

BAB IV PERANCANGAN

4.1 Analisis Sistem Terdahulu

Pada metode sebelumnya, proses pengelolaan data temuan dilakukan secara manual menggunakan file Excel atau dokumen tertulis. Hal ini menyebabkan proses analisis data dan penerbitan Surat Keterangan Bebas Temuan (SKBT) menjadi memakan waktu lebih lama, rentan terhadap kesalahan manusia, dan sulit untuk diakses kembali di masa mendatang. Sistem manual ini juga tidak memiliki integrasi untuk pengelolaan data temuan dan pengambilan keputusan berbasis algoritma.

Proses manual penerbitan SKBT dimulai dengan pemohon yang mengajukan permohonan kepada satuan kerja terkait. Satuan kerja kemudian memeriksa dan memproses pengajuan tersebut. Jika terdapat temuan, surat tidak dapat diproses hingga semua temuan diselesaikan. Apabila tidak ada temuan, SKBT dapat diproses lebih lanjut dan disetujui oleh sekretaris sebelum surat diunduh dan diberikan kepada pemohon. Proses ini menunjukkan perlunya sistem terintegrasi yang tidak hanya mempercepat pengelolaan data temuan, tetapi juga mengurangi potensi kesalahan, memastikan akurasi, dan mempermudah akses data untuk keperluan pengambilan keputusan di masa depan.

4.2 Spesifikasi Kebutuhan Sistem Baru

Aplikasi Surat Keterangan Bebas Temuan ini membutuhkan beberapa spesifikasi khusus agar berjalan dengan baik, spesifikasi yang dibutuhkan aplikasi yaitu berupa software dan hardware.

4.2.1 Spesifikasi Kebutuhan Perangkat Lunak

Berikut adalah komponen yang perlu disiapkan untuk mendukung kelancaran sebuah penelitian. Berikut adalah rincian yang dibutuhkan.

Tabel 4.1 Tabel Spesifikasi Kebutuhan Perangkat Lunak

No.	Perangkat	Keterangan
1.	Windows 10	Merupakan os untuk mendukung proses penelitian.
2.	Figma	Sebuah platform yang dimanfaatkan untuk mendesain antarmuka aplikasi.
3.	Visual Studio Code	Sebuah editor kode untuk membuat aplikasi SKBT.

4.	Web Browser	Sebuah platform yang untuk mencari informasi.
----	-------------	---

4.2.2 Spesifikasi Kebutuhan Perangkat Keras

Berikut yang dibutuhkan dalam melakukan penelitian mencakup :

Tabel 4.2 Spesifikasi Kebutuhan Perangkat Keras

No.	Hardware	Spesifikasi
1.	<i>Processor</i>	Intel i7-8750H
2.	RAM	16 GB
3.	VGA	Nvidia Gforce GTX 1050Ti

4.2.3 Spesifikasi Kebutuhan Input

Dalam membuat aplikasi diperlukan untuk memastikan kinerja aplikasi berjalan secara optimal, maka diperlukan sebuah proses input. Berikut adalah beberapa proses yang dibutuhkan oleh aplikasi Surat Keterangan Bebas Temuan.

1. Input Data Temuan

Aplikasi ini menginput kedalam database dengan melakukan input pada menu surat masuk dengan kolom yang dibutuhkan.

4.2.4 Spesifikasi Kebutuhan Output

Hasil pada aplikasi ini berupa evaluasi model decision tree dan sebuah dokumen SKBT. Berikut adalah spesifikasi output aplikasi Surat Keterangan Bebas Temuan.

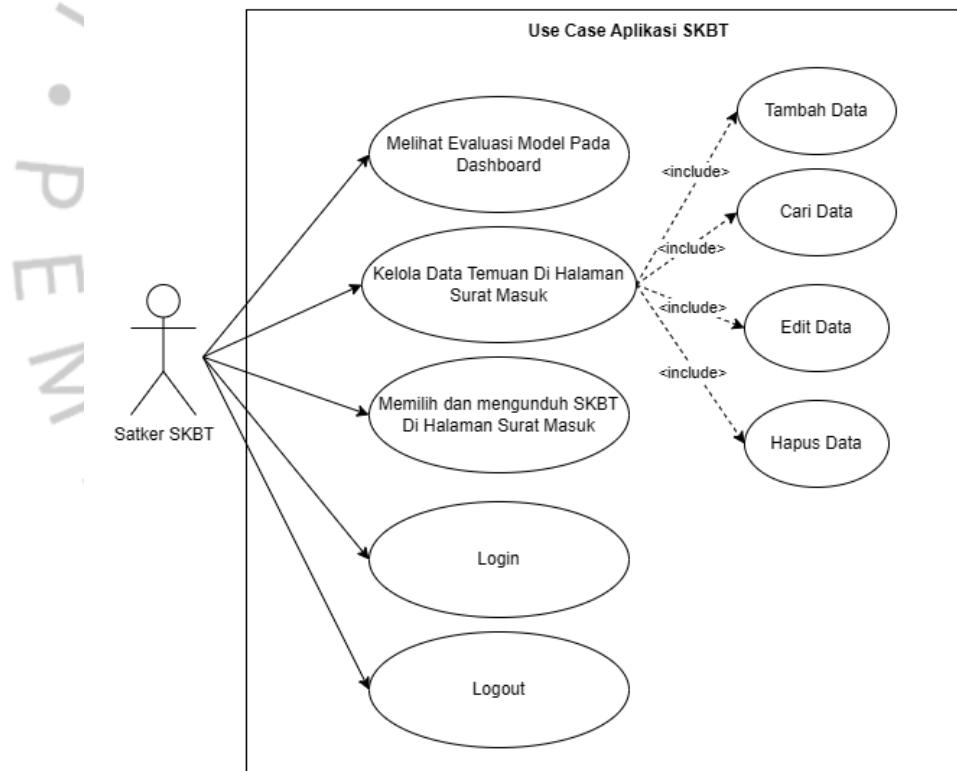
1. Hasil evaluasi model *Algoritma Decision Tree* berupa akurasi, confusion matrix dan tabel prediksi *data training* dan *data testing*.
2. Dokumen SKBT dengan format PDF.
3. Penomoran surat otomatis berdasarkan penomoran surat yang baku.
4. Nama pegawai, NIP, Pangkat/Jabatan, Unit Kerja otomatis terisi pada dokumen pdf.
5. Dokumen pdf akan tampil dan bisa diunduh jika rekomendasi dinyatakan sebagai layak.

4.3 Perancangan Sistem

Tujuan ini untuk mengidentifikasi kebutuhan sistem agar dapat merancang aplikasi secara komprehensif. Proses perancangan ini mencakup penguraian secara rinci mengenai tahapan-tahapan, desain aplikasi, hingga perancangan metode pengujian. Dalam perancangan sistem, digunakan berbagai macam diagram, seperti diagram *flowchart*, *use case*, *activity diagram*, dan desain antarmuka pengguna. Ini memiliki fungsi untuk mendukung pemahaman terhadap setiap proses.

4.3.1 Use Case Diagram

Ini adalah gambaran alur untuk menunjukkan hubungan antara Satker SKBT dengan sistem dalam aplikasi Surat Keterangan Bebas Temuan. Berikut adalah penjelasan dari use case diagram proses data temuan.



