

# PENERAPAN ALGORITMA KLASIFIKASI *K-NEAREST NEIGHBOR* (K-NN) UNTUK MEMPREDIKSI UJIAN NASIONAL DENGAN MENGGUNAKAN HASIL NILAI *TRY OUT* SISWA SEKOLAH MENENGAH KEJURUAN BERBASIS DESKTOP PADA SMK BINA INFORMATIKA BINTARO

Rofid Aljabar<sup>1</sup>, Dewi Kusumaningsih<sup>2</sup>

Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur  
Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260  
E-mail : [rofid.aljabar95@gmail.com](mailto:rofid.aljabar95@gmail.com)<sup>1</sup>, [dewi.kusumaningsih@budiluhur.ac.id](mailto:dewi.kusumaningsih@budiluhur.ac.id)<sup>2</sup>

## Abstrak

*Try out* merupakan tahapan gladi bersih menjelang pelaksanaan Ujian Nasional (UN) yang sesungguhnya. Kegiatan tes uji coba kemampuan peserta didik diselenggarakan oleh Tim Musyawarah Kerja Kepala Sekolah (TMKKS) di SMK Bina Informatika Bintaro. *Try out* digunakan untuk menguji kesiapan siswa dalam menghadapi Ujian Nasional (UN). Hasil *Try Out* dapat digunakan oleh siswa untuk mengetahui materi apa yang sudah dikuasai dan yang belum dikuasai. Dari hasil tersebut diharapkan siswa mampu mengejar ketertinggalan terhadap materi yang belum dikuasai. Soal *Try Out* diambil dari materi kelas 10 sebanyak 20%, materi kelas 11 sebanyak 30%, dan materi kelas 12 sebanyak 50%. SMK Bina Informatika setiap tahunnya menyelenggarakan 2 kali *Try Out* untuk siswa/i yang akan menghadapi Ujian Nasional (UN). Mata pelajaran yang diujikan antara lain Matematika, Bahasa Indonesia, Bahasa Inggris, dan Teori Kejuruan. Berdasarkan hasil ujian *Try Out* pihak sekolah menentukan prediksi apakah siswa/i SMK Bina Informatika Bintaro akan lulus atau tidak dalam Ujian Nasional (UN) berdasarkan dari nilai *Try Out* yang pernah dicapai tersebut. Pihak sekolah masih menggunakan dengan cara manual untuk memprediksi nilai UN dengan menggunakan Microsoft Excel. Untuk menjumlah seluruh nilai *Try Out* tersebut dibutuhkan 2-3 hari untuk memprediksi kelulusan UN. Permasalahan tersebut dapat di selesaikan dengan membangun sebuah aplikasi yang dapat membuat sebuah prediksi dengan menggunakan nilai *Try Out* dengan memperhitungkan nilai dengan algoritma *K-Nearest Neighbor* (KNN). Algoritma *K-Nearest Neighbor* merupakan salah satu metode klasifikasi yang menghitung jarak terdekat untuk mengklasifikasikan objek baru berdasarkan kedekatannya dengan objek lama yang sudah diklasifikasikan sebelumnya, dengan menggunakan dataset selama 3 tahun terakhir berjumlah 475 data, rata-rata akurasi yang diperoleh oleh sistem 83.21%. Nilai *K* yang memiliki tingkat akurasi terbesar pada setiap pengujianya adalah *K*=5, yaitu sebesar 87.75 % dengan tingkat sensitivitas sistem (*Recall*) sebesar 88.59% dan nilai presisi 73.72%. Berdasarkan hasil dari serangkaian proses pengujian yang telah dilakukan, dapat disimpulkan sistem ini telah mengklasifikasi dataset siswa/i SMK Bina Informatika Bintaro dengan baik dan cepat dibandingkan dengan proses penentuan hasil prediksi ujian nasional menggunakan nilai ujian *Try Out* yang dilakukan secara manual menggunakan Microsoft excel.

**Kata kunci:** *K-Nearest Neighbor*, Evaluasi, Klasifikasi, *Try Out*, Prediksi

## 1. PENDAHULUAN

*Try Out* merupakan tahapan gladi bersih menjelang pelaksanaan Ujian Nasional (UN) yang sesungguhnya. Kegiatan tes uji coba kemampuan peserta didik atau yang lebih dikenal dengan istilah *Try Out* diselenggarakan oleh tim Musyawarah Kerja Kepala Sekolah (MKKS) di SMK Bina Informatika Bintaro. Ujian *Try Out* pada hakikatnya merupakan evaluasi hasil belajar yang dilaksanakan oleh lembaga pendidikan sebelum menghadapi Ujian Nasional (UN). *Try Out* digunakan untuk menguji kesiapan siswa dalam menghadapi Ujian Nasional (UN). Hasil *try out* dapat digunakan oleh siswa untuk mengetahui materi apa yang sudah dikuasai dan yang belum dikuasai. Dari hasil tersebut diharapkan siswa mampu mengejar ketertinggalan terhadap materi yang belum dikuasai. Soal *try out* diambil dari materi kelas 10, 11, dan 12. Namun persentasenya berbeda, yaitu untuk materi kelas 10 hanya diambil sebanyak

