

BAB V

HASIL DAN PEMBAHASAN

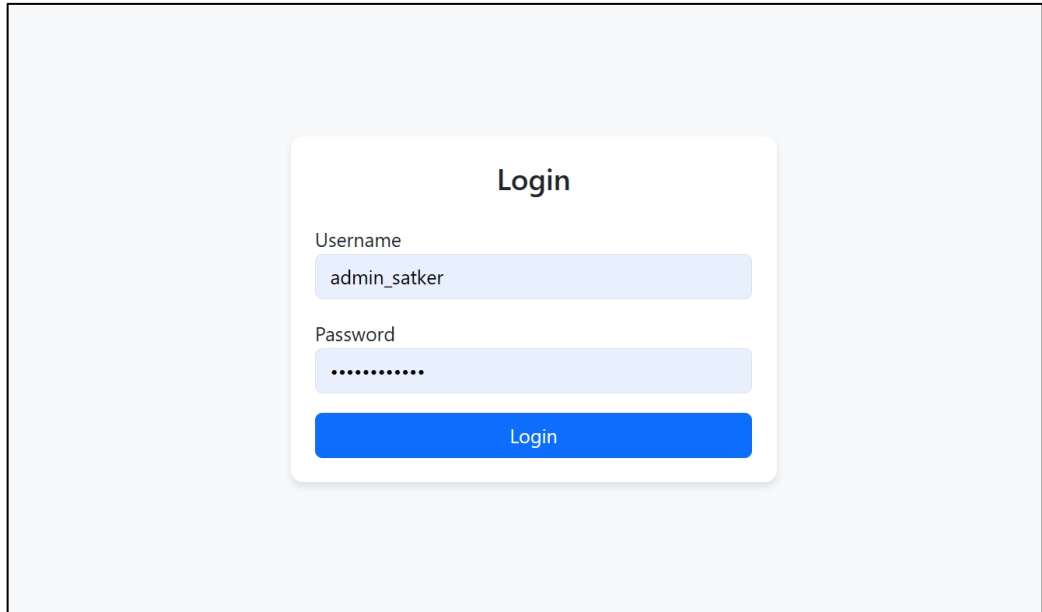
Di bagian ini berfokus pada evaluasi yang telah dicapai. Selain itu, bab ini juga membahas sistem sebelum diterapkan guna memastikan bahwa aplikasi dapat berfungsi sesuai dengan tujuan yang telah ditetapkan.

5.1 Hasil

Aplikasi klasifikasi data temuan menggunakan *Algoritma Decision Tree* ini berhasil memberikan solusi yang efisien untuk menganalisis dan mengevaluasi data temuan. Dengan fitur, training model, hingga evaluasi berbasis confusion matrix, akurasi, precision, dan recall, aplikasi ini memastikan hasil analisis yang akurat dan transparan. Tabel hasil klasifikasi menunjukkan perbandingan antara data aktual dan prediksi model. Selain itu, fitur unduh dokumen dalam format PDF memberikan fleksibilitas dalam pengelolaan hasil analisis, terutama untuk data pegawai yang dinyatakan layak. Aplikasi ini memberikan kontribusi signifikan dalam mendukung pengambilan keputusan secara sistematis dan berbasis data, menggantikan proses manual yang memakan waktu dan rawan kesalahan. Berikut adalah hasil rancangan dan implementasi yang telah di kembangkan.

5.1.2 Implementasi

Tahap implementasi berfungsi untuk mengetahui apakah aplikasi tersebut bekerja sesuai dengan semestinya, dan melihat apakah tampilan *interface* nya sudah sesuai dengan yang di rancang. Berikut adalah implementasi dari aplikasi Surat Keterangan Bebas Temuan (SKBT).



Gambar 5.1 Tampilan Halaman Login

Tampilan diatas merupakan halaman awal dari aplikasi Surat Keterangan Bebas Temuan. Admin atau user harus melakukan pengunggahan data temuan dengan menekan tombol pilih file kemudian klik tombol unggah file. Lalu sistem melakukan preprocessing data untuk memvalidasi apakah data temuan tersebut sudah sesuai dengan ketentuan, dan setelah tahap preprocessing data temuan dilatih dengan *Algoritma Decision Tree* dan menampilkan hasil evaluasi model.

	Prediksi: Tidak Layak	Prediksi: Layak
Aktual: Tidak Layak	14	1
Aktual: Layak	0	43

Gambar 5.2 Tampilan Visualisasi Dashboard Evaluasi Model

Tampilan halaman dashboard pada aplikasi ini menyajikan informasi utama tentang evaluasi model klasifikasi yang digunakan dalam sistem. Pada bagian atas,

terdapat hasil "Akurasi Model," yang mencakup akurasi data pelatihan dan validasi sebagai indikator kinerja model. Selanjutnya, metrik evaluasi seperti Precision, Recall, dan F1-Measure ditampilkan untuk memberikan gambaran detail tentang kualitas prediksi model. Di bawahnya, terdapat visualisasi *Confusion Matrix* yang memperlihatkan distribusi hasil klasifikasi antara prediksi "Layak" dan "Tidak Layak," dibandingkan dengan label aktual. Dengan desain sederhana dan informatif, dashboard ini membantu pengguna memahami performa model secara langsung.

