

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Definisi Sistem**

Definisi sistem berkembang sesuai konteks dimana pengertian sistem itu digunakan. Berikut akan diberikan definisi sistem secara umum:

1. Kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama (Jogiyanto, H.M. 2005).
2. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (Jogiyanto, H.M. 2005).
3. Suatu sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu (Jogiyanto, H.M. 2005).
4. Sistem merupakan kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan (input) yang ditujukan kepada sistem tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (output) yang diinginkan (Kadir A, Triwahyuni TCH. 2003).

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu:

- a. Komponen sistem (components)
- b. Batas sistem (boundary)
- c. Lingkungan luar sistem (environments)
- d. Penghubung sistem (interface)
- e. Masukan sistem (input)
- f. Keluaran sistem (output)
- g. Pengolah sistem
- h. Sasaran sistem (objective)

## 2.2 Pengertian Potensi

Potensi berasal dari bahasa Inggris “*to potent*” yang berarti kekuatan (*powerful*), *daya*, *kekuatan*, *kemampuan*. Setiap individu pada hakekatnya memiliki suatu potensi yang dapat dikembangkan, baik secara individu maupun kelompok melalui latihan- latihan. Sedangkan menurut Prof DR. Buchori Zainun, MPA yang disebut potensi adalah Daya atau kekuatan baik yang sudah teraktualisasi tetapi belum optimal maupun belum teraktualiasasi. Daya tersebut dapat bersifat positif yang berupa kekuatan (*power*), yang bersifat negatif berupa kelemahan (*weakness*). Dalam pengembangan potensi diri yang dikembangkan adalah yang positif, sedangkan yang negatif justru harus dicegah dan dihambat agar tidak berkembang. Potensi-potensi tersebut merupakan salah satu pembeda antara individu yang satu dengan individu yang lain. Sedangkan orang yang Potensial (*potential*) dicirikan dengan adanya potensi, memiliki kemampuan laten untuk melakukan sesuatu atau untuk bertingkah laku dengan cara tertentu, khususnya dengan cara yang mencakup laten atau bakat pembawaan atau intelligensi (Lembaga Administrasi Negara Republik Indonesia; Pengembangan Potensi Diri; 2014).

## 2.3 Pengertian Keahlian

Keahlian merupakan sesuatu minat atau bakat yang harus dimiliki oleh seseorang, dengan keahlian yang dimilikinya memungkinkan untuk dapat menjalankan dan menyelesaikan tugas-tugas secara baik dengan hasil yang maksimal. Keahlian yang dimiliki seseorang dapat diperoleh dari pendidikan formal maupun non formal yang nantinya harus terus menerus ditingkatkan. Salah satu sumber peningkatan keahlian dapat berasal dari pengalaman-pengalaman dalam bidang tertentu. Pengalaman tersebut dapat diperoleh melalui proses yang bertahap, seperti pelaksanaan tugas-tugas, pelatihan ataupun kegiatanlainnya yang berkaitan dengan pengembangan keahlian seseorang.

Hasibuan (2003) menyatakan bahwa keahlian harus mendapat perhatian utama kualifikasi seleksi. Hal ini yang akan menentukan mampu tidaknya seseorang menyelesaikan pekerjaan yang ditugaskan kepadanya. Keahlian ini mencakup *technical skill*, *human skill*, *conceptual skill*, kecakapan untuk memanfaatkan kesempatan serta kecermatan penggunaan peralatan yang dimiliki organisasi dalam mencapai tujuan. Menurut Hasibuan (2003), dalam suatu program pengembangan ditetapkan suatu sasaran, proses, waktu dan metode pelaksanaannya. Supaya lebih baik program itu harus dibuat perencanaan terlebih dahulu, karena metode pengembangan didasarkan pada tujuan yang ingin dicapai. Lebih lanjut Hasibuan (2003) menyatakan bahwa pertimbangan promosi adalah kecakapan, orang yang cakap atau ahli mendapat prioritas pertama dalam promosi. Kecakapan adalah total dari semua keahlian yang diperlukan untuk mencapai hasil yang bisa dipertanggungjawabkan.

