



IMPLEMENTASI METODE K-NEAREST NEIGHBOR (K-NN) PADA SISTEM PENDUKUNG KEPUTUSAN UNTUK PENEMPATAN SISWA KE GRADE DAN LEVEL YANG TEPAT DALAM PROGRAM DI DIGIKIDZ BSD CITY

Muhammad Farhan Alyasis¹, Alvino Octaviano²

^{1,2}Fakultas Ilmu Komputer, Universitas Pamulang

¹frhanalyasiz@gmail.com

ABSTRACT

Placing students into appropriate grades and levels is a crucial aspect of the educational process, ensuring that each student receives instruction tailored to their abilities and needs. Digikidz, as a School of Technology, focuses on developing children's technological skills through various programs aligned with their capabilities and interests, making the process of grade and level placement more complex. This study aims to implement the K-Nearest Neighbor (K-NN) method as a web-based decision support system to assist in accurately placing students into suitable grades and levels. The web-based system leverages K-NN for its ability to classify data based on attribute similarity, utilizing student data from trial classes, such as task completion duration, interaction with instructors, creativity, and concentration. This platform is expected to enhance the accuracy and efficiency of student placement at Digikidz BSD City, while also enabling real-time access and decision-making. The implementation results indicate that the K-NN method can provide accurate and effective student placement recommendations, achieving an accuracy rate of 91.11%. This web-based system offers a more structured and accessible platform, optimizing the student placement process at Digikidz through personalized learning pathways.

Keywords: *K-Nearest Neighbor, Classification, Student Placement, Decision Support System, Grade and level*

ABSTRAK

Penempatan siswa ke dalam grade dan level yang sesuai merupakan aspek penting dalam proses pendidikan, karena memastikan setiap siswa mendapatkan pembelajaran yang disesuaikan dengan kemampuan dan kebutuhannya. Digikidz, sebagai School of Technology, berfokus pada pengembangan keterampilan teknologi anak-anak melalui berbagai program yang disesuaikan dengan kemampuan dan minat mereka, menjadikan proses penempatan grade dan level lebih kompleks. Penelitian ini bertujuan untuk mengimplementasikan metode K-Nearest Neighbor (K-NN) sebagai sistem pendukung keputusan berbasis web untuk membantu penempatan siswa ke dalam grade dan level yang tepat. Sistem berbasis web ini memanfaatkan K-NN karena kemampuannya dalam mengklasifikasikan data berdasarkan kemiripan atribut, dengan memanfaatkan data siswa hasil trial class, seperti durasi penyelesaian tugas, interaksi dengan

Article History:

Received: June 2025

Reviewed: June 2025

Published: June 2025

Plagirism Checker No 234

Prefix DOI:

10.8734/Kohesi.v1i2.365

Copyright: Author

Publish by: Kohesi



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



<p>instruktur, kreativitas, dan konsentrasi. Platform berbasis web ini diharapkan dapat meningkatkan akurasi dan efisiensi penempatan siswa di Digikidz BSD City, serta memungkinkan akses dan pengambilan keputusan secara real-time. Hasil implementasi menunjukkan bahwa metode K-NN mampu memberikan rekomendasi penempatan siswa yang akurat dan efektif, dengan akurasi mencapai 91.11%. Sistem berbasis web ini menyediakan platform yang lebih terstruktur dan mudah diakses, yang mengoptimalkan proses penempatan siswa di Digikidz melalui pembelajaran yang dipersonalisasi.</p> <p>Kata Kunci: K-Nearest Neighbor (K-NN), Sistem Pendukung Keputusan, Website, Penempatan Siswa, Grade dan Level</p>	
--	--

A. PENDAHULUAN

Dalam dunia pendidikan modern, khususnya pendidikan berbasis teknologi, proses penempatan siswa ke dalam level pembelajaran yang sesuai merupakan langkah krusial yang dapat memengaruhi kualitas pengalaman belajar seorang siswa. Penempatan yang tepat memungkinkan siswa untuk belajar sesuai dengan tingkat kemampuan, minat, serta kebutuhan mereka secara individual. Hal ini tidak hanya berkontribusi pada peningkatan hasil belajar, tetapi juga mendorong rasa percaya diri, motivasi, dan keterlibatan siswa dalam proses pembelajaran.

DIGIKIDZ School of Technology, yang telah berdiri sejak tahun 2001, merupakan lembaga pendidikan teknologi kreatif yang fokus pada pengembangan keterampilan teknologi digital untuk anak-anak. Hingga saat ini, DIGIKIDZ telah berkembang pesat dengan lebih dari 20 center cabang dan bekerja sama dengan 90+ partner school di berbagai kota besar di Indonesia. Salah satu cabang utamanya, DIGIKIDZ BSD City, menjalankan program pembelajaran teknologi dengan kurikulum yang disesuaikan untuk anak-anak berdasarkan rentang usia dan tingkat kemampuan mereka.

Untuk menyesuaikan materi pembelajaran dengan kemampuan masing-masing siswa, DIGIKIDZ BSD City mengelompokkan siswa ke dalam tiga level utama, yaitu Little Creator, Junior, dan Teenager. Level Little Creator ditujukan bagi siswa berusia 4-6 tahun yang baru mengenal dunia teknologi dan masih berada pada tahap pengenalan perangkat serta logika dasar. Sementara itu, level Junior diperuntukkan bagi siswa berusia 7-8 tahun yang telah memiliki pemahaman dasar dan mulai menunjukkan keterampilan berpikir logis serta kreativitas dalam menggunakan teknologi. Adapun level Teenager ditujukan untuk siswa berusia 9-16 tahun yang telah siap menghadapi tantangan teknologi yang lebih kompleks, seperti coding, animasi, dan robotik.

Meskipun pengelompokan ini telah dibagi berdasarkan usia, realitanya tidak semua siswa berkembang dengan cara dan kecepatan yang sama. Ada siswa yang masih muda namun sudah sangat mahir dalam hal teknologi, dan sebaliknya, ada pula siswa yang lebih tua namun masih memerlukan pendekatan dasar dalam belajar teknologi. Oleh karena itu, hanya mengandalkan usia sebagai indikator penempatan siswa sering kali menimbulkan ketidaktepatan dalam menentukan level belajar siswa.

Kondisi tersebut menimbulkan kebutuhan akan sistem yang dapat membantu pengambilan keputusan secara lebih objektif dan berdasarkan data yang komprehensif. Dalam hal ini, Sistem Pendukung Keputusan (SPK) dapat menjadi solusi untuk mengatasi permasalahan penempatan siswa yang kompleks. SPK bertujuan untuk membantu pihak institusi dalam mengevaluasi dan menentukan level siswa berdasarkan kombinasi beberapa faktor seperti usia, durasi penyelesaian tugas, interaksi dengan instruktur, kreativitas, dan konsentrasi siswa.



