



MIFORTEKH (Jurnal Manajemen Informatika & Teknologi)
p-ISSN : 2808-7550 (print) e-ISSN : 2798-0235 (online)
Vol. 5, No. 1, Mei 2025
<https://journal.stiestekom.ac.id/index.php/mifortekh>

Optimasi Pemilihan Soal Ujian Try Out Menggunakan Algoritma Greedy Berdasarkan Tingkat Kesulitan dan Waktu Pengerjaan

Zelly Apriza¹, Septi Mujiranda², Tata Sutabri³

Jurusan Teknik Informatika, Universitas Bina Darma, Palembang

Email Author: [1zellyapriza1@gmail.com](mailto:zellyapriza1@gmail.com), [2sepdajr09@gmail.com](mailto:sepdajr09@gmail.com), [3tatasutabri@gmail.com](mailto:tatasutabri@gmail.com)

Article Info

Article history:

Received Maret 28, 2025

Revised April 13, 2025

Accepted April 28, 2025

Keywords:

Greedy Algorithm

Question Selection

Try Out

Difficulty Level

Completion Time

ABSTRACT

The selection of appropriate questions for the trial exam is one of the main factors in objectively evaluating the readiness of participants. This process not only considers the level of difficulty of the questions, but also the duration of the work required. This study proposes the application of the greedy algorithm as a method for selecting questions based on these two main parameters. The greedy algorithm was chosen because it can provide a quick solution by selecting the best option at each step of decision making. In practice, the system will sort the questions based on the ratio of the value between the level of difficulty and the time required to complete them, then select questions until the maximum time is reached. The test results show that this method can form a collection of questions that are balanced between the time to complete and the level of difficulty. This method can also be applied in various trial exam situations with a set time. This study offers an initial overview of the use of the greedy algorithm in developing an efficient and responsive question selection system.

Corresponding Author:

Zelly Apriza,

Universitas Bina Darma

Jl. Jenderal Ahmad Yani No.12, Sei Kambing C II, 20 Ilir D. III, Kec. Ilir Timur I, Kota Palembang, Sumatera Selatan 30113, Indonesia

Email: zellyapriza1@gmail.com



ABSTRAK

Pemilihan soal untuk ujian percobaan yang tepat merupakan salah satu faktor utama dalam mengevaluasi kesiapan peserta secara objektif. Proses ini tidak hanya memperhatikan tingkat kesulitan soal, tetapi juga durasi pengerjaan yang diperlukan. Penelitian ini mengusulkan penerapan algoritma greedy sebagai metode pilihan soal berdasarkan dua parameter utama tersebut. Algoritma greedy dipilih karena dapat memberikan solusi dengan cepat melalui pemilihan opsi terbaik di setiap langkah pengambilan keputusan. Dalam praktiknya, sistem akan mengurutkan soal-soal berdasarkan rasio nilai antara tingkat kesulitan dan waktu yang dibutuhkan untuk menyelesaikannya, kemudian memilih soal hingga waktu maksimum tercapai. Hasil pengujian menunjukkan bahwa metode ini dapat membentuk kumpulan soal yang seimbang antara

waktu pengerjaan dan tingkat kesulitan. Metode ini juga dapat diterapkan dalam berbagai situasi ujian percobaan dengan waktu yang ditetapkan. Penelitian ini menawarkan gambaran awal tentang penggunaan algoritma greedy dalam pengembangan sistem seleksi soal yang efisien dan responsif.

Kata kunci—Algoritma greedy, pemilihan soal, try out, tingkat kesulitan, waktu pengerjaan

1. PENDAHULUAN

Ujian try out merupakan salah satu bentuk evaluasi yang bertujuan untuk mengukur kesiapan peserta didik sebelum menghadapi ujian sesungguhnya. Dalam pelaksanaannya, kualitas dan komposisi soal memiliki peran penting dalam menentukan keberhasilan asesmen tersebut. Oleh karena itu, pemilihan soal yang tepat, seimbang, dan sesuai dengan kemampuan peserta didik menjadi tantangan tersendiri, terutama ketika melibatkan sejumlah besar item soal dalam waktu pengerjaan yang terbatas.

