



# Plagiarism Checker X Originality Report

**Similarity Found: 16%**

Date: Selasa, April 07, 2020

Statistics: 1080 words Plagiarized / 6959 Total words

Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

---

BAB I PENDAHULUAN Latar Belakang Masalah Penyakit di bagian tenggorokan khususnya penyakit laringitis masih menjadi masalah kesehatan yang darurat di kalangan masyarakat. Masalah ini tidak menutup kemungkinan terjadi terhadap generasi millennial yang disebut sebagai generasi yang memiliki investasi kesehatan lebih baik daripada generasi sebelumnya. Untuk itu perlu adanya sesuatu inovasi yang memanfaatkan perkembangan teknologi di bidang kecerdasan buatan atau biasa disebut Artificial Intelligence untuk membantu dalam penyelesaian masalah agar tidak terjadi keterlambatan penanganan terhadap penyakit tersebut.

Kondisi tersebut di atas, mengakibatkan banyak dari penderita penyakit laringitis mengalami keterlambatan dalam penanganan penyakit tersebut yang mengakibatkan kondisi yang lebih parah terhadap penyakit laringitis, apalagi berdasarkan pengamatan yang dilakukan dan kurangnya pengetahuan masyarakat tentang penyakit manusia serta keterbatasan petugas medis yang tersebar di berbagai daerah mengakibatkan terlambatnya penanganan dini terhadap pasien yang menderita penyakit laringitis. Hal ini tentunya berdampak pada kondisi kesehatan masyarakat terutama mereka yang tinggal di daerah yang jauh dari rumah sakit ataupun klinik dan puskesmas maka dirancang sebuah sistem E-Healthcare untuk mempermudah dalam proses kesimpulan diagnosa penyakit

tersebut untuk selanjutnya dilakukan pencegahan dan pengobatan oleh dokter yang terkait. Penelitian ini akan menjelaskan bagaimana pemanfaatan sistem pakar atau E-Healthcare dalam menyelesaikan masalah terkait dengan penyakit laringitis.

Dalam beberapa referensi menjelaskan bahwasanya sistem pakar atau E-Healthcare dapat memecahkan permasalahan diantaranya adalah membantu mendiagnosa penyakit Talasemia [1] dan juga membantu mendiagnosa penyakit faringitis [2]. Dari referensi tersebut dapat dilihat bahwasanya sistem pakar atau E-Healthcare dapat menyelesaikan masalah yang berhubungan dengan penyakit manusia. Dalam sistem pakar konsep untuk membantu mendiagnosa penyakit manusia dapat diadopsi dengan beberapa metode salah satunya adalah metode certainty factor [3].

Untuk menegaskan bahwasanya metode certainty factor dapat diterapkan dalam proses terhadap diagnosa penyakit manusia maka diambil beberapa referensi. Dalam beberapa referensi metode certainty factor dapat diterapkan dalam beberapa persoalan terkait dengan penyakit manusia diantaranya digunakan dalam mendiagnosa penyakit Pankreatitis [4], selain itu dalam referensi lain metode certainty factor dapat menyelesaikan masalah terkait dengan penyakit lupus [5]. dari referensi referensi tersebut terlihat metode certainty factor dapat dinyatakan sebagai solusi metode untuk penyelesaian masalah dalam proses diagnosa penyakit manusia.

Penerapan metode certainty factor akan digunakan dalam penelitian ini, dimana data gejala penyakit laringitis yang didapatkan dari seorang pakar diolah sesuai dengan algoritma metode tersebut untuk menghasilkan informasi diagnosa yang akurat terhadap penyakit laringitis. Dari penelitian ini diharapkan mampu menghasilkan sebuah sistem berbasis web yang mengadopsi metode certainty factor yang dapat membantu menyelesaikan masalah khususnya dalam mendiagnosa penyakit laringitis. Penelitian ini juga diharapkan dapat menjadi solusi bagi dokter maupun masyarakat dalam menghasilkan kesimpulan diagnosa penyakit agar terjadi ke akuratan dalam proses diagnosa terhadap penyakit laringitis.

Berdasarkan permasalahan yang telah diuraikan tersebut, maka dilakukan penelitian skripsi dengan judul "Sistem E-Healthcare Untuk Mendiagnosa Penyakit Laringitis Dengan Menggunakan Metode Certainty Factor". Rumusan Masalah Berdasarkan uraian dan deskripsi dari masalah yang telah dijelaskan pada latar belakang berikut ini adalah rumusan masalah yang diuraikan sebagai berikut: Apakah metode certainty factor dapat diterapkan dalam pemecahan masalah terkait dengan penyakit laringitis? Bagaimana memodelkan sistem E-Healthcare yang dirancang sehingga tepat sesuai dengan yang diharapkan? Bagaimana merancang sebuah sistem E-Healthcare yang mengadopsi metode certainty factor di dalam menyelesaikan masalah terkait penyakit laringitis?

Bagaimana mengimplementasikan sebuah sistem E-Healthcare yang mengadopsi metode certainty factor di dalam menyelesaikan masalah terkait penyakit laringitis? 1.3

Batasan Masalah Agar masalah dalam penelitian ini lebih mudah dipahami sejauh mana masalah dalam penelitian ini di bahas, maka berikut ini adalah batasan masalah dalam penelitian ini yaitu: Dalam penelitian ini hanya membahas masalah terkait penyakit dan gejala klinis yang di alami dalam mengenali penyakit laringitis tidak membahas penyakit selain yang di teliti. Adapun sumber data primer dari penelitian ini adalah hasil wawancara dengan pihak Fakultas Kedokteran USU Program Studi Doktoral (S3). Adapun metode perancangan sistem yang digunakan adalah waterfall algorithm. Adapun sistem yang akan dirancang berbasis web programming. 1.4

Tujuan Penelitian Berdasarkan rumusan masalah yang telah dijelaskan pada sub bab sebelumnya, maka tujuan dari penelitian ini yaitu : Untuk memastikan bahwasanya metode certainty factor dapat diterapkan dalam pemecahan masalah terkait dengan penyakit laringitis. Untuk memodelkan sistem E-Healthcare yang dirancang sehingga tepat sesuai dengan yang diharapkan. Untuk merancang sebuah sistem E-Healthcare yang mengadopsi metode certainty factor di dalam menyelesaikan masalah terkait penyakit laringitis Untuk mengimplementasikan sebuah sistem E-Healthcare yang mengadopsi metode certainty factor di dalam menyelesaikan masalah terkait penyakit laringitis.

Manfaat Penelitian Adapun manfaat yang dapat diperoleh dalam penelitian ini yaitu sebagai berikut : Secara teoritis, penelitian ini dapat menjadi rujukan atau referensi dari penulis lain untuk dapat menerapkan metode certainty factor dalam kasus penyakit manusia lainnya. Secara akademis, penelitian ini merupakan salah satu syarat untuk menyelesaikan studi strata-1 di program studi sistem informasi dan juga dapat menjadi referensi bagi pembaca untuk memahami konsep metode certainty factor dan juga bagaimana menerapkan metode certainty factor dalam sebuah aplikasi atau sistem.

Secara praktis, penelitian ini memiliki manfaat di antaranya yaitu: Dapat membantu seorang pakar atau dokter untuk memberikan informasi dalam proses diagnosa terkait penyakit laringitis melalui sistem E-Healthcare yang mengadopsi metode certainty factor. Dapat mempermudah pasien atau penderita yang terindikasi gejala penyakit laringitis untuk selanjutnya diberikan solusi agar tidak timbul hal yang lebih parah terhadap penyakit tersebut. Dapat menghasilkan informasi diagnosa yang sesuai dengan gejala yang di alami oleh pasien atau penderita.

