

SKRIPSI

**IMPLEMENTASI *GRADIENT BOOSTING MACHINE* UNTUK
KLASIFIKASI DATA KEBERHASILAN SISWA PADA SD
MUHAMMADIYAH MEDAN TIMUR BERBASIS WEB**

DISUSUN OLEH

AFDOLLY AKBAR KHAIDIR SIREGAR

2109010112



UMSU
Unggul | Cerdas | Terpercaya

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

MEDAN

2025

**IMPLEMENTASI *GRADIENT BOOSTING MACHINE* UNTUK
KLASIFIKASI DATA KEBERHASILAN SISWA PADA SD
MUHAMMADIYAH MEDAN TIMUR BERBASIS WEB**

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer
(S.Kom) dalam Program Studi Sistem Informasi pada Fakultas Ilmu Komputer dan
Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara**

AFDOLLY AKBAR KHAIDIR SIREGAR

NPM. 2109010112

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**

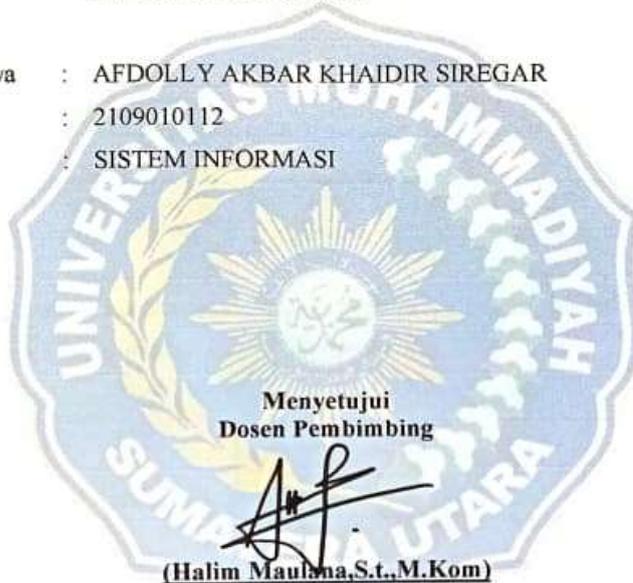
MEDAN

2025

LEMBAR PENGESAHAN

Judul Skripsi : IMPLEMENTASI *GRADIENT BOOSTING MACHINE*
UNTUK KLASIFIKASI DATA KEBERHASILAN
SISWA PADA SD MUHAMMADIYAH MEDAN
TIMUR BERBASIS WEB

Nama Mahasiswa : AFDOLLY AKBAR KHAIDIR SIREGAR
NPM : 2109010112
Program Studi : SISTEM INFORMASI



Menyetujui
Dosen Pembimbing


(Halim Maulana, S.t., M.Kom)
NIDN. 0121119102

Ketua Program Studi



(Dr. Firahmi Rizky, S.Kom., M.Kom.)
NIDN. 0116079201

Dekan




(Dr. Af Khowarizmi, S.Kom., M.Kom.)
NIDN. 0127099201

iii

PERNYATAAN ORISINALITAS

iii

PERNYATAAN ORISINALITAS

**IMPLEMENTASI *GRADIENT BOOSTING MACHINE* UNTUK KLASIFIKASI DATA
KEBERHASILAN SISWA PADA SD MUHAMMADIYAH MEDAN TIMUR
BERBASIS WEB**

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, Agustus 2025

Yang membuat pernyataan



Afdolly Akbar Khaidir Siregar

NPM. 2109010112

**PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama : Afdolly Akbar Khaidir Siregar
NPM : 2109010112
Program Studi : Sistem Informasi
Karya Ilmiah : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bebas Royalti Non-Eksekutif (*Non-Exclusive Royalty free Right*) atas penelitian skripsi saya yang berjudul:

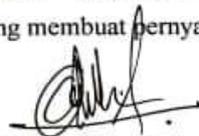
**IMPLEMENTASI *GRADIENT BOOSTING MACHINE* UNTUK KLASIFIKASI DATA
KEBERHASILAN SISWA PADA SD MUHAMMADIYAH MEDAN TIMUR
BERBASIS WEB**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, Agustus 2025

Yang membuat pernyataan



Afdolly Akbar Khaidir Siregar

NPM. 2109010112

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : Afdolly Akbar Khaidir Siregar
Tempat dan Tanggal Lahir : Medan, 03 Februari 2004
Alamat Rumah : Jl. Durian Gg. Pelajar No. 43 Medan
Telepon/Faks/HP : 082320851136
E-mail : afdollyakbar030204@gmail.com
Instansi Tempat Kerja : -
Alamat Kantor : -

DATA PENDIDIKAN

SD : SD 060786 MEDAN TAMAT: 2015
SMP : SMP NEGERI 12 MEDAN TAMAT: 2018
SMA : SMA 18 MEDAN TAMAT: 2021

KATA PENGANTAR



Puji Syukur alhamdulillah, penulis ucapkan terima kasih kepada Allah SWT yang telah memberikan rahmat, hidayah, dan pertolongan-Nya, sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Skripsi yang berjudul "**IMPLEMENTASI GRADIENT BOOSTING MACHINE UNTUK KLASIFIKASI DATA KEBERHASILAN SISWA PADA SD MUHAMMADIYAH MEDAN TIMUR BERBASIS WEB**" ini disusun sebagai salah satu syarat untuk meraih gelar Sarjana Strata 1 (S1) pada Jurusan Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara. Shalawat dan salam selalu terlimpahkan kepada Nabi Muhammad SAW, sebagai teladan terbaik bagi umat manusia, beserta seluruh keluarga, sahabat, dan pengikut beliau.

Penulis menyadari bahwa keberhasilan penyelesaian skripsi ini tidak lepas dari dukungan, bimbingan, dan bantuan dari berbagai pihak. Oleh karena itu, penulis tentunya berterima kasih kepada berbagai pihak dalam dukungan serta doa dalam penyelesaian skripsi. Penulis juga mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Agussani, M.AP., Rektor Universitas Muhammadiyah Sumatera Utara (UMSU)
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara (FIKTI UMSU).
3. Bapak Halim Maulana, S.T., M.Kom Selaku Wakil Dekan I Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI UMSU) dan juga Dosen Pembimbing Skripsi Penulis yang selalu memberikan bimbingan, dukungan, arahan dan motivasi yang luar biasa untuk penulis selama proses pengerjaan skripsi ini.

4. Bapak Lutfi Basit, Sos., M.I.Kom selaku wakil Dekan III Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI UMSU).
5. Ibu Dr. Firaahmi Rizky, S.Kom.,M.Kom selaku Ketua Program Studi Sistem Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara (FIKTI UMSU).
6. Ibu Fatma Sari Hutagalung, S.Kom., M.Kom Selaku Ketua Program Studi Teknologi Informasi Universitas Muhammadiyah Sumatera Utara (FIKTI UMSU).
7. Bapak/Ibu dosen Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara (FIKTI UMSU).
8. Seluruh Pegawai Biro Administrasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara (FIKTI UMSU).
9. Terimakasih Kepada kedua orangtua saya Bapak Hasan Maksom Siregar dan Ibu Dwy Susy Mekar Santi dua orang yang sangat berjasa dalam hidup saya, orang yang selalu menjadi penyemangat saya, yang tidak henti-hentinya memberikan kasih sayang dan juga memberikan motivasi kepada saya dan selalu menjadi orang pertama bagi hidup saya. Terimakasih untuk semuanya berkat do'a ibu dan bapak saya bisa berada dititik ini.
10. Bapak dan ibu pimpinan dan seluruh staff Pegawai yang ada di sekolah Sd Muhammadiyah 28 Medan Timur selaku mitra yang telah memberikan dukungan, kesempatan, serta data yang diperlukan sehingga penelitian ini dapat terlaksana dengan baik.
11. Teman-teman seperjuangan saya M. Kevin Masyaid Siregar, Avner Christian, Aditya Rizky, Mhd Syahputra, Fawwaz Belva, dan Pacar saya sendiri Anisa Ariani yang selalu mensupport saya dan penyemangat untuk saya dan teman-teman informasi Angkatan 2021 yang lainnya.

IMPLEMENTASI *GRADIENT BOOSTING MACHINE* UNTUK KLASIFIKASI DATA KEBERHASILAN SISWA PADA SD MUHAMMADIYAH MEDAN TIMUR BERBASIS WEB

ABSTRAK

Pendidikan dasar merupakan tahap fundamental yang menentukan kualitas pendidikan pada jenjang selanjutnya. Salah satu tantangan yang dihadapi siswa adalah transisi dari Sekolah Dasar (SD) ke Sekolah Menengah Pertama (SMP), yang menuntut kesiapan akademik maupun emosional. SD Muhammadiyah Medan Timur menghadapi kesulitan dalam mempersiapkan siswa menghadapi seleksi masuk SMP Negeri karena belum adanya sistem yang mampu mengklasifikasi keberhasilan siswa secara otomatis. Selama ini, pengelolaan data akademik dan non-akademik masih dilakukan secara manual, sehingga kurang efisien dan rentan terhadap kesalahan. Penelitian ini bertujuan untuk membangun sistem berbasis web yang mengintegrasikan algoritma *Gradient Boosting Machine* (GBM) dalam mengklasifikasi potensi keberhasilan siswa. Data yang digunakan meliputi nilai rapor, tingkat kehadiran, serta prestasi akademik dan non-akademik lainnya. Model GBM dilatih menggunakan data historis siswa untuk menghasilkan prediksi mengenai kemungkinan diterimanya siswa di SMP Negeri. Hasil penelitian menunjukkan bahwa penerapan algoritma GBM mampu memberikan klasifikasi yang akurat terhadap potensi keberhasilan siswa. Dengan sistem ini, pihak sekolah dapat mengidentifikasi siswa yang membutuhkan perhatian lebih sejak dini, sehingga intervensi pembinaan dapat dilakukan secara tepat dan sesuai kebutuhan. Implementasi GBM dalam konteks pendidikan diharapkan dapat meningkatkan efektivitas pengelolaan data akademik sekaligus mendukung pengambilan keputusan yang lebih baik di lingkungan sekolah dasar.

Kata kunci: Pendidikan dasar, klasifikasi, *Gradient Boosting Machine* (GBM), sistem berbasis web, keberhasilan siswa.

IMPLEMENTATION OF GRADIENT BOOSTING MACHINE FOR CLASSIFYING STUDENT SUCCESS DATA AT SD MUHAMMADIYAH MEDAN TIMUR BASED ON A WEB-BASED SYSTEM

ABSTRACT

Primary education is a fundamental stage that determines the quality of learning at higher levels. One of the challenges faced by students is the transition from elementary school (SD) to junior high school (SMP), which requires both academic and emotional readiness. SD Muhammadiyah Medan Timur encounters difficulties in preparing students for admission to public junior high schools due to the absence of an automated system capable of classifying student success potential. Currently, academic and non-academic data are still processed manually, which is inefficient and prone to errors. This study aims to develop a web-based system that integrates the Gradient Boosting Machine (GBM) algorithm to classify students' success potential. The data used include report card scores, attendance rates, as well as academic and non-academic achievements. The GBM model is trained using students' historical data to generate predictions regarding the likelihood of being admitted to public junior high schools. The results show that the implementation of GBM provides accurate classification of students' success potential. Through this system, schools can identify students who require more attention at an early stage, allowing appropriate and targeted interventions. The application of GBM in the educational context is expected to enhance the effectiveness of academic data management and support better decision-making in primary schools.

Keywords: *Primary education, classification, Gradient Boosting Machine (GBM), web-based system, student success.*

DAFTAR ISI

HALAMAN PENGAJUAN.....	ii
LEMBAR PENGESAHAN	iii
PERNYATAAN ORISINALITAS	iv
PERNYATAAN PERSETUJUAN PUBLIKASI	v
RIWAYAT HIDUP	vi
KATA PENGANTAR	vii
ABSTRAK.....	ix
<i>ABSTRACT</i>	x
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	5
BAB II LANDASAN TEORI	6
2.1 Penelitian Terdahulu.....	6
2.2 Pendidikan Dasar dan Transisi Ke Sekolah Menengah.....	7
2.3 Data Mining.....	8
2.4 Algoritma Gradient Boosting Machine	9
2.4.1. Inisialisasi Model Awal.....	9
2.4.2. Membuat <i>Weak Model</i> (Pohon Keputusan Dangkal)	9
2.4.3. Menghitung Kesalahan Residual dan Gradien	10
2.4.4. Menambahkan Model Baru untuk Mengurangi Kesalahan.....	10
2.4.5. Menggabungkan semua model untuk Membentuk Model Akhir.....	11
2.5 Evaluasi Model.....	11
2.6 Teknologi yang Digunakan	13
2.6.1. <i>Website</i>	13
2.6.2. HTML (<i>HyperText Markup Language</i>).....	14
2.6.3. PHP (<i>Hypertext Preprocessor</i>).....	15

2.6.4.	<i>Tailwind CSS</i>	15
2.6.5.	<i>Python</i>	16
2.6.6.	<i>Visual Studio Code</i>	16
2.6.7.	Flask API	16
2.6.8.	XAMPP	17
2.7	Flowchart	17
BAB III METODE PENELITIAN		20
3.1	Alur Penelitian (<i>Flowchart</i>)	20
3.2	Pengumpulan Data	25
3.3	Tahapan Pengolahan Data	26
3.3.1.	Pra-Pemrosesan Data (<i>Preprocessing</i>)	27
3.3.2.	Seleksi Atribut (<i>Feature Selection</i>)	28
3.4	Pemilihan dan Penerapan Model <i>Gradient Boosting Machine</i> (GBM)	28
3.4.1.	Pemisahan Data (<i>Splitting Data</i>)	29
3.4.2.	Implementasi Algoritma GBM	29
3.4.3.	Pengujian Model	29
3.4.4.	Evaluasi Model GBM	30
3.4.5.	Penyimpanan Model	30
3.5	Simulasi Perhitungan <i>Gradient Boosting Machine</i>	30
3.6	Arsitektur Sistem	36
3.7	Desain Sistem	38
3.8	Rancangan Basis Data	54
3.9	Rancangan Antarmuka (<i>Interface</i>)	55
BAB IV HASIL DAN PEMBAHASAN		61
4.1.	Perhitungan Manual <i>Gradient Boosting Machine</i>	61
4.2.	Hasil Tampilan Aplikasi Berbasis Web	66
4.3.	Pengujian Program	79
4.4.	Kelebihan dan Kelemahan Program	81
BAB V KESIMPULAN DAN SARAN		83
5.1	Kesimpulan	83
5.2	Saran	84
DAFTAR PUSTAKA		86
LAMPIRAN-LAMPIRAN		88

DAFTAR GAMBAR

Gambar 3. 1 Flowchart Penelitian	24
Gambar 3. 2 Use Case Diagram	39
Gambar 3. 3 Activity Diagram Login	40
Gambar 3. 4 Activity Diagram Kelola Data Siswa	41
Gambar 3. 5 Activity Diagram Kelola Dataset	42
Gambar 3. 6 Activity Diagram Lakukan Klasifikasi	43
Gambar 3. 7 Activity Diagram Lihat Hasil Klasifikasi	44
Gambar 3. 8 Activity Diagram Latih dan Simpan Model	45
Gambar 3. 9 Activity Diagram Unduh Laporan.....	46
Gambar 3. 10 Activity Diagram Logout.....	47
Gambar 3. 11 Sequence Diagram Login	48
Gambar 3. 12 Sequence Diagram Kelola Data Siswa	49
Gambar 3. 13 Sequence Diagram Lihat Dataset	50
Gambar 3. 14 Sequence Diagram Lakukan Klasifikasi.....	50
Gambar 3. 15 Sequence Diagram Lihat Hasil Klasifikasi.....	51
Gambar 3. 16 Sequence Diagram Latih Model.....	51
Gambar 3. 17 Sequence Diagram Unduh Laporan.....	52
Gambar 3. 18 Sequence Diagram Logout	52
Gambar 3. 19 Class Diagram.....	53
Gambar 3. 20 Desain Form Login.....	56
Gambar 3. 21 Desain Form Dashboard	56
Gambar 3. 22 Desain Form Kelola Data Siswa.....	57
Gambar 3. 23 Desain Form Tambah Data Siswa	57

Gambar 3. 24 Desain Form Lihat Dataset	58
Gambar 3. 25 Desain Form Lakukan Klasifikasi	58
Gambar 3. 26 Desain Form Lihat Hasil Klasifikasi	58
Gambar 3. 27 Desain Form Latih Model	59
Gambar 3. 28 Desain Form Laporan	59
Gambar 3. 29 Desain Form Konfirmasi Logout.....	60
Gambar 4. 1 Tampilan Halaman Login.....	67
Gambar 4. 2 Tampilan Halaman Dashboard	68
Gambar 4. 3 Tampilan Halaman Kelola Data Siswa.....	68
Gambar 4. 4 Tampilan Halaman Tambah Data Siswa	69
Gambar 4. 5 Tampilan Kelola Data Mapel.....	70
Gambar 4. 6 Tampilan Tambah Data Mapel	70
Gambar 4. 7 Tampilan Kelola Data Kelas	71
Gambar 4. 8 Tampilan Halaman Tambah Data Kelas.....	71
Gambar 4. 9 Tampilan Halaman Kategori Prestasi	72
Gambar 4. 10 Tampilan Halaman Tambah Kategori Prestasi	73
Gambar 4. 11 Penjelasan Algoritma.....	74
Gambar 4. 12 Tahap EDA dalam Latih dan Uji Model	74
Gambar 4. 13 Tampilan Pembagian Data.....	75
Gambar 4. 14 Tampilan Proses Training.....	76
Gambar 4. 15 Tampilan Evaluasi Model.....	76
Gambar 4. 16 Tampilan Detail Prediksi	77
Gambar 4. 17 Tampilan Halaman Kelola Model	78
Gambar 4. 18 Tampilan Halaman Kelola Klasifikasi.....	78

DAFTAR TABEL

Tabel 2. 1 Tabel Penelitian Terdahulu	6
Tabel 2. 2 Simbol Flowchart	18
Tabel 3. 1 Tabel Dataset yang digunakan	31
Tabel 3. 2 Nilai Residual	32
Tabel 3. 3 Tabel Perhitungan Gradien.....	34
Tabel 3. 4 Tabel Klasifikasi Baru pada Iterasi Pertama	34
Tabel 3. 5 Tabel Perhitungan Residual pada iterasi kedua.....	35
Tabel 3. 6 Tabel Perhitungan Gradien Pada Iterasi kedua	35
Tabel 3. 7 Komponen Yang digunakan.....	37
Tabel 3. 8 Rancangan Tabel Users	54
Tabel 3. 9 Rancangan Tabel Siswa	54
Tabel 3. 10 Rancangan Tabel Dataset	54
Tabel 3. 11 Rancangan Tabel Model.....	55
Tabel 3. 12 Rancangan Tabel Hasil Klasifikasi	55
Tabel 4. 1 Subset Data Untuk Perhitungan Manual	62
Tabel 4. 2 Tabel Perhitungan Residual	64
Tabel 4. 3 Nilai Output Terminal Node Iterasi Pertama	65
Tabel 4. 4 Hasil Pembaruan Prediksi Iterasi Pertama	66
Tabel 4. 5 Tabel Pengujian Program	79

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pendidikan dasar merupakan tahap fundamental dalam sistem pendidikan yang membentuk dasar pengetahuan, keterampilan, dan karakter siswa. Keberhasilan pada jenjang ini sangat menentukan kualitas pendidikan pada tingkat selanjutnya. Pendidikan di sekolah dasar memiliki kontribusi penting dalam membangun pondasi kognitif dan afektif siswa, serta mempersiapkan mereka menghadapi jenjang pendidikan menengah yang lebih kompleks (Susilahati et al., 2023). Namun, salah satu fase penting yang sering dihadapi siswa adalah transisi dari Sekolah Dasar (SD) ke Sekolah Menengah Pertama (SMP), yang penuh tantangan baik dari segi akademik maupun emosional. Kesiapan siswa dalam menghadapi jenjang pendidikan yang lebih tinggi sangat bergantung pada pembinaan yang efektif selama masa SD (Susilahati et al., 2023).

SD Muhammadiyah Medan Timur menghadapi tantangan dalam mempersiapkan siswa kelas akhir untuk menghadapi seleksi masuk SMP Negeri. Tanpa adanya sistem yang tepat untuk memklasifikasi keberhasilan siswa berdasarkan data akademik dan non-akademik, pihak sekolah kesulitan memberikan intervensi yang sesuai kepada siswa yang memerlukan perhatian lebih. Hal ini dapat memengaruhi kesempatan siswa untuk diterima di SMP Negeri, karena seleksi yang semakin kompetitif. Data yang ada saat ini tidak sepenuhnya dimanfaatkan untuk menganalisis potensi keberhasilan siswa dalam seleksi tersebut (Wicaksono et al., 2023).

Masalah ini terjadi karena tidak adanya sistem berbasis teknologi yang dapat secara otomatis menganalisis dan memklasifikasi kemungkinan keberhasilan siswa dalam seleksi SMP Negeri. Pengelolaan data akademik dan non-akademik yang masih dilakukan secara manual membatasi kemampuan sekolah untuk melakukan evaluasi yang cepat dan akurat. Hal ini menjadi kendala karena data yang tersedia harus diolah terlebih dahulu secara manual, yang tentunya memakan waktu dan tenaga, serta rentan terhadap kesalahan (Harahap & Sulindawaty, 2020).

Untuk mengatasi masalah dalam memklasifikasi keberhasilan siswa dan mempermudah evaluasi akademik di SD Muhammadiyah Medan Timur, dibutuhkan sistem yang dapat menganalisis data akademik dan non-akademik secara efisien. Oleh karena itu, solusi yang dipilih adalah pembangunan sistem dengan implementasi *Gradient Boosting Machine* (GBM) untuk klasifikasi data keberhasilan siswa. Dengan menggunakan GBM, sistem ini akan memproses data historis siswa, seperti nilai rapor, kehadiran, dan prestasi ekstrakurikuler, untuk memklasifikasi kemungkinan diterimanya siswa di SMP Negeri. Dengan demikian, pihak sekolah dapat mengidentifikasi siswa yang memerlukan perhatian lebih awal dan memberikan intervensi yang sesuai berdasarkan hasil klasifikasi yang diperoleh. Penggunaan algoritma *machine learning* dalam konteks pendidikan seperti ini telah terbukti efektif, terutama dalam meningkatkan akurasi dan efisiensi pengolahan data (Harahap & Sulindawaty, 2020).

Gradient Boosting Machine dipilih karena kemampuannya dalam menangani data yang kompleks serta menghasilkan klasifikasi yang akurat. Berbagai penelitian sebelumnya menunjukkan bahwa GBM memiliki keunggulan dalam memberikan hasil yang lebih optimal dibandingkan algoritma klasifikasi

lainnya, Penelitian oleh Elsa Suryana & Warsito (2021) menunjukkan bahwa GBM efektif dalam klasifikasi keberhasilan telemarketing, dengan akurasi mencapai 90,39%, yang menunjukkan kemampuan algoritma ini untuk menangani data besar dengan akurasi tinggi. Selain itu, dalam penelitian oleh (Mahdiyah, 2023), BM terbukti menghasilkan akurasi yang baik dalam klasifikasi kualitas cabai berdasarkan citra, dengan nilai precision, recall, dan F-Score sekitar 69,7%, yang menandakan efektivitasnya dalam pengolahan data yang variatif dan kompleks.

Solusi ini akan diterapkan dengan membangun aplikasi berbasis web yang mengintegrasikan *Gradient Boosting Machine* untuk menganalisis dan mengklasifikasikan data siswa. Aplikasi ini akan mengolah data yang terdiri dari nilai rapor, tingkat kehadiran, serta prestasi akademik dan non-akademik lainnya. Data historis yang ada akan digunakan untuk melatih model GBM, dan setelah model selesai dilatih, aplikasi berbasis web akan memberikan klasifikasi secara otomatis mengenai apakah siswa berpotensi diterima di SMP Negeri atau tidak.

Dengan diterapkannya sistem berbasis web yang menggunakan *Gradient Boosting Machine*, diharapkan pihak sekolah dapat memperoleh klasifikasi yang akurat mengenai keberhasilan siswa dalam seleksi SMP Negeri. Hal ini akan memungkinkan pihak sekolah untuk mengidentifikasi siswa yang perlu mendapatkan perhatian khusus lebih awal, sehingga dapat diberikan pembinaan yang tepat dan sesuai dengan kebutuhan.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini adalah sebagai berikut :

1. Bagaimana cara mengimplementasikan algoritma *Gradient Boosting Machine* (GBM) untuk mengklasifikasikan data keberhasilan siswa diterima di SMP Negeri?
2. Faktor-faktor apa saja yang berpengaruh terhadap keberhasilan siswa SD Muhammadiyah Medan Timur dalam seleksi masuk SMP Negeri?
3. Bagaimana membangun sistem klasifikasi keberhasilan siswa berbasis web yang dapat digunakan di SD Muhammadiyah Medan Timur untuk membantu pengambilan keputusan dalam seleksi SMP Negeri?

1.3 Batasan Masalah

Agar penelitian lebih terfokus, maka diterapkan batasan-batasan masalah seperti berikut ini :

1. Penelitian ini hanya dilakukan pada siswa kelas VI di SD Muhammadiyah Medan Timur.
2. Data yang digunakan meliputi nilai rapor, kehadiran, dan prestasi non akademik yang tersedia dari pihak sekolah untuk kelas 6 yang diambil dari semester 1 – 5.
3. Sistem yang dibangun adalah berbasis web, dengan integrasi model GBM yang dilatih menggunakan Python dan pengembangan sistem berbasis web menggunakan PHP dan MySQL untuk penyimpanan data.

1.4 Tujuan Penelitian

Adapun yang menjadi tujuan dalam penelitian ini adalah sebagai berikut :

1. Mengimplementasikan algoritma *Gradient Boosting Machine* untuk mengklasifikasikan data keberhasilan siswa dalam diterima di SMP Negeri.

2. Mengidentifikasi variabel-variabel penting yang berpengaruh terhadap keberhasilan siswa melalui proses klasifikasi data.
3. Merancang dan membangun sistem klasifikasi keberhasilan siswa berbasis web untuk mendukung proses pengambilan keputusan di SD Muhammadiyah Medan Timur.

1.5 Manfaat Penelitian

Adapun yang menjadi manfaat dalam penelitian ini adalah sebagai berikut

1. Meningkatkan akurasi klasifikasi keberhasilan siswa dengan menggunakan algoritma *Gradient Boosting Machine* (GBM) untuk memklasifikasi kemungkinan diterimanya siswa di SMP Negeri.
2. Membantu pihak sekolah dalam pengambilan keputusan yang lebih efektif dengan memberikan klasifikasi berbasis data yang objektif.
3. Meningkatkan pemanfaatan teknologi dalam Pendidikan dengan mengembangkan sistem klasifikasi berbasis web yang dapat digunakan untuk memantau perkembangan akademik siswa secara efisien.