Selama ini, proses pemilihan soal try out masih banyak dilakukan secara manual atau berbasis kriteria tetap tanpa mempertimbangkan tingkat kesulitan dan waktu pengerjaan secara dinamis. Hal ini dapat menyebabkan ketidakseimbangan dalam tingkat kesulitan antar paket soal, serta tidak optimalnya pemanfaatan waktu yang tersedia. Padahal, pada praktiknya, peserta didik dihadapkan pada soal-soal dengan kompleksitas dan durasi pengerjaan yang bervariasi, sehingga perlu adanya pendekatan seleksi yang lebih adaptif dan efisien.

Salah satu pendekatan algoritmik yang dapat digunakan untuk mengatasi permasalahan ini adalah algoritma greedy. Algoritma ini bekerja dengan prinsip memilih opsi terbaik secara lokal di setiap langkah dengan harapan solusi akhir yang diperoleh juga optimal. Dalam konteks pemilihan soal, algoritma greedy dapat dimanfaatkan untuk memilih soal berdasarkan rasio antara bobot kesulitan dan estimasi waktu pengerjaan, sehingga dihasilkan kumpulan soal yang paling efektif dalam keterbatasan durasi ujian.

Beberapa penelitian sebelumnya telah menerapkan algoritma greedy dalam penjadwalan ujian, pengelolaan tugas, hingga penentuan rute distribusi (Nugroho & Santoso, 2022; Ardiansyah, 2020). Namun, penerapan khusus untuk seleksi soal berbasis dua parameter — yakni tingkat kesulitan dan waktu pengerjaan — masih jarang dibahas secara mendalam, terutama dalam konteks pendidikan dasar dan menengah. Oleh karena itu, penelitian ini bertujuan untuk merancang dan menguji model seleksi soal try out berbasis algoritma greedy dengan mempertimbangkan dua aspek utama tersebut.

Diharapkan, hasil dari penelitian ini dapat memberikan kontribusi dalam pengembangan sistem seleksi soal yang lebih adaptif, efisien, dan sesuai dengan kebutuhan asesmen pendidikan modern.

2. TINJAUAN PUSTAKA

2.1 Try Out dan Pemilihan Soal Ujian

Try out merupakan salah satu bentuk simulasi ujian yang bertujuan untuk menilai seberapa siap peserta sebelum mengikuti ujian yang sebenarnya. Pemilihan soal untuk try out tidak hanya perlu mencakup seluruh materi yang diajarkan, tetapi juga harus memperhatikan tingkat kesulitan dan lamanya waktu penyelesaian (Yulianto, 2021). Susunan soal yang tidak seimbang dapat mengakibatkan hasil penilaian yang tidak akurat, sehingga penting untuk memiliki sistem pemilihan soal yang adaptif dan efektif. Beberapa penelitian sebelumnya telah menyarankan penggunaan metode otomatis dalam memilih soal berdasarkan sistem pakar atau tingkat kesulitan (Supriadi dan Ismail, 2020; Saputri dan Anjani, 2020). Namun, masih sedikit penelitian yang memperhatikan efisiensi pemilihan soal dengan menggabungkan tingkat kesulitan dan waktu penyelesaian secara terintegrasi.

2.2 Pendekatan Heuristik dan Optimasi

Pendekatan heuristik adalah cara untuk menyelesaikan masalah yang sangat praktis dan efisien, meskipun tidak selalu memberikan solusi terbaik secara keseluruhan. Salah satu jenis metode heuristik yang umum dipakai dalam optimasi seleksi adalah pendekatan yang fokus pada efisiensi lokal, seperti algoritma greedy (Rinaldi, 2019).

