleh masing-masing aktor atau pengguna digambarkan *use case diagram* Gambar 2 hingga Gambar 4. Pengguna pada sistem pendukung keputusan yang dirancang antara lain



Gambar 10. Antarmuka menu data atribut



Gambar 13. Antarmuka menu data uji



Gambar 11. Antarmuka menu data nilai atribut



Gambar 14. Antarmuka menu pengelolaan klasifikasi



Gambar 12. Antarmuka menu data pembelajaran



Gambar 15. Antarmuka data klasifikasi pasien

yaitu petugas administrasi, admin, dan juga dokter. Definisi dari setiap *use case* adalah sebagai berikut :

#### 1) *Use Case Registrasi*

*Use case* registrasi dilakukan oleh calon pengguna agar bisa masuk ke dalam sistem.

#### 2) *Use Case Login*

*Use case login* dilakukan setiap pengguna yang sudah terverifikasi untuk masuk ke dalam sistem.

#### 3) *Use Case Logout*

*Use case logout* dilakukan pengguna untuk keluar dari sistem pendukung keputusan.

#### 4) *Use Case Profil Pengguna*

*Use case* profil pengguna merupakan *use case* generalisasi yang memuat *use case* lihat data profil dan ubah data profil. *Use case* profil pengguna menampilkan informasi profil pengguna dan pengguna dapat mengubah data *profil*.

#### 5) *Use Case Pengelolaan Pengguna*

*Use case* pengelolaan pengguna dilakukan oleh admin sistem untuk mengatur proses login pengguna ke dalam sistem. Admin dapat melihat, menambah, mengubah, dan menghapus data pengguna.

#### 6) *Use Case Pengelolaan Data Umum Pasien*

*Use case* pengelolaan data umum pasien dilakukan oleh petugas administrasi untuk mendeskripsikan proses pengelolaan data umum pasien yaitu melihat, menambah, mengubah, atau menghapus data.

#### 7) *Use Case Pengelolaan Atribut*

*Use case* pengelolaan atribut dilakukan oleh admin atau dokter untuk mendeskripsikan atribut data yang diperlukan untuk klasifikasi. Admin atau dokter dapat melihat, menambah, mengubah, dan menghapus data atribut.

#### 8) *Use Case Pengelolaan Nilai Atribut*

*Use case* pengelolaan nilai atribut dilakukan oleh admin atau dokter untuk mendeskripsikan nilai dari masing-masing atribut data yang ada di *database* untuk klasifikasi. Admin dan dokter dapat melihat, menambah, mengubah, dan menghapus data nilai atribut.

#### 9) *Use Case Pengelolaan Data Pembelajaran*

*Use case* pengelolaan data pembelajaran dilakukan oleh admin atau dokter untuk mendeskripsikan data pembelajaran yang nantinya digunakan untuk membuat model klasifikasi. Admin dan dokter dapat melihat, menambah, mengubah, dan menghapus, mengimpor dan mengekspor data pembelajaran..

#### 10) *Use Case Pengelolaan Data Uji*

*Use case* pengelolaan data uji dilakukan oleh admin atau dokter untuk mendeskripsikan data uji yang nantinya digunakan untuk menguji model klasifikasi yang sudah dibuat. Pengguna dapat melihat, menambah, mengubah, menghapus, dan mengimpor data uji.

Tabel 1.  
Pengujian use case registrasi

Butir Uji	Respon	Kesimpulan
Registrasi (skenario utama)	Pesan bahwa registrasi berhasil ditampilkan	Sesuai
Registrasi (skenario alternatif)	Pesan error ditampilkan	Sesuai

Tabel 2.  
Pengujian use case login

Butir Uji	Respon	Kesimpulan
Login (skenario utama)	Pengguna masuk menu beranda sesuai hak akses	Sesuai
Login (skenario alternatif)	Pesan error ditampilkan	Sesuai

Tabel 3.  
Pengujian use case logout

Butir Uji	Respon	Kesimpulan
Logout	Pengguna berhasil logout dan diarahkan ke menu awal	Sesuai

