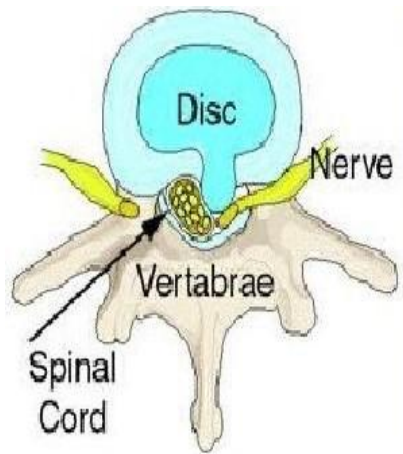


## BAB II

### TINJAUAN PUSTAKA

#### 2.1 *Herniated Nucleus Pulposus*

*Hernia Nucleus Pulposus* (HNP) adalah turunnya kandungan annulus fibrosus dari diskus intervertebralis lumbal pada spinal canal atau rupture annulus fibrosus dengan tekanan dari nucleus pulposus yang menyebabkan kompresi pada element saraf . Secara garis besar, HNP digambarkan sebagai berikut: (Herliana, 2017)



Gambar 2.1. *Hernia Nucleus Pulposus* (Sumber: Muttaqin, 2008 dikutip dari Herlina, 2017)

Penyakit *Hernia Nucleus Pulposus* biasanya terjadi pada usia produktif antara usia 35-55 tahun. Pada usia tersebut, paradigma yang ada di masyarakat bila terdiagnosa penyakit *Hernia Nucleus Pulposus* adalah harus melakukan tindakan operasi sebagai solusi menyembuhkannya. Padahal jika HNP di ketahui sejak dini, maka biasanya dengan melakukan istirahat, fisioterapi dan konsumsi obat yang dianjurkan oleh para ahli syaraf maka rasa sakit akan segera reda dan tindakan operasi tidak akan diperlukan.

### **2.1.2 Penyebab Penyakit *Herniated Nucleus Pulposus***

#### **2.2 Hal-hal yang dapat menyebabkan penyakit *Hernia Nucleus Pulposus* antara lain:**

- a. Aktivitas mengangkat benda yang cukup berat dengan posisi yang salah seperti menggunakan posisi membungkuk sebagai awalan untuk mengangkat benda yang cukup berat.
- b. Kebiasaan sikap duduk yang salah dalam waktu yang cukup lama. Hal ini bisa disebabkan karena profesi yang dijalani membutuhkan waktu yang lama dalam posisi duduk yang kurang nyaman untuk tulang belakang seperti membungkuk.
- c. Melakukan gerakan yang salah baik secara sengaja ataupun tidak yang menyebabkan tulang punggung mengalami penyempitan kebagian tulang bawah seperti mengalami trauma karena kecelakaan dengan posisi akhir dalam keadaan duduk atau membungkuk.
- d. Kelebihan berat badan (obesitas).

### **2.1.2 Pencegahan Penyakit *Herniated Nucleus Pulposus***

Beberapa hal yang dapat dilakukan untuk mencegah penyakit *Herniated Nucleus Pulposus* antara lain :

- a. Jika melakukan aktivitas duduk dalam waktu yang cukup lama, maka harus duduk dalam posisi yang benar dan sering lakukan peregangan sehingga tulang belakang kembali pada posisi yang baik.
- b. Memperhatikan berat badan agar tidak mengalami obesitas
- c. Melakukan olahraga serta mengkonsumsi obat bila perlu untuk menjaga kelenturan dan kekuatan otot tulang belakang.

### **2.1.3 Gejala Penyakit HNP**

Gejala yang sering timbul ialah sebagai berikut (Sumber : Andi, 2017) :

- a. Nyeri punggung, nyeri bokong , nyeri leher belakang, rasa kaku atau tertarik pada punggung bawah.

- b. Nyeri yang menjalar atau seperti rasa kesetrum yang dirasakan dari bokong menjalar ke daerah paha, betis bahkan sampai kaki
- c. Kelemahan anggota badan bawah atau tungkai bawah yang disertai dengan mengecilnya otot-otot tungkai bawah.