BAB II KAJIAN PUSTAKA Penyakit Laringitis Laringitis yaitu suatu penyakit peradangan pada laring yang sering mengakibatkan suara serak bahkan sampai kehilangan suara. Secara umum penyakit laringitis ini dapat bersifat ringan, akut maupun kronis. Pada laringitis kronis sering terjadi terhadap para perokok dan penderita gastroesophageal reflux (GERD). Selain itu, penggunaan suara secara berlebihan dan juga bernyanyi yang berlebihan dapat mengakibatkan laringitis kronis serta faktor seiring bertambahnya usia pita suara mengalami kehilangan kemampuan untuk bergetar serta dapat membuatnya menjadi lebih rentan terhadap laringitis kronis [6]. Pada laringitis akut biasanya hanya akan berlangsung dalam beberapa hari sedangkan laringitis kronis dapat bertahan sampai lebih dari 3 minggu.

Infeksi virus adalah factor penyebab paling sering dari laringitis akut sedangkan refluks asam merupakan factor penyebab yang paling sering dari laringitis kronis. Jika laringitis kronis sudah berat dapat mengakibatkan pneumonia. Penyebab laringitis pada dasarnya yaitu infeksi virus, infeksi bakteri dan jamur, peradangan yang berlebihan pada pita suara, batuk yang berlebih, merokok, mengkonsumsi alkohol dan alergi [7]. Laringitis bisa menyerang semua kalangan usia termasuk anak-anak. Hampir semua orang dapat terkena laringitis ringan, akut maupun kronis.

Penyebab dari laringitis ringan umumnya penggunaan suara secara berlebihan

Sedangkan penyebab laringitis akut adalah bakteri yang mengakibatkan radang lokal atau virus yang membuat peradangan sistemik. Biasanya merupakan perluasan dari radang saluran nafas atas oleh bakteri Haemophilus Influenza, Staphylococcus, Streptococcus, dan Pneumonia. Laringitis kronis nonspesifik dapat diakibatkan faktor eksogen (rangsangan fisik akibat penyalahgunaan suara, rangsangan kimia, infeksi kronis saluran nafas atas dan bawah, asap rokok) atau faktor endogen (bentuk dari tubuh, kelamin yang metabolik) dan yang spesifik disebabkan oleh tuberculosus [7]. Pada laringitis akut terdapat beberapa gejala umum yaitu demam, malaise, gejala rinofaringitis dan batuk disertai suara parau sehingga tidak bersuara sama sekali.

Gejala yang mula-mula muncul adalah rasa kering ditenggorok, nyeri ketika akan menelan dan berbicara. Sering juga disertai dengan batuk kering dan lama-kelamaan akan muncul batuk dengan dahak yang kental. Pada proses yang berkelanjutan sering menyebabkan gejala sumbatan jalan pada napas bagian atas sampai sianosis hal ini biasanya sering terjadi pada anak tetapi juga bisa terjadi terhadap orang dewasa . Pada pemeriksaan laringoskopi nampak mukosa laring kemerahan dan juga mengalami pembengkakan. Gerakan pada pita suara tidak terganggu kecuali apabila sudah terjadi endema pada pita suara [7].

Pada laringitis kronis serak adalah gejala yang paling umum terjadi. Biasanya gejala serak akan terus berubah-ubah sepanjang hari biasanya yang paling parah di pagi hari. Dan sering disertai batuk karena adanya secret lengket dan juga kental di tenggorokan. laringitis kronis ditandai dengan munculnya suara serak yang persisten. Laringitis kronis bisa saja terjadi akibat komplikasi dari sinusitis kronis dan bronchitis kronis [7]. Berikut ini merupakan gejala-gejala laringitis yang diperoleh dari seorang pakar Prof.Dr.dr Delfitri Munir yaitu sebagai berikut: No \_Gejala Penyakit Laringitis \_1 \_Suara Lebih Serak,Berat atau Hilang \_2 \_Batuk Kering \_3 \_Demam \_4 \_Nyeri Tenggorokan \_5 \_Pembesaran Kelenjar Getah Bening di Leher \_6 \_Hidung Berair \_7 \_Tenggorokan Kering atau Gatal \_8 \_Kesulitan Bernafas \_9 \_Terdapat Dahak di Tenggorokan \_10 \_Sensasi menggelitik pada tenggorokan \_

Tabel 2.1 Gejala Penyakit Laringitis (Sumber Prof.Dr.dr Delfitri Munir,Sp.THT-KL(K) Adapun solusi yang diberikan oleh seorang pakar Prof.Dr.dr Delfitri Munir jika terdiagnosa penyakit laringitis adalah sebagai berikut :

Tabel 2.2

Solusi Penyakit Laringitis Penyakit Laringitis \_Solusi \_ \_Ringan \_Istirahatkan suara dari aktivitas yang berlebihan \_ \_Perbanyak minum air putih \_ \_Menonsumsi obat antihistamin \_ \_Akut \_Istirahatkan Suara dari aktivitas yang berlebihan \_ \_Berhenti Merokok dan Perbanyak Minum Air Putih \_ \_Menghirup Inhaler dengan kandungan mentol \_ \_Mengonsumsi permen mint dan berkumur dengan air garam hangat atau obat kumur \_ \_Kronis \_Terapi Istirahat berbicara atau bernyanyi dengan nada tinggi. \_ \_

\_Kortikosteroid, untuk meredakan peradangan pada pita suara \_\_ \_Antibiotik untuk menangani infeksi bakteri \_\_ \_Menggunakan pelembab udara di ruangan \_\_ \_Perbanyak minum air putih dan mengkonsumsi obat mentholspritus 5% \_\_ (Sumber Prof.Dr.dr Delfitri Munir,Sp.THT-KL(K) Pengertian E-Healthcare "World Health Organization (WHO) mendefinisikan e-health sebagai the use of information and communication technologies (ICT) for health.

Sesederhana itu, karena peran dan potensinya untuk mendorong perbaikan dan perubahan pelayanan kesehatan, WHO telah mengeluarkan resolusi mengenai e-health bernomor 58.28 tahun 2005" [8]. "Dalam Keputusan Menteri Kesehatan Nomor 192/MENKES/SK/VI/2012 disebutkan bahwa e-health adalah pemanfaatan teknologi informasi dan komunikasi di sektor kesehatan terutama untuk meningkatkan pelayanan kesehatan" [8]. E-Health adalah aplikasi internet atau teknologi lain yang berkaitan di industri pelayanan kesehatan yang bertujuan untuk meningkatkan akses, efisiensi, efektivitas, dan kualitas dari proses medis [8].

Menurut Georgeff (2007), Dalam [9] "e-health adalah penggunaan teknologi informasi dan komunikasi untuk penanganan kesehatan, seperti pengobatan pasien, mendidik masyarakat, mendeteksi penyakit dan memantau kesehatan masyarakat". Soegijoko (2010) menyatakan bahwa "e-health adalah penggunaan teknologi informasi dan komunikasi untuk memproses berbagai jenis informasi kedokteran, guna melaksanakan pelayanan klinis, administrasi serta pendidikan atau pembelajaran pada bidang kesehatan" [9]. Menurut Deloitte (2008), dalam [9] "e-health adalah cara untuk memastikan bahwa informasi kesehatan yang tepat diberikan ke orang yang tepat di tempat dan waktu yang tepat dan aman dalam bentuk elektronik". 2.3

Sistem Pakar Sistem pakar yaitu cabang dari artificial intelligence yang termasuk cukup tua karena sistem ini mulai dikembangkan pada pertengahan tahun 1960. Sistem pakar yang muncul kali pertama adalah General purpose problem solver (GPS) yang kemudian dikembangkan oleh newel dan simon [10]. Sistem pakar (expert system) merupakan cabang ilmu kecerdasan buatan dan juga merupakan bidang keilmuan yang muncul sejalan dengan perkembangan ilmu komputer saat ini.

Sistem ini adalah sistem komputer yang dapat menyamai atau menirukan kemampuan seorang pakar, sistem ini biasa bekerja dengan memasukkan pengetahuan manusia ke dalam komputer dengan menggabungkan sebuah dasar pengetahuan (knowledge base) dengan sebuah sistem inferensi untuk menggantikan peran seorang pakar untuk menyelesaikan suatu masalah [11]. Dari penjelasan tersebut dapat disimpulkan bahwa sistem pakar adalah suatu sistem yang mengadopsi pengetahuan seorang pakar untuk kemudian diadopsi ke dalam sistem komputer dengan menggabungkan teknik inferensi

dan dasar pengetahuan untuk menyelesaikan suatu masalah. Menurut Fanny, dkk (2017:13) dalam [12] mengatakan bahwa "Sistem pakar yaitu sistem yang berusaha mengadopsi pengetahuan manusia ke komputer yang dirancang untuk memodelkan kemampuan menyelesaikan masalah seperti layaknya seorang pakar".

