

IMPLEMENTASI ALGORITMA DECISION TREE UNTUK PEMBELAJARAN KOLABORATIF KELAS DIGITAL BERBASIS ANDROID

Muhamad Abi Samsu Sidqi¹, Intan Mutia², Muslihatul Hidayah³

^{1,2,3} Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Indraprasta PGRI
Jalan Raya Tengah No.80, Kelurahan Gedong, Pasar Rebo, Jakarta Timur
hi@abisamsu.dev¹, as.syifaraa@gmail.com², muslihatulhidayah@gmail.com³

Abstrak

Perkembangan teknologi mendorong digitalisasi dalam dunia pendidikan untuk meningkatkan efektivitas dan efisiensi proses pembelajaran. Namun, pengelolaan kelas secara manual masih menghadapi berbagai kendala, seperti risiko kehilangan dokumen, kesulitan dalam mengelola tugas, serta kurangnya pengingat bagi siswa. Permasalahan serupa juga ditemukan dalam kegiatan pembelajaran di Majelis Taklim Mubarak RT 02/09. Oleh karena itu, penelitian ini mengembangkan kelas digital berbasis Android dengan menerapkan algoritma Decision Tree untuk membantu pengelolaan dokumen, mengklasifikasikan prioritas tugas, serta menyediakan fitur pengingat dan komunikasi secara real-time antara guru dan siswa. Dataset yang digunakan terdiri atas 139 data tugas, dengan 111 data sebagai data latih dan 28 data sebagai data uji. Proses klasifikasi dilakukan berdasarkan pembentukan pohon keputusan menggunakan nilai entropy dan information gain pada setiap atribut. Hasil evaluasi model menunjukkan bahwa algoritma Decision Tree menghasilkan akurasi sebesar 89,29%, precision sebesar 89,43%, recall sebesar 89,29%, weighted F1-score sebesar 89,20%, dan macro F1-score sebesar 89,77%. Model mampu mengklasifikasikan prioritas tugas ke dalam tiga kategori, yaitu HIGH, MEDIUM, dan LOW, sehingga dapat membantu siswa dalam menentukan tugas yang perlu diprioritaskan. Hasil penelitian menunjukkan bahwa penerapan Decision Tree pada kelas digital berbasis Android memiliki kinerja yang baik dan berpotensi mendukung pengelolaan serta pembelajaran digital secara lebih terstruktur.

Kata Kunci : Android, JavaScript, React Native, MySQL, *Decision Tree*, Kolaboratif

Abstract

The technological developments are driving the digitalization of education to enhance the effectiveness and efficiency of the learning process. However, manual classroom management still faces various challenges, such as the risk of document loss, difficulties in managing assignments, and a lack of reminders for students. Similar issues are encountered in the learning activities at Majelis Taklim Mubarak RT 02/09. Therefore, this study develops an Android-based digital classroom utilizing the Decision Tree algorithm to assist with document management, classify assignment priorities, and provide real-time reminder and communication features for teachers and students. The dataset comprised 139 assignment records, split into 111 for training and 28 for testing. The classification process involved constructing a decision tree based on entropy and information gain values for each attribute. Model evaluation results indicate that the Decision Tree algorithm achieved an accuracy of 89.29%, precision of 89.43%, recall of 89.29%, a weighted F1-score of 89.20%, and a macro F1-score of 89.77%. The model successfully classified assignment priorities into three categories—HIGH, MEDIUM, and LOW—thereby helping students identify which assignments require priority. The findings demonstrate that implementing the Decision Tree algorithm in an Android-based digital classroom yields strong performance and holds the potential to support more structured management and digital learning.

Keyword : Android, Javascript, React, MySQL, Native, *Decision Tree*, Collaborative

PENDAHULUAN

Perkembangan erbagai penelitian sebelumnya telah membuktikan efektivitas metode Decision Tree dalam proses klasifikasi data pada konteks pendidikan. Dwi Lestari dan Sri Lestari menunjukkan bahwa klasifikasi dengan metode Decision Tree dapat diterapkan pada kasus pemahaman kegiatan belajar mengajar siswa-siswi SDN, namun penelitian tersebut masih bersifat deskriptif-analitis dan belum diimplementasikan ke dalam suatu sistem aplikasi yang dapat digunakan secara langsung oleh guru maupun siswa dalam aktivitas kelas sehari-hari

