

KLASIFIKASI PENERIMAAN TENAGA KERJA TERTUTUP MENGUNAKAN ALGORITMA K-NEAREST NEIGHBOR (KNN)

Fridy Mandita¹⁾, Rizal Kurnia Pratama²⁾

^{1,2)}Teknik Informatika, Universitas 17 Agustus 1945 Surabaya

Jl. Semolowaru No.45, Menur Pumpungan, Kec. Sukolilo, Surabaya, Jawa Timur 60118

e-mail: fridymandita@untag-sby.ac.id¹⁾, rkurniapratama@gmail.com²⁾

(Naskah masuk : 16 Januari 2024 Diterima untuk diterbitkan : 16 April 2024)

ABSTRAK

Perusahaan yang telah menerapkan pemanfaatan teknologi tidak akan lepas dari pengolahan data atau informasi, akan tetapi tidak semua perusahaan dapat memaksimalkan pengolahan data yang mereka miliki, salah satunya di CV. Arfa Nusantara Teknologi, data calon karyawan yang masuk ke dalam sebuah perusahaan tidak dilakukan pengolahan lebih lanjut. Penelitian ini mencoba untuk melakukan pengolahan terhadap data calon tenaga kerja yang masuk ke sebuah perusahaan dengan tujuan untuk mengklasifikasikan data calon tenaga kerja menggunakan algoritma K-Nearest Neighbor (KNN) berdasarkan variabel skor tes teknis, total nilai bobot keahlian, minimum gaji yang diminta dan lama pengalaman kerja. Hasilnya akan diklasifikasikan menjadi paling baik, cukup baik dan kurang baik. Tujuannya adalah membantu tim penilai perusahaan dalam memilih calon tenaga kerja terbaik dan mengetahui akurasi algoritma. Penelitian ini juga mencoba berbagai jumlah pengambilan tetangga KNN dan menerapkan K-Fold Cross Validation untuk menentukan model klasifikasi terbaik. Model optimal dengan akurasi, presisi dan recall 100% dicapai pada pembagian $k = 4$ dan 5 dalam uji coba 10 K-Fold Cross Validation. Algoritma KNN menunjukkan kinerja terbaik pada ketetanggaan $k = 7$. Pengujian System Usability Scale mendapatkan skor 80,21 (Kategori B).

Kata kunci: Klasifikasi, K-Nearest Neighbor, K-Fold Cross Validation, Penerimaan Tenaga Kerja

ABSTRACT

Companies that have implemented the use of technology will not be separated from data or information processing, but not all companies can maximise the data processing they have, one of which is at CV. Arfa Nusantara Teknologi, data on prospective employees who enter a company is not processed further. This research tries to process data on prospective workers who enter a company with the aim of classifying prospective workforce data using the K-Nearest Neighbor (KNN) algorithm based on technical test score variables, total skill weight value, minimum salary requested and length of work experience. The results will be classified into best, good enough and less good. The goal is to help the company's assessment team in selecting the best labour candidates and determine the accuracy of the algorithm. This research also tries various numbers of KNN neighbour retrieval and applies K-Fold Cross Validation to determine the best classification model. The optimal model with 100% accuracy, precision and recall was achieved at division $k = 4$ and 5 in 10 K-Fold Cross Validation trials. The KNN algorithm showed the best performance at $k = 7$ neighbourhoods. The System Usability Scale test received a score of 80.21 (Category B).

Keywords: Classification, K-Nearest Neighbor, K-Fold Cross Validation, Labor Recruitment

I. PENDAHULUAN

Departemen Sumber Daya Manusia (SDM) sangat penting dalam perusahaan atau organisasi mana pun, karena departemen ini terutama bertanggung jawab untuk mengawasi operasi penting termasuk manajemen proses rekrutmen, penyediaan pendidikan tenaga kerja, analisis kebijakan tenaga kerja, dan kesejahteraan karyawan. Seringkali, departemen SDM bekerja sama dengan departemen terkait yang membutuhkan personel tambahan untuk membentuk tim penilai sendiri ketika merekrut personel baru. Banyak faktor yang dipertimbangkan oleh kelompok ini dalam proses evaluasi calon kandidat. Proses seleksi, yang merupakan komponen penting dalam manajemen personalia, terjadi setelah analisis pekerjaan, yang menghasilkan deskripsi pekerjaan dan prasyarat bagi calon karyawan untuk memenuhi posisi tertentu [1]. Mengidentifikasi dan menarik calon kandidat, mengevaluasi lamaran mereka, dan

pada akhirnya mempekerjakan individu yang paling memenuhi syarat merupakan prosedur seleksi [2]. Dalam fase penyaringan lamaran, departemen SDM melakukan pemeriksaan menyeluruh terhadap setiap lamaran untuk menentukan kandidat mana yang memenuhi persyaratan organisasi. Prosedur penyaringan mengalami kesulitan karena banyaknya aplikasi yang masuk. Setelah kandidat berhasil menyelesaikan tahap penyaringan CV, departemen SDM dihadapkan pada tugas menilai kesesuaian setiap individu untuk organisasi, yang dapat mengakibatkan gangguan dan inefisiensi proses perekrutan [3]. Banyak waktu yang masih terbuang oleh sejumlah tim penilai karena mereka terus mengandalkan proses manual untuk membandingkan data kandidat. Waktu sangat penting dalam mencapai keakuratan evaluasi oleh karena itu, algoritma yang mudah diinterpretasikan dan diimplementasikan sangatlah penting, khususnya dalam klasifikasi. Terdapat berbagai macam metode yang dapat digunakan untuk melakukan klasifikasi data, yang dimana setiap metode memiliki kelebihan dan kekurangannya masing-masing dan metode dapat berjalan dengan baik berdasarkan dari objek dataset yang akan diolah [4].

Adapun penelitian sebelumnya yang juga membahas mengenai klasifikasi terkait penerimaan siswa baru menggunakan algoritma *K-Nearest Neighbor (KNN)*, *Decision Tree Classifier*, *Naive Bayes Classifier*, *Support Vector Machine (SVM)* dan *AdaBoost*. Tujuan penelitian adalah untuk menilai kualitas algoritma, khususnya kemampuannya dalam memastikan preferensi calon siswa terhadap program studinya. Algoritma *KNN Classifier* menunjukkan tingkat akurasi tertinggi mencapai 72,2%. Sebaliknya, *Decision Tree Classifier* mencapai tingkat akurasi terendah yaitu 33,4%. Algoritma tambahan yang mencapai tingkat akurasi dalam rentang menengah adalah *Naive Bayes Classifier* (60,7%), *Support Vector Machine (SVM)* (61,7%), dan *Adaptive Boosting (AdaBoost)* (62,4%). Berdasarkan hasil evaluasi tersebut, dapat disimpulkan bahwa algoritma *KNN* menunjukkan akurasi yang unggul dalam memodelkan preferensi program studi berdasarkan dataset penerimaan mahasiswa baru, mencapai tingkat akurasi sebesar 72,2% [5]. Adapun penelitian lainnya yang telah dilakukan sebelumnya yang juga membahas mengenai klasifikasi terkait rekomendasi penentuan penerimaan beasiswa pada Universitas Kuningan menggunakan algoritma *K-Nearest Neighbor (KNN)* dan *Naive Bayes*. Pada algoritma *KNN* menggunakan perhitungan *euclidean distance*. Variabel yang digunakan yaitu IPK, pekerjaan orang tua, jumlah tanggungan dan penghasilan orang tua. Dari kesimpulan penelitian tersebut dihasilkan tingkat akurasi untuk algoritma *KNN* adalah sebesar 100% dan algoritma *Naive Bayes* sebesar 99,89% [6].

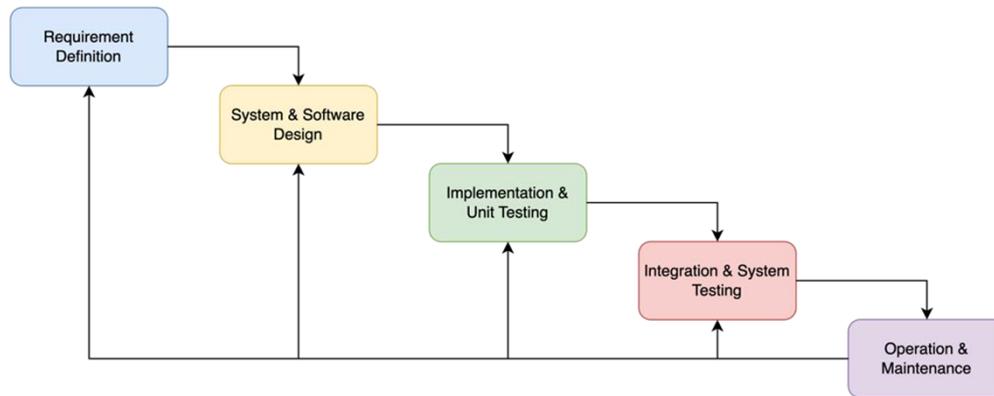
Pada penelitian lain yang telah dilakukan sebelumnya yang juga membahas mengenai klasifikasi terkait diagnosis penyakit diabetes melitus menggunakan algoritma *K-Nearest Neighbor (KNN)* dan *Naive Bayes*. Penelitian ini mengklasifikasikan kedalam 2 kategori yaitu dinyatakan terdiagnosa dan tidak terdiagnosa. Pada algoritma *KNN* menggunakan perhitungan *euclidean distance*. Dari hasil penelitian dengan membandingkan kedua algoritma tersebut di hasilkan tingkat akurasi dimana algoritma *Naive Bayes* mendapatkan tingkat akurasi yang lebih tinggi sebesar 80% dibandingkan dengan algoritma *KNN* sebesar 75%, selain itu nilai *recall* dari algoritma *KNN* lebih tinggi 0,92 dan untuk nilai *precision* algoritma *Naive Bayes* lebih tinggi dengan nilai 0,86 [7].