#### **2.4 Data Mining**

Tan (2006) mendefinisikan *data mining* sebagai proses untuk mendapatkan informasi yang berguna dari gudang basis data yang besar. *Data mining* juga dapat diartikan sebagai pengekstrakan informasi baru yang diambil dari bongkahan data besar yang membantu dalam pengambilan keputusan. Istilah *data mining* kadang disebut juga *knowledge discovery*. Selanjutnya perbedaan *data mining* dan *data warehouse* adalah *data mining* merupakan bidang yang sepenuhnya menggunakan apa yang sepenuhnya digunakan oleh *data warehouse*, bersama dengan bidang yang menangani masalah pelaporan dan manajemen data. Sementara *data warehouse* sendiri bertugas untuk menarik atau meng*query* data dari basis data mentah untuk memberikan hasil data yang nantinya digunakan oleh bidang yang menangani manajemen, pelaporan, dan *data mining*. Dengan *data mining* inilah, penggalian informasi baru dapat dilakukan dengan bekal data mentah yang diberikan

oleh *data warehouse*. Hasil yang diberikan oleh ketiga bidang tersebut berguna untuk mendukung aktivitas bisnis cerdas (*business intellegent*).

Pekerjaan yang berkaitan dengan *data mining* dibagi menjadi 4 (empat) kelompok, yaitu (Prasetyo, E, 2012) :

1. *Prediction Modelling* (model prediksi), berkaitan dengan sebuah model yang dapat melakukan pemetaan dari setiap himpunan variabel ke setiap targetnya, kemudian menggunakan model tersebut untuk memberikan nilai target pada himpunan baru yang didapat.
2. *Cluster Analysis* (analisis kelompok), pengelompokan data-data kedalam sejumlah kelompok (*cluster*) berdasarkan kesamaan karakteristik masing-masing data pada kelompok-kelompok yang ada.
3. *Association Analysis* (analisis asosiasi), digunakan untuk menemukan pola yang menggambarkan kekuatan fitur dalam data dengan tujuan untuk menemukan pola yang menarik dengan cara yang efisien.
4. *Anomaly Detection* (deteksi anomali), berkaitan dengan pengamatan sebuah data dari sejumlah data yang secara signifikan mempunyai karakteristik yang berbeda dari sisa data yang lain.

## 2.5 **Klasifikasi**

Klasifikasi adalah proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui. Dalam mencapai tujuan tersebut, proses klasifikasi membentuk suatu model yang mampu membedakan data kedalam kelas-kelas yang berbeda berdasarkan aturan atau fungsi tertentu.

Tahapan dari klasifikasi dalam data mining menurut (Han dan Kamber, 2006) terdiri dari :

### 1. Pembangunan Model

Pada tahapan ini dibuat sebuah model untuk menyelesaikan masalah klasifikasi class atau attribut dalam data. Tahap ini merupakan fase pelatihan, dimana data latih dianalisis menggunakan algoritma

klasifikasi, sehingga model pembelajaran direpresentasikan dalam bentuk aturan klasifikasi.

## 2. Penerapan Model

Pada tahapan ini model yang sudah dibangun sebelumnya digunakan untuk menentukan atribut/class dari sebuah data baru yang atribut/classnya belum diketahui sebelumnya. Tahap ini digunakan untuk memperkirakan keakuratan aturan klasifikasi terhadap data uji. Jika model dapat diterima, maka aturan dapat diterapkan terhadap klasifikasi data baru.

## 2.6 PHP (Personal Home Page)

PHP adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode – kode ( *script* ) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML. PHP diciptakan pertama kali oleh Ramus Lerdorf pada tahun 1994. Awalnya, PHP digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Rasmus Lerdorf adalah salah seorang pendukung *Open Source*.

Tahun 2004 bulan juli dirilis PHP 5 dengan inti *Zend Engine 2.0*. PHP 5 adalah versi PHP terbaru yang mendukung penuh *object-oriented programming* (OOP), integrasi XML, mendukung semua eksistensi terbaru MySQL, pengembangan *web service* dengan SOAP dan REST, serta ratusan peningkatan lainnya dibandingkan dengan versi seelumnya PHP 4.0. PHP dapat digunakan pada semua sistem operasi, antara lain : Linux, Unix, Microsoft Windows, Mac OS X, RISC OS. PHP juga mendukung banyak *Web Server*. (Harahap, L, 2009).

