# BAB III METODE PENELITIAN

## Jenis, Metode, Subjek, Waktu dan Lokasi Penelitian

Dipandang dari tingkat penerapan maka, penelitian ini merupakan penelitian terapan. Dipandang dari jenis informasi yang diolah maka, penelitian ini merupakan penelitian kuantitatif.

Penelitian ini menggunakan metode penelitian studi kasus. Dengan demikian jenis penelitian ini adalah penelitian deskriptif.

Berdasarkan latar belakang dan kerangka pemikiran seperti yang telah diuraikan diatas maka yang menjadi objek penelitian adalah Penerapan Algoritma *Naïve Bayes* untuk **klasifikasi RTSM**. Penelitian ini dimulai dari Agustus 2018 s/d Maret 2019 yang berlokasi diKantor Camat Bulango Utara Kabupaten Gorontalo.

## Pengumpulan Data

Data primer pada penelitian ini adalah data masuk untuk **klasifikasi RTSM** dari Januari 2018 s/d Desember 2018 yang dikumpulkan menggunakan teknik dokumentasi dan observasi sedangkan data sekunder dikumpulkan menggunakan teknik dokumentasi. Adapun variabel atau atribut dengan tipe datanya masing-masing pada tabel berikut**:**

**Tabel 3. 1** Atrribut data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | NAME | TYPE | VALUE | KETERANGAN |
| 1. | Jenis Kelamin | *integer* | Laki-laki ,perempuan | Variabel Input |
| 2. | Umur | Numeric | 20-70 | Variabel Input |
| No | NAMA | TYPE | VALUE | KETERANGAN |
| 3. | Pekerjaan | Numeric | 2015, 2016, 2017, 2018 | Variabel Input |
| 4. | Penghasilan | Numeric | 100 - 800 | Variabel Input |
| 5. | Status Kepemilikan | *Integer* | Milik Sendiri, Menumpang | Variabel Input |
| 6. | Jumlah Tanggungan | Numeric | 0-10 | Variabel Input |
| 7. | Unsur Dinding | *Integer* | Papan, Bambu, semen | Variabel Input |
| 8. | Unsur Lantai | *Integer* | Papan, tanah, cor, tehel | Variabel Input |
| 9. | Unsur Atap | *Integer* | Rumbia, Seng | Variabel Input |
| 10. | Keterangan | *Integer* | Miskin, sangat miskin | Variabel Output |

## 

## Pemodelan / Abstraksi

### Pengembangan Model

Prosedur atau langkah-langkah pokok dalam prediksi menggunakan algoritma *Naïve Bayes* untuk klasifikasi RTSM dengan menggunakan alat bantu Rapid Miner dan tools PHP, Database MySQL serta *White Box Testing* dan *Black Box Testing* untuk menguji kinerja sistemnya.

### Evaluasi Model

Model yang telah dihasilkan kemudian dievaluasi dengan menggunakan *Confusion Matrix* untuk mengetahui akurasi.

## Pengembangan Sistem

Sistem yang diusulkan dapat digambarkan menggunakan *flowchart* berikut ini

|  |  |
| --- | --- |
| Admin | Camat |
| Password Login Admin    Edit Password  Login  Data Klasifikasi  Save  Password    Input Data  Klasifikasi  Pemrosesan Algoritma *Naïve Bayes*  Save Hasil  Klasifikasi  HasilKlasifikasi | Laporan  Klasifikasi RTSM  **N** |

**Gambar 3. 1** Gambar Sistem Yang Diusulan

### Analisis Sistem

Analisis sistem menggunakan pendekatan *prosedural/struktural* yang digambarkan dalam bentuk :

1).Diagram Konteks, menggunakan alat bantu DFD

2). Diagram Berjenjang, menggunakan alat bantu DFD

3). Diagram Arus Data Level 0, 1, dst, menggunakan alat bantu DFD

4). Kamus Data, menggunakan alat bantu Visio.