20%, kelas 11 sebanyak 30%, dan dari kelas 12 diambil sebanyak 50% (yang sudah terangkum dalam “SKL”). Meski penyusunan soal diambil dari materi pelajaran kelas 10 sampai dengan 12, tidak menutup kemungkinan ada hal – hal baru yang belum pernah diajarkan oleh guru. Oleh karena itu perlu dilaksanakannya *try out* tahapan berikutnya. SMK Bina Informatika Bintaro merupakan sekolah menengah kejuruan yang memfokuskan pada pendidikan yang berbasis teknologi informasi dengan menggunakan kurikulum yang dilaksanakan dengan mengacu pada Permendikbud No. 61 Tahun 2014, SMK Bina Informatika Bintaro setiap tahunnya menyelenggarakan 2 kali *try out* untuk siswa/i yang akan menghadapi Ujian Nasional (UN). Berdasarkan hasil ujian *try out* pihak sekolah menentukan prediksi apakah siswa/i SMK Bina Informatika Bintaro akan lulus atau tidak dalam Ujian Nasional (UN) berdasarkan dari nilai *try out* yang pernah dicapai

tersebut. Permasalahan tersebut membuat penulis tertarik untuk membangun sebuah aplikasi yang dapat membuat sebuah prediksi dengan menggunakan nilai *try out* dengan memperhitungkan nilai dengan algoritma *K-Nearest Neighbor (KNN)*. Dengan adanya aplikasi ini diharapkan nantinya dapat membantu pihak sekolah dalam memperhatikan siswa – siswanya yang mendapatkan nilai *try out* dibawah standar nilai, dan diharapkan siswa mendapat perhatian lebih oleh pihak sekolah agar siswa tersebut dapat mengikuti dan mengerjakan soal – soal Ujian Nasional (UN) dengan lancar. Pemilihan algoritma *K-Nearest Neighbor (KNN)* ini dikarenakan algoritma tersebut mudah diimplementasikan [1], dan juga termasuk salah satu algoritma yang memiliki kinerja yang intensif ketika diberikan training dataset yang benar (Han, Kamber, & Pei, 2012). Metode pengukuran jarak yang digunakan pada penulisan tugas akhir ini adalah menggunakan metode pengukuran jarak *Manhattan Distance* karena metode ini sesuai apabila digunakan untuk mengukur kedekatan data yang memiliki nilai atribut bersifat numerikal. Algoritma *K-Nearest Neighbor* pada dasarnya akan mengevaluasi kedekatan antara data satu dengan data lainnya, yang kemudian kedekatan inilah yang mengidentifikasi kelas atau label data tersebut (Nursalim, Suprapedi, & H.Himawan, 2014). Metode pengukuran jarak yang digunakan pada penulisan tugas akhir ini adalah dengan menggunakan metode pengukuran jarak *Manhattan Distance*, karena metode ini sesuai apabila digunakan untuk mengukur kedekatan data yang memiliki nilai atribut bersifat numerikal.

## 2. LANDASAN TEORI

### 2.1. Pengertian Kurikulum

Undang – undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional menjelaskan bahwa kurikulum adalah seperangkat rencana dan pengaturan mengenai tujuan, isi, dan bahan pelajaran.

### 2.2. Kecerdasan Tiruan

Kecerdasan tiruan (*artificial intelligence*) adalah suatu proses dimana peralatan mekanik dapat melaksanakan kejadian – kejadian dengan menggunakan pemikiran atau kecerdasan seperti manusia. Kecerdasan tiruan mempelajari bagaimana komputer dapat melakukan sesuatu yang mana manusia dapat melakukannya dengan baik. Kecerdasan tiruan merupakan sub bagian dari ilmu komputer yang merupakan suatu teknik perangkat lunak yang pemrogramannya dengan cara menyatakan data, pemrosesan data, dan penyelesaian masalah. Kegiatan yang dapat dikerjakan dalam kecerdasan tiruan diantaranya adalah :

#### a. Machine Vision

Bertujuan pada pengenalan pola sebagai kegiatan sistem visual atau panca indra manusia.

#### b. Robotics

Pada produksi alat – alat mekanik yang dapat mengendalikan gerak.

