Jurnal Ilmu Iformasi Perpustakaan dan Kearsipan
Volume 15 No. 01 Tahun 2026 *ISSN:2302-3511*
https://iipk.ppj.unp.ac.id/index.php/jiipk/index

Perancangan Sistem Informasi Katalog Arsip dan Naskah TerIntegrasi (SIKANTI) Menggunakan Pendekatan *Prototyping* pada Sekretariat BPK Perwakilan Provinsi Jambi

Muhammad Aji Triatama[1]*; M. Farhan Ramadhan[1]; M. Reno Novriansyah[1]; Rayhan Syawal[1]; Noviana Safitri[1]; Nanda Septa Lian Sari[1];

[1]Program Studi Sistem Informasi, Universitas Islam Negeri Sulthan Thaha Saifuddin Jambi
*Corresponding author. Email: m.ajitriatama@uinjambi.ac.id

ABSTRACT

Archive management at the Secretariat of the BPK Representative Office of Jambi Province faces risks of physical damage, spatial inefficiency, and slow cross-subdepartment information retrieval. This research designs the SIKANTI electronic archiving system prototype to secure document distribution through database isolation across five working subdepartments. The system is fortified using a Role-Based Access Control (RBAC) architecture with a hardened deployment method, permanently disabling the public registration route to prevent illegal admin account creation. Staff-level account management is handled via closed-loop provisioning with interface restrictions (UI isolation) centered on the Folder Repository. As an additional control, SIKANTI implements a QR code-based real-time verification gateway for external document borrowing with an instant access revocation feature. This implementation has proven highly effective in drastically cutting search times while realizing absolute information security governance.

***Keywords** : Personnel Archives, Hardened Deployment, Role-Based Access Control, QR Code, Information Retrieval.*

ABSTRAK

Pengelolaan arsip pada Sekretariat BPK Perwakilan Provinsi Jambi dihadapkan pada risiko kerusakan fisik, inefisiensi ruang, dan lambatnya temu balik informasi lintas subbagian. Penelitian ini merancang purwarupa sistem kearsipan elektronik SIKANTI untuk mengamankan distribusi dokumen melalui isolasi basis data (database isolation) pada lima subbagian kerja. Sistem ini diperketat menggunakan arsitektur Role-Based Access Control (RBAC) dengan metode hardened deployment, yakni menonaktifkan rute registrasi publik secara permanen demi mencegah pembuatan akun admin ilegal. Manajemen akun tingkat Staf dikelola melalui closed-loop provisioning dengan restriksi antarmuka (UI isolation) yang terpusat pada Gudang Folder. Sebagai kontrol tambahan, SIKANTI mengimplementasikan gerbang verifikasi real-time berbasis kode QR untuk peminjaman dokumen eksternal dengan fitur pencabutan akses instan. Implementasi ini terbukti ampuh memangkas waktu pencarian secara drastis sekaligus mewujudkan tata kelola keamanan informasi yang absolut.

Kata Kunci : Arsip Kepegawaian, *Hardened Deployment, Role-Based Access Control,* Kode QR, Temu Balik Informasi.

Pendahuluan

Pengelolaan arsip di instansi pemerintah diwajibkan bertransformasi menuju tata kelola digital sesuai dengan amanat UU No. 43 Tahun 2009 tentang Kearsipan dan Perpres No. 95 Tahun 2018 tentang Sistem Pemerintahan Berbasis Elektronik (SPBE) demi mewujudkan pelayanan publik yang efektif dan akuntabel (Iqbal, Daraba, & Indrayani, 2024; Pemerintah Republik Indonesia, 2009). Namun, pada BPK Perwakilan Provinsi Jambi yang membawahi lima unit kerja (Subbagian SDM, Humas, Umum & IT, Keuangan, dan Hukum), pengelolaan berkas digital masih bersifat parsial di komputer masing-masing administrator. Kondisi desentralisasi ini memicu risiko tinggi terhadap kehilangan dokumen akibat bencana, penumpukan kapasitas fisik, serta rendahnya rasio penemuan kembali (*finding ratio*) dokumen saat dibutuhkan (Ebizeen & Cahyono, 2025; Muhidin & Winata, 2016). Transisi alih media sering kali terhambat oleh resistensi budaya kerja manual staf pelaksana di lapangan (Sutirman, Wijayanti, & Purwanto, 2016; Yogopriyatno, Nursanty, & Roeliana, 2024). Oleh karena itu, ketersediaan repositori digital terpusat yang aman menjadi kebutuhan yang sangat mendesak.