## 2.7 Apache

Server HTTP Apache atau Server Web/WWW Apache adalah server web yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya)

yang berguna untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini menggunakan HTTP. Apache memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigur, autentikasi berbasis basis data dan lain-lain. Apache juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah.

## **2.8 SQLyog Community**

SQLyog adalah salah satu tool administrasi untuk database MySQL. Jika kita biasanya menggunakan PhpMyAdmin yang include di dalam aplikasi Xampp untuk melakukan administrasi database, SQLyog adalah aplikasi alternatif untuk melakukan proses administrasi database MySQL. Banyak fitur yang disediakan oleh SQLyog yang tidak disediakan oleh PhpMyAdmin maupun tool administrasi database lainnya seperti MySQLQueryBrowser. Dengan SQLyog kita dapat membuat Store Procedure, Function maupun Trigger dengan mudah.

## **2.9 Buku Panduan Akademik UMG 2013 / 2014**

Buku panduan akademik merupakan buku yang menjadi petunjuk bagi setiap mahasiswa selama melaksanakan perkuliahan. Buku panduan akademik bisa jadi berbeda untuk setiap universitas. Buku panduan akademik yang akan digunakan dalam penelitian ini merupakan buku panduan akademik Universitas Muhammadiyah Gresik tahun 2013 / 2014. Buku ini berisi informasi umum, peraturan akademik, prosedur-prosedur kerja dan kurikulum untuk setiap program studi di Universitas Muhammadiyah Gresik. Selain itu buku panduan akademik juga memberikan informasi tentang penjurusan suatu mahasiswa terhadap fakultas yang dipilihnya.

## 2.10 Rekayasa Perangkat Lunak

Rekayasa Perangkat Lunak (RPL) telah dikenal sebagai bidang ilmu sejak tahun 1960-an. RPL yaitu ilmu yang mempelajari tentang bagaimana memanfaatkan sebuah perangkat keras (*hardware*) dan perangkat lunak (*software*) untuk dijadikan suatu project yang bisa digunakan dengan baik serta ilmu yang mempelajari tentang tahapan dalam pembuatan suatu project. Rekayasa Perangkat Lunak juga pada dasarnya terdiri dari komponen-komponen perangkat lunak yang bertugas membuat suatu desain aplikasi yang ada dilingkungan tugas atau pekerjaan serta membuat deadline berupa pendahuluan, perumusan masalah, analisis, desain, dan kesimpulan (Ladjamuddin B, Al-Bahra, 2006).

Rekayasa yang dilakukan terhadap suatu perangkat lunak memiliki beberapa tujuan diantaranya adalah (Ladjamuddin B, Al-Bahra, 2006) :

1. Biaya produksi rendah
2. Kinerja program yang tinggi
3. Portabilitas yang tinggi
4. Biaya perawatan yang rendah
5. Keandalan sistem dan penyerahan tepat waktu

## 2.11 Sistem Cerdas

Menurut (Zadeh, Lotfi A., 1994) sistem cerdas adalah sistem yang mampu melakukan penalaran dalam ketidakpastian (*uncertainty*), ketidaktepatan (*imprecision*), kebenaran sebagian (*partial truth*), pendekatan (*approximation*). Sistem cerdas sendiri merupakan suatu sistem yang terinspirasi dari kecerdasan manusia. Kecerdasan yang diciptakan tersebut diinputkan ke dalam sebuah mesin dengan tujuan pekerjaan-pekerjaan tertentu seperti pekerjaan yang dilakukan oleh seorang manusia. Sistem cerdas sendiri mempunyai tiga konsep, yaitu (Latifah, 2013) :

1. *Artificial Neural Network*

Konsep tersebut mengenai kerja dari sel saraf dari otak manusia. Untuk mendapatkan mesin dengan cara kerja sel saraf yang sama dengan manusia, diperlukan perumusan persamaan model matematik. Model persamaan tersebut dituliskan pada program-program seperti Visual Basic, Delphi, dan Matlab.

## 2. *Artificial Intelligence*

Konsep ini membahas mengenai sebuah mesin yang dapat mengambil keputusan tanpa diperintah oleh seorang manusia. Sehingga diperlukan logika berpikir mesin yang sama dengan manusia.