## 2.2 Metode *Case Based Reasoning*

*Case Based Reasoning* (CBR) adalah salah satu metode pemecahan masalah dalam mencari solusi berdasarkan suatu kasus yang lama, sistem yang dibuat akan melakukan pencarian terhadap solusi dari kasus lama yang mempunyai permasalahan yang sama dan sudah pernah terjadi sebelumnya. Ilustrasi sebagai contoh dari metode CBR ini ialah ketika seorang dokter yang sedang mendiagnosa pasiennya, dokter tersebut akan teringat dengan pasien lain yang pernah ia rawat sebelumnya karena mempunyai kemiripan gejala kasus penyakit dengan pasien yang saat ini ia tangani. Kemudian dokter menggunakan data hasil diagnose dan perawatan pasien yang pernah ia rawat sebelumnya untuk menentukan diagnose dan perawatan pasien saat ini. (Andreswari, 2016)

Pencarian dilakukan menggunakan nilai kemiripan (*similarity*) dengan rumus sebagai berikut. (Sri, 2017)

$$T = \frac{nX1}{N}$$

T= nilai kesamaan kasus

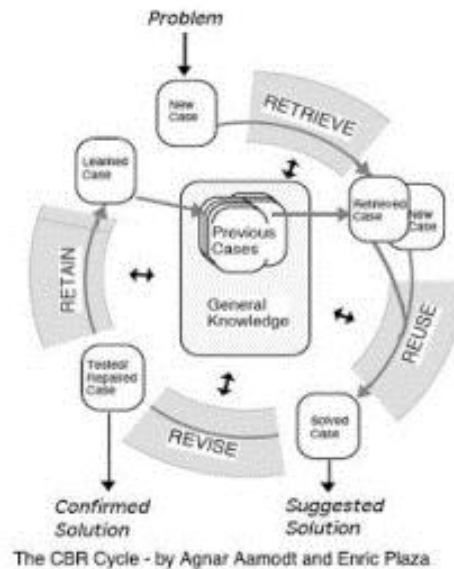
nX1=banyaknya kesamaan subjek

N= banyaknya elemen pada basis kasus

### 2.2.1 Arsitektur *Case Based Reasoning*

Selain CBR *cycle* gambaran dari struktur CBR dapat lebih memperjelas proses dari CBR secara detail seperti tampak pada keempat proses yang dilakukan pada CBR *cycle* yaitu *Retrieve, Reuse, Revise dan Retain..* Proses ini terdiri dari memilih informasi apa dari kasus yang akan disimpan, cara menyusun kasus agar

mudah untuk menentukan masalah yang mirip dan bagaimana mengintegrasikan kasus baru pada struktur memori. CBR *cycle* ditunjukkan pada gambar 2 di bawah ini. (Fitria, 2018)



Gambar 2.2 Cycle CBR(sumber: Fitria, 2018)

Pada saat terjadi permasalahan baru, pertama-tama sistem akan melakukan proses seperti di bawah ini :

- a. Proses *Retrieve* akan melakukan dua langkah pemrosesan yaitu pengenalan masalah dan pencarian persamaan masalah pada database. Setelah proses *Retrieve* selesai dilakukan, selanjutnya sistem akan melakukan proses *Reuse*.
- b. Di dalam proses *Reuse*, sistem akan menggunakan informasi permasalahan sebelumnya yang memiliki kesamaan untuk menyelesaikan permasalahan yang baru. Pada proses *Reuse* akan menyalin, menyeleksi, dan melengkapi informasi yang akan digunakan.
- c. Selanjutnya proses *Revise*, informasi tersebut akan dikalkulasi, dievaluasi, dan diperbaiki kembali untuk mengatasi kesalahan-kesalahan yang terjadi pada permasalahan baru.

- d. Proses terakhir ialah *Retain*, proses *Retain* akan mengintegrasikan, dan mengekstrak solusi yang baru. Selanjutnya solusi baru itu akan disimpan ke dalam *knowledgebase* untuk menyelesaikan permasalahan yang akan datang. Tentunya permasalahan yang akan diselesaikan adalah permasalahan yang memiliki kesamaan dengannya.