#### c. Speech Processing

Bertujuan pada pengenalan dan sintesa pembicaraan manusia.

#### d. Theorem Proving

Merupakan usaha untuk membuktikan secara otomatis masalah – masalah dalam matematika dan logika.

#### e. General Problem Solving

Bertujuan untuk pemecahan masalah yang ditekankan dalam sebuah bahasa formal.

## 2.3. Data Mining

*Data mining* suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan didalam *database*. *Data Mining* adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar [2]. menurut [3] *Data mining* merupakan proses semi otomatis yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengidentifikasi informasi pengetahuan potensial dan berguna yang tersimpan di dalam *database* besar.

## 2.4. Metode Klasifikasi

Klasifikasi adalah proses penemuan model yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan agar dapat digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui, [4]. Model itu sendiri dapat berupa aturan “*if-then-else*”, berupa sebuah pohon keputusan, formula matematis, atau *neural network* [5]. Model yang telah terbentuk pada saat proses pelatihan berlangsung, kemudian dapat digunakan untuk memprediksi atau mengklasifikasikan label kelas data yang belum diketahui label kelasnya. Pembentukan model selama proses pembelajaran membutuhkan suatu algoritma yang lebih dikenal sebagai algoritma pembelajaran. Algoritma yang telah banyak digunakan oleh para peneliti diantaranya adalah *Naive Baiyes*, *K-Nearest Neighbor*, *Decision Tree*, *Support Vector Machine*, *Artificial Neural Network*, dan masih banyak lagi algoritma klasifikasi lainnya. Berdasarkan cara suatu algoritma dalam melakukan pembelajaran, terdapat dua jenis cara pembelajaran pada algoritma klasifikasi, yaitu *eager learner* dan *lazy learner*. *Eager learner* melakukan pembelajaran terdapat data latih (*training dataset*) untuk membangun model klasifikasi yang menggambarkan pola dari data latih tersebut. Model tersebut akan disimpan kedalam memori yang nantinya akan digunakan untuk mengklasifikasikan data yang baru setelah melakukan pembelajaran dan model telah terbentuk, maka selanjutnya tidak melibatkan data pelatihan yang pernah digunakan.

Jenis pembelajaran seperti ini membuat proses klasifikasi berjalan dengan cepat, namun dalam proses pembentukan model membutuhkan waktu yang cukup lama dan memerlukan biaya komputasi yang cukup mahal. Contoh algoritma yang masuk kedalam kategori pembelajaran ini diantaranya adalah *Artificial Neural Network*, *Support Vector Machine*, *Decision Trees*, dan lain sebagainya. Kategori pembelajaran lainnya yaitu *lazy learner* atau yang juga dikenal dengan *instance based learning* yang hanya melakukan sedikit pembelajaran saat data pelatihan dihadirkan, namun melakukan pekerjaan yang lebih saat akan mengklasifikasikan data yang baru. Berdasarkan hal tersebut, tentunya biaya komputasi mahal untuk mengklasifikasikan data yang belum diketahui kelasnya karena pembelajaran terus dilakukan dengan melibatkan data pelatihan. Algoritma *K-Nearest Neighbor (KNN)* dan *Case-Based Reasoning (CBR)* merupakan algoritma yang berada pada kategori pembelajaran *lazy learner* ini [6].

**2.5. Dataset**

Data dalam terminologi statistika adalah kumpulan objek dengan atribut-atribut tertentu, dimana objek tersebut adalah individu berupa data dimana setiap data memiliki sejumlah atribut, [7]. Atribut berpengaruh terhadap dimensi data yang terbentuk, semakin banyak atribut suatu data maka semakin besar pula ukuran dimensi data tersebut. Kumpulan data yang membentuk suatu set data atau dataset dibagi menjadi 3 jenis diantaranya adalah :

- a. Record
  - 1) Matriks Data.
  - 2) Data Dokumen.
  - 3) Data Transaksi.
- b. Graph
  - 1) World Wide Web (WWW).
  - 2) Struktur Molekul.
- c. Ordered Dataset
  - 1) Data Spasial.
  - 2) Data Temporal.
  - 3) Data Sekuensial.
  - 4) Data Urutan Genetik.

Dataset yang berisi sekumpulan data dan data tersebut memiliki nilai atribut, dapat dipandang sebagai suatu vektor dalam ruang multidimensi dimana setiap dimensi mempresentasikan suatu atribut yang berbeda – beda. *Dataset* dengan model tersebut dikenal juga dengan data matriks. Sekumpulan data matriks dapat diinterpretasikan sebagai matriks  $m \times n$ . Nilai  $m$  yang merupakan jumlah baris, satu baris menyatakan suatu data, dan nilai  $n$  yang merupakan jumlah kolom, satu kolom mendefinisikan suatu atribut [7]. Data matriks merupakan jenis *dataset* yang paling umum dan banyak digunakan aplikasi statistik. Contoh data matriks dapat dilihat pada tabel 1.