Sebagai langkah mitigasi dibangunlah SIKANTI (Sistem Informasi Katalog Arsip dan Naskah TerIntegrasi), yang mengadopsi kearifan lokal Jambi bermakna "sahabat" untuk menghadirkan antarmuka ramah pengguna (*user-friendly*) dan menekan resistensi adaptasi. SIKANTI dirancang sebagai platform terpadu yang mampu mengakomodasi kebutuhan kelima subbagian secara serentak dalam satu ekosistem, dengan jaminan kemandirian data melalui mekanisme isolasi basis data (*database isolation*) tingkat subbagian. Tata kelola keamanan diperketat melalui arsitektur *Role-Based Access Control* (RBAC) dan metode *hardened deployment*, di mana rute registrasi publik (seperti /register) dinonaktifkan secara permanen pada peladen produksi untuk menutup celah akun administratif ilegal. Akun tingkat Staf didistribusikan melalui skenario *closed-loop provisioning* oleh Admin dan dikenakan restriksi antarmuka (*UI Isolation*) guna menyembunyikan menu manajerial tingkat atas, namun tetap diberikan wewenang operasional penuh (*full access*) dalam mengelola isi modul Gudang Folder.

Kelebihan utama SIKANTI terletak pada pemutakhiran sistem temu balik informasi (*information retrieval*) memanfaatkan pengindeksan metadata ganda dan algoritma relasional yang memangkas waktu pencarian dari hitungan jam menjadi hitungan detik (Fathansyah, 2018). Pohon hierarki direktori digital (*folder tree*) disusun persis mereplikasi nomenklatur fisik BPK (seperti KP.00 hingga KP.02) untuk mencegah terjadinya gegar budaya (*culture shock*) bagi administrator. Untuk sirkulasi dokumen antarunit, SIKANTI mengimplementasikan gerbang verifikasi *real-time* berbasis kode QR dengan kendali mutlak (*absolute governance*) di tangan Admin untuk menentukan hak tinjau (*view-only*), hak unduh (*download*), maupun pencabutan akses instan. Dikembangkan dengan pendekatan *prototyping* (Pressman & Maxim, 2020), SIKANTI secara dinamis mengintegrasikan pengelolaan siklus hidup dokumen berdasarkan pedoman Jadwal Retensi Arsip (JRA) BPK guna mewujudkan ekosistem informasi yang cepat, akurat, dan aman.

Metode

Metode penelitian yang digunakan dalam perancangan sistem ini mencakup prosedur pengumpulan data lapangan dan penerapan model pengembangan perangkat lunak secara iteratif. Untuk memperoleh data yang akurat dan relevan dengan kebutuhan organisasi, digunakan beberapa teknik pengumpulan data (Sugiyono, 2019) sebagai berikut:

1. Wawancara (*Interview*): Tanya jawab dilakukan secara mendalam menggunakan teknik *purposive sampling*, di mana narasumber difokuskan pada perwakilan Subbagian SDM dan Subbagian Umum & IT BPK Perwakilan Provinsi Jambi. Pemilihan kedua unit ini didasarkan pada pertimbangan strategis: Subbagian SDM mewakili unit dengan kompleksitas dan volume arsip kepegawaian tertinggi (*extreme case*), sementara Subbagian Umum & IT mewakili otoritas pengelola infrastruktur dan regulasi kearsipan terpusat. Pengumpulan data dari kedua representasi utama ini dinilai komprehensif untuk memetakan alur peminjaman dokumen, regulasi retensi arsip, serta mengidentifikasi kendala operasional lintas unit secara institusional.
2. Observasi: Melakukan pengamatan langsung terhadap prosedur pengelolaan berkas di ruang arsip, kondisi fisik rak penyimpanan, serta mengukur durasi waktu yang dibutuhkan staf administrator dalam melakukan pencarian dokumen secara manual.
3. Studi Pustaka: Mengumpulkan landasan teori melalui literatur kearsipan, dokumen Undang-Undang Nomor 43 Tahun 2009, regulasi mengenai Sistem Pemerintahan Berbasis Elektronik (SPBE), serta penelitian terdahulu yang relevan dengan pengembangan sistem pengarsipan

elektronik. Hal ini dilakukan untuk memetakan kesenjangan antara kebijakan digitalisasi nasional dengan kendala psikologis dan budaya kerja manual yang sering ditemukan di lingkungan pemerintahan (Iqbal et al., 2024).

