

# Rancang Bangun Aplikasi Pendeteksi Penyakit Hepatitis Berbasis Sistem Pakar Menggunakan Metode Prototyping

Yudi Irawan Chandra

Sistem Informasi STMIK Jakarta STI&K

Jl. BRI No.17 Radio Dalam Kebayoran Baru Jakarta Selatan 12140

yudi@jak-stik.ac.id

*Intisari*— Sistem pakar merupakan program komputer yang dapat meniru proses pemikiran dan pengetahuan pakar untuk menyelesaikan suatu masalah yang spesifik. Implementasi sistem pakar banyak digunakan untuk kepentingan komersial karena sistem pakar dipandang sebagai cara penyimpanan pengetahuan pakar dalam bidang tertentu ke dalam suatu program, sehingga dapat memberikan keputusan dan melakukan penalaran secara cerdas. Hal ini yang mendasari diperlukannya suatu sistem mengenai diagnosis penyakit selain untuk mengatasi kendala dan keterbatasan di dunia kesehatan. Sehingga diharapkan dengan adanya sistem pakar ini dapat meningkatkan kinerja pelayanan kesehatan, serta dapat mengurangi timbulnya bahaya yang disebabkan oleh gejala penyakit karena telah dapat dideteksi dengan lebih cepat. Diagnosis penyakit dengan menggunakan sistem pakar akan mencatat gejala – gejala dari pasien dan akan mendiagnosis jenis penyakitnya yang berbasis pada pengetahuan yang didapat dari seorang pakar, sehingga diharapkan dapat membantu dan mempermudah pihak – pihak terkait dalam melakukan diagnosis penyakit bagi pasien dan dapat membantu dalam memberikan pelayanan kesehatan terhadap masyarakat. Metoda Prototyping digunakan dalam langkah-langkah pembangunan aplikasi karena dapat menghemat waktu dan hasil perangkat lunak mudah disesuaikan dengan kebutuhan dan keinginan pengguna.

*Kata kunci*— Deteksi Penyakit, Hepatitis, Sistem Pakar, Prototyping, Android

*Abstract*— Expert systems are computer programs that can mimic the thought process and expert knowledge to solve a specific problem. The implementation of expert systems is widely used for commercial purposes because expert systems are seen as a way of storing expert knowledge in certain fields into a program, so that they can make decisions and reason intelligently. This is what underlies the need for a system of disease diagnosis in addition to overcoming obstacles and limitations in the world of health. So that it is expected that with this expert system can improve the performance of health services, and can reduce the emergence of hazards caused by symptoms of disease because it can be detected more quickly. Disease diagnosis using an expert system will record the symptoms of the patient and will diagnose the type of disease based on the knowledge gained from an expert, so that it is expected to help and facilitate the parties involved in diagnosing the disease for patients and can assist in providing services health for the community. The Prototyping method is used in the steps of application development because it can save time and software results easily adapted to the needs and desires of the user.

*Keywords*— Disease Detection, Hepatitis, Expert System, Prototyping, Android

## I. PENDAHULUAN

Perkembangan teknologi dan industri yang semakin cepat serta persaingan antar perusahaan yang semakin global, menuntut setiap perusahaan untuk mampu mengembangkan sistem pengambilan keputusan yang cepat, akurat, dan handal. Aktifitas pengambilan keputusan ini banyak melibatkan pakar. Proses pengambilan keputusan ini disamping memakan biaya yang cukup besar, juga tidak selalu mudah mendapatkan pakar pada saat dibutuhkan maupun kualifikasi kepakarannya.

Disamping itu, kesehatan merupakan hal yang begitu penting bagi masyarakat. Pengetahuan masyarakat yang minim tentang kesehatan dan masih rendahnya untuk memeriksakan diri ke dokter pada saat terkena gejala – gejala penyakit, sering membuat penyakit yang diderita semakin bertambah parah karena kesalahan pada penanganan awal penyakit.

Suatu gejala penyakit dapat merupakan awal dari sebuah penyakit yang dapat membahayakan pasien, tetapi pada kenyataannya gejala penyakit tersebut terkadang dianggap remeh oleh pasien. Dengan adanya kemajuan ilmu pengetahuan dan teknologi saat ini bahaya yang ditimbulkan oleh suatu penyakit dapat diketahui dengan lebih cepat. Aplikasi diagnosis penyakit yang akan dibuat menggunakan sistem pakar.