Berdasarkan kajian tersebut, terlihat adanya gap penelitian pada 10 aspek utama: [1] belum adanya penelitian yang mengintegrasikan algoritma Decision Tree secara langsung ke dalam sistem manajemen

tugas berbasis mobile untuk mengklasifikasikan status penyelesaian tugas siswa secara otomatis; [2] minimnya penelitian yang menyorot segmen pengguna lintas jenjang (TK hingga tingkat akhir) dalam satu platform terpadu, khususnya pada konteks lembaga pendidikan nonformal seperti majelis taklim; dan [3] belum ada penelitian yang mengombinasikan arsitektur full-stack modern (React Native, Express JS, dan MySQL) dengan pendekatan klasifikasi cerdas untuk memberikan transparansi aktivitas kelas secara real-time; [4] mayoritas pemanfaatan algoritma Decision Tree di sektor pendidikan masih terfokus pada prediksi nilai akhir, rekomendasi jurusan mahasiswa baru, dan klasifikasi pemilihan sekolah lanjutan, namun belum dimanfaatkan untuk klasifikasi tingkat prioritas penyelesaian tugas harian siswa; [5] penelitian terkait penerapan data mining dalam pendidikan lebih banyak difokuskan untuk memetakan pola kognitif dan tingkat pemahaman mahasiswa, namun belum menyentuh aspek manajemen waktu dan beban tugas peserta didik; [6] algoritma Decision Tree telah banyak dan sukses diimplementasikan untuk sistem pendukung keputusan di sektor bisnis seperti penentuan kredit, klasifikasi keaktifan nasabah bank, serta analisis penjualan, namun belum banyak diadaptasi ke dalam fitur pengelolaan administrasi kelas digital; [7] studi mengenai pembelajaran kolaboratif umumnya masih berpusat pada analisis dampaknya terhadap keterampilan sosial dan akademik peserta didik, tanpa dibarengi dengan pengembangan alat bantu digital (tools) berbasis algoritma cerdas untuk mendukung kolaborasi tersebut; [8] sistem data mining pada penelitian terdahulu umumnya masih berupa analisis komputasi deskriptif pada lingkungan belajar konvensional, dan belum dirancang menjadi fitur operasional yang real-time di dalam aplikasi berbasis Android; [9] referensi konseptual mengenai dasar algoritma pemrograman belum banyak yang mengeksplorasi pengintegrasian komputasi Decision Tree (penghitungan entropy dan information gain mandiri) secara langsung di dalam ekosistem JavaScript dan React Native; serta [10] belum adanya pengembangan model pohon keputusan yang secara khusus meramu kriteria operasional tugas (seperti waktu pengerjaan, panjang deskripsi, ketersediaan lampiran, dan rentang nilai) untuk menghasilkan indikator prioritas (HIGH, MEDIUM, LOW) guna meminimalisir keterlambatan pengumpulan tugas.

Penelitian ini berkontribusi mengisi celah tersebut dengan mengembangkan aplikasi pembelajaran kolaboratif kelas digital berbasis Android yang tidak hanya berfungsi sebagai media penyimpanan dan penyampaian materi, tetapi juga menerapkan algoritma Decision Tree sebagai

Penanda klasifikasi manajemen tugas—suatu pendekatan yang belum ditemukan pada penelitian-penelitian terdahulu—sehingga sistem mampu memberikan indikasi dini terhadap risiko keterlambatan siswa dan menjadi acuan pengembangan sistem machine learning yang lebih adaptif di masa mendatang.

METODE PENELITIAN

Waktu Penelitian

Penelitian ini dilakukan dengan beberapa tahapan yang disusun secara sistematis dalam kurun waktu yang ditentukan Maret 2026 hingga Juni 2026. Tahap pertama adalah merumuskan data dan melakukan pra-pemrosesan data yang diperlukan yang berlangsung dua minggu pertama pada bulan Maret 2026 sampai minggu pertama pada bulan April 2026. Penelitian ini dilakukan di lokasi Majelis Taklim Mubarak, Jalan Raya Pitara No 16. RT.2/RW.9, Pancoranmas, Kota Depok, Jawa Barat, 16436.