Gambar 4.1 Use Case Diagram Proses Data Temuan

Berikut menjelaskan proses interaksi pengguna yaitu Satker SKBT dengan aplikasi SKBT, yang terdiri dari beberapa fitur utama. Pengguna dapat login untuk mendapatkan akses ke aplikasi dan logout saat selesai menggunakan aplikasi. Setelah login, pengguna dapat melihat evaluasi model pada dashboard untuk

memahami kinerja model yang digunakan. Selain itu, pengguna dapat mengelola data temuan di halaman Surat Masuk dengan kemampuan untuk menambah, mengedit, dan menghapus data yang diperlukan. Fitur lainnya adalah kemampuan untuk memilih dan mengunduh Surat Keterangan Bebas Temuan (SKBT) di halaman Surat Masuk, di mana pengguna dapat memilih data tertentu dan menghasilkan file yang relevan untuk diunduh. Diagram ini mengilustrasikan semua fungsi utama yang dapat dilakukan pengguna dalam aplikasi secara efisien.



4.3.2 Skenario Usecase

Tabel 4.3 Login

No.	Nama Use Case	Login
1	Aktor	Satker SKBT
2	Deskripsi	Pengguna mengakses sistem untuk masuk ke aplikasi dengan memasukkan kredensial yang valid.
3	Tindakan	<ol style="list-style-type: none"> 1. Satker <i>input username dan password</i>. 2. Sistem memvalidasi kredensial. 3. Jika valid, pengguna diarahkan ke dashboard utama. Jika tidak valid, sistem memberikan pesan kesalahan.

Tabel 4.4 Melihat Evaluasi Model pada Dashboard

No.	Nama Use Case	Melihat Evaluasi Model pada Dashboard
1	Aktor	Satker SKBT
2	Deskripsi	Pengguna dapat melihat hasil evaluasi model algoritma decision tree yang ditampilkan pada dashboard utama.
3	Tindakan	<ol style="list-style-type: none"> 1. Pengguna yang sudah login membuka halaman dashboard. 2. Sistem menampilkan evaluasi model, termasuk akurasi, precision, recall, dan f1-score. 3. Pengguna melihat informasi evaluasi.

Tabel 4.5 Mengelola Data Temuan di Halaman Surat Masuk

No.	Nama Use Case	Mengelola Data Temuan di Halaman Surat Masuk
1	Aktor	Satker SKBT
2	Deskripsi	Pengguna dapat menambah, mengedit, atau menghapus data temuan.
3	Tindakan	<ol style="list-style-type: none"> 1. Pengguna membuka halaman "Surat Masuk". 2. Pengguna memilih salah satu Tindakan: <ul style="list-style-type: none"> • Menambah data temuan • Mengedit data temuan • Mencari data temuan • Menghapus data temuan 3. Sistem memproses dan menyimpan perubahan.

Tabel 4.6 Memilih dan Mengunduh SKBT di Halaman Surat Masuk

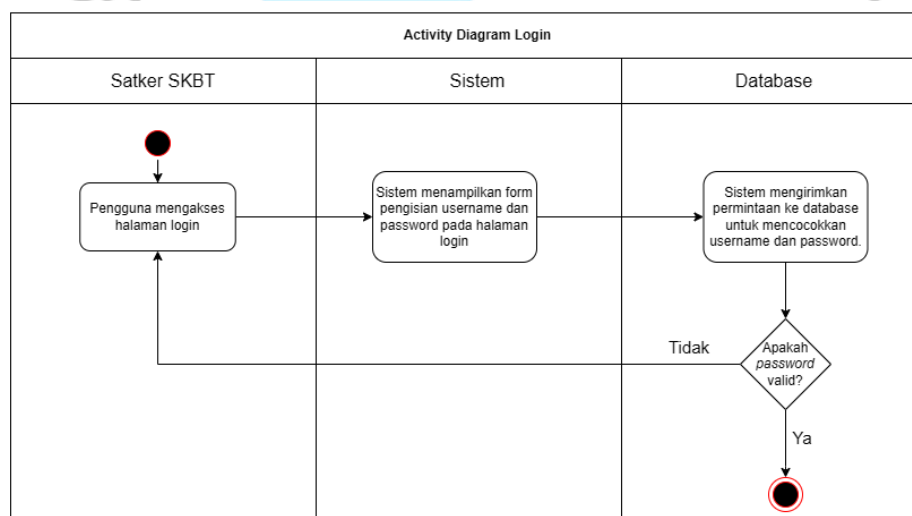
No.	Nama Use Case	Memilih dan Mengunduh SKBT di Halaman Surat Masuk
1	Aktor	Satker SKBT
2	Deskripsi	Pengguna dapat memilih satu atau beberapa data SKBT untuk diunduh dalam format PDF.
3	Tindakan	1.Pengguna membuka halaman "Surat Masuk". 2.Pengguna mencentang kotak data SKBT yang ingin diunduh. 3.Pengguna menekan tombol "Download Selected". 4.Sistem memproses dan memberikan file ZIP berisi PDF yang diunduh.

Tabel 4.7 Logout

No.	Nama Use Case	Logout
1	Aktor	Satker SKBT
2	Deskripsi	Pengguna keluar dari sistem aplikasi SKBT.
3	Tindakan	1.Pengguna menekan tombol "Logout". 2.Sistem menghapus sesi pengguna. 3.Pengguna diarahkan ke halaman login.

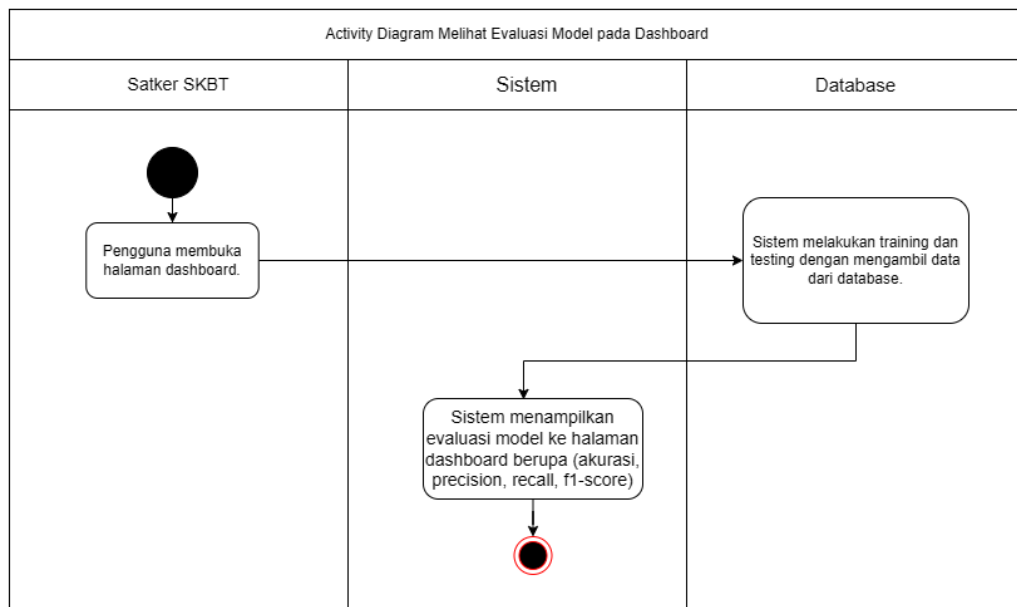
4.3.2 Activity Diagram

Dalam pengembangan aplikasi Surat Keterangan Bebas Temuan, *activity diagram* digunakan untuk menjelaskan urutan sistematis dari aktifitas yang telah dijelaskan dalam *use case diagram* sebelumnya, Berikut adalah *activity diagram* untuk aplikasi Surat Keterangan Bebas Temuan.