Salah satu metode yang dapat diterapkan dalam SPK untuk mengatasi masalah klasifikasi ini adalah metode K-Nearest Neighbor (K-NN). K-NN pertama kali diperkenalkan oleh Evelyn Fix dan Joseph Hodges pada tahun 1951 sebagai metode non-parametrik untuk klasifikasi pola. K-NN merupakan algoritma pembelajaran mesin yang bekerja berdasarkan prinsip kemiripan antar data. Metode ini akan menganalisis data siswa baru dan membandingkannya dengan data siswa yang sudah ada, kemudian mengklasifikasikannya ke level yang paling sesuai berdasarkan atribut-atribut tertentu. Metode ini dipilih karena KNN mampu melakukan klasifikasi secara sederhana namun efektif berdasarkan kedekatan data, sehingga cocok digunakan untuk menentukan grade dan level kemampuan anak berdasarkan parameter seperti usia, durasi penyelesaian, kreativitas, interaksi, dan konsentrasi. Menurut penelitian oleh Riza Purwandi (2023), K-NN telah terbukti efektif dalam mengklasifikasikan mahasiswa untuk memprediksi kelulusan praktikum laboratorium berdasarkan nilai dan parameter pendukung lainnya. Hal yang serupa juga diungkapkan oleh Widiastuti et al. (2023), yang menyatakan bahwa K-NN sangat efektif digunakan dalam klasifikasi berbasis data siswa.

Dengan mengimplementasikan metode K-NN ke dalam sistem penempatan siswa di DIGIKIDZ BSD City, diharapkan proses penentuan level dapat dilakukan secara lebih akurat, efisien, dan terukur. Penggunaan algoritma ini juga dapat mengurangi subjektivitas dalam pengambilan keputusan, memastikan bahwa setiap siswa ditempatkan pada level yang paling tepat dengan mempertimbangkan berbagai indikator kemampuan belajar. Sistem ini diharapkan tidak hanya membantu staf pengajar dalam proses klasifikasi, tetapi juga memberikan manfaat besar bagi siswa dengan menciptakan lingkungan belajar yang kondusif dan sesuai dengan potensi masing-masing individu.



B. Metode Penelitian Seleksi data (*Data Selection*)

Data dalam database tidak selalu digunakan sepenuhnya, sehingga hanya data yang relevan untuk analisis akan diambil. Dalam penelitian ini, data yang digunakan mencakup hasil trial siswa dengan parameter seperti Durasi Penyelesaian, Interaksi dengan Instruktur, Kreativitas, dan Konsentrasi. Sampel data yang dipilih akan melalui proses seleksi agar sesuai dengan kebutuhan analisis, dan contoh data ini dapat dilihat pada gambar di bawah ini:

No.	Nama	Umur	Sekolah	Durasi Penyelesaian	Interaksi dengan Instruktur	Kreativitas	Konsentrasi	Grade
1	Ethan Dharma Tjara	17	DRK SMP BSD	2	3	3	2	Junior
2	Carissa Liana Adhara	17	SMA	2	2	3	2	Junior
3	Umair Al Farhan	17	Estu Alama	2	2	3	2	Late Center
4	Patri Bekan	4.5	TK Pangs	2	3	2	2	Late Center
5	Haidir Rizki Haidir	4.5	TK Pangs	3	2	3	3	Late Center
6	Fera Dora Alhama	11	SMA Al Adhar 1 Panti	3	3	3	3	Teenager
7	Rafaely Ghani Pratomo	17	SD Bina Mulya #8	2	3	2	3	Junior
8	Fitri	17	SD	3	2	3	2	Junior
9	Nalya Alvin Septian	10	SD Muhammadiyah 3 Jakarta	3	3	2	2	Junior
10	Alvino Rakhya Hanah	8	Al Adhar Syih Bina Pendidikan	2	3	3	3	Late Center
11	Alhyan Rakhya Kamil	8	Al Adhar Syih Bina Pendidikan	3	3	2	2	Junior
12	Ayu Rakhya Cahya	8	SD Al Adhar Syih Bina Pendidikan	2	3	3	2	Junior
13	Herkel Rakhya Pratomo	11	Al Adhar	3	3	3	3	Teenager
14	Alvinia Ruffi Alvinia	17	TK Al Adhar Syih Bina Pendidikan	2	2	3	2	Late Center
15	Henny Wipandi	13	OSK SMPN	3	2	3	3	Teenager
16	Fadhil Rasyid Numpika	13	SMPN 8 Tanggung Selatan	3	3	3	3	Teenager
17	Adha	15	GIS 2 Samping	3	2	2	3	Late Center
18	Carlo Zavier Kusuma	17	Sekolah Mase BSD	2	2	3	2	Late Center
19	El Shalwin Alhady	17	SD 1 Jember Kudu	3	3	3	3	Junior
20	Gergorius Rakhil Adhyan	10	SDN Si Kerdawa II	3	3	2	3	Junior
21	Fera Dora Alhama	10	SMA Al Adhar 1 Panti	3	3	3	3	Teenager
22	Alhar Alhady Tharino	9	SD Alah BSD	2	3	3	2	Junior
23	Alhar Alhady Tharino	9	SD Alah BSD	2	3	3	3	Junior
24	Hugo Rival Handjama	10	John Paul School	3	3	2	2	Junior
25	Fera Alhady Cahel	10	SDT Norkanti	2	3	2	3	Late Center
26	Fau Yuhana Alhendana	10	SDT Riza Sekeloa	3	3	2	3	Late Center
27	Rizky Karna Alhady	10	Kudat SBS	3	3	2	3	Junior
28	Kalish Haps Nidhiya	6.87	Sekolah Hikan	2	3	3	2	Junior
29	Kevan Wilam Cusi	7	SD Litar Bina, Sema	3	2	2	3	Junior
30	Rany Wahanda	10	Kudat, BMS Prany	3	3	3	2	Junior
31	Akhil Alvinia Hanika	10	Khalid Ima Prany	3	3	3	3	Junior
32	Muhammad Rafi Akumana	8	SD Islam Bantab BSD	2	3	3	2	Junior
33	Muhammad Alvin Akumana	8	SD Islam Bantab BSD	3	3	2	2	Late Center
34	Evanita Valantia	11	Sekolah Cahat Semping	3	3	2	3	Teenager
35	Aryant Alhady	11	TK Kemari Cemping Tengah	2	3	3	2	Late Center
36	Eka Christin Chua	4	Sekolah Pida Harapan	3	3	2	3	Late Center
37	Lia Thi Naghita	8	SDK Pambur Bantab Jaya	2	3	3	2	Junior
38	Gwen Muladita	10	Sekolah Mase School Gading Prany	3	3	2	3	Junior
39	Iryensudin Dhuha Shandi	10	SD Bina Mulya #8	2	3	2	2	Late Center
40	Josh Wrenny Harwanan	17	TKA, Pambur Gading Prany	3	3	3	3	Late Center
41	Mah Inara Harwanan	17	SDK Pambur Gading Prany	3	3	3	2	Junior
42	Harapan Analis	17	Sekolah Mase School Gading Prany	2	2	3	2	Junior
43	Tarung Andira Wilhena	17	SD Pambur Kera Mardian	2	2	3	3	Junior
44	Willem Geovani Satrik	8	Arabi School	2	3	3	3	Late Center
45	Lavel Timothy Yong	8	BPK Pambur Kera Mardian	3	3	2	3	Junior
46	Jovin Felicia Denna	8	Sekolah Kristen Pambur BSD	3	2	3	3	Junior
47	Muhammad Tamar Radhy Ward	17	SD Terang Bangas	3	2	2	2	Late Center
48	Zilan Hagi Dhuha Adhyan	8	Harapan Bangas School	3	3	2	3	Junior
49	Rana Alhady Panti	10	SD Panti Semping	2	3	3	2	Junior
50	Saria Kusuma Bakara	10	Sekolah Panti Semping	3	3	2	3	Junior
51	Taka Christin Lark	10	SD Panti Harapan BSD	2	3	3	2	Junior
52	Bridita Yusef Anany	8	SMA Harapan Bangas	3	2	3	3	Junior
53	Zahra Alvin Alvin	10	Sekolah Cahel	3	3	2	3	Junior
54	Nahdiah Ivan Gabriel	8	TKA Sekeloa Prany	2	2	3	3	Late Center
55	Jarlan Alhady Satrik	17	Sekolah Pida Harapan	3	3	3	2	Junior
56	Daniel Oscar Wasky	6	Sekolah Kristen BPK Pambur BSD	3	3	3	3	Late Center
57	Kelvin Nalya Nurulhuda	10	Sekolah Pida Harapan BSD	2	2	3	2	Junior
58	Rani Alhady	17	SD Panti Harapan BSD	3	3	2	2	Junior
59	Genta Ghani Semping	17	Sekolah Pida Harapan	2	2	3	3	Junior
60	Karna Alhady	10	Sekolah Cahel BSD	3	3	2	2	Junior
61	Fitri Muhammad Rani	10	BPK Pambur BSD	3	3	3	3	Junior
62	Rafael Alexander Nuzli	8	Hanus Kanti-Kanti Alah	3	3	2	3	Late Center
63	Muhammad Rakhil Hanani	8	Harapan Bangas School	3	3	3	2	Junior
64	Nasrah Fata Anah	7	sd Muhammadiyah 3 Jakarta	2	3	3	3	Junior
65	Laraswati Dora Pratomo	8	Sekolah Kanti-Kanti Alah	3	3	2	3	Junior
66	Nadha Zahra Alhady	3	TK Al Adhar Syih Bina Pendidikan	2	3	3	2	Late Center
67	Felicia Nurana	6	SDN Pondok Asem	2	2	2	3	Late Center
68	Billy Raphael	6	Sekolah Kristen Pambur BSD	2	2	2	3	Late Center
69	Cedric Emmanuel Nidhan	15	St. Peter's School	3	2	3	2	Late Center
70	Adhira Karna Anah	8	St. Peter's School	3	3	3	2	Junior
71	Arya Dhuha Win	17	Sekolah Kristen Alah Alah	2	2	3	3	Late Center
72	Alvin Alhady	8	SMA Semping	2	3	2	3	Junior
73	Rani Arya Adhyan	17	Sekolah Pida Harapan BSD	2	3	3	3	Junior
74	Archi Alvin Adhyan	17	Harapan Bangas School	3	3	3	2	Late Center
75	Alvin Alvin Panti	8	Hanus Kanti-Kanti Alah	3	2	2	3	Late Center
76	Eva Vincent Suantra	6	Harapan Bangas School	2	2	3	3	Junior
77	Valencia Sophia Rosalia	17	SD Al Adhar Syih Bina Pendidikan	3	3	3	2	Junior
78	Rani Alhady	6	SDI Alah Prany	3	3	3	2	Junior
79	Nora Alhady Dhuha	17	TK Panti Harapan BSD	3	3	3	3	Late Center
80	Fera Dora Alhama	10	SD 1 Jember Jaya	3	3	3	3	Late Center

Gambar 3.1 Seleksi Data



Keterangan:

- Durasi Penyelesaian: Penilaian Durasi penyelesaian project murid dalam kegiatan belajar.
- Interaksi dengan Instruktur: Penilaian interaksi siswa dengan instruktur selama proses pembelajaran.
- Konsentrasi: Penilaian konsentrasi murid dalam kegiatan belajar.
- Kreatifitas: Penilaian kreativitas yang ditunjukkan oleh siswa selama kegiatan belajar.

Transformasi data (Data Transformation)

Data diubah atau dikombinasikan ke dalam format yang sesuai agar dapat diproses lebih lanjut. Langkah ini disebut *data transformation* dan bertujuan untuk menyesuaikan data dengan kebutuhan analisis. Contoh data yang digunakan dalam tahap transformasi dapat dilihat pada gambar di bawah ini, di mana data mentah diproses untuk meningkatkan kompatibilitas dan keakuratannya dalam perhitungan selanjutnya.

No	Nama	Umur	Sekolah	Durasi	Interaksi	Kreativitas	Konsentrasi	Grade/Lev
1	Lykan Dominic Tjoe	7	DIGK/IDE BSD	3	2	3	2	1
2	Carmelo Sam Lukardi	7	SMM	3	3	2	2	1
3	Umar Al Farsuq	5	Exiss Abata	2	2	3	2	0
4	Putri Rizka	4,5	TK Pelangi	2	3	2	3	0
5	Habil Ar Razi Hendrik	4,5	TK Pelangi	2	2	3	3	0
6	Faza Bari Abhista	36	SMA Al Athar 1 Pusat	2	2	2	3	2
7	Rafasya Gibran Pratama	9	SD Bakti Mulia 400	3	3	3	2	1
8	Rina	7	Vlaney	3	3	3	3	1
9	Nadya Alena Saputro	10	SD Muhammadiyah 5 Jakad	3	2	2	3	2
10	Aimeer Radhya Hamzah	5	Al-Azhar Syifa Budi Parahyangan	2	2	3	3	0
...
80	Nabeel Abhimanyu P	9	SDT Nurfatmahan	3	2	3	3	1

Gambar 3.2 Transformasi Data

Dari hasil tersebut, ada 5 fitur yang dipilih antara lain:

- Umur
- Durasi Penyelesaian
- Interaksi Dengan Instruktur
- Kreatifitas
- Konsentrasi

Normalisasi Data (Data Normalization)

Normalisasi data adalah teknik yang digunakan untuk mengubah nilai-nilai dalam data agar berada dalam rentang tertentu, biasanya antara 0 dan 1. Tujuannya adalah untuk menghindari dominasi fitur dengan nilai yang lebih besar dalam model machine learning, sehingga setiap fitur memiliki kontribusi yang seimbang dalam proses pembelajaran. Proses ini sangat penting, terutama untuk algoritma yang sensitif terhadap perbedaan skala data, seperti K-Nearest Neighbor (KNN). Dalam penelitian ini, normalisasi dilakukan dengan menggunakan metode MinMaxScaler dari library Scikit-learn, yang bekerja dengan rumus berikut:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Penjelasan :

- *X* adalah nilai asli dari fitur.
- *X_{min}* adalah nilai minimum dari fitur.



