IMPLEMENTASI ALGORITMA *K-NEAREST NEIGHBOR* DENGAN METODE PENGUKUR JARAK *EUCLIDEAN DISTANCE* UNTUK MEMPREDIKSI KELULUSAN UN BERDASARKAN NILAI *TRYOUT* BERBASIS *JAVA DESKTOP* PADA SMK PRIMA UNGGUL

Makhdum Rosadi[1], Dr. M. Syafrullah, M.Kom, M.Sc[2]
[1]*Teknologi Informasi, Fakultas Teknologi Informasi, Universitas Budi Luhur*
[1,2]*Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260*
E-mail: 1311503047@student.budiluhur.ac.id[1], mohammad.syafrullah@budiluhur.ac.id[2]

Abstrak

Perkembangan teknologi dan informasi yang begitu pesat seperti saat ini, komputer dibekali berbagai kemampuan canggih untuk mempermudah pekerjaan manusia yang pada umumnya masih menggunakan tenaga manusia/manual. Prediksi suatu kasus salah satunya dengan menerapkan ilmu algoritma K-Nearest Neighbor dengan metode klasifikasi dan evaluasi. Faktor untuk menentukan kualitas pendidikan tinggi salah satunya berdasarkan kemampuan siswa untuk menyelesaikan pendidikan dengan nilai yang memuaskan. SMK Prima Unggul merupakan salah satu Sekolah Menengah Kejuruan Swasta yang ada di Kota Tangerang Selatan, pihak sekolah mengadakan ujian tryout sebanyak dua (2) kali untuk melakukan persiapan menjelang ujian nasional (UN) sebagai langkah untuk para siswa dalam mempersiapkan kematangan materi menjelang UN. Mata pelajaran yang diujikan pada semua jurusan, antara lain, Bahasa Indonesia, Bahasa Inggris, Matematika, Ilmu Pengetahuan Sosial. Pihak sekolah menjadikan ujian tryout sebagai tolak ukur nilai para siswanya untuk memprediksi kelulusan ujian nasional. Nilai tryout sekolah memiliki standar kelulusan masing-masing pada tiap mata pelajaran yang diujikan, siswa yang salah satu nilai tryoutnya mendapat nilai dibawah standar, maka diprediksi tidak lulus UN. Prediksi nilai UN masih dilakukan pihak sekolah menggunakan cara manual menggunakan Microsoft excel, yaitu dengan mengumpulkan semua nilai tryout para siswa dan menjumlah seluruh nilai tersebut yang memerlukan waktu hingga 4-5 hari untuk memprediksi kelulusan UN. Berdasarkan hal tersebut, maka pelaksanaan prediksi nilai UN siswa menjadi kurang efesien dan efektif mengingat rata-rata siswa disekolah SMK Prima Unggul yang melaksanakan UN tiap tahunnya mencapai 100 siswa/lebih. Menggunakan algoritma K-Nearest Neighbor dengan metode pengukuran jarak Euclidean Distance untuk mengklasifikasi siswa ke dalam dua hasil kelulusan yaitu Lulus dan Tidak dengan atribut yang mempengaruhinya, antara lain, nilai tryout seluruh mata pelajaran, dan hasil tryout tiap siswa. Siswa selama 3 tahun terakhir berjumlah 382 data, rata-rata akurasi yang diperoleh oleh sistem 80,00 %. Nilai K yang memiliki tingkat akurasi terbesar pada setiap pengujiannya adalah K=5, yaitu sebesar 100% dengan nilai presisi 100% dan tingkat sensitivitas sistem (Recall) sebesar 100%. SMK Prima Unggul Kota Tangerang Selatan dengan baik dan cepat dibandingkan dengan proses penentuan hasil prediksi ujian nasional menggunakan nilai ujian tryout yang dilakukan secara manual menggunakan Microsoft excel.

Kata kunci: *Euclidean Distance*, Evaluasi *K-Nearest* Neighbor, Klasifikasi, UN, *tryout*, prediksi

1. PENDAHULUAN

Perkembangan teknologi dan komunikasi yang begitu pesat, komputer mempunyai berbagai fitur yang canggih untuk mempermudah pekerjaan manusia yang pada umumnya masih menggunakan tenaga manusia. Perkembangan teknologi diera globalisasi ini dapat memungkinan sebagian pekerjaan manusia dapat digantikan oleh komputer yang tentunya disesuaikan dengan kebutuhan manusia.