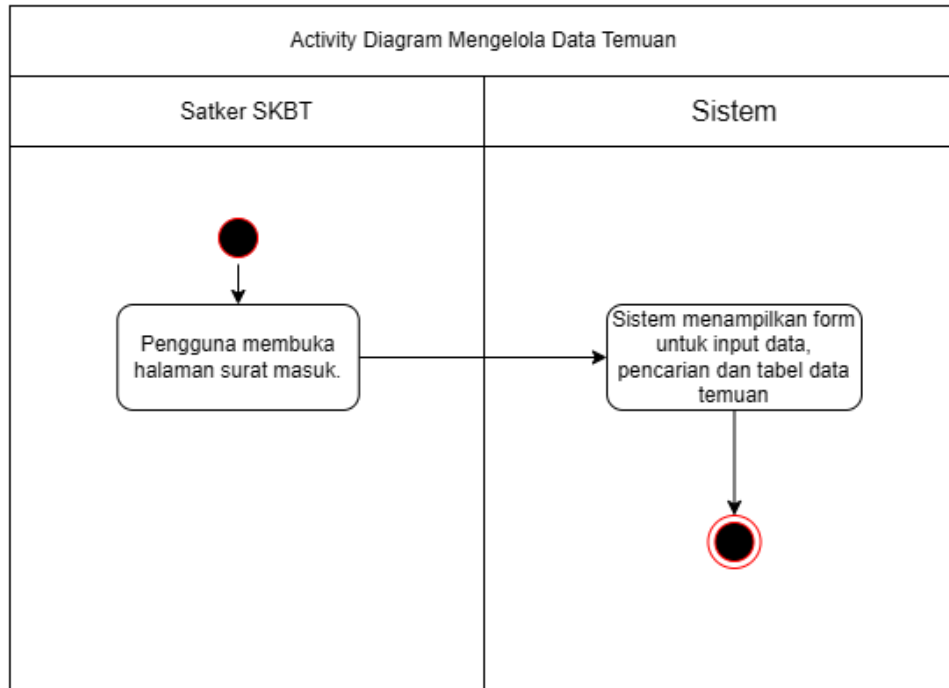
Gambar 4.2 Activity Diagram Login

Berikut adalah proses autentikasi di aplikasi SKBT. Dimulai dengan Satker SKBT mengakses halaman login, di mana sistem menampilkan form untuk pengisian username dan password. Setelah pengguna mengisi form dan mengirimkan permintaan login, sistem meneruskan informasi tersebut ke database untuk diverifikasi. Database akan menyesuaikan nama pengguna dan kata kunci. Misal kata kunci tidak valid, sistem akan mengembalikan pesan error atau meminta pengguna untuk mencoba lagi. Jika validasi berhasil, sistem mengautentikasi pengguna dan memberikan akses ke aplikasi. Proses ini diakhiri dengan pengguna berhasil login ke dalam sistem.



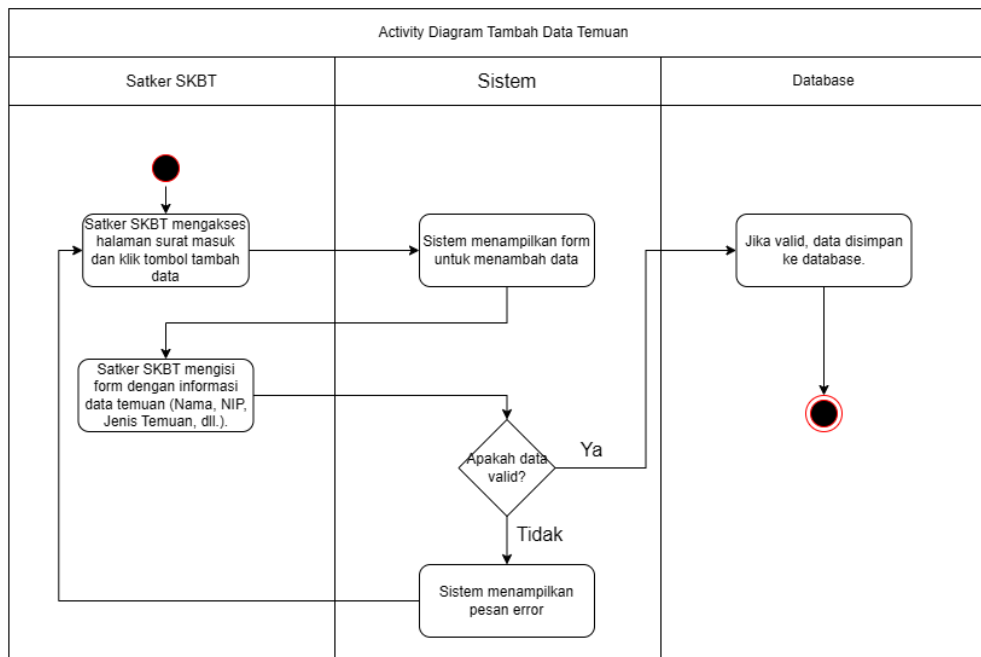
Gambar 4.3 Activity Diagram Melihat Evaluasi Model pada Dashboard

Activity Diagram ini menggambarkan alur proses pengguna dalam melihat evaluasi model pada dashboard aplikasi SKBT. Proses dimulai dengan Satker SKBT membuka halaman dashboard. Sistem kemudian melakukan proses training dan testing dengan mengambil data dari database untuk mengevaluasi model klasifikasi yang digunakan. Setelah proses evaluasi selesai, sistem menampilkan hasil evaluasi model di halaman dashboard berupa metrik yang ada. Proses ini membantu pengguna memahami performa model secara langsung dari dashboard.



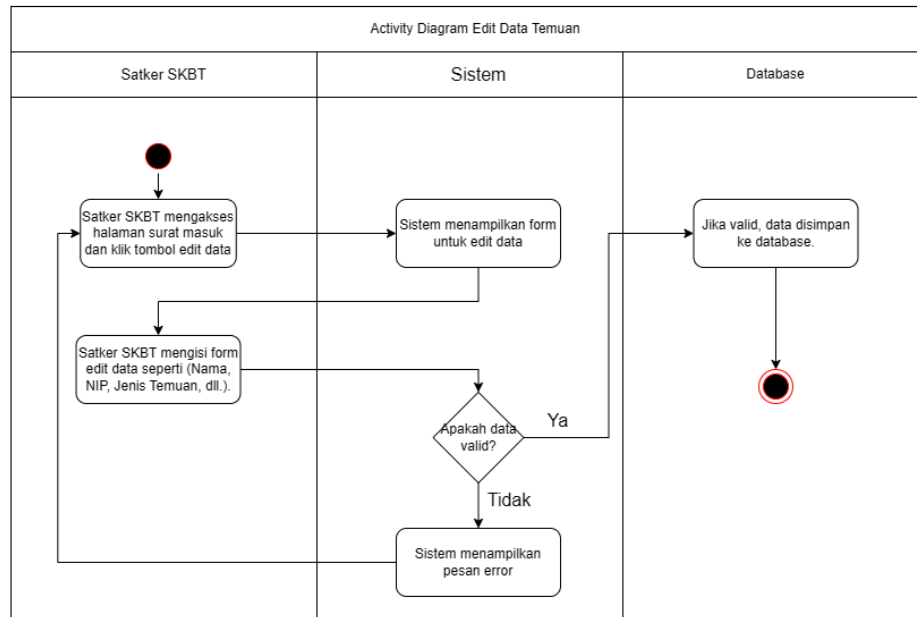
Gambar 4.4 Activity Diagram Mengelola Data Temuan