- X_{max} adalah nilai maksimum dari fitur.
- X' adalah nilai yang telah dinormalisasi.

Dalam penelitian ini, data yang digunakan adalah data pelatihan. Proses normalisasi dilakukan melalui langkah-langkah berikut:

1. Oversampling dengan SMOTE

Teknik SMOTE digunakan untuk melakukan oversampling pada kelas minoritas, sehingga dataset pelatihan menjadi lebih seimbang. Hasil dari proses ini adalah X_{train_smote} (data fitur) dan Y_{train_smote} (label kelas).

```
from imblearn.over_sampling import SMOTE

# Oversampling dengan SMOTE
print("Distribusi Kelas Sebelum Oversampling:")
print(y.value_counts()) # label asli

smote = SMOTE(sampling_strategy='auto', random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_scaled, y_encoded)

print("\nDistribusi Kelas Setelah Oversampling:")
print(pd.Series(y_resampled).value_counts())
```

Distribusi Kelas Sebelum Oversampling:
Grade
Junior 50
Little Creator 29
Teenager 9
Name: count, dtype: int64

Distribusi Kelas Setelah Oversampling:
0 50
1 50
2 50
Name: count, dtype: int64

Gambar 3.3 Oversampling Smote

2. Fitting MinMaxScaler

Objek MinMaxScaler dibuat dan di-fit pada data hasil oversampling (X_{train_smote}) untuk menghitung nilai minimum dan maksimum dari setiap fitur. Proses ini dapat dilihat pada kode yang ditampilkan di bawah ini.

```
from sklearn.preprocessing import MinMaxScaler

# Normalisasi Fitur
scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X)
```

Gambar 3.4 Fitting MinMaxScaler

3. Data Transformation

Proses transformasi dilakukan untuk mengubah nilai fitur agar berada dalam rentang [0, 1]. Transformasi ini diterapkan pada data pelatihan (X_{train_smote}) dan data pengujian (X_{test}) untuk memastikan keduanya memiliki skala yang seragam. Proses ini dapat dilihat pada kode yang ditampilkan di bawah ini:

```
X_scaled = scaler.transform(X)
```

Gambar 3.5 Data Transformation

Pembagian Data

Salah satu langkah penting dalam pengembangan model *machine learning* adalah membagi dataset menjadi dua bagian utama, yaitu data pelatihan (*training data*) dan data pengujian (*testing data*). Pembagian ini bertujuan untuk memastikan bahwa model yang dibangun memiliki kemampuan generalisasi yang baik ketika dihadapkan pada data baru.



1. Tujuan Pembagian Data

a. Data pelatihan

Data pelatihan digunakan untuk melatih algoritma *machine learning* agar dapat mempelajari pola-pola dari dataset. Dalam penelitian ini, data pelatihan digunakan untuk membangun model *K-Nearest Neighbor* (K-NN).

b. Data pengujian

Data pengujian digunakan untuk mengevaluasi kinerja model yang telah dilatih. Evaluasi ini bertujuan untuk mengukur seberapa baik model memprediksi hasil berdasarkan data yang belum pernah dilihat sebelumnya.

2. Proporsi Pembagian Data

Pada penelitian ini, dataset dibagi menjadi:

a. 80% untuk data pelatihan: digunakan untuk melatih model.

b. 20% untuk data pengujian: digunakan untuk menguji performa model.

c. `random_state=42`: digunakan untuk menetapkan nilai acak sehingga pembagian data tetap konsisten setiap kali kode dijalankan.

d.

```
from sklearn.model_selection import train_test_split

# Split Data
X_train, X_test, y_train, y_test = train_test_split(
    X_resampled, y_resampled, test_size=0.3, random_state=42, stratify=y_resampled)
```

Gambar 3.6 Pembagian Data

Pemodelan

Pada tahap ini, sistem klasifikasi mulai dikembangkan menggunakan algoritma *K-Nearest Neighbor* (K-NN). Proses ini mencakup pelatihan model dengan data pelatihan yang sebelumnya telah melewati tahap praproses. Algoritma K-NN dilatih untuk mengenali pola pada dataset dengan cara menghitung jarak (k) guna menentukan kelompok atau level pembelajaran anak secara tepat.

Setelah model berhasil dibangun, pengujian dilakukan menggunakan data pengujian untuk menilai performanya. Evaluasi ini mencakup penghitungan akurasi dan analisis hasil prediksi untuk memastikan bahwa model mampu memberikan rekomendasi level pembelajaran yang objektif dan akurat.

Berikut adalah langkah-langkah dalam tahap pemodelan Perhitungan Klasifikasi Sistem Pendukung Keputusan dengan Menggunakan Algoritma *K-Nearest Neighbor* (KNN)

1. Melatih Model *K-Nearest Neighbor* (K-NN)

Pada tahap ini, model K-NN dilatih menggunakan data pelatihan (*training data*) yang telah melewati proses praproses. Proses pelatihan ini mencakup beberapa langkah berikut:

a. Menentukan nilai K

Nilai K adalah jumlah tetangga terdekat yang digunakan oleh model untuk menentukan kategori atau kelas data. Pemilihan nilai K dilakukan melalui eksperimen, dengan mencoba berbagai nilai untuk menemukan hasil yang paling optimal.

Pemilihan nilai K harus mempertimbangkan keseimbangan antara:

- **Underfitting:** Terjadi jika nilai K terlalu kecil, sehingga model tidak cukup mempelajari pola data.
- **Overfitting:** Terjadi jika nilai K terlalu besar, sehingga model menjadi terlalu rumit dan tidak mampu menangkap pola umum.