## 3. *Fuzzy Logic*

Merupakan logika yang mempunyai perasaan dan toleransi seperti manusia. Pada logika konvensional sebelumnya, hanya dikenal option “YA” atau “TIDAK”. Sedangkan logika samar dimaksudkan bahwa logika yang bernilai samar yang benar terlihat tidak benar dan salah bernilai tidak samar. Sehingga dalam kerjanya, logika tersebut menggunakan perasaan. Contohnya, sebuah robot berlogika samar dapat mengerti arti kata sedikit atau banyak dengan menggunakan logika yang ia miliki.

### 2.12 Normalisasi Data

Normalisasi data linier adalah proses penskalaan nilai atribut data sehingga bisa jatuh pada range tertentu. Tujuan dari normalisasi data adalah untuk mempersempit atau mengecilkan nilai range pada data tersebut. Normalisasi yang digunakan pada penelitian ini adalah *min-max normalization* yang merupakan proses transformasi nilai dari data yang dikumpulkan pada range *value* antara 0.0 dan 1.0, dimana nilai terkecil (*min*) adalah 0.0 dan nilai tertinggi (*max*) adalah 1.0 (Chandrasekhar, Thangavel dan Elayaraja, 2011). Keuntungan dari metode ini adalah keseimbangan nilai perbandingan antara data saat sebelum dan sesudah nilai normalisasi. Kekurangannya adalah jika ada data baru metode ini akan memungkinkan terjebak pada out of bound error. Normalisasi data



sangat di perlukan ketika data yang ada terlalu besar atau terlalu kecil sehingga pengguna kesulitan memahami informasi yang di maksud. Jika rentan nilai normalisasi yang di inginkan berada pada rentan [0,1] maka dapat juga menggunakan persamaan berikut :

$$\text{normalisasi } X^* = \frac{(X - \min X)}{(max X - \min X)} \dots\dots\dots (2.1)$$

Keterangan :

$X^*$  = nilai hasil normalisasi

$X$  = nilai x sebelum normalisasi

min = nilai minimum dari fitur

max = nilai maksimum dari fitur

### 2.13 *K-Nearest Neighbor* (KNN)

*K-Nearest Neighbors* (KNN) adalah sebuah metode klasifikasi terhadap sekumpulan data berdasarkan pembelajaran data yang sudah terklasifikasikan sebelumnya. KNN termasuk dalam golongan *supervised learning*, dimana hasil *query instance* yang baru diklasifikasikan berdasarkan mayoritas kedekatan jarak dari kategori yang ada dalam KNN. Nantinya kelas yang baru dari suatu data akan dipilih berdasarkan grup kelas yang paling dekat jarak vektornya.

Tujuan dari algoritma ini adalah mengklasifikasikan obyek baru berdasarkan atribut dan *training sample*. *Classifier* tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori. Diberikan titik *query*, akan ditemukan sejumlah k obyek atau (titik *training*) yang paling dekat dengan titik *query*. Klasifikasi menggunakan *voting* terbanyak diantara klasifikasi dari k obyek. Algoritma *K-Nearest Neighbors* menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari *query instance* yang baru.

Metode *K-Nearest Neighbors* sangatlah sederhana, bekerja berdasarkan jarak terpendek dari *query instance* ke *training sample* untuk menentukan KNN- nya. *Training sample* diproyeksikan ke ruang

berdimensi banyak, dimana masing- masing dimensi merepresentasikan fitur dari data. Ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi *training sample*. Sebuah titik pada ruang ini ditandai jika kelas *c* merupakan klasifikasi yang paling banyak ditemui pada *k* buah tetangga terdekat dari titik tersebut. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan *Euclidean Distance*.

Jarak *Euclidean* paling sering digunakan dalam menghitung jarak karena sangat cocok untuk menggunakan jarak terdekat (lurus) antara dua data. Jarak *euclidean* berfungsi menguji ukuran yang bisa digunakan sebagai interpretasi kedekatan jarak antara dua obyek yang direpresentasikan sebagai berikut:

$$D(a, b) = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2} \dots \dots \dots (2.2)$$

Keterangan :