Berikut tahapan pengguna dalam mengelola data temuan pada aplikasi SKBT. Proses dimulai ketika Satker SKBT membuka halaman "Surat Masuk". Setelah itu, sistem menampilkan antarmuka yang terdiri dari form untuk input data, fitur pencarian data, dan tabel yang berisi daftar data temuan. Diagram ini menyoroti langkah-langkah utama dalam interaksi pengguna dengan sistem untuk mengelola data secara efisien, termasuk memasukkan data baru, mencari data spesifik, atau melihat data yang telah ada.



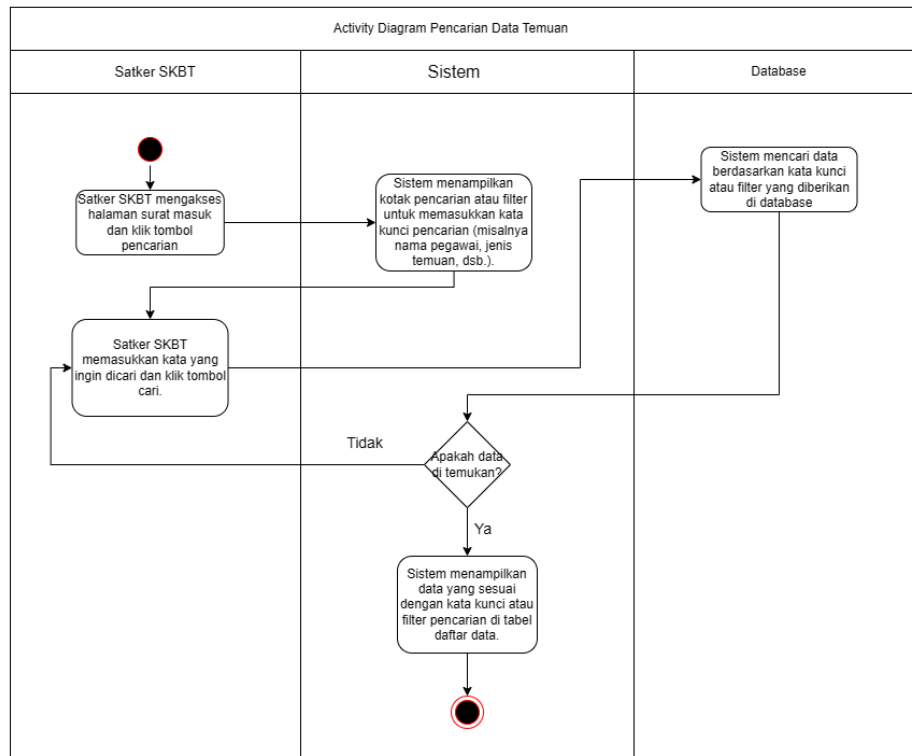
Gambar 4.5 Activity Diagram Tambah Data Temuan

Berikut tahap penambahan data temuan pada aplikasi SKBT. Proses dimulai ketika Satker SKBT membuka halaman surat masuk dan klik tambah data. Kemudian menampilkan form untuk memasukkan data temuan. Pengguna mengisi form tersebut dengan informasi data temuan seperti nama, NIP, jenis temuan, dan detail lainnya. Setelah form diisi, sistem akan memvalidasi data yang dimasukkan. Jika sesuai maka data tersimpan ke database. Namun, jika data tidak valid, menampilkan pesan gagal. Proses berakhir setelah data berhasil disimpan atau pengguna memperbaiki input yang salah.



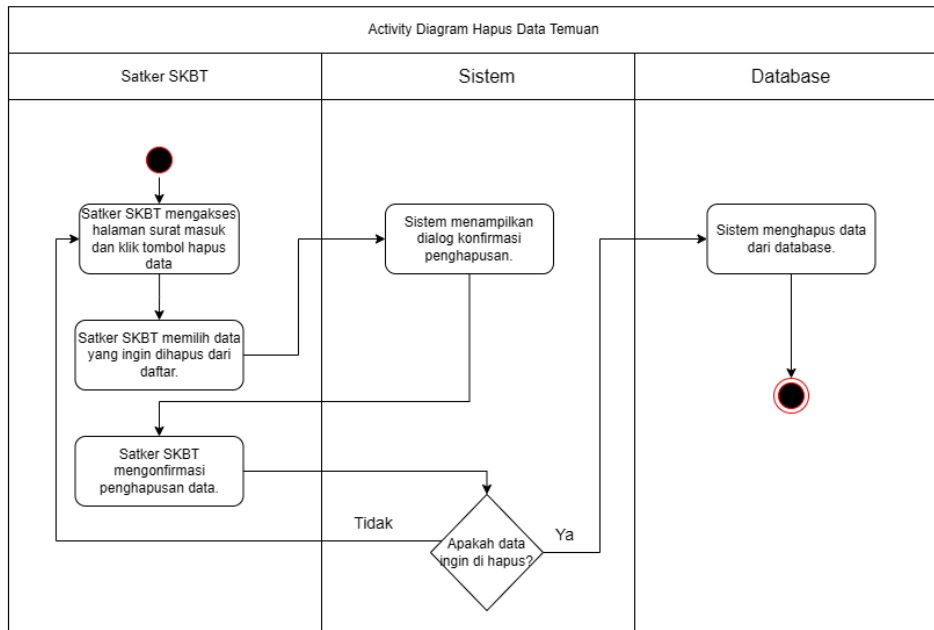
Gambar 4.6 Activity Diagram Edit Data Temuan

Berikut tahap pengeditan data temuan pada aplikasi SKBT. Proses dimulai ketika Satker SKBT mengakses halaman surat masuk lalu klik ubah pada data. Sistem kemudian menampilkan form yang berisi data temuan yang akan diedit. Pengguna mengisi form tersebut dengan informasi yang diperbarui, seperti nama, NIP, jenis temuan, dan lainnya. Setelah form diperbarui, sistem memvalidasi data yang diinput. Jika data valid, sistem akan menyimpan perubahan tersebut ke dalam database. Namun, jika data tidak valid (misalnya ada format yang salah atau field kosong), sistem akan menampilkan pesan error kepada pengguna. Proses ini memastikan bahwa hanya data valid yang dapat disimpan, dan pengguna memiliki kesempatan untuk memperbaiki input jika terjadi kesalahan. Proses berakhir setelah data berhasil diperbarui atau pengguna memperbaiki data yang tidak valid.