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu berfungsi sebagai acuan dan pembandingan dalam penelitian ini. Berbagai studi sebelumnya telah menunjukkan bahwa algoritma *Gradient Boosting Machine* (GBM) memiliki performa yang baik dalam melakukan klasifikasi data dalam konteks pendidikan dan klasifikasi akademik siswa. Berikut adalah beberapa penelitian terdahulu yang relevan.

Tabel 2. 1 Tabel Penelitian Terdahulu

No	Peneliti & Tahun	Judul Penelitian	Objek Penelitian	Hasil Utama
1	(Atlantic et al., 2024)	Gradient Boosting Machine Pada Klasifikasi Kelulusan Mahasiswa	Data kelulusan mahasiswa Prodi Statistika UNTAN	GBM memperoleh akurasi 71,09%, lebih baik dibandingkan dengan model CART yang hanya mencapai 67,97%
2	(Wardhana et al., 2022)	Gradient Boosting Machine, Random Forest dan Light GBM untuk Klasifikasi Kacang Kering	<i>Dataset</i> kacang kering	GBM menghasilkan akurasi pelatihan 99%, dengan akurasi validasi 91%, lebih baik dari Random Forest
3	(Mahdiyah, 2023)	Klasifikasi Kualitas Cabai dengan Menggunakan Algoritma Gradient Boosting	<i>Dataset</i> kualitas cabai	GBM mencapai akurasi sekitar 76%, dengan precision 69.7% dan recall 69.1%
4	Suryana, S. E. (2021)	Penerapan Gradient Boosting dengan Hyperopt untuk Memklasifikasi Keberhasilan	Data telemarketing bank	GBM dengan Hyperopt mencapai akurasi 90,39%, precision 94,91%, dan AUC 0,939

		Telemarketing Bank		
5	(Gede et al., 2025)	Penerapan Machine Learning Extreme Gradient Boosting Dalam Klasifikasi Potensi Tsunami Berdasarkan Data Gempa	Data gempa bumi	GBM menghasilkan akurasi 85%, recall 0.79, precision 0.71, dan F1-score 0.75 setelah menggunakan SMOTE
6	(Tri Pratiwi et al., 2024)	Systematic Literature Review Penerapan Gradient Boosting Untuk Klasifikasi Penyakit Diabetes Tipe 2	<i>Dataset</i> medis diabetes	GBM mencapai akurasi hingga 82%, lebih baik dibandingkan dengan Random Forest dan Naïve Bayes
7	(Mahmoud & Sayed, 2024)	Enhancing Academic Performance Prediction for At-Risk Students Using Machine Learning Algorithms	Data akademik siswa	GBM memberikan klasifikasi akurat dengan R^2 0,9494, serta RMSE 0,0516, MSE 0,0027, MAE 0,0255

2.2 Pendidikan Dasar dan Transisi Ke Sekolah Menengah

Pendidikan dasar merupakan jenjang pendidikan yang melandasi jenjang pendidikan menengah. Berdasarkan Undang-Undang Republik Indonesia Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional, pendidikan dasar terdiri atas Sekolah Dasar (SD) dan Madrasah Ibtidaiyah (MI) atau bentuk lain yang sederajat, serta Sekolah Menengah Pertama (SMP) dan Madrasah Tsanawiyah (MTs) atau bentuk lain yang sederajat. Jenjang ini berfungsi memberikan pengetahuan dan keterampilan dasar kepada peserta didik, menumbuhkan sikap dasar yang diperlukan dalam masyarakat, serta mempersiapkan mereka untuk mengikuti pendidikan menengah.

Jenjang SD memiliki peran krusial dalam sistem pendidikan nasional karena menjadi dasar bagi pengembangan pengetahuan dan keterampilan siswa. Pendidikan di SD berkontribusi dalam membangun dasar pengetahuan siswa yang akan digunakan pada jenjang pendidikan selanjutnya. Oleh karena itu, pelaksanaan

pembelajaran di SD harus berjalan optimal untuk memastikan fondasi yang kuat bagi peserta didik (Andri Aka, 2022).

Transisi dari SD ke SMP merupakan fase penting dalam perjalanan pendidikan siswa yang sering kali diwarnai oleh berbagai tantangan. Beberapa tantangan utama dalam transisi ini meliputi:

1. Zonasi: Kebijakan zonasi dalam penerimaan siswa baru di SMP Negeri bertujuan untuk pemerataan akses pendidikan. Namun, implementasi kebijakan ini dapat menimbulkan tantangan bagi siswa yang berada di luar zona tertentu, sehingga mempengaruhi kesempatan mereka untuk diterima di SMP Negeri yang diinginkan.
2. Seleksi: Beberapa SMP Negeri menerapkan seleksi berdasarkan nilai akademik atau prestasi lainnya. Hal ini menuntut siswa SD untuk memiliki pencapaian akademik yang tinggi agar dapat bersaing dalam proses seleksi tersebut.
3. Kompetisi: Persaingan yang ketat untuk masuk ke SMP Negeri favorit dapat menimbulkan tekanan bagi siswa SD. Mereka dituntut untuk tidak hanya unggul dalam akademik, tetapi juga dalam bidang ekstrakurikuler dan prestasi lainnya.

2.3 Data Mining

Data Mining merupakan bentuk penggalian data yang digunakan untuk menggali pengetahuan dari jumlah data yang besar (Putra et al., 2023). *Data Mining* diperlukan dalam melakukan klasifikasi untuk hubungan ditemukan yang memiliki arti, pola, dan kecenderungan dengan diperiksa sekumpulan besar data yang disimpan dalam penyimpanan dengan menggunakan teknik pengenalan pola statistika ataupun matematis (Iriane & Nurfaizah, 2023). Dalam *Data Mining*,

proses pencarian pola atau informasi yang berguna dari data yang telah dipilih atau diolah dinamakan sebagai *Knowledge Data Discovery* (KDD).

2.4 Algoritma *Gradient Boosting Machine*

Gradient Boosting Machine (GBM) adalah algoritma *machine learning* berbasis *ensemble* yang menggabungkan beberapa model *weak learners* (pohon keputusan sederhana) untuk membentuk model yang lebih kuat. Model-model ini dibangun secara bertahap, di mana setiap model baru berfungsi untuk memperbaiki kesalahan dari model sebelumnya, melalui proses yang dikenal sebagai *boosting* (Elsa Suryana & Warsito, 2021).

Adapun proses pembelajaran dan prinsip kerja *Gradient Boosting Machine* (GBM) melalui beberapa tahap seperti dibawah ini

2.4.1. Inisialisasi Model Awal

Pada tahap pertama, model awal dibentuk dengan menggunakan rata-rata dari target variabel sebagai klasifikasi pertama. Model ini adalah klasifikasi awal yang digunakan untuk menghitung kesalahan residual dari langkah selanjutnya.

Dengan rumus :

$$\hat{y}_o = \frac{1}{N} \sum_{i=1}^N y_i \dots\dots\dots(2.1)$$

Dimana \hat{y}_o adalah klasifikasi awal (rata-rata dari target variabel), y_i adalah nilai target pada data ke-I, dan N adalah jumlah total data

2.4.2. Membuat *Weak Model* (Pohon Keputusan Dangkal)

Setelah model awal dibentuk, model selanjutnya adalah pohon keputusan yang sangat dangkal (biasanya hanya memiliki satu atau dua cabang). Pohon ini dibangun untuk memklasifikasi kesalahan residual yang dibuat oleh model

sebelumnya. Kesalahan residual dihitung sebagai nilai selisih antara nilai target yang sebenarnya dengan klasifikasi model sebelumnya.

$$r_i = y_i - \hat{y}_{i-1} \dots \dots \dots (2.2)$$

Dimana, r_i adalah residual untuk data ke-i, y_i adalah nilai target yang sesungguhnya, \hat{y}_{i-1} adalah klasifikasi dari model sebelumnya.

2.4.3. Menghitung Kesalahan Residual dan Gradien

Pada tahap ini, *residual error* dihitung dari perbedaan antara nilai yang diklasifikasi oleh model dan nilai yang sesungguhnya. GBM kemudian menghitung gradien dari fungsi kerugian untuk memperbaiki kesalahan. Gradien dihitung sebagai turunan dari fungsi kerugian yang digunakan, yang sering kali adalah **log loss** pada masalah klasifikasi:

$$g_i = \frac{\partial L}{\partial f(x_i)} \dots \dots \dots (2.3)$$

Dimana g_i adalah gradien dari fungsi kerugian L pada titik data ke-i dan $f(x_i)$ adalah *output* dari model pada data ke-i

2.4.4. Menambahkan Model Baru untuk Mengurangi Kesalahan

Model baru ditambahkan untuk memperbaiki kesalahan dari model sebelumnya. Ini dilakukan dengan penyesuaian bobot model baru untuk memastikan kesalahan dari model sebelumnya dikoreksi. Setiap pohon keputusan baru akan berusaha untuk mengurangi *residual error* atau kesalahan yang belum terklasifikasi. Setiap klasifikasi diperbarui berdasarkan hasil dari model baru yang dibentuk, dengan memperhitungkan *learning rate* yang mengontrol kontribusi model baru pada pembaruan.

$$\hat{y}_i^{baru} = \hat{y}_i^{lama} + \eta x f_i \dots \dots \dots (2.4)$$

Dimana \hat{y}_i^{baru} adalah klasifikasi baru, \hat{y}_i^{lama} adalah klasifikasi lama, η adalah *learning rate*, f_i adalah klasifikasi dari pohon keputusan baru.

2.4.5. Menggabungkan semua model untuk Membentuk Model Akhir

Setelah model-model baru dibuat, semua model digabungkan dalam satu ensemble untuk membentuk model yang lebih kuat. Proses ini memastikan bahwa kesalahan dari model sebelumnya diperbaiki dengan akurasi yang lebih tinggi pada model akhir. Model akhir adalah hasil dari penjumlahan semua model yang ada dalam *ensemble*.

$$\hat{y}_{final} = \sum_{k=1}^K \hat{y}_k \dots \dots \dots (2.5)$$

2.5 Evaluasi Model

Dalam evaluasi model *Gradient Boosting Machine (GBM)*, beberapa metrik digunakan untuk mengukur kinerja model dalam klasifikasi. Metrik evaluasi ini memberikan wawasan tentang seberapa baik model dalam memklasifikasi kelas-kelas tertentu, serta sejauh mana model dapat menangani kesalahan klasifikasi. Metrik utama yang digunakan untuk mengevaluasi model klasifikasi seperti **GBM** antara lain adalah akurasi, precision, recall, F1-score, dan *confusion matrix*. Evaluasi ini tidak hanya mengukur kemampuan model untuk menghasilkan klasifikasi yang benar, tetapi juga memberikan gambaran tentang performa model dalam menangani data yang seimbang atau tidak seimbang.

2.5.1. Akurasi

Akurasi mengukur seberapa sering model menghasilkan klasifikasi yang benar dibandingkan dengan jumlah total klasifikasi yang dilakukan. Metrik ini sangat berguna dalam kasus di mana distribusi kelas relatif seimbang. Dalam GBM, akurasi dapat diukur untuk setiap model setelah proses pelatihan.

$$\text{Akurasi} = \frac{\text{Jumlah Klasifikasi Benar}}{\text{Total Klasifikasi}} \dots\dots\dots (2.6)$$

2.5.2. *Confusion Matrix*

Confusion matrix adalah matriks yang digunakan untuk menggambarkan performa model klasifikasi. Matriks ini menunjukkan *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN) yang digunakan untuk menghitung metrik lain seperti *precision*, *recall*, dan *F1-score*.

- a. *True Positive* (TP): Jumlah data positif yang diklasifikasi dengan benar.
- b. *False Positive* (FP): Jumlah data negatif yang diklasifikasi sebagai positif.
- c. *True Negative* (TN): Jumlah data negatif yang diklasifikasi dengan benar.
- d. *False Negative* (FN): Jumlah data positif yang diklasifikasi sebagai negatif.

2.5.3. *Precision, Recall, dan F1-Score*

Precision adalah perbandingan antara *True Positive* (TP) dengan banyaknya data yang diklasifikasi positif, secara matematis dapat dilihat dibawah ini

$$\text{Precision} = \frac{TP}{TP+FP} \dots\dots\dots (2.7)$$

Untuk *recall* sendiri adalah perbandingan antara *true positive* (TP) dengan banyaknya data yang sebenarnya positif. Dapat dinyatakan secara matematis seperti dibawah ini

$$\text{Recall} = \frac{TP}{TP+FN} \dots\dots\dots (2.8)$$

Sedangkan *F1 Score* adalah nilai tengah dari *precision* dan *recall*. Nilai terbaik dari *F1 Score* adalah 1 dan nilai terburuknya adalah 0, secara matematis dapat dituliskan seperti dibawah ini

$$\frac{1}{F1} = \frac{1}{2} \left(\frac{1}{\text{precision}} + \frac{1}{\text{recall}} \right) \dots\dots\dots (2.9)$$

Nilai F1 Score yang baik menandakan bahwa model klasifikasi baik

2.5.4. Evaluasi Model GBM

Evaluasi model GBM melibatkan pengukuran akurasi dan perhitungan *confusion matrix* untuk memahami kemampuan model dalam mengklasifikasikan data. Dalam konteks model GBM, akurasi saja tidak cukup untuk memberikan gambaran lengkap tentang kinerja model, terutama ketika ada ketidakseimbangan kelas. Oleh karena itu, penting untuk menggunakan *precision*, *recall*, dan *F1-score* bersama dengan *confusion matrix* untuk mendapatkan gambaran yang lebih jelas tentang keefektifan model.

2.6 Teknologi yang Digunakan

Dalam pengembangan sistem pada penelitian ini, berbagai teknologi digunakan untuk membangun dan menjalankan aplikasi *web* secara efektif. Berikut adalah penjelasan mengenai masing masing komponen.

2.6.1. Website

Website merupakan kumpulan halaman web yang terdapat dalam domain atau subdomain di *World Wide Web* Internet. Konten yang tersedia pada situs web merupakan alasan seseorang untuk mengunjunginya. *Website* adalah kumpulan halaman web yang saling terhubung dan seluruh file saling terkait. Web terdiri dari page atau halaman dan kumpulan halaman yang dinamakan *homepage*. *Homepage* berada pada posisi teratas dengan halaman-halaman terkait berada di bawahnya. Biasanya, setiap halaman di bawah *homepage* (child page) berisi *hyperlink* ke halaman lain dalam web (Artaye et al., 2022).

Berdasarkan pengertian tersebut, *website* dapat dibagi menjadi dua bagian. Dengan kata lain, web bersifat statis dan dinamis. *Website* statis jika konten informasinya tetap dan disediakan hanya oleh pemilik *website*, tetapi *website*

dinamis jika konten informasi terus berubah dan dapat diubah oleh pemilik *website* atau pengguna. Contoh Web Statis: Situs Web Profil Perusahaan, Facebook, Twitter, dll. Contoh Web Dinamis (Artaye et al., 2022).

2.6.2. HTML (*HyperText Markup Language*)

HTML, atau *HyperText Markup Language*, adalah bahasa markup standar yang digunakan untuk membuat struktur halaman web. HTML memberikan struktur dasar pada setiap halaman web dengan menggunakan berbagai elemen seperti teks, gambar, tautan, dan form. HTML mendefinisikan elemen-elemen tersebut dalam bentuk tag, yang biasanya dibuka dan ditutup dengan tanda kurung sudut. Setiap tag memiliki fungsi khusus, seperti `<h1>` untuk heading, `<p>` untuk paragraf, dan `<a>` untuk hyperlink. Versi terbaru dari HTML, yaitu HTML5, menambahkan banyak elemen semantik yang mendukung pengelolaan konten multimedia, formulir interaktif, serta pengoptimalan untuk perangkat mobile.

HTML memiliki peran yang sangat penting dalam pengembangan web modern, karena ia memberikan fondasi untuk penataan konten dan navigasi situs web. Keberhasilan penggunaan HTML yang baik dapat meningkatkan pengalaman pengguna dan aksesibilitas web. Sebagai contoh, HTML5 memperkenalkan elemen-elemen baru seperti `<article>`, `<section>`, dan `<footer>` yang membuat struktur halaman web lebih semantik dan mudah diinterpretasikan oleh mesin pencari serta alat bantu seperti pembaca layar. Ini berkontribusi pada peningkatan SEO (Search Engine Optimization) dan memudahkan pengelolaan konten dinamis (Yusuf, 2022).

2.6.3. PHP (*Hypertext Preprocessor*)

PHP adalah bahasa pemrograman server-side yang digunakan untuk mengembangkan aplikasi web dinamis. PHP memungkinkan pengembang untuk menulis kode yang berinteraksi dengan *database*, mengelola sesi pengguna, dan menghasilkan konten web secara dinamis. PHP sering digunakan bersama HTML untuk membangun situs web interaktif dan dapat bekerja dengan berbagai jenis *database* seperti MySQL dan PostgreSQL. Kemampuannya dalam menghasilkan halaman yang dinamis membuat PHP sangat populer dalam pengembangan situs web dan aplikasi berbasis web (Tofir et al., 2020).

2.6.4. *Tailwind CSS*

Tailwind CSS adalah *Framework* CSS berbasis utilitas yang memungkinkan pengembang untuk membangun desain web secara cepat dan fleksibel tanpa menulis banyak CSS kustom. *Framework* ini mengutamakan penggunaan kelas-kelas utilitas yang langsung diterapkan pada elemen HTML untuk mengatur berbagai properti gaya seperti margin, padding, warna, tipografi, dan lain-lain. Salah satu keunggulan Tailwind adalah kemampuannya untuk mempermudah pembuatan desain responsif dengan menggunakan media query berbasis kelas, tanpa perlu menulis CSS tambahan. Tailwind sangat berguna untuk pengembang yang ingin memiliki kontrol penuh atas desain web tanpa mengorbankan keterbacaan atau kinerja (Franz & Rachmadani, 2025).

2.6.5. Python

Python adalah bahasa pemrograman tingkat tinggi yang dikenal karena sintaksisnya yang sederhana dan kemampuannya dalam berbagai bidang, termasuk pengembangan web, analisis data, pembelajaran mesin, dan otomatisasi. *Python* sangat populer di kalangan pengembang karena kemudahan penggunaannya dan komunitas yang besar, yang menyediakan berbagai pustaka dan modul siap pakai. *Python* mendukung berbagai paradigma pemrograman seperti pemrograman prosedural, berorientasi objek, dan fungsional (Galahartlambang et al., 2022).

2.6.6. Visual Studio Code

Visual Studio Code, atau sering disebut *Vs Code*, adalah editor kode sumber lintas platform yang dikembangkan Microsoft, bersifat ringan, mendukung berbagai sistem operasi (Windows, macOS, Linux) dan bisa diperluas fungsinya melalui plugin atau ekstensi. *VS Code* mendukung banyak bahasa pemrograman melalui ekstensi, serta aspek aksesibilitas seperti dukungan terminal buffer non-visual, cues audio, dan menu bantuan yang dapat disesuaikan untuk memperluas kegunaan bagi pengguna. (Sufyan bin uzayr et al., 2022).

2.6.7. Flask API

Flask adalah *Framework* mikro untuk *Python* yang digunakan untuk membangun aplikasi web dan *API* (Application Programming Interface). Dirancang untuk menjadi ringan dan fleksibel, *Flask* tidak menyertakan banyak fitur bawaannya, memungkinkan pengembang untuk memilih komponen yang

dibutuhkan sesuai dengan spesifikasi aplikasi. Dengan *Flask*, pengembang dapat membangun aplikasi web sederhana hingga kompleks dengan cepat, menggunakan berbagai ekstensi untuk mendukung fungsionalitas tambahan seperti pengelolaan basis data dan autentikasi pengguna. Salah satu penggunaan utama *Flask* adalah membangun *RESTful API*, yang memungkinkan aplikasi untuk berkomunikasi dengan sistem lain melalui HTTP (Arugia et al., 2022).

2.6.8. XAMPP

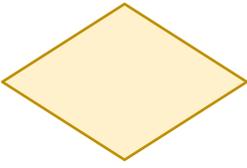
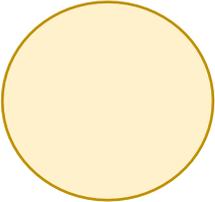
XAMPP adalah paket perangkat lunak yang menggabungkan Apache, MySQL, PHP, dan Perl dalam satu instalasi yang mudah digunakan. Dengan *XAMPP*, pengembang dapat dengan cepat membuat server lokal untuk mengembangkan dan menguji aplikasi web sebelum diterapkan di server produksi. Paket ini memudahkan pengguna dalam memulai pengembangan web tanpa perlu mengkonfigurasi server secara manual. *XAMPP* sering digunakan dalam pengembangan aplikasi web berbasis *PHP* yang memerlukan basis data *MySQL*. Keunggulannya adalah kemudahan instalasi dan kemampuannya untuk bekerja di berbagai sistem operasi, termasuk Windows, Linux, dan macOS (Roza et al., 2020).

2.7 Flowchart

Flowchart adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian (khoerul ummah, 2022). Flowchart terdiri dari simbol-simbol grafis, flowchart mengilustrasikan tugas, keputusan, serta aliran data dalam alur kerja yang

bertujuan untuk mempermudah pemahaman, merancang, dan menganalisis agar dapat dipahami dengan mudah. Setiap *step* atau tugas dalam flowchart digambarkan dengan menggunakan simbol khusus dan dihubungkan dengan garis atau panah. Berikut merupakan simbol flowchart, nama, serta arti dari tiap simbol.

Tabel 2. 2 Simbol *Flowchart*

No.	Simbol <i>Flowchart</i>	Nama	Arti Simbol <i>Flowchart</i>
1		<i>Terminator</i>	Awalan atau akhir konsep (prosedur)
2		<i>Process</i>	Penggambaran bentuk proses operasional
3		<i>Document</i>	Dokumen/laporan berbentuk <i>print out</i>
4		<i>Decision</i>	Keputusan atau <i>sub-pont.</i>
5		<i>Data</i>	<i>Input</i> atau <i>output</i>
6		<i>On-Page Reference/Connector</i>	Penghubung alur dalam <i>page</i> yang sama

7		<i>Off-Page Reference / Off-Page Connector</i>	Penghubung alur dalam <i>different page</i>
8		<i>Flow</i>	Arah alur dalam sebuah prosedur

Terdapat 8 simbol *flowchart* yang masing-masing memiliki nama dan arti yang berbeda. *Flowchart* memiliki manfaat yang besar dalam membantu mengkomunikasikan ide dan rencana secara visual dan mudah untuk dipahami oleh orang yang melihatnya.

BAB III

METODE PENELITIAN

3.1 Alur Penelitian (*Flowchart*)

Alur penelitian ini dirancang untuk mengembangkan sistem berbasis web yang dapat memklasifikasi keberhasilan siswa dalam seleksi SMP Negeri dengan menggunakan algoritma *Gradient Boosting Machine* (GBM). Penelitian ini dimulai dengan pengumpulan data yang relevan, dilanjutkan dengan pengolahan dan pemodelan data menggunakan GBM, dan diakhiri dengan implementasi sistem berbasis web. Berikut adalah tahapan utama yang dilakukan dalam penelitian ini.

3.1.1. Pengumpulan Data

Pengumpulan data dilakukan dengan mengakses data historis siswa dari SD Muhammadiyah Medan Timur, yang meliputi data nilai rapor, tingkat kehadiran, dan prestasi non-akademik. Data ini dikumpulkan melalui pihak sekolah dan digunakan untuk membangun *dataset* yang akan dianalisis menggunakan model GBM.

3.1.2. Pra-Pemrosesan Data

Data yang telah dikumpulkan tidak langsung digunakan dalam model. Sebelum itu, dilakukan beberapa tahap pengolahan data, termasuk pembersihan data (*data cleaning*), penanganan data yang hilang (*missing values*), dan konversi data kategorikal menjadi format numerik menggunakan teknik *encoding*. Selain itu, data dinormalisasi untuk memastikan bahwa fitur-fitur numerik memiliki skala yang sama.

3.1.3. Pemisahan Dataset

Dataset yang sudah bersih kemudian dibagi menjadi dua bagian terpisah,

- a. **Data Latih (80%)**: Bagian terbesar dari data yang digunakan untuk "melatih" atau "mengajari" algoritma Gradient Boosting Machine (GBM) dalam mengenali pola keberhasilan siswa.
- b. **Data Uji (20%)**: Bagian yang lebih kecil yang **tidak pernah dilihat** oleh model selama pelatihan. Data ini digunakan nanti untuk menguji seberapa baik performa model pada data baru, sehingga evaluasinya menjadi adil dan objektif.

3.1.4. Pelatihan Model *Gradient Boosting Machine*

Setelah data siap, algoritma GBM digunakan untuk memodelkan data. GBM merupakan teknik ensemble learning yang menggabungkan beberapa model sederhana untuk membentuk model yang lebih kuat. Proses pelatihan model dilakukan dengan menggunakan data pelatihan, dan parameter model disesuaikan menggunakan teknik *hyperparameter tuning*.

3.1.5. Evaluasi dan Simpan Model

Model yang telah dilatih diuji menggunakan data uji untuk mengukur akurasi dan performa klasifikasi. Evaluasi dilakukan menggunakan metrik-metrik seperti akurasi, *precision*, *recall*, dan *F1-score* untuk memastikan bahwa model dapat memberikan klasifikasi yang akurat dan dapat diandalkan. Jika hasilnya memuaskan, model yang sudah "pintar" ini akan disimpan dalam format file `.pkl` atau `.joblib`. Tujuannya agar model dapat digunakan kembali di dalam aplikasi tanpa perlu melatih ulang dari awal.

3.1.6. Implementasi dan Pengujian Sistem

Pada bagian ini, sistem berbasis web dibangun dengan tujuan untuk mengintegrasikan model *Gradient Boosting Machine (GBM)* dengan antarmuka pengguna berbasis web. Untuk mewujudkan hal ini, digunakan *Flask*, sebuah *Framework web* berbasis *Python*, yang memungkinkan pembuatan aplikasi web yang ringan dan mudah diintegrasikan dengan berbagai komponen. Proses implementasi sistem berbasis web dilakukan dengan mengikuti langkah-langkah berikut:

1. Penggunaan *Flask* sebagai *Backend*

Flask digunakan sebagai *backend* dalam pengembangan sistem ini. *Framework* ini memungkinkan pembuatan *API* yang dapat menangani berbagai permintaan (*requests*) dari pengguna, termasuk pengolahan data yang dimasukkan ke dalam sistem. *Flask* akan mengatur alur data dari pengguna, memprosesnya menggunakan model *GBM* yang telah dilatih, dan mengembalikan hasil klasifikasi kepada pengguna.