Dari latar belakang yang dipaparkan, maka permasalahan yang diamati adalah bagaimana cara menganalisis gejala – gejala penyakit untuk mengetahui penyakit yang diderita. Batasan masalah penelitian ini adalah sebagai berikut :

- Mendiagnosis suatu penyakit berdasarkan gejala – gejala yang diderita.
- Penyakit yang akan didiagnosis adalah penyakit Hepatitis A, Hepatitis B, Hepatitis C.
- Membuat rule ± rule untuk diagnosis penyakit.

Pembuatan sistem pakar ini dirancang untuk membantu masyarakat umum dalam mendiagnosis penyakit hepatitis agar dapat mengetahui perkiraan penyakit yang diderita, dan agar dapat diberikan pengobatan lebih dini, serta dapat memenuhi kebutuhan masyarakat akan informasi penyakit yang disusun berdasarkan hasil penelitian dan perkembangan.

## II. METODOLOGI PENELITIAN

Skema bagan alir dalam tahapan penelitian tentang pembuatan aplikasi ini dapat dilihat pada gambar berikut :



Gambar 1. Skema Metode Penelitian

Adapun teknik yang dilakukan untuk pengumpulan data adalah sebagai berikut :

### 1. Metode observasi lapangan

Metode pengumpulan data dengan cara mengamati secara langsung tentang kegiatan, keadaan umum, dan kejadian-kejadian yang ada dalam objek penelitian dengan pencatatan secara otomatis, selain itu metode ini juga dapat dilakukan dengan cara mengadakan tanya jawab langsung dengan masalah yang diteliti bersama narasumber yang dapat dipercaya.

### 2. Metode Kajian Pustaka

Metode pengumpulan data yang dapat diperoleh melalui perpustakaan atau nara sumber buku lain untuk

memperoleh data tambahan yang berhubungan dengan penelitian

## Teknik Prototype

Metode Prototype merupakan satu metode dalam pengembangan perangkat lunak, metode ini merupakan suatu paradigma baru dalam pembuatan / pengembangan perangkat lunak. Metode ini adalah evolusi dalam dunia pengembangan / pembuatan perangkat lunak, metode ini juga merovolusi metode pengembangan / pembuatan perangkat lunak yang lama, yaitu sistem sekuensial yang biasa dikenal dengan nama Metode Waterfall. Dalam metode prototype/prototyping, perangkat lunak yang dihasilkan kemudian dipresentasikan kepada klien, dan klien tersebut diberikan kesempatan untuk memberikan masukan dan kritikan, sehingga perangkat lunak / software yang dihasilkan sesuai dengan kebutuhan dan keinginan pelanggan. Perubahan perangkat lunak dapat dilakukan berkali-kali hingga dicapai kesepakatan bentuk dari software yang akan dikembangkan.

Seringkali seorang customer sulit menentukan input yang lebih terinci, proses yang diinginkan dan output yang diharapkan. Tentu saja ini menyebabkan developer tidak yakin dengan efisiensi algoritma yang dibuatnya, sulit menyesuaikan sistem operasi, serta interaksi manusia dan mesin yang harus diambil. Dalam hal seperti ini, pendekatan prototype untuk software engineering merupakan langkah yang terbaik. *Prototype sebenarnya adalah suatu proses yang memungkinkan developer membuat sebuah model software.*

Ada 2 bentuk dari model ini, yaitu :

### 1. Paper Prototype

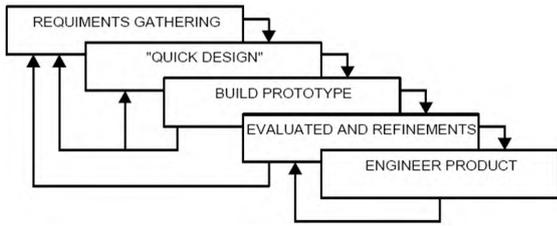
Menggambarkan interaksi manusia dan mesin dalam sebuah bentuk yang memungkinkan user mengerti bagaimana interaksi itu terjadi.