Dari pengertian tersebut disimpulkan bahwa sistem pakar yaitu suatu sistem yang di dalamnya berisi pengetahuan seorang pakar yang dipindahkan ke sebuah sistem yang nantinya akan diolah oleh program agar dapat digunakan.

Sistem pakar membuat seseorang yang kurang memahami suatu permasalahan terbantu melalui aplikasi sistem pakar. Bagi para pakar itu sendiri, sistem pakar dapat membantu aktivitas-aktivitas seperti layaknya seorang asisten yang berpengalaman.

2.3.1 Manfaat Sistem Pakar Secara umum, banyak manfaat yang dapat diperoleh dengan adanya sistem pakar, antara lain : Memungkinkan setiap orang bisa mengerjakan pekerjaan para ahli. Dapat melakukan proses secara berulang secara otomatis.

Menyimpan pengetahuan dan keahlian dari para pakar. Meningkatkan output dan juga produktivitas. Meningkatkan sebuah kualitas. Mampu menyerap dan juga melestarikan keahlian dari para pakar (terutama yang masuk dalam keahlian langka). Bisa beroperasi di dalam lingkungan yang berbahaya. Memiliki kemampuan yang bisa mengakses pengetahuan. Memiliki sifat reliabilitas (kehandalan). Memiliki kemampuan yang dapat bekerja dengan informasi yang tidak lengkap maupun yang mengandung ketidakpastian. Dapat menjadi suatu media pelengkap dalam penelitian. Meningkatkan kapabilitas dalam penyelesaian suatu masalah. Menghemat waktu di dalam pengambilan keputusan 2.3.2

Keunggulan Sistem Pakar Sistem pakar memiliki beberapa keunggulan adapun keunggulan sistem pakar adalah sebagai berikut: Memungkinkan orang awam agar bisa mengerjakan pekerjaan seorang pakar. Meningkatkan setiap produktivitas dalam kerja dengan cara meningkatkan efisiensi pekerjaan. Menghemat waktu di dalam menyelesaikan pekerjaan atau permasalahan yang rumit. Pengolahan yang dilakukan secara berulang-ulang secara otomatis. Tersedianya pengetahuan dari pakar untuk masyarakat luas. 2.3.3 Kelemahan Sistem Pakar Disamping memiliki beberapa keunggulan, sistem pakar juga memiliki beberapa kelemahan diantaranya : Biaya yang dibutuhkan untuk membuat dan memeliharanya sangat mahal.

Sulit berkembang dikarenakan terbatasnya ketersediaan pakar dibidangnya. Sistem pakar tidak memiliki 100% nilai kebenaran. 2.3.4 Ciri-ciri Sistem Pakar Di dalam sistem pakar terdapat beberapa ciri-ciri yang menggambarkan sistem pakar diantara sebagai berikut. Terbatas pada domain keahlian tertentu Mampu memberikan penalaran terhadap sebuah data yang tidak lengkap dan juga tidak pasti. Dapat menjelaskan alasan-alasan yang diberinya dengan cara yang mudah dipahami. Bekerja dengan dasar kaidah atau rule tertentu.

Mudah dimodifikasi. Basis pengetahuan dan juga mekanisme inferensi terpisah. Keluarannya atau output bersifat anjuran. Sistem pakar dapat mengaktifkan kaidah secara searah yang sesuai, tuntunan dialog dengan pengguna. 2.3.5 Area Permasalahan Sistem Pakar Biasanya aplikasi sistem pakar menyentuh beberapa area permasalahan sebagai berikut: [13]. Interpretasi : yaitu pengambilan keputusan dari hasil observasi,

yang termasuk diantaranya pengawasan, pengenalan, pengucapan, analisis terhadap citra, interpretasi sinyal, dan analisis kecerdasan.

Prediksi : Termasuk di dalamnya sebuah peramalan, prediksi dalam demografis, peramalan tentang ekonomi, prediksi dalam lalu lintas, estimasi hasil, dalam militer, pemasaran, dan peramalan keuangan. Diagnosis : Termasuk di dalamnya medis, elektronis, mekanis, dan diagnosis perangkat lunak Perancangan : Termasuk diantaranya adalah layout sirkuit dan perancangan bangunan. Perencanaan : Termasuk diantaranya adalah perencanaan keuangan, perencanaan komunikasi, militer, pengembangan produk, routing, dan manajemen proyek. Monitoring : Misalnya yaitu Computer-Aided Monitoring Systems.

Debugging : memberikan solusi resep terhadap suatu kegagalan. Repair : melakukan perbaikan Instruksi : Melakukan instruksi untuk setiap proses diagnosis, debugging, dan perbaikan kerja. Kontrol : Melakukan kontrol terhadap interpretasi, terhadap prediksi, terhadap perbaikan, dan monitoring kelakulan sistem. 2.3.6 Konsep Dasar Sistem Pakar Sistem pakar terbentuk dari beberapa konsep yang harus dimilikinya. Konsep dasar dalam sistem pakar mencakup enam hal berikut ini yaitu: Kepakaran (Expertise) Kepakaran yaitu suatu pengetahuan yang didapat dari pelatihan membaca dan juga pengalaman.

Kepakaran memungkinkan seorang pakar untuk dapat mengambil keputusan dengan lebih cepat dan **baik daripada seseorang yang bukan pakar. Kepakaran** memiliki pengetahuan tentang [14] : a. Fakta-fakta yang memuat bidang permasalahan tertentu. Teori-teori yang memuat bidang permasalahan tertentu. Aturan dan prosedur-prosedur menurut bidang permasalahan pada umumnya Aturan heuristic yang harus dikerjakan di dalam situasi tertentu. Strategi global yang dipergunakan untuk memecahkan permasalahan. Pengetahuan yang memuat pengetahuan (meta knowledge). Pakar (Expert) Pakar adalah seorang yang mempunyai pengetahuan, pengalaman dan juga metode khusus serta mampu menerapkannya dalam memecahkan masalah atau memberikan nasihat.

Seorang pakar diwajibkan untuk menjelaskan dan mempelajari hal-hal baru yang erat kaitannya dengan topik permasalahan, jika perlu harus mampu menyusun kembali pengetahuan yang dimilikinya dan dapat memecahkan aturan-aturan serta menemukan relevansi kepakarannya. Seorang pakar dapat melakukan kegiatan-kegiatan berikut ini : Mengenali serta memformulasikan permasalahan Dapat **memecahkan permasalahan secara cepat dan tepat Menerangkan pemecahan** masalah Belajar dari setiap pengalaman **Merestrukturisasi pengetahuan Memecahkan aturan-aturan Menentukan relevansi** Pemindahan Kepakaran (Transferring Expertise) Tujuan dari sebuah sistem

pakar yaitu memindahkan kepakaran seorang pakar ke dalam sebuah sistem komputer untuk kemudian di transfer kepada orang yang bukan pakar dan dalam proses ini akan melibatkan empat kegiatan, yaitu: Transfer atau akuisisi pengetahuan (dari pakar atau sumber yang lain) Representasi pengetahuan pada komputer.

Inferensi pengetahuan Pemindahan pengetahuan ke pengguna Inferensi (inferencing) Inferensi merupakan sebuah prosedur yang memiliki kemampuan dalam melakukan sebuah penalaran. Inferensi dapat ditampilkan terhadap suatu komponen yang dinamakan mesin inferensi yang mencakupi prosedur-prosedur tentang pemecahan masalah. Semua pengetahuan yang dimiliki seorang pakar akan disimpan pada basis pengetahuan oleh sebuah sistem pakar. Tugas mesin inferensi nantinya mengambil sebuah kesimpulan berdasarkan basis pengetahuan yang dimiliki sistem pakar.