Dalam penelitian ini, proses eksperimen untuk menentukan nilai K telah dilakukan, seperti yang dapat dilihat pada gambar berikut:

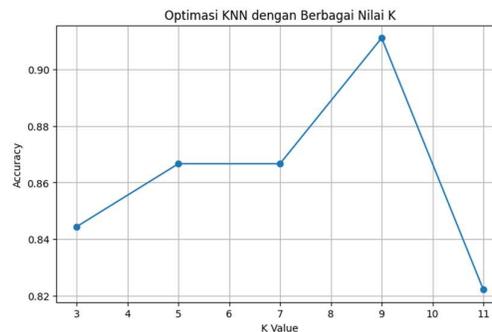


```
# Coba berbagai nilai K
k_values = [3, 5, 7, 9, 11]
results = {}

for k in k_values:
    knn = KNeighborsClassifier(n_neighbors=k)
    knn.fit(X_train, y_train)

    y_pred = knn.predict(X_test)
    acc = accuracy_score(y_test, y_pred)
```

Gambar 3.7 Model KNN



Gambar 3.8 Grafik Akurasi terhadap Nilai k

Evaluasi Akurasi Training dan Testing

Dalam penelitian ini, evaluasi dilakukan untuk mengukur kinerja model. Proses ini menggunakan dua set data utama, yaitu data pelatihan (training data) dan data pengujian (testing data). Data pelatihan digunakan untuk melatih model, sementara data pengujian digunakan untuk menilai kemampuan model dalam menggeneralisasi hasil terhadap data baru yang belum pernah dilihat sebelumnya.

Evaluasi ini sangat penting untuk memastikan bahwa model tidak hanya mampu menghafal data pelatihan (yang dapat menyebabkan overfitting), tetapi juga dapat memprediksi data baru dengan akurasi yang memadai. Hasil proses evaluasi dapat dilihat pada kode dan output yang ditampilkan berikut ini:

```
# Evaluasi Akurasi pada Data Training
y_train_pred = knn.predict(X_train)
y_test_pred = knn.predict(X_test)

train_accuracy = knn.score(X_train, y_train)
test_accuracy = knn.score(X_test, y_test)

print(f"Akurasi Training: {train_accuracy:.2f}")
print(f"Akurasi Testing: {test_accuracy:.2f}")
```

Akurasi Training: 0.84
Akurasi Testing: 0.85

Gambar 3.9 Evaluasi Model

C. Hasil Penelitian dan Pembahasan (Huruf 12 dan Ditebalkan) Implementasi

Implementasi adalah tahap di mana rancangan sistem sebelumnya diwujudkan menjadi sistem yang siap digunakan. Pada tahap ini, sistem yang telah dirancang akan dikembangkan dan diuji agar dapat berfungsi sesuai dengan kebutuhan. Implementasi mencakup penggunaan perangkat, tampilan sistem, serta berbagai operasi yang ada di dalamnya. Berikut ini adalah tabel yang menjelaskan struktur tampilan dan operasi pada sistem.



Implementasi Perangkat Keras

Perangkat keras komputer (hardware) adalah komponen fisik yang membentuk satu kesatuan dalam sebuah sistem komputer (PC). Komponen-komponen ini biasanya dirakit, dengan sebagian besar ditempatkan di dalam casing komputer, sementara beberapa lainnya berada di luar casing. Perangkat keras komputer mencakup sistem input, sistem output, penyimpanan (storage), dan unit pemrosesan (system unit).

Untuk mendukung pengembangan aplikasi ini, perangkat keras minimum yang digunakan adalah sebagai berikut:

1. Laptop : Asus X441UV Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz 1.99 GHz
2. RAM : 12 GB
3. SSD : 256 GB
4. Hardisk : 500 GB

Implementasi Perangkat Lunak

Perangkat lunak (*software*) adalah sekumpulan program yang berisi instruksi untuk mengoperasikan komputer atau perangkat keras yang terkait. Dalam bahasa Indonesia, *software* disebut perangkat lunak karena tidak memiliki bentuk fisik yang dapat dilihat, tetapi dapat dijalankan dan digunakan.

Dalam pengembangan aplikasi ini, perangkat lunak yang digunakan mencakup:

1. Windows 11
2. XAMPP
3. MySQL
4. Visual Studio Code
5. Draw.io
6. Google Collaboration
7. Microsoft Excel
8. Microsoft Word

Implementasi Antar Muka

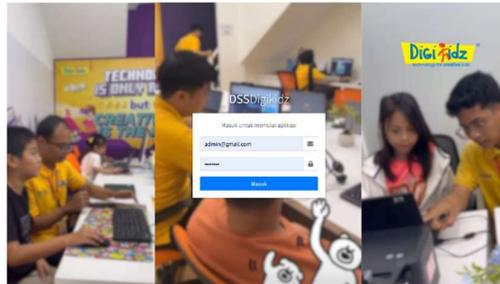
Implementasi program merupakan tahap lanjutan dari perancangan sistem, di mana sistem yang telah dirancang direalisasikan agar dapat digunakan secara fungsional. Dalam penelitian ini, implementasi difokuskan pada proses pengklasifikasian dalam sistem pendukung keputusan menggunakan metode K-Nearest Neighbor (KNN). Berikut adalah implementasi program dari sistem pendukung keputusan berbasis metode KNN.

Halaman Login

Halaman login merupakan halaman pertama yang muncul saat pengguna mengakses sistem **DSS Digikidz**. Pada halaman ini, pengguna harus memasukkan **email** dan **password** untuk dapat masuk ke dalam sistem.

Tampilan login ini dirancang untuk memberikan pengalaman yang **aman** dan **mudah digunakan**. Proses **validasi input** dilakukan secara otomatis untuk memastikan bahwa data yang dimasukkan sesuai dengan standar keamanan yang berlaku. Jika pengguna salah memasukkan email atau password, maka sistem akan menampilkan **pesan kesalahan** yang memberitahukan bahwa login gagal.

Selain itu, tampilan login menggunakan desain modern dengan **background gambar**, serta elemen UI yang responsif agar nyaman digunakan di berbagai perangkat. Berikut adalah tampilan halaman login pada sistem **DSS Digikidz**.



Gambar 4.1 Halaman Login

Halaman Dashboard adalah tampilan utama setelah pengguna login. Halaman ini memberikan ringkasan sistem serta panduan penggunaan fitur utama. Fitur Utama:

- Manajemen Data Murid : Mengelola informasi siswa untuk klasifikasi.
- Pengaturan Data Kriteria : Mengatur faktor seperti durasi penyelesaian, interaksi dengan instruktur, kreativitas, dan konsentrasi.
- Proses KNN : Menganalisis data dan menentukan level murid.
- History : Menampilkan hasil klasifikasi murid.