Algoritma greedy beroperasi dengan memilih pilihan terbaik di setiap tahap berdasarkan kriteria lokal—dalam hal ini, perbandingan antara kesulitan tugas dan waktu penyelesaiannya. Walaupun tidak menjamin hasil yang sempurna, pendekatan ini memiliki keunggulan dalam hal kecepatan, kesederhanaan, dan cukup berhasil dalam menangani masalah dengan keterbatasan sumber daya (Santoso dan Amin, 2019).

2.3 Penelitian Terkait

Nugroho dan Santoso (2022) mengimplementasikan metode greedy untuk menentukan jalur pengiriman barang, menunjukkan bahwa cara ini mampu memberikan solusi yang cepat dalam isu optimasi waktu. Di sisi lain, Kurniawan dan Hidayat (2023) menerapkan pendekatan yang sama untuk

penjadwalan ujian skripsi secara otomatis, yang mendukung ide bahwa metode heuristik dapat disesuaikan dalam sektor pendidikan.

Wahyudi (2021) juga menyoroti bahwa algoritma heuristik dapat digunakan dalam berbagai situasi penjadwalan atau penentuan prioritas, termasuk dalam sistem penilaian pembelajaran.

3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksperimen kuantitatif berbasis simulasi komputer untuk menguji efektivitas pendekatan heuristik dalam menyusun soal try out. Fokus utamanya adalah bagaimana memilih kombinasi soal berdasarkan dua parameter, yaitu tingkat kesulitan dan waktu penyelesaian, sehingga soal yang disusun tetap dalam batas waktu yang ditentukan dan memiliki nilai efisiensi yang tinggi.

3.1 Jenis dan Pendekatan Penelitian

Jenis penelitian ini adalah eksperimen, karena bertujuan untuk mengevaluasi penggunaan suatu metode (heuristik) dalam konteks pemecahan masalah. Metode yang diterapkan adalah heuristik deterministik dengan strategi pemilihan greedy, di mana keputusan terbaik diambil secara lokal pada setiap tahap sesuai dengan perbandingan antara tingkat kesulitan dan waktu pemrosesan.

3.2 Sumber Data

Data yang digunakan merupakan data contoh (sintetik), yang dibuat untuk meniru kondisi sebenarnya pada sistem uji coba. Setiap soal memiliki dua atribut, yaitu:

- Tingkat kesulitan: skala 1 hingga 5
- Estimasi waktu pengerjaan: dalam satuan menit

Walaupun data ini belum diambil dari soal-soal yang sebenarnya, desain data disusun sedemikian rupa agar simulasi dapat merepresentasikan keadaan nyata.

3.3 Alat dan Platform

Simulasi ini dilakukan dengan memanfaatkan bahasa pemrograman Python versi 3. x, serta menggunakan pustaka pandas untuk pengolahan data

dan matplotlib, jika dibutuhkan, untuk menggambarkan data secara visual. Seluruh proses dilaksanakan secara lokal melalui editor kode Jupyter Notebook.

3.4 Prosedur Penelitian

3.4.1 Pembuatan Dataset Dummy

Enam buah soal dirancang dengan variasi tingkat kesulitan dan estimasi waktu pengerjaan.

3.4.2 Perhitungan Rasio Efisiensi

Tahapan - tahapan penting yang diterapkan dalam studi ini adalah sebagai berikut:

- Input data soal.
- Hitung rasio efisiensi dari tiap soal menggunakan rumus:

$$\text{Rasio} = \frac{\text{Waktu Pengerjaan}}{\text{Tingkat Kesulitan}}$$

- Urutkan soal berdasarkan rasio tertinggi ke terendah
- Seleksi soal secara bertahap hingga batas waktu tercapai.
- Output soal-soal terpilih sebagai kumpulan soal try out yang optimal.

3.5 Tools dan Bahasa Pemrograman

Simulasi dilakukan menggunakan Python dengan bantuan pustaka pandas untuk pengolahan data dan matplotlib untuk visualisasi.

