OPTIMISASI SISTEM DETEKSI PHISHING BERBASIS WEB MENGGUNAKAN ALGORITMA DECISION TREE

Muhammad Reza Fatiha[1], Ito Setiawan[2*], Ali Nur Ikhsan[3], Ika Romadoni Yunita[4]
[1,2,3,4]Universitas Amikom Purwokerto
Jl. Letjend Pol. Soemarto No.127, Watumas, Purwanegara, Kec. Purwokerto Utara, Kabupaten Banyumas, Jawa Tengah 53127
Email : [1]rezafatiha38@gmail.com, [2]itosetiawan@amikompurwokerto.ac.id, [3]alinurikhsan@amikompurwokerto.ac.id, [4]ikarom@amikompurwokerto.ac.id

Abstract

The development of information technology facilitates various aspects of life, but also presents challenges, such as the threat of phishing. Phishing is a cyberattack that tricks users into providing sensitive information, often through fake emails or legitimate-looking websites. The impact is wide, covering the banking, e-commerce, and government services sectors. These attacks continue to evolve with increasingly sophisticated techniques, such as the use of lookalike domains and social engineering, so accurate detection is an urgent need. This research develops a web-based phishing detection system using the Decision Tree algorithm and the Rapid Application Development (RAD) method. The public dataset contains 11,055 data with 26 attributes used. Data is processed through deduplication, handling of missing values, and normalization. After being divided into 80% training data and 20% test data, the Decision Tree model was built and tested using k-fold cross-validation. The model achieved 95.07% accuracy in detecting phishing. The RAD method supports the development of fast and efficient systems. These results are expected to help individuals and companies protect data from phishing attacks. Future research may explore other algorithms such as Random Forest and integrate real-time monitoring features to improve protection.

Keywords: *Decision Trees, Detection, Phishing, RAD, Websites*

Abstraksi

Perkembangan teknologi informasi memudahkan berbagai aspek kehidupan, namun juga menghadirkan tantangan, seperti ancaman phishing. Phishing adalah serangan siber yang mengecoh pengguna untuk memberikan informasi sensitif, sering melalui email palsu atau situs web yang tampak resmi. Dampaknya luas, mencakup sektor perbankan, e-commerce, dan layanan pemerintah. Serangan ini terus berkembang dengan teknik yang semakin canggih, seperti penggunaan domain mirip dan social engineering, sehingga deteksi yang akurat menjadi kebutuhan mendesak. Penelitian ini mengembangkan sistem deteksi phishing berbasis web menggunakan algoritma Decision Tree dan metode Rapid Application Development (RAD). Dataset publik berisi 11.055 data dengan 26 atribut digunakan. Data diproses melalui penghapusan duplikasi, penanganan nilai hilang, dan normalisasi. Setelah dibagi menjadi 80% data pelatihan dan 20% data pengujian, model Decision Tree dibangun dan diuji menggunakan k-fold cross-validation. Model mencapai

akurasi 95,07% dalam mendeteksi phishing. Metode RAD mendukung pengembangan sistem yang cepat dan efisien. Hasil ini diharapkan membantu individu dan perusahaan melindungi data dari serangan phishing. Penelitian mendatang dapat mengeksplorasi algoritma lain seperti Random Forest dan mengintegrasikan fitur real-time monitoring untuk meningkatkan perlindungan.

Kata Kunci: *Decision Tree, Deteksi, Phishing, RAD, Website*

1. PENDAHULUAN

Teknologi informasi dan komunikasi telah mengalami perkembangan yang sangat pesat dalam beberapa tahun terakhir, memudahkan pengguna untuk menyelesaikan berbagai pekerjaan secara efisien [1] . Kemajuan teknologi ini juga membawa tantangan baru, terutama dalam bentuk kejahatan siber yang memanfaatkan kelemahan sistem dan kurangnya kesadaran pengguna [2], [3]. Salah satu bentuk kejahatan siber yang paling meresahkan adalah phishing, yang bertujuan untuk menipu korban agarmemberikan informasi pribadi mereka secara tidak sadar [4]. *Phishing* sering kali dilakukan melalui pengiriman *email* palsu atau pembuatan *website* palsu yang terlihat seperti situs resmi, dengan tujuan memperoleh informasi sensitif dari pengguna [5] . P*hishing* dapat juga berupa serangan di mana pelaku berpura-pura menjadi pihak yang dipercaya untuk memperoleh data penting [6]. Taktik *phishing* umumnya melibatkan pembuatan *link* yang mengarah ke situs web palsu yang dirancang untuk menipu korban [7].