SKBT Decision Tree		Data Test (Tanpa Rekomendasi)			
ID	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Prediksi	
290	Defisit Anggaran	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak	
291	Pembayaran Honor Tidak Sesuai	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak	
292	Indisipliner	Keuangan	Proses pengembalian	Tidak Layak	
293	Pembayaran Honor Tidak Sesuai	Keuangan	Proses pengembalian	Tidak Layak	
298	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Tidak Layak	
299	Kelebihan Pembayaran Tunjangan Kinerja	Keuangan	Proses pengembalian	Tidak Layak	
300	Indisipliner	Kepegawaian	Disiplin internal	Layak	
301	Indisipliner	Kepegawaian	Disiplin internal	Layak	
302	Pembayaran Honor Tidak Sesuai	Keuangan	Dana dikembalikan	Layak	
303	Kelebihan Pembayaran Tunjangan Kinerja	Keuangan	Dana dikembalikan	Layak	
304	Pembayaran Honor Tidak Sesuai	Keuangan	Dana dikembalikan	Layak	
305	Indisipliner	Kepegawaian	Disiplin internal	Tidak Layak	
306	Indisipliner	Kepegawaian	Disiplin internal	Tidak Layak	
308	Pembayaran Honor Tidak Sesuai	Keuangan	Proses pengembalian	Tidak Layak	
314	Defisit Anggaran	Keuangan	Pengembalian dalam proses	Tidak Layak	

Gambar 5.3 Tampilan Visualisasi Dashboard Data Test

Pada bagian tambahan halaman dashboard ini, ditampilkan tabel data uji (*Data Test*) tanpa rekomendasi yang memberikan rincian data yang diuji oleh model. Tabel ini mencakup kolom ID, Jenis Temuan, Aspek Temuan, Tindak Lanjut, dan Prediksi. Kolom Prediksi memperlihatkan hasil klasifikasi dari model Decision Tree, yaitu "Layak" atau "Tidak Layak," berdasarkan data masukan seperti jenis temuan dan tindak lanjut yang diambil. Dengan penyajian tabel yang terstruktur dan rapi, pengguna dapat dengan mudah memverifikasi hasil prediksi dan membandingkannya dengan data input yang relevan, mendukung analisis lebih lanjut terhadap performa model.

SKBT Decision Tree		Data Training			
ID	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Rekomendasi	Prediksi
1	Defisit Anggaran	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak	Tidak Layak
2	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Layak	Layak
3	Ketidakpatuhan Prosedur	Tugas dan Fungsi	Laporan belum diterima	Tidak Layak	Tidak Layak
4	Indisipliner	Tugas dan Fungsi	Teguran internal diberikan	Tidak Layak	Tidak Layak
5	Ketidakpatuhan Prosedur Nikah	Tugas dan Fungsi	Teguran tertulis	Layak	Layak
6	Pengangkatan PNS Tidak Sah	Kepegawaian	Pemberhentian dengan hormat	Tidak Layak	Layak
7	Pengabaian Pembayaran Gaji	Kepegawaian	Pembayaran dihentikan	Layak	Layak
8	Kelebihan Pembayaran Honor	Keuangan	Pengembalian kelebihan pembayaran	Layak	Layak
9	Kelebihan Pembayaran Honor	Keuangan	Dana telah disetor ke kas negara	Layak	Layak
10	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
11	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
12	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
13	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
14	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
15	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
16	Double Account Perjalanan Dinas	Keuangan	Dana telah dikembalikan	Layak	Layak

Gambar 5.4 Tampilan Visualisasi Dashboard Data Training

Pada bagian halaman dashboard ini, ditampilkan tabel *Data Training* yang berisi data yang digunakan untuk melatih model *Decision Tree*. Tabel ini mencakup kolom ID, Jenis Temuan, Aspek Temuan, Tindak Lanjut, Rekomendasi, dan Prediksi. Kolom Rekomendasi menunjukkan nilai target yang telah diberikan sebelumnya, sedangkan kolom Prediksi menunjukkan hasil prediksi dari model berdasarkan data input. Dengan menyandingkan Rekomendasi dan Prediksi, pengguna dapat mengevaluasi akurasi model dalam memprediksi status kelayakan (*Layak* atau *Tidak Layak*). Penyajian data yang terorganisir ini memberikan wawasan mendalam kepada pengguna untuk menilai performa model dan memahami pola-pola pada data.

ID	Nama Pegawai	NIP	Pangkat/Golongan	Jabatan	Unit Kerja	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Prediksi	Aksi
1	Hamsi Djamsi	196102101986031000	III/c	Kepala Urusan Tata Usaha MAN Insan Cendekia Kabupaten Halmahera Barat Provinsi Maluku Utara	Kandepag Ternate	Defisit Anggaran	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak	Edit Hapus
2	Drs. H. AMIR TOMAGOLA, MMMH	196712081995031000	IV/b	Kepala Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara	Kanwil Maluku Utara	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Layak	Edit Hapus
3	ABD RAHMAN KHARIE, S.Ag	195903201986031000	IV/b	Pegawai Subbagian Tata Usaha Sekretaris Jenderal Bagian Tata Usaha Pimpinan Biro Umum Sekretariat Jenderal	Kanwil Maluku Utara	Ketidakpatuhan Prosedur	Tugas dan Fungsi	Laporan belum diterima	Tidak Layak	Edit Hapus

Gambar 5.5 Tampilan Visualisasi Surat Masuk

Halaman Surat Masuk pada aplikasi ini menampilkan daftar data pegawai yang terorganisir dalam tabel interaktif. Tabel ini mencakup kolom ID, Nama Pegawai, NIP, Pangkat/Golongan, Jabatan, Unit Kerja, Jenis Temuan, Aspek Temuan, Tindak Lanjut, Prediksi, dan Aksi. Fitur pencarian di bagian atas memungkinkan pengguna untuk mencari data berdasarkan nama pegawai. Tombol "Tambah Data Baru" memberikan akses untuk memasukkan data baru. Kolom Aksi menyediakan tombol Edit dan Hapus untuk mengelola data secara langsung. Desain yang responsif dan intuitif ini mendukung kemudahan pengelolaan data temuan dalam proses pengawasan dan pengelolaan administratif.