Aturan-aturan (rule) Kebanyakan software sistem pakar bersifat komersial yaitu sebuah sistem yang berbasis rule, sebuah pengetahuan disimpan terutama di dalam bentuk rule, sebagai langkah-langkah dalam pemecahan masalah. Kemampuan Menjelaskan (Explanation Capability) Fasilitas lain yang dimiliki sistem pakar adalah kemampuannya yang bisa memberikan saran atau rekomendasi yang diperoleh dari sistem pakar. Penjelasan dilakukan dalam sub sistem yang disebut sub sistem penjelasan (explanation).

Bagian dari sistem ini memungkinkan sistem untuk dapat memeriksa penalaran yang dibuatnya sendiri untuk menjelaskan operasi-operasinya Karakteristik serta kemampuan yang dimiliki oleh sistem pakar berbeda dengan sistem konvensional. Perbedaan ini dapat dilihat pada tabel berikut ini: Tabel 2.3 Perbandingan sistem pakar dengan sistem konvensional Sistem Konvensional \_Sistem Pakar \_ Informasi dan pemrosesannya biasanya digabungkan dalam suatu program \_Basis pengetahuan dipisahkan secara jelas dengan metode inferensi.

\_ Program tidak membuat kesalahan (yang membuat kesalahan pemrogram atau pengguna) \_ Program dapat membuat kesalahan \_ Biasanya tidak menjelaskan mengapa data masukan diperlukan atau bagaimana output dihasilkan \_ Penjelasan yang merupakan bagian terpenting dari semua sistem pakar \_ Perubahan program sangat menyulitkan \_ Perubahan dalam aturan-aturan mudah untuk dilakukan \_ Tabel 2.3 Perbandingan sistem pakar dengan sistem konvensional (lanjutan) Sistem Konvensional \_Sistem Pakar \_ Eksekusi dapat dilakukan langkah demi langkah (algoritmik) \_ Eksekusi dapat dilakukan dengan menggunakan heuristik dan logika pada seluruh basis pengetahuan \_ Sistem ini hanya bisa beroperasi setelah lengkap atau selesai \_ Sistem ini dapat beroperasi hanya dengan aturan-aturan yang sedikit (sebagai prototipe awal) \_ Mudah berurusan dengan data kuantitatif \_ Mudah berurusan dengan data kualitatif \_ Manipulasi efektif dari basis data yang besar \_ Manipulasi yang efektif dari basis

pengetahuan yang besar \_ \_Menggunakan data \_Menggunakan pengetahuan \_  
\_Menangkap, dan juga menambah serta mendistribusikan akses ke data numerik atau  
informasi \_Menangkap, dan juga menambah serta mendistribusikan akses ke  
pertimbangan dan pengetahuan \_ \_Tujuan utama : efisiensi \_Tujuan utama : efektivitas \_  
\_ (Sumber : T. Sutojo dkk, Kecerdasan Buatan 2011) 2.3.7

Struktur Sistem Pakar Terdapat dua bagian penting dalam sistem pakar, diantaranya lingkungan pengembangan (development environment) dan juga lingkungan konsultasi (consultation environment). Lingkungan pengembangan dipergunakan oleh pembuat sistem pakar untuk membangun sebuah komponen-komponen dan memasukkan pengetahuan ke dalam sebuah basis pengetahuan (knowledge base) yang digunakan untuk melakukan konsultasi agar mendapat sebuah nasihat seperti layaknya berkonsultasi dengan seorang pakar [14]. Adapun komponen-komponen yang penting dalam sebuah sistem pakar dapat dilihat pada gambar berikut ini : / ( Sumber : T. Sutojo dkk, Kecerdasan Buatan 2011) Gambar 2.1

Struktur Sistem Pakar Keterangan gambar adalah sebagai berikut : Akuisisi Pengetahuan (Knowledge Acquisition) Sub sistem ini digunakan untuk memasukkan pengetahuan dari seorang pakar caranya adalah merekayasa pengetahuan agar dapat diproses oleh komputer dan menaruhnya ke dalam basis pengetahuan dalam bentuk representasi pengetahuan. Sumber pengetahuan di dapatkan dari pakar, buku, dokumen multimedia, basis data, laporan riset khusus, dan informasi yang terdapat pada web. Basis Pengetahuan (Knowledge Base) Basis pengetahuan yaitu merupakan inti dari sebuah sistem pakar, yaitu merupakan representasi pengetahuan dari seorang pakar. Basis pengetahuan tersusun dari fakta dan kaidah.

Fakta yaitu informasi tentang objek, peristiwa, atau situasi. Sedangkan kaidah adalah sebuah cara untuk membangkitkan suatu fakta baru yang didapatkan dari fakta yang sudah diketahui. Mesin Inferensi Merupakan bagian dari sebuah sistem pakar yang dapat melakukan penalaran dengan menggunakan daftar aturan berdasarkan urutan pola tertentu. Selama dalam proses konsultasi baik sistem dan pemakai, mekanisme inferensi menguji aturan dengan cara satu persatu sampai kondisi aturan itu benar.

Secara umum terdapat dua teknik utama yang digunakan untuk mekanisme inferensi pengujian aturan, yaitu penalaran maju dan penalaran mundur. Daerah Kerja (Blackboard) Untuk dapat merekam hasil sementara yang nantinya dijadikan sebagai keputusan dan juga untuk dapat menjelaskan tentang masalah yang sedang terjadi, sistem pakar membutuhkan sebuah blackboard, yaitu area yang terdapat pada memori dan berfungsi sebagai basis data. Antarmuka Pemakai (User Interface) Dipergunakan sebagai media komunikasi antara pengguna dan sistem pakar.

Komunikasi ini paling bagus apabila disajikan dalam bentuk bahasa alami (natural language) juga dilengkapi dengan grafik, menu, dan formulir elektronik. **Subsistem Penjelasan (Explanation Subsystem / Justifier) Subsistem penjelasan** yaitu komponen tambahan untuk meningkatkan kemampuan sistem pakar, yang **digunakan untuk melacak respon dan memberikan penjelasan tentang** perilaku sistem pakar secara interaktif melalui pertanyaan. 2.4 Metode Certainty Factor Dalam [15] menyebutkan bahwa "Teori Certainty Factor pertama kali digagas oleh Shortliffe dan Buchanan pada 1975 yang mengadopsi permasalahan ketidakpastian oleh seorang pakar.

Metode Certainty Factor ini dipilih ketika menghadapi suatu permasalahan atau kejadian yang tidak pasti dalam jawaban". Seorang pakar, (contohnya dokter) sering sekali menganalisa informasi yang ada dengan sebuah ungkapan misalnya : mungkin, besar kemungkinan, hampir pasti. Untuk mengakomodasi hal ini dapat menggunakan certainty factor (CF) guna memberikan **tingkat keyakinan pakar terhadap masalah yang dihadapi** [16].

Berdasarkan Kusri (2008) dalam [17], certainty factor atau faktor kepastian juga merupakan cara penggabungan kepercayaan dan ketidakpercayaan dalam sebuah bilangan certainty theory tunggal. Dalam data-data kualitatif disajikan sebagai derajat keyakinan (degree of belief). Terdapat dua langkah dalam mempresentasikan data-data kualitatif. Langkah yang **pertama adalah kemampuan untuk mengekspresikan derajat keyakinan sesuai dengan metode** yang dibahas sebelumnya, langkah kedua adalah kemampuan menempatkan dan mengkombinasikan derajat keyakinan yang dimiliki dalam sistem pakar.

Certainty Factor memberikan **ukuran kepastian terhadap suatu fakta atau aturan nilai yang tertinggi dalam CF adalah + 1.0 (pasti benar atau Definity) dan juga nilai terendah dalam CF adalah -1,0 (pasti salah atau Definity not) nilai positif** mempresentasikan terhadap derajat keyakinan. Dan nilai negatif mempresentasikan derajat ketidakpercayaan [10]. Terdapat dua cara untuk mendapatkan tingkat keyakinan (CF) dari sebuah rule yaitu : [4]. Metode Net Belief yang diusulkan oleh E.H. Shortliffe dan B.G.