Berdasarkan 2 penelitian sebelumnya dimana algoritma *KNN* mendapatkan akurasi yang tinggi dibandingkan algoritma lainnya akan tetapi di salah satu penelitian terdahulu juga menyebutkan algoritma selain *KNN* lebih tinggi akurasinya, dari sini dapat dikatakan bahwa akurasi dari algoritma juga bergantung dari data studi kasus yang akan digunakan dalam klasifikasi, maka dari itu dilakukan penelitian yang bertujuan untuk menyelidiki dan menerapkan algoritma *KNN* dalam masalah klasifikasi terhadap data penerimaan calon tenaga kerja yang ada di CV. Arfa Nusantara Teknologi guna untuk mendapatkan tingkat akurasi dari algoritma, selain itu dalam penelitian ini juga akan membahas mengenai evaluasi performa model menggunakan *K-Fold Cross Validation* untuk mendapatkan model terbaik yang akan di terapkan dalam klasifikasi. Dari hasil penelitian ini akan dikembangkan sebuah sistem berbasis web dengan menerapkan proses klasifikasi terhadap data calon tenaga kerja menggunakan algoritma *KNN* berdasarkan variabel hasil skor tes teknis dari tim penilai, total nilai bobot keahlian, minimum gaji yang diminta dan jumlah tahun pengalaman bekerja. Diharapkan dengan adanya penelitian ini sistem yang akan akan dikembangkan dapat memberikan daftar calon tenaga kerja kedalam beberapa kategori yaitu sangat baik, cukup baik dan kurang baik sehingga masalah yang dihadapi oleh departemen SDM dapat terselesaikan.

II. METODE PENELITIAN

2.1. Metode Pengembangan Waterfall

Terdapat berbagai metode dalam proses pengembangan perangkat lunak salah satunya adalah *waterfall* yang banyak digunakan dalam proses pengembangan. Metode *waterfall* memiliki pendekatan desain air terjun menggunakan prosedur yang secara bertahap bergerak dari atas ke bawah secara sequensial, menyerupai penampilan air terjun [8].



Gambar 2. 1 Metode Pengembangan *Waterfall*

Tahapan metode pengembangan *waterfall* memiliki 5 tahapan sebagai berikut [8][9]:

1. *Requirement Definition*
Pada tahap ini dilakukan proses pengumpulan informasi terkait kebutuhan fungsional maupun non-fungsional sistem dengan melibatkan pengguna dan pemangku kepentingan. Dalam tahap ini juga dilakukan proses pengamatan lapangan untuk mengidentifikasi masalah dan kebutuhan apa saja yang nantinya akan diakomodir oleh sistem yang akan dibangun.
2. *System & Software Design*
Pada tahap ini dilakukan proses pembuatan desain sistem yang lebih spesifik yang nanti akan ditujukan untuk memudahkan tim pengembang dalam mengembangkan sistem. Desain sistem mencakup *User Interface*, *ERD*, *Use Case*, *Activity Diagram* dan lain-lain.
3. *Implementation & Unit Testing*
Pada tahap ini tim pengembang akan mulai melakukan proses pengkodean sistem berdasarkan desain sistem yang telah dibuat pada proses sebelumnya dan melakukan pengujian secara kecil untuk memastikan kode yang dibuat berjalan sesuai dengan desain yang dibuat.
4. *Integration & System Testing*
Pada tahap ini dilakukan pengujian secara menyeluruh terhadap sistem yang telah dibangun untuk memastikan apakah sistem yang dibangun telah memenuhi kebutuhan yang telah didefinisikan di tahap sebelumnya, sehingga apabila terdapat kebutuhan yang belum terpenuhi maka akan dilakukan pengkajian ulang terhadap sistem.
5. *Operation & Maintenance*
Pada tahap ini sistem sudah bisa digunakan oleh pengguna, akan tetapi sistem akan butuh *maintenance* secara rutin untuk memastikan agar sistem yang telah dibangun berjalan dengan optimal dan bahkan memungkinkan untuk terjadinya penambahan fitur baru seiring dengan peningkatan kebutuhan.

2.2. K-Nearest Neighbor

Data yang telah dihitung jarak kedekatannya dan diurutkan berdasarkan yang terdekat ke yang terjauh nantinya hanya akan diambil data sejumlah nilai dari k . Dimana nilai k ialah jumlah dari tetangga terdekat yang akan diambil sebagai penentu klasifikasi dengan melihat kelas dominan yang muncul pada data yang diambil sejumlah nilai k tersebut [10]. Menurut penelitian terdahulu terdapat berbagai macam cara untuk menghitung jarak kedekatan antar data diantaranya *Euclidean*, *Manhattan*,

Minkowski dan *Chebychev* [11]. Perhitungan jarak pada data berdimensi tinggi lebih baik menggunakan *Euclidean*, *Manhattan* dan *Minkowski* sedangkan untuk data berdimensi rendah lebih baik menggunakan *Chebychev*, selain itu dari perhitungan jarak *Euclidean*, *Manhattan* dan *Minkowski* hanya *Manhattan* yang tidak menggunakan perhitungan kuadrat sehingga hanya pada $k = 1$ saja mendapatkan hasil terbaik jika dibandingkan dengan *Euclidean* dan *Minkowski* yang memiliki akurasi terbaik di beberapa jumlah k . Rumus yang di gunakan pada *Euclidean* adalah yang paling umum dan mudah dimengerti secara intuitif dibandingkan dengan *Minkowski* yang dimana diharuskan menentukan nilai p yang menjadikannya sebuah tantangan dalam beberapa kasus. Sehingga penulis memilih menggunakan perhitungan jarak *Euclidean Distance*.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.1)$$

Keterangan:

$d(x, y)$	=	Jarak
x_i	=	Data testing
y_i	=	Data training
i	=	Variabel data
n	=	Dimensi data

2.3. Confusion Matrix

Confusion Matrix adalah statistik yang digunakan untuk menilai efektivitas dan kualitas model yang dibangun [12]. Terutama diterapkan dalam domain penambangan data, ini menilai akurasi dengan menyediakan tabel yang mengilustrasikan data pengujian yang benar dan salah diklasifikasikan. Nilai-nilai dalam matriks konfusi kemudian digunakan untuk menghitung akurasi, presisi, dan *recall*.

Tabel 2. 1 *Confusion Matrix*

	Aktual	
Prediksi	<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	TP (<i>True Positive</i>)	FP (<i>False Positive</i>)
<i>Negative</i>	FN (<i>False Negative</i>)	TN (<i>True Negative</i>)

Keterangan:

TP	=	Jumlah di mana data sebenarnya positif dan prediksi juga menunjukkan positif.
FP	=	Jumlah di mana data sebenarnya negatif tapi prediksi menunjukkan positif.
FN	=	Jumlah di mana data sebenarnya positif tapi prediksi menunjukkan negatif.
TN	=	Jumlah di mana data sebenarnya negatif dan prediksi juga menunjukkan negatif

1. Akurasi

Akurasi berfungsi sebagai metrik penilaian dengan membandingkan jumlah kelas yang diprediksi positif dan negatif pada kenyataannya dibandingkan dengan seluruh kumpulan data yang digunakan. Perhitungan nilai akurasi dapat dilakukan melalui rumus berikut:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.2)$$

2. Presisi

Presisi berfungsi sebagai metrik penilaian dengan membandingkan jumlah kelas yang diprediksi positif pada kenyataannya dengan jumlah total kelas yang diprediksi positif. Perhitungan nilai presisi dapat dilakukan melalui rumus berikut:

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

3. Recall

Recall berfungsi sebagai metrik penilaian dengan mengontraskan jumlah kelas yang diprediksi positif pada kenyataannya dengan jumlah total dari semua kelas yang diprediksi positif dan negatif pada kenyataannya. Perhitungan nilai *recall* dapat dilakukan melalui rumus berikut:

$$recall = \frac{TP}{TP + FN} \quad (2.4)$$

2.4. K-Fold Cross Validation

Dengan menggunakan metode ini untuk mempartisi data ke dalam set pelatihan dan pengujian, peneliti sering kali mengurangi bias pengambilan sampel. Dengan mempartisi kumpulan data secara berulang menjadi subset pelatihan dan pengujian, *K-Fold Cross Validation* memberikan setiap titik data peluang untuk disertakan dalam kumpulan pengujian [13]. Metodologi ini melibatkan pembagian stokastik kumpulan data menjadi beberapa partisi. Setiap eksperimen menggunakan partisi ke-*k* sebagai set pengujian dan partisi yang tersisa sebagai set pelatihan di seluruh *k* iterasi data yang dipartisi. Pada penelitian ini akan menggunakan *K-Fold Cross Validation* dengan nilai *k* = 5 dan 10. Pada penelitian sebelumnya bahwa nilai *k* = 5 dan 10 sama-sama direkomendasikan walaupun semakin kecil nilai *k* akan memiliki bias yang tinggi dari pada nilai *k* yang tinggi akan memiliki bias yang rendah, akan tetapi semua bergantung pada tujuannya [14]. Penelitian ini juga memiliki tujuan untuk menguji ketangguhan dataset yang ada, maka dibutuhkan nilai *k* yang kecil pada *K-Fold Cross Validation* agar mengetahui seberapa besar bias pada data.