Risiko yang ditimbulkan oleh phishing sangat signifikan, termasuk kerugian privasi, eksploitasi data, dan kerugian finansial [6]. Ancaman ini tidak hanya terbatas di Indonesia, tetapi juga merupakan masalah global yang mempengaruhi berbagai aspek kegiatan *online*. Data dari Indonesia *Anti-Phishing* Data *Exchange* (IDADX) menunjukkan peningkatan laporan kasus phishing secara signifikan, dengan 6.106 laporan pada kuartal 4 tahun 2022 dan meningkat menjadi 26.675 laporan. pada kuartal 1 tahun 2023 [8]. Selain itu, biaya rata-rata akibat pelanggaran data secara global pada tahun 2023 diperkirakan mencapai USD 4,45 juta, dan laporan dari FBI menunjukkan kerugian lebih dari *USD* 10 miliar akibat serangan siber pada tahun 2021.

Deteksi dini merupakan kunci penting. Penggunaan teknik klasifikasi untuk mendeteksi website dan link phishing menjadi solusi yang efektif. Penelitian ini berfokus pada pengembangan website deteksi phishing dengan menggunakan algoritma *decision*

tree. Algoritma ini dipilih karena telah terbukti efektif dalam mengidentifikasi *phishing* dengan hasil yang stabil dibandingkan dengan algoritma lain seperti *logistic regression* . Untuk mendukung pengembangan sistem ini, dataset phishing yang digunakan adalah dataset dari [Kaggle](https://www.kaggle.com), yang berisi 11.055 data dan 26 atribut. Dataset ini akan menjadi basis untuk pelatihan dan pengujian algoritma *decision tree* dalam mendeteksi serangan *phishing*.

Metode *Rapid Application Development* (*RAD*) akan diterapkan dalam pengembangan sistem ini. *RAD* merupakan pendekatan pengembangan perangkat lunak yang menekankan pada siklus pengembangan yang cepat dan efisien, memungkinkan pembangunan sistem dalam waktu yang lebih singkat dan dengan biaya yang lebih rendah [9]. Penelitian menunjukkan bahwa aplikasi deteksi *phishing* berbasis website yang menggunakan algoritma decision tree dapat mencapai hasil yang lebih baik dalam mendeteksi *URL phishing* [9]. Penelitian sebelumnya menunjukkan bahwa algoritma Decision Tree efektif dalam mendeteksi phishing [10]. Deteksi phising baik dalam analisis URL, email berbasis cloud, maupun berbagai jenis data phishing lainnya [11], [12]. Penekanan pada pentingnya dataset yang berkualitas dan pendekatan pengembangan yang adaptif [13],[14]. Penelitian saat ini berfokus pada optimisasi sistem deteksi phishing berbasis web menggunakan algoritma Decision Tree, bertujuan untuk meningkatkan akurasi dan efisiensi deteksi melalui penggunaan dataset yang lebih representatif dan teknik pengembangan yang responsif. Penelitian ini bertujuan untuk mengembangkan sistem deteksi phishing berbasis web yang efektif dengan pendekatan algoritma Decision Tree dan metode Rapid Application Development (RAD), serta memberikan kontribusi sebagai referensi bagi individu dan perusahaan dalam melindungi data dari serangan phishing. Selain itu, penelitian ini diharapkan dapat menjadi acuan untuk pendekatan alternatif dalam pengembangan sistem deteksi phishing.

2. METODE PENELITIAN

Proses pengembangan aplikasi deteksi *phishing* berbasis *website* yang dikerjakan pada penelitian ini secara garis beras menggunakan 2 metode yaitu metode klasifikasi data *link* dan *web phishing*menggunakan *algoritma Decision Tree*, dan metode *Rapid Application Development* (RAD) sebagaimetode pengembanyan sistem yaitu sistem deteksi phishing

berbasis *website*. Pada proses pengembangan sistem deteksi *phishing* dengan algoritma *decision tree* menggunakan metode *rapid application development* (*RAD*) dalam penelitian ini,dilakukan dalam lima tahapan utama yaitu identifikasi masalah, pengumpulan data, klasifikasi deteksi *phishing*, pengembangan sistem dan pengujian yang dapat dilihat pada Gambar 1.