Selanjutnya, untuk merancang dan membangun perangkat lunak, penelitian ini menerapkan metode *prototyping*. Pemilihan metode ini didasarkan pada efektivitasnya dalam memfasilitasi komunikasi dua arah antara pengembang dan pengguna, di mana evaluasi sistem dapat dilakukan secara iteratif (berulang) guna menjamin keselarasan antara rancangan sistem dengan kebutuhan operasional yang dinamis di lapangan (Ebizeen & Cahyono, 2025; Giawa, Raudhah, Abdy, & Satria Alasi, 2026). Menurut Pressman & Maxim (2020), tahapan pengembangan *prototyping* yang dilalui meliputi:

1. Pengumpulan Kebutuhan (*Communication*): Tahap awal untuk mendefinisikan fitur utama sistem. Berdasarkan hasil wawancara dari unit representatif, ditarik sebuah kesimpulan rancangan (*design abstraction*) bahwa sistem harus mampu melayani seluruh unit kerja. Oleh karena itu, diidentifikasi kebutuhan fungsional lintas unit, seperti pengunggahan massal (*batch upload*), sistem klasifikasi dokumen berdasarkan nomenklatur spesifik masing-masing subbagian, serta mekanisme gerbang verifikasi untuk akses data antar unit.
2. Perancangan Cepat (*Quick Design*): Menyusun rancangan antarmuka (*User Interface*) dan pemodelan proses menggunakan diagram teknis seperti *Use Case Diagram*, *Activity Diagram* (yang mencakup pemodelan interaksi dua *swimlane* untuk alur verifikasi serta pemodelan percabangan paralel untuk alur administrator), serta perancangan basis data melalui *Entity Relationship Diagram* (ERD) yang mengakomodasi isolasi data per subbagian.
3. Pembangunan Prototipe (*Construction of Prototype*): Merealisasikan rancangan menjadi aplikasi fungsional berbasis *web* menggunakan *framework* Laravel. Prototipe ini mencakup fungsi inti seperti manajemen direktori folder hierarki, algoritma pencarian metadata ganda, penerapan isolasi basis data (*database isolation*), serta pengamanan *Role-Based Access Control* (RBAC) yang krusial untuk menjamin segregasi hak akses dan integritas data antar subbagian (Gunawan, Mulyanto, & Iskandar, 2026).
4. Evaluasi Prototipe (*Deployment, Delivery, & Feedback*): Sistem versi awal diserahkan kepada perwakilan administrator dan staf untuk diuji coba secara komprehensif. Tahapan ini sangat penting mengingat kendala rendahnya literasi digital dan kesiapan SDM sering kali menjadi faktor penghambat utama dalam implementasi sistem informasi di instansi pemerintah (Yogopriyatno et al., 2024). Hasil pengujian menunjukkan adanya masukan terkait penyederhanaan navigasi, seperti perlunya penerapan *UI isolation* di mana pengguna level Staf hanya melihat menu yang relevan, serta perubahan struktur menu yang sebelumnya dianggap terlalu teknis oleh pengguna akhir.
5. Perbaikan Prototipe (*Refinement*): Berdasarkan evaluasi dari pengguna lintas unit, dilakukan modifikasi pada antarmuka dengan mengadopsi model *dashboard* interaktif yang dinamis menyesuaikan sesi *login* subbagian, penambahan fitur *drag-and-drop*, dan optimasi kotak pencarian tunggal. Proses evaluasi dan perbaikan ini dilakukan secara berulang hingga sistem dinyatakan memenuhi standar operasional terintegrasi bagi seluruh subbagian di Sekretariat BPK Perwakilan Provinsi Jambi.

Hasil dan Pembahasan

A. Analisis Kebutuhan Sistem

Tahap analisis kebutuhan dilakukan berdasarkan hasil pengumpulan data melalui wawancara mendalam (*purposive sampling*) dengan perwakilan representatif dari Subbagian SDM serta Subbagian Umum & IT BPK Perwakilan Provinsi Jambi. Selain itu, analisis juga didasarkan pada tinjauan terhadap dokumen regulasi internal, yakni Peraturan Sekretaris Jenderal BPK tentang Jadwal Retensi Arsip (JRA) dan Pedoman Kode Klasifikasi Arsip. Dari tahapan abstraksi desain tersebut, dirumuskan bahwa sistem tidak boleh dibangun secara *silo* (terpisah-pisah), melainkan harus menjadi satu ekosistem terpadu lintas unit. Oleh karena itu, spesifikasi kebutuhan Sistem SIKANTI diklasifikasikan menjadi dua kategori utama yang mengikat secara institusional, yaitu kebutuhan fungsional dan kebutuhan non-fungsional.