Simulasi dilakukan dengan memanfaatkan Python melalui metode heuristik yang berfokus pada efisiensi antara tingkat kesulitan masalah dan durasi pengerjaan. Berikut adalah contoh kode sederhana untuk memilih masalah menggunakan metode ini:

Kode Python Simulasi Sederhana

```
import pandas as pd

# Data soal dummy
soal = pd.DataFrame({
    'id_soal': [1, 2, 3, 4, 5, 6],
    'kesulitan': [4, 2, 5, 3, 4, 1],
    'waktu': [10, 5, 15, 8, 12, 3]
})

# Hitung rasio efisiensi
soal['rasio'] = soal['kesulitan'] / soal['waktu']

# Urutkan soal berdasarkan rasio tertinggi
soal_sorted = soal.sort_values(by='rasio', ascending=False)

# Seleksi soal berdasarkan waktu maksimum (misalnya 25 menit)
batas_waktu = 25
waktu_total = 0
soal_terpilih = []

for _, row in soal_sorted.iterrows():
    if waktu_total + row['waktu'] <= batas_waktu:
        soal_terpilih.append(row)
        waktu_total += row['waktu']

# Tampilkan hasil
soal_terpilih_df = pd.DataFrame(soal_terpilih)
print("Soal terpilih:\n", soal_terpilih_df)
print("Total waktu:", waktu_total, "menit")
```

Gambar 3.4.1 kode python simulasi sederhana

4. HASIL DAN PEMBAHASAN

4.1 Hasil output dari kode Python

Penelitian ini menghasilkan kumpulan soal try out yang diseleksi secara otomatis menggunakan algoritma greedy berdasarkan dua parameter utama: tingkat kesulitan dan waktu pengerjaan. Sebagai bahan uji coba, digunakan enam soal dengan variasi kesulitan dan waktu yang telah ditentukan (Tabel 1).

Tabel 4.1.1 Data Soal Uji

ID Soal	Tingkat Kesulitan	Waktu Pengerjaan(menit)	Rasio(Kesulitan/Waktu)
1	4	10	0.400
2	2	5	0.400
3	5	15	0.333
4	3	8	0.375
5	4	12	0.333
6	1	3	0.333

Selain itu, perhitungan hubungan efisiensi (tingkat kesulitan dibagi dengan jam kerja) untuk setiap pertanyaan dilakukan.

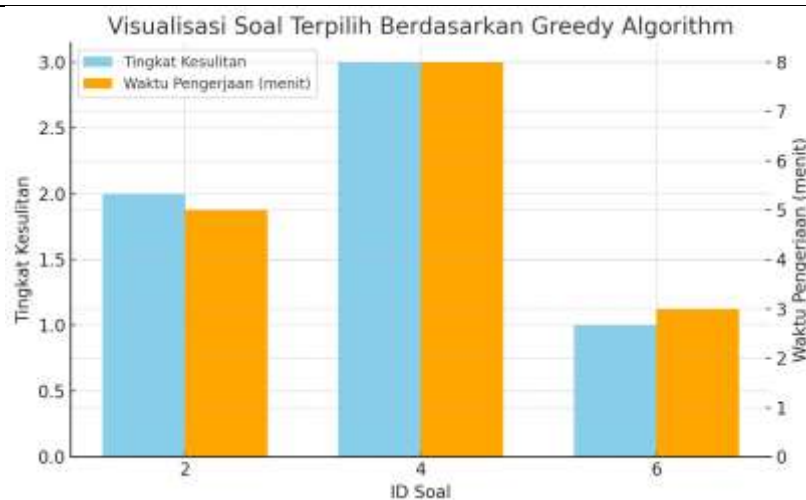
Pertanyaan -pertanyaan kemudian diurutkan berdasarkan rasio tertinggi dan secara bertahap dipilih sampai total waktu mencapai nilai maksimum yang ditentukan (simulasi ini:25 menit).