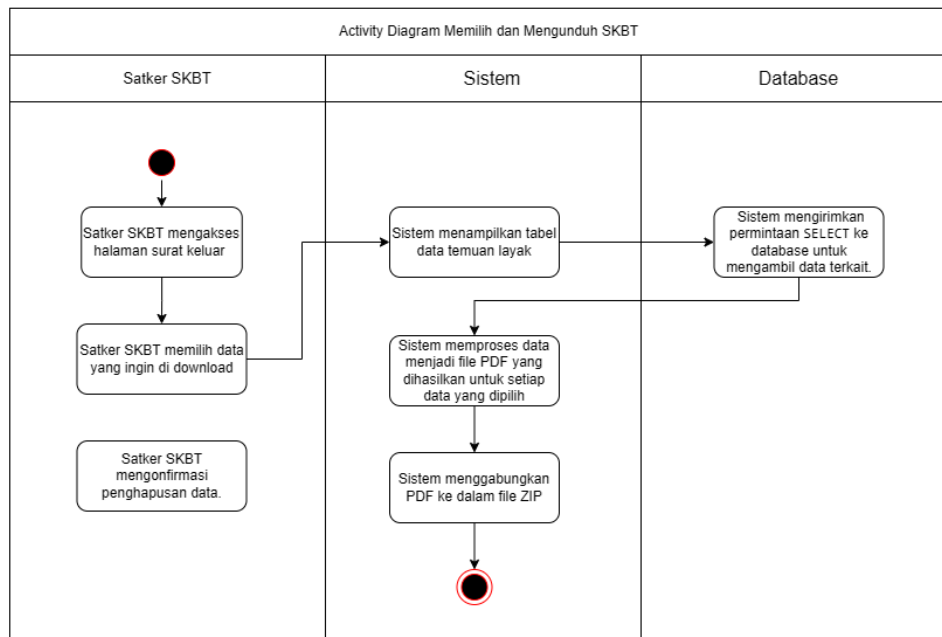
Gambar 4.7 Activity Diagram Pencarian Data Temuan

Activity Diagram di atas menggambarkan proses pencarian data temuan pada aplikasi SKBT. Proses dimulai ketika Satker SKBT mengakses halaman "Surat Masuk" dan memilih tombol pencarian. Sistem kemudian menampilkan kotak pencarian atau filter untuk memasukkan kata kunci, seperti nama pegawai atau jenis temuan. Satker SKBT memasukkan kata kunci yang ingin dicari dan mengklik tombol cari. Sistem mengirimkan permintaan ke database untuk mencari data yang sesuai berdasarkan kata kunci atau filter yang diberikan. Jika data ditemukan, sistem akan menampilkan hasil pencarian di tabel daftar data. Namun, jika data tidak ditemukan, sistem akan memberikan notifikasi bahwa data yang dicari tidak tersedia. Proses ini memastikan pengguna dapat dengan mudah menemukan data spesifik berdasarkan filter yang diinginkan.



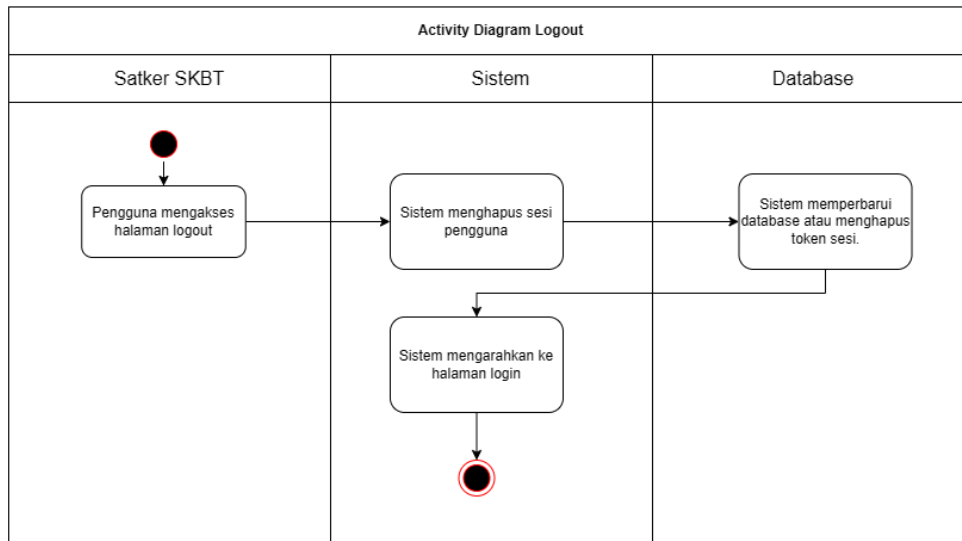
Gambar 4.8 Activity Diagram Hapus Data Temuan

Activity Diagram ini menjelaskan alur proses penghapusan data temuan pada aplikasi SKBT. Proses dimulai ketika Satker SKBT mengakses halaman "Surat Masuk" dan memilih tombol hapus data. Sistem kemudian menampilkan daftar data temuan, di mana pengguna memilih data yang ingin dihapus. Setelah itu, sistem menampilkan dialog konfirmasi untuk memastikan bahwa data yang dipilih benar-benar ingin dihapus. Jika pengguna membatalkan proses (memilih "Tidak"), maka sistem akan membatalkan penghapusan dan kembali ke daftar data. Namun, jika pengguna mengonfirmasi (memilih "Ya"), sistem akan mengirimkan permintaan ke database untuk menghapus data yang dipilih. Proses ini memastikan bahwa penghapusan data dilakukan secara aman dan hanya atas persetujuan pengguna.



Gambar 4.9 Activity Diagram Memilih dan Mengunduh SKBT

Activity Diagram ini menjelaskan proses memilih dan mengunduh Surat Keterangan Bebas Temuan (SKBT) dalam aplikasi SKBT. Proses dimulai ketika Satker SKBT mengakses halaman "Surat Keluar", di mana sistem menampilkan tabel yang berisi data temuan layak untuk diunduh. Sistem mengirimkan permintaan ke database untuk mengambil data terkait dan menampilkannya kepada pengguna. Pengguna kemudian memilih data yang ingin diunduh dan mengonfirmasi permintaan tersebut. Setelah konfirmasi, sistem memproses setiap data yang dipilih menjadi file PDF. Semua file PDF yang dihasilkan kemudian digabungkan ke dalam satu file ZIP untuk mempermudah pengunduhan. Proses ini diakhiri dengan pengiriman file ZIP yang berisi SKBT yang dipilih kepada pengguna.



Gambar 4.10 Activity Diagram Logout

Activity Diagram Logout ini menjelaskan proses ketika pengguna melakukan logout dari sistem. Proses dimulai saat Satker SKBT mengakses halaman logout. Sistem kemudian menghapus sesi pengguna untuk memastikan bahwa pengguna tidak lagi memiliki akses ke sistem. Setelah itu, sistem memperbarui database untuk menghapus token sesi atau informasi lainnya guna menjaga keamanan. Sebagai langkah akhir, sistem mengarahkan pengguna kembali ke halaman login, menandakan bahwa sesi telah berakhir dan pengguna harus login kembali jika ingin mengakses sistem.

4.3.4 Perancangan Basis Data

Perancangan ini meliputi tahapan dalam menyusun *database* sebagai tempat penyimpanan data sementara dalam sistem. Basis data ini dirancang untuk mengelola informasi yang terkait dengan pegawai, temuan administrasi, rekomendasi, dan pembuatan Surat Keterangan Bebas Temuan (SKBT). Basis data terdiri dari empat tabel utama yang memiliki peran spesifik dalam mendukung fungsionalitas aplikasi. Berikut adalah rancangan *database* untuk aplikasi Surat Keterangan Bebas Temuan.