2. Integrasi dengan Model *GBM* menggunakan *Flask API*

Untuk menghubungkan model *GBM* yang dibangun dengan *scikit-learn* pada sisi *backend* dengan aplikasi berbasis web, digunakan *Flask API*. Model *GBM* yang telah dilatih disimpan dalam bentuk objek *pickle* atau *joblib* dan dimuat saat aplikasi dijalankan. Ketika pengguna memasukkan data siswa melalui antarmuka pengguna, *Flask* akan menerima permintaan tersebut melalui *API* dan memprosesnya menggunakan model *GBM*. Setelah data diproses, hasil klasifikasi (apakah siswa berpotensi diterima di SMP Negeri atau tidak) akan dikirim kembali ke pengguna melalui *response* yang dikembalikan oleh *Flask API*.

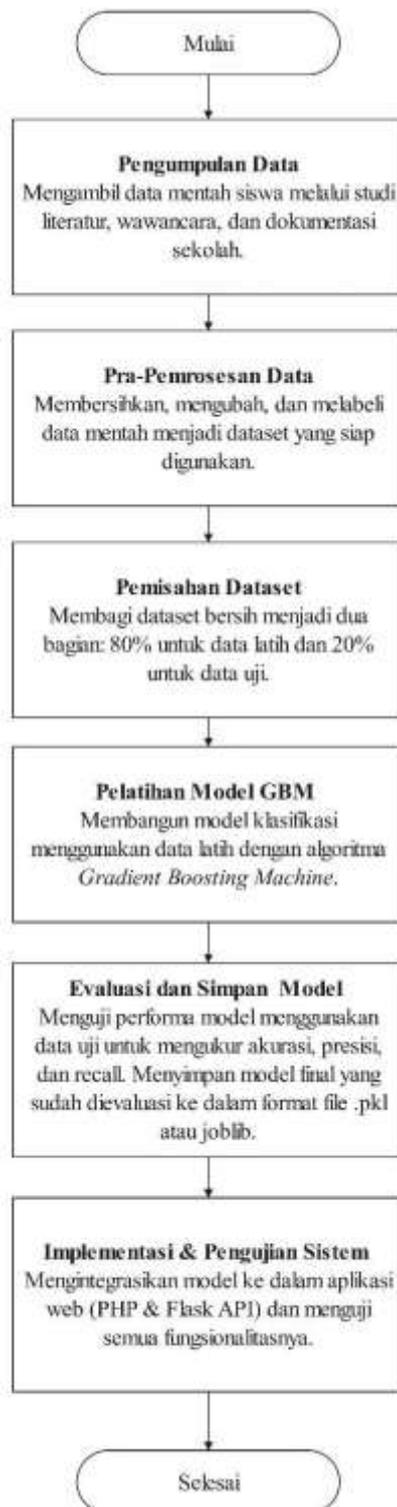
3. *Frontend* menggunakan HTML, CSS, dan Javascript

Antarmuka pengguna dibangun dengan menggunakan HTML, CSS, dan JavaScript untuk memastikan tampilan yang responsif dan interaktif. Data yang dimasukkan oleh pengguna, seperti nilai rapor, tingkat kehadiran, dan prestasi non-akademik, akan dikirim ke *backend* melalui *POST request* menggunakan formulir yang telah disediakan. Setelah data diproses oleh *backend* menggunakan *Flask API*, hasil klasifikasi akan ditampilkan di halaman web kepada pengguna.

4. Pengolahan Data dan Pengambilan Keputusan di *Backend*

Setelah menerima data siswa dari antarmuka pengguna, *Flask* akan mengirim data tersebut ke model *GBM* yang telah dilatih. Model akan memproses data tersebut dan menghasilkan klasifikasi berdasarkan pola-pola yang telah ditemukan selama pelatihan. Hasil klasifikasi ini kemudian diteruskan kembali ke *frontend*, di mana informasi tersebut dapat dilihat oleh pihak sekolah untuk mengambil keputusan yang lebih tepat.

Setelah integrasi model *GBM* dengan *Flask API* selesai, sistem akan diuji untuk memastikan bahwa *backend* dapat berkomunikasi dengan benar dengan *frontend*. Pengujian ini melibatkan pengisian data oleh pengguna dan pengecekan apakah hasil klasifikasi yang diberikan sesuai dengan data yang dimasukkan. Selain itu, validasi sistem dilakukan untuk memastikan bahwa API dapat menangani berbagai jenis permintaan dengan baik dan respons yang diberikan sesuai dengan yang diharapkan.



Gambar 3.1 Flowchart Penelitian

3.2 Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan melalui dua metode utama, yaitu wawancara dan dokumentasi. Metode ini dipilih untuk memastikan bahwa data yang diperoleh bersumber langsung dari pihak yang berkompeten di sekolah dan menggunakan data historis yang tersedia untuk analisis lebih lanjut. Berikut adalah penjelasan mendalam tentang kedua metode pengumpulan data yang digunakan

3.2.1. Wawancara

Wawancara dilakukan dengan pihak-pihak yang memiliki pengetahuan mendalam tentang proses akademik dan bimbingan siswa, khususnya Kepala Sekolah dan pihak tata usaha di SD Muhammadiyah Medan Timur. Wawancara ini bersifat semi-terstruktur, yang memungkinkan peneliti untuk menggali informasi lebih dalam mengenai faktor-faktor yang mempengaruhi keberhasilan siswa dalam seleksi SMP Negeri.

Beberapa pertanyaan yang diajukan dalam wawancara meliputi:

1. Apa saja indikator yang biasanya menunjukkan keberhasilan siswa dalam diterima di SMP Negeri?
2. Sejauh mana nilai rapor mempengaruhi peluang siswa diterima di sekolah lanjutan?
3. Apakah kehadiran siswa atau keikutsertaan dalam kegiatan ekstrakurikuler juga diperhitungkan?
4. Data apa saja yang tersedia dan biasa digunakan sekolah untuk mengevaluasi siswa kelas VI?

5. Apakah pihak sekolah pernah atau sedang menggunakan sistem berbasis teknologi untuk membantu klasifikasi atau bimbingan siswa?
6. Menurut Anda, apakah sistem klasifikasi keberhasilan siswa berbasis web akan bermanfaat bagi sekolah?

3.2.2. Dokumentasi

Selain wawancara, metode dokumentasi digunakan untuk memperoleh data yang lebih spesifik dan objektif, yang dapat digunakan sebagai *dataset* utama dalam penelitian ini. Dokumentasi dilakukan dengan meminta arsip nilai dan data siswa dari pihak sekolah, yang kemudian akan diolah untuk kebutuhan pelatihan dan pengujian model klasifikasi.

Dokumentasi ini meliputi data-data berikut:

1. Nilai rapor siswa: Data nilai rapor siswa dari semester 1 hingga semester 5 yang menunjukkan kinerja akademik siswa selama di SD.
2. Data kehadiran siswa: Catatan kehadiran siswa dalam setiap kegiatan pembelajaran dan kegiatan lain yang relevan.
3. Data prestasi akademik dan non-akademik: Informasi tentang prestasi siswa di luar kegiatan akademik, seperti keterlibatan dalam ekstrakurikuler, lomba, atau kegiatan sosial lainnya, jika tersedia.
4. Hasil kelulusan siswa ke SMP Negeri: Data kelulusan siswa ke SMP Negeri dalam 2-3 tahun terakhir, termasuk label apakah siswa diterima atau tidak diterima.

3.3 Tahapan Pengolahan Data

Pada tahapan ini, data mentah yang telah dikumpulkan akan diproses untuk diubah menjadi *dataset* yang siap digunakan dalam membangun model klasifikasi

menggunakan algoritma *Gradient Boosting Machine (GBM)*. Pengolahan data melibatkan beberapa tahapan penting yang perlu dijelaskan secara rinci:

3.3.1. Pra-Pemrosesan Data (*Preprocessing*)

Tahapan pra-pemrosesan bertujuan untuk mempersiapkan data agar dapat digunakan dalam model *GBM*. Beberapa tahapan penting dalam proses ini antara lain:

1. *Data Cleaning*

Pada langkah ini, data yang telah dikumpulkan akan dibersihkan dengan menghapus data duplikat, menangani data yang hilang (*missing values*) dengan metode yang sesuai (misalnya, imputasi menggunakan rata-rata atau modus), serta menghapus entri yang tidak relevan dengan tujuan penelitian.

2. *Data Transformation*

Data kategorikal seperti jenis kelamin atau status penerimaan siswa akan diubah menjadi format numerik agar dapat digunakan dalam model. Misalnya, status penerimaan siswa bisa diubah menjadi format biner dengan penandaan "Diterima" = 1 dan "Tidak Diterima" = 0.

3. Normalisasi Data

Jika data memiliki nilai yang sangat bervariasi (misalnya nilai rapor yang memiliki rentang nilai besar), normalisasi dilakukan untuk menyamakan skala antar atribut. Metode yang digunakan dapat berupa *Z-Score* atau *Min-Max Scaling*, tergantung pada distribusi data.

4. *Labelling*

Pada tahapan ini, label kelas akan ditentukan. Label kelas ini mengacu pada status keberhasilan siswa, apakah diterima di SMP Negeri atau tidak, berdasarkan data historis yang ada.

3.3.2. Seleksi Atribut (*Feature Selection*)

Pada tahap ini, penulis memilih atribut-atribut penting yang dianggap mempengaruhi keberhasilan siswa berdasarkan hasil wawancara dengan pihak sekolah dan dokumen yang tersedia. Beberapa atribut yang dipilih untuk dimasukkan dalam model *GBM* dapat berupa:

1. **Nilai Rata-rata Raport:** Sebagai indikator utama untuk melihat kinerja akademik siswa.
2. **Jumlah Kehadiran:** Mencerminkan kedisiplinan siswa yang dapat memengaruhi keberhasilan mereka.
3. **Prestasi Non-Akademik:** Jika tersedia, seperti keterlibatan dalam ekstrakurikuler atau prestasi lainnya.
4. **Status Penerimaan di SMP Negeri:** Sebagai label yang digunakan untuk pelabelan data, yaitu apakah siswa diterima di *SMP Negeri* atau tidak.

3.4 Pemilihan dan Penerapan Model *Gradient Boosting Machine* (GBM)

Setelah tahap pengolahan data selesai, langkah selanjutnya adalah membangun model klasifikasi menggunakan algoritma *Gradient Boosting Machine* (GBM). Proses pembangunan model ini mencakup beberapa tahapan sebagai berikut:

3.4.1. Pemisahan Data (*Splitting Data*)

Dataset yang sudah diproses dibagi menjadi dua bagian, Data Pelatihan (*Training Data*): Biasanya sekitar 80% dari total data, yang digunakan untuk melatih model. Data Pengujian (*Testing Data*): Sekitar 20% dari total data, yang digunakan untuk menguji dan mengevaluasi kinerja model. Pemisahan data ini bertujuan untuk memastikan bahwa model dapat diuji dengan data yang belum pernah dilihat sebelumnya, sehingga dapat mengukur kemampuan generalisasi model.

3.4.2. Implementasi Algoritma GBM

Pada tahap ini, model *GBM* diimplementasikan menggunakan pustaka *scikit-learn* atau pustaka lainnya di *Python*. *GBM* adalah algoritma *ensemble learning* yang menggabungkan banyak model pembelajaran lemah (*weak learners*) untuk membentuk model yang lebih kuat. Proses pelatihan model dilakukan dengan menggunakan data pelatihan yang telah dipersiapkan sebelumnya, dan parameter model seperti *learning rate*, jumlah *estimators*, dan kedalaman pohon keputusan disesuaikan menggunakan teknik *hyperparameter tuning* untuk meningkatkan kinerja model.

3.4.3. Pengujian Model

Setelah model dilatih, data pengujian digunakan untuk mengukur kinerja model. Pengujian ini melibatkan perhitungan klasifikasi yang benar dan salah, yang membantu dalam mengevaluasi seberapa baik model dapat memklasifikasi status penerimaan siswa di SMP Negeri. Hasil evaluasi model akan dihitung menggunakan beberapa metrik kinerja seperti *akurasi*, *precision*, *recall*, dan *F1-score*.

3.4.4. Evaluasi Model GBM

Setelah model diuji, evaluasi dilakukan menggunakan beberapa metrik yang mengukur kualitas klasifikasi model. Beberapa metrik evaluasi yang digunakan dalam penelitian ini adalah sebagai berikut:

1. **Confusion Matrix:** Menunjukkan jumlah klasifikasi benar dan salah berdasarkan kelas aktual (apakah diterima atau tidak diterima).
2. **Akurasi:** Rasio klasifikasi yang benar dibandingkan dengan total data.
3. **Presisi:** Rasio klasifikasi positif yang benar dari seluruh klasifikasi positif.
4. **Recall:** Rasio klasifikasi positif yang benar dari seluruh data aktual yang positif.
5. **F1-Score:** Rata-rata harmonis antara presisi dan recall, memberikan gambaran umum tentang kinerja model.

3.4.5. Penyimpanan Model

Setelah model *GBM* berhasil dilatih dan dievaluasi, model yang telah terbentuk kemudian disimpan dalam format Pickle (.pkl) atau JSON untuk digunakan di dalam sistem berbasis web. Penyimpanan model ini memungkinkan model untuk dipanggil kembali saat proses klasifikasi data siswa baru dilakukan melalui antarmuka sistem. Dengan demikian, model yang telah dilatih dapat diakses dan digunakan oleh pihak sekolah secara berkelanjutan tanpa perlu melatih ulang model setiap kali.

3.5 Simulasi Perhitungan *Gradient Boosting Machine*

Pada bagian ini, penulis akan menjelaskan Langkah-langkah perhitungan algoritma *Gradient Boosting Machine* (GBM) yang digunakan untuk mengklasifikasikan keberhasilan siswa dalam seleksi SMP Negeri. Simulasi ini bertujuan untuk mempermudah pemahaman mengenai bagaimana algoritma GBM

bekerja secara iteratif untuk membangun model yang lebih kuat dalam melakukan klasifikasi.

Berikut adalah dataset yang digunakan dalam simulasi ini untuk memprediksi keberhasilan siswa dalam diterima di SMP Negeri. Dataset ini mencakup lima siswa dengan atribut-atribut berikut :

- a. Nilai Raport : Rata-rata nilai akademik siswa dari Semester 1 hingga Semester 5 (nilai antara 0 hingga 100).
- b. Kehadiran (%) : Persentase kehadiran siswa di kelas, mencerminkan tingkat disiplin dan keterlibatan dalam pembelajaran (nilai antara 0 hingga 100%).
- c. Prestasi Non-Akademik: Skor yang menunjukkan partisipasi siswa dalam kegiatan ekstrakurikuler (nilai antara 0 hingga 5).
- d. Status Diterima: Label yang menunjukkan apakah siswa diterima di SMP Negeri (1 = diterima, 0 = tidak diterima).

Berikut adalah contoh dataset simulasi yang digunakan :

Tabel 3. 1 Tabel Dataset yang digunakan

Siswa	Nilai Rapor	Kehadiran (%)	Prestasi Non-Akademik	Status Diterima (y)
1	85	95	4	1
2	70	80	0	0
3	90	98	5	1
4	60	75	2	0
5	80	85	3	1

Perhitungan Iterasi 1 : Klasifikasi Awal

Pada iterasi pertama, model GBM dimulai dengan model dasar yang membuat klasifikasi berdasarkan nilai rata-rata dari target variabel, yaitu Status Diterima. Hal ini dilakukan dengan menghitung rata-rata dari nilai target y_i untuk seluruh dataset, yang akan digunakan sebagai prediksi awal untuk semua siswa.

Rumus untuk menghitung klasifikasi awal adalah sebagai berikut:

$$\hat{y}_0 = \frac{1}{N} \sum_{i=1}^N y_i \dots \dots \dots (3.1)$$

Dengan y_i adalah nilai target (Status Diterima) dan N adalah jumlah total data (5 siswa). Berdasarkan data yang ada, rata-rata nilai target adalah :

$$\hat{y}_0 = \frac{1 + 0 + 1 + 0 + 1}{5}$$

$$\hat{y}_0 = \frac{3}{5}$$

$$\hat{y}_0 = 0.6$$

Dengan demikian, klasifikasi pertama untuk semua siswa adalah 0.6, yang berarti model pertama mengasumsikan bahwa setiap siswa memiliki 60% kemungkinan diterima di SMP Negeri. Setelah klasifikasi awal dihitung, langkah berikutnya adalah menghitung residual error r_i untuk setiap siswa. Residual ini menunjukkan seberapa besar kesalahan antara nilai target yang sesungguhnya dan klasifikasi yang diberikan oleh model. Perhitungan residual untuk setiap siswa adalah sebagai berikut

Tabel 3. 2 Nilai Residual

Siswa	Status Diterima	Klasifikasi Model Awal	Residual
1	1	0.6	$1 - 0.6 = 0.41 - 0.6 = 0.4$
2	0	0.6	$0 - 0.6 = -0.60 - 0.6 = -0.6$
3	1	0.6	$1 - 0.6 = 0.41 - 0.6 = 0.4$
4	0	0.6	$0 - 0.6 = -0.60 - 0.6 = -0.6$
5	1	0.6	$1 - 0.6 = 0.41 - 0.6 = 0.4$

Nilai residual dalam tabel ini menunjukkan selisih antara nilai target sebenarnya (Status Diterima) dan klasifikasi model awal. Residual positif (seperti pada siswa 1, 3, dan 5) menunjukkan bahwa model meremehkan kemungkinan siswa diterima, yaitu model memprediksi siswa memiliki peluang lebih rendah dari

yang seharusnya, sedangkan residual negatif (seperti pada siswa 2 dan 4) menunjukkan bahwa model terlalu optimis dalam memprediksi penerimaan siswa, memberikan peluang lebih tinggi dari kenyataannya. Dengan kata lain, residual ini menggambarkan kesalahan prediksi model dan memberikan informasi penting tentang bagaimana model perlu diperbaiki pada iterasi berikutnya, baik dengan menurunkan atau meningkatkan prediksi untuk mencapai klasifikasi yang lebih akurat.

Setelah menghitung residual, langkah berikutnya dalam proses Gradient Boosting Machine (GBM) adalah menghitung gradien dari fungsi kerugian untuk memperbaiki kesalahan yang ada. Gradien ini digunakan untuk memberi arahan kepada model dalam memperbaiki prediksi yang telah dilakukan, dengan tujuan mengurangi kesalahan residual yang terdeteksi pada iterasi pertama. Fungsi kerugian yang digunakan dalam GBM adalah Mean Squared Error (MSE), yang dihitung dengan rumus:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \dots\dots\dots (3.2)$$

Gradien dihitung dengan mengambil turunan pertama dari fungsi kerugian ini terhadap prediksi model, yang dapat dituliskan sebagai

$$g_i = 2 \cdot (\hat{y}_i - y_i) \dots\dots\dots (3.3)$$

Di mana :

g_i adalah gradien untuk data ke- i

\hat{y}_i adalah prediksi model (klasifikasi awal)

y_i adalah nilai target sebenarnya (Status Diterima).

Gradien memberikan sinyal kepada model tentang arah dan besar perubahan yang perlu dilakukan untuk memperbaiki kesalahan pada prediksi sebelumnya. Jika

gradien positif, model perlu meningkatkan prediksinya, dan jika gradien negatif, model perlu mengurangi prediksinya. Berikut adalah perhitungan gradien untuk setiap siswa berdasarkan residual yang telah menghitung sebelumnya :

Tabel 3. 3 Tabel Perhitungan Gradien

Siswa	Residual (y_i)	Klasifikasi Model Awal (\hat{y}_i)	Gradien (g_i)
1	0.4	0.6	$2 \times (0.6 - 1) = -0.8$
2	-0.6	0.6	$2 \times (0.6 - 0) = 1.2$
3	0.4	0.6	$2 \times (0.6 - 1) = -0.8$
4	-0.6	0.6	$2 \times (0.6 - 0) = 1.2$
5	0.4	0.6	$2 \times (0.6 - 1) = -0.8$

Dengan menggunakan gradien yang dihitung, langkah berikutnya adalah pembaruan klasifikasi model. Pembaruan ini dilakukan dengan menambahkan kontribusi dari gradien yang dihitung pada model sebelumnya, yang akan memperbaiki kesalahan model pada iterasi pertama. Pembaruan klasifikasi dihitung dengan rumus :

$$\hat{y}_1 = \hat{y}_0 + \eta \cdot g_i \dots\dots\dots(3.3)$$

Dimana :

\hat{y}_1 adalah klasifikasi yang diperbarui

\hat{y}_0 adalah klasifikasi lama (awal),

g_i adalah gradien yang dihitung

η adalah *learning rate* yang mengontrol seberapa besar kontribusi gradien terhadap pembaruan (dalam contoh perhitungan ini, *learning rate* = 0.1).

Perhitungan pembaruan klasifikasi untuk setiap siswa adalah sebagai berikut

Tabel 3. 4 Tabel Klasifikasi Baru pada Iterasi Pertama

Siswa	Klasifikasi Model Awal (\hat{y}_0)	Gradien (g_i)	Klasifikasi Baru (\hat{y}_1)
1	0.6	-0.8	$0.6 + 0.1 \times (-0.8) = 0.52$

2	0.6	1.2	$0.6 + 0.1 \times 1.2 = 0.72$
3	0.6	-0.8	$0.6 + 0.1 \times (-0.8) = 0.52$
4	0.6	1.2	$0.6 + 0.1 \times 1.2 = 0.72$
5	0.6	-0.8	$0.6 + 0.1 \times (-0.8) = 0.52$

Setelah pembaruan klasifikasi ini, model GBM akan terus berlanjut untuk iterasi berikutnya, di mana langkah-langkah yang sama akan diulang untuk memperbaiki prediksi lebih lanjut dan meningkatkan akurasi model.

Perhitungan Iterasi 2 :

Pada iterasi pertama, telah diperoleh klasifikasi model pertama, pada iterasi kedua, digunakan klasifikasi yang diperbarui dari iterasi pertama sebagai dasar menghitung residual baru. Perhitungan residual untuk setiap siswa pada iterasi kedua.

Tabel 3. 5 Tabel Perhitungan Residual pada iterasi kedua

Siswa	Status Diterima	Klasifikasi model iterasi sebelumnya \hat{y}_i	Residual
1	1	0.52	$1 - 0.6 = 0.41 - 0.6 = 0.4$
2	0	0.72	$0 - 0.6 = -0.60 - 0.6 = -0.6$
3	1	0.52	$1 - 0.6 = 0.41 - 0.6 = 0.4$
4	0	0.72	$0 - 0.72 = -0.72$
5	1	0.52	$1 - 0.52 = 0.48$

Setelah menghitung residual pada iterasi kedua, perhitungan gradiennya adalah sebagai berikut

Tabel 3. 6 Tabel Perhitungan Gradien Pada Iterasi kedua

Siswa	Residual (y_i)	Klasifikasi Model Iterasi Sebelumnya (\hat{y}_i)	Gradien (g_i)
1	0.48	0.52	$2 \cdot (0.52-1) = -0.96$
2	-0.72	0.72	$2 \cdot (0.72 - 0) = 1.44$
3	0.48	0.52	$2 \cdot (0.52 - 1) = -0.96$
4	-0.72	0.72	$2 \cdot (0.72 - 0) = 1.44$
5	0.48	0.52	$2 \cdot (0.52 - 1) = -0.96$

Setelah menghitung gradien, klasifikasi model diperbarui dengan menambahkan kontribusi dari gradien yang dihitung sebagai berikut :

Siswa	Klasifikasi Lama (\hat{y}_1)	Gradien (g_i)	Klasifikasi Baru (\hat{y}_2)
1	0.52	-0.96	$0.52 + 0.1 \times (-0.96) = 0.42$
2	0.72	1.44	$0.72 + 0.1 \times 1.44 = 0.86$
3	0.52	-0.96	$0.52 + 0.1 \times (-0.96) = 0.42$
4	0.72	1.44	$0.72 + 0.1 \times 1.44 = 0.86$
5	0.52	-0.96	$0.52 + 0.1 \times (-0.96) = 0.42$

Pada iterasi kedua, model memperbarui prediksinya berdasarkan gradien yang dihitung. Pembaruan klasifikasi dilakukan untuk setiap siswa dengan menambahkan kontribusi dari gradien yang dihitung. Hasil klasifikasi baru (iterasi kedua) menunjukkan prediksi yang lebih baik daripada klasifikasi awal, namun proses ini akan terus berlanjut pada iterasi berikutnya untuk meningkatkan akurasi model dengan lebih banyak weak models.

3.6 Arsitektur Sistem

Untuk membangun sistem klasifikasi keberhasilan siswa yang efektif dan interaktif, penelitian ini menggunakan arsitektur berbasis web modern. Arsitektur ini memisahkan logika backend (pemrosesan data dan machine learning) dengan frontend (antarmuka pengguna). Sistem ini menggunakan model Gradient Boosting

Machine (GBM) yang dikembangkan dengan Python dan diintegrasikan ke dalam aplikasi web melalui API. Dengan demikian, sekolah dapat dengan mudah memasukkan data siswa, menjalankan klasifikasi, dan melihat hasilnya secara langsung melalui web.