Sumber Dataset

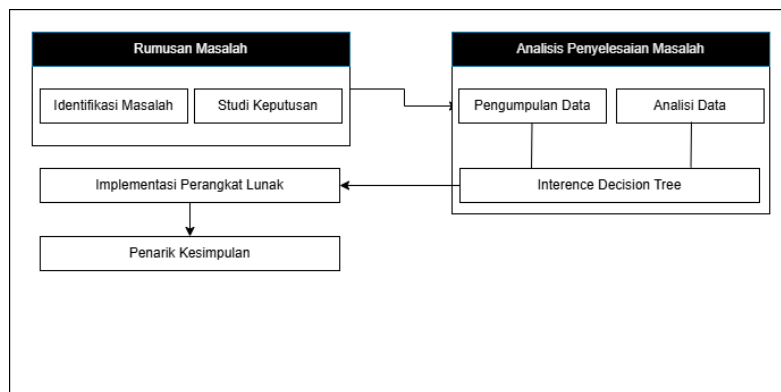
Dataset yang digunakan dalam penelitian ini diperoleh dari arsip aktivitas pembelajaran pada Google Classroom kelas-kelas terdahulu selama masa pandemi COVID-19, ketika proses pembelajaran dilaksanakan secara daring. Data tersebut mencerminkan aktivitas siswa selama mengikuti pembelajaran online, seperti pengumpulan tugas, ketepatan waktu pengumpulan, keaktifan dalam diskusi, serta nilai atau hasil tugas. Penggunaan data historis tersebut dipilih karena dapat memberikan gambaran nyata mengenai pola keterlibatan siswa dalam proses pembelajaran digital.

Data yang diperoleh selanjutnya digunakan sebagai dataset untuk proses pengolahan dan klasifikasi

menggunakan algoritma Decision Tree. Sebelum digunakan, data terlebih dahulu melalui tahap seleksi, pembersihan, dan transformasi agar sesuai dengan kebutuhan penelitian. Informasi yang bersifat pribadi atau dapat mengidentifikasi siswa dianonimkan, sedangkan atribut yang berkaitan dengan aktivitas pembelajaran dipertahankan sebagai variabel penelitian. Hasil pengolahan dataset tersebut kemudian digunakan untuk membangun model yang dapat mengidentifikasi pola aktivitas dan keterlibatan siswa sebagai dasar dalam mendukung pembelajaran kolaboratif pada kelas digital berbasis Android.

Tahapan Penelitian

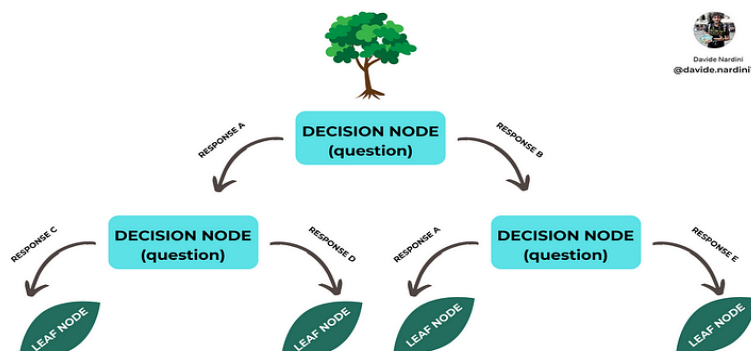
Pada tahap ini penelitian melakukan pengumpulan data ke majelis taklim mubarak dan mewawancarai ketua komite majelis taklim mubarak. Data yang diperoleh kemudian akan dilengkapi terlebih dahulu. Apa bila data belum lengkap maka proses pengambilan dan pengumpulan data diulangi hingga data terpenuhi. Setelah data dinyatakan lengkap proses selanjutnya adalah membangun sistem yang mengimplementasikan algoritma Decision Tree dalam klasifikasi tugas berdasarkan prioritasnya. Sistem ini akan melakukan pemrosesan dengan rumus yang sudah ditentukan lalu melakukan validasi setiap hasil pemrosesan algoritma tersebut. Tahapan terakhir adalah menyusun saran dan kesimpulan dari hasil penelitian, yang kemudian menandakan berakhirnya proses penelitian secara menyeluruh.