### 2. Working Prototype

Adalah prototype yang mengimplementasikan beberapa bagian dari fungsi software yang diinginkan seperti pada pendekatan pengembangan software. Model ini dimulai dengan :

- Pengumpulan kebutuhan developer dan customer
- Menentukan semua tujuan software
- Mengidentifikasi kebutuhan-kebutuhan yang diketahui

Hasil dari pengumpulan kebutuhan diteruskan pada *Quick Design*. Quick Design ini memfokuskan pada representasi aspek-aspek software yang dapat dilihat oleh user, misalnya format input dan output, selanjutnya dari desain cepat diteruskan pada pembentukan prototype (langkah ke 3). Prototype ini dievaluasi oleh customer / user dan digunakan untuk memperbaiki kebutuhan-kebutuhan software. Proses iterasi terjadi agar prototype yang dihasilkan memenuhi kebutuhan customer, juga pada saat yang sama developer mengerti lebih baik tentang apa yang harus dikerjakan.



Gambar 2. Metode Prototype

Masalah yang dihadapi oleh prototyping paradigma ini adalah :

- a. Customer hanya melihat pada apa yang dihasilkan oleh software, tidak peduli pada hal-hal yang berhubungan dengan kualitas software dan pemeliharaan jangka panjang.
- b. Developer seringkali menyetujui apa yang diterangkan oleh customer agar prototype dapat dihasilkan dengan cepat. Akibatnya timbul pemilihan sistem operasi / bahasa pemrograman yang tidak tepat.

Keunggulan prototyping adalah :

1. Adanya komunikasi yang baik antara pengembang dan pelanggan.
2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.
3. Pelanggan berperan aktif dalam pengembangan sistem.
4. Lebih menghemat waktu dalam pengembangan sistem.
5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya

Sedangkan kelemahan prototyping adalah :

- Pelanggan tidak melihat bahwa perangkat lunak belum mencerminkan kualitas perangkat lunak secara keseluruhan dan belum memikirkan peneliharaan dalam jangka waktu yang lama.
- Pengembang biasanya ingin cepat menyelesaikan proyek sehingga menggunakan algoritma dan bahasa pemrograman sederhana.
- Hubungan pelanggan dengan komputer mungkin tidak menggambarkan teknik perancangan yang baik.

III. HASIL DAN PEMBAHASAN

**Perancangan Data**

Dalam perancangan data, akan dijelaskan bagaimana data – data yang terdapat dalam sistem sesuai dengan fungsinya sebagai data input ataupun data output sistem.

Tabel 3.1 Gejala – Gejala Penyakit Hepatitis

Gejala	Hepatitis						
	A	B	C	D	E	F	G
Gejala mirip flu	*	*					
Rasa lelah	*	*	*		*		
Demam	*	*	*		*		
Diare	*						
Mual	*	*	*				
Nyeri perut	*						
Mata kuning	*	*					
Hilang nafsu makan (anoreksia)	*		*		*		
Lemah		*					
Lesu	*	*					
Sakit otot / pegal – pegal pada otot (Mialgia)	*	*	*				
Demam ringan		*					
Kurang nafsu makan		*					
Kulit kuning		*					
Air kencing berwarna gelap	*	*	*				
Muntah	*	*	*				
Sakit perut		*	*		*		
Sakit kepala		*					
Pegel linu					*		
Malaise	*						
Nausea	*						
Pusing	*	*					
Air kencing kemerahan	*						
Bola mata bagian putih menjadi kekuningan	*						
Nyeri pada sendi (Arthralgia)	*	*					
Rasa tidak enak pada tenggorokan	*						
Munculnya ikterus selama beberapa hari	*						
Insidious		*					
Rash		*					
Jaundice		*	*				
Menggigil			*				
Nyeri perut sebelah kanan			*				
Penurunan berat badan yang tidak diketahui sebabnya			*				
Gangguan abdominal tidak jelas			*				
Kembung			*				
Mencret			*				

**Decision Tree (Pohon Keputusan)**

Cara representasi pengetahuan yang tepat diperlukan untuk membuat suatu system pakar dapat melakukan penalaran yang baik. Perancangan basis pengetahuan (knowledge base) ini dimulai dengan membuat pohon keputusan (decision tree), pemilihan teknik representasi pengetahuan yang terbaik yang sesuai dengan masalah yang dihadapi. Berikut ini adalah pohon keputusan dari system pakar yang dirancang :