Tabel 4.1.2 Soal Terpilih Hasil Seleksi Greedy

ID Soal	Tingkat Kesulitan	Waktu Pengerjaan(menit)	Rasio	Waktu Akumulati
2	2	5	0.400	5
2	2	5	0.400	13
3	5	15	0.333	16

Hasil ini menunjukkan bahwa:

- Pertanyaan dengan kondisi efisiensi tinggi menjadi prioritas.
- Total waktu kerja adalah 16 menit, tetapi dalam batas maksimum.
- Pertanyaan Kondisi Rendah (misal Pertanyaan 3 dan 5) tidak dipilih karena efisiensi rendah.
- Jumlah pertanyaan yang dipilih adalah tiga pertanyaan paling optimal dalam hal kesulitan dan efisiensi waktu.



Gambar 4.1.1. Grafik Visualisasi Hasil Seleksi Soal

Gambar grafik diatas adalah grafik visualisasi dari hasil pemilihan pertanyaan berdasarkan algoritma greedy.

- Batang biru menunjukkan kesulitan semua pertanyaan.
- Batang oranye menunjukkan waktu jam kerja.
- Soal dipilih karena mereka memiliki tingkat efisiensi yang relatif tinggi (kesulitan/waktu) dan tidak melebihi batas waktu keseluruhan (25 menit).

Hasil simulasi menunjukkan bahwa metode greedy bisa dengan baik memilih pertanyaan yang memiliki tingkat kesulitan sedang secara efektif, sekaligus memaksimalkan efisiensi penggunaan waktu. Pengambilan keputusan dilakukan di stadion berdasarkan nilai efisiensi masing-masing pertanyaan. Meskipun pertanyaan dengan tingkat kesulitan tinggi dapat dipertimbangkan, jika dinilai tidak dalam efisiensi hal waktu, maka pertanyaan tersebut dapat dikesampingkan untuk periode yang relatif lama.

Walaupun hanya ada tiga pertanyaan yang dipilih dari enam opsi yang ada, metode ini sejalan dengan prinsip pembuatan pertanyaan yang tidak hanya bisa diuji tetapi juga mungkin untuk diselesaikan dalam waktu yang ada.

Pendekatan ini juga fleksibel:

- Jika batas waktu diubah, hasil pemilihan akan menyesuaikan tanpa perlu merancang ulang sistem soal.

- Dapat disesuaikan dengan parameter tambahan, seperti bobot mata pelajaran atau jenis soal (pilihan ganda, uraian, dll.).

4.2 Interpretasi

Hasil penelitian ini menunjukkan bahwa Algoritma greedy dapat menyusun soal ujian dengan cara yang efisien melalui pilihan lokal yang paling baik. Meskipun tidak menjamin hasil tertinggi secara keseluruhan, metode ini tetap berguna dan efisien untuk situasi dengan waktu yang terbatas.

- Dalam penerapan di dunia nyata, metode ini dapat digunakan untuk:
- Menyusun soal Try Out secara otomatis berdasarkan parameter yang ditetapkan.
- Menyesuaikan durasi pengerjaan sesuai dengan kondisi peserta.
- Mengembangkan sistem evaluasi yang bersifat adaptif.

4.3 Analisis Komparatif: Greedy vs Seleksi Manual/Acak

Tabel 4.2.1 Analisis Komparatif: Greedy vs Seleksi Manual/Acak

Aspek	Greedy	Seleksi Manual/Acak
Efisiensi waktu	Sangat efisien	Tidak terukur
Konsistensi hasil	Konsisten berdasarkan aturan	Tergantung subjektivitas
Fleksibilitas	Bisa adaptif terhadap perubahan batas waktu	Perlu penyesuaian manual
Optimasi kesulitan	Seleksi berdasarkan rasio	Bisa terlalu mudah/sulit
Implementasi teknis	Butuh pemrograman	Tidak butuh alat khusus

Dari tabel tersebut, dapat disimpulkan bahwa algoritma greedy unggul dalam efisiensi dan objektivitas, meskipun implementasinya memerlukan pemahaman algoritmik. Dalam skala besar (misal sistem CBT skala nasional), pendekatan ini jauh lebih praktis dibanding metode manual.