Gambar 1. Workflow Tahapan Penelitian

Decision Tree

Algoritma Decision tree digunakan untuk membantu proses untuk mengklasifikasikan dalam menentukan apakah sebuah tugas yang diberikan prioritas atau tidak. Metode ini dilakukan dengan mencari nilai-nilai pada setiap kriteria atau fitur pada data yang akan dijadikan sebuah dataset. Pembentukan ini dihitung dengan cara satu persatu entrophy dan gain lalu diurutkan dengan nilai gain yang paling terbesar untuk menentukan root atau pohon utama dari sebuah decision tree. Yang dimana pohon ini sebagai penentu sistem dalam menentukan prioritas setiap tugas yang baru dibuat



Gambar 2. Konsep Pohon Decision Tree

Pelatihan Data

Dataset Data yang digunakan berjumlah 139 data tugas (*HIGH* = 46, *MEDIUM* = 52, *LOW* = 41). Data tersebut menghasilkan 111 data latih dan 28 data uji. Pembagian dilakukan secara stratified sampling, yaitu proporsi masing-masing kelas (*HIGH*, *MEDIUM*, *LOW*) dijaga tetap seimbang baik pada data latih maupun data uji, agar model tidak bias terhadap kelas yang jumlahnya lebih dominan (dalam hal ini *MEDIUM*).

Proses Data

Sebelum data digunakan untuk pemodelan, dilakukan beberapa tahap preprocessing. Tahap pertama adalah data cleaning, yaitu penyaringan (*filtering*) data karena dataset mentah berisi baris kosong dan tabel perhitungan entropy manual yang tercampur dengan data tugas asli, sehingga hanya baris yang

berisi data tugas valid dengan label *HIGH/MEDIUM/LOW* yang digunakan. Selanjutnya dilakukan standarisasi label untuk menyeragamkan variasi penulisan label (huruf besar/kecil, spasi berlebih) menjadi format yang konsisten. Pada tahap encoding atribut kategorikal, atribut seperti Jenis Tugas dan Tempat Pengerjaan yang sudah direpresentasikan dalam bentuk biner (0/1), serta Tingkat Kesulitan yang direpresentasikan dalam skala numerik ordinal (1–5), tidak memerlukan encoding tambahan dan dapat langsung diproses oleh algoritma Decision Tree. Terakhir, dilakukan pemeriksaan missing value dengan mengeliminasi baris yang memiliki nilai kosong pada salah satu atribut agar tidak mengganggu proses perhitungan entropy dan information gain.

Validasi Data

Validasi model dilakukan dengan pendekatan train-test split dan diperkuat dengan pengujian tambahan melalui confusion matrix untuk melihat distribusi prediksi benar dan salah pada tiap kelas. Evaluasi performa model menggunakan empat metrik utama: akurasi, precision, recall, dan F1-score, guna memastikan model tidak hanya akurat secara umum tetapi juga konsisten dalam mengklasifikasikan tiap kategori prioritas tugas. Sebagai validasi tambahan, dilakukan pula 10-fold cross-validation untuk menguji kestabilan model pada berbagai kombinasi data latih dan data uji, guna memastikan hasil akurasi tidak hanya bergantung pada satu skema pembagian data tertentu.

```
=====
 EVALUASI MODEL: Decision Tree Tuned (Best)
=====

Accuracy           : 89.29%
F1-Score (weighted) : 89.20%
F1-Score (macro)    : 89.77%
Precision (weighted) : 89.43%
Recall (weighted)    : 89.29%

 Classification Report:
      precision    recall  f1-score   support

    HIGH           0.88     0.78     0.82         9
    LOW            1.00     1.00     1.00         8
    MEDIUM         0.83     0.91     0.87        11

 accuracy                   0.89         28
 macro avg                 0.90         28
 weighted avg              0.89         28
```

Gambar 3. Hasil Validas

HASIL DAN PEMBAHASAN

Penentuan Kriteria atau Fitur

Penelitian ini menggunakan 8 kriteria atau fitur sebagai acuan dataset dalam mengklasifikasi sebuah tugas, yaitu Deadline (Hari), Panjang Deskripsi (Karakter), Lampiran (Jumlah Files), Waktu Pengerjaan (Jam), Jenis Tugas (1 Personal, 0 Kelompok), Tempat Pengerjaan (1 Tulis Tangan, 0 Digital File), Batasan Nilai (10-100), Tingkat Kesulitan (1-5).

Tabel 1. Kriteria	
Kriteria	Keterangan

Deadline	Hari
Panjang Deskripsi	Karakter
Lampiran	Jumlah Files
Waktu Pengerjaan	Jam
Jenis Tugas	1 (Personal), 0 (Kelompok)
Tempat Pengerjaan	1 (Tulis Tangan), 0 (Digital Files)
Batasan Nilai	(10 - 100)
Tingkat Kesulitan	(1 - 5)

Perhitungan Entropy

Perhitungan dilakukan secara satu persatu setiap kriteria yang sudah ditentukan dengan rumus:

$$Info(D) = - \sum_{i=1}^m p_i \log_2(p_i)$$

Keterangan:

$Info(D)$: Fitur yang akan dihitung
 p_i : Proporsi fitur ke-i
 $\log_2$: Logaritma basis 2
 $\sum_{i=1}^m$: penjumlahan untuk semua kelas.