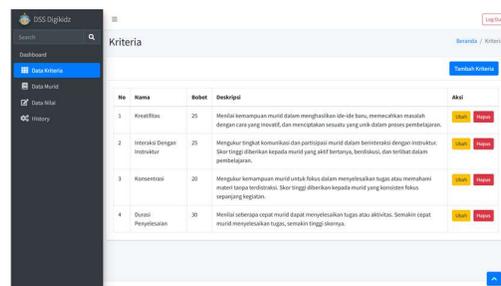
Sistem ini dirancang untuk membantu menentukan grade dan level murid secara akurat menggunakan metode K-Nearest Neighbor (KNN).



Gambar 4.2 Halaman Dashboard

Halaman Data Kriteria

Halaman Kriteria digunakan untuk mengelola faktor-faktor yang mempengaruhi klasifikasi level murid dalam sistem DSS Digikidz. Pengguna dapat menambahkan kriteria baru, mengedit kriteria yang sudah ada, atau menghapus kriteria yang tidak diperlukan. Data kriteria ditampilkan dalam bentuk tabel yang mencakup nama, bobot, dan deskripsi, serta dilengkapi dengan tombol aksi untuk mengubah atau menghapus data. Halaman ini memastikan bahwa pengelolaan kriteria dapat dilakukan dengan mudah dan efisien.

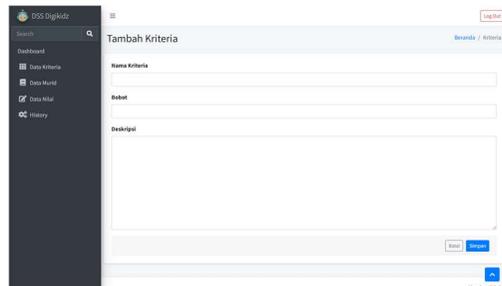


Gambar 4.3 Halaman Data Kriteria



Halaman Tambah Data Kriteria

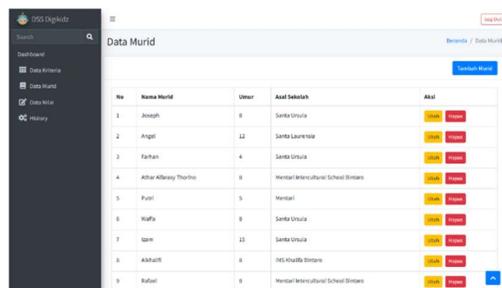
Halaman Tambah Kriteria digunakan untuk menambahkan faktor penilaian baru dalam sistem DSS Digikidz. Pengguna dapat mengisi nama kriteria, bobot, dan deskripsi dalam formulir yang tersedia. Sistem akan melakukan validasi input untuk memastikan data yang dimasukkan sesuai standar. Jika terjadi kesalahan, pesan peringatan akan ditampilkan. Setelah data diisi, pengguna dapat menyimpannya dengan menekan tombol Simpan atau membatalkan proses dengan tombol Batal.



Gambar 4.4 Halaman Tambah Data Kriteria

Halaman Data Murid

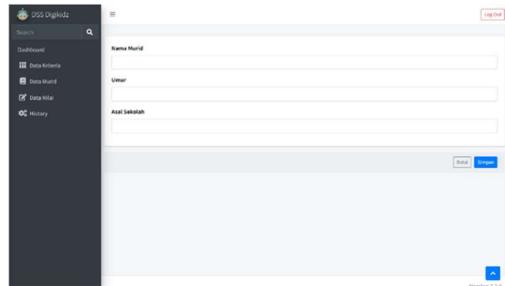
Halaman Data Murid digunakan untuk menampilkan informasi dasar mengenai murid yang terdaftar dalam sistem. Data yang ditampilkan mencakup **nama murid**, **umur**, **asal sekolah**, dan **grade**. Halaman ini juga berfungsi sebagai referensi utama sebelum proses penilaian dan prediksi dilakukan. Pengguna dapat menambahkan, mengedit, atau menghapus data murid sesuai kebutuhan. Data pada halaman ini menjadi acuan untuk proses lebih lanjut seperti input nilai dan prediksi level menggunakan metode K-Nearest Neighbor (KNN).



Gambar 4. 5 Halaman Data Murid

Halaman Input data murid

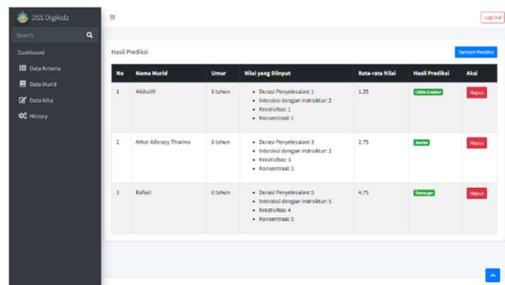
Halaman Halaman Input Data Murid digunakan untuk menambahkan data murid baru ke dalam sistem. Informasi yang diinput meliputi **nama murid**, **umur**, dan **asal sekolah**. Form ini bertujuan untuk memastikan bahwa data murid tersimpan secara lengkap dan valid sebelum dilakukan proses penilaian atau prediksi level. Setelah data disimpan, murid akan ditampilkan dalam daftar pada halaman Data Murid.



Gambar 4. 6 Halaman Input Data Murid

Halaman Data Nilai

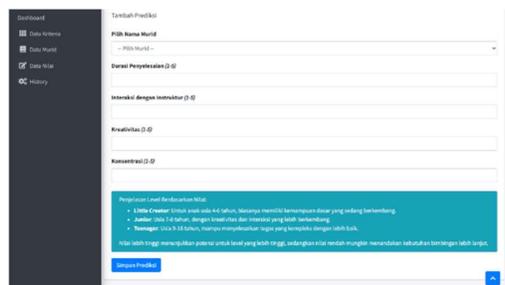
Halaman Data Nilai menampilkan hasil prediksi level murid berdasarkan metode K-Nearest Neighbor (KNN). Data yang ditampilkan mencakup nama murid, nilai yang diinput (durasi penyelesaian, interaksi dengan instruktur, kreativitas, dan konsentrasi), serta hasil prediksi level murid.



Gambar 4.7 Halaman Data Nilai

Halaman Tambah Prediksi

Halaman Tambah Prediksi memungkinkan pengguna untuk melihat detail hasil prediksi level murid berdasarkan metode K-Nearest Neighbor (KNN). Informasi yang ditampilkan meliputi nama murid, umur, hasil prediksi, dan level yang ditentukan.