Tabel 1 : Data Record Matriks

Nomor	Nama	Jenis Kelamin	Mtk 1	Bindo 1	Bing 1	TK 1	Mtk 2	Bindo 2	Bing 2	TK 2	Ket	Hasil
1110001	Aditva Agnesa Putra	L	40	40	58	47	45	68	66	67	LULUS	-
1110002	Aditva Arya Mulya Arsyad	L	45	52	48	38	37	42	74	62	LULUS	-
1110003	Aditva Ikhbal Maulana	L	42	40	42	60	27	52	44	52	LULUS	-
1110004	Alan Taufiquracuman	L	40	54	30	40	40	47	70	55	LULUS	-
1110005	Ali Syahid Ibrahim	L	45	42	74	51	32	40	72	55	LULUS	-
1110006	Amri Aziz Multazam	L	40	60	64	60	40	66	76	62	TIDAK	-

**2.6. Atribut**

Atribut merupakan fitur/karakteristik yang digunakan sebagai parameter data, dalam hal ini dianggap sebagai nilai yang dimiliki oleh atribut. Menurut [1], menyatakan bahwa pada umumnya atribut dapat dibagi menjadi dua jenis, yaitu tipe atribut numerikal (kuantitatif) dan tipe atribut kategorikal (kualitatif). Penjelasananya seperti pada Tabel 2.

Tabel 2 : Jenis Atribut

Tipe Atribut	Deskripsi	Contoh	
Numerikal (Kuantitatif)	Diskrit	Atribut yang bernilai diskrit adalah atribut yang direpresentasikan dengan bilangan <i>integer</i> .	1. Jumlah beberapa anak dalam suatu keluarga. 2. Jumlah kata pada suatu dokumen. 3. Usia seseorang. 4. Kode pos.
	Kontinu	Atribut yang nilainya bersifat kontinu, pada umumnya dihasilkan dari hasil pengukuran. Dan biasa direpresentasikan dengan tipe data <i>float</i> atau <i>double</i> .	1. Berat dan tinggi seseorang. 2. Temperatur suhu ruangan. 3. Tekanan darah seseorang.
Kategorikal (Kualitatif)	Nominal	Tipe atribut kategorikal yang memiliki nilai berupa nama. Nama inilah yang membedakan satu atribut dengan atribut lainnya.	1. Jenis kelamin (laki – laki dan perempuan). 2. Status perkawinan (menikah, lajang, dan cerai).
	Ordinal	Jenis atribut kategorikal dengan nilai atribut yang memiliki tingkatan.	1. <i>Grading</i> nilai (A, B, C, D, dan seterusnya). 2. Status pendidikan (SMP, SMA, S1, dan seterusnya). 3. Tingkatan tinggi badan ( <i>small, medium, high</i> ).

Tipe atribut numerikal dengan nilai diskrit diberlakukan sebagai tipe atribut kategorikal, contohnya kategori usia anak – anak, usia remaja, dan usia dewasa. Proses dalam pengujian algoritma *K-Nearest Neighbor* dipengaruhi oleh tipe atribut yang memiliki hubungan dengan objek yang dianalisis. Berdasarkan hal tersebut, maka secara teknis ada perbedaan metode ketika mempertimbangkan status perkawinan seseorang dalam perbandingannya dengan analisa rekening bank mereka dikarenakan atribut status perkawinan seseorang tersebut memiliki tipe atribut yang bersifat kategorikal, sedangkan atribut rekening bank itu sendiri memiliki tipe atribut yang bersifat numerikal.

**2.7. Algoritma K-Nearest Neighbor (K-NN)**

Algoritma *K-Nearest Neighbor (KNN)* merupakan salah satu algoritma klasifikasi yang cukup populer. Algoritma klasifikasi ini termasuk kedalam algoritma *instance-based classifier* yang