Contoh Menghitung Kriteria Deadline:

Tabel 2. Data Clean Kriteria

Deadline Tuas	Medium	Low	High	Jumlah
<= 10	17	2	11	30
> 10 && <=30	2	6	2	10
Jumlah	19	8	13	40

$$\leq 10 : \left(-\frac{17}{30} \times \log_2 \left(\frac{17}{30} \right) \right) + \left(-\frac{2}{30} \times \log_2 \left(\frac{2}{30} \right) \right) + \left(-\frac{11}{30} \times \log_2 \left(\frac{11}{30} \right) \right)$$

: 1,255536725

$$> 10 \&\& \leq 30 : \left(-\frac{2}{10} \times \log_2 \left(\frac{2}{10} \right) \right) + \left(-\frac{6}{10} \times \log_2 \left(\frac{6}{10} \right) \right) + \left(-\frac{2}{10} \times \log_2 \left(\frac{2}{10} \right) \right)$$

: 1,370950594

$$E(Deadline) : 1,255536725 - 1,370950594$$

: 1,284390193

Tabel 3. Hasil Perhitungan Entropy

Kriteria	Entropy
Deadline	1,284390193
Panjang Deskripsi	1,499587182
Lampiran	1,390008222
Waktu Pengerjaan	0,862405459
Jenis Tugas	1,373692709
Tempat Pengerjaan	1,472089759
Batasan Nilai	0,95260079
Tingkat Kesulitan	1,195233874

Perhitungan Gain

Perhitungan dilakukan secara satu persatu setiap entropy yang sudah dihitung dengan rumus:

$$Information(Y, X) = E(Y) - E(Y|X)$$

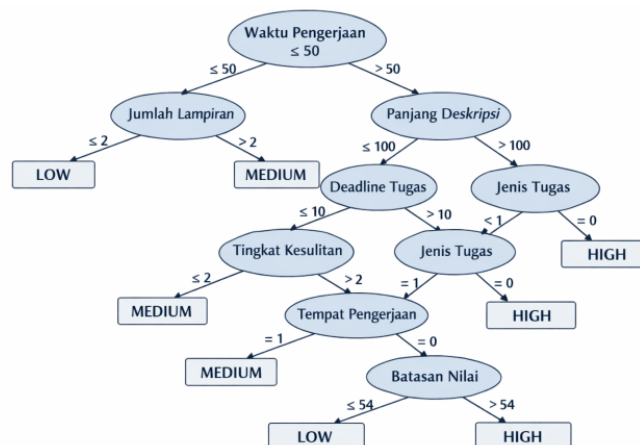
Keterangan:

Y : Variabel target/kelas yang ingin diprediksi.
 X : Atribut yang akan diuji sebagai pemisah.

Gain : 1,501519618 – 1,284390193
: 0,217129425

Tabel 4. Hasil Perhitungan Gain

Kriteria	Gain
Deadline	0,217129425
Panjang Deskripsi	0,001932436
Lampiran	0,111511396
Waktu Pengerjaan	0,639114158
Jenis Tugas	0,127826908
Tempat Pengerjaan	0,029429858
Batasan Nilai	0,548918828
Tingkat Kesulitan	0,306285743