Tabel 4.  
Pengujian use case profil pengguna

Butir Uji	Respon	Kesimpulan
Lihat data profil	Data profil ditampilkan	Sesuai
Ubah data profil (skenario utama)	Pesan bahwa data berhasil diubah ditampilkan	Sesuai
Ubah data profil (skenario alternatif)	Pesan error ditampilkan	Sesuai

11) Use Case Pengelolaan Klasifikasi

Use case pengelolaan klasifikasi dilakukan oleh admin atau dokter untuk melakukan pengelolaan klasifikasi termasuk membuat model klasifikasi, uji klasifikasi, serta penyetingan atribut target. Buat model klasifikasi dilakukan untuk menghasilkan model klasifikasi dari proses *learning* yang nantinya digunakan untuk klasifikasi pasien. Uji klasifikasi dilakukan untuk menguji seberapa handal model klasifikasi yang sudah dibuat untuk mengklasifikasikan data uji yang ada di dalam sistem. Penyetingan atribut target dilakukan untuk menentukan atribut mana yang digunakan untuk menentukan kelas dalam klasifikasi.

12) Use Case Pengelolaan Data Klasifikasi Pasien

Use case pengelolaan data klasifikasi pasien dilakukan oleh dokter untuk melakukan diagnosa terhadap pasien. Use case pengelolaan data klasifikasi pasien memiliki beberapa use case yaitu input data, ubah data, lihat data, dan hapus data. Dokter akan mendapatkan rekomendasi hasil klasifikasi berdasarkan model klasifikasi yang sudah dibuat dan memutuskan hasil diagnosa pasien. Dokter dapat melihat, mengubah, dan menghapus data klasifikasi pasien.

C. Perancangan Database

Rancangan database pada sistem pendukung keputusan ditunjukkan oleh Gambar (5). Perancangan database didasarkan pada perancangan use case dan fungsionalitas sistem. Tabel "tb\_pengguna", "tb\_data\_umum\_pasien", "tb\_atribut", "tb\_nilai", "tb\_data\_pembelajaran",

Tabel 5.  
Pengujian use case pengelolaan data umum pasien, data pengguna, data atribut, data nilai atribut, dan data klasifikasi pasien

Butir Uji	Respon	Kesimpulan
Input data (skenario utama)	Pesan bahwa data berhasil disimpan ditampilkan	Sesuai
Input data (skenario alternatif)	Pesan error ditampilkan	Sesuai
Ubah data (skenario utama)	Pesan bahwa data berhasil diubah ditampilkan	Sesuai
Ubah data (skenario alternatif)	Pesan error ditampilkan	Sesuai
Lihat data	Data yang diinginkan ditampilkan	Sesuai
Hapus data	Data terhapus dari database	Sesuai

Tabel 6.  
Pengujian use case pengelolaan data pembelajaran

Butir Uji	Respon	Kesimpulan
Input data (skenario utama)	Pesan bahwa data berhasil disimpan ditampilkan	Sesuai
Input data (skenario alternatif)	Pesan error ditampilkan	Sesuai
Ubah data (skenario utama)	Pesan bahwa data berhasil diubah ditampilkan	Sesuai
Ubah data (skenario alternatif)	Pesan error ditampilkan	Sesuai
Lihat data	Data yang diinginkan ditampilkan	Sesuai
Impor data	Data berhasil diimpor ke database	Sesuai
Expord data	Data berhasil tersimpan ke komputer	Sesuai
Hapus data	Data terhapus dari database	Sesuai