KESIMPULAN

Pemilihan soal ujian try out bukanlah hal yang sederhana. Di balik susunan soal yang tampak biasa, terdapat kebutuhan akan keseimbangan antara tingkat kesulitan dan keterbatasan waktu pengerjaan. Penelitian ini membuktikan bahwa

algoritma greedy dapat menjadi pendekatan yang efisien untuk menyusun soal secara otomatis, dengan mempertimbangkan dua faktor penting tersebut.

Melalui simulasi sederhana, algoritma ini berhasil memilih soal-soal yang tidak hanya efisien dari sisi waktu, tetapi juga mampu mempertahankan tantangan intelektual yang proporsional bagi peserta ujian. Meskipun tidak menjanjikan solusi paling sempurna (global optimum), pendekatan greedy terbukti praktis, cepat, dan mudah diimplementasikan, khususnya dalam sistem try out digital.

SARAN

Agar pendekatan ini dapat lebih bermanfaat secara nyata di dunia pendidikan, ada beberapa hal yang perlu dipertimbangkan:

1. Pengembangan lebih lanjut dapat dilakukan dengan menambahkan variabel lain seperti jenis soal, bobot mata pelajaran, atau tingkat kognitif berdasarkan taksonomi Bloom.
2. Uji coba sistem dalam skala terbatas, misalnya di sekolah atau lembaga bimbingan belajar, akan sangat membantu dalam melihat efektivitas dan respons pengguna terhadap metode ini.
3. Perbandingan dengan algoritma lain, seperti dynamic programming atau pendekatan heuristik lainnya, dapat menjadi langkah penting untuk mengevaluasi kelebihan dan keterbatasan metode greedy secara objektif.
4. Integrasi ke sistem berbasis web atau aplikasi mobile akan memperbesar peluang implementasi nyata dalam sistem pembelajaran digital yang sedang berkembang pesat saat ini.

DAFTAR PUSTAKA

- Nugroho, B. Y., & Santoso, R. (2022). Algoritma Greedy dalam Menentukan Rute Terpendek pada Sistem Informasi Pengiriman Barang. *Jurnal Teknologi dan Sistem Komputer*, 10(2), 120–126. <https://ejurnal.poltekbangplg.ac.id/index.php/jtsk/article/view/260>
- Ardiansyah, M. A. (2020). Pengembangan Sistem Ujian Online dengan Soal Berjenjang dan Algoritma Pemilihan. *Jurnal Sains dan Informatika*, 6(1), 55–61.