Kebutuhan Fungsional (*Functional Requirements*)

Kebutuhan fungsional mendefinisikan layanan, proses, dan fitur utama yang harus disediakan oleh perangkat lunak untuk menjawab permasalahan di lapangan. Spesifikasi kebutuhan fungsional dari aplikasi SIKANTI meliputi:

1. Manajemen Klasifikasi Lintas Nomenklatur: Sistem harus mampu mereplikasi struktur penyimpanan ordner fisik ke dalam direktori digital (*folder tree*) yang dinamis. Sistem wajib mengakomodasi dan mengelompokkan arsip secara otomatis berdasarkan pedoman klasifikasi masing-masing unit kerja (misalnya nomenklatur KP untuk SDM, KU untuk Keuangan, dan HM untuk Humas) untuk mencegah gegar budaya (*culture shock*) bagi administrator di masing-masing subbagian.
2. Keamanan Arsitektur *Role-Based Access Control* (RBAC) & *Database Isolation*: Sistem harus membedakan hak akses secara mutlak. Sistem harus mengimplementasikan *Hardened Deployment* dengan meniadakan rute registrasi publik untuk mencegah pembuatan akun admin ilegal. Ekosistem pengguna dibagi berdasarkan isolasi basis data tingkat subbagian (*database isolation*):
    a. Administrator Subbagian: Didaftarkan secara *pre-registered*, memiliki kendali penuh atas manajemen folder, dokumen, kelola sampah, dan otorisasi akses di unitnya.
    b. Staf (User): Didaftarkan melalui *closed-loop provisioning* oleh Admin terkait. Akun tingkat staf harus dikenakan *UI Isolation* (restriksi antarmuka) yang mengunci pandangan mereka hanya pada menu Dashboard dan Gudang Folder serta membatasi akses pada menu sistem lainnya.
3. Gerbang Verifikasi *Real-Time* Berbasis Kode QR: Untuk memfasilitasi peminjaman dokumen fisik maupun *soft file* secara lintas unit/eksternal, sistem wajib menyediakan fitur otorisasi berbasis Kode QR. Pemindaian kode ini akan menahan akses pengguna luar hingga Administrator Subbagian pemilik arsip memberikan verifikasi. Admin memiliki kendali penuh untuk menyetujui hak tinjau (*view-only*), memberikan hak unduh (*download*), wajib mencantumkan alasan jika menolak, hingga opsi pencabutan akses paksa secara instan.
4. Sistem Temu Balik Informasi (*Information Retrieval*): Sistem harus menyediakan kotak pencarian tunggal (*single search bar*) yang didukung oleh algoritma pencarian relasional berbasis *metadata* ganda. Pengguna harus dapat menemukan dokumen secara presisi dalam hitungan detik.
5. Manajemen Siklus Hidup dan Retensi Arsip (JRA): Mengacu pada Peraturan Sekjen BPK, sistem harus memiliki fitur penyusutan digital otomatis. Sistem mampu melacak umur dokumen dan memberikan indikator visual yang tegas untuk membedakan status retensi arsip: masa aktif, masa inaktif, atau masa habis retensi dengan rekomendasi akhir simpan permanen atau musnah.
6. Registrasi Arsip Presisi (*Structured Document Registration*): Mempertimbangkan sensitivitas dan kerahasiaan data arsip BPK, sistem meniadakan fungsionalitas impor massal dari pihak ketiga (seperti integrasi Excel) guna mencegah risiko galat format (*formatting errors*) dan anomali (*bypass* validasi). Sebagai gantinya, sistem mewajibkan penginputan *metadata* dilakukan secara presisi dan terstruktur (*single-entry registration*) melalui antarmuka aplikasi. Pendekatan ini merupakan protokol wajib untuk memastikan bahwa setiap dokumen yang diregistrasikan telah divalidasi secara absolut oleh Administrator sebelum dikunci ke dalam basis data.