Gambar 3. Pohon Keputusan Gejala Penyakit

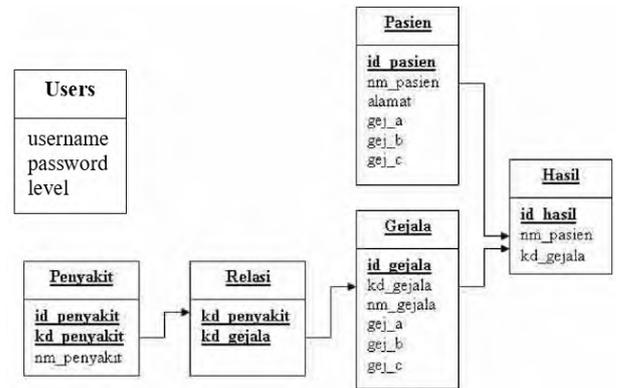
**IF – THEN Rule**

Setelah membuat decision tree, langkah berikutnya adalah dengan mengkonversikan decision tree tadi menjadi IF – THEN rule. Klausa diantara IF dan THEN, berisi tentang premis dan value premis yang benar, sedangkan klausa sesudah THEN dan sesudah ELSE merupakan hasil kesimpulan dari rule tersebut.

Berikut contoh hasil konversi dari decision tree area keluhan pertama pada table 3.1 menjadi bentuk IF – THEN rule :

- Rule 1 :  
IF demam = Yes and Diare = Yes THEN  
Kesimpulan = Hepatitis A
- Rule 2  
IF demam = Yes and Diare = No and Lemah = Yes THEN  
Kesimpulan = Hepatitis B
- Rule 3  
IF demam = Yes and Diare = No and Nyeri Perut = No and Hilang Nafsu makan = Yes THEN  
Kesimpulan = Hepatitis C

Di bawah ini adalah skema relasi antar tabel yang digunakan dalam pembangunan aplikasi :

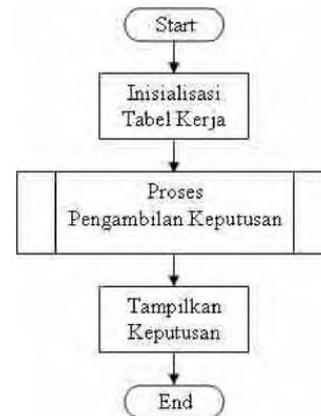


Gambar 4. Skema Tabel Database

**Perancangan Mesin Inferensi**

Metode penalaran yang digunakan dalam sistem adalah penalaran pelacakan maju (Forward Chaining) yaitu dimulai dari sekumpulan fakta – fakta tentang suatu gejala yang diberikan oleh pengguna sebagai masukan sistem, untuk kemudian dilakukan pelacakan sampai tujuan akhir berupa diagnosis kemungkinan penyakit hepatitis yang diderita dan nilai kepercayaannya.

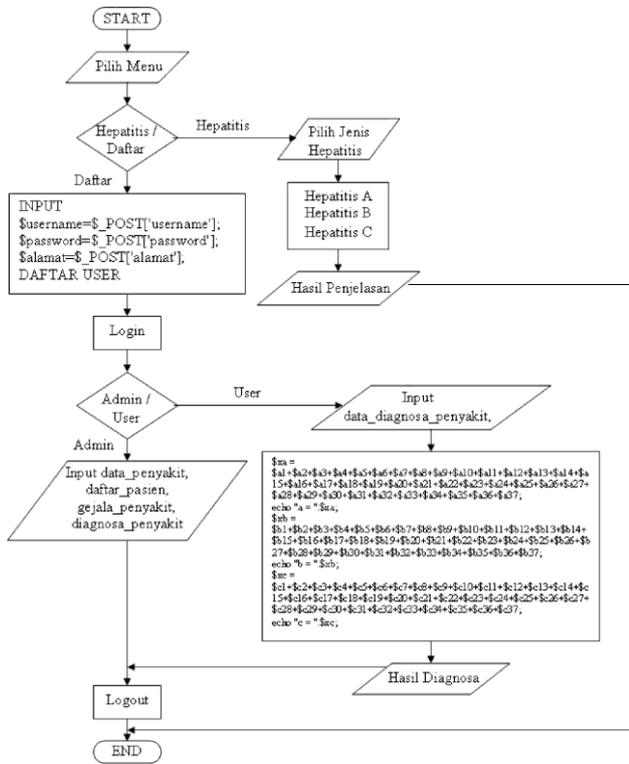
Secara umum, proses kerja inference engine dapat digambarkan menjadi seperti diagram alur pada gambar berikut :