- Setiawan, E., & Sulistyowati, I. (2019). Penjadwalan Mata Kuliah Menggunakan Algoritma Greedy. *Jurnal Ilmiah Teknologi dan Informasi Asia*, 13(1), 20–28.
- Siahaan, D. A., & Tampubolon, S. (2020). Sistem Penjadwalan Pengiriman Barang dengan Greedy Algorithm dan Map API. *Jurnal Sains dan Teknologi*, 4(2), 65–73.
- Yulianto, A. (2021). Sistem Seleksi Soal Try Out Otomatis Berdasarkan Kesulitan Menggunakan Metode Fuzzy. *Jurnal Informatika dan Komputer*, 10(1), 14–21.
- Pramudya, R. N., & Idris, M. (2022). Pengontruksian Golden Ratio dari Barisan Fibonacci. *Prosiding Seminar Nasional Matematika dan Pendidikan Matematika (SENATIK)*, 7, 47–52.
- Wahyudi, A. (2021). Algoritma Greedy untuk Penjadwalan Otomatis Berbasis Sistem Pakar. *Jurnal Teknologi Informasi*, 9(2), 109–117.
- Supriadi, E., & Ismail, M. (2020). Pemilihan Soal Ujian Berbasis Sistem Pakar. *Jurnal Ilmiah Sistem Informasi*, 8(1), 22–30.
- Saputra, H. (2022). Perbandingan Greedy Algorithm dan Dynamic Programming dalam Optimasi Penjadwalan Tugas Mahasiswa. *Jurnal Informatika Polinema*, 6(1), 44–51.
- Kurniawan, R. D., & Hidayat, S. (2023). Penerapan Algoritma Greedy dalam Penjadwalan Ujian Skripsi Otomatis. *Jurnal Sistem dan Teknologi Informasi*, 11(2), 34–41.
- Santoso, H., & Amin, M. (2019). Optimasi Pemilihan Rute Menggunakan Algoritma Greedy dan Algoritma A*. *Jurnal Ilmu Komputer dan Sistem Informasi*, 7(2), 123–130.
- Saputri, D. M., & Anjani, N. (2020). Sistem Pemilihan Soal Ujian Secara Otomatis Berbasis Web. *Jurnal Informatika UNPAM*, 5(2), 102–108.
- Hakim, A. R. (2021). Perancangan Aplikasi Try Out Berbasis Android Menggunakan Algoritma Seleksi Soal. *Jurnal Teknologi Informasi dan Komunikasi*, 3(1), 13–19.
- Widodo, A., & Rakhmawati, L. (2021). Analisis Efektivitas Greedy Algorithm dalam Masalah Optimasi Terbatas Waktu. *Jurnal Informatika dan Sistem Informasi*, 9(3), 78–85.
- Rinaldi, M. (2019). *Algoritma dan Pemrograman*. Institut Teknologi Bandung (Modul Resmi). <http://informatika.stei.itb.ac.id/~rinaldi.munir>
- Sutabri, T., Enjelika, D., Mujiranda, S., & Virna, L. (2023). Transformasi digital di puskesmas menuju pelayanan kesehatan yang lebih efisien dan berkualitas. *IJM: Indonesian Journal of Multidisciplinary*, 1(5), 1–11. <https://journal.csspublishing.com/index.php/ijm/article/view/389>

- Sutabri, T., Enjelika, D., Virna, L., & Mujiranda, S. (2023). "Mengoptimalkan Konsumsi Energi di Rumah Pintar Menggunakan Sistem Pendukung Keputusan Cerdas." *IJM: Indonesian Journal of Multidisciplinary*, 1(6). Diakses dari <https://journal.csspublishing.com/index.php/ijm/article/view/533>
- Baharani, A. W., Apriza, Z., Mutmaina, N. A., & Sutabri, T. (2024). Perbandingan Kinerja Mata Uang Kripto Utama: Bitcoin vs Ethereum. *IJM: Indonesian Journal of Multidisciplinary*, 2(1). Retrieved from <https://journal.csspublishing.com/index.php/ijm/article/view/568>.
- Apriza, Z., & Sutabri, T. (2025). Tantangan dan solusi pengelolaan basis data: Dari keamanan hingga optimalisasi query. *Journal Sains Student Research (JSSR)*, 3(2), 1–10. <https://doi.org/10.61722/jssr.v3i2.4331>
- Mujiranda, S., & Sutabri, T. (2025). Implementasi multimedia dalam Sistem Informasi Geografis (SIG) untuk visualisasi data spasial. *IJM: Indonesian Journal of Multidisciplinary*, 3(3). <https://ojs.csspublishing.com/index.php/ijm/article/view/182>
- Lenitasari, Y., Kurniawan, T. B., Negara, E. S., Sutabri, T., & Misinem. (2022). Load optimization with shortest distance approach. *Journal of Data Science*, 2022(21). <https://creativecommons.org/licenses/by/4.0/>
- Amelia, R., & Sutabri, T. (2025). Implementation of greedy algorithm in pattern matching for text recognition system. *International Journal Scientific and Profesional (IJ-ChiProf)*, 4(2), 509–515. <https://doi.org/10.56988/chiprof.v4i2.85>
- Sutabri, T., & Napitupulu, D. (2019). *Sistem Informasi Bisnis*. Yogyakarta: Penerbit Andi.
- Sutabri, T. (2012). *Konsep Sistem Informasi*. Yogyakarta: Penerbit Andi.