2.5. PHP

PHP merupakan singkatan dari *Hypertext Preprocessor*. Ini adalah bahasa pemrograman *open source* yang paling banyak digunakan dalam membangun web dinamis. Bahasa pemrograman ini merupakan *Server-Side Scripting* yang dimana sumber kode dijalankan diserver dan hasilnya akan dikirimkan ke peramban web sehingga bisa diintegrasikan dengan *HTML* [15]. Bahasa pemrograman ini juga memiliki dukungan untuk dapat terhubung dengan database seperti *MySQL*, *PostgreSQL*, *Oracle* dan lain-lain.

2.6. Laravel

Laravel merupakan sebuah *framework* web yang berbasis bahasa pemrograman *PHP* yang dikembangkan oleh Taylor Otwell. *Framework* ini bersifat *open source* yang bisa dengan mudah dilakukan kustomisasi sumber kodenya. *Framework* ini memiliki arsitektur *MVC (Model View Controller)* yang dimana membuat sumber kode yang akan dibuat menjadi lebih tersusun rapi [16]. *Laravel* merupakan *framework PHP* yang saat ini paling populer digunakan oleh para pengembang karena memiliki berbagai macam fitur mulai dari autentikasi, otorisasi, *ORM (Object Relation Mapping)*, validasi dan lain-lain, tidak hanya itu *laravel* ini memiliki banyak dukungan ekstensi gratis dari berbagai pengembang yang bisa diunduh.

2.7. System Usability Scale

Pada tahun 1986, John Brooke mengembangkan *System Usability Scale (SUS)*. *SUS* adalah metode yang digunakan untuk mengukur seberapa efektif suatu sistem dengan mengajukan pertanyaan khusus kepada sekelompok peserta [17]. *SUS* terdiri dari sepuluh pertanyaan, dan setiap pertanyaan dilengkapi dengan lima pilihan jawaban, mulai dari sangat setuju hingga sangat tidak setuju dan nilai maksimal skor berkisar 0 hingga 100. Adapun aturan untuk menentukan skor *SUS* [18]:

1. Setiap pertanyaan yang memiliki nomor ganjil nantinya untuk skor pertanyaan yang dipilih dikurangi 1 ($X-1$).
2. Setiap pertanyaan yang memiliki nomor genap nantinya nilai 5 dikurangi skor pertanyaan yang dipilih ($5-X$).
3. Setelah semua skor pertanyaan didapatkan dan dijumlahkan secara keseluruhan kemudian hasilnya dikalikan dengan 2,5.

$$Skor\ SUS = ((X1 - 1) + (5 - X2) + (X3 - 1) + (5 - X4) + (X5 - 1) + (5 - X6) + (X7 - 1) + (5 - X8) + (X9 - 1) + (5 - X10)) \times 2,5 \quad (2.5)$$

Rata-rata skor *SUS* yang diperoleh dari beberapa investigasi adalah 68. Oleh karena itu, Skor *SUS* di atas 68 dianggap di atas rata-rata, sedangkan skor di bawah 68 dianggap di bawah rata-rata. Namun,

penting untuk diperhatikan bahwa kategori penilaian pada *SUS* telah ditetapkan oleh penelitian sebelumnya [19].

Tabel 2. 2 Kategori Nilai *SUS*

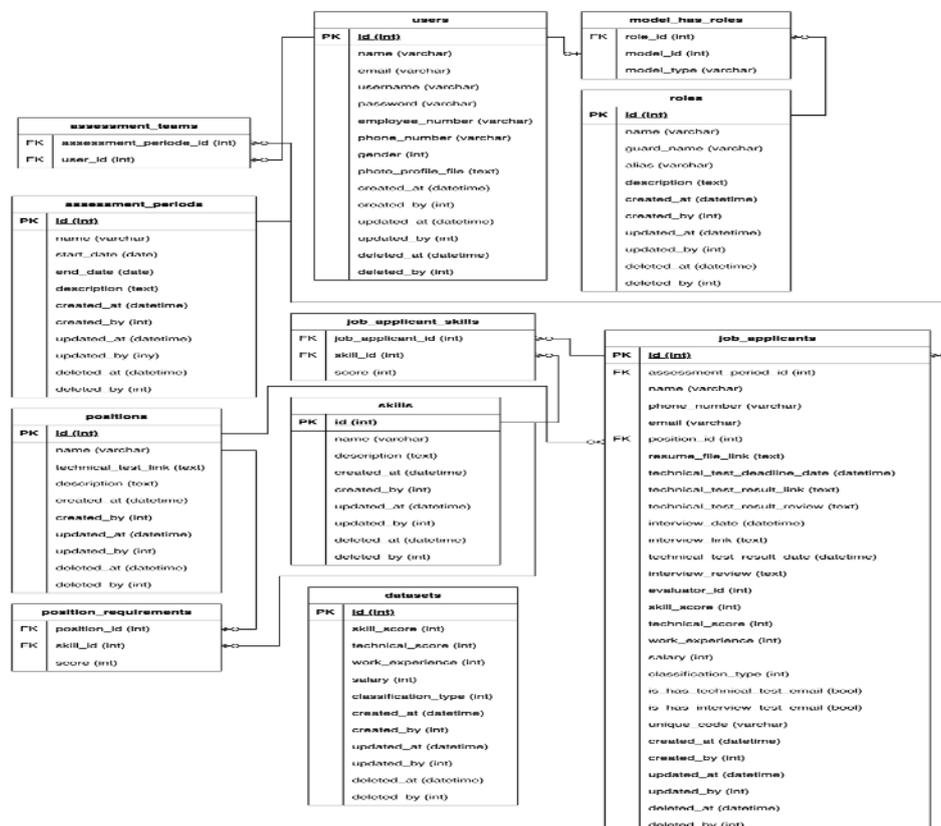
Kategori	Skor Penilaian
A	90 - 100
B	80 - 90
C	70 - 80
D	60 - 70
F	< 60

III. HASIL DAN PEMBAHASAN

Penelitian ini mengembangkan suatu sistem berbasis web yang dapat memberikan bantuan tim penilai saat melakukan proses rekrutmen dengan memprediksi calon tenaga kerja yang melamar diperusahaan ke dalam tiga kategori: sangat baik, cukup baik dan kurang baik menggunakan algoritma *KNN* dengan perhitungan jarak *euclidean distance* dengan nilai *k* ketetanggaan 5, 7 dan 10. Untuk pengujian algoritma *KNN* agar mendapatkan model terbaik yang nanti akan digunakan oleh sistem akan mengimplementasikan *K-Fold Cross Validation* menggunakan nilai *k* pembagian 5 dan 10. Pada sistem yang telah dibuat nantinya juga akan dilakukan penilaian *SUS* untuk mendapatkan penilaian dari responden secara langsung terkait dengan fungsionalitas sistem dan kemudahan dalam menggunakan sistem.

3.1. Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan diagram visual yang dapat menjelaskan tentang hubungan tabel satu dengan lainnya yang akan menjadi tempat penampungan data dari proses yang akan terjadi pada sistem atau biasa disebut *database* [20]. Database yang akan digunakan dalam penelitian ini adalah *PostgreSQL*.



Gambar 3. 1 Entity Relationship Diagram

3.2. Hasil Evaluasi K-Nearest Neighbor dan K-Fold Cross Validation

Berdasarkan penelitian ini akan digunakan data calon tenaga kerja yang pernah melamar di perusahaan yang dimana akan dilakukan proses klasifikasi berdasarkan variabel hasil skor tes teknis dari tim penilai, total nilai bobot keahlian, minimum gaji yang diminta dan jumlah tahun pengalaman bekerja.

Tabel 3. 2 Dataset Calon Tenaga Kerja

Indeks	Nilai Tes	Bobot Keahlian	Pengalaman Kerja	Min. Gaji	Klasifikasi
1	75	55	0	4000000	3
2	85	75	0	4500000	2
3	80	65	0	4500000	3
4	80	75	0	6000000	3
5	70	60	0	4000000	3
6	85	90	0	4000000	1
7	70	65	0	3500000	3
8	80	75	0	6000000	3
9	75	75	0	3500000	2
10	80	80	0	4000000	1
11	85	85	1	3500000	1
...
106	75	50	0	4000000	3
107	80	70	0	4000000	2
108	85	75	0	4500000	2
109	70	70	0	6000000	3
110	80	75	0	6000000	3