$D(a, b)$  = Jarak Euclidean Data a dan Data b

X = Koordinat titik X

Y = Koordinat titik Y

Menurut Eko Prasetyo (Eko Prasetyo, 2012) dalam buku Konsep dan Aplikasi *Data Mining*, salah satu masalah yang dihadapi KNN adalah pemilihan nilai *K* yang tepat. Misalnya, diambil *K* bernilai 13, kelas 0 dimiliki oleh 7 tetangga yang jauh, sedangkan kelas 1 dimiliki oleh 6 tetangga yang lebih dekat. Hal ini mengakibatkan data uji tersebut akan terdistorsi sehingga ikut bergabung dengan kelas 0. Hal ini karena setiap tetangga tersebut mempunyai bobot yang sama terhadap data uji, sedangkan *K* yang terlalu kecil bisa menyebabkan algoritma terlalu sensitif terhadap *noise*. Nilai *k* yang bagus dapat dipilih berdasarkan optimisasi parameter, misalkan dengan *cross validation*. Kasus dimana klasifikasi diprediksikan berdasarkan *training* data yang paling dekat (dengan kata lain,  $k = 1$ ) disebut algoritma *nearest neighbors*.

Ketepatan algoritma KNN sangat dipengaruhi oleh ada atau tidaknya fitur- fitur yang tidak relevan atau jika bobot fitur tersebut tidak

setara dengan relevansinya terhadap klasifikasi. Riset terhadap algoritma ini sebagian besar membahas bagaimana memilih dan memberi bobot terhadap fitur agar performa klasifikasi menjadi lebih baik (Aryo, 2012). Berikut ini adalah urutan proses pada algoritma *K-Nearest Neighbors* :

1. Menentukan parameter K (jumlah tetangga paling dekat).
2. Menghitung kuadrat jarak *euclidean* masing-masing obyek terhadap data sampel yang diberikan.
3. Mengurutkan objek-objek tersebut ke dalam kelompok yang mempunyai jarak *euclidean* terkecil.
4. Mengumpulkan kategori Y (klasifikasi *nearest neighbors*).
5. Dengan menggunakan kategori mayoritas, maka dapat diprediksikan nilai *query instance* yang telah dihitung.

KNN merupakan teknik klasifikasi sederhana, tetapi mempunyai hasil kerja yang cukup bagus. Meskipun begitu, KNN juga mempunyai kelebihan dan kekurangan. Beberapa karakteristik KNN adalah sebagai berikut (Eko Prasetyo, 2012) :

1. KNN merupakan algoritma yang menggunakan seluruh data latih untuk melakukan proses klasifikasi (*complete storage*). Hal ini mengakibatkan proses prediksi yang sangat lama untuk data dalam jumlah yang sangat besar. Pendekatan lain adalah dengan menggunakan *mean* data dari setiap kelas, kemudian menghitung jarak terdekat data uji ke *mean* data setiap kelas tersebut. Hal ini memberi keuntungan kerja yang lebih cepat, tetapi hasilnya kurang memuaskan karena model hanya membentuk *hyperplane* tepat di tengah-tengah di antara 2 kelas yang memisahkan 2 kelas (untuk kasus 2 kelas). Semakin banyak data latih, semakin halus *hyperplane* yang dibuat. Ada relasi pertukaran (*trade-off relation*) antara jumlah data latih pada biaya komputasi dengan kualitas batas keputusan (*decision boundary*) yang dihasilkan.
2. Algoritma KNN tidak membedakan setiap fitur dengan suatu bobot seperti pada *Artificial Neural Network (ANN)* yang berusaha menekan

fitur yang tidak mempunyai kontribusi terhadap klasifikasi menjadi 0 pada bagian bobot. KNN tidak memiliki bobot untuk masing- masing fitur.

3. Karena KNN masuk kategori *lazy learning* yang menyimpan sebagian atau semua data dan hampir tidak ada proses pelatihan, KNN sangat cepat dalam proses pelatihan (karena memang tidak ada), tetapi sangat lambat dalam proses prediksi.
4. Hal yang rumit adalah menentukan nilai K yang paling sesuai.
5. Karena KNN pada prinsipnya memilih tetangga terdekat, parameter jarak juga penting untuk dipertimbangkan sesuai dengan kasus datanya. *Euclidean* sangat cocok untuk menggunakan jarak terdekat

#### 2.14 Metode Pengukuran Kesalahan

Menurut (Olson, David, Yong Shi. 2008) tujuan evaluasi percobaan pada klasifikasi yaitu untuk mengukur keefektifan apakah sistem mengklasifikasi secara benar. Pada pengukuran kesalahan biasanya membutuhkan sebuah matriks yang disebut berupa *confusion matrix*. Berikut merupakan tabel *confusion matrix*.