### 2.3 Sistem Pakar

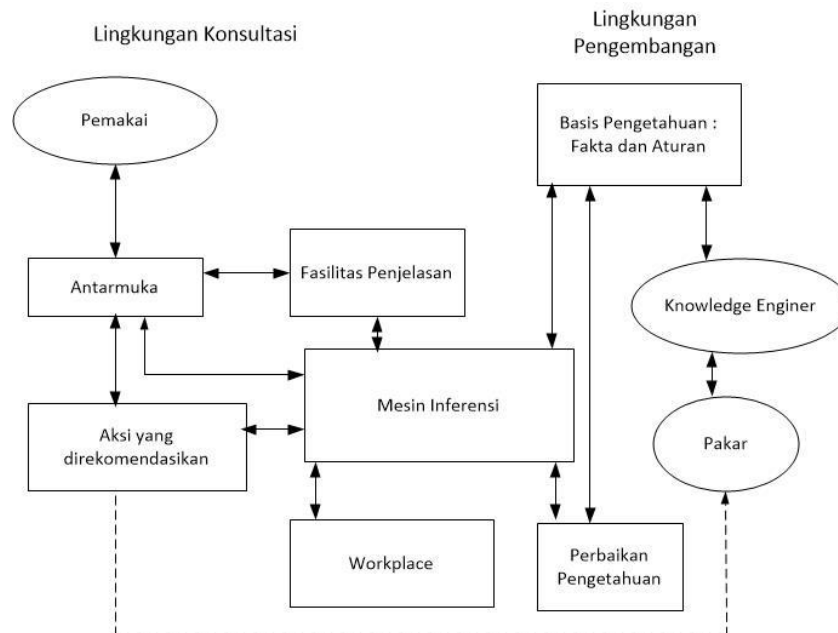
Menurut Arhami dari seorang pelopor awal teknologi sistem pakar Feigenbaum:1982, yang mendefinisikan sistem pakar sebagai suatu program komputer cerdas yang menggunakan *knowledge* (pengetahuan) dan prosedur inferensi untuk menyelesaikan masalah yang cukup sulit sehingga membutuhkan seorang yang ahli untuk menyelesaikannya. Sistem pakar (*expert system*) merupakan paket program komputer yang ditujukan sebagai penyedia nasihat dan sarana bantu dalam memecahkan masalah (Arhami, 2005). Ada beberapa keunggulan sistem pakar :

- a. Menghimpun data dalam jumlah besar.
- b. Menyimpan data tersebut untuk jangka waktu yang panjang
- c. Mengerjakan perhitungan secara cepat tanpa mencari kembali data yang tersimpan dengan kecepatan tinggi.

#### 2.3.1 Arsitektur Sistem Pakar

Sistem pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan untuk memasukkan pengetahuan pakar ke dalam sistem dan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar guna memperoleh pengetahuan pakar. Komponen-komponen sistem pakar seperti pada gambar 2.3 yaitu *user interface* (antar muka pengguna) yang merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi, basis pengetahuan yang mengandung pengetahuan untuk pemahaman dan penyelesaian masalah disusun atas dua elemen yaitu fakta dan aturan, akuisisi pengetahuan adalah akumulasi dan transformasi keahlian dalam menyelesaikan masalah dari

sumber pengetahuan ke dalam program komputer, mesin inferensi mengandungi mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah, *workplace* merupakan area dari sekumpulan memori kerja yang digunakan untuk merekam hasil-hasil antara dan kesimpulan yang dicapai, fasilitas penjelasan adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar, perbaikan pengetahuan, pakar memiliki kemampuan untuk menganalisa dan meningkatkan kinerjanya serta kemampuan untuk belajar dari kinerjanya (Arhami,2005).

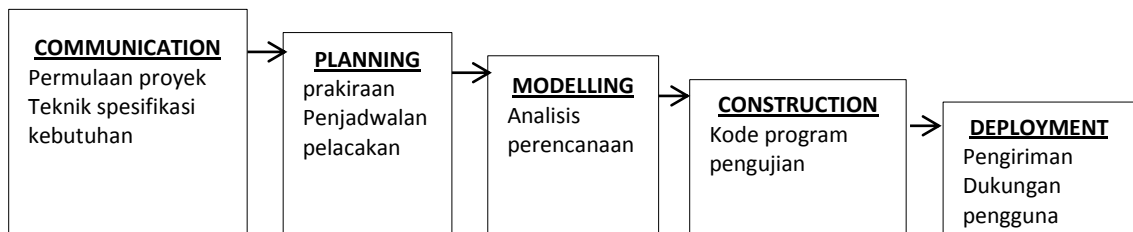


Gambar 2.3. Arsitektur sistem pakar (Sumber: Turban 1995 dikutip dari Arhami 2005)

#### 2.4 Metode Pengembangan Perangkat Lunak *Waterfall*

Model air terjun (*waterfall*) menyiratkan pendekatan yang sistematis dan berurutan pada pengembangan perangkat lunak, yang dimulai dengan spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan (*planning*), Pemodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem atau perangkat lunak ke para pelanggan atau pengguna (*deployment*), yang

diakhiri dengan dukungan berkelanjutan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2012) (Gambar 2.3).