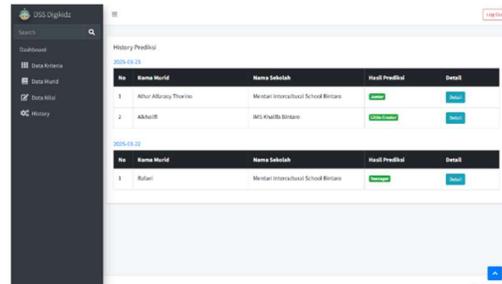


Gambar 4.8 Halaman Tambah Prediksi



Halaman History

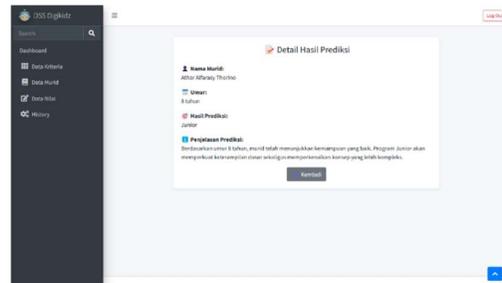
Halaman History Prediksi menampilkan daftar histori hasil prediksi untuk setiap murid yang telah diproses oleh sistem. Data dipisahkan berdasarkan tanggal prediksi untuk memudahkan pengguna dalam melihat hasil prediksi yang telah dilakukan pada tanggal tertentu.



Gambar 4.9 Halaman History

Halaman Detail Hasil Prediksi

Halaman Detail Hasil Prediksi menampilkan informasi lengkap mengenai hasil prediksi yang telah dilakukan terhadap seorang murid. Halaman ini bertujuan untuk memberikan gambaran mendalam tentang hasil prediksi dan penjelasan mengenai level yang didapatkan murid berdasarkan usia dan hasil prediksi.



Gambar 4.10 Halaman Detail Hasil Prediksi

Pengujian Sistem

Pengujian perangkat lunak ini bertujuan untuk memastikan bahwa aplikasi yang dibangun memenuhi standar kualitas yang telah ditetapkan serta untuk mengetahui apakah perangkat lunak dapat berfungsi sesuai dengan harapan. Pengujian dilakukan untuk memastikan bahwa aplikasi dapat menjalankan fungsi-fungsi yang ada dengan baik, sesuai dengan spesifikasi yang telah dianalisis dan dirancang sebelumnya.

Metode yang digunakan dalam pengujian ini adalah Black Box Testing, yang fokus pada pengujian fungsionalitas perangkat lunak tanpa melihat struktur internalnya. Dengan menggunakan metode ini, pengujian dilakukan dengan cara memberikan input tertentu dan memeriksa apakah output yang dihasilkan sesuai dengan yang diharapkan. Keuntungan dari metode ini adalah memudahkan untuk mengidentifikasi kesalahan pada antarmuka pengguna dan memastikan bahwa sistem berfungsi sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan.



Rencana Pengujian

a. Pengujian *Black Box Testing Login*

Tabel 4.1 Pengujian *Blackbox Testing Login*

No	Skenario Uji	Input	Ekspetasi Output	Hasil Pengujian
1	Login dengan kredensial valid	Email & password benar	Pengguna berhasil masuk ke dashboard	Valid
2	Login dengan email atau password salah	Email atau password salah	Muncul pesan error "Email atau password salah"	Valid
3	Login dengan email & password kosong	(kosong), (kosong)	Muncul pesan error " The email field is required. "	Valid
4	Login dengan format email salah	contohuser.com, password benar	Muncul pesan error " Email atau password tidak sesuai "	Valid
5	Logout dari sistem	Klik tombol logout	Pengguna keluar dan diarahkan ke halaman login	Valid

b. Pengujian *Black Box Testing Data Kriteria*

Tabel 4.2 Pengujian *Blackbox Testing Data Kriteria*

No	Skenario Uji	Input	Ekspektasi Output	Hasil Pengujian
1	Menambahkan kriteria baru	Nama: "Kreativitas", Bobot: 25	Kriteria berhasil ditambahkan	Valid
2	Mengedit data kriteria	Ubah bobot "Kreativitas" menjadi 25	Data kriteria berhasil diperbarui	Valid
3	Menghapus kriteria	Klik tombol hapus pada satu kriteria	Kriteria berhasil dihapus dari database	Valid
4	Menampilkan daftar kriteria	Akses menu "Data Kriteria"	Semua data kriteria tampil di tabel	Valid



c. Pengujian *Black Box Testing* Data Murid

Tabel 4.3 Pengujian *Blackbox Testing* Data Murid

No	Skenario Uji	Input	Ekspektasi Output	Hasil Pengujian
1	Menambahkan data murid	Nama: "Joseph", Umur: 7, Asal: "Sanur"	Data murid berhasil ditambahkan	Valid
2	Mengedit data murid	Ubah umur "Adit" menjadi 8	Data murid berhasil diperbarui	Valid
3	Menghapus data murid	Klik tombol hapus pada satu murid	Data murid berhasil dihapus	Valid
4	Menampilkan daftar murid	Akses menu "Data Murid"	Semua data murid tampil di tabel	Valid

d. Pengujian *Black Box Testing* Data Nilai

Tabel 4.4 Pengujian *Blackbox Testing* Data Nilai

No	Skenario Uji	Input	Ekspektasi Output	Hasil Pengujian
1	Menambahkan nilai murid	Pilih murid, isi nilai kriteria	Nilai murid berhasil ditambahkan	Valid
2	Mengedit nilai murid	Ubah nilai kreativitas menjadi 4	Data nilai berhasil diperbarui	Valid
3	Menghapus nilai murid	Klik tombol hapus pada satu data nilai	Data nilai berhasil dihapus	Valid
4	Menampilkan daftar nilai murid	Akses menu "Data Nilai"	Semua data nilai tampil di tabel	Valid

e. Pengujian *Black Box Testing* History

Tabel 4.5 Pengujian *Blackbox Testing* History

No	Skenario Uji	Input	Ekspektasi Output	Hasil Pengujian
1	Menampilkan hasil prediksi	Akses menu "Hasil Prediksi"	Semua hasil prediksi ditampilkan	Valid
2	Menghapus hasil prediksi	Klik tombol hapus pada satu prediksi	Data hasil prediksi berhasil dihapus	Valid



E. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai implementasi metode K-Nearest Neighbor (K-NN) dalam sistem pendukung keputusan untuk menentukan grade dan level murid di DIGIKIDZ BSD City, beberapa kesimpulan dapat diambil sebagai berikut:

- A. Sistem berhasil dibangun dan mampu mengklasifikasikan siswa ke dalam tiga kategori level pembelajaran, yaitu Little Creator, Junior, dan Teenager, berdasarkan data siswa yang mencakup umur, durasi penyelesaian, interaksi dengan instruktur, kreatifitas, dan konsentrasi.
- B. Metode K-Nearest Neighbor terbukti efektif digunakan dalam klasifikasi penempatan siswa. Dari hasil pengujian terhadap beberapa nilai k diperoleh bahwa nilai k = 9 memberikan akurasi tertinggi sebesar 91%, menunjukkan performa yang cukup baik dalam memprediksi level siswa.
- C. Sistem yang dibangun mampu membantu pengambilan keputusan secara objektif berdasarkan data, serta dapat mengurangi potensi subjektivitas dalam proses penentuan grade siswa. Hal ini mempermudah pihak pengelola DIGIKIDZ dalam menempatkan siswa pada level pembelajaran yang sesuai dengan kemampuan mereka.