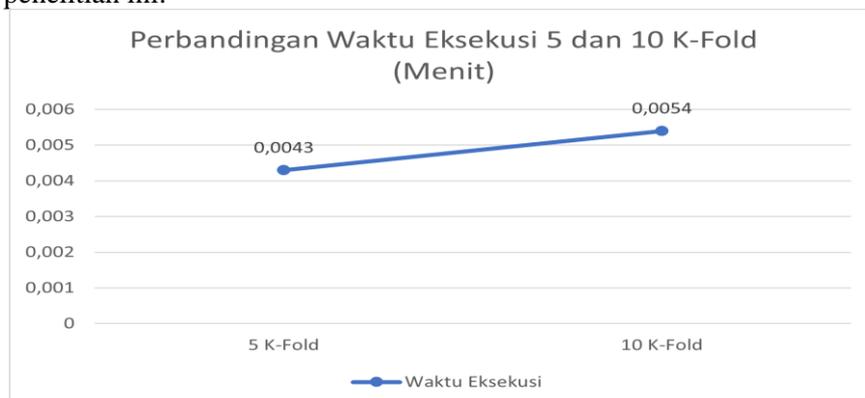
Dalam investigasi ini, sejumlah eksperimen dilakukan agar model prediktif optimal untuk klasifikasi data rekrutmen tenaga kerja dapat dikembangkan. Eksperimen yang dilakukan meliputi evaluasi algoritma *KNN* melalui pemanfaatan rumus (2.1) untuk menghitung jarak *euclidean distance*, dengan memperhatikan berbagai nilai *k* (5, 7, dan 10), dan mengeksekusi *K-Fold Cross Validation* dengan nilai *k* yang bervariasi (5 dan 10). Investigasi tersebut melibatkan pembagian kumpulan data lengkap, yang terdiri dari 110 data yang dapat dilihat pada **Tabel 3.**, menjadi dua bagian berbeda: kumpulan pelatihan, yang dipilih secara acak, dan kumpulan pengujian. Pembagian ini ditentukan oleh *K-Fold Cross Validation*, dengan *K-Fold 5* menerima pembagian 80%:20% dan *K-Fold 10* menerima pembagian 90%:10%. Hasilnya, rasio data pelatihan-pengujian untuk setiap *K-Fold* adalah sebagai berikut: 88:22 untuk *K-Fold 5* dan 99:11 untuk *K-Fold 10*. Evaluasi akurasi, presisi dan *recall* dihitung menggunakan rumus (2.2) untuk akurasi, menggunakan rumus (2.3) untuk presisi dan menggunakan rumus (2.4) untuk *recall*. Hasil percobaan yang dilakukan dirinci pada **Tabel 3. 1**.

Tabel 3. 1 Hasil Evaluasi Klasifikasi

Jumlah K-Fold	K (K-Fold)	K (Tetangga KNN)	Akurasi	Presisi	Recall
5	1	5	0,86	0,86	0,89
		7	0,82	0,83	0,84
		10	0,73	0,8	0,75
5	2	5	0,95	0,97	0,89
		7	0,91	0,86	0,86
		10	0,77	0,73	0,74
5	3	5	0,82	0,78	0,78
		7	0,73	0,67	0,67
		10	0,91	0,92	0,89
5	4	5	1	1	1
		7	0,95	0,92	0,98
		10	0,95	0,92	0,98
5	5	5	0,95	0,93	0,93
		7	0,95	0,93	0,93
		10	0,77	0,68	0,63
10	1	5	0,82	0,87	0,78

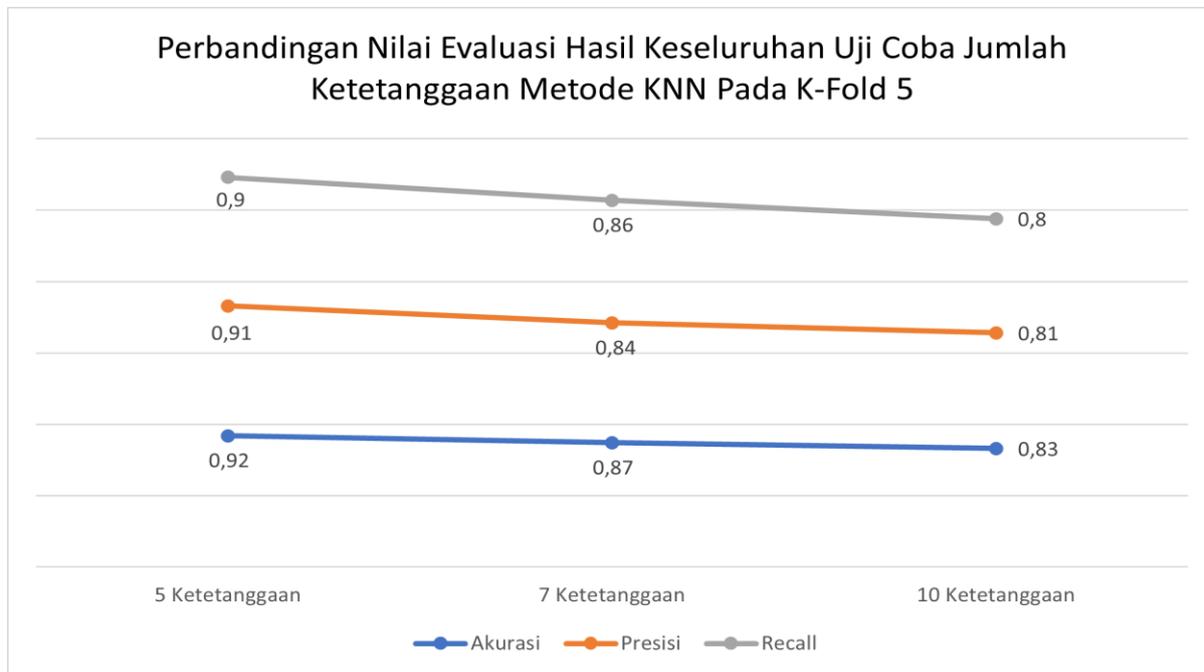
Jumlah K-Fold	K (K-Fold)	K (Tetangga KNN)	Akurasi	Presisi	Recall
		7	0,82	0,9	0,78
		10	1	1	1
	2	5	0,91	0,93	0,93
		7	0,91	0,93	0,93
		10	0,82	0,85	0,85
	3	5	0,82	0,83	0,85
		7	0,82	0,83	0,85
		10	0,82	0,81	0,85
	4	5	1	1	1
		7	1	1	1
		10	1	1	1
	5	5	1	1	1
		7	1	1	1
		10	1	1	1
	6	5	1	1	1
		7	1	1	1
		10	0,91	0,92	0,89
	7	5	1	1	1
		7	1	1	1
		10	0,91	0,89	0,92
	8	5	1	1	1
		7	1	1	1
		10	0,91	0,58	0,67
	9	5	1	1	1
		7	1	1	1
		10	0,91	0,92	0,92
	10	5	1	1	1
		7	1	1	1
		10	0,82	0,83	0,89

Berdasarkan hasil evaluasi pada **Tabel 3. 1**, terlihat bahwa terdapat beberapa uji coba yang menghasilkan nilai evaluasi yang sangat baik dan berhasil memprediksi semua data *testing* dengan tepat. Jika diamati lebih detail, penerapan 10 pembagian *K-Fold Cross Validation* lebih banyak memprediksi semua data *testing* dengan tepat dengan rentang nilai akurasi antara 82% hingga 100% jika dibandingkan pada 5 pembagian *K-Fold Cross Validation* yang memiliki rentang nilai akurasi 73% hingga 100%. Semakin tinggi nilai pembagi *K-Fold* maka semakin tinggi pula variasi data yang dapat terjadi pada data *training* dan *testing*. Sehingga semakin banyak kemungkinan model akan jauh lebih baik maupun sebaliknya. Hal tersebut juga sangat dipengaruhi oleh proses *random* data pada *K-Fold Cross Validation* yang memungkinkan data akan terbagi menjadi sangat baik, cukup baik, maupun kurang baik. Dengan beberapa kombinasi variasi data ini, penulis akan mendapatkan urutan data yang paling tepat, yang dapat merepresentasikan data *training* dan *testing* dengan baik, sehingga model yang dihasilkan jauh lebih optimal. Selanjutnya dapat dilihat beberapa grafik sebagai analisis hasil penerapan uji coba pada penelitian ini.



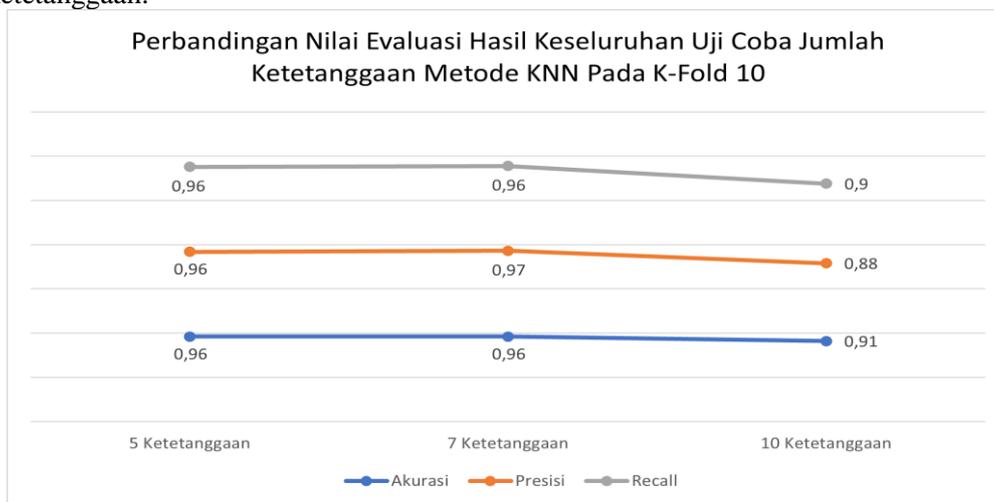
Gambar 3. 2 Grafik Hasil Perbandingan Waktu Eksekusi

Berdasarkan grafik **Gambar 3. 2** menunjukkan perbandingan waktu eksekusi antara *K-Fold Cross Validation* menggunakan k bernilai 5 dan 10 untuk menguji coba model terlihat hasilnya tidak terlalu jauh dikarenakan jumlah dataset yang digunakan juga tidak terlalu besar pada penelitian ini. Pada 5 *K-Fold* mendapatkan waktu eksekusi yang lebih rendah dikarenakan jumlah perulangan untuk menguji model lebih sedikit dibandingkan dengan 10 *K-Fold*.