SKBT Decision Tree

Dashboard

Surat Masuk

Surat Keluar

Logout

Surat Keluar

Download Selected

Select	ID	Nama Pegawai	NIP	Pangkat/Golongan	Jabatan	Unit Kerja	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Rekomendasi	Aksi
<input type="checkbox"/>	2	Drs. H. AMIR TOMAGOLA, MM, MH	196712081995031000	IV/b	Kepala Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara	Kanwil Maluku Utara	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Layak	Download
<input type="checkbox"/>	5	Drs. KAMAL DO HAMZAH	196010141991031000	III/d	Pengadministrasi Sub Bagian Tata Usaha Kantor Kementerian Agama Kota Ternate Provinsi Maluku	Kanwil Maluku Utara	Ketidaktepatan Prosedur Nikah	Tugas dan Fungsi	Teguran tertulis	Layak	Download
<input type="checkbox"/>	7	Drs. NASARUDDIN ALY, MM	196804131995031000	IV/b	Kepala Kantor Kementerian Agama Kabupaten Pulau Taliabu Provinsi Maluku Utara	Kanwil Halmahera Barat	Pengabaian Pembayaran Gaji	Kepegawaian	Pembayaran dihentikan	Layak	Download
<input type="checkbox"/>	8	HARTATI TASWA S.Pd.I	197510212006042002	III/b	Analisis Informasi Pendidikan (Jabatan Pelaksana) pada Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara	Kanwil Halmahera Barat	Kelebihan Pembayaran Honor	Keuangan	Pengembalian kelebihan pembayaran	Layak	Download
<input type="checkbox"/>	9	FAISAL HI. S. TALIB, S.HI	198208022006041003	III/d	Pranata Keuangan APBN Penyelia pada Seksi Bimbingan Masyarakat Islam Kantor Kementerian Agama Kabupaten Halmahera Utara Kanwil Provinsi Maluku Utara	Kankemenag Maluku Utara	Kelebihan Pembayaran Honor	Keuangan	Dana telah disetor ke kas negara	Layak	Download
<input type="checkbox"/>	10	FAHRIA HI HASAN S.EI	198212302006042000	III/d	Pranata Keuangan APBN Penyelia pada Sub Bagian Tata Usaha Kantor Kementerian Agama Kabupaten Halmahera Utara Kanwil Provinsi Maluku Utara	Kankemenag Maluku Utara	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Download
<input type="checkbox"/>	11	YUNIATI ST. K.A.S.P.	198101262006042003	II/d	Pranata Keuangan APBN Terampil pada Sub Bagian Tata Usaha Kantor Kementerian Agama Kabupaten Halmahera Utara Kanwil Provinsi Maluku Utara	Kankemenag Maluku Utara	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Download

Gambar 5.6 Tampilan Visualisasi Surat Keluar

Halaman *Surat Keluar* pada aplikasi ini dirancang untuk menampilkan daftar pegawai yang telah memiliki rekomendasi layak dalam bentuk tabel terstruktur. Tabel ini mencakup kolom *Select*, *ID*, *Nama Pegawai*, *NIP*, *Pangkat/Golongan*, *Jabatan*, *Unit Kerja*, *Jenis Temuan*, *Aspek Temuan*, *Tindak Lanjut*, *Rekomendasi*, dan *Aksi*. Tombol *Download* pada kolom *Aksi* memungkinkan pengguna untuk mengunduh surat terkait pegawai secara individu, sementara tombol *Download Selected* di bagian atas menyediakan opsi untuk mengunduh surat secara massal berdasarkan pilihan yang ditandai. Desain yang bersih dan navigasi yang mudah dipahami mendukung efisiensi pengguna dalam mengelola dan mendistribusikan surat keluar.

5.1.4 Hasil Pengujian Black Box

Pengujian ini bertujuan untuk menguji fungsi sistem sistem perangkat lunak dengan menitikberatkan pada pemenuhan persyaratan utama perangkat lunak. Dalam pengujian ini, penguji bertugas menentukan kumpulan kondisi input yang valid dan memastikan luarannya sesuai dengan spesifikasi perangkat lunak.

Tabel 5.1 Hasil Pengujian *Black box*

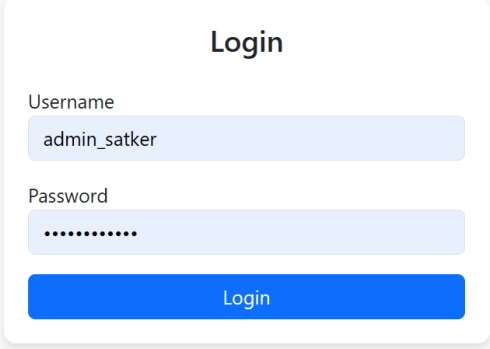
No.	Pengujian	Proses	Hasil Yang Diharapkan
1	Login	Pengguna memasukkan username dan password, kemudian klik tombol Login.	Sistem berhasil mengautentikasi pengguna dan mengarahkan ke halaman utama.
2	Logout	Pengguna klik tombol Logout.	Sistem menghapus sesi pengguna dan mengarahkan ke halaman login.
3	Tambah Data	Pengguna mengisi form data temuan, lalu klik tombol Simpan.	Data temuan berhasil disimpan ke database dan ditampilkan di tabel Surat Masuk.
4	Edit Data	Pengguna memilih data di tabel Surat Masuk, mengedit data, lalu klik tombol Simpan.	Data temuan berhasil diperbarui di database dan diperbarui di tabel Surat Masuk.
5	Hapus Data	Pengguna memilih data di tabel Surat Masuk, lalu klik tombol Hapus.	Data temuan berhasil dihapus dari database dan hilang dari tabel Surat Masuk.
6	Pencarian Data	Pengguna memasukkan kata kunci pada kotak pencarian, lalu klik tombol Cari.	Data yang sesuai dengan kata kunci ditampilkan di tabel Surat Masuk.
7	Unduh Surat Individu	Pengguna klik tombol Download pada salah satu data di tabel Surat Keluar.	Sistem menghasilkan file PDF surat untuk data yang dipilih dan mengunduhnya.
8	Unduh Surat Massal	Pengguna memilih beberapa data, lalu klik tombol Download Selected di tabel Surat Keluar.	Sistem menghasilkan file ZIP berisi PDF surat untuk semua data yang dipilih dan mengunduhnya.
9	Tampilkan Evaluasi Model	Pengguna mengakses halaman Dashboard.	Sistem menampilkan akurasi model, metrik evaluasi (precision, recall, F1-score), dan matriks.

5.1.5 Hasil Pengujian *White Box*

Metode ini didasarkan pada analisis mendalam terhadap detail spesifik desain sistem. Pengujian ini dilakukan dengan memanfaatkan deskripsi program secara berurutan untuk membagi proses pengujian menjadi beberapa kasus uji. Berikut adalah hasil pengujian yang menggunakan pendekatan white box.