Sistem yang dibuat oleh manusia bertujuan untuk meningkatkan produktivitas, meringankan pekerjaan manusia dan menghemat waktu dalam menjalankan pekerjaan yang membutuhkan waktu yang lebih lama. Sekolah SMK Prima Unggul adalah salah satu sekolah menengah kejuruan. Sekolah SMK Prima Unggul adalah salah satu sekolah menengah kejuruan di Provinsi Banten yang sudah menerapkan Kurikulum 2013 ini dalam sistem pendidikannya. Sekolah SMK Prima Unggul setiap tahunnya menyelenggarakan 2 kali ujian *TryOut* untuk siswa/i yang akan menghadapi Ujian Nasional. Berdasarkan hasil ujian *TryOut* pihak sekolah menentukan prediksi Lulus atau Tidak siswa/i dalam Ujian Nasional berdasarhan nilai ujian *TryOut* yang pernah dicapai tersebut.

2. LANDASAN TEORI

2.1. Kecerdasan Tiruan

Kecerdasan tiruan adalah merupakan sub bagian dari sebuah ilmu komputer yang merupakan dalam suatu teknik perangkat lunak yang berjalan dipemrogramannya dengan cara menyatakan sebuah data, pemrosesan data tersebut, dan penyelesaian masalahnya adalah sebagai berikut kegiatan yang dapat dikerjakan dalam suatu kecerdasan tiruan tersebut.

2.2. Atribut

Menurut Gorunescu (2011), menyatakan bahwa pada umumnya atribut ini kita dapat dibagi menjadi dua buah jenis, yaitu tipe atribut numerikal (kuantitatif) dan juga tipe atirbut kategorikal (kualitatif) ini.

2.3. Algoritma K-Neirest Neighbor (KNN)

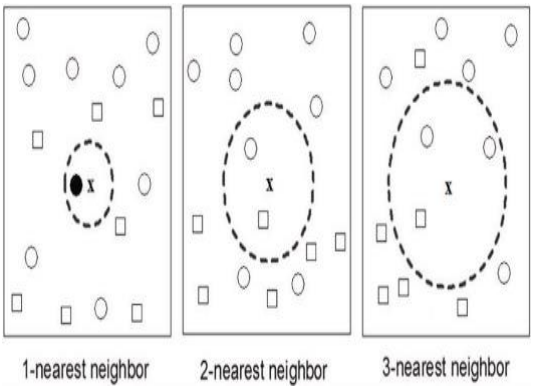
Algoritma *K-Nearest Neighbor* ialah salah satu algoritma ini klasifikasi yang cukup sangat populer. Algoritma klasifikasi ini masuk kedalam algoritma *instance-based classifier* yang dimana proses tersebut diklasifikasi dengan baru hanya akan dilakukan ketika ada objek yang ingin kita ketahui kelasnya, sehingga algoritma ini tidak dapat membangun suatu model dari data pelatihan melainkan data pelatihan itu sendiri dan sedangkan yang merepresentasikan suatu *knowledge* dari model klasifikasi tersebut.

Objek baru ini yang ingin kita ketahui label kelasnya tersebut akan diukur tingkat kedekatannya dengan menggunakan data pelatihan, dimana nilai "K" ini adalah merupakan parameter jumlah tetangga atau data pelatihan terdekat yang akan dilibatkan dan juga mempunyai pengaruh besar terhadap penentuan label kelas objek ini baru tersebut (Indriyanto et al., 2014). Kelas pada tetangga terdekat yang hanya memiliki sejumlah terbanyak akan dianggap sebagai kelas yang hasil klasifikasi objek baru tersebut. Persamaan untuk mengukur kedekatan antara objek baru dengan objek data pelatihan yang dapat digunakan pada suatu algoritma *K-Nearest Neighbor* menurut Indriyanto et al. (2014) dan Sillueta (2016).