### 3.3.2 Desain Sistem

Desain sistem menggunakan pendekatan *prosedural/struktural* yang digambarkan dalam bentuk :

1). Desain Output, menggunakan alat bantu DFD, dalam bentuk :

1. Desain Output secara Umum.

2. Desain Output secara Terinci.

2). Desain Input, menggunakan alat bantu DFD, dalam bentuk :

1. Desain Output secara Umum.

2. Desain Output secara Terinci.

3). Desain Basis Data*,* menggunakan alat bantu DFD, dalam bentuk :

1. Struktur Data.

2. Entity Relationship Diagram.

4). Desain Teknologi, menggunakan alat bantu DFD, dalam bentuk :

1. Model jaringan dari sistem *client server.*

2. Spesifikasi *hardware* dan *software* yang direkomendasikan adalah :

1). Sistem Operasi : Windows 10

2). Prosesor Dengan Kecepatan Minimal 2.4GHz

3). RAM 4 GB

4). HDD 500GB

### Konstruki Sistem

Pada tahap ini dilakukan pembuatan sistem menggunakan *tools PHP* dan Database *MySQL*serta *White Box Testing* dan *Black Box Testing* untuk menguji kinerja sistem dan pengukuran akurasi menggunakan *Confusion Matrix*. Pada tahap ini kita melakukan tahap produksi sistem hasil analisa dan desain sistem sebelumnya. Termasuk didalamnya menginstal paket tambahan untuk menjalankan program, menulis listing program dan membangunnya dalam bentuk sebuah formulir, antarmuka dan integrasi sistem-sistem program yang terdiri dari input, proses dan output yang tersusun dalam sebuah sistem menu sehingga dapat dijalankan oleh pengguna sistem.

### Pengujian Sistem

Setelah dilakukan tahap analisa, desain dan produksi sistem, maka kita melakukan tahap pengujian, dimana seluruh perangkat lunak, program tambahan dan semua program yang terlibat dalam pembangunan sistem diuji untuk memastikan system dapat berjalan dengan semestinya. Testing difokuskan pada logika internal, fungsi eksternal dan mencari segala kemungkinan kesalahan dari sistem yang dibuat. Pada tahap ini dilakukan review dan evaluasi terhadap sistem yang dikembangkan, apakah sudah sesuai dengan rancangan atau belum. Jika terjadi hal-hal yang tidak sesuai dengan yang diharapkan, kemudian dilakukan revisi atau perbaikan supaya produk tersebut dapat dioperasikan dengan baik dan siap untuk di implementasikan. Pengujian yang dilakukan dengan menggunakan teknik pengujian perangkat lunak yaitu :

1. Pengujian *White Box*

Software yang sudah direkayasa kemudian diuji dengan metode white box testing pada kode program proses penerapan metode-modelnya. Kode program tersebut dibuatkan flowchart programnya, kemudian dipetakan kedalam bentuk *flowgraph* (bagan alir kontrol) yang tersusun dari beberapa nide dan *edge.* Berdasarkan *flowgraph,* ditentukan jumlah *region* dan *Cyclomatic Complexity* (CC). Apabila *Independent Path* = V(G)=(CC) = *Region,* di mana setiap *Path* hanya dieksekusi sekali dan sudah benar, maka sistem dinyatakan efisien dari segi kelayakan logika pemrograman

1. Pengujian *Black Box*

Pengujian *Black Box* melalui program *PHP* dan Database *MySQL.*Selanjutnya software diuji pula dengan metode black box testing yang fokus pada keperluan fungsional dari *software* dan berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya : (1) Fungsi-fungsi yang salah atau hilang; (2) kesalahan *interface*; (3) kesalahan dala struktur data atau akses basis data ekseternal; (4) kesalahan performa; (5) kesalahan inisialisasi dan terminasi. Jika sudah tidak ada kesalahan-kesalahan tersebut, maka sistem dinyatakan efisien dari segi kesalahan komponen-komponen sistem.