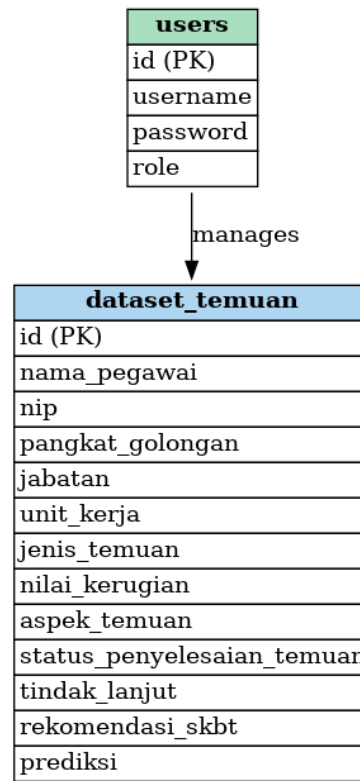
Tabel 4.8 Perancangan Basis Data Dataset Temuan

No.	Kolom	Tipe Data	Keterangan
1	id	int	Primary key
2	nama_pegawai	varchar(255)	Nama lengkap pegawai
3	nip	varchar(20)	Nomor Induk Pegawai
4	pangkat_golongan	varchar(16)	Pangkat dan golongan pegawai
5	jabatan	varchar(220)	Jabatan pegawai
6	unit_kerja	varchar(62)	Unit kerja pegawai
7	jenis_temuan	varchar(38)	Jenis temuan terkait
8	nilai_kerugian	varchar(19)	Nilai kerugian yang ditemukan
9	aspek_temuan	varchar(25)	Aspek temuan yang diamati
10	status_penyelesaian_temuan	varchar(26)	Status penyelesaian temuan
11	tindak_lanjut	varchar(33)	Tindak lanjut yang diberikan
12	rekomendasi_skbt	varchar(16)	Rekomendasi SKBT
13	prediksi	varchar(20)	Hasil prediksi sistem

Tabel 4.9 Perancangan Basis Data Users

No.	Kolom	Tipe Data	Keterangan
1	id	int	Primary key
2	username	varchar(50)	Nama pengguna untuk login
3	password	varchar(255)	Password pengguna yang terenkripsi
4	role	enum('admin_satker', 'sekretaris')	Peran pengguna dalam sistem

4.3.5 Perancangan ERD

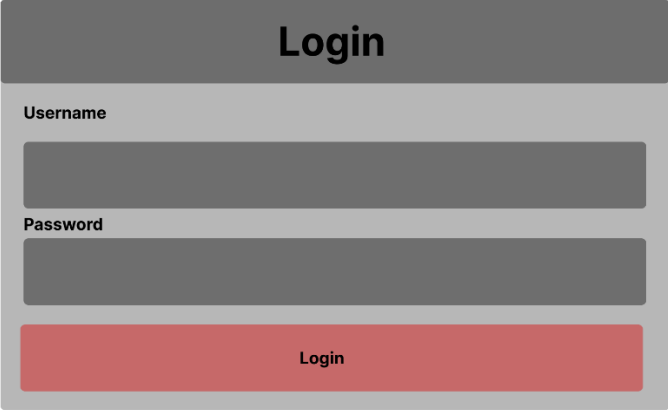


Gambar 4.11 Perancangan ERD

ERD (*Entity Relationship Diagram*) di atas merepresentasikan hubungan antara dua entitas utama, yaitu *users* dan *dataset_temuan*, yang digunakan dalam aplikasi. Entitas *users* berisi informasi autentikasi pengguna, seperti *id* sebagai *Primary Key*, *username*, *password*, dan *role*, yang menunjukkan peran pengguna (misalnya, *admin*). Entitas *dataset_temuan* mencatat data temuan administratif, seperti *id* sebagai *Primary Key*, *nama_pegawai*, *nip*, *pangkat_golongan*, *jabatan*, *unit_kerja*, dan kolom lain yang terkait dengan jenis temuan, aspek, nilai kerugian, serta rekomendasi dan prediksi hasil pengolahan data. Hubungan "*manages*" menunjukkan bahwa entitas *users* mengelola data pada entitas *dataset_temuan*, memungkinkan pengguna untuk menambah, mengedit, atau menghapus data berdasarkan peran yang dimiliki. Diagram ini membantu menggambarkan struktur database secara terorganisir untuk mendukung proses aplikasi.

4.3.6 Perancangan Antarmuka

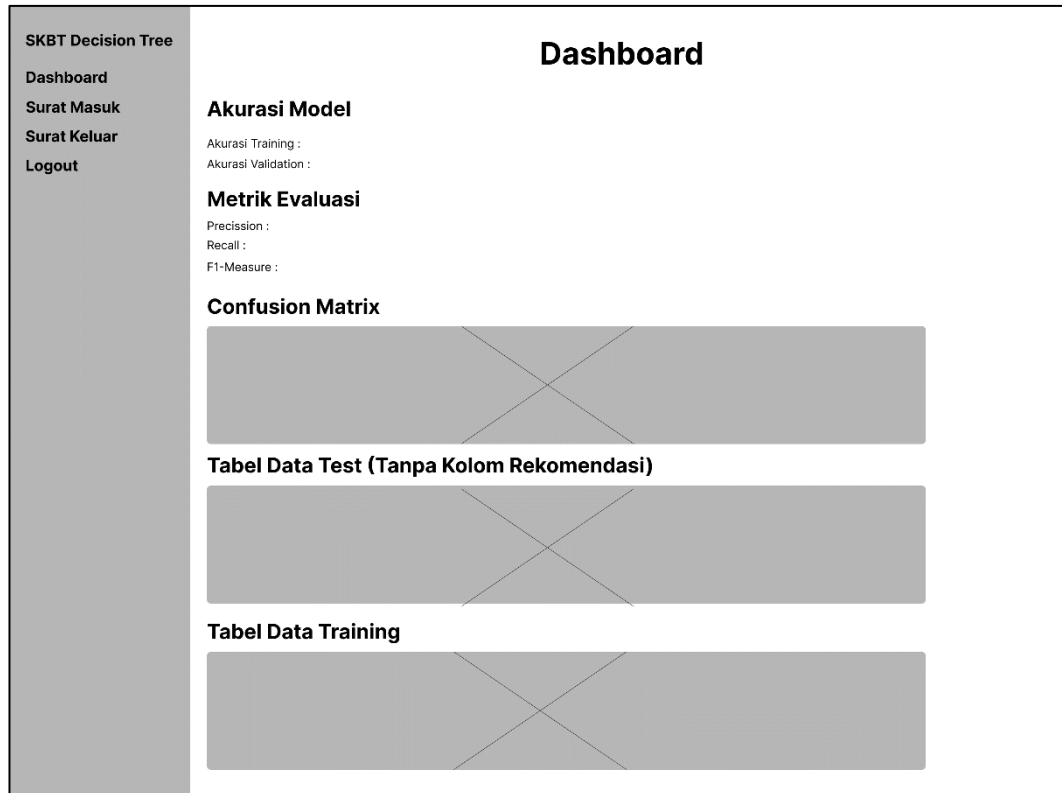
Perancangan antarmuka adalah proses untuk merancang tampilan dari fitur fitur dan tampilan dari aplikasi yang bertujuan untuk adanya interaksi antara pengguna dengan sistem. Tujuan dari perancangan antarmuka tidak hanya untuk menciptakan tampilan yang bagus dan interaktif, tetapi juga agar proses desain aplikasinya lebih sederhana sesuai dengan yang diinginkan. Berikut adalah perancangan antarmuka untuk aplikasi Surat Keterangan Bebas Temuan.