Tabel 5.2 Hasil Pengujian White Box

No	Komponen yang Diuji	Nama Kode Program	Hasil yang Diharapkan
1	Fungsi Login	<pre> @app.route('/login', methods=['GET', 'POST']) def login(): if request.method == 'POST': username = request.form['username'] password = request.form['password'] # Verifikasi pengguna dari database connection = get_db_connection() cursor = connection.cursor() # Tidak perlu dictionary=True karena menggunakan DictCursor cursor.execute("SELECT * FROM users WHERE username = %s", (username,)) user_data = cursor.fetchone() cursor.close() connection.close() if user_data and check_password_hash(user_data['password'], password): user = User(id=user_data['id'], username=user_data['username'], role=user_data['role']) login_user(user) return redirect(url_for('index')) # Redirect ke halaman utama else: return "Login gagal! Periksa username dan password." return render_template('login.html') </pre>	<p>Sistem memverifikasi username dan password yang valid, menampilkan dashboard pengguna jika login berhasil, atau pesan kesalahan jika gagal.</p>
Hasil Pengamatan			

			
2	Fungsi Logout	<pre>@app.route('/logout') @login_required def logout(): logout_user() return redirect(url_for('login'))</pre>	Sistem menghapus sesi pengguna dan mengarahkan ke halaman login.
Hasil Pengamatan Sistem berhasil menghapus sesi pengguna dan mengarahkan ke halaman login			
3	Fungsi Tambah Data	<pre>@app.route('/create', methods=['POST']) def create_data(): if request.method == 'POST': try: nama_pegawai = request.form['nama_pegawai'] nip = request.form['nip'] pangkat_golongan = request.form['pangkat_golongan'] jabatan = request.form['jabatan'] unit_kerja = request.form['unit_kerja'] jenis_temuan = request.form['jenis_temuan'] nilai_kerugian = request.form.get('nilai_kerugian', None) # Ambil nilai kerugian, default None jika kosong aspek_temuan = request.form['aspek_temuan'] status_penyelesaian_temuan = request.form['status_penyelesaian_temuan'] tindak_lanjut = request.form['tindak_lanjut']</pre>	Sistem berhasil menambahkan data baru ke database dan menampilkan data terbaru di halaman Surat Masuk.

```

        # Validasi bahwa input sesuai dengan
data di database
        connection = get_db_connection()
        cursor = connection.cursor()

        # Ambil kategori yang ada untuk
validasi
        cursor.execute("SELECT DISTINCT
jenis_temuan FROM dataset_temuan")
        jenis_temuan_options =
[ row['jenis_temuan'] for row in cursor.fetchall()]

        cursor.execute("SELECT DISTINCT
aspek_temuan FROM dataset_temuan")
        aspek_temuan_options =
[ row['aspek_temuan'] for row in cursor.fetchall()]

        cursor.execute("SELECT DISTINCT
tindak_lanjut FROM dataset_temuan")
        tindak_lanjut_options =
[ row['tindak_lanjut'] for row in cursor.fetchall()]

        # Validasi input
        if jenis_temuan not in
jenis_temuan_options:
            logging.error(f"Jenis temuan
'{jenis_temuan}' tidak valid.")
            jenis_temuan = None # Atur ke None
jika tidak valid

            if aspek_temuan not in
aspek_temuan_options:
                logging.error(f"Aspek temuan
'{aspek_temuan}' tidak valid.")
                aspek_temuan = None # Atur ke None
jika tidak valid

            if tindak_lanjut not in
tindak_lanjut_options:
                logging.error(f"Tindak lanjut
'{tindak_lanjut}' tidak valid.")
                tindak_lanjut = None # Atur ke
None jika tidak valid

        # Masukkan data ke dalam database
        cursor.execute("""

```

		<pre> INSERT INTO dataset_temuan (nama_pegawai, nip, pangkat_golongan, jabatan, unit_kerja, jenis_temuan, nilai_kerugian, aspek_temuan, status_penyelesaian_temuan, tindak_lanjut) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s) """', (nama_pegawai, nip, pangkat_golongan, jabatan, unit_kerja, jenis_temuan, nilai_kerugian, aspek_temuan, status_penyelesaian_temuan, tindak_lanjut)) # Ambil ID data yang baru saja ditambahkan new_data_id = cursor.lastrowid connection.commit() # Ambil data yang baru dimasukkan untuk prediksi cursor.execute("SELECT id, jenis_temuan, aspek_temuan, tindak_lanjut, status_penyelesaian_temuan FROM dataset_temuan WHERE id = %s", (new_data_id,)) new_data = cursor.fetchone() # Lakukan prediksi menggunakan Decision Tree model, label_encoders, _, _, _, _ = train_decision_tree() input_data = { 'jenis_temuan': new_data['jenis_temuan'], 'aspek_temuan': new_data['aspek_temuan'], 'tindak_lanjut': new_data['tindak_lanjut'], 'status_penyelesaian_temuan': new_data['status_penyelesaian_temuan'] } prediksi = predict_rekomendasi(model, label_encoders, input_data) # Simpan hasil prediksi ke database cursor.execute("UPDATE dataset_temuan SET prediksi = %s WHERE id = %s", (prediksi, new_data_id)) connection.commit() </pre>	
--	--	---	--

```

        cursor.close()
        connection.close()

        # Redirect ke halaman utama setelah
        data ditambahkan dan diprediksi
        return redirect(url_for('index'))
    except Exception as e:
        logging.error(f"Unexpected Error: {e}")
        # Tetap redirect ke halaman utama
        meskipun terjadi error
        return redirect(url_for('index'))

```

Hasil Pengamatan



Sistem berhasil menambah data baru pada tombol tersebut, dan pada baris dengan nama aziz adalah data yang telah di tambahkan.