Gambar 5. Perancangan Mesin Inferensi

**Perancangan Proses**

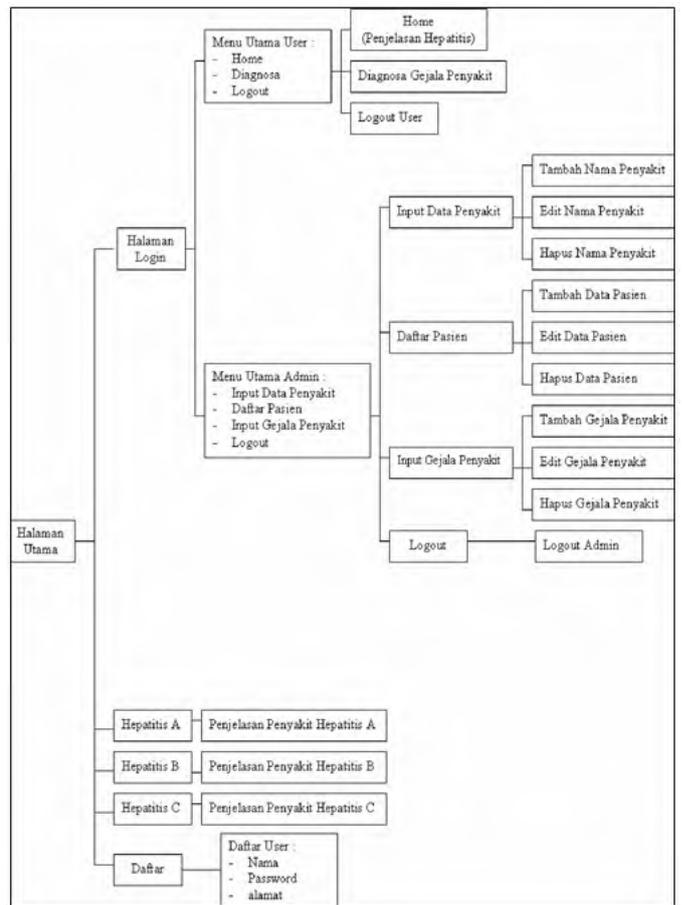
Perancangan proses akan menjelaskan bagaimana sistem bekerja untuk mengolah data input menjadi data output dengan fungsi – fungsi yang telah direncanakan. Dibawah ini adalah diagram alir dari sistem pakar diagnosis penyakit hepatitis :



Gambar 6. Perancangan Proses

**Struktur Menu**

Berikut ini adalah struktur menu dari aplikasi yang dibuat dalam mendeteksi penyakit hepatitis :



Gambar 7. Struktur Menu Aplikasi

**Implementasi Sistem**

Berikut ini adalah beberapa hasil tampilan dari aplikasi yang telah dibuat :



Gambar 8. Tampilan Aplikasi

**Spesifikasi Perangkat Keras dan Perangkat Lunak**

Di dalam penerapan rancangan aplikasi ini dibutuhkan perangkat keras berupa komputer untuk membuat dan menyajikan aplikasi tersebut. Adapun spesifikasi dari Komputer yang digunakan agar dapat menjalankan dan membuat aplikasinya adalah sebagai berikut :

1. Processor Intel Pentium Dual-Core (minimal Pentium 4)
2. Kapasitas RAM 2 GB minimal 1Gb.
3. Kapasitas harddisk yang dibutuhkan 4 Gb.

Untuk menerapkan rancangan aplikasi ini membutuhkan software pendukung dalam pembuatan aplikasi. Berikut software yang diperlukan:

1. JDK 7u1 for windows
2. Android SDK versi 18 for windows
3. Eclipse Indigo for windows 32
4. AVD (Android Virtual Device)
5. ADT 11.0.0 – Eclipse plug in

#### IV. KESIMPULAN

##### Simpulan

Setelah dilakukan pengujian dan analisa program, maka dapat diperoleh kesimpulan sebagai berikut :