Gambar 1. Tahapan Penelitian

a. Pengumpulan Data

Dalam penelitian ini, pengumpulan data difokuskan pada memperoleh dataset yang relevan untuk mengidentifikasi pola phishing pada aplikasi berbasis web. Dataset yang digunakan, bernama **phishing_dataset.csv**, diambil dari *Kaggle* dan mencakup 11.055 data dengan 26 atribut, yang meliputi *URL*, konten halaman web, metadata, dan informasi sertifikat keamanan. Dataset ini mencakup data *phishing* serta aplikasi web yang *legitimate* untuk membangun basis perbandingan yang kuat dalam klasifikasi. Proses selanjutnya adalah pra- pengolahan data, termasuk pembersihan data untuk menghilangkan duplikasi dan menangani data yang hilang, sebelum digunakan untuk melatih dan menguji model deteksi *phishing* berbasis algoritma *Decision Tree*.

b. Preprocessing Data

Proses preprocessing dilakukan untuk memastikan kualitas data yang optimal, meliputi: pembersihan data dari nilai yang hilang atau tidak valid dan normalisasi atribut agar data lebih homogen. Pembagian data menjadi 80% untuk pelatihan dan 20% untuk pengujian.

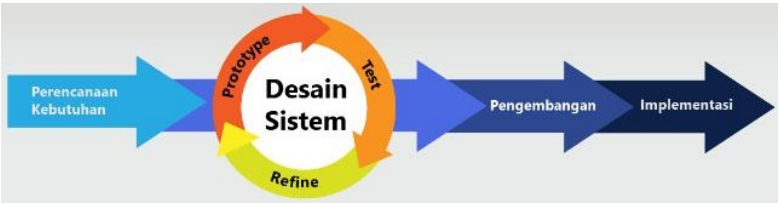
c. Klasifikasi Deteksi *Phishing* (Algoritma *Decision Tree*)

Dalam penelitian ini, algoritma *Decision Tree* digunakan untuk mengembangkan sistem deteksi *phishing* berbasis web dengan mengoptimalkan proses klasifikasi. Tahap pertama mencakup pengumpulan data dari berbagai sumber, termasuk *log server*, database insiden keamanan siber, dan dataset *phishing* publik yang sudah diklasifikasikan, serta data dari aplikasi web yang sah untuk perbandingan. Data ini meliputi *URL,* konten halaman web,

metadata, dan informasi sertifikat keamanan. Setelah data terkumpul, dilakukan pra-pengolahan data untuk memastikan kualitas, termasuk pembersihan data, penanganan data yang hilang, dan pemilihan fitur penting. Dataset yang telah dipersiapkan kemudian digunakan untuk melatih model *Decision Tree,* yang membagi data berdasarkan fitur yang dipilih untuk mengidentifikasi aplikasi *phishing* secara otomatis.

d. Pengembangan Sistem *(Rapid Application Development)*

Model pengembangan sistem, *Rapid Aplication Development (RAD)* yang merupakan salah satu model dari *System Development Life Cycle (SDLC). Rapid Aplication Development* (RAD) merupakan model proses pengembangan perangkat lunak secara linear sequential yang menekankan pada siklus pengembangan yang sangat singkat, dengan empat tahapan utama yaitu perencanaan kebutuhan, desain sistem, pengembangan sistem dan implementasi yang dapat dilihat pada Gambar 2 [15].



Gambar 2. Metode *Rapid Application Development (RAD)*

e. Pengujian

Untuk pengujian pada penelitian ini menggunakan *confusion matrix*, dimana akan dihitung nilai *precision, recall* dan *accuracy* [16]. *Confusion matrix* terdiri dari *true positive, false positive, true negative* dan *false negative* untuk menghitung presisi, *recall* dan akurasi. *Precision* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. Sedangkan *recall* adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Dalam penelitian ini, pengumpulan data dilakukan dengan menggunakan dataset *phishing_dataset.csv* yang diambil dari Kaggle. Dataset ini berisi 11.055 data dengan 26 atribut, seperti URL, konten halaman web, metadata, dan informasi sertifikat

keamanan. Dataset mencakup data phishing serta aplikasi web yang legitimate, sehingga memungkinkan peneliti untuk membangun model klasifikasi yang dapat membedakan antara situs web yang berpotensi berbahaya dan yang sah.