dimana proses klasifikasi baru hanya akan dilakukan ketika ada objek yang ingin diketahui kelasnya, sehingga algoritma tidak membangun suatu model dari data pelatihan melainkan data pelatihan itu sendiri yang merepresentasikan suatu *knowledge* dari model klasifikasi tersebut [8]. *Instance-based classifier* sering juga disebut sebagai metode pembelajaran *lazy learner* sehingga sangat berbeda dengan konsep pembelajaran *eager learner* yang membangun suatu model dari data pelatihan yang kemudian model tersebut akan digunakan untuk mengklasifikasikan objek baru dan kemudian mengabaikan data pelatihannya. Algoritma *K-Nearest Neighbor* juga merupakan jenis algoritma pembelajaran terawasi (*supervised learning*) dimana algoritma *K-Nearest Neighbor* bertujuan untuk menentukan suatu kelas *query instance* yang baru terhadap data-data yang sudah ditentukan atau diketahui kelasnya [9]. Pendekatan yang dilakukan oleh metode *K-Nearest Neighbor (KNN)* ini adalah dengan mengklasifikasikan objek atau *query instance* baru kedalam suatu kelas berdasarkan kedekatan atribut atau fitur dengan “K” buah tetangganya pada data pelatihan [1]. Data pelatihan sebelumnya akan membentuk vektor yang diproyeksikan pada ruang dimensi, dimana masing-masing vektor merepresentasikan atribut atau fitur dari suatu data dan kemudian ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi yang telah dilakukan terhadap data pelatihan [7]. Objek baru yang ingin diketahui label kelasnya akan diukur tingkat kedekatannya dengan data pelatihan, dimana nilai “K” disini merupakan parameter jumlah tetangga atau data pelatihan terdekat yang akan dilibatkan dan mempunyai pengaruh besar terhadap penentuan label kelas objek baru tersebut [7]. Penentuan kelas objek baru, algoritma *K-Nearest Neighbor* akan mengklasifikasikan berdasarkan label kelas yang mendominasi tetangga terdekatnya tersebut. Kelas pada tetangga terdekat yang memiliki jumlah terbanyak akan dianggap sebagai kelas hasil klasifikasi objek baru tersebut. Persamaan untuk mengukur kedekatan antara objek baru dengan data pelatihan yang dapat digunakan pada algoritma *K-Nearest Neighbor* menurut [7], diantaranya adalah :

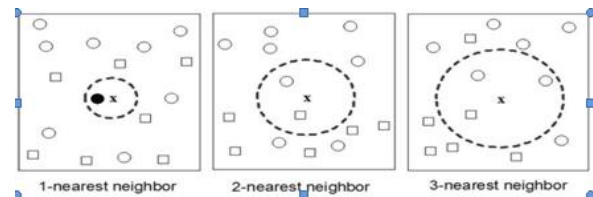
- a. *Euclidian Distance*
- b. *Manhattan Distance*
- c. *Square Euclidian Distance*
- d. *Cosine Distance*
- e. *Correlation Distance*
- f. *Hamming Distance*

Alur kerja dari algoritma *K-Nearest Neighbor* menurut [10] adalah sebagai berikut :

- 1) Tentukan nilai K, sebagai parameter ketetanggaan terdekat.
- 2) Hitung jarak antara *query instance* dan semua data pelatihan.
- 3) Urutkan berdasarkan jarak terkecil.

- 4) Tentukan label kelas *query instance* berdasarkan label mayoritas pada K objek data latih terdekat.

Cara kerja dari algoritma *K-Nearest Neighbor* dapat diilustrasikan pada gambar dibawah ini dimana pada gambar terdapat beberapa contoh kasus ketetanggaan terdekat antara objek baru dengan objek lainnya yang sudah diketahui kelasnya. Contoh, berdasarkan 3 ketetanggaan terdekat pada gambar didapati dua buah kelas yang telah terklasifikasi (data pelatihan) didalam (Gambar 1) disimpulkan bahwa objek merupakan kelas lingkaran.



Gambar 1 : Ilustrasi Cara Kerja K-NN

## 2.8. Manhattan Distance

*Manhattan* atau *City Distance* digunakan untuk mengambil kasus yang cocok dari basis kasus dengan menghitung jumlah bobot *absolute* dari perbedaan antara kasus yang sekarang dan kasus yang lain dalam basis kasus.

## 2.9. Confusion Matrix

Sebuah sistem yang melakukan proses klasifikasi diharapkan dapat menentukan semua kelas data input dengan benar, namun tidak dapat dipungkiri bahwa kinerja suatu sistem tidak dapat seratus persen benar, sehingga perlu adanya metode untuk mengukur unjuk kerja algoritma klasifikasi tersebut. *Confusion Matrix* merupakan salah satu metode yang umum digunakan sebagai metode evaluasi untuk mengetahui unjuk kerja model klasifikasi yang didasarkan kepada tingkat hasil akurasi yang dihasilkan [7].

## 2.10. Proses Data Mining

Dalam *data mining* terdapat proses dan tahapan – tahapan yang setidaknya harus dilalui terhadap data yang ingin digali pengetahuannya dari gudang data [1], proses tersebut diantaranya adalah :

- a. *Data cleansing*

Bisa disebut dengan pembersihan data adalah proses untuk membersihkan data dari gangguan (*noise*) dan ketidak konsistenan data. Dalam proses ini juga dilakukan pembersihan terhadap duplikasi data.