4	Fungsi Edit Data	<pre> @app.route('/update/<int:id>', methods=['GET', 'POST']) def update_data(id): connection = get_db_connection() cursor = connection.cursor() if request.method == 'POST': # Ambil data dari form nama_pegawai = request.form['nama_pegawai'] nip = request.form['nip'] pangkat_golongan = request.form['pangkat_golongan'] jabatan = request.form['jabatan'] unit_kerja = request.form['unit_kerja'] jenis_temuan = request.form['jenis_temuan'] nilai_kerugian = request.form['nilai_kerugian'] aspek_temuan = request.form['aspek_temuan'] status_penyelesaian_temuan = request.form['status_penyelesaian_temuan'] </pre>	Sistem berhasil mengupdate data yang telah diubah pengguna dan menyimpan pembaruan ke database.
---	------------------	--	---

		<pre> tindak_lanjut = request.form['tindak_lanjut'] # Update data ke database cursor.execute(""" UPDATE dataset_temuan SET nama_pegawai = %s, nip = %s, pangkat_golongan = %s, jabatan = %s, unit_kerja = %s, jenis_temuan = %s, nilai_kerugian = %s, aspek_temuan = %s, status_penyelesaian_temuan = %s, tindak_lanjut = %s WHERE id = %s """, (nama_pegawai, nip, pangkat_golongan, jabatan, unit_kerja, jenis_temuan, nilai_kerugian, aspek_temuan, status_penyelesaian_temuan, tindak_lanjut, id)) connection.commit() cursor.close() connection.close() # Redirect ke halaman utama setelah update return redirect(url_for('all_data')) # Jika method GET, ambil data yang akan diedit cursor.execute("SELECT * FROM dataset_temuan WHERE id = %s", (id,)) temuan_data = cursor.fetchone() # Ambil data dropdown cursor.execute("SELECT DISTINCT jenis_temuan FROM dataset_temuan") jenis_temuan_options = [row['jenis_temuan'] for row in cursor.fetchall()] cursor.execute("SELECT DISTINCT aspek_temuan FROM dataset_temuan") aspek_temuan_options = [row['aspek_temuan'] for row in cursor.fetchall()] cursor.execute("SELECT DISTINCT tindak_lanjut FROM dataset_temuan") </pre>	
--	--	---	--

```

tindak_lanjut_options = [row['tindak_lanjut']
for row in cursor.fetchall()]

cursor.close()
connection.close()

if not temuan_data:
    return "Data tidak ditemukan!", 404

# Render template form dengan data temuan dan
opsi dropdown
return render_template(
    'form.html',
    temuan_data=temuan_data,
    jenis_temuan_options=jenis_temuan_options,
    aspek_temuan_options=aspek_temuan_options,
    tindak_lanjut_options=tindak_lanjut_options
)

```

Hasil Pengamatan

314	Azis Hirto	2021071991	I/c	Guru	Kabupaten Dompu	Defisit Anggaran	Keuangan	Pengembalian dalam proses	Tidak Layak	Edit Hapus
-----	------------	------------	-----	------	-----------------	------------------	----------	---------------------------	-------------	---

Perbarui Data Temuan

Kembali

Nama Pegawai
Azis Hirto

NIP
2021071991

Pangkat/Golongan
I/c

Jabatan
Guru IPS

Unit Kerja
Kabupaten Dompu

Nilai Kerugian
None

Status Penyelesaian Temuan
Dalam Proses

Jenis Temuan
Defisit Anggaran

Aspek Temuan
Keuangan

Tindak Lanjut
Pengembalian dalam proses

Perbarui Data

314	Azis Hirto	2021071991	I/c	Guru IPS	Kabupaten Dompu	Defisit Anggaran	Keuangan	Pengembalian dalam proses	Tidak Layak	Edit Hapus
-----	------------	------------	-----	----------	-----------------	------------------	----------	---------------------------	-------------	---

Data tersebut berhasil terupdate pada kolom yang saya ingin ubah contohnya pada kolom jabatan.

5 Fungsi Hapus Data

@app.route('/delete/<int:id>')

Sistem berhasil menghapus data yang dipilih dari database dan memperbarui tabel Surat Masuk.

Hasil Pengamatan

314	Azis Hirto	2021071991	I/c	Guru	Kabupaten Dompu	Defisit Anggaran	Keuangan	Pengembalian dalam proses	Tidak Layak	Edit Hapus
-----	------------	------------	-----	------	-----------------	------------------	----------	---------------------------	-------------	---

SKBT Decision Tree										
301	Dr. Drs. H. Abdul Rosyid, M.Pd	196803241989031001	IV/b	Kepala Seksi Pendidikan Diniyah dan Pondok Pesantren	Kota Jakarta Barat	Indisipliner	Kepegawaian	Disiplin internal	Layak	Edisi Hapus
302	Dr. H.Kardi, S.Ag., M.Hum.	197210311999031000	IV/b	Pustakawan Ahli Madya	IAIN	Pembayaran Honor Tidak Sesuai	Keuangan	Dana dikembalikan	Layak	Edisi Hapus
303	Yosi Widayati, S.Pd.I	198212052009012000	III/c	Pengadministrasi pada Biro Umum Sekretariat Jenderal Kementerian Agama	Biro Umum Sekretariat Jenderal Kementerian Agama	Kelebihan Pembayaran Tunjangan Kinerja	Keuangan	Dana dikembalikan	Layak	Edisi Hapus
304	Nurul Huda, S.Kom., M.Pd.I	198108242005011000	IV/a	Pengembang Teknologi Pembelajaran Ahli Muda	UIN Kiai Haji Achmad Siddiq Jember	Pembayaran Honor Tidak Sesuai	Keuangan	Dana dikembalikan	Layak	Edisi Hapus
305	Isnayani Saparing, S.Ag	197302121996032001	IV/a	Guru Ahli Madya pada MAN Tanah Bumbu	Kabupaten Tanah Bumbu	Indisipliner	Kepegawaian	Disiplin internal	Tidak Layak	Edisi Hapus
306	Fitriana Aenun, S.Pd., M.Ed.	197106241997032001	IV/a	Guru Madya/Kepala MISN 3 Purworejo	Kabupaten Purworejo	Indisipliner	Kepegawaian	Disiplin internal	Tidak Layak	Edisi Hapus
308	Dr. H. Syaifuddin, MA	196211071989031001	IV/b	Analisis Kepegawaian Ahli Madya	Setditjen Pendidikan Islam	Pembayaran Honor Tidak Sesuai	Keuangan	Proses pengembalian	Tidak Layak	Edisi Hapus

Data berhasil dilakukan penghapusan dari nama aziz.