**Tabel 2.1** *confusion matrix*

Kelas Asli \ Kelas Hasil Prediksi	Positif	Negatif
Positif	True Positive (TP)	False Negative (FN)
Negatif	False Positive (FP)	True Negative (TN)

Keterangan **Tabel 2.1** :

1. TP adalah jumlah dari kelas positif (kelas yang mempunyai jumlah data lebih sedikit) yang benar diklasifikasikan.
2. FN adalah jumlah kelas positif yang salah diklasifikasikan kedalam kelas negatif.

3. FP adalah jumlah kelas negatif yang salah diklasifikasikan kedalam kelas positif.
4. TN adalah jumlah kelas negatif yang benar diklasifikasikan.

Berdasarkan data yang didapatkan dari *confusion matrix*, maka akan dihitung (Weng dan Poon 2008) :

#### 1. Akurasi dan Laju Error

Akurasi digunakan untuk mengukur prosentase pengenalan secara keseluruhan dan dihitung sebagai jumlah data uji yang dikenali dengan benar, dibagi dengan jumlah seluruh data uji. Berikut rumus akurasi dan laju *error* pada persamaan 2.4 dan persamaan 2.5.

$$\begin{aligned} \text{Akurasi} &= \frac{\text{Jumlah data yang diprediksi secara benar}}{\text{Jumlah prediksi yang dilakukan}} \\ &= \frac{TP+TN}{TP+FN+FP+TN} \times 100\% \dots \dots \dots (2.4) \end{aligned}$$

$$\begin{aligned} \text{Laju error} &= \frac{\text{Jumlah data yang diprediksi secara salah}}{\text{Jumlah prediksi yang dilakukan}} \\ &= \frac{FN+FP}{TP+FN+FP+TN} \times 100\% \dots \dots \dots (2.5) \end{aligned}$$

### 2.15 Penelitian Sebelumnya

Penelitian sebelumnya dilakukan oleh seorang mahasiswa Universitas Muhammadiyah Gresik yang bernama Mochamad Beni Mashud untuk skripsi yang berjudul “**Aplikasi Pencarian Potensi Akademi Mahasiswa Di Bidang Keahlian Rekayasa Perangkat Lunak Dan Sistem Cerdas Menggunakan *Fuzzy C-Means***”. Dari hasil pengujian diketahui bahwa perbandingan hasil akurasi pada tingkat 2 cluster mendapatkan akurasi sebesar 80% ; tingkat 3 cluster mendapatkan akurasi sebesar 69.17% ; dan untuk tingkat 4 cluster mendapatkan akurasi sebesar 40%. Sehingga dapat disimpulkan bahwa semakin besar nilai

cluster (K) maka presentase konsistensi sistem semakin kecil. Nilai validitas terbaik diperoleh ketika nilai mendekati 0. Hasil terbaik didapatkan ketika nilai cluster (K) sebanyak dua yaitu sebesar 0.3306 untuk sistem cerdas dan 0,8938 untuk rekayasa perangkat lunak.

Selain itu metode *KNN* juga digunakan untuk penelitian seseorang yang bernama Sumarlin mahasiswa Universitas STIKOM Uyelindo, Kupang dengan judul “**Implementasi Algoritma K-Nearest Neighbor Sebagai Pendukung Keputusan Klasifikasi Penerima Beasiswa PPA dan BBM**”. Penelitian ini membahas tentang klasifikasi beasiswa Peningkatan Prestasi Akademik (PPA) dan Bantuan Belajar Mahasiswa (BBM) berdasarkan variabel-variabel yang telah ditentukan dengan menerapkan algoritma *k-nearest neighbor*. Hasil klasifikasi dari sistem ini akan digunakan sebagai keputusan dalam pemberian beasiswa bagi mahasiswa yang mengajukannya. Sehingga dari hasil pengujian diketahui bahwa akurasi yang diperoleh untuk beasiswa Peningkatan Prestasi Akademik (PPA) mencapai 83.33% dan beasiswa Bantuan Belajar Mahasiswa (BBM) mencapai 90%.