- b. *Data Integration*

Proses dimana data dikumpulkan dari beberapa sumber data yang berbeda menjadi satu



basis data baru yang terpisah dari basis data operasional.

c. *Data Selection*

Proses untuk memilih data yang dianggap relevan dari database untuk dianalisis.

d. *Data Transformation*

Merupakan proses dimana data ditransformasi ke dalam bentuk yang sesuai tergantung pada jenis atau pola informasi yang ingin ditemukan dengan melakukan ringkasan atau operasi agregasi.

3. ANALISA DAN PERANCANGAN PROGRAM

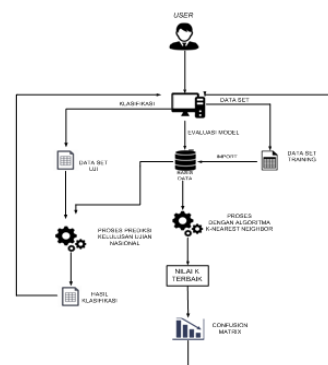
3.1. Skema Proses Aplikasi

Aplikasi yang dibuat oleh penulis pada Tugas Akhir ini adalah aplikasi berbasis *desktop*, yang dikembangkan dengan bahasa pemrograman *java*. *Data Mining* merupakan proses pengolahan data dari data yang dimiliki sebelumnya sehingga dapat digunakan untuk sebuah hasil prediksi. Proses aplikasi ini dapat diuraikan sebagai berikut :

- a. *User* meng-import file dataset (berupa file spreadsheet berekstensi \*.xls atau \*.xlsx) ke database melalui menu dataset, file dataset ini dijadikan model pembelajaran algoritma *K-Nearest Neighbor*.
- b. *User* melakukan evaluasi atau pengujian dengan menu evaluasi dengan data berdasarkan dataset dan dibagi menjadi 2 bagian yaitu *training* dataset dan *testing dataset*.
- c. Memilih metode pengujian, metode dibagi menjadi 2 yakni *Single Testing* dan *Multi Testing*.
- d. Menginput nilai k, untuk membandingkan tetangga terdekat pada parameter.
- e. Setelah *user* menginput nilai K, sistem akan melakukan proses pembelajaran dan proses pengklasifikasian terhadap dataset dan akan menghasilkan hasil evaluasi dengan metode *Confusion Matrix*.
- f. Setelah mendapatkan nilai K terbaik, *user* harus melakukan proses pengklasifikasian pada menu klasifikasi untuk menentukan prediksi kelulusan ujian nasional siswa.
- g. Untuk proses klasifikasi, *user* terlebih dahulu harus memilih file spreadsheet yang berisi data siswa yang belum dan ingin diketahui kelulusan Ujian Nasionalnya.
- h. Kemudian *user* menginput nilai K sebagai pembanding tetangga terdekat parameter.
- i. Setelah menginput nilai K, sistem akan melakukan proses klasifikasi terhadap model yang tersimpan di basis data.

j. Lalu ketika proses data selesai, data dapat disimpan dalam bentuk file.

Skema proses keseluruhan aplikasi dapat dilihat pada gambar 2 berikut dibawah ini :



Gambar 2 : Skema Proses Keseluruhan Aplikasi

3.2. Perangkat yang Digunakan

Pada tahap kebutuhan perangkat ini dilakukan pengumpulan kebutuhan – kebutuhan dari semua elemen sistem perangkat yang akan digunakan dalam pembuatan program. Adapun perangkat keras dan perangkat lunak yang digunakan antara lain sebagai berikut :

a. Perangkat Keras (Hardware)

Perangkat keras komputer yang digunakan dalam pengembangan sistem ini terdiri dari satu buah laptop Asus A455L dengan spesifikasi sebagai berikut :

- 1) Prosesor : Processor Intel®™ i3-4030U CPU 1.9 GHz.
- 2) Memori : 4GB DDR3.
- 3) Media Penyimpanan : Hard Drive 500GB data.

b. Perangkat Lunak (Software)

Perangkat lunak yang digunakan terdiri dari perangkat lunak untuk mengembangkan aplikasi dan algoritma kriptografi. Spesifikasi perangkat lunak adalah sebagai berikut :

- 1) Sistem Operasi : Windows 10 x64 bit
- 2) IDE : Netbeans IDE 8.2
- 3) Java Platform : JSE Development Kit (JDK 8u102-windows x64)
- Database : SQLite Studio

3.3. Spesifikasi Basis Data

a. Tabel Data Keseluruhan

Menjelaskan tabel data yang digunakan untuk menampung atribut data kelulusan. Atribut tersebut dapat dilihat pada tabel 3 dibawah ini :