Buchanan  $CF(Rule) = MB(H,E) - MD(H,E)$   $MB(H,E) = \max(P(H|E), P(H) - P(H)P(H) = 1)$   $MD(H,E) = \min(P(H|E), P(H) - P(H)P(H) = 0)$  Di mana : **CF(Rule) = faktor kepastian** **MB(H,E) = measure of belief (ukuran kepercayaan) terhadap hipotesis H, jika diberikan evidence E (antara 0 dan 1)** **MD(H,E) = measure of disbelief (ukuran ketidakpercayaan) terhadap evidence H, jika diberikan evidence E (antara 0 dan 1)** **P(H) = probabilitas kebenaran hipotesis H** **P(H|E) = probabilitas bahwa H benar karena fakta E** Dengan cara mewawancarai seorang pakar Nilai CF(Rule) didapat dari interpretasi "term" dari pakar,



persentase : Persentase keyakinan = CF combine \* 100 % [16]. 2.4.2

Kelebihan Dan Kekurangan Metode Certainty Factor Adapun kelebihan dan kekurangan dari metode certainty factor adalah sebagai berikut [18]: Kelebihan metode certainty factor Metode ini cocok digunakan dalam sistem pakar yang mengandung sebuah ketidakpastian. Dalam satu kali proses perhitungan hanya dapat mengolah 2 data saja, sehingga keakuratan dari data tersebut dapat terjaga. Kekurangan metode certainty factor Pemodelan ketidakpastian yang mengadopsi perhitungan metode Certainty Factor biasanya masih akan diperdebatkan. Untuk sebuah data yang melebihi dari 2 buah, harus dilakukan beberapa kali pengolahan terhadap data tersebut.

Nilai CF yang diberikan pakar bersifat subyektif karena setiap penilaian bisa saja berbeda-beda tergantung pengetahuan dan pengalaman pakar. Pemodelan Sistem Pemodelan adalah proses membuat suatu model. Model adalah representasi (sederhana) cara kerja suatu sistem. Suatu model mirip dengan sistem yang direpresentasikan, tetapi lebih sederhana. Pemodelan diperlukan dalam proses analisis sistem, tujuannya agar kita dapat memprediksi perubahan yang mungkin terjadi pada sistem. Permasalahan penting dalam pemodelan adalah kebenaran model tersebut [19].

Sedangkan sistem adalah sesuatu yang mempunyai aksi untuk dilakukan dan bergantung pada banyak objek yang disebut entitas dari sistem tersebut [19]. UML (Unified Modeling Language) Menurut (Braun, et. al. 2001) [20] " Unified Modelling Language (UML) merupakan suatu alat untuk memvisualisasikan juga mendokumentasikan hasil analisa dan desain yang berisi sintaks dalam memodelkan suatu sistem secara visual" . Sedangkan menurut (Whitten, et. al. 2004) [20] "Juga merupakan suatu kumpulan konvensi pemodelan yang dipergunakan untuk menentukan atau menggambarkan sebuah sistem atau software yang berkaitan dengan objek".

Pengenalan UML Sejarah dari UML (Unified Modelling Language) sendiri terbagi dalam dua fase penting sebelum dan sesudah kemunculan UML. Dalam fase sebelum kemunculan sebenarnya UML sudah mulai diperkenalkan sejak awal tahun 1990an namun karena notasi yang dikembangkan oleh para ahli analisis dan desain berbeda-beda, sehingga UML dikatakan belum memiliki standarisasi [20]. Fase kedua dilandasi pemikiran untuk menyatukan metode tersebut dan digagas oleh Object Management Group (OMG) maka pengembangan UML dimulai pada akhir tahun 1994 ketika Grady Booch dengan metode OOD (Object-Oriented Design), Jim Rumbaugh dengan metode OMT (Object Modelling Technique) mereka semua bekerja pada Rational Software Corporation dan Ivar Jacobson dengan metode OOSE (Object-Oriented Software Engineering) yang bekerja pada perusahaan Objectory

Rasional [20].

Sebagai penggagas metode-metode tersebut mereka bertiga berinisiatif untuk menciptakan bahasa pemodelan terpadu sehingga pada tahun 1996 mereka berhasil merilis UML versi 0.9 dan 0.91 melalui Request for Proposal (RFP) yang dikeluarkan oleh OMG [20]. Saat ini sebagian besar dari para perancang sistem informasi dalam menggambarkan sebuah sistem informasi memanfaatkan UML diagram sebagai tujuan utama yang membantu tim proyek berkomunikasi, mengeksplorasi potensi dalam desain, serta memvalidasi desain arsitektur perangkat lunak atau pembuat suatu program [20].

Secara filosofis UML diilhami sebagai konsep yang telah ada yaitu konsep pemodelan Object Oriented sebab konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh obyek yang dapat digambarkan atau dinotasikan ke dalam simbol-simbol yang cukup spesifik. Ada 4 (empat) prinsip dasar dari pemrograman berorientasi kepada obyek yang menjadi landasan kemunculan UML, yaitu abstraksi, enkapsulasi, modularitas dan hirarki. Berikut penjelasan dari masing-masing landasan secara singkat [20]. Abstraksi memfokuskan perhatian kepada karakteristik obyek yang paling penting serta paling dominan yang bisa digunakan untuk membedakan obyek tersebut daripada obyek lainnya.

Enkapsulasi menyembunyikan banyak hal yang terdapat didalam obyek yang tidak perlu diketahui obyek lainnya. Dalam praktek pemrograman, enkapsulasi diwujudkan dengan cara membuat suatu kelas interface yang akan dipanggil oleh obyek lainnya, sementara didalam obyek yang dipanggil terdapat kelas lain yang mengimplementasikan apa yang terdapat dalam kelas interface. Modularitas berfungsi membagi sistem yang rumit menjadi bagian-bagian yang lebih kecil yang dapat mempermudah developer dalam memahami dan mengelola obyek tersebut.

Hirarki berhubungan dengan abstraksi dan modularitas, sebagai pembagian berdasarkan urutan dan pengelompokkan tertentu. Contohnya untuk menentukan obyek mana yang berada di kelompok yang sama, obyek mana yang merupakan suatu komponen dari obyek yang memiliki hirarki lebih tinggi. Dimana semakin rendah hirarki obyek maka semakin jauh abstraksi dilakukan terhadap suatu obyek. Tujuan Pemanfaatan UML Menurut (Sugrue J.

2009) [20] Berikut tujuan utama dalam desain UML yaitu : Menyediakan kepada pengguna (analisis dan desain sistem) suatu bahasa dalam pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan serta melakukan pertukaran model data yang bermakna. Menyediakan spesialisasi mekanisme untuk memperluas konsep

inti. Karena mengadopsi bahasa pemodelan visual dalam proses pembangunannya sebuah UML bersifat independen terhadap bahasa pemrograman tertentu. Memberikan dasar formal untuk pemahaman dalam bahasa pemodelan.

Mendorong pertumbuhan pasar dalam penggunaan alat desain sistem yang berorientasi objek (OO) Mendukung terhadap konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola serta komponen terhadap suatu sistem Memiliki sebuah integrasi praktik terbaik. Use Case Diagram Use case diagram menggambarkan fungsionalitas yang diharapkan dari suatu sistem. Use case menekankan "apa" yang diperbuat sistem, dan bukan "bagaimana". Sebuah use case dapat merepresentasikan sebuah interaksi antara aktor dengan sistem. Use case merupakan suatu pekerjaan tertentu, misalnya adalah login ke sistem, meng-create sebuah daftar belanja, dan sebagainya.