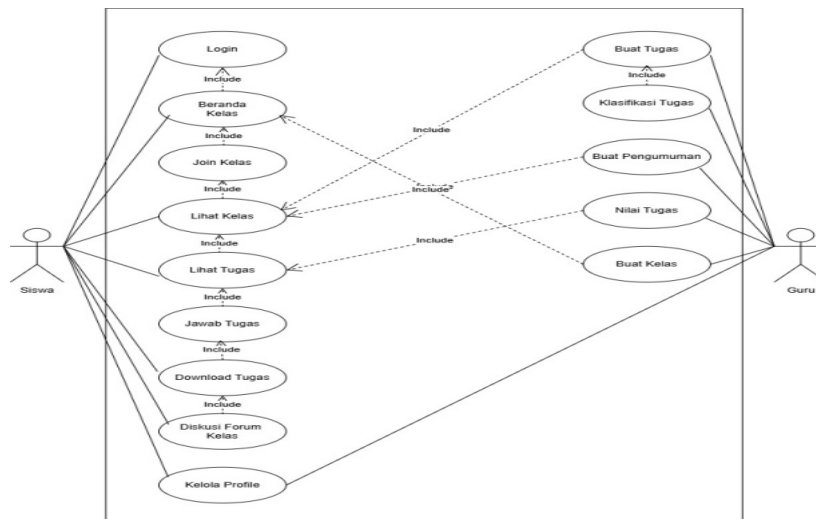


Gambar 4. Hasil Keputusan Decision Tree

Unified Modeling Language (UML)

merupakan sebuah bahasa yang di terjemahkan menjadi visual grafis untuk mendokumentasi sebuah hasil analisis dan design. Juga merupakan berisi symbol-simbol dan diagram untuk menerjemahkan software. Penggunaan UML ditujukan untuk memakai bahasa pemrograman apa saja selama bahasa tersebut mendukung Object Oriented Programming (OOP), karena metode OOP menurut para ahli merupakan metedologi yang terbaik di terapkan pada UML.

a. Use Case Diagram



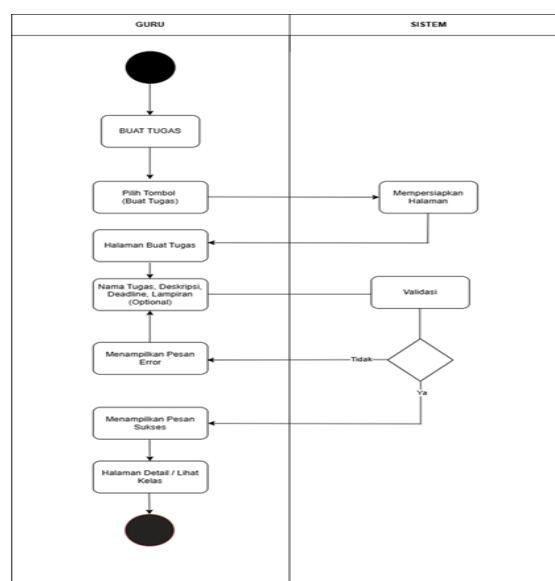
Gambar 5. Use Case Diagram

Diagram use case ini menggambarkan alur interaksi antara pengguna dan sistem pembelajaran daring (e-learning), yang melibatkan dua aktor utama, yaitu Siswa dan Guru. Secara umum, kedua aktor wajib melakukan autentikasi melalui proses Login sebelum dapat mengakses fungsi utama sistem, serta sama-sama memiliki hak akses untuk Kelola Profile pengguna masing-masing.

Di sisi Siswa, pengguna dapat mengakses Lihat Kelas dan Join Kelas yang terhubung langsung dengan Beranda Kelas. Dalam pembelajaran, siswa dapat melakukan Lihat Tugas yang berlanjut pada proses Jawab Tugas dan Download Tugas untuk mengunduh materi atau lembar kerja. Selain itu, siswa juga dapat berinteraksi secara aktif melalui fitur Diskusi Forum Kelas.

Sementara itu, di sisi Guru, aktor ini memiliki wewenang untuk mengelola aktivitas pembelajaran dengan fitur Buat Kelas dan Buat Pengumuman. Guru juga bertugas menyusun evaluasi pembelajaran melalui fitur Buat Tugas yang mencakup Klasifikasi Tugas, serta melakukan pemeriksaan dan Nilai Tugas atas hasil pekerjaan yang telah dikirimkan oleh siswa.

b. Activity Diagram



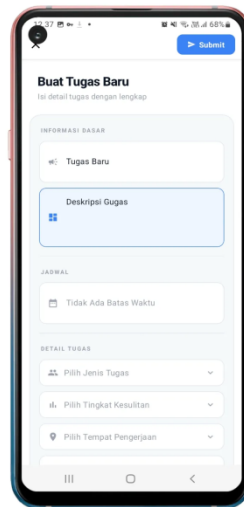
Gambar 6. Activity Buat Tugas

Diagram aktivitas menggambarkan alur kerja (workflow) interaksi antara Guru dan Sistem dalam proses pembuatan tugas baru. Proses dimulai saat Guru memilih tombol "Buat Tugas", yang memicu Sistem untuk mempersiapkan dan menampilkan halaman form pembuatan tugas. Selanjutnya, Guru mengisi data berupa Nama Tugas, Deskripsi, Deadline, serta Lampiran (Opsional) dan mengirimkannya ke Sistem.