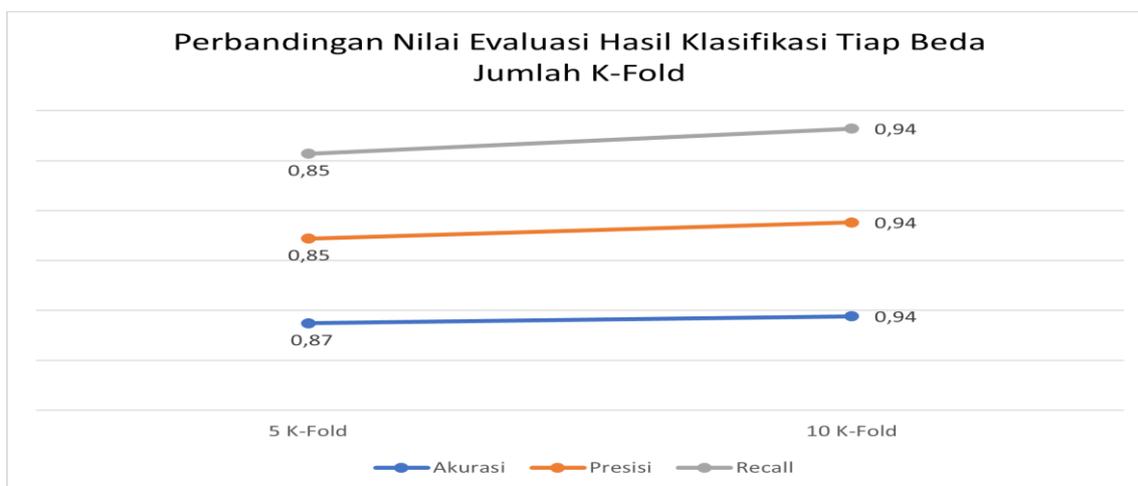
Gambar 3. 3 Grafik Hasil Evaluasi Ketetangaan Pada K-Fold 5

Berdasarkan grafik **Gambar 3. 3** menunjukkan bahwa rata-rata nilai akurasi, presisi, dan *recall* pada pemilihan jumlah ketetangaan algoritma *KNN* pada *K-Fold Cross Validation 5* yang memiliki rata-rata nilai paling baik ada pada ketetangaan 5 dengan rata-rata nilai akurasi 92%, presisi 91% dan *recall* 90%. Sedangkan pada ketetangaan 7 memiliki rata-rata nilai akurasi 87%, presisi 84% dan *recall* 86% dimana nilai ini lebih kecil dibandingkan ketetangaan 5. Sedangkan pada ketetangaan 10 memiliki rata-rata nilai akurasi 83%, presisi 81% dan *recall* 80% dimana nilai ini yang paling kecil diantara semua ketetangaan.



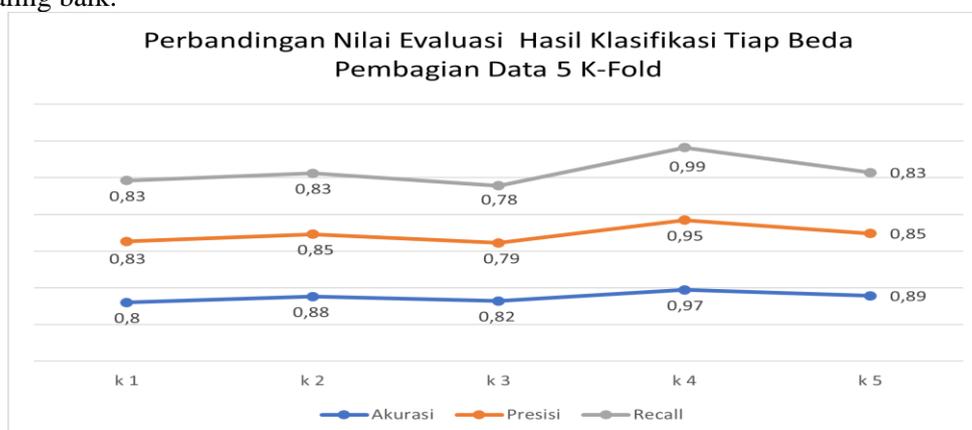
Gambar 3. 4 Grafik Hasil Evaluasi Ketetangaan Pada K-Fold 10

Berdasarkan grafik **Gambar 3. 4** menggambarkan bahwa rata-rata nilai dari akurasi, presisi, dan *recall* berdasarkan pemilihan jumlah ketetangaan algoritma *KNN* di *K-Fold Cross Validation* 10 yang memiliki rata-rata nilai paling baik ada pada ketetangaan 7 dengan hasil dari rata-rata nilai akurasi 96%, presisi 97% dan *recall* 96%. Sedangkan pada ketetangaan 5 memiliki rata-rata nilai akurasi, presisi dan *recall* yang sama yaitu 96% dimana nilai ini hampir sama dengan ketetangaan 7 hanya saja pada bagian presisi ketetangaan 5 lebih rendah 1%. Sedangkan pada ketetangaan 10 memiliki rata-rata nilai akurasi 91%, presisi 88% dan *recall* 90% dimana nilai ini yang paling kecil diantara semua ketetangaan. Jika dibandingkan dengan rata-rata nilai tertinggi ketetangaan 5 pada *K-Fold* 5 untuk rata-rata nilai ketetangaan 7 pada *K-Fold* 10 masih lebih besar. Dapat disimpulkan bahwa jumlah ketetangaan *KNN* sangat menentukan hasil prediksi akhir, semakin jauh ketetangaan yang diperhitungkan, semakin bermacam-macam hasil kelas yang diprediksi, sedangkan probabilitas suara penentuan kelas data *testing* sama antar tetangga. Sehingga semakin tinggi jumlah ketetangaan yang diperhitungkan, banyak kemungkinan hasil prediksi semakin tidak akurat.



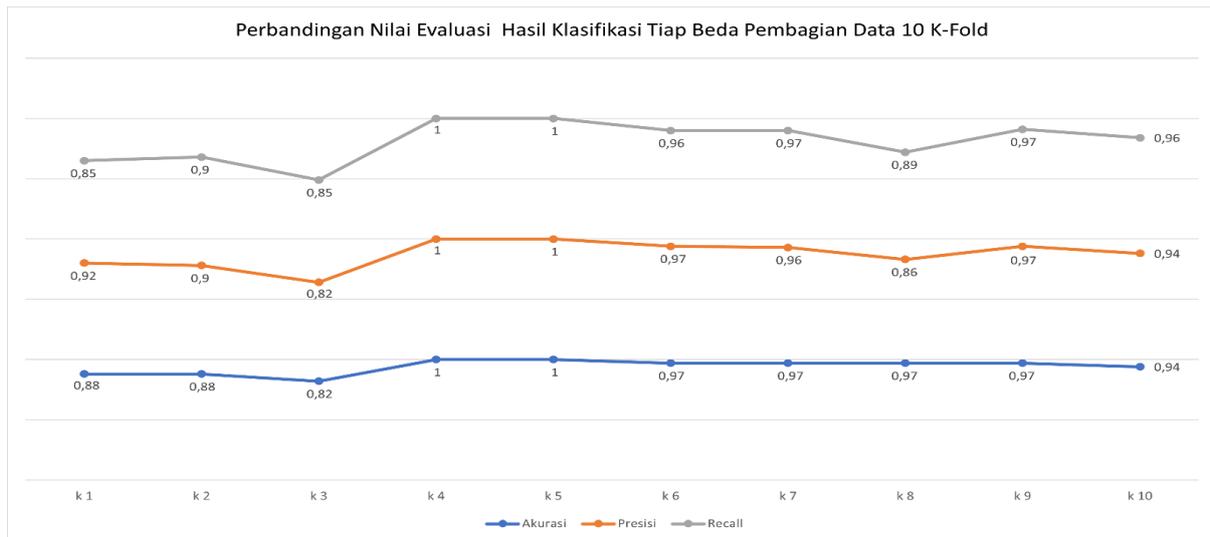
Gambar 3. 5 Grafik Hasil Evaluasi K-Fold

Selanjutnya berdasarkan grafik **Gambar 3. 5** sama dengan penjelasan sebelumnya yaitu hasil prediksi pada penerapan 10 jumlah *K-Fold Cross Validation* lebih tinggi dibandingkan dengan 5 *K-Fold Cross Validation*. Nilai rata-rata akurasi, presisi dan *recall* mencapai 94%. Sedangkan penerapan 5 *K-Fold Cross Validation* memiliki nilai rata-rata akurasi 87%, presisi 85% dan *recall* 85%. Nilai-nilai tersebut membuktikan pernyataan diatas bahwa semakin tinggi nilai pembagi *K-Fold* maka semakin tinggi pula variasi data. Tidak hanya mendapatkan variasi data yang paling baik, bisa juga terdapat kemungkinan mendapatkan variasi data yang paling buruk. Sebaliknya semakin kecil nilai pembagi *K-Fold* maka semakin kecil pula variasi data dan tidak menjamin variasi data yang didapatkan menjadi yang paling baik.