"tb\_data\_uji", "tb\_klasifikasi\_pasien", dan "tb\_hasil\_klasifikasi" merupakan tabel yang nantinya akan diimplementasikan pada sebuah database MySQL antara lain sebagai berikut: (1) *Informasi Tabel*. Tabel "tb\_pengguna" memuat informasi mengenai data pengguna yang dapat melakukan login ke sistem pendukung keputusan serta hak aksesnya. Tabel "tb\_data\_umum\_pasien" memuat informasi mengenai data pasien secara umum. Tabel "tb\_atribut" memuat informasi mengenai data atribut yang digunakan untuk klasifikasi beserta jenis datanya. Tabel "tb\_nilai" memuat informasi mengenai nilai untuk masing-masing atribut yang didefinisikan untuk klasifikasi. Tabel "tb\_data\_pembelajaran" memuat informasi mengenai data pembelajaran yang digunakan untuk membuat model klasifikasi. Tabel "tb\_data\_uji" memuat informasi mengenai data uji yang digunakan untuk menguji model klasifikasi. Tabel "tb\_klasifikasi\_pasien" memuat informasi mengenai data pasien yang digunakan untuk mengklasifikasikan penyakit pasien. Tabel "tb\_hasil\_klasifikasi" memuat informasi mengenai hasil klasifikasi terhadap data klasifikasi pasien.; (2) *Aspek Fleksibilitas Database*. Pada perancangan tabel, data mengenai atribut beserta nilainya dirancang untuk bisa adaptif terhadap perubahan kebutuhan diagnosa. Alih-alih mendefinisikan setiap data atribut dalam bentuk *field*, pada penelitian tugas akhir ini data atribut dan nilai atribut

Tabel 7.

Pengujian use case pengelolaan data uji

Butir Uji	Respon	Kesimpulan
Input data (skenario utama)	Pesan bahwa data berhasil disimpan ditampilkan	Sesuai
Input data (skenario alternatif)	Pesan error ditampilkan	Sesuai
Ubah data (skenario utama)	Pesan bahwa data berhasil diubah ditampilkan	Sesuai
Ubah data (skenario alternatif)	Pesan error ditampilkan	Sesuai
Lihat data	Data yang diinginkan ditampilkan	Sesuai
Impor data	Data berhasil diimpor ke database	Sesuai
Hapus data	Data terhapus dari database	Sesuai

Tabel 8.

Pengujian use case generalisasi pengelolaan klasifikasi

Butir Uji	Respon	Kesimpulan
Setting atribut target (skenario utama)	Pesan bahwa data berhasil disimpan ditampilkan	Sesuai
Setting atribut target (skenario alternatif)	Pesan error ditampilkan	Sesuai
Buat model klasifikasi	Data model klasifikasi berhasil disimpan	Sesuai
Uji klasifikasi	Hasil perhitungan akurasi, presisi, recall, dan error rate ditampilkan	Sesuai

Tabel 9. Spesifikasi server

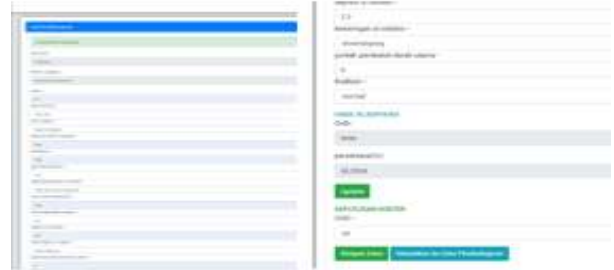
Spesifikasi	Keterangan
Processor	Intel(R) Xeon(R) CPU E3-1241 v3 @ 3.50GHz (8 core), maximum 1 core
Physical Memory	1GB
PHP version	7.2.24
MySQL version	10.1.43-MariaDB-cll-lve
Architecture	X86_x64
Operating System	Linux

dirancang untuk berekspansi dalam bentuk baris sehingga perubahan atau penambahan atribut tidak mengubah struktur tabel yang sudah diimplementasikan dalam database MySQL. Tabel Atribut berelasi *one to many* dengan tabel nilai, tabel data pembelajaran, tabel data uji, dan tabel data klasifikasi pasien dimana *field* 'id\_atribut' didefinisikan sebagai *primary key*.