Tabel 3 : Atribut Data Keseluruhan

No.	Nama Field	Tipe Data	Panjang	Keterangan
1.	tahun_ajaran	Varchar	10	Tahun Ajaran
2.	Nis	Varchar	9	NIS
3.	nama	Varchar	50	Nama
4.	Jenis_Kelamin	Char	1	Jenis Kelamin
5.	Mtk1	Double		Nilai <i>Tryout</i> pertama Matematika
6.	Bindo1	Double		Nilai <i>Tryout</i> pertama Bahasa Indonesia
7.	Bing1	Double		Nilai <i>Tryout</i> pertama Bahasa Inggris
8.	TK1	Double		Nilai <i>Tryout</i> pertama Teori Kejuruan
9.	Mtk2	Double		Nilai <i>Tryout</i> kedua Matematika
10.	Bindo2	Double		Nilai <i>Tryout</i> kedua Bahasa Indonesia
11.	Bing2	Double		Nilai <i>Tryout</i> kedua Bahasa Inggris
12.	TK2	Double		Nilai <i>Tryout</i> kedua Teori Kejuruan
13.	Keterangan			Keterangan hasil ujian <i>Tryout</i>
14.	Hasil			Keterangan hasil Ujian Nasional

### 3.4. Sumber Data

Pengumpulan data dalam penelitian ini dilakukan dengan memperoleh data primer dan data sekunder. Data primer pada kasus ini diperoleh dari data nilai *Tryout* siswa SMK Bina Informatika Bintaro. *Dataset* yang digunakan adalah data siswa yang sudah melaksanakan ujian *Tryout* dan Ujian Nasional selama tiga tahun terakhir, detail dari *dataset* dapat dilihat pada tabel 4 dibawah ini :

Tabel 4 : Data Penelitian

No	Tahun Ajaran	Jumlah Data
1.	2013/2014	169
2.	2014/2015	159
3.	2015/2016	147

### 3.5. Proses Prediksi dan Klasifikasi

#### a. Pembersihan Data

Pada proses ini dilakukan pembersihan terhadap data yang dianggap sebagai *outlier* atau data yang mengandung *noise*. Kriteria data yang dianggap sebagai *outlier* diantaranya :

- 1) Data nilai *Tryout* yang bernilai 0 atau *null*.
- 2) Duplikasi Data.

#### b. Pemilihan Data

Proses ini bertujuan untuk memilih *attribute* yang tepat untuk proses analisis data. Pemilihan *attribute* ini didasari oleh hasil diskusi dengan guru Bidang Kurikulum SMK Bina Informatika Bintaro yang mendapatkan keputusan terkait *attribute* yang dipakai untuk kelulusan Ujian Nasional (UN) SMK Bina Informatika Bintaro. *Attribute* yang didapat diantaranya :

- 1) Nilai *Tryout* Pertama Matematika.
- 2) Nilai *Tryout* Pertama Bahasa Indonesia.
- 3) Nilai *Tryout* Pertama Bahasa Inggris.
- 4) Nilai *Tryout* Pertama Teori Kejuruan.
- 5) Nilai *Tryout* Kedua Matematika.
- 6) Nilai *Tryout* Kedua Bahasa Indonesia.
- 7) Nilai *Tryout* Kedua Bahasa Inggris.
- 8) Nilai *Tryout* Kedua Teori Kejuruan.
- 9) Keterangan Hasil *Tryout*.

Keterangan Hasil Ujian Nasional (UN).

## 4. IMPLEMENTASI DAN UJIAN COBA PROGRAM

### a. Tampilan Form Layar Utama

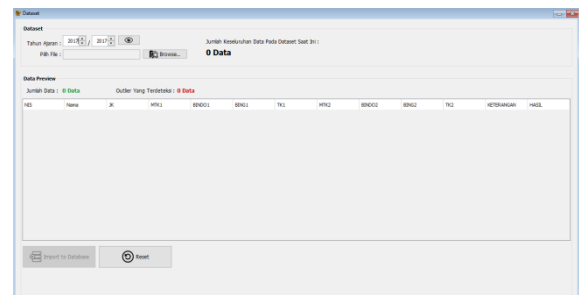
Menu utama adalah tampilan layar yang ditampilkan pertama kali kepada *user*. Tampilan layar ini terdapat beberapa menu yaitu ada menu *data*, *process*, *about*, dan *option*. Tampilan layar form utama dapat dilihat pada gambar 3 berikut ini :