Alur kerja dari algoritma *K-Nearest Neighbor* menurut Sukma et al. (2014) ialah sebagai berikut:

a. Tentukan nilai K ini sebagai parameter ketetanggan terdekat tersebut.
b. Hitung sebuah jarak antara *query instance* dan semua data pelatihan.
c. Urutkan data berdasarkan jarak yang terkecil.
d. Tentukan sebuah label kelas *query instance* berdasarkan label mayoritas pada K objek data dan dilatih terdekatnya.

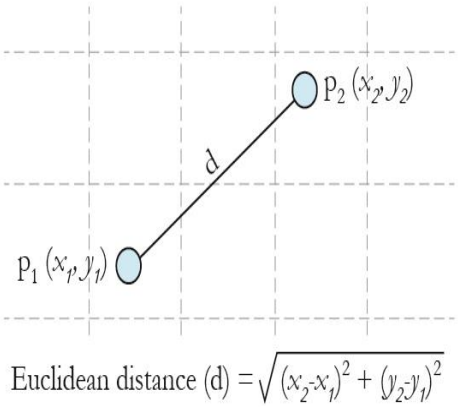
Algoritma *K-Nearest Neighbor* dapat diilustrasikan pada gambar di bawah ini terdapat beberapa contoh kasus ketetanggan terdekat objek baru dengan objek lain yang sudah diketahui kelasnya., contoh, berdasarkan 3 ketetanggan terdekat pada gambar didapati dua buah kelas yang telah terklasifikasi (data pelatihan), kelas lingkaran dan kelas persegi, saat ada objek baru yang ingin diketahui kelasnya maka objek tersebut akan mengukur kedekatannya dengan semua objek yang telah diketahui kelasnya, dan selanjutnya akan mengambil sejumlah 3 objek yang paling besar tingkat kedekatannya dengan objek. Kelas yang paling mendominasi di ruang lingkup ketetanggan tersebut akan menjadi kelas objek tersebut, berdasarkan buku Wu et al. (2008) terdapat beberapa hal yang menjadi isu penting dalam pengaruhnya terhadap kinerja dari algoritma *K-Nearest Neighbor* ini diantaranya yaitu pemilihan nilai K sebagai parameter ketetanggan terdekat, apabila nilai K terlalu kecil hasil klasifikasi akan rentan terhadap *noise*, namun apabila nilai K juga terlalu besar maka suatu ketetanggaan terdekat dapat mencakup kriteria atau atribut yang harusnya dimiliki kelas lain. Kemudian isu lainnya adalah pendekatan *K-Nearest Neighbor* yang memberikan label kelas dengan menentukan dari hasil voting yang terbanyak, hal ini memungkinkan tetangga terdekat yang menentukan kelas objek memiliki variasi jarak yang berbeda.



Gambar 1 Ilustrasi Cara Kerja Algoritma K-Neirest Neighbor (Gorunescu 2011**)**

2.4. *Euclidean Distance*

Euclidean Distance adalah fungsi heuristic Tetapi sebelum mendapatkan hasil kedua titik harus di representasikan ke dalam koordinat 2 dimensi (x, y). Dua buah titik p1 = (x1, y1) dan p2 = (x2, y2) menjadi persamaan berikut (rumus *Euclidean*)



Gambar 2 *Euclidean Distance*

2.5. *Confusion Matrix*

Proses klasifikasi yang diharapkan dapat kita menentukan semua kelas data input dengan benar, namun tidak dapat kita pungkiri bahwa sebuah kinerja suatu system ini tidak dapat seratus persen benar, sehingga perlu adanya salah satu metode untuk mengukur sebuah unjuk kerja algoritma klasifikasi tersebut (Lukman & Labellapansa, 2015). *Confusion Matrix* memberikan data nilai kinerja model klasifikasi ini berdasarkan perhitungan suatu objek yang diklasifikasikan dengan benar dan objek ini yang akan diklasifikasi, yang mana hasil perhitungan ini ditabulasikan kedalam sebuah tabel, seperti yang digambarkan pada tabel 3.