### III. HASIL DAN PEMBAHASAN

#### A. Antarmuka Aplikasi

Keseluruhan sistem memiliki menu yaitu menu *login*, registrasi, beranda, profil pengguna, data pengguna, data umum pasien, atribut, nilai atribut, data pembelajaran, data uji, pengelolaan klasifikasi, serta data klasifikasi pasien. Perancangan antarmuka aplikasi sistem pendukung keputusan didasarkan atas fungsionalitas pada rancangan *use case* aplikasi secara keseluruhan. Menu yang dapat diakses oleh pengguna dengan hak akses admin adalah semua menu kecuali data umum pasien dan data klasifikasi pasien, sementara pengguna dengan hak akses petugas administrasi adalah menu *login*, registrasi, *dashboard*, dan data umum pasien, sedangkan pengguna dengan hak akses dokter adalah



Gambar 16. Hasil rekomendasi keputusan oleh sistem



Gambar 17. Pengujian trafik normal, jumlah maksimal *virtual user* = 25



Gambar 18. Pengujian trafik normal, jumlah maksimal *virtual user* = 50



Gambar 19. Pengujian saat lonjakan trafik, jumlah maksimal *virtual user* = 50

semua menu kecuali menu data pengguna. Antarmuka menu login, profil pengguna, data umum pasien, data pengguna, atribut, nilai atribut, data pembelajaran, data uji, pengelolaan klasifikasi, serta data klasifikasi pasien masing-masing ditunjukkan oleh Gambar (6) hingga Gambar (15). Dokter dapat mengakses menu data klasifikasi pasien guna memperoleh rekomendasi keputusan diagnosa. Data data yang berkaitan dengan diagnosa dapat dimasukkan sehingga didapatkan rekomendasi keputusan diagnosa seperti Gambar (16). Selanjutnya dokter dapat menyimpan data tersebut dan memasukkannya ke data pembelajaran.

#### B. Pengujian Use Case

Terdapat 12 *use case* yang diujikan pada pengujian ini. Dalam pengujian akan diberikan suatu masukan pada sistem untuk tiap *use case* tertentu dan selanjutnya akan dilihat reaksi atau respon sistem terhadap masukan tersebut. Apabila hasil respon sistem sesuai dengan fungsionalitas *use*

case maka hasil pengujian disimpulkan sesuai. Skenario normal adalah skenario bila sistem berjalan normal tanpa terjadi kesalahan, sedangkan skenario alternatif adalah skenario bila sistem tidak berjalan normal atau mengalami *error*. Hasil pengujian untuk keseluruhan *use case* ditunjukkan oleh Tabel (1) hingga Tabel (8). Berdasarkan keseluruhan hasil pengujian dapat ditarik kesimpulan bahwa perangkat lunak bebas dari kesalahan dan secara fungsional mengeluarkan respon yang sesuai dengan yang diharapkan.

### C. Pengujian Performa Sistem Pendukung Keputusan

Pengujian dilakukan terhadap sistem pendukung keputusan yang telah diimplementasikan ke dalam sebuah *server hosting*. Spesifikasi dari *server* yang digunakan ditunjukkan oleh Tabel (9). Pada pengujian, beberapa *virtual user* diarahkan untuk melakukan *request* secara paralel dan berulang ke alamat HTTP sistem pendukung keputusan yang dibuat sementara di lain sisi proses *learning* juga dilakukan terus menerus. Pengujian ditujukan untuk mengetahui bagaimana performa sistem pada saat kondisi trafik normal dan saat terdapat lonjakan trafik. Gambar (17) dan (18) merupakan pengujian pada saat kondisi trafik normal dengan jumlah maksimal *virtual user* yaitu 25 dan 50, sedangkan merupakan pengujian pada saat kondisi terdapat lonjakan trafik dengan jumlah maksimal *virtual user* yaitu 50 pada Gambar 19. Pada saat kondisi normal, berdasarkan Gambar (17) waktu respon rata-rata sistem yaitu 71ms, *request rate* sebesar 22r/s, dan jumlah *request* yang berhasil dilakukan sebanyak 7703 *request*, sementara berdasarkan Gambar (18) waktu respon rata-rata sistem sebesar 71ms, *request rate* sebesar 45r/s, dan jumlah *request* yang berhasil dilakukan sebanyak 16186 *request*. Berdasarkan pengujian saat kondisi normal, nilai *request rate* dan jumlah *request* semakin besar seiring bertambahnya jumlah maksimal *virtual user* sementara waktu respon rata-rata sistem cenderung tetap sehingga dapat disimpulkan bahwa performa sistem baik dan tidak terpengaruh perubahan jumlah *virtual user* yang melakukan *request*. Selanjutnya untuk pengujian dengan lonjakan trafik, waktu respon rata-rata sistem yaitu 72ms,