The image shows a mockup of a login page. It has a dark grey header with the word 'Login' in white. Below the header, there are two input fields: 'Username' and 'Password', each with a dark grey input box. At the bottom, there is a red button labeled 'Login'.

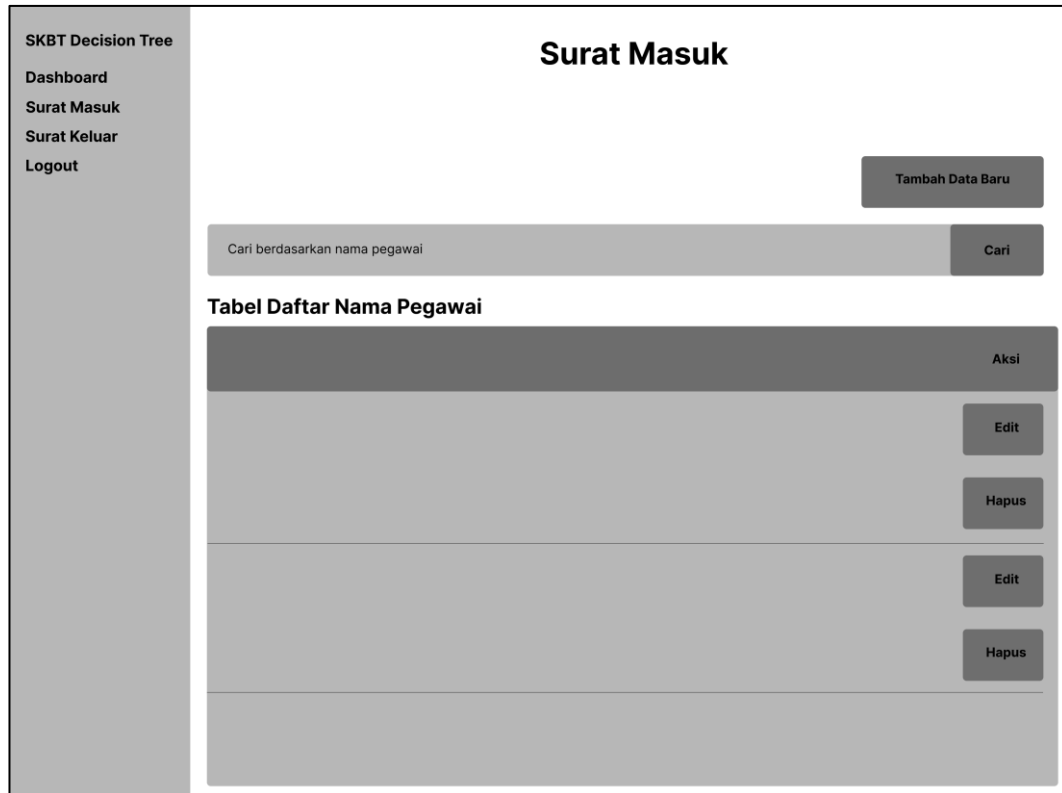
Gambar 4.12 Perancangan Mockup Halaman Login

Mockup login tersebut merupakan bagian penting dari aplikasi Surat Keterangan Bebas Temuan (SKBT) yang dirancang untuk memberikan akses aman kepada pengguna, seperti Satker SKBT, untuk masuk ke dalam sistem. Tampilan login ini menampilkan elemen sederhana namun fungsional, yaitu kolom Username dan Password yang harus diisi oleh pengguna untuk mengautentikasi identitas mereka. Setelah data diisi, pengguna dapat menekan tombol Login untuk memulai sesi kerja di aplikasi. Login ini memastikan bahwa hanya pengguna yang berwenang, seperti admin atau sekretaris, yang dapat mengakses fitur-fitur aplikasi, termasuk pengelolaan data temuan, evaluasi model, dan pengunduhan dokumen SKBT. Desain sederhana pada mockup ini mencerminkan fokus aplikasi terhadap kemudahan penggunaan dan keamanan data.



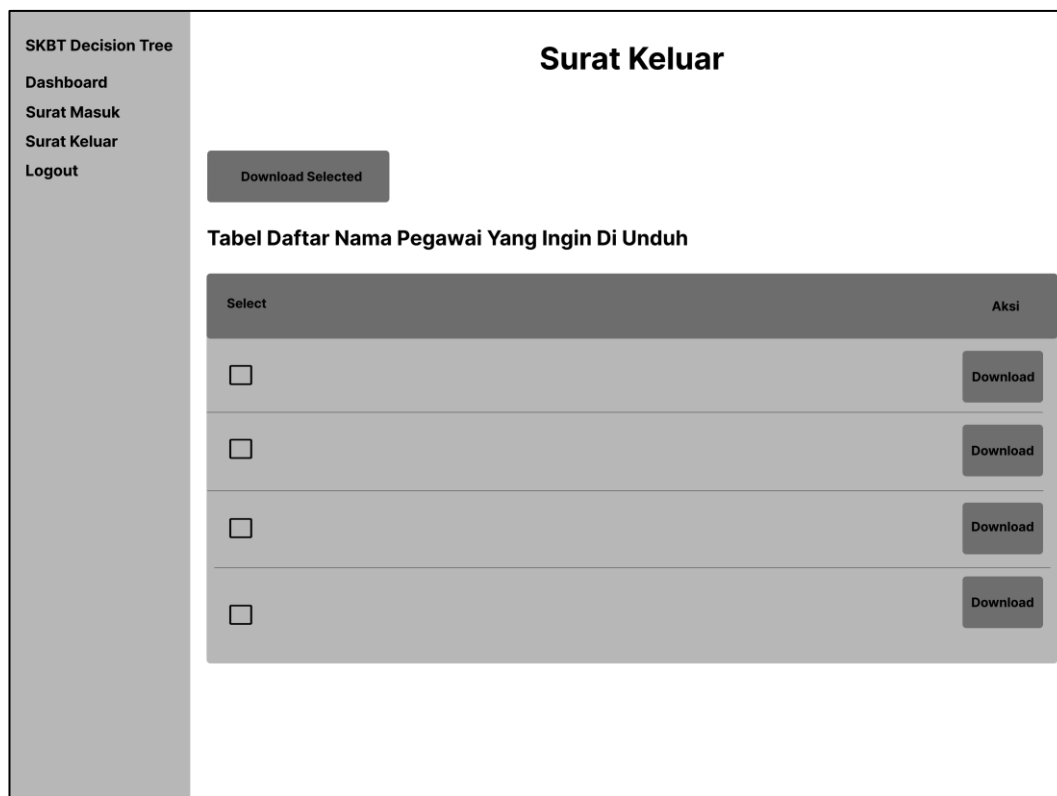
Gambar 4.13 Perancangan Mockup Halaman Dashboard

Mockup dashboard pada aplikasi Surat Keterangan Bebas Temuan (SKBT) ini dirancang untuk menampilkan evaluasi performa model Decision Tree yang digunakan dalam sistem, mencakup akurasi model, metrik evaluasi (*precision*, *recall*, dan *F1-measure*), serta *confusion matrix* untuk visualisasi hasil prediksi. Selain itu, terdapat tabel data test (tanpa kolom rekomendasi) dan tabel data training untuk transparansi proses pengelolaan data dan pelatihan model. Dengan navigasi di sisi kiri, pengguna dapat dengan mudah mengakses fitur utama seperti dashboard, surat masuk, surat keluar, dan logout, menjadikan antarmuka ini informatif dan mudah digunakan.