Kebutuhan Non-Fungsional (*Non-Functional Requirements*)

Kebutuhan non-fungsional mendefinisikan batasan, kinerja, dan kualitas dari sistem yang dibangun. Kebutuhan tersebut meliputi:

1. Kegunaan (Usability): Antarmuka pengguna (User Interface) harus dirancang dengan model dashboard interaktif dan menyediakan fitur drag-and-drop untuk pengunggahan dokumen. Desain harus intuitif agar staf SDM dapat mengoperasikan sistem tanpa memerlukan pelatihan teknis yang rumit.
2. Kinerja (Performance): Sistem harus memiliki waktu respons yang cepat, terutama saat melakukan kueri pencarian dokumen dari basis data relasional yang memuat ribuan data kepegawaian.
3. Keamanan (Security): Sistem harus menjamin kerahasiaan data (enkripsi basis data) dan mencegah akses tidak sah dari pihak luar, memastikan bahwa dokumen vital kepegawaian tidak

bocor atau dimanipulasi.

Pada tahap perancangan sistem (*Quick Design*), fokus utama adalah mentransformasikan analisis kebutuhan ke dalam model teknis yang menjadi cetak biru pembangunan aplikasi. Perancangan ini mencakup pemodelan proses menggunakan *Unified Modeling Language* (UML) dan perancangan basis data untuk memastikan integritas data arsip kepegawaian.

B. Perancangan Sistem

Kebutuhan non-fungsional mendefinisikan batasan, standar kinerja, dan kualitas dari sistem arsitektur yang dibangun. Kebutuhan tersebut meliputi:

1. Keamanan Ekstra (*Security Governance*): Selain isolasi basis data, sistem harus menjamin kerahasiaan (*confidentiality*) dan integritas (*integrity*) data tingkat tinggi. *File* digital yang diunggah harus diproteksi sehingga dokumen vital instansi tidak dapat diakses melalui peretasan *URL* langsung (*direct link access*) tanpa melewati lapisan autentikasi aplikasi.
2. Kinerja (*Performance*): Sistem harus mempertahankan waktu respons yang optimal (tanpa *lag* yang signifikan), terutama saat melakukan kueri pemuatan *dashboard* maupun penelusuran dokumen dari basis data relasional yang menampung ribuan rekaman arsip institusional.
3. Kegunaan (*Usability*): Antarmuka pengguna (*User Interface*) harus dirancang dengan pendekatan *SaaS-style minimalism* untuk memastikan kemudahan navigasi. Tampilan halaman Dashboard harus menyajikan informasi secara interaktif melalui representasi visual berupa kartu statistik (volume folder, dokumen, retensi), grafik area (*chart*), dan lini masa (*timeline*) aktivitas secara *real-time*. Selain itu, untuk proses pengunggahan dokumen digital di dalam menu Gudang Folder, sistem wajib menyediakan fungsionalitas *drag-and-drop*. Desain antarmuka dashboard mengintegrasikan representasi visual kartu statistik, grafik area, dan lini masa, serta menyediakan fungsionalitas drag-and-drop pada menu Gudang Folder.

Pada tahap perancangan sistem (*Quick Design*), fokus utama adalah mentransformasikan analisis kebutuhan yang ketat ini ke dalam model teknis sebagai cetak biru pembangunan aplikasi. Perancangan ini mencakup pemodelan proses menggunakan pendekatan *Unified Modeling Language* (UML) untuk memastikan integritas relasi data antarsubbagian.