Seorang aktor yaitu sebuah entitas manusia ataupun mesin yang berinteraksi dengan sistem untuk dapat melakukan pekerjaan-pekerjaan tertentu [21]. Berikut ini adalah simbol simbol yang terdapat di dalam use case diagram: Tabel 2.6 Simbol-Simbol Use Case Diagram No \_Simbol\_Nama\_Deskripsi \_\_1 \_\_ Use case \_Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal fawal rase nama use case. \_\_Tabel 2.6

Simbol-Simbol Use Case Diagram (lanjutan) No \_Simbol\_Nama\_Deskripsi \_\_2 \_/ \_Aktor / Actor \_Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor \_\_3 \_/ \_Asosiasi / Association \_Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.

\_\_4 \_ <<extend>>/ \_Ekstensi / extend \_Relasi case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, misal \_ Arah panah mengarah pada use case yang ditambahkan; biasanya use case yang menjadi extendnya merupakan jenis yang sama \_\_ Tabel 2.6

Simbol-Simbol Use Case Diagram (lanjutan) No \_Simbol\_Nama\_Deskripsi \_\_5 \_/ \_Generalisasi / Generalization \_Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya: \_ Arah panah mengarah pada use case yang menjadi \_\_6

<<extend>>/ / <<uses>> \_Menggunakan / include / uses \_Ada dua sudut pandang yang cukup besar mengenai include di use case: Include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan, misal pada kasus berikut: \_ Include berarti use case yang tambahan akan selalu melakukan pengecekan apakah use case yang ditambahkan telah dijalankan sebelum use case tambahan dijalankan, misal pada kasus berikut: \_\_ (Rosa & Shalahuddin 2018 : 156) Activity Diagram Activity diagram menggambarkan berbagai alir dari aktivitas dalam sistem yang sedang dirancang, bagaimana alir masing-masing berawal, decision yang mungkin terjadi, serta bagaimana mereka berakhir.

Activity diagram merupakan state diagram khusus, dimana sebahagian besar state adalah action dan sebahagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karenanya activity diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar sub sistem) secara eksak, tetapi lebih menggambarkan jalur-jalur dan proses-proses aktivitas dari level atas secara general/umum [21]. Berikut merupakan simbol-simbol yang terdapat pada Activity Diagram : Tabel 2.7

Simbol-Simbol Activity Diagram No \_Simbol \_Nama \_Deskripsi \_\_1 / \_Status Awal \_Status awal dari aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal \_\_2 / \_Aktivitas \_Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. \_\_3 \_\_ \_Percabangan / decision \_Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu. \_\_4 / \_Penggabungan / join \_Asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu. \_\_5 / \_Status akhir \_Status akhir yang dilakukan sistem. sebuah diagram aktivitas memiliki sebuah status akhir \_\_6 \_\_ \_Swimlane \_Memisahkan organisasi bisnis yang bertanggungjawab terhadap aktivitas yang terjadi.

\_\_ (Rosa & Shalahuddin , 2018 : 162) 2.5.1.5 Class Diagram Class yaitu sebuah spesifikasi yang jika diinstansiasi akan menghasilkan suatu objek dan merupakan inti dari sebuah pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Class diagram menampilkan struktur dan deskripsi class, package serta objek beserta hubungan satu dengan yang lain seperti containment, pewarisan, asosiasi, dan lainnya.

Class diagram memiliki tiga area pokok, meliputi yaitu [21] : Nama (dan stereotipe) Atribut Metode Berikut merupakan simbol-simbol yang terdapat pada Class Diagram [22] : Tabel 2.8 Simbol-Simbol Class Diagram No \_Simbol \_Nama \_Deskripsi \_\_1 \_\_ \_Kelas \_Kelas dari struktur sistem. \_\_2 \_\_ \_Antarmuka / interface \_Sama dengan konsep

interface dalam pemrograman berorientasi objek \_\_3 \_\_ / \_Asosiasi / association \_Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity \_\_4 \_\_ / \_Asosiasi berarah / directed association \_Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas lain asosiasi biasanya juga disertai dengan multiplicity \_\_

Tabel 2.8

Simbol-Simbol Class Diagram (lanjutan) No\_Simbol\_Nama\_Deskripsi \_\_5 \_\_ / \_Generalisasi \_Relasi dari antarkelas dengan makna sebagai generalisasi-spesialisasi (umum – khusus) \_\_6 \_\_ \_Kebergantungan / dependency \_Relasi dari antarkelas yang menjelaskan makna sebuah kebergantungan antarkelas \_\_7 \_\_ \_Agregasi / aggregation \_Relasi antarkelas dengan makna semua bagian (whole-part) \_\_ (Rosa & Shalahuddin , 2018 : 146) Bagan Alir (Flowchart) flowchart yaitu penggambaran secara grafik dari urutan-urutan dan langkah-langkah prosedur dari suatu program. Flowchart menolong analisis dalam memecahkan masalah kedalam sebuah segmen-segmen yang lebih kecil dan menolong di dalam menganalisa alternatif-alternatif lain dalam pengoperasian.

Flowchart adalah diagram yang menyatakan aliran proses dengan mengadopsi anotasi dalam bidang-bidang geometri [23]. Ada 5 macam jenis flowchart yaitu sebagai berikut : Bagan Alir Sistem (System Flowchart) Flowchart Sistem yaitu sebuah bagan yang menggambarkan alur kerja atau apa saja yang sedang dilakukan di dalam sebuah sistem secara keseluruhan serta menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan Alir Dokumen (Document Flowchart) Flowchart Dokumen (Paperwork) menelusuri alur data yang ditulis melalui sistem.

Kegunaan utamanya yaitu untuk menelusuri alur form serta laporan sistem dari satu bagian ke bagian yang lain baik itu bagaimana alur form dan laporan diproses, dicatat dan juga disimpan. Bagan Alir Skematik (Schematic Flowchart) Flowchart skematik serupa dengan Flowchart Sistem yang menunjukkan sebuah sistem atau prosedur. Flowchart Skematik ini bukan hanya menggunakan suatu simbol-simbol Flowchart standar, tetapi juga menggunakan gambar-gambar yang ada di dalam komputer, peripheral, form-form ataupun di dalam peralatan lain yang digunakan dalam sistem.

Bagan Alir Program (Program Flowchart) Flowchart Program dilahirkan dari Flowchart Sistem. Flowchart Program yaitu sebuah keterangan yang lebih rinci tentang bagaimana setiap langkah program ataupun prosedur sesungguhnya dilaksanakan. Flowchart ini menggambarkan setiap langkah program atau prosedur di dalam urutan yang tepat saat terjadi. Bagan Alir Proses (Process Flowchart) Flowchart Proses adalah teknik penggambaran rekayasa industrial yang dapat memecah serta menganalisis langkah-langkah selanjutnya di dalam suatu prosedur atau sistem.

Berikut ini merupakan simbol-simbol yang terdapat pada flowchart yaitu: Tabel 2.9 Simbol-simbol flowchart No\_Nama\_Simbol\_Fungsi Flowchart \_\_1\_Start/End \_\_yaitu terminal atau start dan stop atau end pada suatu program untuk memulai dan mengakhiri suatu program. \_\_ Tabel 2.9 Simbol-simbol flowchart (lanjutan) No\_Nama\_Simbol\_Fungsi Flowchart \_\_2\_Proses \_\_yaitu suatu proses dimana sebuah data diolah suatu computer \_\_3\_Sub Program \_\_Yaitu sebuah pemulaan pada sub program/proses menjalankan sub program.

\_\_4\_Arrow \_\_yaitu suatu anak panah yang menunjukkan arah aliran dari algoritma. \_\_5\_Input atau Output \_\_yaitu untuk memproses masukan atau pengeluaran data \_\_6\_Simbol Connector \_\_yaitu simbol untuk keluar/masuk prosedur atau proses dalam lembar/halaman yang sama. \_\_7\_Simbol off-time Connector \_\_yaitu simbol untuk keluar/masuk prosedur atau proses dalam lembar/halaman yang lain.