3.2 Preprocessing Data

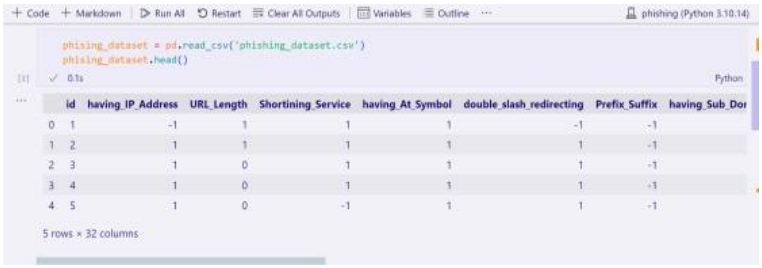
Proses preprocessing merupakan langkah krusial dalam mempersiapkan data sebelum analisis lebih lanjut, yang meliputi pembersihan data untuk menghilangkan duplikasi dan mengoreksi format, penanganan data hilang dengan mengganti nilai yang hilang menggunakan rata-rata atau modus, pengkodean kategori untuk mengubah variabel kategorikal menjadi format numerik, serta normalisasi/standarisasi untuk mengubah skala data agar berada dalam rentang yang sama guna meningkatkan performa model.

3.3 Implementasi Algoritma

Phishing Detector menggunakan algoritma klasifikasi *decision tree.* Persoalan phishing detector dalam penelitian ini yaitu sebuah kualitiatif yang ditranslasi ke dalam numerik (numeric encoding). *Dataset* yang digunakan untuk melakukan *phishing detector* adalah dataset yang memuat fitur-fitur *website phishing. Dataset* tersebut merupakan fitur-fitur atau kondisi yang menggambarkan bahwa fitur tersebut merupakan *website phishing* atau bukan. Data pada *dataset* akan diolah menjadi -1 dan 1 atau *true* dan *false.* Dimana, *decision tree* menggunakan *binary tress* dalam menganalisa dan memproses data. Sehingga dihasilkan *model tree* atau pemodelan *decision tree* yang efektif dan memiliki performa yang baik untuk menganalisa apakah sebuah *website* itu *phishing* atau tidak dengan menggunakan machine *learning. Import* model dan matriks dapat dilihat pada gambar 3.

Gambar 3. *Import* model dan matriks

Dataset yang digunakan terdiri atas 32 kolom variabel fitur website phishing untuk melakukan prediksi bahwa suatu web itu phishing atau bukan dapat dilihat pada gambar 4.



```
phising_dataset = pd.read_csv('phishing_dataset.csv')
phising_dataset.head()
```

	id	having_IP_Address	URL_Length	Shortining_Service	having_At_Symbol	double_slash_redirecting	Prefix_Suffix	having_Sub_Dor
0	1	-1	1	1	1	-1	-1	
1	2	1	1	1	1	1	-1	
2	3	1	0	1	1	1	-1	
3	4	1	0	1	1	1	-1	
4	5	1	0	-1	1	1	-1	

5 rows × 32 columns

Gambar 4. Dataset_phishing

Dataset_phishing dibagi menjadi dua bagian yaitu data sample dan data label. Sampel = phising_dataset.iloc[:,:-1] artinya mengambil semua baris (":") dari dataset "phising_dataset", dan semua kolom kecuali kolom terakhir (":-1"). Dengan kata lain, ini mengambil semua fitur atau atribut dari dataset. Sedangkan, label = phising_dataset ['Result'] artinya mengambil kolom yang bernama 'Result' dari dataset_phising dan menyimpannya dalam variabel "label". Kolom 'Result' ini kemungkinan besar berisi label atau kelas yang menunjukkan apakah suatu sampel dianggap sebagai phishing atau tidak dapat dilihat pada gambar 5.

```
sampel = phising_dataset.iloc[:,:-1]
label = phising_dataset['Result']
```

```
print(label)
```

```
0       -1
1       -1
2       -1
3       -1
4        1
       ..
11050    1
11051   -1
11052   -1
11053   -1
11054   -1
Name: Result, Length: 11055, dtype: int64
```