Gambar 2.3 Arsitektur *Waterfall* (Sumber: pressmen,2012)

### **2.6.1 Communication**

Pada komunikasi berisi permulaan proyek dengan teknik untuk mendapatkan spesifikasi kebutuhan pengguna serta teknik-teknik analisis data.

### **2.6.2 Planning**

Tahap ini merupakan perencanaan untuk prakiraan-prakiraan penjadwalan, pelacakan.

### **2.6.3 Modelling**

Pada tahap pemodelan ini dilakukan analisis data yang sudah dikumpulkan dari komunikasi yang kemudian di bentuk rancangan desain sistem

### **2.6.4 Construction**

Merupakan bagian untuk menerjemahkan data yang telah dirancang ke dalam kode bahasa pemrograman komputer yang telah ditentukan. Dan dilakukan pengujian sistem untuk menentukan kualitas sistem

### **2.6.5 Deployment**

Yaitu penyerahan sistem kepada pengguna untuk mendapatkan dukungan terhadap pengguna.

## 2.5 Pengujian *Black-Box Testing*

*Black Box Testing* atau Pengujian Kotak Hitam berfokus pada persyaratan fungsional dari perangkat lunak. Dengan demikian pengujian *Black-Box Testing* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. *Black-Box Testing* bukan merupakan alternatif dari pengujian *White Box Testing*. Tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan daripada metode *whitebox*. Tidak seperti pengujian *whitebox* yang dilakukan pada awal proses pengujian, pengujian *blackbox* cenderung diaplikasikan selama tahap akhir pengujian. Karena pengujian *blackbox* memperhatikan struktur control, maka perhatian berfokus pada domain informasi. *Black Box Testing* berusaha untuk menemukan kesalahan dalam kategori sebagai berikut. (Pressman, 2012)

- a. Fungsi-fungsi yang tidak benar atau hilang.
- b. Kesalahan interface atau antarmuka.
- c. Kesalahan dalam struktur data atau akses database eksternal.
- d. Kesalahan kinerja atau perilaku.
- e. Kesalahan inisialisasi dan terminasi.

## 2.6 *MySQL*

*MySQL* merupakan *software* RDBMS (atau server *database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*). Saat ini, *MySQL* banyak digunakan berbagai kalangan untuk melakukan penyimpanan dan pengolahan data, mulai dari kalangan akademis sampai keindustri, baik industry kecil, menengah, maupun besar. Lisensi *MySQL* terbagi menjadi dua. Anda dapat menggunakan *MySQL* sebagai produk open source dibawa GNU *General Public License* (gratis) atau dapat membeli lisensi dari versi komersialnya. *MySQL* versi komersial tentu memiliki nilai lebih atau kemampuan-kemampuan yang tidak disertakan pada



versi gratis. Pada kenyataannya, keperluan industri menengah kebawah, versi gratis masih dapat digunakan dengan baik.(Liliany, 2014)

### **2.7 PHP (*Hypertext Preprocessor*)**

PHP merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses server. Hasilnya yang dikirimkan ke klien, tempat pemakai menggunakan *browser*. *PHP* dirancang untuk membentuk web dinamis, artinya ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, menampilkan isi database ke halaman web. Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip perl yang dapat mengamati siapa saja yang dapat melihat daftar riwayat hidupnya pada tahun 1994. Skrip-skrip ini dikemas menjadi *tool* yang disebut *personal home page*. Inilah yang menjadi cikal bakal PHP. Pada tahun 1995 Rasmus menciptakan PHP/FI versi 2. Pada versi inilah program dapat menempelkan kode terstruktur di dalam tag HTML. Kode PHP juga dapat berkomunikasi dengan database dan melakukan perhitungan yang kompleks (Abdul Kadir).

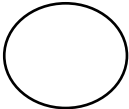
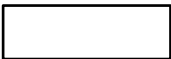
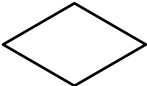

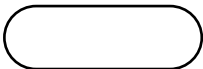
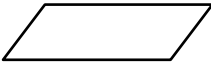




### **2.8 HTML 5 (*Hypertext Markup Language*)**

*HTML* merupakan file teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen *HTML* disajikan dalam browser web surfer. *HTML 5* merupakan salah satu karya *World Wide Web Consortium, W3C* untuk mendefinisikan sebuah bahasa *markup* tunggal yang dapat ditulis dengan cara *HTML* ataupun *XHTML*. *HTML 5* merupakan jawaban atas pengembangan *HTML 4.01* dan *XHTML 1.1* yang selama ini berjalan terpisah, dan diimplementasikan secara berbeda-beda oleh banyak perangkat lunak pembuat *web* (Sumber : betha sidiq)