Tabel 3. Tabel *Confusion Matrix* (Han et al,.2013)

		Predicted class		Total
		YES	**NO**	
Actuall Class	**YES**	TP	FN	P
	NO	FP	TN	N
Total		P'	N'	P+N

Keterangan:

a. P (*Positive*): Pada jumlah kelas yang positif (*YES*)
b. N (*Negative*): Pada jumlah kelas yang negatif (*NO*)
c. TP (*True Positive*): Pada jumlah tupel positif yang terklasifikasi dengan benar
d. TN (*True Negative*): Jumlah tupel negatif yang terklasifikasi dengan benar
e. FP (*False Positive*): Pada jumlah tupel positif yang terklasifikasi dengan tidak benar
f. FN (*False Negative*): Pada jumlah tupel negative yang terklasifikasi dengan tidak benar

Berdasarkan Matrix tersebut dapat diperoleh beberapa pengukuran evaluasi diantaranya *accuracy*, *recall* (*sensitivity*), dan *precision*, untuk menghitung nilai *accuracy* digunakan persamaan (2.1), nilai *recall* menggunakan persamaan (2.2), dan nilai *precision* pada persamaan (2.3)

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad (2.1)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2.2)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2.3)$$

Accuracy merupakan persentase dari *dataset* tes yang akan diklasifikasikan dengan benar oleh *classifier*, sedangkan menurut dari hasugian (2014) *recall* berhubungan dengan sebuah rasio kemampuan *classifier* dalam memperoleh data kelas yang relevan dan benar, dan *Precision* adalah ukuran suatu tingkat ketepatan *classifier* untuk kita tidak mengklasifikasikan kelas kedalam suatu kelas yang kita tidak tepat atau salah.

2.6. Proses Data Mining

Dalam *data mining* ini terdapat sebuah proses dan tahapan berikut, menurut Prasetyo (2012) dalam Lukman & Labellapansa (2015) dipaparkan tahapan-tahapan ini yang akan setidaknya harus dilalui terhadap sebuah data yang ingin kita digali pengetahuannya dari sebuah gudang data, proses tersebut ini melalui diantaranya adalah:

A. *Pattern evaluation* merupakan tahapan dimana pola atau model yang terbentuk dianalisis serta dievaluasi. Dimana model yang memiliki tingkat akurasi yang baik dan paling mewakili pengetahuan adalah model yang dipilih untuk nantinya digunakan lebih lanjut.
B. *Knowledge presentation* adalah teknik penyajian atau visualisasi pengetahuan yang telah diperoleh dari proses *mining* data kepada pengguna.

3. Analisa Dan Perancangan Program

3.1. Permasalahan dan Strategi Pemecahan Masalah

3.1.1. Analisa Masalah

SMK Prima Unggul adalah salah satu sekolah menengah kejuruan di Pondok Aren Provinsi Banten yang baru saja menerapkan Kurikulum 2013 dalam sistem pendidikannya. Dalam kurikulum ini Ujian Nasional (UN) memang sudah bukan acuan utama untuk kelulusan siswa/i SMK Prima Unggul, namun Ujian Nasional (UN) masih dijadikan acuan untuk mempetahankan akreditasi berdasarkan jumlah dan nilai kelulusan ujian nasional. Dari nilai *TryOut* tersebut sekolah menentukan siswa/i mana yang mempunyai peluang besar untuk lulus pada Ujian Nasional. Namun dalam penentuannya pihak sekolah masih memakai cara manual menggunakan *Microsoft Exel* sehingga kurang efektif dan efisien. Oleh karena itu pihak sekolah membutuhkan sistem untuk melakukan prediksi Lulus atau Tidak Siswa/i dengan teknik klasifikasi. Pengklasifikasian ini dilakukan berdasarkan parameter-

parameter yang mendefinisikan kemampuan akademik dari siswa/i tersebut.