Tabel 3. 7 Komponen Yang digunakan

Komponen	Teknologi	Fungsi/Peran dalam Sistem
Bahasa Pemrograman (Backend)	Python	Digunakan untuk mengembangkan dan melatih model Gradient Boosting Machine (GBM) dalam menganalisis data siswa. Python juga digunakan untuk menghubungkan model dengan API pada backend aplikasi berbasis web.
Lingkungan Machine Learning	Google Colaboratory	Platform cloud berbasis Python untuk mengembangkan dan menjalankan model GBM serta memungkinkan eksperimen analisis data tanpa perlu setup server lokal.
Framework Backend & API	Flask	Framework ringan untuk membangun aplikasi web dan API, memungkinkan interaksi antara frontend dan model GBM untuk memproses data siswa dan mengembalikan hasil klasifikasi.
Bahasa Pemrograman (Frontend)	PHP (Hypertext Preprocessor)	Digunakan untuk pengembangan sisi server web yang mengelola interaksi data antara frontend dan database, serta menyediakan fungsionalitas dinamis pada halaman web.
Struktur Halaman Web	HTML (HyperText Markup Language)	Digunakan untuk membangun struktur halaman web dasar, seperti form input untuk data siswa, tampilan dashboard, dan tampilan hasil klasifikasi yang dihasilkan oleh model GBM.
Styling Antarmuka	Tailwind CSS	Framework CSS yang digunakan untuk mendesain tampilan antarmuka pengguna (UI), membuatnya responsif dan mudah diadaptasi untuk berbagai perangkat, memastikan pengalaman pengguna yang optimal.
Server Lokal	XAMPP	Digunakan untuk menjalankan server lokal yang mendukung pengembangan aplikasi berbasis web, termasuk pengelolaan database MySQL dan

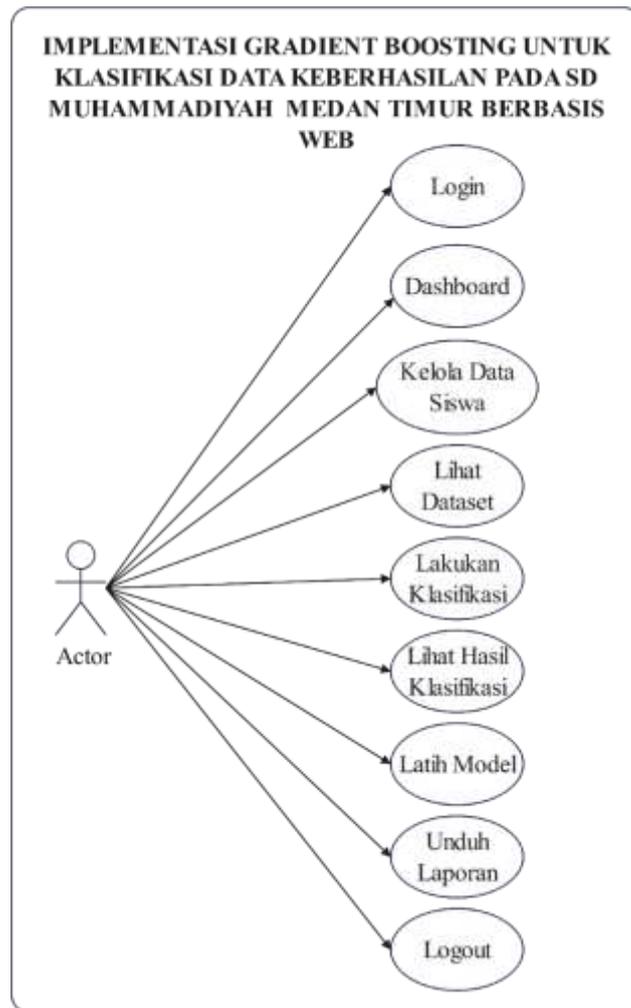
		pengujian sistem sebelum diluncurkan secara penuh.
Basis Data	MySQL	Menyimpan data yang diperlukan untuk aplikasi seperti data siswa, hasil klasifikasi, dan model yang telah dilatih, memungkinkan integrasi data yang efisien untuk analisis lebih lanjut.

3.7 Desain Sistem

Proses ini berfokus pada perancangan seperti struktur data dan arsitektur perangkat lunak yang dibuat dengan pemodelan UML seperti *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*.

3.7.1 Use Case Diagram

Use Case Diagram yang dirancang menggambarkan interaksi antara aktor utama, yaitu Admin atau Guru, dengan sistem klasifikasi keberhasilan siswa masuk SMP Negeri. Diagram ini menunjukkan delapan use case utama yang mencerminkan fungsionalitas sistem, dimulai dari proses *Login* ke Sistem untuk mengakses aplikasi, dilanjutkan dengan *Kelola Data Siswa* yang memungkinkan pengguna untuk menambah, mengedit, atau menghapus data. Pengguna juga dapat *Melihat Dataset* yang tersedia dan melakukan *Klasifikasi* untuk mengetahui kemungkinan keberhasilan siswa. Setelah klasifikasi dilakukan, hasilnya dapat *Dilihat dan Diunduh* dalam bentuk laporan. Selain itu, sistem juga menyediakan fitur *Latih Model Ulang* untuk memperbarui model klasifikasi berdasarkan data terbaru, serta *Simpan Model* agar model yang sudah dilatih dapat digunakan kembali pada klasifikasi berikutnya. Dapat dilihat pada gambar dibawah ini.

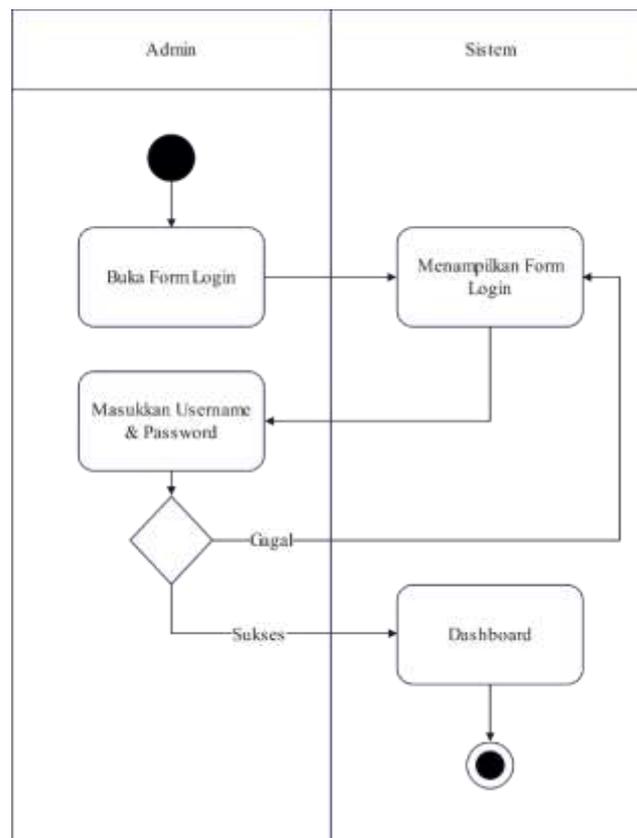


Gambar 3. 2 Use Case Diagram

3.7.2 Activity Diagram

1. Activity Diagram Login

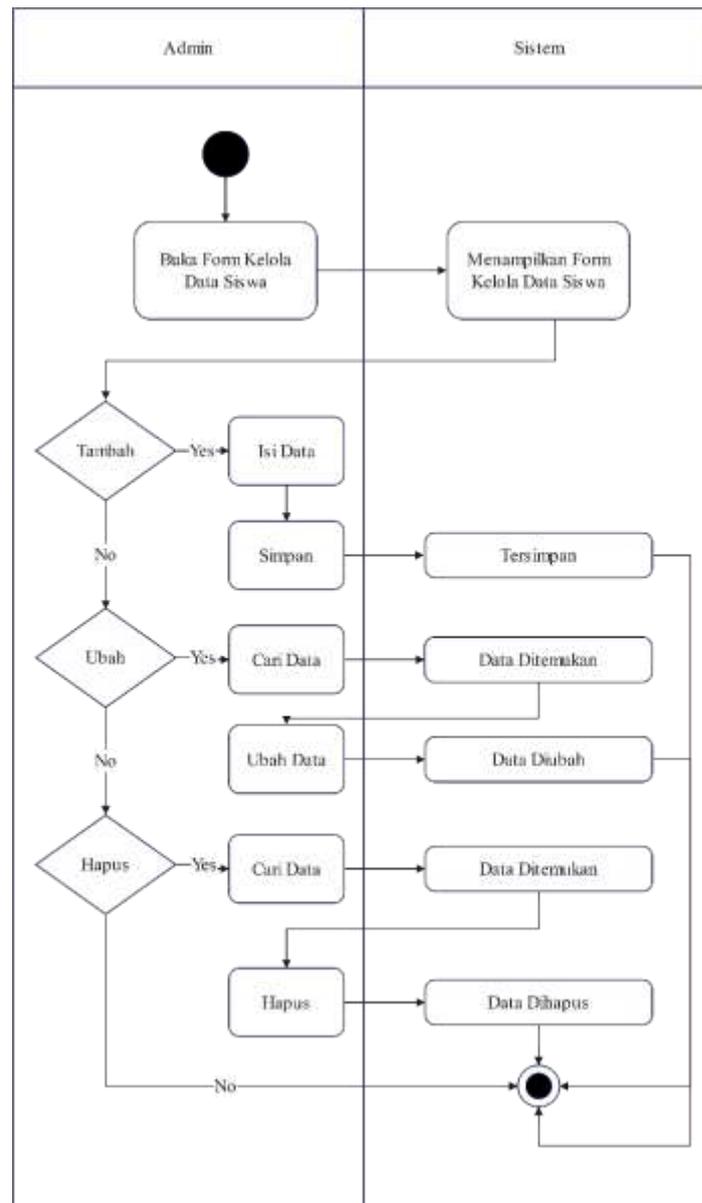
Dalam aktivitas ini, Proses *Login* dimulai ketika admin memasukkan *Username* dan password melalui antarmuka sistem. Sistem kemudian melakukan validasi terhadap kredensial yang dimasukkan. Jika valid, sistem akan mengarahkan admin ke halaman *Dashboard* utama. Jika tidak valid, sistem akan menampilkan halaman *Login*. Dapat dilihat pada gambar dibawah ini



Gambar 3. 3 Activity Diagram Login

2. Activity Diagram Kelola Data Siswa

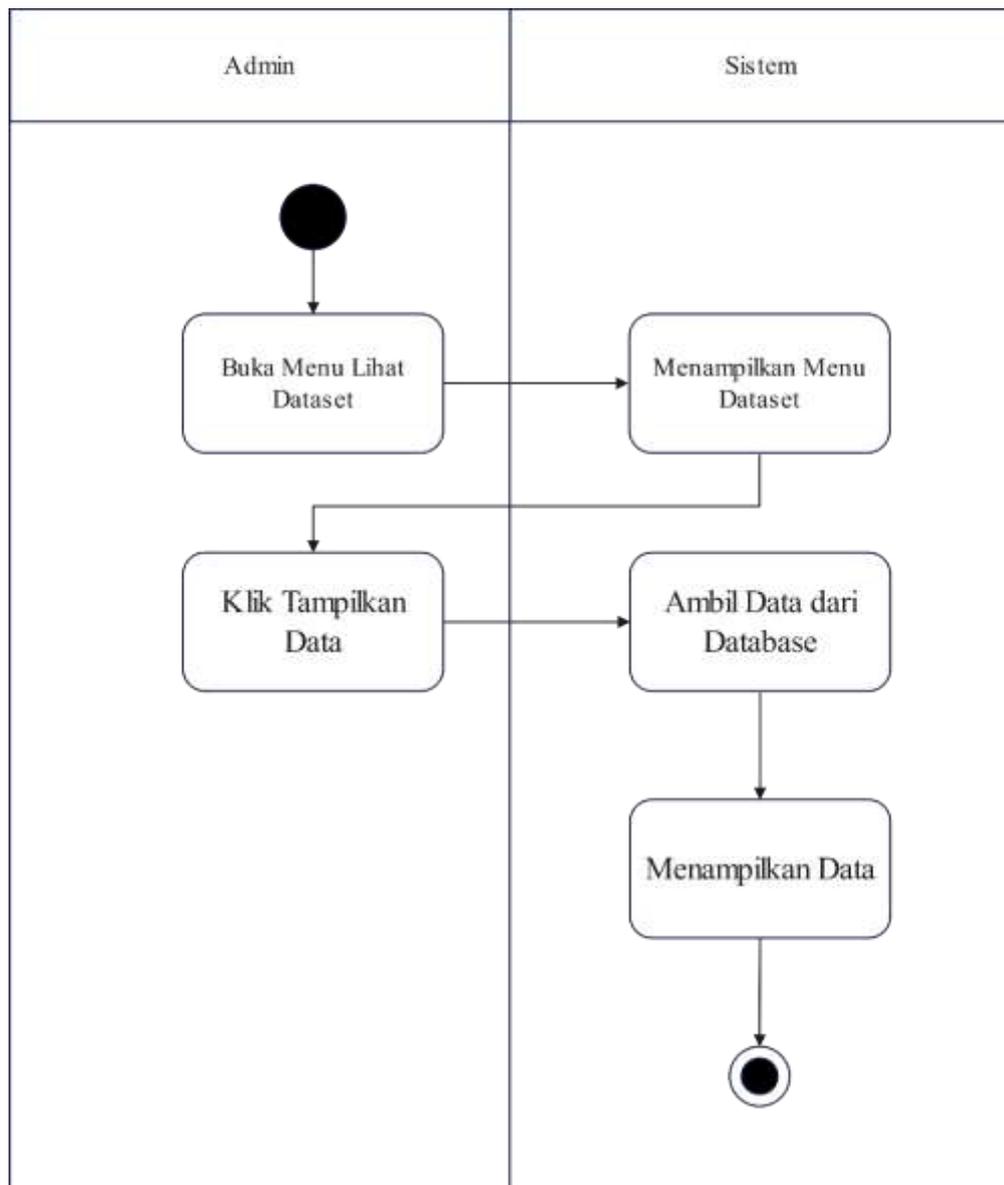
Dalam aktifitas ini, Admin memilih menu “Kelola Data Siswa” untuk melakukan manajemen data siswa. Sistem akan menampilkan daftar siswa dan formulir untuk input data. Admin dapat menambahkan data baru, mengedit data yang sudah ada, atau menghapus data. Setiap perubahan yang dilakukan akan disimpan oleh sistem ke dalam basis data.



Gambar 3. 4 Activity Diagram Kelola Data Siswa

3. Activity Diagram Kelola Dataset

Dalam aktivitas ini, admin mengakses menu “Lihat *Dataset*” untuk meninjau semua data siswa yang telah dimasukkan ke sistem. Sistem akan mengambil data tersebut dari *database* dan menampilkannya dalam bentuk tabel, sehingga admin dapat melakukan review sebelum melakukan proses klasifikasi atau pelatihan ulang model. Dapat dilihat pada gambar dibawah ini

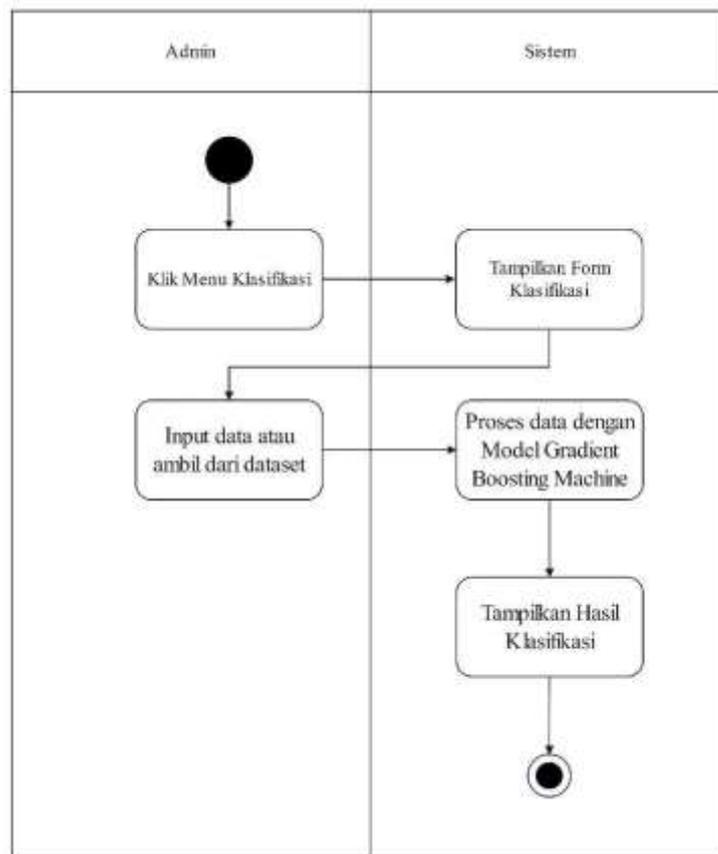


Gambar 3. 5 Activity Diagram Kelola *Dataset*

4. *Activity Diagram* Lakukan Klasifikasi

Dalam aktivitas ini, Admin mengklik menu “Lakukan Klasifikasi” lalu mengisi data siswa baru pada form yang disediakan, seperti nilai rapor, kehadiran, dan prestasi. Sistem kemudian menggunakan model *Gradient Boosting Machine* yang telah dilatih untuk mengklasifikasikan apakah siswa tersebut berpotensi diterima atau

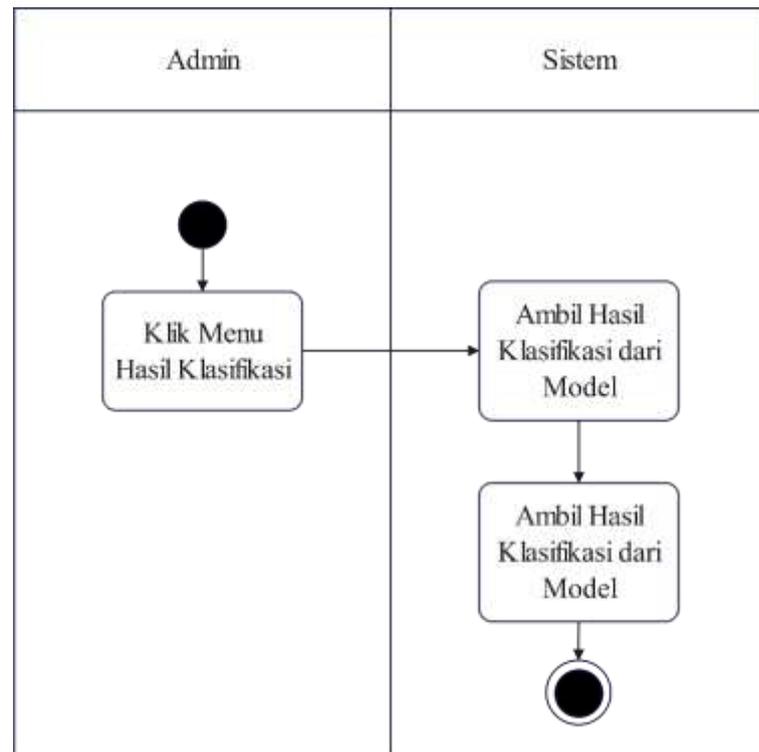
tidak di SMP Negeri. Hasil klasifikasi ditampilkan ke admin secara langsung. Dapat dilihat pada gambar dibawah ini



Gambar 3. 6 Activity Diagram Lakukan Klasifikasi

5. *Activity Diagram* Lihat Hasil Klasifikasi

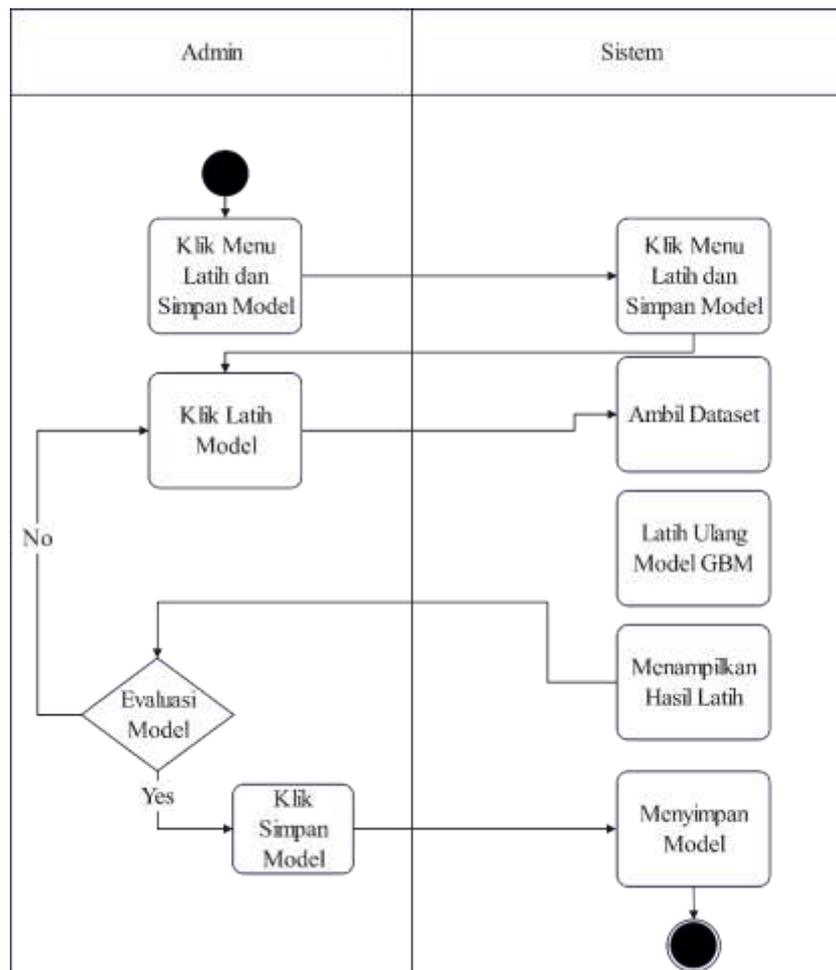
Dalam aktivitas ini, Setelah proses klasifikasi dilakukan, admin dapat melihat hasilnya melalui menu “Lihat Hasil Klasifikasi”. Sistem akan menampilkan label hasil klasifikasi (Ya atau Tidak) beserta informasi pendukung seperti nilai atribut dan aturan keputusan (decision rule) yang digunakan.



Gambar 3. 7 Activity Diagram Lihat Hasil Klasifikasi

6. Activity Diagram Latih dan Simpan Model

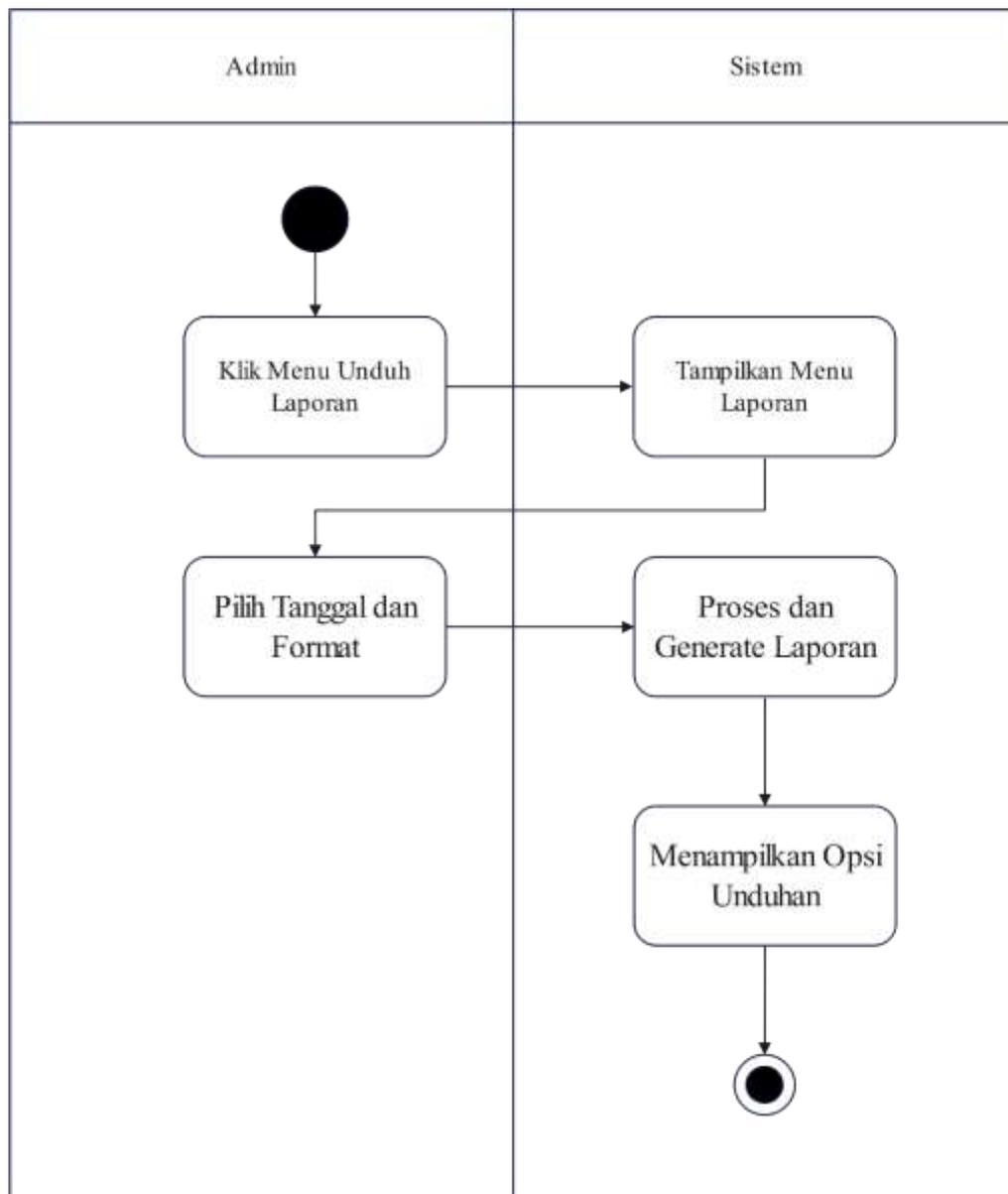
Dalam aktivitas ini, Jika admin ingin memperbarui model berdasarkan data terbaru, mereka dapat mengklik menu “Latih dan Simpan Model”. Sistem akan mengambil semua *dataset* siswa dan menjalankan ulang algoritma *Gradient Boosting Machine* untuk membentuk pohon keputusan baru. Model yang dihasilkan kemudian disimpan agar bisa digunakan kembali untuk proses klasifikasi selanjutnya.