*request rate* sebesar 36r/s, dan jumlah *request* yang berhasil dilakukan sebanyak 12925 *request*. Apabila dibandingkan dengan pengujian saat kondisi normal, terlihat bahwa waktu respon sistem cenderung sama sehingga dapat disimpulkan bahwa performa sistem baik.

## IV. KESIMPULAN

Pada penelitian tugas akhir ini dirancang sebuah sistem pendukung keputusan berbasis web untuk diagnosa penyakit. Berdasarkan hasil penelitian, sistem pendukung keputusan yang dirancang berhasil diimplementasikan ke dalam sistem berbasis web dengan bahasa pemrograman PHP dengan RDBMS MySQL. Sistem pendukung keputusan yang dirancang dapat digunakan untuk membantu diagnosa suatu penyakit sesuai dengan kebutuhan. Terdapat mekanisme yang mengatur akses ke dalam sistem pendukung keputusan dan integrasi dengan data pasien. Berdasarkan pengujian performansi sistem pada saat kondisi normal, performansi sistem baik dan tidak terpengaruh jumlah *user* yang melakukan akses. Selain itu tidak terdapat perbedaan performa yang signifikan saat kondisi normal dan pada saat kondisi terdapat lonjakan trafik.

## DAFTAR PUSTAKA

- [1] R. B. H. and S. E. H. D. L. Hunt, "Effects of Computer-Based Clinical Decision Support Systems on Physician Performance and Patient Outcomes: a Systematic Review," *J. Am. Med.*, vol. 280, no. 15, pp. 1339–1346, 1998.
- [2] B. Malmir, M. Amini, and S. I. Chang, "A medical Decision Support System for Disease Diagnosis Under Uncertainty," *Expert Syst. Appl.*, vol. 88, pp. 95–108, Dec. 2017.
- [3] Y. Kaya and M. Uyar, "A Hybrid Decision Support System Based on Rough Set and Extreme Learning Machine for Diagnosis of Hepatitis Disease," *Appl. Soft Comput. J.*, vol. 13, no. 8, pp. 3429–3438, Aug. 2013.
- [4] S. Nazari, M. Fallah, H. Kazemipoor, and A. Salehipour, "A Fuzzy Inference- Fuzzy Analytic Hierarchy Process-Based Clinical Decision Support System for Diagnosis of Heart Diseases," *Expert Syst. Appl.*, vol. 95, pp. 261–271, Apr. 2018.
- [5] D. Siahaan, *Analisa Kebutuhan dalam Rekayasa Perangkat Lunak*, 1st ed. Yogyakarta: Penerbit Andi, 2012.
- [6] F. Mauladani and D. O. Siahaan, "Perancangan SMKI Berdasarkan SNI ISO/IEC27001:2013 dan SNI ISO/IEC27005:2013 (Studi Kasus DPTSI-ITS)," *CSRID (Computer Sci. Res. Its Dev. Journal)*, vol. 10, no. 1, p. 32, Mar. 2018.
- [7] P. Gelu, R. Sarno, and D. Siahaan, "Requirements Association Extraction based on Use Cases Diagram," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 9, no. 1, pp. 11–19, May 2018.
- [8] A. M. Yuwantoko, S. Daniel, and A. S. Ahmadiyah, "Pembuatan Kakas Bantu untuk Mendeteksi Ketidaksesuaian Diagram Urutan (Sequence Diagram) dengan Diagram Kasus Penggunaan (Use Case Diagram)," *J. Tek. ITS*, vol. 6, no. 1, pp. 64–70, Feb. 2017.