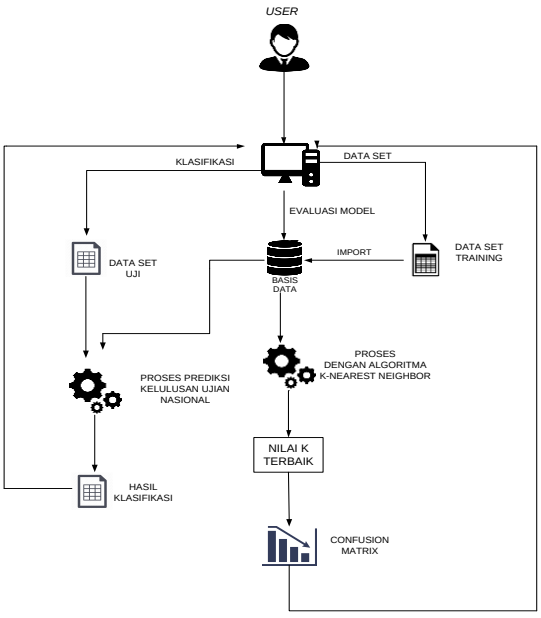
3.1.2. Strategi Pemecahan Masalah

Permasalahan yang telah dijelaskan yang ada di atas dapat kita atasi dengan cara menggunakan sistem yang mampu memprediksi kelulusan Ujian Nasional (UN) dengan mengklasifikasikan siswa yang ada dengan mengimplementasikan algoritma *K-Nearest Neighbor* sebagai metode klasifikasinya dan *Euclidean Distance* sebagai pencarian terdekatnya. Dalam kasus ini penggunaan algoritma *K-Nearest Neighbor* dengan metode klasifikasi menggunakan pencarian terdekat *Euclidean Distance* dianggap mampu menyelesaikan masalah yang ada di SMK Prima Unggul. Pola atau model yang terdapat dari data nilai *TryOut* selanjutnya akan diolah oleh algoritma *K-Nearest Neighbor* untuk mengklasifikasikan data baru (data yang belum diketahui target *class*-nya) dengan melakukan perhitungan kedekatan jarak antara data baru tersebut dengan semua data yang membentuk suatu pola prediksi (data yang sudah diketahui target *class*-nya). Kedekatan inilah yang akan mengindikasikan target class dari data baru tersebut. Dengan diterapkannya algoritma *K-Nearest Neighbor* ini diharapkan dapat membantu sekolah untuk memprediksi siswa/i dapat lulus Ujian Nasional (UN) dengan baik.

3.2. Skema Proses Keseluruhan Aplikasi

Aplikasi yang telah dibuat oleh penulis adalah aplikasi yang berbasis *desktop,* yang telah dikembangkan dengan menggunakan bahasa pemrograman Java. *Data Mining* ini merupakan proses pengolahan data dari data yang telah dimiliki seblumnya sehingga dapat kita gunakan untuk sebuah hasil prediksi. Proses keseluruhan aplikasi ini dapat diuraikan sebagai berikut:

a. *User* akan meng-*import* yang berada di dataset yang berupa *file spreadsheet* berekstensi *.xls atau *.xlsx ke *database* yang melalui menu *dataset*, file *dataset* ini akan dijadikan sebuah model pembelajaran algoritma K-Nearest Neighbor.
b. *User akan* melakukan evaluasi atau pengujian dengan sebuah menu evaluasi dengan data berdasarkan *dataset* ini.
c. Memilih metode pengujian, metode dibagi menjadi dua yakni *Single Testing* dan *Multi Testing*
d. Meng-*input* data nilai K, untuk membandingkan suatu tetangga terdekat pada sebuah parameter
e. Setelah *user* meng-*input* nilai K, sistem akan melakukan proses pembelajaran dan proses pengkalisifikasian terhadap dataset dan akan menghasilkan hasil evaluasi dengan metode *Confusion Matrix*
f. Setelah mendapatkan nilai K terbaik, *user ini* harus dapat melakukan proses pengklasifikasian pada suatu menu klasifikasi untuk menentukan suatu prediksi kelulusan ujian nasional siswa
g. Dan juga untuk melakukan suatu proses ini akan diklasifikasi user terlebih dahulu dan harus memilih suatu *file spreadsheet* yang data ini akan berisi data siswa yang belum dan jika ingin diketahui kelulusan ujian nasionalnya
h. Kemudian *user* akan meng-*input* nilai K sebagai pembanding tetangga terdekat parameter tersebut
i. Setelah meng-*input* nilai K, system akan melakukan proses klasifikasi terhadap model yang tersimpan dibasis data
j. Lalu ketika proses telah selesai, data dapat disimpan dalam bentuk *file spreadsheet* untuk digunakan kembali sebagai pertimbangan selanjutnya
k. Skema proses keseluruhan aplikasi tersebut dapat kita lihat pada suatu gambar 3 berikut di bawah ini:



Gambar 3 Skema Keseluruhan Proses Aplikasi

3.3. Analisa Perangkat

Pada tahap analisa kebutuhan perangkat dilakukan data pengumpulan kebutuhan-kebutuhan dari semua elemen sistem perangkat yang kita akan gunakan dalam pembuatan program ini. Adapun perangkat keras dan

perangkat lunak yang kita gunakan antara lain adalah sebagai berikut:

a. Perangkat Keras

Perangkat keras computer ini yang digunakan ke dalam mengembangkan sistem ini terdiri dari satu buah laptop dengan spesifikasi sebagai berikut ini:

1) *Processor* : AMD Radeon R2 Graphics ~(1,4Ghz)
2) *Memory* : 16GB DDR3L
3) *RAM* : 2048MB

b. Perangkat Lunak (*Software*)

Perangkat lunak yang akan digunakan ini adalah untuk mengembangkan system dengan sebagai berikut:

1) *OS* : *Windows 7 Ultimate (64-bit)*
2) *IDE : NetBeans IDE 8.2*
3) *Java Platform: JSE Development Kit (JDK 8u102-windows x64).*
4) *Database : SQLite Studio*
5) *Library : Apache POI 3.14, JCalendar 1.4, JFreeChart 1.01*

4. Implementasi Dan Uji Coba Program

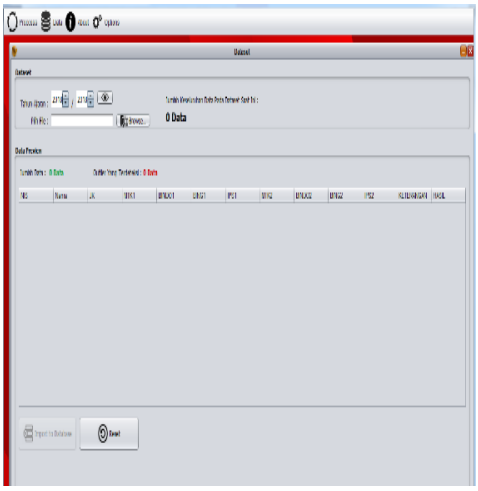
a. Tampilan Form Menu Utama

Menu Utama adalah suatu tampilan layar yang dapat mentampilkan pertama kali kepada *user*. Tampilan layar ini terdapat beberapa suatu menu, antara lain, menu data yang didalamnya terdapat menu *dataset* yang berfungsi untuk meng-*import* data yang akan dijadikan *dataset* uji dan *dataset* model ke dalam *database*, menu *proccess* yang didalam nya berisi menu evaluasi dan klasifikasi, evaluasi berfungsi untuk pengujian dan mencari parameter (k) terbaik pada *dataset* uji, menu klasifikasi berfungsi untuk mengklasifikasi data kelulusan ujian nasional siswa berdasarkan nilai ujian *TryOut*, menu *about* yang didalam nya berisi menu *author* berfungsi untuk mengetahui informasi dari penulis dan menu *option* yang didalamnya ada menu *exit* berfungsi untuk keluar dari sistem ini. Tampilan layar *form* menu utama dapat dilihat pada (gambar 4) berikut ini:

Gambar 4. Tampilan *Form* Menu Utama

b. Tampilan *Form Dataset*

Tampilan *Form Dataset* ini akan muncul saat tombol menu data tersebut memilih suatu menu *dataset* Menu Utama. Pada fungsi *form* ini adalah suatu *database* dan akan digunakan sebagai *dataset* uji dan *dataset* model pada proses ini akan diprediksi dan klasifikasi kelulusan ujian nasional berdasarkan suatu nilai ujian *TryOut*. Tampilan layar *form dataset* yang bisa kita lihat pada gambar 5 berikut ini:

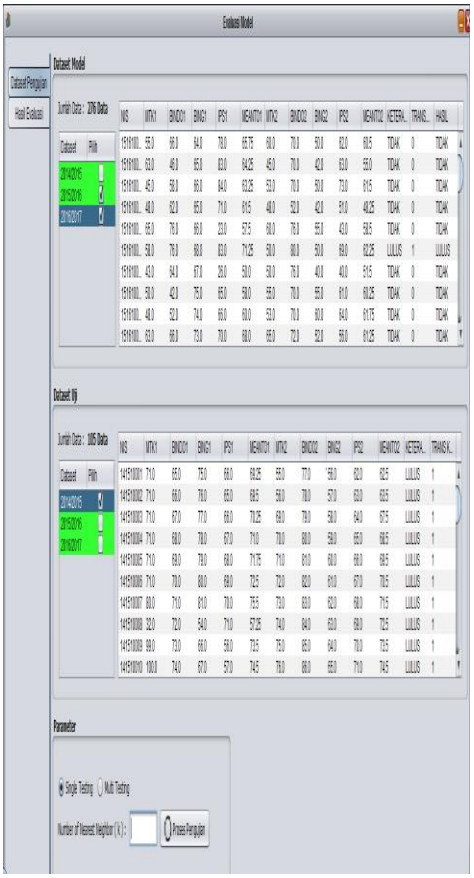


Gambar 5. Tampilan *Form Dataset*

c. Tampilan *Form* Evaluasi

Tampilan layar *form* evaluasi ini akan dimuncul pada saat tombol menu *process* yang telah dipilih. *Form* ini memiliki dua *tab* yaitu *tab Dataset Pengujian* dan *Hasil Evaluasi*. *Form Tab Dataset* Pengujian berfungsi untuk memilih *dataset* mana saja yang akan dilakukan proses pengujian. Didalam *tab dataset*

pengujian ada *dataset* model dan *dataset* uji yang berguna untuk melakukan pencarian nilai K terbaik dalam sistem ini dengan cara membandingkan *dataset* model dengan *dataset* uji. Tampilan Layar *Form* Evaluasi *Tab Dataset* Pengujian ditunjukan pada (gambar 5) berikut ini:



Gambar 6. Tampilan *Form* Evaluasi *Tab*

d. Tampilan *Form Author*

Tampilan layar *form author* akan menampilkan *form* berisi data dari penulis seperti nama nim fakultas dan lainnya. Tampilan layar *form author* dapat dilihat pada (gambar 7) berikut ini:

Gambar 7 Tampilan Layar *Form Author*

5. Kesimpulan

Berdasarkan hasil penelitian ini, maka dapat simpulkan bahwa:

a. Menggunakan Algoritma *K-Nearest Neighbor* menggunakan metode evaluasi dan klasifikasi dengan pengukuran jarak *Euclidean Distance* dapat diimplementasikan melalui proses prediksi untuk Kelulusan Ujian Nasional

b. Nilai akurasi pada suatu pengujian ini akan menggunakan algoritma *K-Nearest Neighbor* dengan pengukur jarak *Euclidean distance* berdasarkan jumlah uji ini sebanyak 2 (dua) kali dengan total 382 data selama 3 (tiga) angkatan, didapat nilai K=5 dengan nilai akurasi terbesar 96,96% , nilai presisi 88,57% dan nilai *recall* 100% pada data tahun 2014-2015

c. Waktu yang dibutuhkan dalam proses prediksi kululusan siswa terbilang cukup cepat apabila dibandingkan dengan proses sebelumnya yang memakan waktu hingga 4-5 hari

5. Daftar Pustaka

[1] Abdelmoula, A. K. 2015. Bank Credit Risk Analysis with K-Nearest-Neighbor Classifier : Case of Tunisian Banks. Accounting and Mangement Information Systems, 14(1), 79–106.

[2] Sasmita, A, S., Z, A., & A, Y. 2013. Penerapan Decision Tree Untuk Penentuan Pola Data Penerimaan Mahasiswa Baru. Jurnal Penelitian Sitritika, 7. http://doi.org/10.1017/CBO9781107415324.004

[3] Banjarbaru, S. M. A. P. 2016. PENERAPAN METODE LEARNING VECTOR QUANTIZATION (LVQ) PADA PREDIKSI JURUSAN DI, 4(1), 11–20.

[4] Hasugian, J. 2006. Penggunaan Bahasa Alamiah Dan Kosa Kata Terkontrol Dalam Sistem Temu Kembali Informasi Berbasis Teks. Jurnal Studi Perpustakaan Dan Informasi, 2(2), 1–10.