\_\_8\_Document \_\_yaitu menyatakan simbol untuk data berbentuk kertas maupun suatu informasi \_\_9\_Decision \_\_Adalah proses pengambilan untuk memilih satu diantara alternative yang ada. \_\_ (Sumber: Taupik, Irfan, & Nurpianti 2013 : 38) Aplikasi Pendukung Penelitian Beberapa tools atau aplikasi yang digunakan dalam pengembangan sistem berbasis Web untuk memudahkan dalam pembuatan aplikasi beberapa diantaranya adalah sebagai berikut: XAMPP Menurut Arif dalam Sujana Cristian (2011) [24], "XAMPP merupakan aplikasi yang mengintegrasikan beberapa aplikasi utama web di dalamnya. Dalam XAMPP terdapat instalasi modul PHP, Mysql, webserver Apache".

/ (Sumber: Fribel Sipakkar 2017) Gambar 2.2 Tampilan XAMPP Sublime Text Menurut Eric Haughee dalam Sujana Cristian (2013) [24], "Bahwa Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan di berbagai platform operating system dengan menggunakan teknologi Python API". / (Sumber: Fribel Sipakkar 2017) Gambar 2.3 Tampilan Sublime Text CodeIgniter CodeIgniter merupakan framework PHP yang diklaim memiliki eksekusi tercepat bila dibandingkan dengan framework yang lainnya. CodeIgniter memiliki konsep yang dinamakan MVC.

MVC adalah kepanjangan dari Model, View, Controller, alur kerja framework khususnya disini CodeIgniter Adapun sistem kerja CodeIgniter memiliki beberapa tahapan yang harus dilewati. Alur kerjanya dapat dilihat sebagai berikut [25] : / (Sumber: Muh Rais, & M.Apriadi Hadi Sirad 2019 : 28) Gambar 2.4 Alur Kerja CodeIgniter Web Browser Web browser adalah perangkat lunak yang berfungsi untuk menerima dan menampilkan sumber informasi di internet.

Suatu sumber informasi diidentifikasi dengan pengidentifikasian sumber yang bisa

berupa halaman web, video, gambar, ataupun jenis konten lainnya. Contoh web browser adalah Mozilla Firefox , Google Chrome, Microsoft Edge , Opera, dan lain sebagainya [26] / (Sumber: Fribel Sipakkar 2017) Gambar 2.5 Tampilan Halaman Web Browser Goggle Chrome

DAFTAR PUSTAKA [1] S. H. Sihombing, "Penyakit Talasemia Dengan Menggunakan," *pelita Inform.*, vol. 18, pp. 107–112, 2019. [2] P. S. Hasibuan and M. I. Batubara, "Penerapan Metode Dempster Shafer Dalam Mendiagnosa Penyakit Faringitis," *J. Media Inform. Budidarma*, vol. 3, no.

1, pp. 59–64, 2019. [3] P. S. Ramadhan, "Aplikasi Diagnosa Granulomatous Dermatis Menggunakan Certainty Factor," *J. Media Inform. Budidarma*, vol. 3, no. 2, pp. 78–83, 2019. [4] E. F. Nasution, N. A. Hasibuan, and N. Silalahi, "Rancangan Aplikasi Sistem Pakar Diagnosa Penyakit Pankreatitis Dengan Metode Certainty Factor," *Ilm. Inti*, vol. 13, no. September, pp. 270–273, 2018. [5] H. N. Hidayat, Thofik and Nasution, "Sistem Pakar Untuk Mendiagnosa Penyakit Lupus Dengan Menggunakan Metode Certainty Factor," *J. Educ. Dev.*, vol. 7, no. 3, pp. 114--114, 2019. [6] D. FARROW, "Laringitis," *rekursif*, vol. 151, no. 1, pp. 10–17, 2015. [7] M.

sauqi Ahmad, "Penyakit Laringitis," *Rekursif*, vol. 2, no. 5, p. 255, 2015. [8] P. E. Pada, S. Reservasi, P. Kulit, H. Azizah, and S. D. Putra, "PENERAPAN E-HEALTH PADA SISTEM RESERVASI PERAWATAN KULIT WAJAH," *JISAMAR*, vol. 3, no. 2, pp. 121–133, 2019. [9] Haryanto, N. Riyani, J. T. Napitupulu, and W. Manurian, "Pemodelan E-Health Pada Klinik Untuk Proses Konsultasi Dokter dan Penjualan Obat," *Sinergi*, vol. 3, no. 2, pp. 121–130, 2016. [10] H. Fahmi, "Sistem Pakar Mendiagnosa Penyakit Mata Katarak Dengan Metode Certainty Factor Berbasis Web," *Matics*, vol. 11, no. 1, p. 27, 2019. [11] R. R. Fanny, N. A. Hasibuan, and E.

Buulolo, "RENALIS MENGGUNAKAN METODE CERTAINTY FACTOR DENGAN PENULUSURAN FORWARD CHAINING," vol. 1, no. 1, pp. 13–16, 2017. [12] Y. Ervinaeni, A. S. Hidayat, and E. Riana, "Sistem Pakar Diagnosa Gangguan Hiperaktif Pada Anak Dengan Metode Naive Bayes Berbasis Web," *J. Media Inform. Budidarma*, vol. 3, no. 2, pp. 90–104, 2019. [13] A. Fadli, "Sistem Pakar Dasar," *ILKOM*, no. 2, pp. 1–8, 2010. [14] Hibbert, "Konsep Sistem Pakar," *ILMU Komput.*, vol. 2, no. August, p. 32, 2016. [15] P. S. Ramadhan and Fatimah, "Sistem E-Healthcare Untuk Mendiagnosa Penyakit Inflamasi Dermatitis Imun Anak Dengan Menggunakan Metode Certainty Factor," *SENSASI*, vol.

1, no. 1, pp. 251–256, 2018. [16] N. A. Hasibuan, H. Sunandar, S. Alas, and S. Suginam, "Sistem Pakar Mendiagnosa Penyakit Kaki Gajah Menggunakan Metode Certainty Factor," *Jurasik (Jurnal Ris. Sist. Inf. dan Tek. Inform.)*, vol. 2, no. 1, p. 29, 2017. [17] Z. Arifin, "Naskah publikasi sistem pakar diagnosa penyakit pada balita menggunakan metode certainty factor," pp. 1–10, 2019. [18] J. S. Simatupang, E. Panggabean, and M. Kom, "Sistem Pakar Untuk Mendiagnosa Kerusakan Mesin Fotocopy Canon IR 6000 Menggunakan Metode Certainty Factor," vol. 1, pp. 61–66, 2019. [19] F. MZI, *Informatika*, "Pengenalan Pemodelan Sistem," *Acknowledgements*, vol. 11, no. 1, pp. 1–116, 2016.

[20] Havaluddin, "Memahami Penggunaan UML ( Unified Modelling Language )," Memahami Pengguna. UML (Unified Model. Lang., vol. 6, no. 1, pp. 1–15, 2011. [21] S. Dharwiyanti and R. S. Wahono, "Pengantar Unified Modeling Language (UML)," IlmuKomputer.com, vol. 11, no. 1, pp. 1–13, 2003. [22] M. S. Rosa A.S, REKAYASA PERANGKAT LUNAK, Revisi. Bandung: Informatika, 2018. [23] I. A. Ridlo, "Panduan pembuatan flowchart," Fak. Kesehat. Masy., vol. 11, no. 1, pp. 1–27, 2017. [24] Yanuardi and A. A. Permana, "Rancang Bangun Sistem Informasi Keuangan Pada Pt. Secret Discoveries Travel and Leisure Berbasis Web," J. Tek. Inform., vol. 1, no. 11, pp. 1–7, 2018. [25] M. R. Asiz and M. A.

Hadi Sirad, "Inventory Information System of Goods Using Codeigniter Framework," Patria Artha Technol. J., vol. 3, no. 1, pp. 23–30, 2019. [26] A. Saputra and Y. Astuti, "Analisis Pengaruh Struktur Html Terhadap," J. Mantik Penusa, vol. 2, no. 2, pp. 34–39, 2018.