Gambar 3. 8 Activity Diagram Latih dan Simpan Model

7. Activity Diagram Unduh Laporan

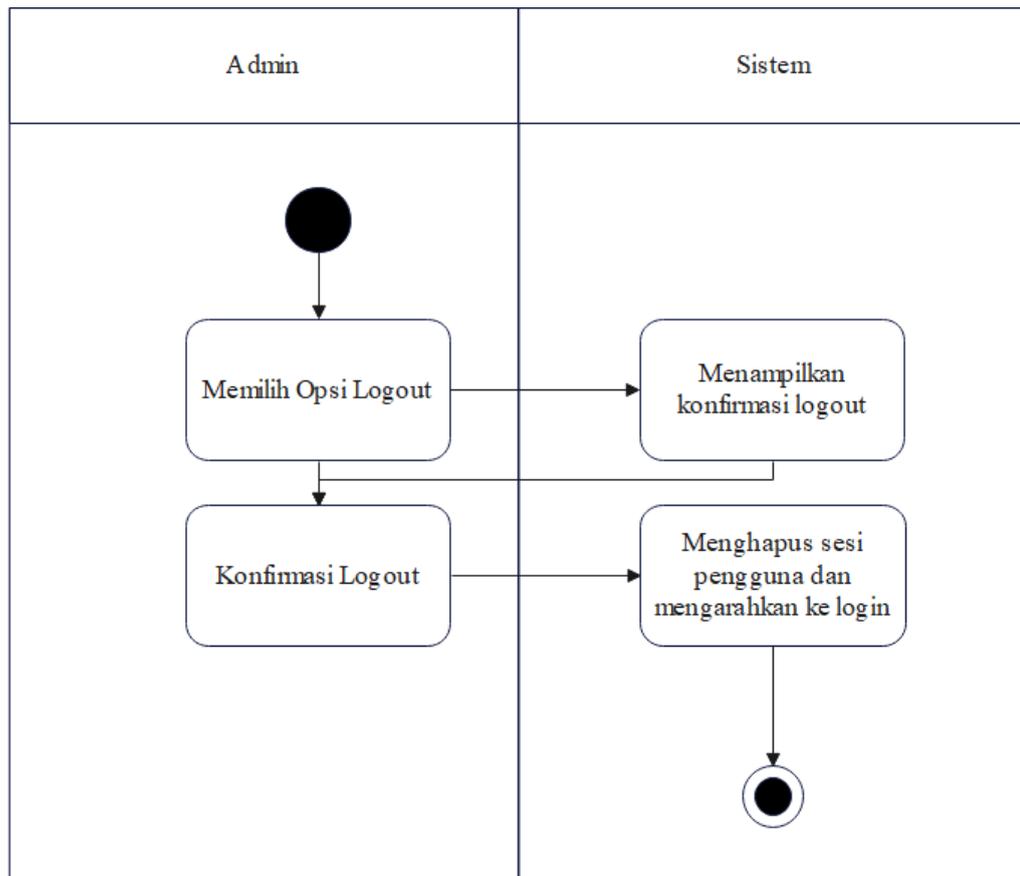
Dalam aktivitas ini, Admin dapat menghasilkan dan mengunduh laporan hasil klasifikasi melalui menu “Unduh Laporan”. Sistem akan memformat data dan hasil klasifikasi ke dalam file seperti PDF atau Excel, lalu memberikan tautan atau tombol untuk mengunduh file tersebut.



Gambar 3.9 Activity Diagram Unduh Laporan

8. Activity Diagram Logout

Dalam aktivitas ini, untuk mengakhiri sesi penggunaan sistem, admin menekan tombol “Logout”. Sistem akan menghapus sesi *Login* dan mengalihkan admin kembali ke halaman *Login* awal, memastikan keamanan akses terhadap data.

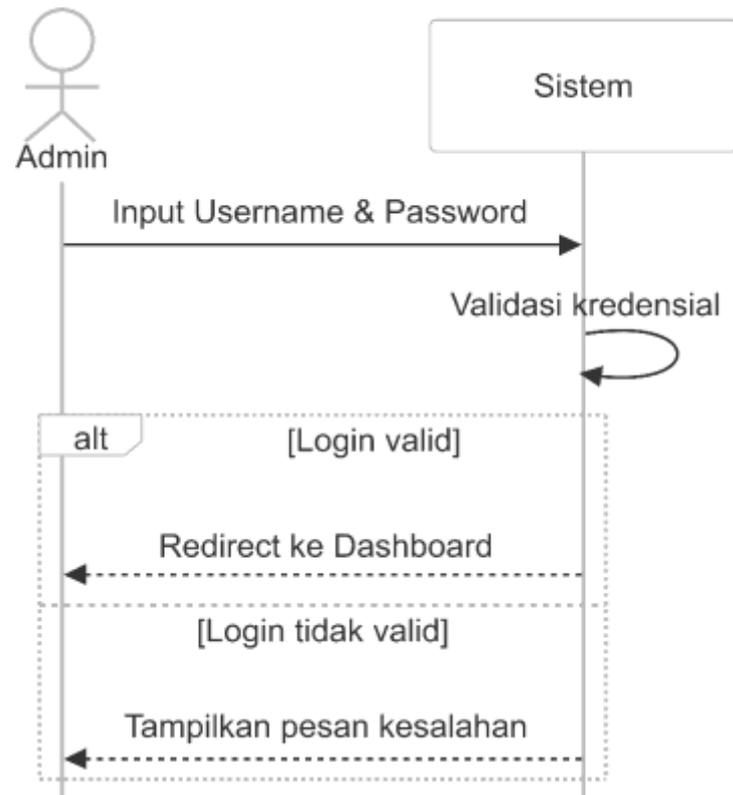


Gambar 3. 10 Activity Diagram *Logout*

3.7.3 *Sequence Diagram*

1. *Sequence Diagram Login*

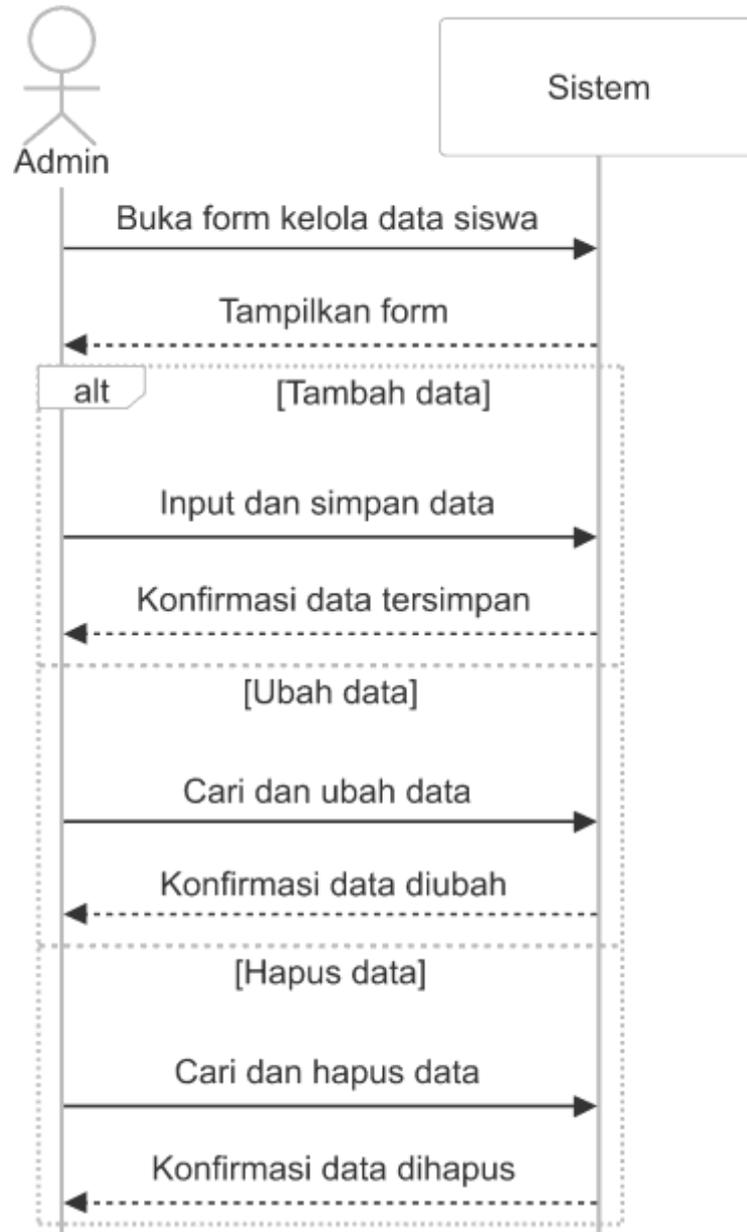
Sequence Diagram yang menggambarkan proses *Login* dapat dilihat pada gambar dibawah ini.



Gambar 3. 11 Sequence Diagram Login

2. Sequence Diagram Kelola Data Siswa

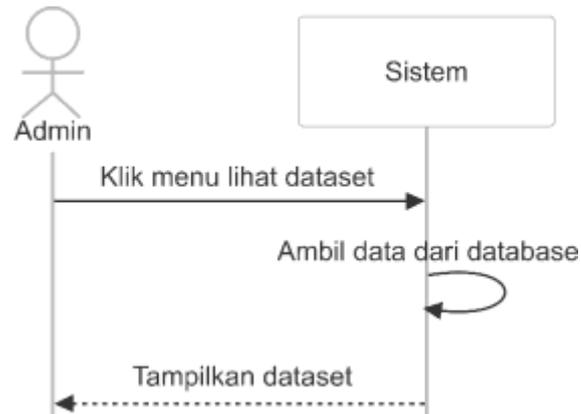
Sequence Diagram yang menggambarkan proses Kelola data siswa dapat dilihat pada gambar dibawah ini.



Gambar 3. 12 *Sequence Diagram* Kelola Data Siswa

3. *Sequence Diagram* Lihat Dataset

Sequence Diagram yang menggambarkan proses lihat *dataset* dapat dilihat pada gambar dibawah ini



Gambar 3. 13 Sequence Diagram Lihat Dataset

4. Sequence Diagram Lakukan Klasifikasi

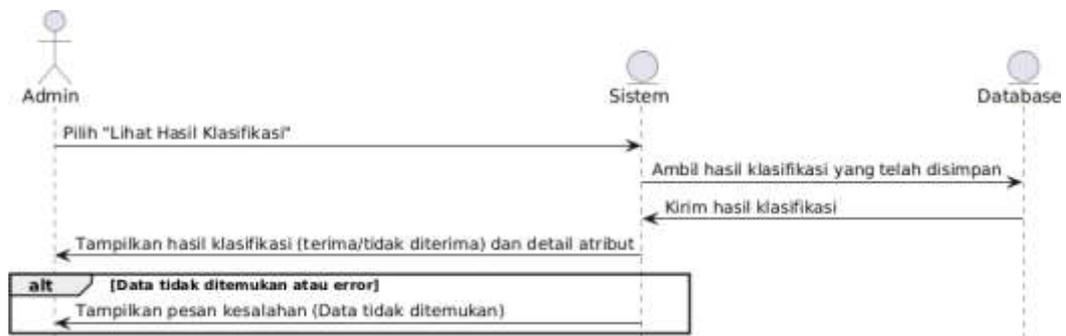
Sequence Diagram yang menggambarkan proses lakukan klasifikasi dengan model *Gradient Boosting Machine* dapat dilihat pada gambar dibawah ini.



Gambar 3. 14 Sequence Diagram Lakukan Klasifikasi

5. Sequence Diagram Lihat Hasil Klasifikasi

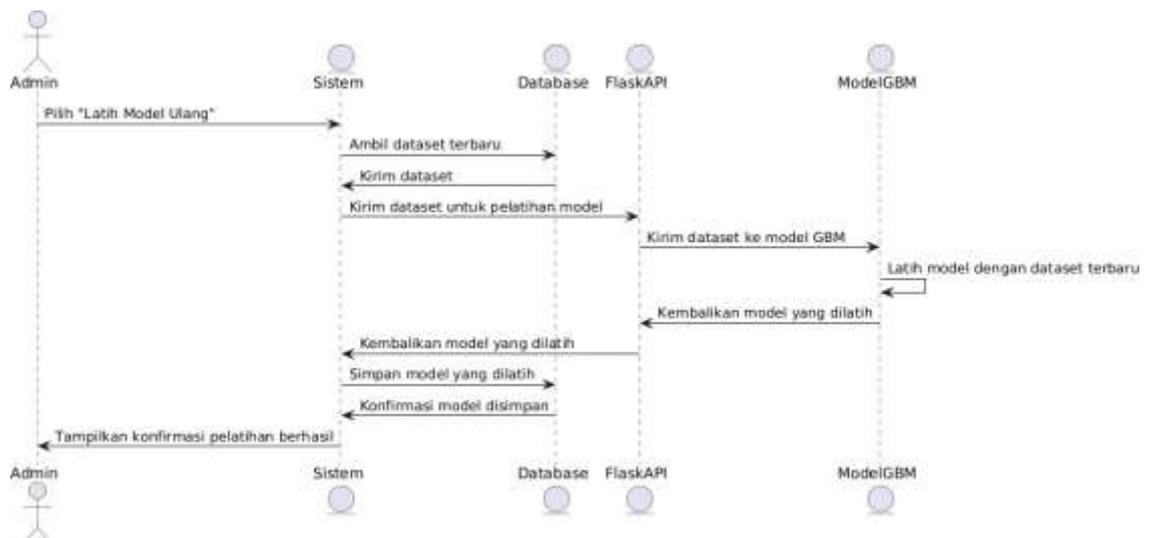
Sequence Diagram yang menggambarkan proses lihat hasil klasifikasi dapat dilihat pada gambar dibawah ini.



Gambar 3. 15 Sequence Diagram Lihat Hasil Klasifikasi

6. Sequence Diagram Latih Model

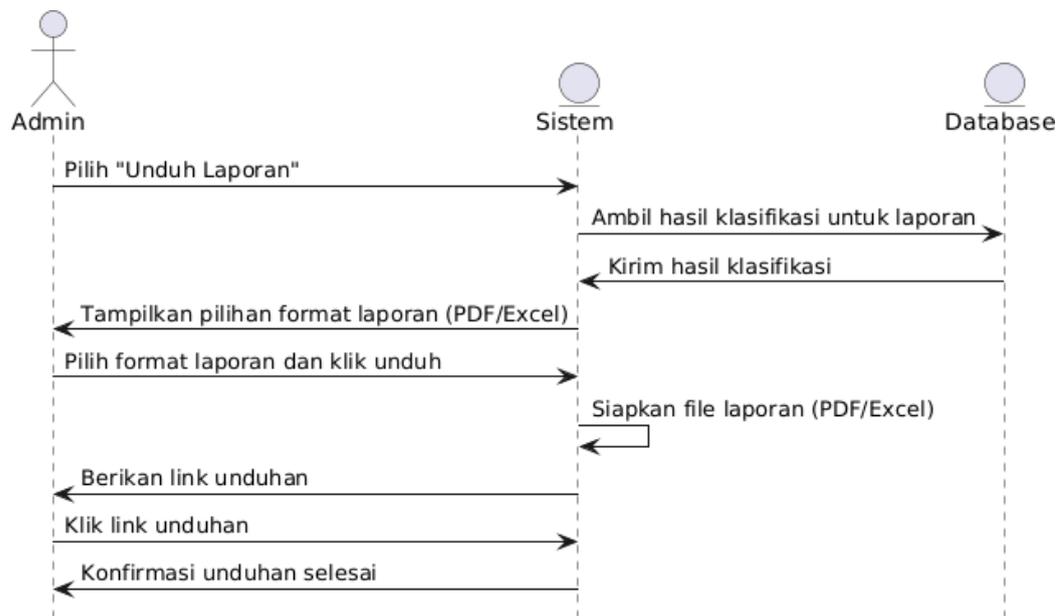
Sequence Diagram yang menggambarkan proses latih model dapat dilihat pada gambar dibawah ini.



Gambar 3. 16 Sequence Diagram Latih Model

7. Sequence Diagram Unduh Laporan

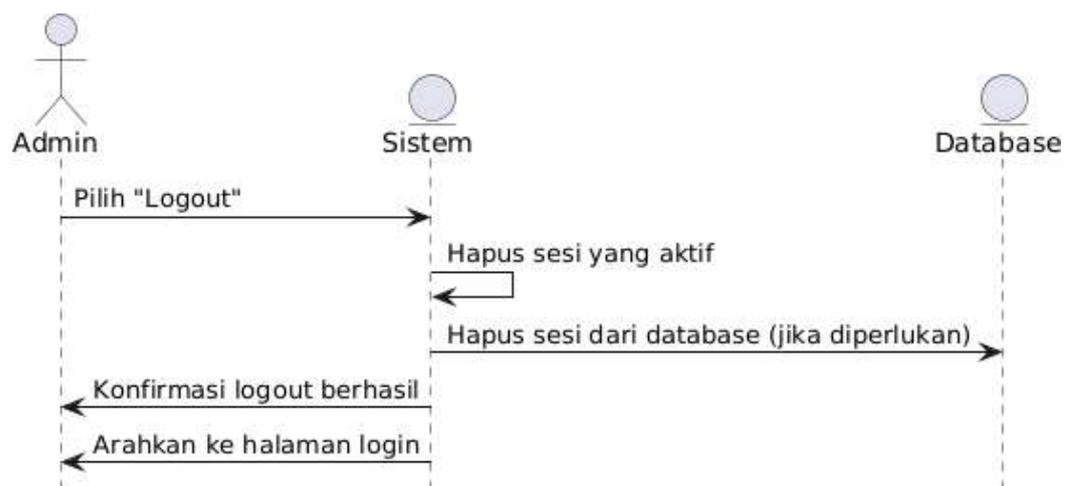
Sequence Diagram yang menggambarkan proses unduh laporan dapat dilihat pada gambar dibawah ini.



Gambar 3. 17 Sequence Diagram Unduh Laporan

8. *Sequence Diagram Logout*

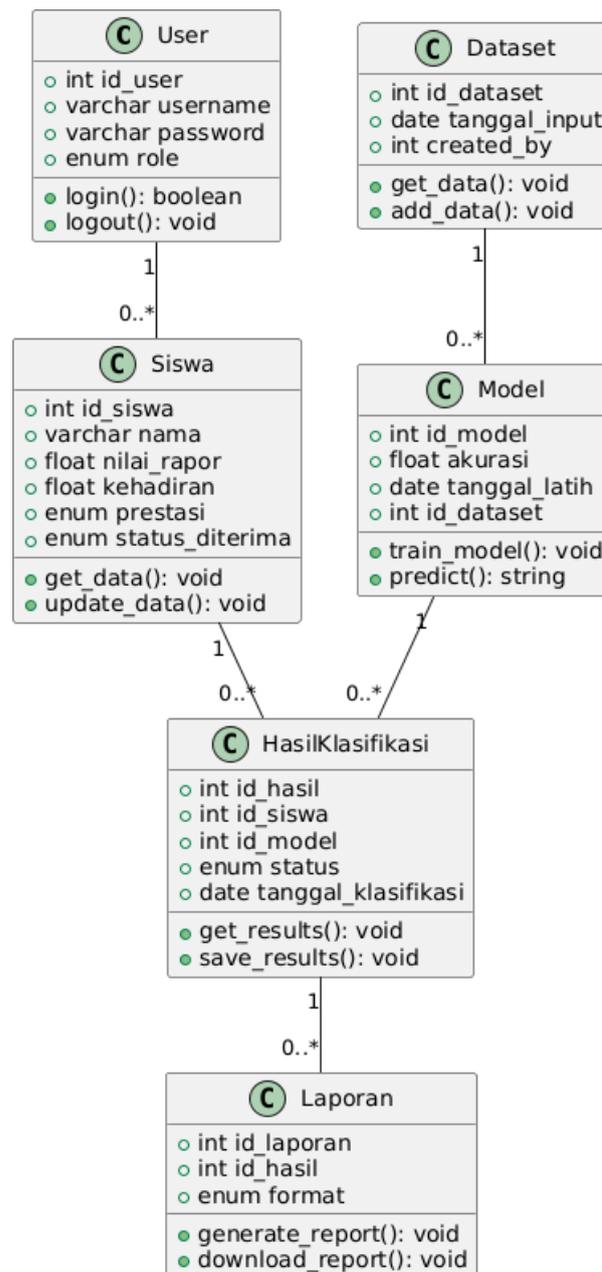
Sequence Diagram yang menggambarkan proses logout dapat dilihat pada gambar dibawah ini.



Gambar 3. 18 Sequence Diagram Logout

3.7.4 Class Diagram

Class Diagram ini menunjukkan struktur sistem klasifikasi keberhasilan siswa dengan algoritma *Gradient Boosting Machine*. dapat dilihat pada gambar dibawah ini.



Gambar 3. 19 Class Diagram

3.8 Rancangan Basis Data

1. Rancangan Tabel *Users*

Tabel ini menyimpan data pengguna sistem seperti admin atau guru.

Tabel 3. 8 Rancangan Tabel *Users*

Field	Type Data	Keterangan
id_user	INT (PK)	Primary Key, ID unik untuk setiap pengguna
username	VARCHAR(50)	Nama pengguna (unik)
password	VARCHAR(255)	Kata sandi yang di-hash
role	ENUM('admin', 'guru')	Peran pengguna (admin atau guru)

2. Rancangan Tabel Siswa

Tabel ini menyimpan data siswa yang akan diklasifikasi keberhasilannya.

Tabel 3. 9 Rancangan Tabel Siswa

Field	Type Data	Keterangan
id_siswa	INT (PK)	Primary Key, ID unik untuk setiap siswa
nama	VARCHAR(100)	Nama lengkap siswa
nilai_rapor	FLOAT	Nilai rapor rata-rata siswa
kehadiran	FLOAT	Persentase kehadiran siswa
prestasi	ENUM('Ya', 'Tidak')	Status prestasi non-akademik siswa
status_diterima	ENUM('terima', 'tidak')	Hasil klasifikasi (diterima/tidak diterima)
id_user	INT (FK)	Foreign Key yang mengacu ke <i>Users</i> (pengelola siswa)

3. Rancangan Tabel *Dataset*

Tabel ini menyimpan data Kumpulan siswa yang digunakan untuk pelatihan atau klasifikasi.

Tabel 3. 10 Rancangan Tabel *Dataset*

Field	Type Data	Keterangan
id_dataset	INT (PK)	Primary Key, ID unik untuk dataset
tanggal_input	DATE	Tanggal input dataset

created_by	INT (FK)	Foreign Key yang mengacu ke <i>Users</i> (Admin/Guru yang membuat dataset)
nama_dataset	VARCHAR(100)	Nama atau deskripsi dataset
status	ENUM('aktif', 'non-aktif')	Status dataset (aktif atau non-aktif)

4. Rancangan Tabel Model

Tabel ini menyimpan informasi tentang model *Gradient Boosting Machine* yang dilatih.

Tabel 3. 11 Rancangan Tabel Model

Field	Tipe Data	Keterangan
id_model	INT (PK)	<i>Primary Key</i>
akurasi	FLOAT	Akurasi model
tanggal_latih	DATE	Tanggal pelatihan
id_dataset	INT (FK)	<i>Dataset yang digunakan</i>

5. Rancangan Tabel Hasil Klasifikasi

Tabel ini menyimpan hasil klasifikasi dari model terhadap siswa, dapat dilihat pada tabel dibawah ini

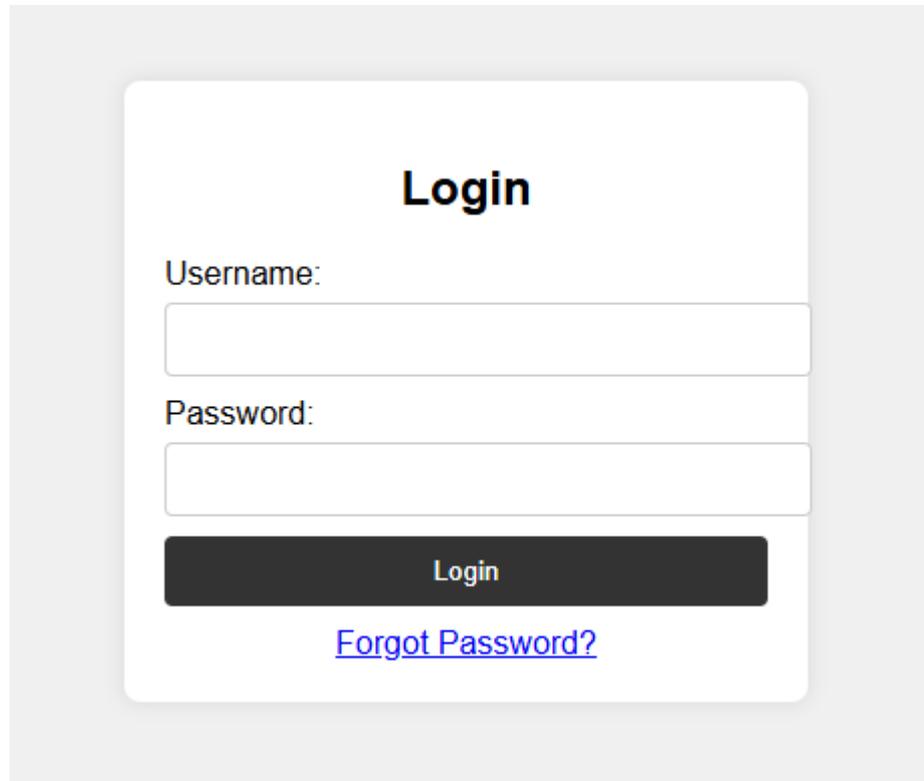
Tabel 3. 12 Rancangan Tabel Hasil Klasifikasi

Field	Tipe Data	Keterangan
id_hasil	INT (PK)	Primary Key, ID unik untuk setiap hasil klasifikasi
id_siswa	INT (FK)	Foreign Key yang mengacu ke <i>Siswa</i>
id_model	INT (FK)	Foreign Key yang mengacu ke <i>Model</i>
status	ENUM('diterima', 'tidak diterima')	Status klasifikasi (diterima/tidak diterima)
tanggal_klasifikasi	DATE	Tanggal klasifikasi dilakukan

3.9 Rancangan Antarmuka (*Interface*)

1. Desain *Form Login*

Desain untuk form *Login* pada sistem dapat dilihat pada gambar dibawah ini

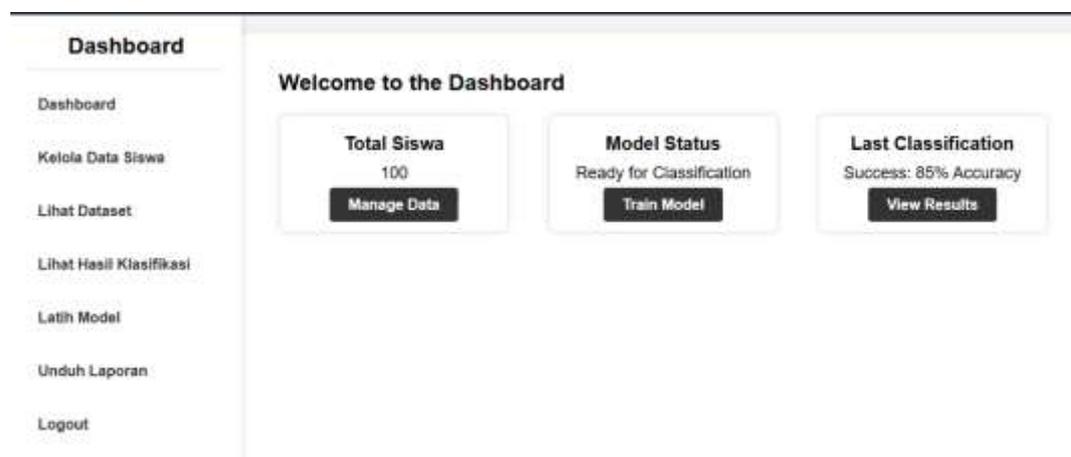


The image shows a login form with a white background and rounded corners, centered on a light gray background. At the top, the word "Login" is written in a bold, black font. Below it, there are two input fields: one for "Username:" and one for "Password:". The "Password:" field has a small eye icon to its right. Below the input fields is a dark gray button with the text "Login" in white. At the bottom of the form, there is a blue, underlined link that says "Forgot Password?".