Gambar 5. Sample dan Label

Dataset_phishing akan dibagi menjadi dua bagian yaitu satu untuk pelatihan model (data_train, label_train) dan yang lainnya untuk pengujian model (data_test, label_test), dengan proporsi pengujian sebesar 20% dari keseluruhan dataset dapat dilihat pada gambar 6. .

```
from sklearn.model_selection import train_test_split
data_train, data_test, label_train, label_test = train_test_split(sampel, label, test_size=0.2, random_state=0)
```

Gambar 6. Split Dataset_phishing

Gambar 7. Tree classifier

Berdasarkan gambar 7. Dapat diketahui bahwa model machine learning ini dilakukan untuk membuat sebuah model *Decision Tree Classifier* dan melatihnya menggunakan dataset pelatihan yang sudah dibagi sebelumnya

Gambar 8. Accuracy Score

Berdasarkan gambar 8. Didapatkan informasi bahwa model *machine learning* untuk *phishing detector* akan membuat prediksi menggunakanmodel *decision tree* yang telah dilatih, menghitung akurasi prediksi tersebut, danmencetak akurasi ke layar.

3.4 Pengembangan Sistem

Pengembangan sistem *phishing detector*merupakan implementasi dari *machine learning phishing detector*, yang diimplementasi kedalam sistem berbasis *website.* Dibawah ini merupakan penjelasan dari masing-masing fitur pada sistem *phishing detector.* Tampilan awal dari sistem phishing *detector* berbasis website tersaji pada gambar 9.

Gambar 9. *Home page*

Kemudian klik *"get started"* untuk mulai menggunakan fitur pada sistem *phishing detector*. Berikut halaman phising detector tersaji pada gambar 10.

Gambar 10. *Phising Detector*

Dapat diketahui bahwa input halaman ini memuat *form* untuk menginputkan *field* link yang kemudian akan diproses dengan diprediksi menggunakan algoritma *decission tree* tersaji pada gambar 11.

Gambar 11. Proses

Akhir proses, algoritma *Decision Tree* memberikan klasifikasi "*phishing*" atau "*not phishing*" berdasarkan analisis tautan. menunjukkan *output* sistem yangmencerminkan hasil klasifikasi tautan yang diinputkan: jika tautan sesuai dengan karakteristik *phishing*

dalam *Decision Tree*, hasilnya akan menunjukkan "*phishing*"; jika tidak, hasilnya akan menunjukkan "*not phishing*". Proses ini memberikan kesimpulan yang jelas dan terstruktur, membantu pengguna mengidentifikasi dan menghindari tautan berbahaya. Tampilan menu yang memungkinkan pengguna memasukkan tautan ke dalam *form* untuk dideteksi apakah tautan tersebut merupakan phishing atau tidak. Berikut halaman outputnya tersaji pada gambar 12.

Gambar 12. *Output*

3.5 Evaluasi dan Validasi

Hasil pengujian dari penelitian menggunakan model Decision Tree untuk klasifikasi sebagai detektor phishing dengan dataset **dataset_phising.csv** yang terdiri dari 11.055 data, dibagi menjadi dua bagian—80% untuk data pelatihan (8.844 data) dan 20% untuk data pengujian (2.211 data)—menunjukkan skor akurasi prediksi sebesar **95,07%**. Hasil ini menggambarkan kemampuan model dalam mengklasifikasikan data phishing secara efektif. Detail hasil pengujian terkait data tersebut dapat dilihat pada **Tabel 1**..

Tabel 1. Hasil Pengujian Akurasi

Deskripsi	Nilai
Jumlah Total Data	11.055
PersentaseData Pelatihan	80% (8844 data)
Persentase Data Pengujian	20% (2211 data)
Akurasi = $\frac{Jumlah\ Prediksi\ Benar}{Total\ Presentase\ Data\ Pengujian}$ x 100%	
Akurasi = $\frac{2101}{2211}$ x 100%	
Akurasi = 0,9507 x 100%	
Skor Akurasi Prediksi	95,07%

4. KESIMPULAN

Penggunaan algoritma Decision Tree dalam sistem deteksi phishing berbasis web menunjukkan akurasi prediksi sebesar 95,07%, membuktikan efektivitasnya dalam mengidentifikasi situs phishing. Dengan metode Rapid Application Development (RAD), sistem dikembangkan secara efisien dan adaptif terhadap ancaman phishing. Untuk meningkatkan keandalan, penelitian selanjutnya disarankan mengeksplorasi fitur lebih mendalam, teknik pemrosesan data lanjutan, serta metrik evaluasi tambahan seperti presisi, recall, dan F1-score, serta memperluas pengujian dengan dataset yang lebih besar.