## 2.9 Flowchart

*Flowchart* adalah representasi grafik yang menggambarkan setiap langkah yang akan dilakukan didalam suatu proses, yang merupakan alat bantu yang banyak digunakan untuk menggambarkan sistem secara pisikal. *Flowchart* disebut juga diagram alir yang mewakili alir kerja proses yang ditampilkan dalam bentuk simbol-simbol grafis secara berurutan. Untuk membuat *flowchart* digunakan simbol-simbol sebagaiberikut. (Abdillah, 2009)




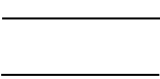
**Tabel 2.1 Simbol *Flowchart***

Simbol	Nama	Fungsi
	On page connector	Penghubung dalam satu halaman
	Proses	Berisi proses didalam aliran
	Decision	Penyeleleksi data
	Garis alir	Menunjuk arah aliran program
	Terminator	Untuk memulai/mengakhiri aliran
	Input/output	Berisi proses input/output
	Document	Mencetak hasil
	Manual operation	Pekerjaan/operasi secara manual
	Storage data	Input/output yang berasal dari disk
	Predefined process	Sebagai pendefinisi program ataupun subprogram

### 2.10 Data Flow Diagram

*Data flow diagram (DFD)* memperlihatkan gambaran tentang masukan-proses-keluaran dari suatu sistem atau perangkat lunak. Yaitu, objek-objek data mengalir ke dalam perangkat lunak, kemudian objek-objek data itu akan ditransformasi oleh elemen-elemen pemrosesan, dan objek-objek data itu hasilnya akan mengalir keluar dari sistem atau perangkat lunak. Objek-objek data dalam penggambaran *DFD* biasanya direpresentasikan menggunakan tanda panah berlabel, dan transformasi-transformasi biasanya di representasikan menggunakan lingkaranlingkaran. *DFD* pada dasarnya digambarkan dalam bentuk hierarki. *DFD* yang pertama (sering dinamakan sebagai *DFD* peringkat 0 atau diagram konteks) menggambarkan sistem secara keseluruhan. *DFD-DFD* berikutnya sesungguhnya merupakan penghalusan dari diagram konteks, memberikan gambaran yang semakin rinci dari ari diagram konteks, dan hal ini akan berkelanjutan ke peringkat-peringkat selanjutnya, simbol-simbol yang digunakan pada aliran data *DFD* sebagai berikut. (Pressman, 2012 )

**Tabel 2.2 Simbol Data Flow Diagram**

Simbol	Nama	Fungsi
	Entitas	Entitas eksternal dapat berupa unit terkait yang berinteraksi dengan sistem tetati diluar sistem.
	Proses	Unit yan mempergunakan transformasi data.
	Aliran data	Aliran data dengan arah khusus dari sumber ke tujuan.
	Data store	Penyimpanan data atau tempat data direfer oleh proses.

### 2.11 Penelitian Terkait

Penelitian-penelitian sebelumnya digunakan sebagai tinjauan dalam penelitian sistem pakar penyakit *herniated nucleus pulposus* menggunakan metode *case base reasoning*. Pada tabel berikut ini adalah beberapa penelitian yang berkaitan dengan metode *case based reasoning* dan penyakit *herniated nucleus pulposus*.

**Tabel.2.3 Penelitian Terkait**

No	Nama	Judul	Terbit	Uraian
1	Fitria, Sri L, Wawan D.	Metode <i>case based reasoning</i> pada sistem diagnosa penyakit kulit	IIB Darmajaya, juni 2018	Metode <i>CBR</i> , merupakan metode pemecahan masalah berdasarkan pengalaman pada masa lalu yang di simpan dalam studi kasus.
2	Chairan i Fauzi	Rancang Bangun sistem pakar pendiagnosa penyakit paru-paru menggunakan metode <i>case based reasoning</i> .	Universitas Gadjah Mada, januari 2015	Penelitian ini memberikan keluaran berupa penyakit dan saran pengobatan yang didasarkan pada kemiripan kasus yang dimiliki sistem.
3	Asti Herlina	Sistem Pakar <i>Diagnosa Hernia Nukleus Pulposus</i> Menggunakan <i>Forward chaining</i> Berbasis <i>Web</i>	Universitas Bhayangkara raya Jakarta September 2017	Sistem pakar yang berfungsi untuk mendiagnosis gejala awal penyakit <i>HNP</i> berbasis <i>web</i> .