Gambar 3. 20 Desain Form *Login*

2. Desain *Dashboard*

Desain untuk menu *Dashboard* pada sistem yang menampilkan menu-menu pada sistem dan juga statistik ringkasan dapat dilihat pada gambar dibawah ini.



The image shows a dashboard interface. On the left side, there is a vertical sidebar menu with the title "Dashboard" at the top. The menu items are: "Dashboard", "Kelola Data Siswa", "Lihat Dataset", "Lihat Hasil Klasifikasi", "Latih Model", "Unduh Laporan", and "Logout". The main content area on the right has a heading "Welcome to the Dashboard". Below this heading, there are three summary cards. The first card is titled "Total Siswa" and shows the number "100" with a "Manage Data" button below it. The second card is titled "Model Status" and shows "Ready for Classification" with a "Train Model" button below it. The third card is titled "Last Classification" and shows "Success: 85% Accuracy" with a "View Results" button below it.

Gambar 3. 21 Desain Form *Dashboard*

3. Desain *Form* Kelola Data Siswa

Desain untuk *Form* Kelola Data Siswa dapat dilihat pada gambar dibawah ini

ID	Nama	Nilai Rapor	Kehadiran	Prestasi	Status	Actions
1	John Doe	85	90%	Ya	Diterima	Edit Hapus
2	Jane Smith	78	85%	Tidak	Tidak Diterima	Edit Hapus

Gambar 3. 22 Desain Form Kelola Data Siswa

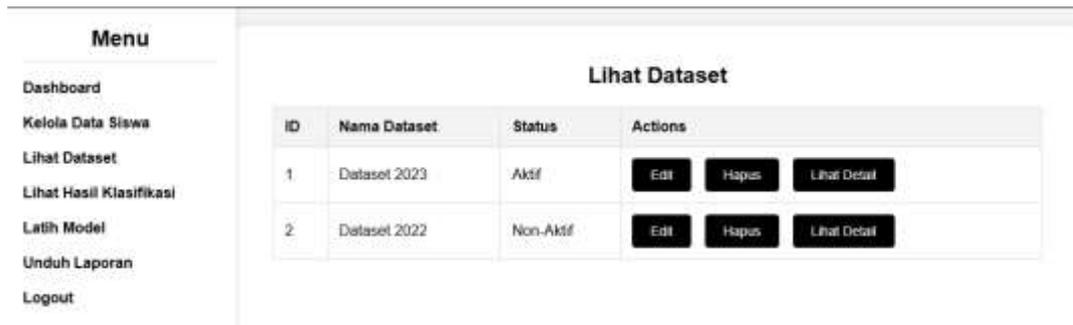
4. Desain *Form* Tambah Data Siswa

Desain untuk *Form* Tambah Data Siswa dapat dilihat pada gambar dibawah ini

Gambar 3. 23 Desain Form Tambah Data Siswa

5. Desain *Form* Lihat *Dataset*

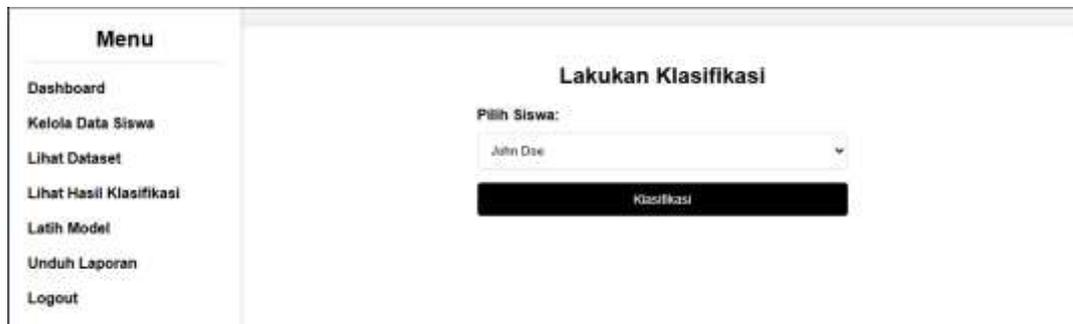
Desain Form untuk melihat *dataset* yang tersedia dapat dilihat pada gambar dibawah ini.



Gambar 3. 24 Desain Form Lihat *Dataset*

6. Desain *Form* Lakukan Klasifikasi

Desain *Form* Lakukan Klasifikasi dapat dilihat pada gambar dibawah ini.



Gambar 3. 25 Desain Form Lakukan Klasifikasi

7. Desain *Form* Lihat Hasil Klasifikasi

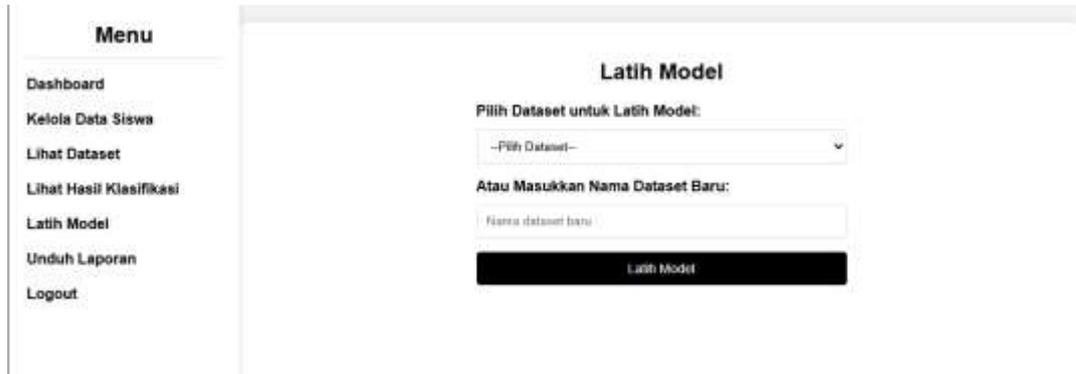
Desain *Form* untuk menu lihat hasil klasifikasi dapat dilihat pada gambar dibawah ini.



Gambar 3. 26 Desain Form Lihat Hasil Klasifikasi

8. Desain *Form* Latih Model

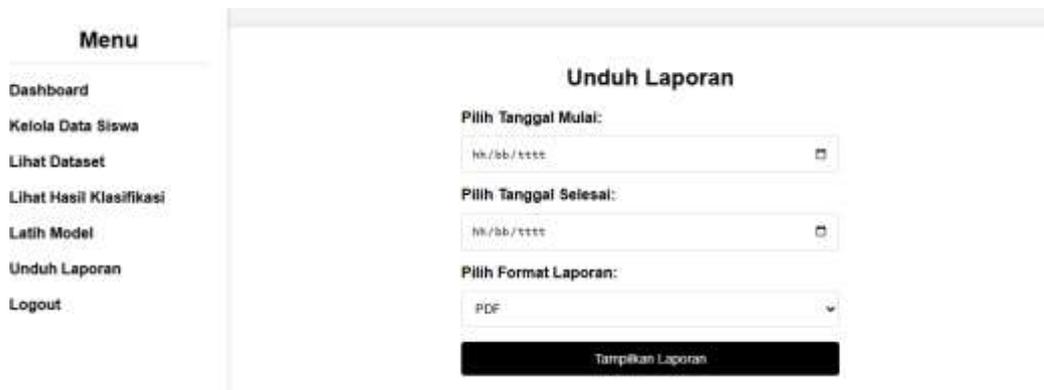
Desain *Form* untuk latih dan menyimpan model dapat dilihat pada gambar dibawah ini.



Gambar 3. 27 Desain Form Latih Model

9. Desain *Form* Unduh Laporan

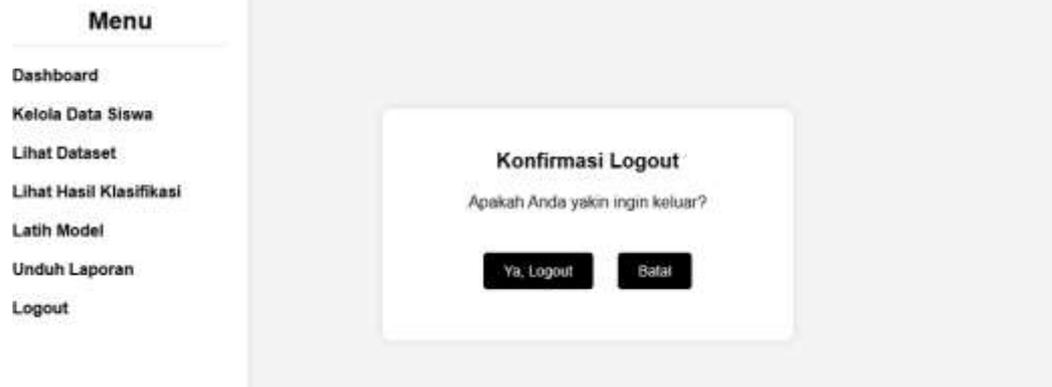
Desain *Form* Unduh laporan dapat dilihat pada gambar dibawah ini



Gambar 3. 28 Desain Form Laporan

10. Desain *Form* Logout

Desain *Form* untuk notifikasi logout dapat dilihat pada gambar dibawah ini



Gambar 3. 29 Desain Form Konfirmasi Logout

BAB IV

HASIL DAN PEMBAHASAN

4.1. Perhitungan Manual *Gradient Boosting Machine*

Pada sub-bab ini, dijelaskan proses pemilihan subset data yang akan digunakan sebagai studi kasus untuk perhitungan manual. Pemilihan subset data ini bertujuan untuk menyederhanakan proses perhitungan agar dapat diikuti dengan mudah, namun tetap representatif terhadap keseluruhan dataset yang dimiliki.

Dari total 40 data siswa yang tersedia, dipilih 10 data secara purposif untuk simulasi. Pemilihan ini didasarkan pada keterwakilan kelas target, di mana 5 siswa memiliki status "Diterima" (Status = 1) dan 5 siswa lainnya memiliki status "Tidak Diterima" (Status = 0). Kriteria ini memastikan bahwa proses perhitungan dapat menggambarkan bagaimana model belajar membedakan kedua kelas tersebut secara seimbang. Selain keseimbangan kelas, subset data yang dipilih juga mengandung variasi nilai yang signifikan pada setiap atribut. Sebagai contoh, Skor_Prestasi memiliki rentang dari 0 hingga 40, dan Nilai_Rapor_Rata berkisar dari 85.5 hingga 95.0. Variasi ini esensial untuk mendemonstrasikan bagaimana pohon keputusan (*decision tree*), sebagai *weak learner* dalam GBM, mampu mengidentifikasi titik pemisah (*split point*) yang bermakna untuk memisahkan data. Apabila data yang digunakan terlalu homogen, maka proses pembelajaran algoritma akan menjadi kurang ilustratif. Subset data yang digunakan disajikan pada Tabel 4.1.

Tabel 4. 1 *Subset Data Untuk Perhitungan Manual*

ID	Nilai Rapor Rata	Skor Prestasi	Persentase Kehadiran	Status (y)
1	92.5	30	98	1
2	88.0	0	95	0
3	90.2	10	97	1
4	85.5	0	92	0
5	95.0	40	99	1
6	89.0	5	94	0
7	91.5	20	96	1
8	86.2	0	90	0
9	93.8	25	97	1
10	87.5	0	93	0

Langkah pertama dalam GBM adalah membuat prediksi awal yang seragam untuk semua observasi. Untuk masalah klasifikasi, prediksi awal ini tidak dinyatakan sebagai probabilitas, melainkan sebagai nilai dalam ruang *log-odds*. Pendekatan ini merupakan fondasi matematis dari GBM untuk klasifikasi. GBM secara aditif membangun model pada skala log-odds, di mana setiap pohon keputusan berikutnya akan menambah atau mengurangi nilai ini. Skala log-odds memungkinkan pembaruan model yang tidak terbatas antara 0 dan 1, sehingga secara matematis lebih stabil dan fleksibel. Probabilitas adalah hasil akhir dari transformasi, bukan unit dasar dari proses pembelajaran.

Model diinisialisasi dengan logaritma dari *odds*. *Odds* dihitung sebagai rasio jumlah observasi kelas positif ($y=1$) terhadap jumlah observasi kelas negatif ($y=0$). Berdasarkan Tabel 4.1, terdapat 5 siswa dengan status diterima ($y=1$) dan 5 siswa dengan status tidak diterima ($y=0$). Prediksi awal $F_0(x)$ dihitung menggunakan rumus berikut:

$$F_0(x) = \log(\text{odds}) = \log\left(\frac{\sum_{i=1}^N y_i}{N - \sum_{i=1}^N y_i}\right)$$

$$F_0(x) = \log\left(\frac{5}{5}\right) = \log(1) = 0$$

Nilai prediksi awal $F_0(x)=0$ mengindikasikan bahwa, tanpa informasi dari atribut manapun, model mengasumsikan setiap siswa memiliki peluang yang sama untuk diterima atau tidak diterima. Untuk menghitung residual pada langkah berikutnya, nilai log-odds ini perlu dikonversi menjadi probabilitas (p_0) menggunakan fungsi logistik (sigmoid).

$$p = \frac{1}{1 + e^{-(\log\text{-odds})}}$$

$$p_0 = \frac{1}{1 + e^{-0}} = \frac{1}{1 + 1} = 0.5$$

Dengan demikian, probabilitas awal untuk setiap siswa adalah 0.5, yang mencerminkan ketidakpastian model pada tahap inisialisasi.

Pada iterasi pertama, model mulai belajar dari kesalahan prediksi awal. Proses ini melibatkan empat langkah utama: menghitung residual, membangun pohon keputusan, menghitung nilai output untuk setiap daun pohon, dan memperbarui model.

Langkah 1: Perhitungan Pseudo-Residual (ri1)

Residual dalam klasifikasi GBM, yang juga dikenal sebagai *pseudo-residual*, adalah selisih antara nilai target aktual (y_i , yaitu 0 atau 1) dan probabilitas yang diprediksi oleh model sebelumnya (p_0). Residual ini secara matematis merepresentasikan gradien dari fungsi kerugian *Log-Loss* dan menunjukkan arah

serta besaran kesalahan prediksi model. Rumus untuk menghitung pseudo-residual pada iterasi pertama adalah:

$$r_{i1} = y_i - p_0$$

Perhitungan residual untuk setiap siswa pada iterasi pertama disajikan pada Tabel 4.2.

Tabel 4. 2 Tabel Perhitungan Residual

ID	Status (y_i)	Prediksi Probabilitas Awal (p_0)	Pseudo-Residual (r_{i1})
1	1	0.5	0.5
2	0	0.5	-0.5
3	1	0.5	0.5
4	0	0.5	-0.5
5	1	0.5	0.5
6	0	0.5	-0.5
7	1	0.5	0.5
8	0	0.5	-0.5
9	1	0.5	0.5
10	0	0.5	-0.5

Sebuah model *decision tree regressor* sederhana dilatih dengan menggunakan atribut Nilai_Rapor_Rata, Skor_Prestasi, dan Persentase_Kehadiran sebagai variabel independen dan *pseudo-residual* (r_{i1}) sebagai variabel target. Konsep inti dari *boosting* adalah bahwa pohon keputusan ini tidak mencoba memprediksi hasil akhir ($y=1$ atau $y=0$), melainkan mencoba memprediksi *kesalahan* dari model sebelumnya (r_{i1}). Dengan kata lain, pohon ini belajar untuk mengidentifikasi pola pada data yang menyebabkan model awal membuat prediksi yang salah.

Untuk tujuan perhitungan manual, dibangun sebuah pohon keputusan yang sangat sederhana. Berdasarkan analisis data pada Tabel 4.1 dan Tabel 4.2, terlihat bahwa atribut Skor_Prestasi dapat memisahkan data dengan sempurna berdasarkan

nilai residualnya. Semua siswa dengan $r_{i1}=0.5$ memiliki Skor_Prestasi yang lebih besar dari 5, dan semua siswa dengan $r_{i1}=-0.5$ memiliki Skor_Prestasi kurang dari atau sama dengan 5. Dengan demikian, aturan pemisah (*split rule*) yang dapat digunakan adalah:

- Jika Skor_Prestasi ≤ 7.5 , maka data masuk ke daun kiri.
- Jika Skor_Prestasi > 7.5 , maka data masuk ke daun kanan.

Berdasarkan aturan ini, siswa dikelompokkan sebagai berikut:

- **Daun Kiri:** Siswa dengan ID 2, 4, 6, 8, 10 (semua memiliki residual -0.5).
- **Daun Kanan:** Siswa dengan ID 1, 3, 5, 7, 9 (semua memiliki residual 0.5)

Untuk setiap daun (*terminal node*) dari pohon keputusan, dihitung sebuah nilai output baru (γ). Dalam klasifikasi GBM, formula ini lebih kompleks daripada sekadar menghitung rata-rata residual. Tujuannya adalah untuk menstabilkan pembaruan model. Hasil perhitungan nilai output untuk setiap daun disajikan pada Tabel 4.3.

Tabel 4.3 Nilai Output Terminal Node Iterasi Pertama

Siswa Masuk ke Daun	Nilai Output (γ_1)
Daun Kanan (Skor_Prestasi > 7.5)	2.0
Daun Kiri (Skor_Prestasi ≤ 7.5)	-2.0

Model diperbarui dengan menambahkan kontribusi dari pohon baru, yang telah diskalakan dengan *learning rate* ($\eta=0.1$). Pembaruan ini terjadi pada skala log-odds. Rumus pembaruan model adalah:

$$F_1(x) = F_0(x) + \eta \times \gamma_1$$

Tabel 4. 4 Hasil Pembaruan Prediksi Iterasi Pertama

ID Siswa	Prediksi Awal ($F_0(x)$)	Kontribusi Pohon ($\eta \times \gamma_i$)	Prediksi Baru ($F_1(x)$)	Probabilitas Baru (p_i)
1, 3, 5, 7, 9	0	$0.1 \times 2.0 = 0.2$	0.2	0.5498
2, 4, 6, 8, 10	0	$0.1 \times (-2.0) = -0.2$	-0.2	0.4502

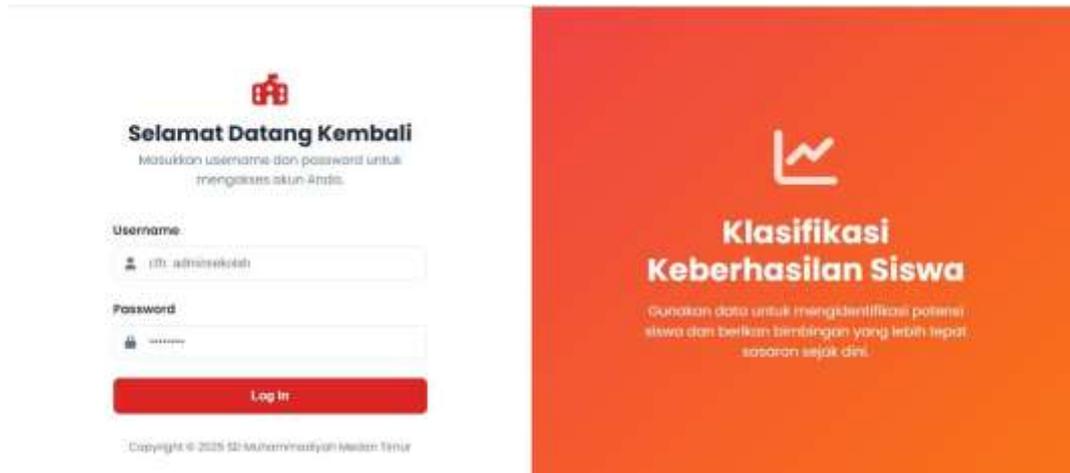
Setelah iterasi pertama, model tidak lagi memberikan prediksi 0.5 untuk semua siswa. Prediksi probabilitas untuk siswa yang sebenarnya diterima (Status = 1) telah meningkat menjadi ~ 0.55 , sementara untuk siswa yang tidak diterima (Status = 0) telah menurun menjadi ~ 0.45 .

Proses yang diuraikan di atas untuk iterasi pertama akan diulang kembali untuk iterasi-iterasi berikutnya.

4.2. Hasil Tampilan Aplikasi Berbasis Web

1. Tampilan Halaman *Login*

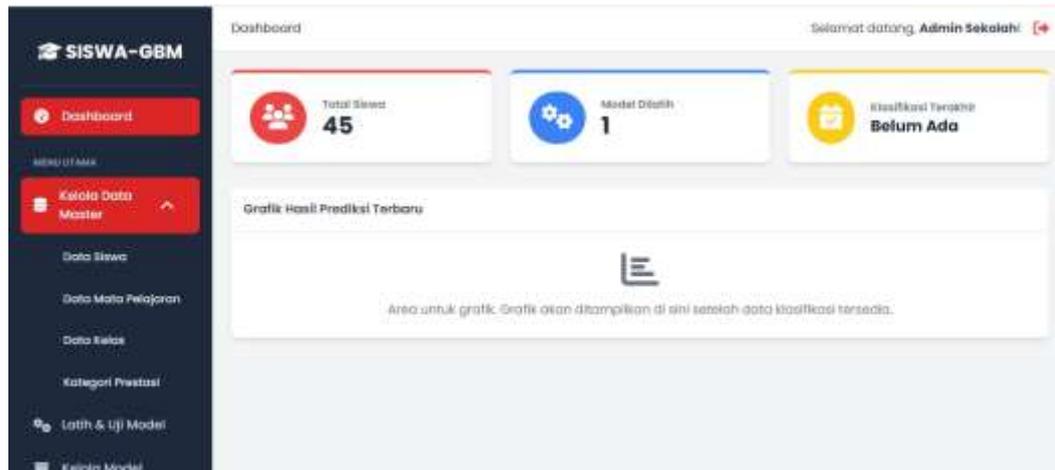
Halaman ini merupakan gerbang utama untuk mengakses sistem klasifikasi. Pengguna, seperti admin sekolah, harus memasukkan *username* dan *password* yang valid untuk dapat masuk. Halaman ini dirancang sederhana dengan fokus pada keamanan dan kemudahan akses. Di sisi kanan, terdapat pesan sambutan dan penjelasan singkat mengenai fungsi utama sistem, yaitu untuk mengidentifikasi potensi siswa menggunakan data.



Gambar 4. 1 Tampilan Halaman *Login*

2. Tampilan Halaman *Dashboard*

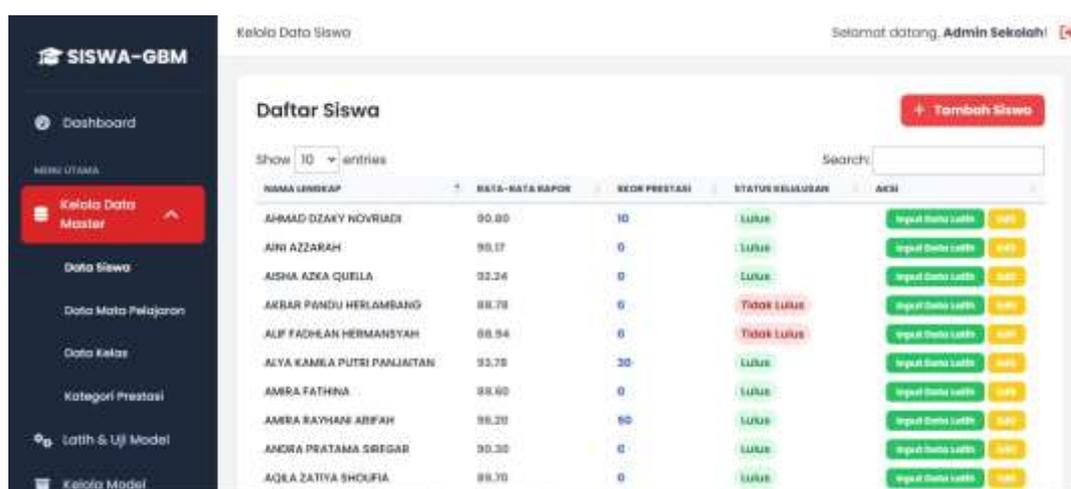
Dashboard adalah halaman pertama yang dilihat pengguna setelah berhasil login. Halaman ini menyajikan ringkasan informasi penting secara visual melalui panel-panel statistik, seperti Total Siswa, jumlah Model Dilatih, dan status Klasifikasi Terakhir. Terdapat juga area kosong yang nantinya akan menampilkan grafik hasil prediksi terbaru setelah data klasifikasi tersedia. Menu navigasi utama terletak di sisi kiri, memberikan akses mudah ke semua fitur sistem seperti Kelola Data, Latih & Uji Model, dan lainnya.



Gambar 4. 2 Tampilan Halaman *Dashboard*

3. Tampilan Halaman Kelola Data Siswa

Halaman ini berfungsi sebagai pusat pengelolaan data siswa. Admin dapat melihat daftar lengkap siswa beserta informasi kunci seperti nama, rata-rata rapor, skor prestasi, dan status kelulusan. Terdapat fitur pencarian (search) untuk menemukan siswa secara cepat dan fungsi untuk mengatur jumlah data yang ditampilkan per halaman. Setiap baris data siswa dilengkapi dengan tombol aksi ("Input Data Latih" dan "Edit") yang memungkinkan admin untuk memperbarui informasi atau memasukkan data siswa ke dalam dataset pelatihan model.