Dari hasil yang diperoleh, sistem ini telah mencapai tujuan penelitian, yaitu membangun Sistem Pendukung Keputusan (SPK) berbasis K-NN yang mampu mengklasifikasikan murid ke dalam level yang sesuai dengan akurat dan efisien.

DAFTAR PUSTAKA

- Andani, M., Asia, M., Jendral Yani No, J. A., KomerlingUlu, O., & Selatan, S. (2021). Sistem Informasi Pelayanan Kependudukan Desa Lecah Berbasis Web Menggunakan Php Dan Mysql. *Jurnal Sistem Informasi Mahakarya (JSIM)*, 4(1), 15-27.
- Dinata, R. K., Akbar, H., & Hasdyna, N. (2020). Algoritma K-Nearest Neighbor dengan Euclidean Distance dan Manhattan Distance untuk Klasifikasi Transportasi Bus. *ILKOM Jurnal Ilmiah*, 12(2), 104-111. <https://doi.org/10.33096/ilkom.v12i2.539.104-111>
- Febrian, N., & Noviandi, N. (2024). Perbandingan Manhattan dan Euclidean Distance Untuk Pengelompokan Penyakit Jantung Menggunakan Algoritma K-Means. *ICIT Journal*, 10(1), 61-70. <https://doi.org/10.33050/icit.v10i1.2860>
- Givandi, M., & Affandi, M. (2023). Perancangan Sistem Pendukung Keputusan Penerimaan Anak Asuh Pada Panti Asuhan Hanifa III Dengan Metode Vikor. *Jurnal Sains Informatika Terapan*, 2(3), 114-118. <https://doi.org/10.62357/jsit.v2i3.203>
- Haris Andri, R., & Permana Sitanggang, D. (2022). Sistem Penunjang Keputusan (SPK) Pemilihan Supplier Terbaik Dengan Metode MOORA. *Jurnal Sains Informatika Terapan*, 2(3), 79-84. <https://doi.org/10.62357/jsit.v2i3.181>
- Hendri, R., Hartanto, M. B., & Agustin, A. (2023). Rancang Bangun Sistem Pendukung Keputusan Validasi Data Pegawai Polda Dengan Metode AHP Berbasis WEB. *Jurnal Teknologi Dan Informatika (JEDA)*, 4(1), 1-9. <https://doi.org/10.57084/jeda.v4i1.1189>
- Jailani, A., & Ainul Yaqin, M. (2024). Pengujian Aplikasi Sistem Informasi Akademik menggunakan Metode Blackbox dengan Teknik Boundary Value Analysis. *JACIS : Journal Automation Computer Information System*, 60-66. <https://doi.org/10.47134/jacis.v4i2.78>
- Mira Orisa, Ahmad Faisol, & Mochammad Ibrahim Ashari. (2023). Perancangan Website Company Profile Menggunakan Design Science Research Methodology (Dsrn). *Jurnal Informatika Teknologi Dan Sains*, 5(1), 160-164. <https://doi.org/10.51401/jinteks.v5i1.2576>
- Muhaimin, A., Amin Hariyadi, M., & Imamudin, M. I. (2024). Klasifikasi Prestasi Akademik Siswa Berdasarkan Nilai Rapor dan Kedisiplinan dengan Metode K-Nearest Neighbor. *Jurnal Ilmu Komputer Dan Sistem Informasi (JIKOMSI)*, 7(1), 193-202. <https://doi.org/10.55338/jikoms.v7i1.2865>



- Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 5(2), 697-711.
- Pangestu, P., & Setyadi, R. (2023). Penerapan Metode K-Nearest Neighbor Untuk Pemilihan Rekomendasi Game FPS Pada Aplikasi Google Play Store. *Journal of Information System Research (JOSH)*, 4(2), 742-747. <https://doi.org/10.47065/josh.v4i2.3006>
- Prasetyo, S. M., Pambudi, S., & Arkansyah, I. (2023). Perancangan Backend Database Dengan Mysql Pada Sistem Management Asset Management Asset. *OKTAL : Jurnal Ilmu Komputer Dan Science*, 2(5), 1425-1431. <https://journal.mediapublikasi.id/index.php/oktal>
- Purwandi, R. (2023). Penerapan Algoritma k-Nearest Neighbour. 1.
- Rahmatika, A. K., Pradana, F., & Bachtiar, F. A. (2020). Pengembangan Sistem Pembelajaran HTML dan CSS dengan Konsep Gamification berbasis Web. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 4(8 Agustus), 2655-2663.
- Rahmawati, R., & Hermaliani, E. H. (2024). Perancangan Sistem Informasi Perjalanan Dinas (SILADIN) Berbasis Web Pada PPSDMAP. *Swabumi*, 12(1), 27-38. <https://doi.org/10.31294/swabumi.v12i1.16960>
- Surahmat, A., Korneilis, K., & Fu'ady, T. D. (2023). Perancangan Sistem Pendukung Keputusan Pengangkatan Karyawan Kontrak Menjadi Karyawan Organik Dengan Metode Saw Pada Pt. Krakatau Jasa Logistik. *Journal of Innovation And Future Technology (IFTECH)*, 5(2), 43-55. <https://doi.org/10.47080/iftech.v5i2.2702>
- Tejasukmana Putra, R., Adi Wibowo, S., & Agus Pranoto, Y. (2021). Sistem Pendukung Keputusan Penerimaan Blt Di Kecamatan Sampang Menggunakan Metode Saw Dan Metode Ahp Berbasis Web. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 5(1), 321-327. <https://doi.org/10.36040/jati.v5i1.3236>
- Widiastuti, N. A., Azhar, M., & Mulyo, H. (2023). Implementasi Algoritma K-Nearest Neighbor untuk Klasifikasi Jurusan pada Peserta Didik Baru. *Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, 14(2), 195-208. <https://doi.org/10.24176/simet.v14i2>. *Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, 14(2), 195-208. <https://doi.org/10.24176/simet.v14i2.10092>
- Yanasari, C., & Arifin, T. (2023). Implementasi Algoritma K-Nearest Neighbor Untuk Klasifikasi Penerimaan Beasiswa Program Indonesia Pintar. *Jurnal Sistem Informasi Dan Ilmu Komputer*, 1(4), 178-194. <https://doi.org/10.59581/jusiik-widyakarya.v1i4.1862>