Gambar 3. 6 Grafik Hasil Evaluasi Pembagian 5 K-Fold

Selanjutnya berdasarkan grafik yang ditunjukkan pada **Gambar 3. 6** terlihat bahwa rata-rata nilai evaluasi pada $k = 4$ merupakan pembagian data yang paling stabil dibandingkan pembagian data dengan nilai k yang lain pada *5 K-Fold Cross Validation* dengan rata-rata hasil nilai akurasi 97%, presisi 95% dan *recall* 99%. Dapat disimpulkan bahwa $k = 4$ merupakan pembagian yang paling baik dengan random data *training* dan *testing* paling optimal. Dengan demikian seluruh data *training* ini dapat merepresentasikan keadaan data *testing* dengan baik. Sebaliknya nilai $k = 3$ terlihat paling rendah nilai evaluasinya dengan nilai hasil rata-rata akurasi 82%, presisi 79% dan *recall* 78%. Oleh sebab itu, pembagian pada $k = 3$ merupakan variasi random data *training* dan *testing* yang buruk, sehingga prediksi data *testing* banyak yang gagal atau salah prediksi.

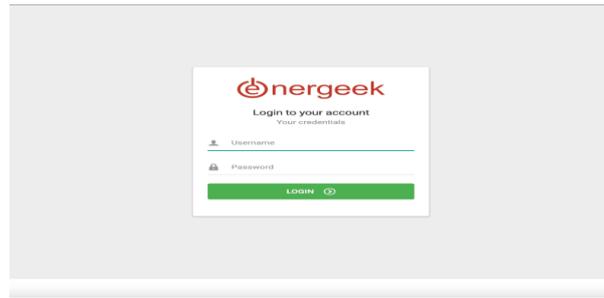


Gambar 3. 7 Grafik Hasil Evaluasi Pembagian 10 K-Fold

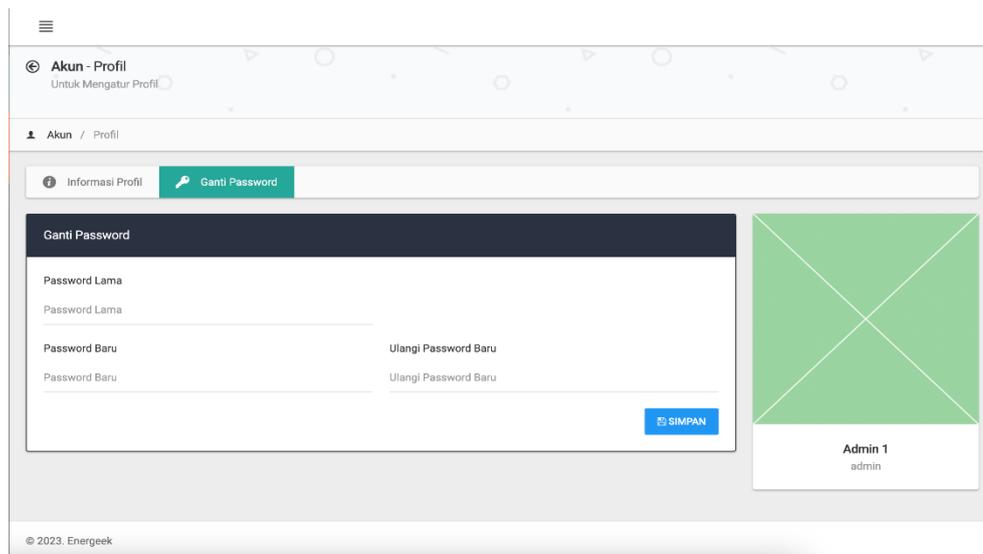
Berdasarkan grafik **Gambar 3. 7** menunjukkan bahwa rata-rata nilai evaluasi pada $k = 4$ dan $k = 5$ merupakan pembagian data yang paling stabil dibandingkan pembagian data dengan hasil dari k yang lain pada *10 K-Fold Cross Validation* dengan hasil nilai rata-rata akurasi, presisi dan *recall* adalah 100%. Dapat disimpulkan bahwa $k = 4$ dan $k = 5$ merupakan pembagian yang paling baik dengan random data *training* dan *testing* paling optimal. Artinya bahwa seluruh data *training* dapat merepresentasikan keadaan data *testing* dengan baik. Sebaliknya nilai $k = 3$ terlihat paling rendah nilai-nilai evaluasinya dengan hasil nilai rata-rata adalah akurasi 82%, presisi 82% dan *recall* 85%. Oleh sebab itu, pembagian pada $k = 3$ merupakan variasi random data *training* dan *testing* yang buruk, sehingga prediksi data *testing* banyak yang gagal atau salah prediksi. Dapat dilihat pada grafik pembagian *K-Fold Cross Validation* dengan pembagian 10 memiliki rata-rata nilai akurasi, presisi dan *recall* diatas 80% bahkan terdapat k yang menyentuh nilai 100%, jika dibandingkan dengan pembagian 5 terdapat rata-rata nilai dibawah 80%.

3.3. Antarmuka Sistem

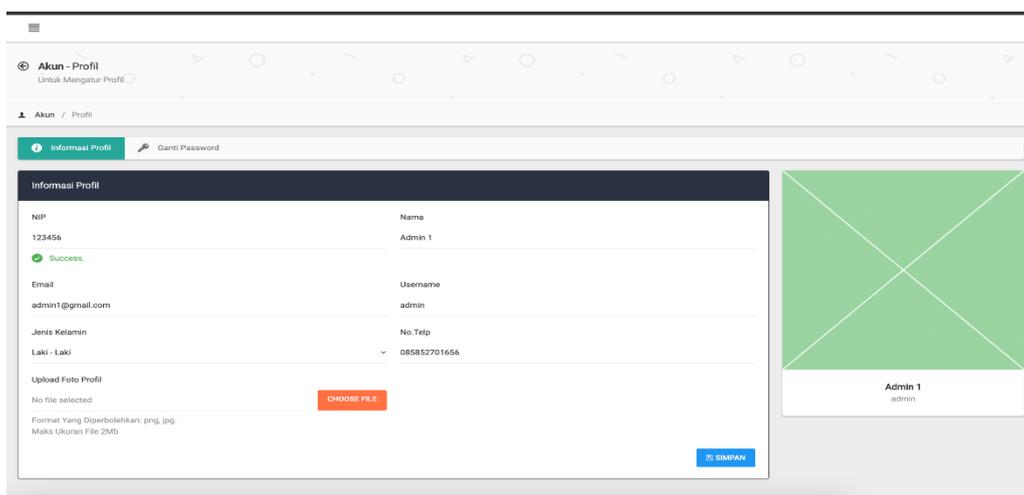
Sistem yang dibangun menggunakan bahasa pemrograman *Laravel Framework* yang berbasiskan *PHP* sedangkan untuk desain menggunakan *material design*, berikut antarmuka sistem masing-masing menu yang ada pada sistem.



Gambar 3. 8 Tampilan Login



Gambar 3. 9 Tampilan Form Ubah Password



Gambar 3. 10 Tampilan Form Ubah Profil

Master - User
Untuk Mengatur User

Master / User

User

Filter: Type to filter... Show: 10 RELOAD TAMBAH

NIP	Nama	Email	Username	Jenis Kelamin	No. Telp	Role	Aksi
100002	Rizal Kurnia Pratama	rizal.pratama@energeek.co.id	rizal	Laki - Laki	087267167467	Tim Penilai	⋮
100003	Rohmat Hanif Basuki	hanif@energeek.co.id	hanif	Laki - Laki	083172857321	Tim Penilai	⋮
100001	Zulmi Adi Rizki	zulmi.ar@energeek.co.id	zulmi	Laki - Laki	083716746726	Admin	⋮

Showing 1 to 3 of 3 entries

- Edit
- Detail
- Hapus

© 2023. Energeek

Gambar 3. 11 Tampilan Tabel User

Master - User
Untuk Mengatur User

Master / User

User

NIP

Nama

Email

Username

Password

Konfirmasi Password

Jenis Kelamin

No. Telp

Role

Pilih Role

SIMPAN BERSIHKAN

Gambar 3. 12 Tampilan Form User

Master - Keahlian
Untuk Mengatur Keahlian

Master / Keahlian

Keahlian

Filter: Integrasi Show: 10 RELOAD TAMBAH

Nama	Aksi
Integrasi Firebase Notification	⋮
Integrasi SMTP	⋮
Integrasi UI Ke API	⋮

Showing 1 to 3 of 3 entries (filtered from 16 total entries)