6	Fungsi Pencarian Data	<pre> @app.route('/all_data', methods=['GET', 'POST']) @login_required @role_required('admin_satker') def all_data(): connection = get_db_connection() cursor = connection.cursor() # Cek apakah ada pencarian search_query = request.form.get('search_name', '') if search_query: cursor.execute("SELECT * FROM dataset_temuan WHERE nama_pegawai LIKE %s", ('%' + search_query + '%',)) else: cursor.execute("SELECT * FROM dataset_temuan") all_data = cursor.fetchall() cursor.close() connection.close() return render_template('all_data.html', all_data=all_data, search_query=search_query) </pre>	Sistem menampilkan data sesuai dengan kata kunci pencarian dari pengguna.
Hasil Pengamatan			

SKBT Decision Tree

Dashboard

Surat Masuk

Surat Keluar

Logout

Surat Masuk

Tambah Data Baru

Cari berdasarkan nama pegawai Cari

Nama ID Pegawai	NIP	Pangkat/Golongan	Jabatan	Unit Kerja	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Prediksi	Aksi	
1	Hamsi Djamsi	196102101986031000	III/c	Kepala Urusan Tata Usaha MAN Insan Cendikia Kabupaten Halmahera Barat Provinsi Maluku Utara	Kandepag Ternate	Defisit Anggaran	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak	Edit Hapus
2	Drs. H. AMIR TOMAGOLA, MMMH	196712081995031000	IV/b	Kepala Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara	Kanwil Maluku Utara	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Layak	Edit Hapus
3	ABD RAHMAN KHARIE, S.Ag	195903201986031000	IV/b	Pegawai Subbagian Tata Usaha Sekretaris Jenderal Bagian Tata Usaha Pimpinan Biro Umum Sekretariat Jenderal	Kanwil Maluku Utara	Ketidakpatuhan Prosedur	Tugas dan Fungsi	Laporan belum diterima	Tidak Layak	Edit Hapus

hartati Cari

Nama ID Pegawai	NIP	Pangkat/Golongan	Jabatan	Unit Kerja	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Prediksi	Aksi	
8	HARTATI TASWA S.Pd.I	197510212006042002	III/b	Analisis Informasi Pendidikan (Jabatan Pelaksana) pada Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara	Kanwil Halmahera Barat	Kelebihan Pembayaran Honor	Keuangan	Pengembalian kelebihan pembayaran	Layak	Edit Hapus

Sistem berhasil melakukan pencarian dengan nama atau kata kunci yaitu hartati.

7	Fungsi Prediksi Rekomendasi	<pre style="font-family: monospace; color: green;"># Fungsi untuk melakukan prediksi def predict_rekomendasi(model, label_encoders, input_data): # Encode input data menggunakan label encoders for column, encoder in label_encoders.items(): if column in input_data: input_data[column] = encoder.transform([input_data[column]])[0] # Convert input data to DataFrame for prediction input_df = pd.DataFrame([input_data]) prediction = model.predict(input_df) return 'Setuju' if prediction[0] == 1 else 'Tidak Setuju'</pre>	Sistem berhasil memprediksi hasil rekomendasi menggunakan Decision Tree dan menyimpan hasil ke database.
---	-----------------------------	---	--

Hasil Pengamatan

Data Test (Tanpa Rekomendasi)

ID	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Prediksi
290	Defisit Anggaran	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak
291	Pembayaran Honor Tidak Sesuai	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak
292	Indisipliner	Keuangan	Proses pengembalian	Tidak Layak
293	Pembayaran Honor Tidak Sesuai	Keuangan	Proses pengembalian	Tidak Layak
298	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Tidak Layak
299	Kelebihan Pembayaran Tunjangan Kinerja	Keuangan	Proses pengembalian	Tidak Layak
300	Indisipliner	Kepegawaian	Disiplin internal	Layak
301	Indisipliner	Kepegawaian	Disiplin internal	Layak
302	Pembayaran Honor Tidak Sesuai	Keuangan	Dana dikembalikan	Layak
303	Kelebihan Pembayaran Tunjangan Kinerja	Keuangan	Dana dikembalikan	Layak
304	Pembayaran Honor Tidak Sesuai	Keuangan	Dana dikembalikan	Layak
305	Indisipliner	Kepegawaian	Disiplin internal	Tidak Layak
306	Indisipliner	Kepegawaian	Disiplin internal	Tidak Layak
308	Pembayaran Honor Tidak Sesuai	Keuangan	Proses pengembalian	Tidak Layak

Data Training

ID	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Rekomendasi	Prediksi
1	Defisit Anggaran	Keuangan	Sebagian dana telah dikembalikan	Tidak Layak	Tidak Layak
2	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Layak	Layak
3	Ketidakpatuhan Prosedur	Tugas dan Fungsi	Laporan belum diterima	Tidak Layak	Tidak Layak
4	Indisipliner	Tugas dan Fungsi	Teguran internal diberikan	Tidak Layak	Tidak Layak
5	Ketidakpatuhan Prosedur Nikah	Tugas dan Fungsi	Teguran tertulis	Layak	Layak
6	Pengangkatan PNS Tidak Sah	Kepegawaian	Pemberhentian dengan hormat	Tidak Layak	Layak
7	Pengabaian Pembayaran Gaji	Kepegawaian	Pembayaran dihentikan	Layak	Layak
8	Kelebihan Pembayaran Honor	Keuangan	Pengembalian kelebihan pembayaran	Layak	Layak
9	Kelebihan Pembayaran Honor	Keuangan	Dana telah disetor ke kas negara	Layak	Layak
10	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
11	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
12	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
13	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
14	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
15	Kelebihan Pembayaran Honor	Keuangan	Dana telah dikembalikan	Layak	Layak
16	Double Account Perjalanan Dinas	Keuangan	Dana telah dikembalikan	Layak	Layak