Sistem kemudian melakukan proses Validasi terhadap data yang dimasukkan. Jika data tidak valid (Tidak), Sistem akan memerintahkan antarmuka untuk Menampilkan Pesan Error agar Guru dapat memperbaikinya. Sebaliknya, jika data valid (Ya), Sistem akan mengirimkan konfirmasi sehingga antarmuka Menampilkan Pesan Sukses, lalu mengarahkan Guru ke halaman Detail / Lihat Kelas, dan alur aktivitas pun selesai.

Tampilan Layar

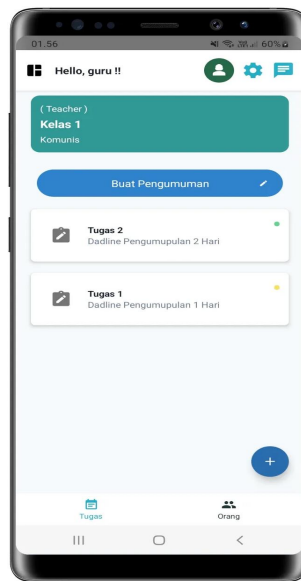
Tampilan Layar Buat Tugas



Gambar 7. Tampilan Buat Tugas

Halaman Buat Tugas Baru (Create Assignment Page) berfungsi sebagai sarana bagi pengajar untuk merancang dan mempublikasikan tugas akademik baru ke dalam sistem. Tampilan ini menyajikan formulir pengisian data yang terstruktur ke dalam beberapa bagian, meliputi bagian Informasi Dasar (Judul Tugas dan Deskripsi Tugas), bagian Jadwal (Batas Waktu), serta bagian Detail Tugas yang memuat opsi pilihan seperti Pilih Jenis Tugas, Pilih Tingkat Kesulitan, dan Pilih Tempat Pengerjaan. Dan setiap formulir yang di isi nanti akan dijadikan sebagai bahan acuan untuk menjalankan algoritma decision tree. Halaman ini juga dilengkapi dengan tombol navigasi tutup (X) di pojok kiri atas dan tombol aksi Submit di pojok kanan atas untuk menyimpan dan mendistribusikan tugas kepada seluruh siswa di kelas.

Tampilan Layar Data Tugas



Gambar 8. Tampilan Layar Data Tugas

Halaman Detail Kelas (Class Detail Page) berfungsi untuk menyajikan informasi spesifik mengenai kelas yang sedang diakses serta daftar tugas akademik yang perlu diselesaikan oleh pengguna. Tampilan ini memuat header personalisasi salam, card identitas kelas (seperti nama kelas dan mata pelajaran), daftar tugas (Tugas 1 dan Tugas 2) yang dilengkapi dengan informasi tenggat waktu pengumpulan (deadline), serta indikator status berbentuk titik warna yang merupakan penanda algoritma apakah tugas tersebut merupakan prioritas atau tidak. Pada bagian bawah antarmuka, terdapat bottom navigation bar dengan dua menu utama (Tugas dan Orang) untuk memudahkan pengguna berpindah antara daftar tugas dan daftar anggota kelas.

SIMPULAN

Berdasarkan hasil penelitian dan implementasi sistem, dapat disimpulkan bahwa penerapan metode Decision Tree mampu membantu proses penentuan tugas yang diprioritaskan dalam kelas. Sistem yang dikembangkan berhasil mengolah data menjadi hasil rekomendasi yang dibuat secara otomatis. Model mampu membentuk pohon keputusan dari hasil perhitungan yang dilakukan untuk menentukan kriteria yang relevan dalam penentuan prioritas tugas. Sistem ini diharapkan dapat memberikan manfaat kepada siswa maupun mahasiswa yang membutuhkan informasi mengenai tugas yang harus diprioritaskan terlebih dahulu.