- Edit
- Detail
- Hapus

1

© 2023. Energeek

Gambar 3. 13 Tampilan Tabel Keahlian

Master - Keahlian
Untuk Mengatur Keahlian

Master / Keahlian

Keahlian

Nama
Nama

Deskripsi
Deskripsi

SIMPAN BERSIHKAN

Gambar 3. 14 Tampilan Form Keahlian

Master - Jabatan
Untuk Mengatur Jabatan

Master / Jabatan

Jabatan

Filter: Type to filter... RELOAD + TAMBAH

Nama	Aksi
Back-End	
Front-End	

Showing 1 to 2 of 2 entries

- Edit
- Detail
- Hapus
- Tes Teknis

© 2023. Energeek

Gambar 3. 15 Tampilan Tabel Jabatan

Master - Jabatan
Untuk Mengatur Jabatan

Master / Jabatan

Jabatan

Nama Link Tes Teknis
Nama Link Tes Teknis

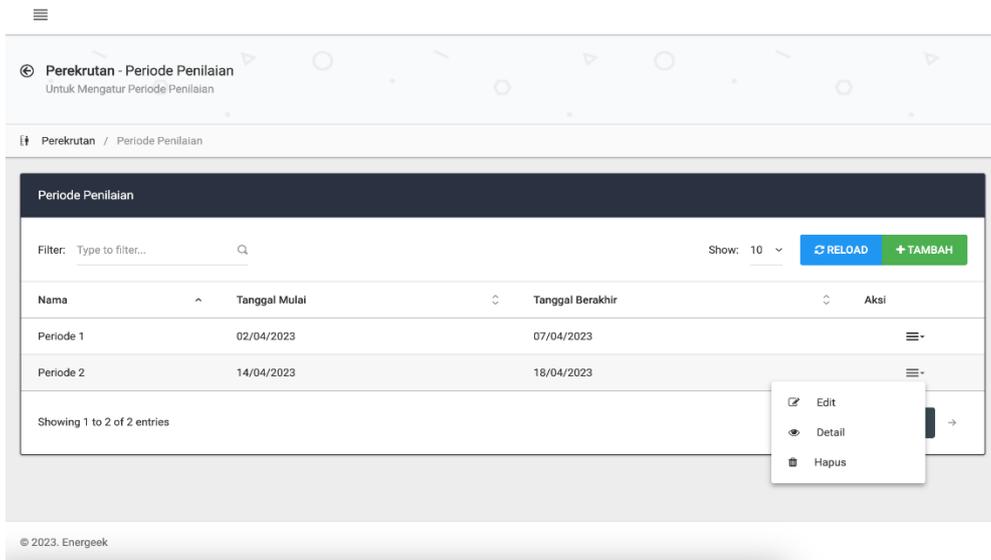
Deskripsi
Deskripsi

Keahlian	Skor	Aksi
Pilih Keahlian	Skor	
Pilih Keahlian	Skor	

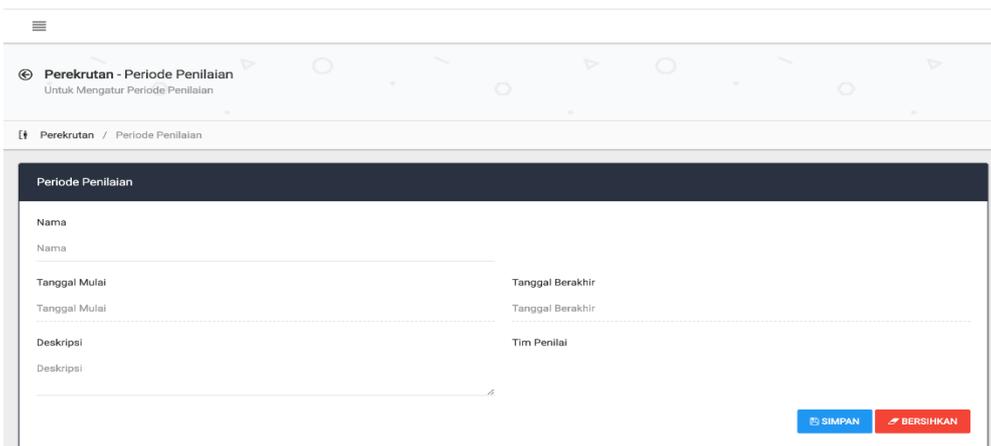
Total Skor 0/100 Kategori

SIMPAN BERSIHKAN

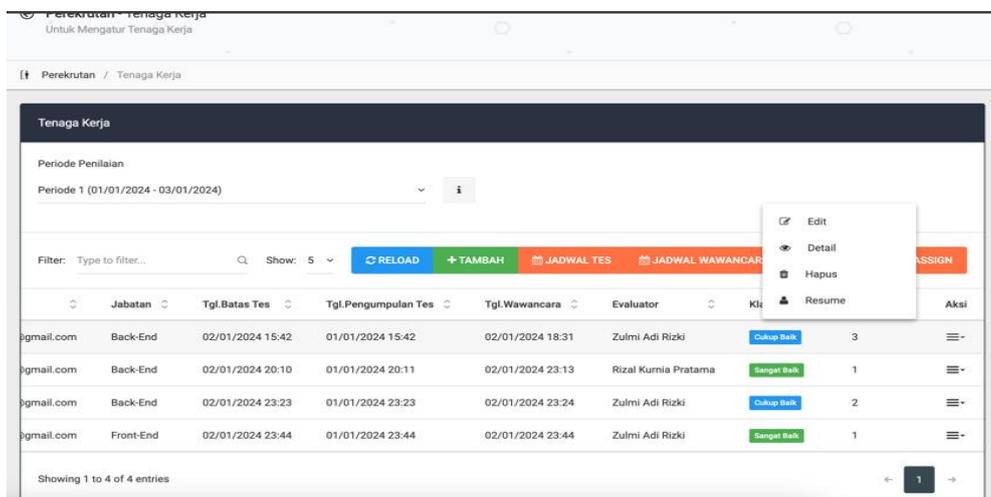
Gambar 3. 16 Tampilan Form Jabatan



Gambar 3. 17 Tampilan Tabel Periode Penilaian



Gambar 3. 18 Tampilan Form Periode Penilaian



Gambar 3. 19 Tampilan Tabel Tenaga Kerja

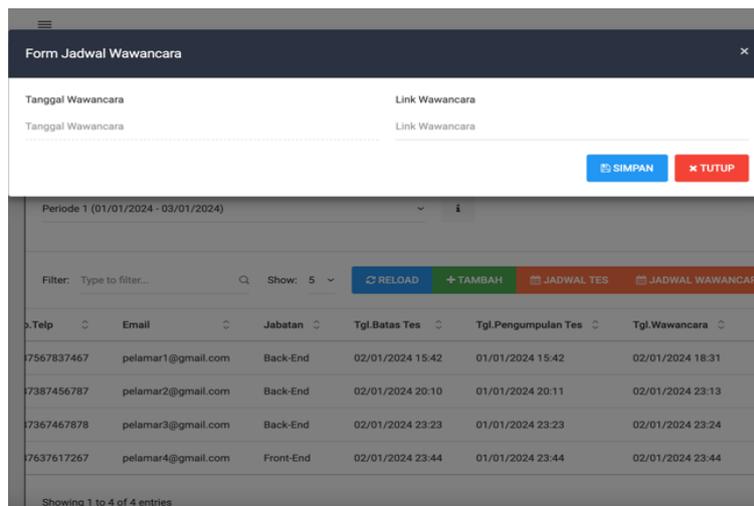
Gambar 3. 20 Tampilan Form Tenaga Kerja

No.Telp	Email	Jabatan	Tgl.Batas Tes	Tgl.Pengumpulan Tes	Tgl.Wawancara
7567837467	pelamar1@gmail.com	Back-End	02/01/2024 15:42	01/01/2024 15:42	02/01/2024 18:31
7387456787	pelamar2@gmail.com	Back-End	02/01/2024 20:10	01/01/2024 20:11	02/01/2024 23:13
7367467878	pelamar3@gmail.com	Back-End	02/01/2024 23:23	01/01/2024 23:23	02/01/2024 23:24
7637617267	pelamar4@gmail.com	Front-End	02/01/2024 23:44	01/01/2024 23:44	02/01/2024 23:44

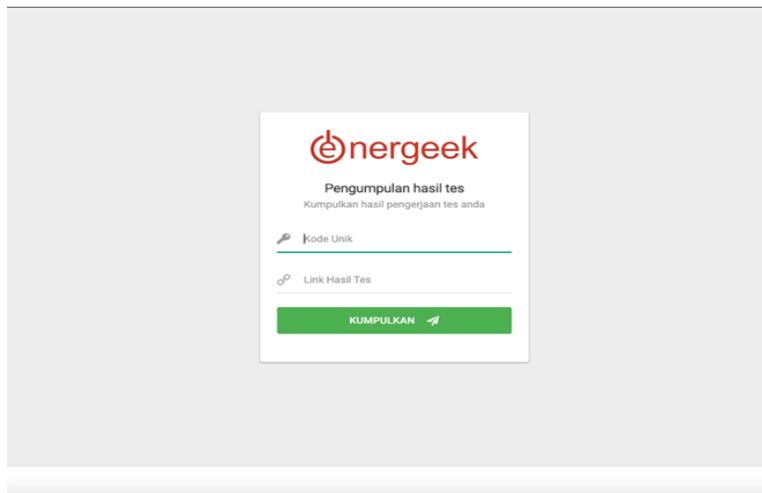
Gambar 3. 21 Tampilan Form Assignment Tim Penilai

No.Telp	Email	Jabatan	Tgl.Batas Tes	Tgl.Pengumpulan Tes	Tgl.Wawancara
7567837467	pelamar1@gmail.com	Back-End	02/01/2024 15:42	01/01/2024 15:42	02/01/2024 18:31
7387456787	pelamar2@gmail.com	Back-End	02/01/2024 20:10	01/01/2024 20:11	02/01/2024 23:13
7367467878	pelamar3@gmail.com	Back-End	02/01/2024 23:23	01/01/2024 23:23	02/01/2024 23:24
7637617267	pelamar4@gmail.com	Front-End	02/01/2024 23:44	01/01/2024 23:44	02/01/2024 23:44

Gambar 3. 22 Tampilan Form Jadwal Tes



Gambar 3. 23 Tampilan Form Jadwal Wawancara



Gambar 3. 24 Tampilan Form Pengumpulan Hasil Tes

Gambar 3. 25 Tampilan Form Penilaian

3.4. Pengujian System Usability Scale

Pada pengujian *SUS* ini guna menilai seberapa besar kegunaan dari sebuah sistem yang telah dibuat maka penulis telah mengumpulkan beberapa jawaban dari responden melalui formulir kuesioner yang berisi 10 pertanyaan yang ditunjukkan pada **Tabel 3. 2** [21].