8

Fungsi Menampilkan evaluasi model Algoritma Decision Tree di Dashboard

```
def train_decision_tree():
    connection = get_db_connection()
    with connection.cursor() as cursor:
        cursor.execute("SELECT id, jenis_temuan,
        aspek_temuan, tindak_lanjut,
        status_penyelesaian_temuan, rekomendasi_skbt FROM
        dataset_temuan")
        data = cursor.fetchall()

        df = pd.DataFrame(data)

        # Pisahkan data training dan testing
        train_data =
        df[df['rekomendasi_skbt'].notna()] # Data dengan
        rekomendasi
```

Sistem menampilkan metrik evaluasi (akurasi, precision, recall, F1-score), confusion matrix, serta tabel data training dan test.

```

test_data =
df[df['rekomendasi_skbt'].isna()] # Data tanpa
rekomendasi

if train_data.empty:
    raise ValueError("Tidak ada data untuk
training.")

# Encode data kategorikal
label_encoders = {}
for column in ['jenis_temuan', 'aspek_temuan',
'tindak_lanjut', 'status_penyelesaian_temuan']:
    le = LabelEncoder()
    train_data[column] =
le.fit_transform(train_data[column])
    if not test_data.empty:
        test_data[column] =
le.transform(test_data[column])
        label_encoders[column] = le

# Siapkan data training
X = train_data[['jenis_temuan', 'aspek_temuan',
'tindak_lanjut', 'status_penyelesaian_temuan']]
y =
train_data['rekomendasi_skbt'].map({'Layak': 1,
'Tidak Layak': 0})

# Split data menjadi training dan test untuk
evaluasi internal
X_train, X_val, y_train, y_val =
train_test_split(X, y, test_size=0.2,
random_state=42)

# Latih model
model = DecisionTreeClassifier(max_depth=5,
random_state=42)
model.fit(X_train, y_train)

# Prediksi seluruh data training
train_data['prediksi'] = model.predict(X)

# Ubah nilai numerik ke label string
train_data['prediksi'] =
train_data['prediksi'].map({1: 'Layak', 0: 'Tidak
Layak'})

# Evaluasi model

```

		<pre> val_predictions = model.predict(X_val) val_accuracy = accuracy_score(y_val, val_predictions) # Perhitungan metrik tambahan confusion = confusion_matrix(y_val, val_predictions) precision = precision_score(y_val, val_predictions, zero_division=1) recall = recall_score(y_val, val_predictions, zero_division=1) f1 = f1_score(y_val, val_predictions, zero_division=1) # Prediksi untuk data baru if not test_data.empty: X_test_new = test_data[['jenis_temuan', 'aspek_temuan', 'tindak_lanjut', 'status_penyelesaian_temuan']] test_data['prediksi'] = model.predict(X_test_new) test_data['prediksi'] = test_data['prediksi'].map({1: 'Layak', 0: 'Tidak Layak'}) # Simpan prediksi ke database with connection.cursor() as cursor: for _, row in test_data.iterrows(): cursor.execute(""" UPDATE dataset_temuan SET prediksi = %s WHERE id = %s """, (row['prediksi'], row['id'])) connection.commit() connection.close() # Decode data sebelum dikembalikan for column, encoder in label_encoders.items(): train_data[column] = encoder.inverse_transform(train_data[column]) if not test_data.empty: test_data[column] = encoder.inverse_transform(test_data[column]) </pre>	
--	--	--	--

		<pre>return model, label_encoders, train_data, test_data, val_accuracy, confusion, precision, recall, f1</pre>	
--	--	--	--

Hasil Pengamatan

Dashboard

Akurasi Model

Akurasi Training:

Akurasi Validation: 0.9827586206896551

Metrik Evaluasi

Precision: 0.9772727272727273

Recall: 1.0

F1-Measure: 0.9885057471264368

Confusion Matrix

	Prediksi: Tidak Layak	Prediksi: Layak
Aktual: Tidak Layak	14	1
Aktual: Layak	0	43

Sistem berhasil menampilkan hasil evaluasi model berupa akurasi model, metrik akurasi, dan *confusion matrix*.

9	Fungsi Download Surat Keluar	<pre>@app.route('/download/<int:id>', methods=['GET']) def download_pdf(id): connection = get_db_connection() cursor = connection.cursor() # Ambil data berdasarkan ID cursor.execute("SELECT * FROM dataset_temuan WHERE id = %s AND rekomendasi_skbt = 'Layak'", (id,)) data = cursor.fetchone() cursor.close() connection.close() if not data: return "Data tidak ditemukan atau tidak layak!", 404 # Buat PDF pdf_path = create_pdf(data) # Kirim file ke pengguna return send_file(pdf_path, as_attachment=True)</pre>	Sistem menghasilkan dan mengunduh file PDF untuk data pegawai yang dipilih.
---	------------------------------------	---	---

Hasil Pengamatan

Surat Keluar

Download Selected

Select	ID	Nama Pegawai	NIP	Pangkat/Golongan	Jabatan	Unit Kerja	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Rekomendasi	Aksi
<input checked="" type="checkbox"/>	2	Drs. H. AMIR TOMAGOLA, MM,MH	196712081995031000	IV/b	Kepala Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara	Kanwil Maluku Utara	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Layak	Download



**KEMENTERIAN AGAMA REPUBLIK INDONESIA
INSPEKTORAT JENDERAL**

Jalan RS Fatmawati Nomor 33 A, Cipete, Jakarta 12420
Telepon (021) 75910638, 75875533, 3818498 Fax: (021) 7629112
Website: www.itjen.kemenag.go.id

SURAT KETERANGAN BEBAS TEMUAN

NOMOR: R/2/KP.01/01/2025

Yang bertanda tangan di bawah ini:

Nama : Kastolan
NIP : 197004202000121002
Golongan : Pembina Utama Muda/IVc
Jabatan : Sekretaris Inspektorat Jenderal Kementerian Agama

Dengan ini menerangkan bahwa:

Nama : Drs. H. AMIR TOMAGOLA, MM,MH
NIP : 196712081995031000
Pangkat/Golongan : IV/b
Jabatan : Kepala Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara
Unit Kerja : Kanwil Maluku Utara

Berdasarkan database pengelolaan hasil pengawasan Inspektorat Jenderal Kementerian Agama, dengan ini menerangkan bahwa pegawai di atas dinyatakan bebas dari temuan administrasi dan keuangan.