#### INTERNET SOURCES:

-----

<1% - <https://khaidirmuhaj.blogspot.com/2010/06/>  
<1% - <https://www.scribd.com/doc/245055824/LARINGITIS>  
<1% - <https://free-medical.blogspot.com/2007/08/obstruksi-saluran-nafas-atas.html>  
<1% - <https://kristynilansari91.blogspot.com/2012/10/tinjauan-teori-rhinitis.html>  
<1% - <https://muhyasir.wordpress.com/2012/03/28/e-health-di-indonesia-sudah-siap/>  
<1% - <http://journal.stmikjayakarta.ac.id/index.php/jisamar/article/download/97/86/>  
<1% -  
<https://riskyinform.blogspot.com/2016/11/defenisi-e-government-e-business-e.html>  
<1% -  
<https://id.scribd.com/doc/259777149/E-health-Kelompok-6-Sistem-Informasi-Kesehatan-Kelas-A>  
<1% - <http://jti.respati.ac.id/index.php/jurnaljti/article/download/83/74>  
<1% - <https://kaerrika.blogspot.com/2016/01/makalah-sistem-pakar.html>  
<1% - [https://id.wikipedia.org/wiki/Sistem\\_pakar](https://id.wikipedia.org/wiki/Sistem_pakar)  
<1% - <https://kalosoedimampir.blogspot.com/2011/03/penulisan-ilmiah.html>  
<1% - <https://irhamworld.blogspot.com/2017/>  
<1% -  
<http://repository.usu.ac.id/bitstream/handle/123456789/27986/Chapter%20II.pdf?sequence=4&isAllowed=y>  
<1% -  
<http://repository.usu.ac.id/bitstream/handle/123456789/29633/Chapter%20II.pdf;sequence=4>

<1% - [http://eprints.dinus.ac.id/18267/10/bab2\\_17832.pdf](http://eprints.dinus.ac.id/18267/10/bab2_17832.pdf)  
<1% -  
[http://www.wayanfm.lecture.ub.ac.id/files/2014/04/FP\\_SPK\\_F\\_D-2013-2014-Ganjil-.pdf](http://www.wayanfm.lecture.ub.ac.id/files/2014/04/FP_SPK_F_D-2013-2014-Ganjil-.pdf)  
1% -  
<https://www.slideshare.net/AryPrasetyo10/ary-phapzi-ali-desain-sistem-pakar-ut-2017>  
<1% - <https://dosenit.com/jaringan-komputer/konsep-jaringan/jaringan-vlan>  
<1% - <https://unibsi.blogspot.com/2016/08/soal-dan-jawaban-her-sistem-pakar.html>  
<1% -  
[https://repository.bsi.ac.id/index.php/unduh/item/132104/FILE\\_16-BAB-II---LANDASAN-TEORI.pdf](https://repository.bsi.ac.id/index.php/unduh/item/132104/FILE_16-BAB-II---LANDASAN-TEORI.pdf)  
<1% - <http://tunasbangsa.ac.id/ejurnal/index.php/jurasik/article/download/16/13>  
<1% -  
<http://ojs.amikom.ac.id/index.php/semnasteknomedia/article/download/1178/1127>  
<1% - <http://ejournal.uin-malang.ac.id/index.php/saintek/article/download/7673/7386>  
<1% - <https://core.ac.uk/download/pdf/35380749.pdf>  
<1% - <http://jip.polinema.ac.id/ojs3/index.php/jip/article/view/82/81>  
<1% - <https://www.scribd.com/document/341026275/KAPITA-SELEKTA-3>  
<1% -  
<https://www.scribd.com/document/364938216/Nur-Aini-Et-All-Sistem-Pakar-Pendiagnosa-Penyakit-Tuberkulosis>  
<1% - <https://jurnal.ugm.ac.id/mgi/article/download/27801/18792>  
<1% - <http://jurnal.dinamika.ac.id/index.php/jsika/article/download/818/496>  
<1% - <http://e-journals.unmul.ac.id/index.php/SAKTI/article/download/270/pdf>  
<1% - <http://e-jurnal.pelitanusantara.ac.id/index.php/mantik/issue/download/8/3>  
<1% - [https://eprints.sinus.ac.id/92/2/002C2016SSI\\_13.4.10032\\_BAB\\_II.pdf](https://eprints.sinus.ac.id/92/2/002C2016SSI_13.4.10032_BAB_II.pdf)  
<1% - <https://computeraddict13.blogspot.com/2016/>  
1% -  
<https://id.scribd.com/doc/207156193/01-Jurnal-Informatika-Mulawarman-Feb-2011>  
<1% - <http://e-journals.unmul.ac.id/index.php/JIM/article/download/16/pdf>  
<1% -  
<https://anikhanda23.blogspot.com/2015/11/penerapan-google-translate-pada.html>  
<1% -  
<http://amuslim.staff.gunadarma.ac.id/Downloads/files/8219/materi+UML+dan+use+case.pdf>  
<1% - <https://felixsantosa.blogspot.com/2011/06/>  
<1% -  
<http://eprints.binadarma.ac.id/303/1/PROPOSAL%20ANALISIS%20DAN%20PERANCANGAN%20BASIS%20DATA%20AKADEMIK.pdf>  
<1% - <https://repository.bsi.ac.id/index.php/unduh/item/94826/Skripsi-Ok.pdf>  
<1% -

<http://download.garuda.ristekdikti.go.id/article.php?article=940598&val=14580&title=Perancangan%20dan%20Implementasi%20Sistem%20Informasi%20Layanan%20Kesehatan%20Masyarakat>

<1% - <http://digilib.unila.ac.id/14982/16/BAB%20II.pdf>

<1% - <https://plj.ac.id/ojs/index.php/jrict/article/download/247/205>

<1% - <https://riotugasmelit.blogspot.com/>

1% - <https://core.ac.uk/download/pdf/35319291.pdf>

<1% - <https://id.scribd.com/doc/309826808/BAB-II>

<1% - <http://digilib.unila.ac.id/4254/102/BAB%20II.pdf>

<1% - <http://digilib.unila.ac.id/5493/16/16%20-%20Bab%202.pdf>

<1% - <https://elearning.ithb.ac.id/mod/resource/view.php?id=6813>

<1% -

[https://informaasi.blogspot.com/2015/09/skripsi-sistem-pendukung-keputusan\\_66.html](https://informaasi.blogspot.com/2015/09/skripsi-sistem-pendukung-keputusan_66.html)

<1% -

<https://azharkesatriawolawolawol.blogspot.com/2014/06/pengenalan-simbol-clas-diagram-pada-uml.html>

<1% - <http://eprints.radenfatah.ac.id/157/2/BAB%20II.pdf>

<1% - <http://anyflip.com/lczt/rtiu/basic>

<1% - <https://lizhaerlianty.blogspot.com/2013/10/makalah-flowchart.html>

<1% - <http://widuri.raharja.info/index.php?title=SI1431482890>

<1% -

[https://erpeeldua03.blogspot.com/2014/11/contoh-flowchart-atau-diagram-alir\\_13.html](https://erpeeldua03.blogspot.com/2014/11/contoh-flowchart-atau-diagram-alir_13.html)

<1% - <http://febriani.staff.gunadarma.ac.id/Downloads/files/5616/Flowchart.pdf>

<1% - <https://chusnulnh.blogspot.com/>

<1% - <https://www.pelajaran.co.id/2019/02/pengertian-flowchart.html>

<1% -

[https://rani24081992.blogspot.com/2012/07/babi-konsepdasar-bahasa-pascal-1\\_17.html](https://rani24081992.blogspot.com/2012/07/babi-konsepdasar-bahasa-pascal-1_17.html)

<1% -

<https://fairuzelsaid.wordpress.com/2010/01/13/analisis-sistem-informasi-pedoman-membuat-flowchart/>

<1% -

[https://www.researchgate.net/publication/332783526\\_INVENTORY\\_INFORMATION\\_SYSTEM\\_OF\\_GOODS\\_USING\\_CODEIGNITER\\_FRAMEWORK](https://www.researchgate.net/publication/332783526_INVENTORY_INFORMATION_SYSTEM_OF_GOODS_USING_CODEIGNITER_FRAMEWORK)

<1% - [https://id.wikipedia.org/wiki/Situs\\_web](https://id.wikipedia.org/wiki/Situs_web)