Gambar 3 : Tampilan Layar Form Utama

### b. Tampilan Layar Form Dataset

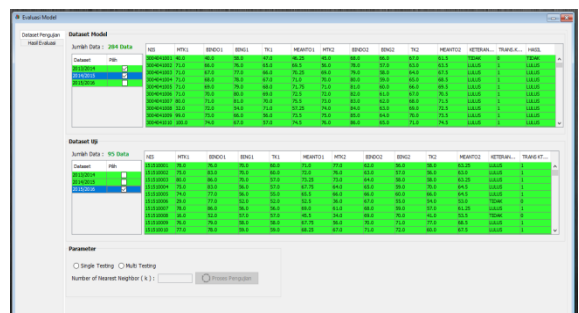
Menu form *dataset* ini akan muncul saat tombol menu data dan memilih menu *dataset* pada Menu Utama. Form *dataset* dapat dilihat pada gambar 4 berikut ini :



Gambar 4 : Tampilan Form Dataset

### c. Tampilan Layar Form Evaluasi

Menu form Evaluasi ini akan muncul saat tombol menu *process* di pilih. Form Evaluasi dapat dilihat pada gambar 5 berikut ini :

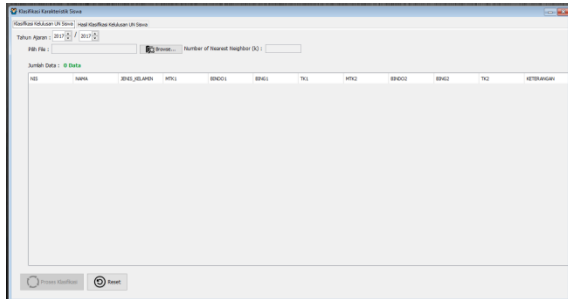


Gambar 5 : Tampilan Form Evaluasi

**d. Tampilan Layar Form Klasifikasi**

2015.

Menu form Klasifikasi ini akan muncul saat tombol menu *process*. Form Klasifikasi dapat dilihat pada gambar 6 berikut ini :



Gambar 6 : Tampilan Form Klasifikasi

**5. KESIMPULAN**

Berdasarkan hasil penelitian Tugas Akhir (TA) ini, maka dapat disimpulkan bahwa menggunakan algoritma *K-Nearest Neighbor (K-NN)* dengan metode evaluasi dan klasifikasi dengan pengukuran jarak menggunakan *Manhattan Distance* dapat diimplementasikan ke dalam proses prediksi untuk kelulusan Ujian Nasional Siswa/i.

**6. DAFTAR PUSTAKA**

- [1] F. Gorunescu, *Data mining: concepts and techniques*. 2011.
- [2] M. Mustakim and G. Oktaviani, "Algoritma K-Nearest Neighbor Classification Sebagai Sistem Prediksi Predikat Prestasi Mahasiswa," *J. Sains dan Teknol. Ind.*, vol. 13, no. 2, pp. 195–202, 2016.
- [3] E. Turban, J. E. Aronson, and T.-P. Liang, "Decision Support Systems and Intelligent Systems," *Decis. Support Syst. Intell. Syst.*, vol. 7, p. 867, 2007.
- [4] H. Leidiyana, "Penerapan Algoritma K-Nearest Neighbor Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor," *J. Penelit. Ilmu Komputer, Syst. Embed. Log.*, vol. 1, no. 1, pp. 65–76, 2013.
- [5] A. Nugroho, "Implementasi Algoritma K-Nearest Neighbor Dalam Memprediksi Potensi Calon Kreditur Di Ksp Galih Manunggal," *J. Ilm. DASI (Data Manaj. dan Teknol. Informasi)*, 2015.
- [6] Rusdy Anha, "Pengertian Perancangan, Konser dasar sistem, Karakteristik Sistem, Klasifikasi sistem ~ Rusdyanha," *12 February 2017*, 2017. .
- [7] Nursalim, Suprapedi, and H.himawan, "Klasifikasi Bidang Kerja Lulusan Menggunakan Algoritma K-Nearest Neighbor," *Teknol. Inf.*, vol. 10, no. April, pp. 31–43, 2014.
- [8] M. E. I. Lestari, "Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) untuk Mendeteksi Penyakit Jantung," *Fakt. Exacta*, vol. 7, no. 4, pp. 366–371, 2014.
- [9] H. Risman, D. Nugroho, and Y. Retno, "Penerapan Metode K-Nearest Neighbor Pada Aplikasi Penentu Penerima Beasiswa Mahasiswa Di Stmik Sinar Nusantara Surakarta," *TIKOMSiN*, pp. 19–25, 2012.
- [10] F. Liantoni, "Klasifikasi Daun Dengan Perbaikan Fitur Citra Menggunakan Metode K-Nearest Neighbor," *Ultimatics*, vol. VII, no. 2, pp. 98–104,