Jakarta, 22-01-2025
a.n. Inspektur Jenderal
Sekretaris,
Dalman, S.Pd., M.Pd.

Tembusan: Inspektur Jenderal Kementerian Agama

Sistem berhasil mengunduh surat yang dipilih.

10	Fungsi Download Multiple Surat Keluar	<pre> @app.route('/download_selected', methods=['POST']) def download_selected_pdfs(): selected_ids = request.form.getlist('selected_ids') if not selected_ids: return "Tidak ada data yang dipilih!", 400 connection = get_db_connection() cursor = connection.cursor() # Ambil data berdasarkan ID yang dipilih format_strings = ','.join(['%s'] * len(selected_ids)) cursor.execute(f"SELECT * FROM dataset_temuan WHERE id IN ({format_strings}) AND rekomendasi_skbt = 'Layak'", tuple(selected_ids)) selected_data = cursor.fetchall() cursor.close() connection.close() if not selected_data: return "Data tidak ditemukan!", 404 # Buat folder sementara untuk menyimpan PDF temp_folder = os.path.join(app.config['UPLOAD_FOLDER'], 'temp_pdfs') if not os.path.exists(temp_folder): os.makedirs(temp_folder) # Buat PDF untuk setiap data pdf_paths = [] for data in selected_data: pdf_path = create_pdf(data, filename=f"SKBT_{data['nama_pegawai']}.pdf") pdf_paths.append(pdf_path) # Buat file ZIP zip_path = os.path.join(app.config['UPLOAD_FOLDER'], 'SKBT_Selected.zip') with zipfile.ZipFile(zip_path, 'w') as zipf: for pdf_path in pdf_paths: zipf.write(pdf_path, os.path.basename(pdf_path)) </pre>	Sistem menghasilkan file ZIP berisi PDF untuk beberapa data pegawai yang dipilih.
----	--	--	---

```
# Hapus file PDF sementara
for pdf_path in pdf_paths:
    os.remove(pdf_path)


# Kirim file ZIP ke pengguna
return send_file(zip_path, as_attachment=True)
```

Hasil Pengamatan


Surat Keluar

Download Selected


Select	ID	Nama Pegawai	NIP	Pangkat/Golongan	Jabatan	Unit Kerja	Jenis Temuan	Aspek Temuan	Tindak Lanjut	Rekomendasi	Aksi
<input checked="" type="checkbox"/>	2	Drs. H. AMIR TOMAGOLA, MM,MH	196712081995031000	IV/b	Kepala Kantor Kementerian Agama Kabupaten Halmahera Barat Provinsi Maluku Utara	Kanwil Maluku Utara	Penyalahgunaan Wewenang	Tugas dan Fungsi	Disiplin internal	Layak	Download
<input checked="" type="checkbox"/>	5	Drs. KAMAL DO HAMZAH	196010141991031000	II/d	Pengadministrasi Sub Bagian Tata Usaha Kantor Kementerian Agama Kota Ternate Provinsi Maluku	Kanwil Maluku Utara	Ketidakpatuhan Prosedur Nikah	Tugas dan Fungsi	Teguran tertulis	Layak	Download
<input checked="" type="checkbox"/>	7	Drs. NASARUDDIN ALY, MM	196804131995031000	IV/b	Kepala Kantor Kementerian Agama Kabupaten Pulau Taliabu Provinsi Maluku Utara	Kanwil Halmahera Barat	Pengabaian Pembayaran Gaji	Kepegawaian	Pembayaran dihentikan	Layak	Download



SKBT_Drs. H. AMIR TOMAGOLA, MM,MH



SKBT_Drs. KAMAL DO HAMZAH



SKBT_Drs. NASARUDDIN ALY, MM

Sistem berhasil mengunduh file secara multiple sesuai dengan yang dipilih.

5.2 Pembahasan

Penelitian ini berfokus pada pengembangan aplikasi berbasis web untuk mendukung penerbitan SKBT dengan menggunakan pendekatan *Algoritma Decision Tree*. Permasalahan utama yang diidentifikasi adalah proses penerbitan SKBT yang masih dilakukan secara manual, yang berpotensi menimbulkan ketidakakuratan dalam penilaian dan inefisiensi waktu. Untuk mengatasi permasalahan tersebut, penelitian ini merancang sebuah aplikasi yang mampu menganalisis data temuan secara sistematis dan memprediksi kelayakan penerbitan

SKBT dengan lebih akurat dan efisien. Aplikasi ini dirancang untuk memanfaatkan data historis dari Inspektorat Jenderal Kementerian Agama RI sebagai dasar pengambilan keputusan, sementara ruang lingkup penelitian dibatasi hanya pada penerapan *Algoritma Decision Tree* tanpa membahas algoritma lainnya. Implementasi algoritma ini menunjukkan keunggulannya dalam melakukan klasifikasi data melalui proses yang mencakup unggah file, *preprocessing data*, evaluasi model dengan *confusion matrix*, dan penyajian tabel klasifikasi hasil prediksi. Berdasarkan hasil evaluasi, akurasi model mencapai 98.28%, dengan tingkat *precision* dan *recall* yang tinggi untuk kategori "Layak", meskipun performa untuk kategori "Tidak Layak" menunjukkan perlunya pengembangan lebih lanjut. Selain itu, fitur seperti penomoran surat otomatis, pengunduhan dokumen PDF untuk pegawai yang layak dirancang untuk meningkatkan kemudahan penggunaan. Pengujian *Black box* berfungsi untuk memastikan fungsionalitas aplikasi sesuai dengan spesifikasi yang telah ditentukan, serta metode *white box* untuk menguji jalur logika program secara mendalam. Penelitian ini menunjukkan bahwa aplikasi dapat meningkatkan efisiensi serta akurasi dalam proses penerbitan SKBT, sekaligus memberikan kontribusi yang signifikan terhadap pengembangan teknologi berbasis algoritma dalam konteks pemerintahan.