Gambar 4.14 Perancangan Mockup Halaman Surat Masuk

Mockup halaman Surat Masuk pada aplikasi SKBT ini dirancang untuk mengelola data temuan pegawai secara efisien. Di bagian atas, terdapat fitur pencarian yang memungkinkan pengguna mencari data pegawai berdasarkan nama. Tombol "Tambah Data Baru" mempermudah pengguna untuk menambahkan data baru ke dalam sistem. Bagian tabel menampilkan daftar nama pegawai yang dapat diedit atau dihapus melalui tombol aksi "Edit" dan "Hapus" di sebelah kanan. Navigasi di sisi kiri menyediakan akses cepat ke fitur lain, seperti dashboard, surat keluar, dan logout. Antarmuka ini dirancang untuk mempermudah pengelolaan data temuan secara efektif dan user-friendly.



Gambar 4. 15 Perancangan Mockup Halaman Surat Keluar

Mockup halaman Surat Keluar pada aplikasi SKBT ini dirancang untuk mempermudah pengguna dalam mengunduh dokumen Surat Keterangan Bebas Temuan (SKBT). Pada bagian atas terdapat tombol "*Download Selected*" yang memungkinkan pengguna mengunduh beberapa dokumen sekaligus dalam format ZIP berdasarkan pilihan mereka. Tabel di bawahnya menampilkan daftar pegawai yang telah mendapatkan rekomendasi SKBT, lengkap dengan kolom pilihan (*checkbox*) untuk menandai data yang ingin diunduh dan tombol aksi "*Download*" di sebelah kanan untuk mengunduh dokumen individu. Navigasi di sisi kiri memudahkan akses ke fitur lain, seperti dashboard, surat masuk, dan logout. Desain ini fokus pada kemudahan dan efisiensi proses pengunduhan dokumen.

4.3.7 Perancangan Pengujian

Tahapan ini bertujuan untuk merancang rencana dalam menguji sebuah sistem secara keseluruhan. Dalam penelitian ini melibatkan penggunaan metode *black box* dan *white box*.

4.2.7.1 Perancangan *Black Box*

Black box testing berfokus pada pengujian aplikasi berdasarkan spesifikasi fungsional tanpa melihat implementasi internal kode. Dalam konteks ini, perancangan *black box* digunakan untuk memastikan apakah aplikasi menghasilkan output yang sesuai dengan input yang diberikan, dengan memvalidasi hasil terhadap ekspektasi spesifikasi. Dengan pendekatan ini, pengujian dapat mengidentifikasi masalah pada fitur atau validasi data dari sudut pandang pengguna.



Tabel 4.10 Perancangan Black Box

No.	Pengujian	Proses	Hasil Yang Diharapkan
1	Login	Masukkan username dan password yang valid, lalu klik tombol Login.	Sistem berhasil masuk dan diarahkan ke halaman dashboard.
2	Tambah data	Isi form tambah data dengan informasi lengkap dan valid, lalu klik tombol Submit.	Data berhasil disimpan ke database, halaman diarahkan ke daftar data.
3	Tambah data dengan input salah	Isi form tambah data dengan informasi tidak valid, lalu klik tombol Submit.	Sistem menampilkan pesan error terkait validasi input.
4	Edit data	Pilih data yang akan diubah dari daftar, ubah salah satu informasi valid, lalu klik tombol Update.	Data berhasil diperbarui di database.
5	Hapus data	Pilih data dari daftar, klik tombol Hapus, lalu konfirmasi penghapusan.	Data berhasil dihapus dari database.
6	Pencarian data	Masukkan kata kunci di kolom pencarian dan klik tombol Cari.	Tabel daftar data menampilkan hasil pencarian sesuai kata kunci yang dimasukkan.
7	Mengunduh file PDF	Pilih data dari daftar Surat Keluar, lalu klik tombol Download atau Download Selected untuk unduhan massal.	File PDF berhasil diunduh dan file ZIP dihasilkan untuk unduhan massal.
8	Logout	Klik tombol Logout.	Sistem menghapus sesi dan mengarahkan pengguna kembali ke halaman login.
9	Dashboard evaluasi model	Akses halaman dashboard untuk melihat metrik evaluasi model (akurasi, precision, recall, f1-score).	Dashboard menampilkan metrik evaluasi model dan data training/test.

4.3.7.2 Perancangan *White Box*

White box testing adalah metode yang digunakan untuk memverifikasi bahwa setiap komponen dalam aplikasi telah dirancang dengan benar, diuji secara menyeluruh, dan berfungsi sesuai dengan skenario yang direncanakan. *white box testing* dilakukan dengan melihat ke dalam struktur internal kode untuk memverifikasi logika program, jalur kontrol, alur data, serta proses yang terlibat. Tujuan utama dari pengujian ini adalah untuk mendeteksi dan memperbaiki kesalahan atau potensi kegagalan yang dapat terjadi sebelum aplikasi dijalankan oleh pengguna.

Tabel 4.11 Perancangan White Box

No	Komponen yang Diuji	Nama Kode Program	Hasil yang Diharapkan
1	Fungsi Login	@app.route('/login')	Sistem memverifikasi username dan password yang valid, menampilkan dashboard pengguna jika login berhasil, atau pesan kesalahan jika gagal.
2	Fungsi Logout	@app.route('/logout')	Sistem menghapus sesi pengguna dan mengarahkan ke halaman login.
3	Fungsi Tambah Data	@app.route('/create')	Sistem berhasil menambahkan data baru ke database dan menampilkan data terbaru di halaman Surat Masuk.
4	Fungsi Edit Data	@app.route('/update/<int:id>')	Sistem berhasil mengupdate data yang telah diubah pengguna dan menyimpan pembaruan ke database.
5	Fungsi Hapus Data	@app.route('/delete/<int:id>')	Sistem berhasil menghapus data yang dipilih dari database dan memperbarui tabel Surat Masuk.
6	Fungsi Pencarian Data	@app.route('/all_data')	Sistem menampilkan data sesuai dengan kata

			kunci pencarian dari pengguna.
7	Fungsi Prediksi Rekomendasi	predict_rekomendasi	Sistem berhasil memprediksi hasil rekomendasi menggunakan Decision Tree dan menyimpan hasil ke database.
8	Fungsi Menampilkan Data di Dashboard	@app.route('/')	Sistem menampilkan metrik evaluasi (akurasi, precision, recall, F1-score), confusion matrix, serta tabel data training dan test.
9	Fungsi Download Surat Keluar	@app.route('/download/<int:id>')	Sistem menghasilkan dan mengunduh file PDF untuk data pegawai yang dipilih.
10	Fungsi Download Multiple Surat Keluar	@app.route('/download_selected')	Sistem menghasilkan file ZIP berisi PDF untuk beberapa data pegawai yang dipilih.