Gambar 4. 3 Tampilan Halaman Kelola Data Siswa

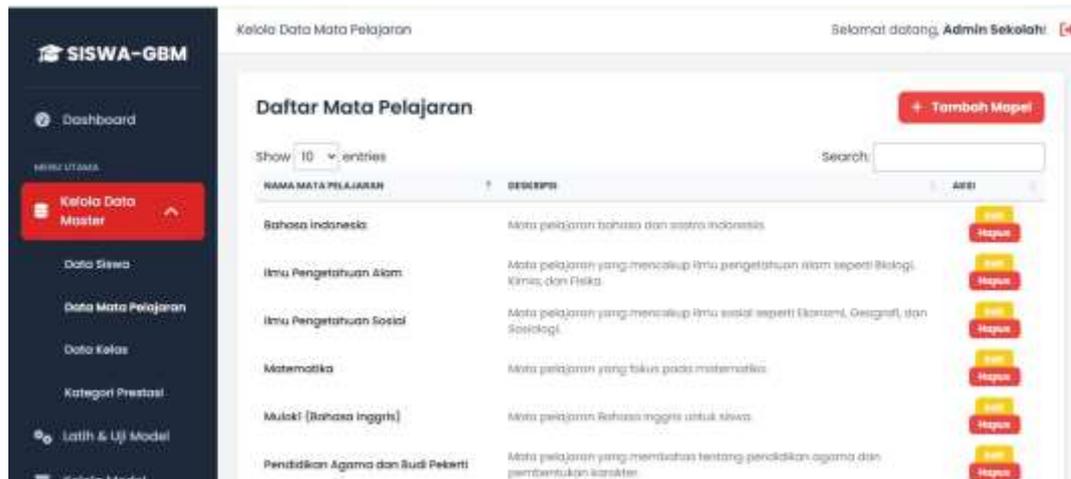
4. Tampilan Halaman Tambah Data Siswa

Halaman ini menyediakan formulir untuk menambahkan data siswa baru ke dalam sistem. Admin dapat mengisi informasi detail siswa, termasuk Nama Lengkap, Tempat Lahir, NIS/NISN, Tanggal Lahir, Kelas, Jenis Kelamin, dan Alamat. Formulir ini dirancang untuk memastikan kelengkapan data yang diperlukan untuk proses analisis dan klasifikasi. Setelah semua kolom diisi, data dapat disimpan ke dalam database dengan menekan tombol "Simpan Data Siswa".

Gambar 4. 4 Tampilan Halaman Tambah Data Siswa

5. Tampilan Halaman Kelola Data Mata Pelajaran

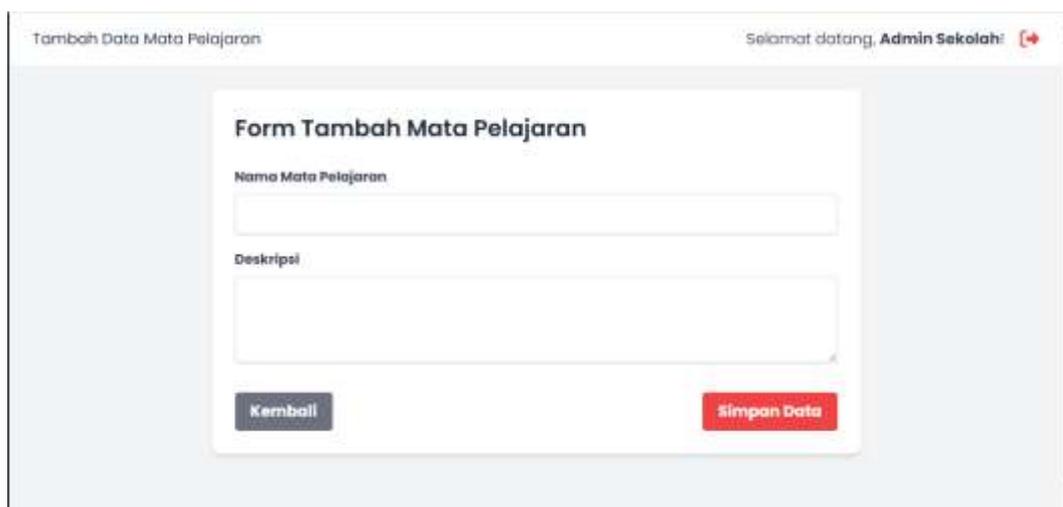
Halaman ini memungkinkan admin untuk mengelola daftar mata pelajaran yang ada di sekolah. Informasi yang ditampilkan meliputi nama mata pelajaran dan deskripsi singkatnya. Halaman ini dilengkapi fitur pencarian untuk memudahkan admin menemukan mata pelajaran tertentu. Untuk setiap mata pelajaran, tersedia tombol aksi "Edit" dan "Hapus", yang memberikan fleksibilitas bagi admin untuk memperbarui atau menghapus data sesuai kebutuhan kurikulum.



Gambar 4. 5 Tampilan Kelola Data Mapel

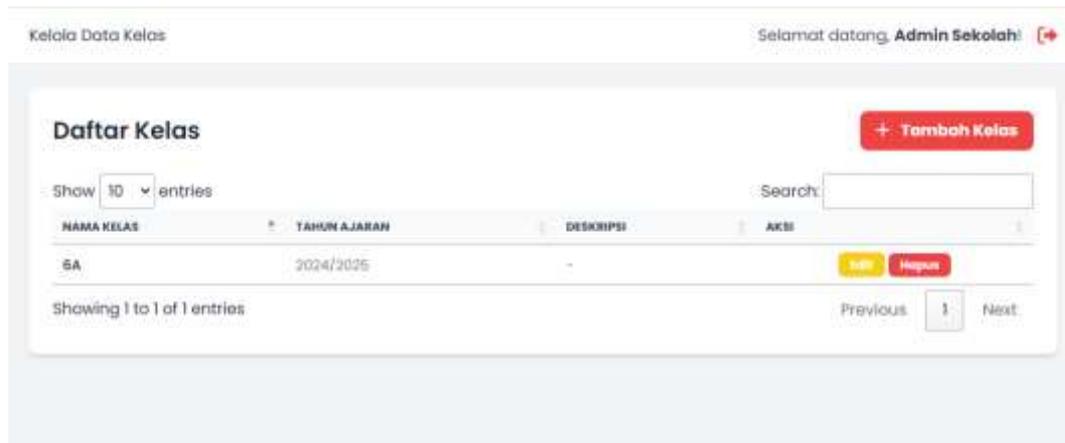
6. Tampilan Halaman Tambah Data Mata Pelajaran

Halaman Ini adalah halaman formulir yang digunakan untuk menambahkan mata pelajaran baru ke dalam sistem. Admin perlu memasukkan Nama Mata Pelajaran dan Deskripsi singkat mengenai mata pelajaran tersebut. Setelah informasi diisi, admin dapat menyimpannya dengan mengklik tombol "Simpan Data". Halaman ini penting untuk menjaga agar daftar mata pelajaran di sistem selalu relevan dengan kurikulum yang berlaku.



Gambar 4. 6 Tampilan Tambah Data Mapel

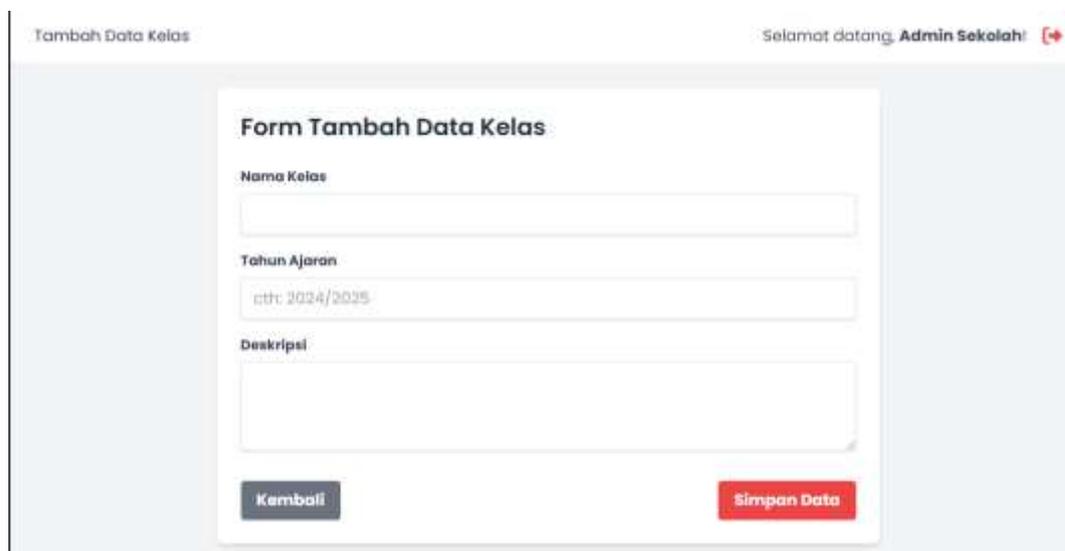
7. Tampilan Halaman Kelola Data Kelas



Gambar 4. 7 Tampilan Kelola Data Kelas

8. Tampilan Halaman Tambah Data Kelas

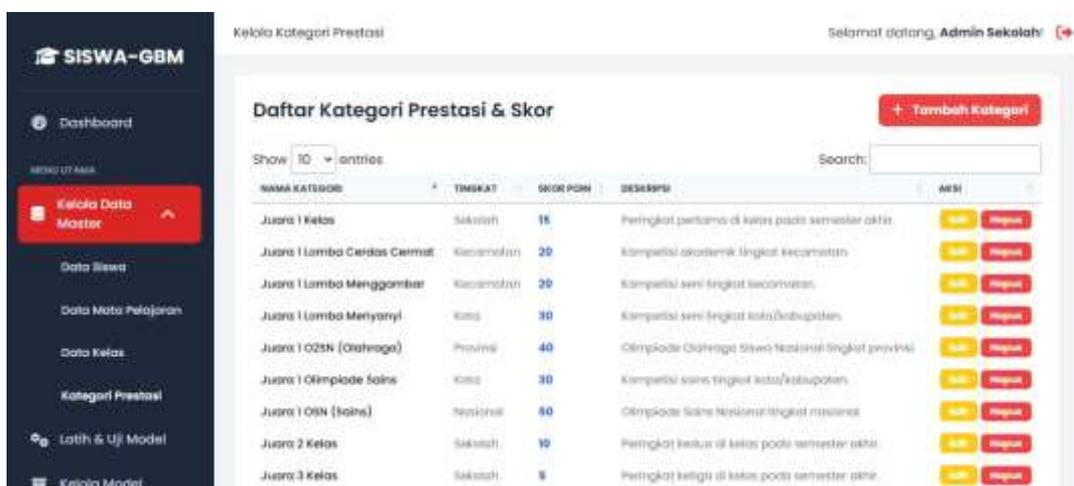
Halaman ini berfungsi untuk mengelola data kelas yang ada di sekolah. Dalam bentuk tabel, halaman ini menampilkan informasi seperti Nama Kelas, Tahun Ajaran, dan Deskripsi. Admin dapat dengan mudah melihat kelas-kelas yang aktif pada tahun ajaran tertentu. Setiap data kelas dilengkapi dengan tombol aksi "Edit" dan "Hapus" untuk memperbarui atau menghapus data kelas jika diperlukan.



Gambar 4. 8 Tampilan Halaman Tambah Data Kelas

9. Tampilan Halaman Kelola Kategori Prestasi

Halaman Kelola Kategori Prestasi berfungsi sebagai pusat pengaturan berbagai jenis prestasi siswa beserta skornya. Admin dapat melihat daftar lengkap prestasi yang sudah ada, seperti "Juara 1 Kelas" atau "Juara 1 OSN (Sains)," beserta tingkat (misalnya, Sekolah, Kecamatan, Nasional) dan skor poin yang sesuai. Halaman ini dilengkapi fitur pencarian untuk memudahkan pengelolaan dan tombol aksi "Edit" serta "Hapus" untuk setiap kategori, memungkinkan admin memperbarui atau menghapus data prestasi dengan fleksibel.



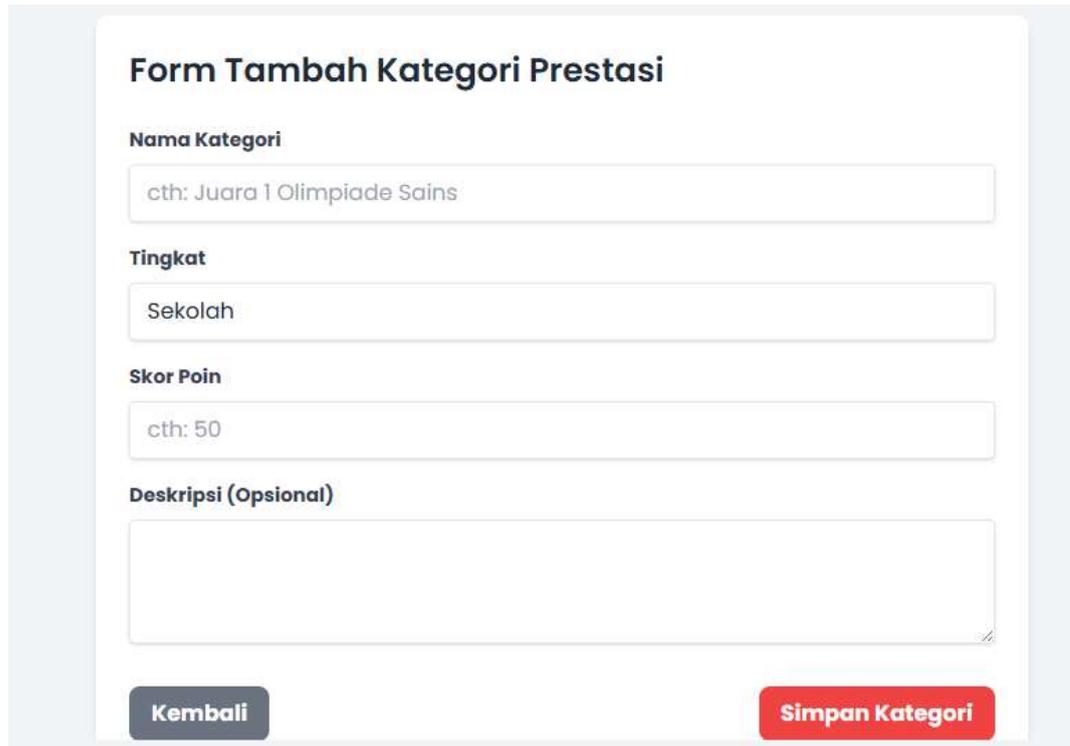
The screenshot shows a web application interface for managing achievement categories. The main content area is titled 'Daftar Kategori Prestasi & Skor' and contains a table with the following data:

NAMA KATEGORI	TINGKAT	SKOR POIN	DESKRIPSI	Aksi
Juara 1 Kelas	Sekolah	15	Pemegang pertama di kelas pada semester aktif	[Edit] [Hapus]
Juara 1 Lomba Cerdas Cermat	Kecamatan	20	Kompetisi akademik tingkat kecamatan	[Edit] [Hapus]
Juara 1 Lomba Menggambar	Kecamatan	20	Kompetisi seni tingkat kecamatan	[Edit] [Hapus]
Juara 1 Lomba Menyanyi	Kota	30	Kompetisi seni tingkat kota/kabupaten	[Edit] [Hapus]
Juara 1 OSN (Olimpiade)	Provinsi	40	Olimpiade Olahraga Siswa Nasional tingkat provinsi	[Edit] [Hapus]
Juara 1 Olimpiade Sains	Kota	30	Kompetisi sains tingkat kota/kabupaten	[Edit] [Hapus]
Juara 1 OSN (Sains)	Nasional	50	Olimpiade Sains Nasional tingkat nasional	[Edit] [Hapus]
Juara 2 Kelas	Sekolah	10	Pemegang kedua di kelas pada semester aktif	[Edit] [Hapus]
Juara 3 Kelas	Sekolah	5	Pemegang ketiga di kelas pada semester aktif	[Edit] [Hapus]

Gambar 4. 9 Tamplan Halaman Kategori Prestasi

10. Tampilan Halaman Tambah Kategori Prestasi

Halaman ini menyediakan formulir khusus untuk menambahkan kategori prestasi baru ke dalam sistem. Admin dapat mengisi detail prestasi, meliputi Nama Kategori, Tingkat, dan Skor Poin. Terdapat juga kolom Deskripsi yang bersifat opsional untuk memberikan penjelasan lebih lanjut. Setelah semua informasi yang diperlukan diisi, admin dapat menyimpannya dengan menekan tombol "Simpan Kategori," yang akan langsung menambahkan prestasi baru tersebut ke dalam daftar yang dapat digunakan untuk penilaian siswa.

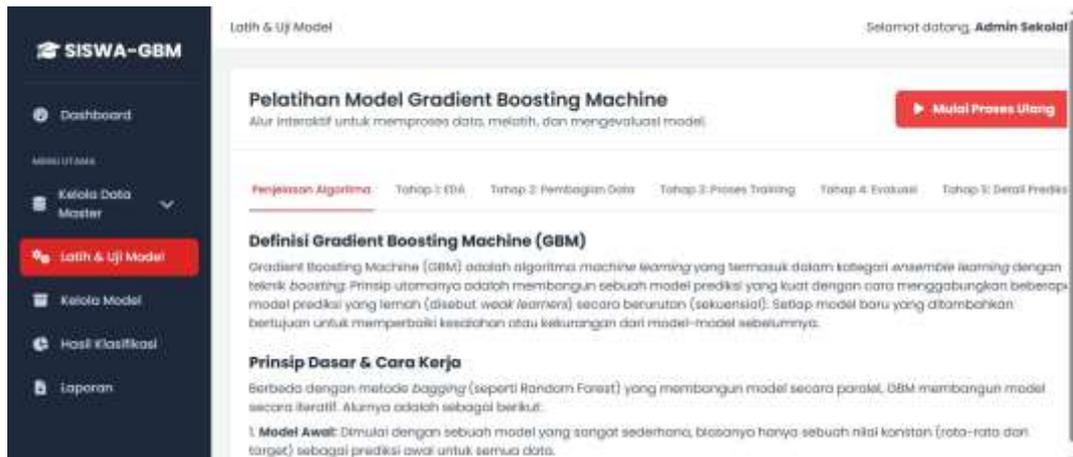


The image shows a web form titled "Form Tambah Kategori Prestasi". It contains four input fields and two buttons. The first field is labeled "Nama Kategori" and contains the text "cth: Juara 1 Olimpiade Sains". The second field is labeled "Tingkat" and contains "Sekolah". The third field is labeled "Skor Poin" and contains "cth: 50". The fourth field is labeled "Deskripsi (Opsional)" and is empty. At the bottom left is a grey button labeled "Kembali" (Back), and at the bottom right is a red button labeled "Simpan Kategori" (Save Category).

Gambar 4. 10 Tampilan Halaman Tambah Kategori Prestasi

11. Tampilan Halaman Latih dan Uji Model

Halaman Latih & Uji Model adalah pusat untuk memproses, melatih, dan mengevaluasi model klasifikasi. Tab pertama, "Penjelasan Algoritma," memberikan pemahaman dasar mengenai Gradient Boosting Machine (GBM). Di sini dijelaskan bahwa GBM adalah algoritma ensemble learning yang membangun model secara berurutan (boosting) untuk memperbaiki kesalahan dari model sebelumnya. Halaman ini juga menguraikan prinsip dasar dan cara kerja GBM, dimulai dari model awal yang sederhana hingga menjadi model prediksi yang kuat dan akurat. Tombol "Mulai Proses Ulang" tersedia untuk memulai siklus pelatihan model dari awal.



Gambar 4. 11 Penjelasan Algoritma

"Tahap 1: EDA" (Exploratory Data Analysis) menampilkan analisis awal dari dataset yang digunakan untuk pelatihan. Halaman ini menginformasikan bahwa dataset terdiri dari 45 total baris data latih yang valid. Dijelaskan pula fitur (X) yang digunakan, yaitu nilai_rapor_rata, total_skor_prestasi, dan persentase_kehadiran, serta target (y) prediksi, yaitu status_kelulusan. Di bagian bawah, disajikan Statistik Deskriptif Fitur yang mencakup nilai count, mean, std, min, dan max untuk setiap fitur, memberikan gambaran umum tentang distribusi dan rentang data.

Statistik Deskriptif Fitur:			
	nilai_rapor_rata	total_skor_prestasi	persentase_kehadiran
count	45.000000	45.000000	45.000000
mean	89.827111	7.666667	95.148889
std	2.356997	33.84637	3.742788
min	85.040000	0.000000	85.000000
25%	88.230000	0.000000	92.800000
50%	88.620000	0.000000	96.000000
75%	89.010000	10.000000	98.100000
max	88.200000	50.000000	100.000000

Gambar 4. 12 Tahap EDA dalam Latih dan Uji Model

Tab "Tahap 2: Pembagian Data" menjelaskan bagaimana dataset dibagi untuk proses pelatihan dan pengujian model. Secara spesifik, dataset bersih dibagi menjadi 80% Data Latih dan 20% Data Uji. Terdapat Tabel Distribusi Label yang

merinci jumlah data untuk kategori "Lulus" (1) dan "Tidak Lulus" (0) pada masing-masing set data. Untuk data latih, terdapat 18 data lulus dan 18 data tidak lulus, sedangkan pada data uji terdapat 5 data lulus dan 4 data tidak lulus. Pembagian ini penting untuk memastikan model dievaluasi secara objektif menggunakan data yang belum pernah dilihat sebelumnya.



Dataset	Jumlah Lulus (1)	Jumlah Tidak Lulus (0)	Total
Data Latih	18	18	36
Data Uji	5	4	9

Gambar 4. 13 Tampilan Pembagian Data

Tab "Tahap 3: Proses Training" menampilkan langkah-langkah detail bagaimana model Gradient Boosting Machine (GBM) dilatih. Proses ini diawali dengan Langkah Inisialisasi, di mana model membuat "tebakan awal" (F_0) yang seragam untuk semua data. Tebakan ini dihitung berdasarkan log-odds dari probabilitas rata-rata kelas 'Lulus' pada data latih, yang dalam kasus ini adalah 0.5000. Dari sini, dihitung nilai *pseudo-residual* atau kesalahan awal untuk setiap data, yaitu selisih antara nilai aktual (1 untuk Lulus, 0 untuk Tidak Lulus) dan probabilitas awal.

Pelatihan Model Gradient Boosting Machine
Alur interaktif untuk memproses data, melatih, dan mengevaluasi model.

► Mulai Proses Ulang

Penjelasan Algoritma Tahap 1: EDA Tahap 2: Pembagian Data **Tahap 3: Proses Training** Tahap 4: Evaluasi Tahap 5: Detail Prediksi

Langkah Inisialisasi: Membuat Tebakan Awal (F_0)
Model dimulai dengan 'tebakan awal' yang sama untuk semua data, yaitu **log-odds** dari probabilitas rata-rata.

- Probabilitas rata-rata 'Lulus' (\bar{y}) di data latih: **0.5000**
- Perhitungan Log-Odds: $F_0(x) = \log\left(\frac{0.5000}{1-0.5000}\right) = 0.0000$

Langkah Berikutnya: Apa Kesalahan dari Tebakan Awal Ini?
Sekarang kita hitung 'kesalahan' atau **pseudo-residual** dari tebakan awal ini. Kesalahan inilah yang akan coba diperbaiki oleh pohon pertama.

- Rumus Probabilitas dari Log-Odds: $p_0 = \frac{1}{1+e^{-\eta_0}} = 0.5000$
- Rumus Residual: $r_i = y_i - p_0$ (Aktual - Prediksi Probabilitas)
- Contoh residual untuk siswa yang aktualnya **Lulus ($y=1$)**: $r_i = 1 - 0.5000 = 0.5000$
- Contoh residual untuk siswa yang aktualnya **Tidak Lulus ($y=0$)**: $r_i = 0 - 0.5000 = -0.5000$

Gambar 4. 14 Tampilan Proses Training

Tab "Tahap 4: Evaluasi" menyajikan hasil performa model setelah dilatih. Halaman ini akan menampilkan metrik-metrik evaluasi kunci seperti Akurasi, Presisi, Recall, dan F1-Score. Selain itu, ditampilkan juga *Confusion Matrix* yang memberikan rincian visual mengenai jumlah prediksi yang benar (True Positives dan True Negatives) dan yang salah (False Positives dan False Negatives). Evaluasi ini sangat penting untuk mengukur seberapa baik model dapat menggeneralisasi dan membuat prediksi yang akurat pada data yang belum pernah dilihat sebelumnya.

Pelatihan Model Gradient Boosting Machine
Alur interaktif untuk memproses data, melatih, dan mengevaluasi model.

► Mulai Proses Ulang

Penjelasan Algoritma Tahap 1: EDA Tahap 2: Pembagian Data Tahap 3: Proses Training **Tahap 4: Evaluasi** Tahap 5: Detail Prediksi

Model diuji pada **Data Uji** untuk mengukur performa pada data baru.

Diagram Confusion Matrix

	Prediksi Tidak Lulus (0)	Prediksi Lulus (1)
Aktual Tidak Lulus (0)	4	0
Aktual Lulus (1)	0	5

Metrik Kinerja

Akurasi:	100.00%
Presisi:	100.00%
Recall:	100.00%
F1-Score:	100.00%

Gambar 4. 15 Tampilan Evaluasi Model

Tab "Tahap 5: Detail Prediksi" menampilkan tabel perbandingan antara hasil prediksi model dengan data aktual untuk setiap siswa dalam data uji. Setiap baris data siswa ditampilkan beserta fitur-fiturnya, probabilitas lulus yang dihasilkan model, prediksi akhir (Lulus/Tidak Lulus), dan status aktual. Untuk mempermudah analisis, baris yang ditandai hijau menunjukkan prediksi yang benar, sementara baris yang ditandai merah (jika ada) menunjukkan prediksi yang salah. Ini memberikan gambaran transparan tentang bagaimana model mengambil keputusan untuk setiap individu.

Pelatihan Model Gradient Boosting Machine
Alur Interaktif untuk memproses data, melatih, dan mengevaluasi model.