Tabel 3. 2 Komponen Pertanyaan *SUS*

No	Komponen Pertanyaan
1	Saya berpikir akan menggunakan sistem ini lagi.
2	Saya merasa sistem ini rumit untuk digunakan.
3	Saya merasa sistem ini mudah digunakan.
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini.
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini).
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.
8	Saya merasa sistem ini membingungkan.
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini.
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.

Jawaban dari setiap responden nantinya akan dilakukan perhitungan menggunakan rumus (2. 5). Berikut data skor akhir jawaban dari responden yang berhasil di dapatkan yang berjumlah 12 responden yang selanjutnya dilakukan rata-rata terhadap semua skor akhir responden untuk mendapatkan kategori nilai *SUS*.

Tabel 3. 3 Jawaban Kuesioner *SUS*

No	Nama	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	Skor
1	Riska	5	1	5	1	5	1	5	1	5	3	95
2	Chafid	5	1	5	1	5	1	5	1	5	1	100
3	Farida	4	5	5	2	5	1	5	1	4	4	75
4	Diiky	5	1	5	1	5	1	5	1	5	1	100
5	Yuri	5	1	5	5	5	1	4	2	4	5	72,5
6	Gama	4	1	5	5	5	2	4	1	4	4	72,5

No	Nama	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	Skor
7	Rizky	1	1	1	2	5	1	5	2	5	2	72,5
8	Ikbar	5	1	5	3	4	2	5	2	5	3	82,5
9	Tama	4	2	4	1	5	2	4	2	4	2	80
10	Hayu	5	1	5	4	5	1	4	5	5	5	70
11	Hafied	4	3	4	4	4	3	4	3	4	4	57,5
12	Aloy	4	2	4	4	4	2	4	2	4	4	85
Rata - Rata											80,21	

Dari hasil rata-rata diatas diperoleh nilai sebesar 80,21 yang jika dilihat pada **Tabel 2. 2** maka kategori nilai yang di dapat untuk kegunaan sistem yang telah dibuat adalah B.

IV. KESIMPULAN

Penelitian ini memiliki beberapa kesimpulan dari hasil penelitian yang telah dilakukan. Aplikasi dari hasil uji *SUS* mendapatkan rata-rata nilai dari keseluruhan responden sebesar 80,21 yang dimana skor tersebut masuk kedalam kategori nilai B yang bisa dikatakan dari segi kegunaan dan kemudahan yang dirasakan oleh responden cukup memenuhi. Pada pengujian algoritma *KNN* menggunakan perhitungan jarak *euclidean distance* didapatkan model terbaik dengan nilai rata-rata akurasi, presisi, *recall* sebesar 100% pada pembagian $k = 4$ dan $k = 5$ di uji coba pengambilan *10 K-Fold Cross Validation*, sedangkan nilai ketetangaan metode *KNN* paling baik ada di $k = 7$. Hasil penelitian menunjukkan bahwa masih ada beberapa kekurangan, seperti fitur yang dimiliki oleh sistem masih sangat sedikit dikarenakan dalam pengembangannya disesuaikan dengan kebutuhan saat penelitian berlangsung, akan tetapi tidak menutup kemungkinan akan terjadi peningkatan kebutuhan dalam perusahaan yang mengharuskan terjadinya penambahan fitur atau penyesuaian terhadap fitur sebelumnya. Dataset yang digunakan dalam penelitian ini juga sedikit sehingga dibutuhkan penambahan jumlah dataset agar tingkat akurasi dalam proses klasifikasi lebih tinggi dari sebelumnya.

DAFTAR PUSTAKA

- [1] H. Chandra dan S. Tinggi Ilmu Ekonomi YPUP Makassar, “ANALISIS SISTEM SELEKSI PENERIMAAN KARYAWAN TERHADAP KINERJA KARYAWAN PADA PT ANUGRAH BUSANA INDAH DI MAKASSAR,” 2020.
- [2] Y. Yanti, Gunawan A, dan R. Firdaos, “Rekrutmen, Seleksi dan Penempatan Pegawai,” *Jurnal Ilmiah Wahana Pendidikan*, vol. 9, no. 10, hlm. 676–688, 2023, doi: 10.5281/zenodo.7991063.
- [3] A. Rahmat, K. Auliasari, dan Y. A. Pranoto, “IMPLEMENTASI METODE K-NEAREST NEIGHBOR (KNN) UNTUK SELEKSI CALON KARYAWAN BARU (Studi Kasus : BFI Finance Surabaya),” 2020.
- [4] A. M. Argina, “Indonesian Journal of Data and Science Penerapan Metode Klasifikasi K-Nearest Neighbor pada Dataset Penderita Penyakit Diabetes,” vol. 1, no. 2, hlm. 29–33, 2020.
- [5] I. G. T. Isa dan F. Elfaladonna, “Penilaian Kinerja Akurasi Metode Klasifikasi dalam Dataset Penerimaan Mahasiswa Baru Universitas XYZ,” *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, vol. 8, no. 2, hlm. 292–298, 2022, [Daring]. Tersedia pada: <https://jurnal.untan.ac.id/index.php/jepin/article/view/54316>
- [6] A. Sumiah dan N. Mirantika, “Perbandingan Metode K-Nearest Neighbor dan Naive Bayes untuk Rekomendasi Penentuan Mahasiswa Penerima Beasiswa pada Universitas Kuningan,” vol. 6, 2020, [Daring]. Tersedia pada: <https://journal.uniku.ac.id/index.php/buffer>
- [7] N. Marito Putry dan B. Nurina Sari, “KOMPARASI ALGORITMA KNN DAN NAÏVE BAYES UNTUK KLASIFIKASI DIAGNOSIS PENYAKIT DIABETES MELITUS,” *Jurnal Sains dan Manajemen*, vol. 10, no. 1, 2022.
- [8] G. W. Sasmito, J. T. Informatika, H. Bersama, J. Mataram, N. 09, dan P. Lor, “Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal,” vol. 2, no. 1, 2017, [Daring]. Tersedia pada: <http://www.tegalkab.go.id>,
- [9] M. Badrul, “PENERAPAN METODE WATERFALL UNTUK PERANCANGAN SISTEM INFORMASI INVENTORY PADA TOKO KERAMIK BINTANG TERANG,” vol. 8, no. 2, 2021.

- [10] R. D. Sitepu dan E. Buulolo, “KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer) IMPLEMENTASI ALGORITMA NEAREST NEIGHBOR PADA PENERIMAAN PEGAWAI BARU PADA MTS IKHWANUTS TSALITS TALUN KENAS”, [Daring]. Tersedia pada: <http://ejurnal.stmik-budidarma.ac.id/index.php/komik>
- [11] W. Wahyono, I. N. P. Trisna, S. L. Sariwening, M. Fajar, dan D. Wijayanto, “Comparison of distance measurement on k-nearest neighbour in textual data classification,” *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 1, hlm. 54–58, Jan 2020, doi: 10.14710/jtsiskom.8.1.2020.54-58.
- [12] R. B. Anshori *dkk.*, “Klasifikasi Citra Kanker Serviks Menggunakan Deep Residual Network,” vol. 8, no. 6, hlm. 3163–3170, 2022.
- [13] E. Gokgoz dan A. Subasi, “Comparison of decision tree algorithms for EMG signal classification using DWT,” 2015.
- [14] T. Hastie, R. Tibshirani, dan J. Friedman, “Springer Series in Statistics The Elements of Statistical Learning Data Mining, Inference, and Prediction.”
- [15] P. F. Laravel *dkk.*, “PEMANFAATAN FRAMEWORK LARAVEL DAN FRAMEWORK BOOTSTRAP PADA PEMBANGUNAN APLIKASI PENJUALAN HIJAB BERBASIS WEB,” *Jurnal Media Infotama*, vol. 18, no. 1, hlm. 2022.
- [16] D. Purnama Sari, R. Wijanarko, dan J. X. Menoreh Tengah, “Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus Di Rumah Kamera Semarang),” vol. 2, no. 1, hlm. 32–36, 2019.
- [17] I. Listiowarni dan N. P. Dewi, “Pemanfaatan Klasifikasi Soal Biologi Cognitive Domain Bloom’s Taxonomy Menggunakan KNN Chi-Square Sebagai Penyusunan Naskah Soal,” *Digital Zone: Jurnal Teknologi Informasi Dan Komunikasi*, vol. 11, no. 2, hlm. 185–195, 2020.
- [18] D. W. Ramadhan, B. Soedijono, dan E. Pramono, “PENGUJIAN USABILITY WEBSITE TIME EXCELINDO MENGGUNAKAN SYSTEM USABILITY SCALE (SUS) (STUDI KASUS: WEBSITE TIME EXCELINDO).” [Daring]. Tersedia pada: <https://excelindo.co.id>
- [19] A. Bangor, P. Kortum, dan J. Miller, “Determining what individual SUS scores mean: Adding an adjective rating scale,” *J Usability Stud*, vol. 4, no. 3, hlm. 114–123, 2009.
- [20] K. ’ Afiifah, Z. Fira Azzahra, A. D. Anggoro, D. Redaksi, R. Akhir, dan D. Online, “Universitas Negeri Jakarta; Jl. Rawamangun Muka Raya No.11 RW.14 Rawamangun,” *JURNAL INTECH*, vol. 3, no. 1, hlm. 8–11.
- [21] B. Said dan F. Prasetyo E.P., “Layanan e-Surat Berbasis Mobile Application di Desa Waru Barat Pamekasan,” *InfoTekJar (Jurnal Nasional Informatika dan Teknologi Jaringan)*, vol. 4, no. 1, hlm. 111–115, Sep 2019, doi: 10.30743/infotekjar.v4i1.1637.