1. Sistem pakar untuk mendiagnosa penyakit hepatitis adalah suatu aplikasi untuk mendiagnosa penyakit hepatitis berdasarkan pengetahuan dari para pakar.
2. Dengan menggunakan aplikasi Android maka pengguna dapat mendiagnosa kemungkinan penyakit hepatitis yang dideritanya sebelum mengambil tindakan lebih lanjut seperti konsultasi ke dokter atau tes laboratorium di rumah sakit.
3. Sistem user yang telah dibuat mampu melakukan proses penalaran data dengan metode forward chaining.
4. Sistem admin dapat melakukan proses penambahan, penghapusan dan update data untuk gejala penyakit pada penyakit tertentu atau untuk penyakit itu sendiri.
5. Walau cara kerja di dalam system pakar mungkin rumit, *user interface*-nya sangat memudahkan pemakai.
6. Sistem pakar mampu menyimpan representasi pengetahuan pakar berdasarkan nilai kebenaran MB dan nilai ketidakbenaran MD. Aplikasi sistem pakar ini sudah dapat menjelaskan definisi jenis penyakit, penyebab, dan pengobatannya.

##### Saran

Adapun saran – saran yang akan membantu untuk pengembangan perangkat lunak ini selanjutnya adalah sebagai berikut .:

1. Perlu diadakan penambahan data untuk jenis penyakit hepatitis beserta gejala-gejalanya sehingga informasi yang dimiliki akan semakin luas dan banyak.
2. Pada sistem admin, jika terjadi penambahan data yang lebih kompleks, admin haruslah mendesain tree dan database yang cukup fleksibel sehingga mampu memudahkan admin dalam melakukan manajemen sistem.
3. Tampilan dari aplikasi diagnosis penyakit hepatitis ini masih terbatas, sehingga untuk pengembangan selanjutnya tampilan dari aplikasi dapat dibuat lebih menarik lagi.
4. Adanya fasilitas untuk melakukan pengeditan data yang digunakan untuk penghitungan diagnosis.

5. Belum adanya pengelompokan gejala-gejala sejenis yang hanya boleh dipilih satu dari kelompok gejala tersebut. Akibatnya, jika user kurang teliti dalam memilih gejala, maka sistem akan memberikan kesimpulan yang kurang benar.

#### REFERENSI

- [1]. Britton, Carol (2001). *Object-Oriented Systems Development*. McGraw-Hill. hlm. 28–29, 269. ISBN 0-07-709544-8.
- [2]. Mc.,Leod, R. Jr., (2002), *System Development: A Project Management Approach*. New York: Leigh Publishing LLC
- [3]. Whitten, J.L. & Bentley, L.D., (2004), *System Analysis & Design Methods: Sixth Edition*. New York: Mc.Graw-Hill.
- [4]. Muhammad, Arhami., (2005), *Konsep dasar Sistem Pakar*, Penerbit Andi, Yogyakarta.
- [5]. Munawar, (2005), *Pemodelan Visual dengan UML*. Penerbit Graha Ilmu, Jakarta
- [6]. Marakas, G.M., (2006), *System Analysis Design: an Active Approach*. New York: Mc.Graw-Hill.
- [7]. Kendall, J.E. & Kendall, K.E., (2010), *Analisis dan Perancangan Sistem*. Jakarta: Indeks.
- [8]. Siswanto, (2010), *Algoritma dan Struktur Data Linear dengan Java*, Graha Ilmu, Edisi Pertama, Yogyakarta.
- [9]. Stephanus Hermawan S., (2011), *Mudah Membuat Aplikasi Android*, CV Andi Offset, Yogyakarta.
- [10]. Ivan Michael Siregar, (2011), *Membongkar Source Code Berbagai Aplikasi Android*, Gava Media, Cetakan I, Yogyakarta.
- [11]. Edy Winarto ST, M.Eng, Ali Zaki & SmitDev Community, (2012), *Hacking & Programming dengan Android SDK untuk Advance*, Elex Media Komputindo, Jakarta.
- [12]. Pressman, R.S., (2012), *Rekayasa Perangkat Lunak: Pendekatan Praktisi*. Penerbit Andi, Yogyakarta.
- [13]. dr. Wening Sari, M.Kes dkk, (2013), *Care Your self: Hepatitis*, Niaga Swadaya, Jakarta, ISBN: 9793927453, 9789793927459
- [14]. Srikandi Waluyo, (2013), *100 Questions & Answers: Hepatitis*, Elex Media Komputindo, Jakarta, ISBN : 6020402371, 9786020402376