Penjelasan Algoritma Tahap 1: EDA Tahap 2: Pembagian Data Tahap 3: Proses Training Tahap 4: Evaluasi Tahap 5: Detail Prediksi

Tabel perbandingan antara hasil prediksi model dengan data aktual. Baris hijau menandakan prediksi benar, baris merah menandakan prediksi salah.

	nilai_rapor_rata	total_skor_prestasi	persentase_kehadiran	Probabilitas_Lulus	Prediksi	Aktual
41	91.550000	15	99.800000	0.605100	Lulus	Lulus
35	88.840000	0	82.000000	0.394900	Tidak Lulus	Tidak Lulus
22	98.100000	0	82.800000	0.394900	Tidak Lulus	Tidak Lulus
18	85.040000	0	85.000000	0.394900	Tidak Lulus	Tidak Lulus
3	98.200000	50	98.500000	0.605100	Lulus	Lulus
42	91.010000	5	100.000000	0.605100	Lulus	Lulus
2	83.780000	30	100.000000	0.605100	Lulus	Lulus
27	89.600000	0	96.000000	0.500200	Lulus	Lulus
13	88.010000	0	84.000000	0.394900	Tidak Lulus	Tidak Lulus

Gambar 4. 16 Tampilan Detail Prediksi

12. Tampilan Halaman Kelola Model

Halaman Kelola Model berfungsi sebagai repositori untuk model-model *machine learning* yang telah dilatih dan disimpan dalam sistem. Pada halaman ini, admin dapat melihat daftar semua model yang tersedia, beserta informasi esensial seperti nama model, tingkat akurasi yang dicapai, dan tanggal pelatihan. Setiap model memiliki status, seperti "Aktif", yang menandakan model tersebut sedang digunakan untuk proses klasifikasi. Selain itu, terdapat serangkaian tombol aksi yang memungkinkan admin untuk melihat detail performa model, menonaktifkan

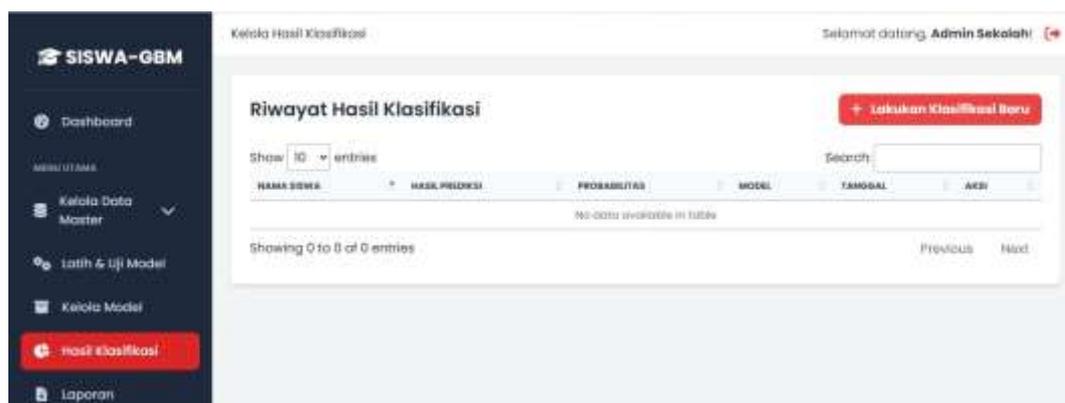
model jika tidak lagi digunakan, atau menghapusnya secara permanen dari sistem. Halaman ini memastikan bahwa manajemen siklus hidup model dapat dilakukan secara efisien.



Gambar 4. 17 Tampilan Halaman Kelola Model

13. Tampilan Halaman Kelola Klasifikasi

Halaman Hasil Klasifikasi menyajikan riwayat dari seluruh proses klasifikasi yang telah dilakukan terhadap data siswa. Halaman ini dirancang untuk menampilkan tabel yang berisi informasi rinci, mencakup nama siswa yang diklasifikasi, hasil prediksi (misalnya, Lulus atau Tidak Lulus), nilai probabilitas yang dihasilkan model, model yang digunakan, serta tanggal klasifikasi dilakukan.



Gambar 4. 18 Tampilan Halaman Kelola Klasifikasi

4.3. Pengujian Program

Pengujian program dilakukan untuk memastikan bahwa semua fungsionalitas sistem klasifikasi keberhasilan siswa berbasis web ini berjalan sesuai dengan yang diharapkan. Pengujian mencakup serangkaian skenario yang menguji setiap fitur utama, mulai dari proses login, pengelolaan data master, pelatihan model, hingga proses klasifikasi dan pembuatan laporan. Metode pengujian yang digunakan adalah *black-box testing*, di mana penguji berinteraksi dengan antarmuka pengguna tanpa perlu mengetahui struktur kode internalnya, untuk memverifikasi bahwa setiap input menghasilkan output yang sesuai.

Tabel 4. 5 Tabel Pengujian Program

No .	Skenario Pengujian	Langkah-Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
1	Login Pengguna	1. Buka halaman login.2. Masukkan username dan password yang valid.3. Klik tombol Log In .	Pengguna berhasil login dan diarahkan ke halaman Dashboard.	Pengguna berhasil login dan diarahkan ke halaman Dashboard.	Berhasil
2	Login dengan Kredensial Salah	1. Buka halaman login.2. Masukkan username atau password yang salah.3. Klik tombol Log In .	Sistem menampilkan pesan kesalahan bahwa username atau password tidak valid.	Sistem menampilkan pesan kesalahan yang sesuai.	Berhasil
3	Menambah Data Siswa Baru	1. Login ke sistem.2. Masuk ke menu Kelola Data Master → Data Siswa .3. Klik	Data siswa baru berhasil ditambahkan dan muncul dalam tabel daftar siswa.	Data siswa baru tersimpan dengan benar dan tampil di tabel.	Berhasil

		tombol Tambah Siswa .4. Isi semua data pada formulir.5. Klik Simpan Data Siswa .			
4	Mengedit Data Siswa	1. Masuk ke menu Data Siswa .2. Pilih salah satu siswa dan klik tombol Edit .3. Ubah informasi (misalnya Rata-rata Rapor).4. Klik Simpan .	Perubahan data siswa berhasil disimpan dan diperbarui di dalam tabel.	Data siswa berhasil diperbarui.	Berhasil 1
5	Menghapus Data Siswa	1. Masuk ke menu Data Siswa .2. Pilih salah satu siswa dan klik tombol Hapus .3. Konfirmasi penghapusan.	Data siswa terhapus dari daftar dan database.	Data siswa berhasil dihapus.	Berhasil 1
6	Menjalankan Pelatihan Model	1. Masuk ke menu Latih & Uji Model .2. Klik Mulai Proses Ulang .3. Ikuti alur Tahap 1 (EDA) hingga Tahap 5 (Detail Prediksi).	Sistem berhasil memproses data, melatih model GBM, dan menampilkan metrik evaluasi.	Proses pelatihan berjalan lancar dan semua metrik evaluasi ditampilkan benar.	Berhasil 1
7	Menyimpan Model yang Dilatih	1. Setelah evaluasi di menu Latih & Uji Model , sistem menyimpan model otomatis.2.	Model yang baru dilatih muncul di daftar model tersimpan dengan status Aktif .	Model baru tersimpan dan tercatat di halaman Kelola Model.	Berhasil 1

		Masuk ke menu Kelola Model .			
8	Melakukan Klasifikasi Baru	1. Masuk ke menu Hasil Klasifikasi .2. Klik Lakukan Klasifikasi Baru .3. Pilih siswa yang akan diklasifikasi.4. Klik tombol Klasifikasi .	Sistem menampilkan hasil prediksi (Lulus/Tidak Lulus) beserta probabilitasnya , dan riwayat tersimpan.	Hasil klasifikasi ditampilkan akurat dan tersimpan dalam riwayat.	Berhasil 1
9	Mengunduh Laporan	1. Masuk ke menu Laporan .2. Pilih rentang tanggal dan format (PDF/Excel).3 . Klik tombol Unduh .	Sistem berhasil membuat file laporan sesuai format yang dipilih dan dapat diunduh.	Laporan berhasil di-generate dan diunduh.	Berhasil 1
10	Logout dari Sistem	1. Klik nama pengguna di pojok kanan atas.2. Pilih opsi Logout .3. Konfirmasi logout.	Pengguna berhasil keluar dari sesi dan diarahkan kembali ke halaman login.	Pengguna berhasil logout dengan aman.	Berhasil 1

4.4. Kelebihan dan Kelemahan Program

Berdasarkan aplikasi yang sudah dibuat, terdapat beberapa kelebihan dan kelemahan yang dapat diidentifikasi. Berikut adalah analisisnya:

istem ini memiliki sejumlah keunggulan, di antaranya akurasi yang tinggi berkat penerapan algoritma Gradient Boosting Machine (GBM), antarmuka berbasis web yang interaktif, serta alur pelatihan model yang transparan. Selain itu, sistem juga mendukung pengambilan keputusan berbasis data, memungkinkan sekolah untuk memperoleh wawasan yang objektif dalam menilai potensi siswa.

Manajemen data yang terpusat serta fleksibilitas untuk melakukan *retraining* model menjadikan sistem ini adaptif terhadap data baru.

Namun, kinerja sistem sangat bergantung pada kualitas data yang digunakan. Data yang kurang lengkap atau bias berpotensi menurunkan akurasi prediksi. Model juga memerlukan pemeliharaan rutin melalui evaluasi dan pelatihan ulang secara berkala, yang membutuhkan perhatian teknis dari admin. Selain itu, GBM berisiko mengalami *overfitting* jika konfigurasi tidak tepat atau data latih terbatas.

Keterbatasan lainnya terletak pada kompleksitas interpretasi hasil prediksi, karena logika GBM cenderung sulit dipahami pengguna non-teknis. Model juga hanya menggunakan beberapa atribut utama, sehingga faktor eksternal lain seperti kondisi sosial-ekonomi atau dukungan keluarga belum diperhitungkan. Hal ini membatasi cakupan analisis, meskipun tetap memberi kontribusi besar dalam mendukung pengambilan keputusan berbasis data.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan analisis, implementasi, dan pengujian sistem yang telah dilakukan, maka dapat ditarik beberapa kesimpulan yang menjawab rumusan masalah penelitian sebagai berikut:

1. Implementasi Algoritma Gradient Boosting Machine (GBM) Berhasil Dilakukan: Algoritma GBM berhasil diimplementasikan untuk mengklasifikasikan data keberhasilan siswa. Proses ini berjalan melalui tahapan sistematis yang dimulai dengan inisialisasi model pada skala *log-odds* untuk menghasilkan prediksi probabilitas awal. Selanjutnya, algoritma secara iteratif menghitung *pseudo-residual*, yang merepresentasikan kesalahan dari prediksi sebelumnya. Sebuah model pohon keputusan sederhana (*weak learner*) kemudian dibangun untuk memprediksi kesalahan tersebut, di mana atribut Skor Prestasi teridentifikasi sebagai pemisah data yang paling signifikan. Model kemudian diperbarui secara aditif dengan kontribusi dari pohon baru, yang pada akhirnya meningkatkan probabilitas prediksi mendekati nilai aktual.
2. Faktor Paling Berpengaruh adalah Skor Prestasi: Hasil dari proses pelatihan model dan perhitungan manual menunjukkan bahwa Skor Prestasi adalah faktor yang paling berpengaruh terhadap keberhasilan siswa. Pada iterasi pertama, model mampu memisahkan siswa dengan status "Lulus" dan "Tidak Lulus" secara sempurna hanya dengan menggunakan atribut ini. Hal ini mengindikasikan bahwa siswa yang memiliki catatan prestasi non-akademik,

sekecil apa pun, memiliki kecenderungan yang jauh lebih tinggi untuk berhasil dibandingkan siswa yang tidak memiliki prestasi sama sekali.

3. Sistem Klasifikasi Berbasis Web Berhasil Dibangun: Sebuah sistem berbasis web telah berhasil dibangun untuk membantu pihak sekolah dalam pengambilan keputusan. Sistem ini dirancang dengan arsitektur yang memisahkan *frontend* (dibangun dengan PHP dan HTML) dan *backend* (dibangun dengan Python dan Flask API). Sistem ini menyediakan fungsionalitas lengkap, mulai dari manajemen data master (siswa, kelas, mata pelajaran, prestasi) ,alur pelatihan model yang interaktif , hingga pengelolaan model dan riwayat hasil klasifikasi

5.2 Saran

Berdasarkan batasan masalah dan temuan selama penelitian, berikut adalah beberapa saran untuk pengembangan sistem di masa mendatang:

1. Perluasan Cakupan Data: Penelitian ini hanya terbatas pada data siswa kelas VI di SD Muhammadiyah Medan Timur. Untuk meningkatkan kemampuan generalisasi model, penelitian selanjutnya disarankan untuk menggunakan dataset yang lebih besar, mencakup beberapa tahun ajaran atau bahkan dari sekolah lain. Hal ini akan membantu model mengenali pola yang lebih beragam dan mengurangi risiko *overfitting*.
2. Penambahan Atribut Data: Data yang digunakan hanya meliputi nilai rapor, kehadiran, dan prestasi non-akademik. Pengembangan selanjutnya dapat mempertimbangkan atribut lain yang relevan, seperti data demografi sstatus sosial ekonomi, hasil tes psikologi (minat dan bakat), atau bahkan data

keikutsertaan dalam bimbingan belajar, untuk menghasilkan prediksi yang lebih komprehensif.

3. Integrasi dan Otomatisasi Sistem: Sistem yang dibangun saat ini masih memerlukan input data manual dan berjalan di server lokal menggunakan XAMPP. Untuk meningkatkan efisiensi, sistem disarankan untuk diintegrasikan langsung dengan Sistem Informasi Akademik (SIKAD) sekolah. Dengan demikian, data siswa dapat ditarik secara otomatis, dan proses klasifikasi dapat dijadwalkan secara berkala tanpa intervensi manual dari admin.

DAFTAR PUSTAKA

- Andri Aka, K. (2022). *MODEL QUANTUM TEACHING DENGAN PENDEKATAN COOPERATIVE LEARNING UNTUK MENINGKATKAN KUALITAS PEMBELAJARAN PKn*.
- Arugia, A. W., Junirianto, E., & Maria, E. (2022). Design and Build Web and API on “Absenplus” with Face Recognition using Deep Learning Method. *TEPIAN*, 3(2), 65–75. <https://doi.org/10.51967/tepiian.v3i2.738>
- Atlantic, V., Sulistianingsih, E., & Perdana, H. (2024). GRADIENT BOOSTING MACHINE PADA KLASIFIKASI KELULUSAN MAHASISWA. In *Buletin Ilmiah Math. Stat. dan Terapannya (Bimaster)* (Vol. 13, Issue 2).
- Elsa Suryana, S., & Warsito, B. (2021). *PENERAPAN GRADIENT BOOSTING DENGAN HYPEROPT UNTUK MEMKLASIFIKASI KEBERHASILAN TELEMARKETING BANK*. 10, 617–623. <https://ejournal3.undip.ac.id/index.php/gaussian/>
- Franz, A., & Rachmadani, B. (2025). *Information System for Services and Management of Fishermen’s Data from the Kutai Kartanegara Marine and Fisheries Service*. 6(1), 38. <https://doi.org/10.51967/tepiian.v6i1.1848>
- Galahartlambang, Y., Khotiah, T., & Jumain. (2022). *Visualisasi Data Dari Dataset COVID-19 Menggunakan Pemrograman Python*.
- Gede, I., Kurniarwan, J., Dewi, C., & Rahman, M. A. (2025). *PENERAPAN MACHINE LEARNING EXTREME GRADIENT BOOSTING DALAM KLASIFIKASI POTENSI TSUNAMI BERDASARKAN DATA GEMPA BUMI* (Vol. 9, Issue 2). <http://j-ptiik.ub.ac.id>
- Harahap, P. N., & Sulindawaty, S. (2020). Implementasi Data Mining Dalam Memklasifikasi Transaksi Penjualan Menggunakan Algoritma Apriori (Studi Kasus PT.Arma Anugerah Abadi Cabang Sei Rampah). *MATICS*, 11(2), 46. <https://doi.org/10.18860/mat.v11i2.7821>
- Iriane, R., & Nurfaizah. (2023). Penerapan Data Mining Untuk Klasifikasi Penjualan Produk Pangan Hewan Menggunakan Metode K-Nearest Neighbor. *KLIK : Kajian Ilmiah Informatika Dan Komputer*, 3(5), 509–515. <https://djournals.com/klik>

- Mahdiyah, U. (2023). KLASIFIKASI KUALITAS CITRA CABAI DENGAN MENGGUNAKAN ALGORITMA GRADIEN BOOSTING. *JAMI: Jurnal Ahli Muda Indonesia*, 4(1), 61–69. <https://doi.org/10.46510/jami.v4i1.137>
- Mahmoud, Z., & Sayed, A. W. (2024). ENHANCING ACADEMIC PERFORMANCE PREDICTION FOR AT-RISK STUDENTS: COMPARATIVE ANALYSIS OF MACHINE LEARNING ALGORITHMS IN EARLY WARNING SYSTEMS. *Journal of Theoretical and Applied Information Technology*, 15(19). www.jatit.org
- Putra, V., Tri Pranoto, G., & Eko Putra, F. (2023). Klasifikasi Kebutuhan Sparepart Dengan Algoritma K-Nearest Neighbor Untuk Meningkatkan Penjualan Sparepart. *Bulletin of Information Technology (BIT)*, 4(2), 287–293. <https://doi.org/10.47065/bit.v3i1>
- Susilahati, S., Nurmalia, L., Widiawati, H., Laksana, A. M., & Maliadani, L. (2023). Upaya Penerapan Transisi PAUD Ke SD yang Menyenangkan: Ditinjau dari PPDB, MPLS dan Proses Pembelajaran. *Jurnal Obsesi : Jurnal Pendidikan Anak Usia Dini*, 7(5), 5779–5794. <https://doi.org/10.31004/obsesi.v7i5.5320>
- Tofir, S., Leppang, I., & Hamzah, M. A. (2020). PERANCANGAN SISTEM INFORMASI PADA DINAS PENDIDIKAN KOTA PALOPO BERBASIS WEB. *Jurnal Syntax Admiration*, 1(6).
- Tri Pratiwi, A., Barizi, A., Irfan Maulana, M., & Rosyani, P. (2024). *Systematic Literature Review Penerapan Gradient Boosting Untuk Klasifikasi Penyakit Diabetes Tipe 2*. <https://jurnalmahasiswa.com/index.php/biikma>
- Wardhana, I., Ariawijaya, M., Ahmad Isnaini, V., & Putri Wirman, R. (2022). *Gradient Boosting Machine, Random Forest dan Light GBM untuk Klasifikasi Kacang Kering*.
- Wicaksono, A., Putro, G., & Setiadi, T. (2023). Penerapan Klasifikasi Decision Tree (*GRADIENT BOOSTING MACHINE*) untuk Memklasifikasi Kelulusan Siswa Sekolah Dasar di Kecamatan Juai. *Jurnal Format*, 12.
- Yusuf, P. M. (2022). Design of Semi Automatic Announcement System In The Public Address System Part of The Main Branch Office Soekarno-Hatta International Airport Angkasa Pura II, Inc. In *Journal of Electrical, Electronic, Information, and Communication Technology (JEEICT)* (Vol. 4, Issue 2).

LAMPIRAN

1. PENETAPAN DOSEN PEMBIMBING



**PENETAPAN DOSEN PEMBIMBING
 PROPOSAL/SKRIPSI MAHASISWA
 NOMOR : 902/IL3-AU/UMSU-09/F/2024**

Assalamu'alaikum Warahmatullahi Wabarakatuh

Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara, berdasarkan Persetujuan permohonan judul penelitian Proposal / Skripsi dari Ketua / Sekretaris:

Program Studi : Sistem Informasi
 Pada tanggal : 20 November 2024

Dengan ini menetapkan Dosen Pembimbing Proposal / Skripsi Mahasiswa

Nama : Afdolly Akbar Khaidir Siregar
 NPM : 2109010112
 Semester : VII (Tujuh)
 Program studi : Sistem Informasi
 Judul Proposal / Skripsi : Analisis Dan Implementasi Algoritma Fuzzy Tsukamoto Dalam Penentuan Kepala Lingkungan Di Kelurahan Sidorame Barat 2

Dosen Pembimbing : Halim Maulana, S.T., M.Kom.

Dengan demikian di izinkan menulis Proposal / Skripsi dengan ketentuan

1. Penulisan berpedoman pada buku panduan penulisan Proposal / Skripsi Fakultas Ilmu Komputer dan Teknologi Informasi UMSU
2. Pelaksanaan Sidang Skripsi harus berjarak 3 bulan setelah dikeluarkannya Surat Penetapan Dosen Pembimbing Skripsi.
3. **Proyek Proposal / Skripsi dinyatakan " BATAL " bila tidak selesai sebelum Masa Kadaluarsa tanggal : 20 November 2025**
4. Revisi judul.....

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Ditetapkan di : Medan
 Pada Tanggal : 18 Jumadil Awwal 1446 H
 20 November 2024 M



Dekan

 Dr. H. Khairul Rizmi, S.Kom., M.Kom.
 NIDN : 0127099201

Cc. File



2. TEMPAT PENELITIAN





3. UNDANGAN SEMINAR PROPOSAL

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
 Nomor : 709/ILJ-AU/UMSU-09/F/2025

UNDANGAN SEMINAR PROPOSAL

Fakultas : Ilmu Komputer dan Teknologi Informasi
 Program Studi : Sistem Informasi
 Hari/Tanggal : Senin, 07 Juli 2025
 Waktu /Tempat : 09.00G
 Peninggil Seminar : Yohida Sary, S.E., S.Kom., M.Kom.

5	2109010112	Adhdy- Albar Khasdir Surgar	Halin Maulana, S.T., M.Kom	Yohida Sary, S.E., S.Kom., M.Kom.	Implementasi Gradient Boosting Machine Untuk Klasifikasi Data Kesehatan Siswa Pada SD Muhammadiyah Medan Timur Berbasis Web
6	2109010042	Mubandaul Ayo	Wida Rana HSB S.T., M.Kom	Yohida Sary, S.E., S.Kom., M.Kom.	Analisis Data Implementasi Algoritma SVM Dalam Pengendalian Peredaran Bahan Baku Komsumsi Kapsul Di PT. Pabrik Cawang Medan
7	2109010073	Aldi Pratama	Fandi Albar Surga S.Kom., M.Kom	Dr. Al-Khawazmi, M.Kom.	Analisis dan Implementasi Menganalisa Algoritma decision tree dalam Memprediksi Tingkat Kemauan Pada Perbankan Ayan Breder


 N/B - Link-laki berkesempatan himon penuh dan memadai dari
 Persempitan keribawa mufarika himon penuh


 Medan, 08 Muharram 1447 H
 03 Juli 2025 M
 Syarif Al-Khawazmi, M.Kom
 NIBN.S.170992301



4. BERITA ACARA BIMBINGAN SKRIPSI (SIDANG)



MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/IBAN-PT/Akred/PT/III/2019
 Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003
 www.umsumedan.ac.id | info@umsumedan.ac.id | umsumedan | umsumedan | umsumedan | umsumedan

Berita Acara Pembimbingan Skripsi

Nama Mahasiswa : Afdolly Akbar Khaidir Siregar Program Studi : Sistem Informasi
 NPM : 2109010112 Konsentrasi : *implementasi aplikasi berbasis web*
 Nama Dosen Pembimbing : Halim Maulana, S.T., M.Kom Judul Penelitian : *machine learning untuk klasifikasi data berdasarkan screen pada SP Muhammadiyah Medan Timur*

Item	Hasil Evaluasi	Tanggal	Paraf Dosen
1	Bimbingan Revisi Seminar Proposal. Perhitungannya disesuaikan dengan Model	18/06/25	<i>[Signature]</i>
2	Tampilkan Gambar Modelnya dan tambahkan Perhitungannya	8/8/25	<i>[Signature]</i>
3	ACC Sidang	2/9/25	<i>[Signature]</i>

Medan, 02 SEP 2025

Diketahui oleh :
 Ketua Program Studi
 Sistem Informasi
[Signature]
 (Mardiana, S.Pd., S.Kom., M.Kom)

Disetujui oleh :
 Dosen Pembimbing
[Signature]
 (Halim Maulana, S.T., M.Kom.)



5. UNDANGAN SIDANG MEJA HIJAU

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
 Nomor : 907/II.3-A/U/UMSU-09/F/2025
 HAL LULIAN MEJA HIJAU SARJANA (S1)

UNDANGAN
 PANGGILAN



UMSU
 Mengajar, Menginspirasi, Mengembangkan

Fakultas : Ilmu Komputer dan Teknologi Informasi
 Program Studi : Sistem Informasi
 Hari/Tanggal : Sabtu, 13 September 2025
 Waktu /Tempat : 08.00-14.00 WIB/Gedung G

Kepada Yang Terhormat
 Bapak/Ibu Dosen Pengajar Meja Hijau
 di
 Medan
 Catatan :
 *Harap datang tepat waktu karena ujian
 dalam bentuk tim (2 Orang) pengajar I & II
 *Dosen Pengajar yang terlambat 30 menit
 akan diganti

No	MAHASISWA	NPM	JUDUL SKRIPSI	DOSIS PENGUJI I	DOSIS PENGUJI II	Direksi Pengembangan/ Pengajar II	KET
20	Aldilly Abbar Khaedir Siragat	2109010112	Implementasi Gradient Boosting Machine Untuk Klasifikasi Data Kebersihan Siswa Pada SD Muhammadiyah Medan Timur Berbasis Web	Yoshida Sary, S.E., S.Kom., M.Kom.	Rizaldy Khair, M.Kom.	Halim Maulana, S.T., M.Kom	
21	Dimas Arya Panyoga	2109010026	Identifikasi Buah Kelapa Sawit Berdasarkan Citra Red,Green,Blue Menggunakan Algoritma Naive Bayes Classifier	Yohanni Syahr, S.Si., M.Kom.	Dr. Al-Khowarizmi, M.Kom	Muhammad Azhar, M.Kom	
22	Asnu ul Husniah	2109010105	Analisis Dan Implementasi Sistem Keamanan Data Informasi Kampus Menggunakan Algoritma Camellia Pada SD Muhammadiyah 08 Medan	Yoshida Sary, S.Kom., S.E., M.Kom.	Halim Maulana, S.T., M.Kom	Ferdy Riza S.T., M.T	
23	Calaya Ramadhani	2109010150	Analisa Penerapan Model K-Nearest Neighbors (KNN) Dalam Machine Learning Untuk Menentukan Konsentrasi Program Studi Siswa Kelas 12 Di Sekolah SMK Tritech Informatika	Dr. Firaeni Raky, S.Kom., M.Kom	Halim Maulana, S.T., M.Kom	Annulad, S.Kom., M.Kom	

Asisten Pengambilan Berita Acara :

1. Suvia Agustina S.I.Kom
2. Andika Suras Saputra, S.M

Ditandatangani Oleh

Penyita Ujian

a.n Rektor
 Wakil Rektor I
 Prof. Dr. MUHAMMAD ARIFFIN, S.H., M.Hum.

Ketua
 Universitas Muhammadiyah Sumatera Utara
 Halim Maulana, M.Kom.

Sekretaris
 Wakil Dekan I
 Halim Maulana, S.T., M.Kom.

Medan, 19 Rabu, 11 Agustus 1447 H
 11 September 2025 M

6. HASIL CEK TURNITIN

Skripsi Afdolly Akbar Khaidir Srg.docx			
ORIGINALITY REPORT			
25%	22%	13%	16%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	Submitted to Universitas Muria Kudus Student Paper		1%
2	repository.umsu.ac.id Internet Source		1%
3	bajangjournal.com Internet Source		1%
4	Submitted to Universitas Pamulang Student Paper		1%
5	Submitted to Universitas Putera Batam Student Paper		1%
6	docplayer.info Internet Source		1%
7	e-journal.politanisamarinda.ac.id Internet Source		<1%
8	Submitted to Universitas Musamus Merauke Student Paper		<1%
9	repository.uin-suska.ac.id Internet Source		<1%
10	repository.dinamika.ac.id Internet Source		<1%
11	Submitted to Universitas Muhammadiyah Sumatera Utara Student Paper		<1%