Berdasarkan hasil penelitian dan implementasi sistem, dapat disimpulkan bahwa penerapan algoritma Decision Tree mampu membantu proses klasifikasi prioritas tugas pada kelas digital berdasarkan atribut waktu pengerjaan, jumlah lampiran, panjang deskripsi, deadline tugas, jenis tugas, tingkat kesulitan, tempat pengerjaan, dan batasan nilai. Model Decision Tree yang telah melalui proses tuning menghasilkan akurasi sebesar 89,29%, dengan weighted F1-score sebesar 89,20%, macro F1-score sebesar 89,77%, precision sebesar 89,43%, dan recall sebesar 89,29% pada 28 data uji. Berdasarkan hasil klasifikasi, kelas LOW memperoleh precision, recall, dan F1-score masing-masing sebesar 1,00, kelas MEDIUM memperoleh precision 0,83, recall 0,91, dan F1-score 0,87, sedangkan kelas HIGH memperoleh precision 0,88, recall 0,78, dan F1-score 0,82. Hasil tersebut menunjukkan bahwa model memiliki kemampuan klasifikasi yang baik, terutama dalam mengenali tugas dengan kategori LOW dan MEDIUM, meskipun masih terdapat kesalahan klasifikasi pada kategori HIGH. Dengan demikian, sistem yang dikembangkan dapat digunakan sebagai pendukung dalam memberikan informasi mengenai tingkat prioritas tugas kepada siswa maupun mahasiswa. Penelitian selanjutnya disarankan

untuk menambah jumlah dataset dan menjaga keseimbangan distribusi label agar performa model menjadi lebih stabil serta dapat meningkatkan kemampuan klasifikasi pada seluruh kategori prioritas tugas.

DAFTAR PUSTAKA

- [1] Septiana, D. F. S., Triyanto, W. A., & Muzid, S. (2025). Penerapan Metode Decision Tree Algoritma C4.5 Dalam Sistem Rekomendasi Jurusan Bagi Calon Mahasiswa Baru. Vol. 18, ISSN 089-3957.
- [2] Sivi, N. A., Mualim, I., & Lestari, C. A. (2024). Data Mining Menggunakan Decision Tree untuk Prediksi Nilai Akhir Siswa. Vol. 4, No. 3, ISSN 2827-8127.
- [3] Lestari, D., & Lestari, S. (2024). Penerapan Data Mining Klasifikasi Tingkat Pemahaman Siswa Pada Kegiatan Belajar Mengajar dengan Metode Decision Tree (Studi Kasus SDN Malaka Jaya 11 Duren Sawit). Vol. 5, No. 2, ISSN 2776-8074.
- [4] Putra, B. A., Azhari, M.. (2025). Implementasi Algoritma Decision Tree Untuk Klasifikasi Pemilihan Sekolah Lanjutan Di MTs Al-Hasanah Medan. Vol. 5, No. 1.
- [5] Sari, S. N., Annisa, P., Rahma, A. N. D., Ritonga, R. P., & Utomo, D. P. (2025). Teknik Data Mining Dengan Menggunakan Algoritma Decision Tree Untuk Mengetahui Pola Pemahaman Mahasiswa Pada Matakuliah Pemrograman. Vol. 6, No. 4, ISSN 2722-052.
- [6] Junaedy, Musdar, I. A., & Angriani, H. (2017). Implementasi Metode Decision Tree Dalam Menentukan Pemberian Kredit Mobil Menggunakan Visual Basic (Studi Kasus UD PUTRA MAS Makassar). Vol. 4, No. 1, ISSN 2355-3677.
- [7] Ramadhon, R. N., Ogi, A., Agung, A. P., Putra, R., Febrihartina, S. S., & Firdaus, U. (2024). Implementasi Algoritma Decision Tree untuk Klasifikasi Pelanggan Aktif atau Tidak Aktif pada Data Bank. Vol. 3, No. 2, ISSN 2963-590.
- [8] Ramadhani, N. A., Albab, M. S. U., Habibie, M. A., Setiawan, A. V. A. G., Lifaz, M. F., Handayani, N., Fawwaz, F. H., Maurian, F. S., Dharmakusuma, F. C., & Humayra, R. (2025). Analisis Peran Pembelajaran Kolaboratif dalam Mengembangkan Keterampilan Sosial dan Akademik di Fakultas Hukum Universitas Negeri Semarang. Vol. 4, No. 1.
- [9] Anggreani, S. (2024). Pengantar dan Pengenalan Konsep Dasar Algoritma Pemrograman. Vol. 2, No. 1, ISSN 2988-1129.
- [10] Faisal, Dhika, H., & Veris, H. (2021). Penerapan Algoritma Decision Tree Dalam Penjualan Handpone. Vol. 1, No. 4, ISSN 2776-5873.