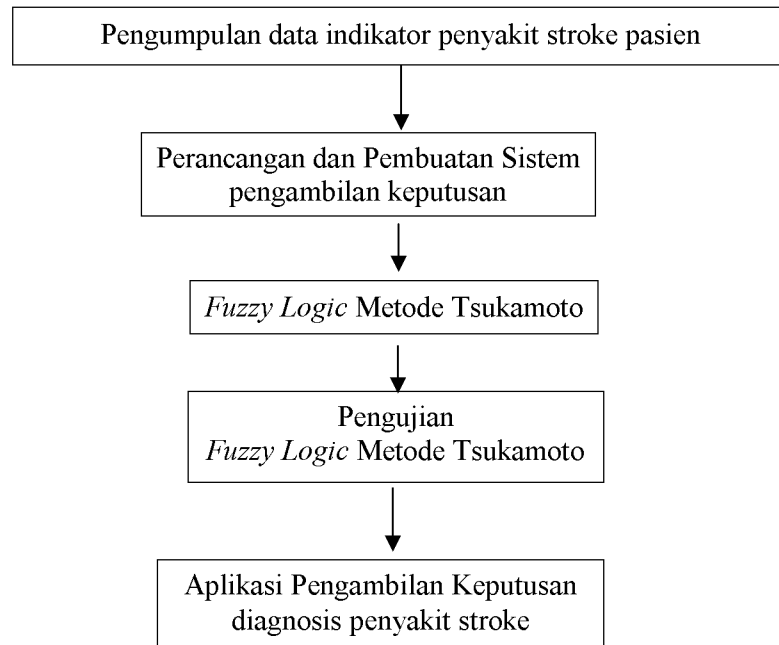


## BAB III

### METODE PENELITIAN

#### 3.1. Kerangka Penelitian



Gambar 5 Kerangka Penelitian

#### 3.2. Variabel Penelitian

Variabel dalam penelitian ini adalah

1. Deteksi dini penyakit stroke adalah upaya diagnosis yang diduga mengalami stroke dengan gejala-gejala tertentu.
2. Logika Fuzzy Pendeteksi Dini Penyakit Stroke

Kecerdasan teknologi, yang dibuat untuk melakukan diagnosis yang diduga mengalami stroke dengan gejala-gejala tertentu.

#### 3.3. Tahapan Penelitian

##### 3.3.1. Pengumpulan Data

Pada tahap pengumpulan data, data didapatkan dari berbagai sumber seperti diperoleh dari artikel-artikel, e-book, jurnal, dan

diperoleh dari referensi-referensi lain, dan penelitian ini nantinya akan dilakukan wawancara terhadap pakarnya, agar data yang didapatkan lebih akurat

### 3.3.2. Analisa Data

Tahapan analisa data adalah tahapan dimana seluruh data dianalisa, mengidentifikasi masalah yang ada, menentukan rancangan sistem dan langkah langkah yang dibutuhkan untuk perancangan yang diinginkan. Pada tahap inilah nantinya akan dilakukan evaluasi kinerja.

### 3.3.3. Analisa Sistem

Analisa ini dilakukan untuk mengetahui apa saja kebutuhan yang dibutuhkan untuk merancang program untuk mendiagnosa penyakit Dalam (Stroke Iskemik TIA, Stroke Iskemik RIND dan disertai penyakit lainnya yaitu Diabetes Mellitus, Hipertensi, Stroke, Gagal Jantung dan Gagal ginjal kronik), dan bagaimana kerja dari program yang dirancang seperti kerja dari input, proses, dan outputnya. sehingga sistem pakar yang di rancang menjadi efektif dan efesien dalam pengimplementasiannya nanti.

### 3.3.4. Perancangan Sistem

Tahap ini dilakukan sebelum melakukan coding. Tahap ini bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan dan bagaimana tampilannya. Sebelum melakukan gambaran tentang perancangan sistem ini, ada beberapa tahapan untuk menganalisis

data yang ada yaitu dengan cara menggunakan metode UML (unified modelling language) antara lain:

a. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah use case mempresentasikan sebuah interaksi antara actor dengan sistem. Use case merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng – create sebuah daftar belanja dan sebagainya. Seorang actor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan – pekerjaan tertentu.

b. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/property) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).

c. State Chart Diagram

State chart diagram menggambarkan transisi dan perubahan keadaan (dari satu state ke state lainnya) suatu objek pada sistem sebagai akibat dari stimulasi yang diterima. Pada umumnya state chart diagram menggambarkan class tertentu (satu class dapat memiliki lebih dari satu state chart diagram). Dalam UML, state

digambarkan berbentuk segiempat dengan sudut membulat dan memiliki nama sesuai kondisinya saat itu. Transisi antar state umumnya memiliki kondisi guard yang merupakan syarat terjadinya transisi yang bersangkutan, dituliskan dalam kurung siku. Action yang dilakukan sebagai akibat dari event tertentu dituliskan dengan diawali garis miring.

#### d. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa message yang digambarkan terhadap waktu. Sequence diagram terdiri antar dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait). Sequence diagram biasa digunakan untuk menggambarkan scenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu. Masing-masing objek, termasuk actor, memiliki lifeline vertical. Message digambarkan sebagai garis berpanah dari suatu objek ke objek lainnya. Pada fase desain berikutnya, message akan dipetakan menjadi operasi/ metode dari class. Activation menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message.

#### e. Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi dan bagaimana mereka berakhir.

Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Activity diagram merupakan state diagram khusus, dimana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (Internal Processing). Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana actor menggunakan istem untuk melakukan aktivitas.

#### f. Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antar objek seperti sequence diagram, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian message. Setiap message memiliki sequence number, dimana message dari level tertinggi memiliki nomor 1. Message dari level yang sama memiliki prefix yang sama.

#### g. Deployment Diagram

Deployment/ physical diagram menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, dimana komponen akan terletak (pada mesin, server atau piranti keras lainnya), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisik. Sebuah node adalah server, workstation atau piranti keras lainnya yang digunakan untuk men-deploy komponen dalam lingkungan sebenarnya. Hubungan

antar node (misalnya TCP/IP) dan requirement dapat juga didefinisikan dalam diagram ini.

### **3.4 Pengujian**

Proses selanjutnya adalah pengujian program, program yang telah dibangun akan diuji untuk mengetahui apakah program tersebut berjalan dengan benar dan sesuai dengan perancangan yang dilakukan. Setelah program selesai maka tahap selanjutnya adalah tahapan pemeliharaan. Untuk pengujian aplikasi ini menggunakan cara aplikasi langsung pasien yang diduga mengalami gejala stroke.

### **3.5 Implementasi**

Implementasi ini dilakukan untuk mengetahui spesifikasi komputer untuk menjalankan program dan program apa saja yang di butuhkan. Merupakan tahap penelitian yang dilakukan untuk mempraktekkan langsung hasil dari analisa yang bertujuan untuk menguji kebenaran sistem yang dirancang.