DAFTAR PUSTAKA

[1] M. Rizki, "Perkembangan Sistem Pertahanan/Keamanan Siber Indonesia dalam Menghadapi Tantangan Perkembangan Teknologi dan Informasi," *J. Ilmu Polit.*, vol. 14, no. 1, pp. 54–62, 2022.

[2] F. Surahman, "11 Tantangan dalam Menjaga Keamanan Data," *J. Ilm. Multidisipline*, vol. 1, no. 11, 2023.

[3] I. A. Saputro, L. Sugiarto, and F. S. Nugraha, "Analisis Kesadaran Masyarakat Terhadap Bahaya Internet Phishing Menggunakan K-Means Clustering," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 9, no. 2, 2024.

[4] N. B. Putri and A. W. Wijayanto, "Analisis Komparasi Algoritma Klasifikasi Data Mining Dalam Klasifikasi Website Phishing," *Komputika J. Sist. Komput.*, vol. 11, no. 1, pp. 59–66, 2022.

[5] Amin Muftiadi Tri Putri Mulyadi Agustina and M. Evi, "Studi Kasus Keamanan Jaringan Komputer: Analisis Ancaman Phishing terhadap Layanan Online Banking," *J. Ilm. Tek.*, vol. 1, no. 2, pp. 60–65, 2022.

[6] H. Ahmadian and A. Sabri, *Teknik Penyerangan Phishing Pada Social Engineering Menggunakan Set dan Pencegahannya*, vol. 2. 2021.

[7] N. Vadila and A. R. Pratama, "Analisis Kesadaran Keamanan Terhadap Ancaman Phishing," 2021.

[8] I. A.-P. D. E. (IDADX), "Laporan Aktivitas Phishing Domain ~.ID." 2023.

[9] A. Kulkarni and L. L. Brown, "Phishing Websites Detection using Machine Learning," vol. 10, 2019.

[10] M. Ali, J. Khan, and S. Hussain, "A Comparative Study of URL Classification Algorithms for Phishing Detection," *J. Cyber Secur. Mobil.*, vol. 10, no. 2, pp. 137–156, 2021, doi: 10.13052/jcsm2245-1439.1021.

[11] A. Sharif, K. A. Butt, and S. H. Asghar, "Phishing Detection in Cloud-Based Email Systems Using Machine Learning Techniques," *IEEE Access*, vol. 8, pp. 150192–150208, 2020, doi: 10.1109/ACCESS.2020.3016689.

[12] S. Patil and R. Kumar, "An Efficient Implementation of Rapid Application Development in Agile Projects," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 11, no. 1, pp. 45–52, 2022, doi: 10.30534/ijatcse/2022/01112022.

[13] Y. Zhang, J. Hong, and L. Cranor, "Dataset Challenges in Phishing Detection and Recommendations for Improvement," *J. Cyber Secur.*, 2019, doi: 10.1093/cyber/cyz023.

[14] Z. Tan and D. Wang, "Phishing Detection with Machine Learning: A Survey and Comparison," *Comput. Secur.*, vol. 103, pp. 102–120, 2021, doi: 10.1016/j.cose.2021.102120.

[15] A. Suhaimah, A. Triayudi, and E. T. Esthi Handayani, "Cyber Library: Pengembangan Perpustakaan Online Berbasis Web Menggunakan Metode Prototyping (Studi Kasus Universitas Nasional)," *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 5, no. 1, pp. 41–48, 2021, doi: 10.35870/jtik.v5i1.199.

[16] A. Widyanto, K. Kusrini, and K. Kusnawi, "Pengaruh Keseimbangan Data terhadap Akurasi Model Support Vector Machine pada Data Set Donor Darah," *J. Teknol. Terpadu*, vol. 9, no. 2, pp. 79–88, 2023, doi: 10.54914/jtt.v9i2.771.