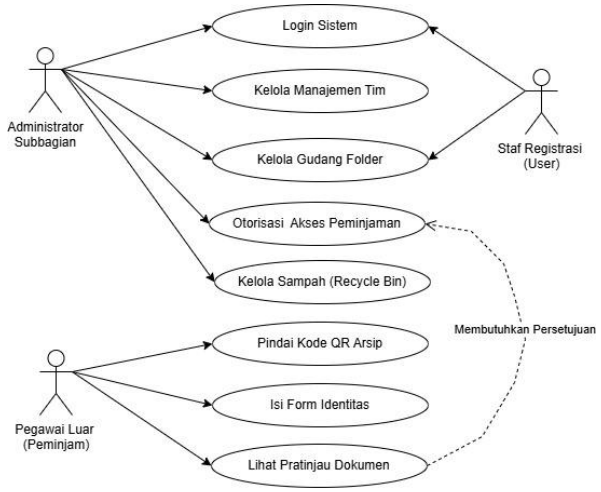

Figure 1. *Use Case Diagram* Sistem SIKANTI

1. Pemodelan Alur Kerja dan Keamanan Sistem *(Activity Diagram)*

Perancangan alur kerja operasional sekaligus mekanisme keamanan pada Sistem SIKANTI direpresentasikan melalui *Activity Diagram*. Menurut Rosa & Shalahuddin (2018), *Activity Diagram* sangat efektif untuk memodelkan alur kerja (*workflow*) dari sebuah urutan aktivitas yang melibatkan berbagai aktor di dalam sistem. Pada aplikasi ini, pemodelan dibagi menjadi tiga representasi visual utama yang saling melengkapi: alur operasional paralel tingkat Administrator Subbagian, alur pendaftaran dan restriksi otorisasi akun Staf, serta alur interaksi verifikasi peminjaman arsip lintas unit.

Sebagai pemegang otoritas penuh di unit kerjanya, Administrator Subbagian (Admin Sektor) memiliki alur kerja manajerial yang dikonsep menggunakan percabangan paralel (*forking*). Setelah melewati tahap autentikasi (*login*), sistem akan mengidentifikasi sesi dan menerapkan isolasi basis data (*database isolation*) sebelum mengarahkan Admin ke *Dashboard* utama. Dari titik sentral ini, Administrator memiliki keleluasaan untuk mengakses empat jalur operasional utama secara paralel: (1) Gudang Folder, tempat Admin memiliki kendali penuh untuk meregistrasi, mengedit, dan memanajemen siklus hidup dokumen di unitnya; (2) Manajemen Tim, di mana Admin menerapkan *closed-loop provisioning* untuk mendaftarkan akun staf pelaksana baru atau mereset kata sandi; (3) Verifikasi Akses, untuk menjadi gerbang otorisasi permintaan dokumen; serta (4) Navigasi Sistem, yang memberi hak eksklusif kepada Admin untuk mengakses Kelola Sampah (pemulihan atau pemusnahan permanen data), memantau log audit di Jejak Aktivitas, dan memodifikasi identitas instansi di Pengaturan Utama.

Sebagai representasi operasional di tingkat pelaksana, pemodelan kedua menggambarkan alur pendaftaran hingga akses otorisasi akun Staf (*User*) yang melibatkan tiga *swimlane* (Admin Sektor, Sistem SIKANTI, dan Staf). Sesuai dengan protokol keamanan *hardened deployment*, alur ini dipicu secara eksklusif oleh Administrator yang mendaftarkan kredensial staf baru di menu Manajemen Tim. Setelah data dikirim, Sistem SIKANTI secara otomatis melakukan enkripsi kata sandi dan mengikat akun tersebut pada ID Subbagian yang sama dengan Admin pembuatnya (*database isolation*). Ketika Staf menggunakan kredensial tersebut untuk melakukan *login*, sistem langsung mengeksekusi protokol *UI Isolation*. Sistem secara proaktif menyembunyikan menu-menu manajerial tingkat atas, sehingga sesi aktif Staf langsung diarahkan ke *Dashboard* dengan hak akses eksekusi yang difokuskan secara eksklusif pada menu operasional Gudang Folder.

Selanjutnya, perancangan keamanan untuk sirkulasi dokumen antar unit dimodelkan melalui interaksi dua *swimlane* yang melibatkan Staf (Pemohon) dan Admin Sektor (*Verifier*). Mekanisme ini mengadopsi gerbang verifikasi (*gatekeeping*) berbasis *QR Code*. Alur dimulai ketika Staf Pemohon melakukan pemindaian *QR Code* pada sebuah dokumen arsip (baik secara digital maupun fisik) dan mengirimkan formulir tujuan akses. Tindakan ini memicu notifikasi *real-time* di layar Admin Sektor pemilik dokumen. Admin bertindak sebagai pemegang kendali mutlak: apabila permintaan ditolak, Admin diwajibkan untuk mengisi catatan alasan penolakan. Sebaliknya, apabila disetujui, sistem memberikan fleksibilitas kepada Admin untuk menentukan limitasi akses apakah Staf tersebut hanya diberikan hak tinjau dokumen (*view-only*) atau diizinkan untuk mengunduh *file* (*download*). Melalui kombinasi protokol verifikasi ketat dan restriksi antarmuka bagi pengguna level Staf ini, SIKANTI menjamin tidak ada satupun dokumen vital yang bocor melintasi batas subbagian tanpa otorisasi digital yang sah.

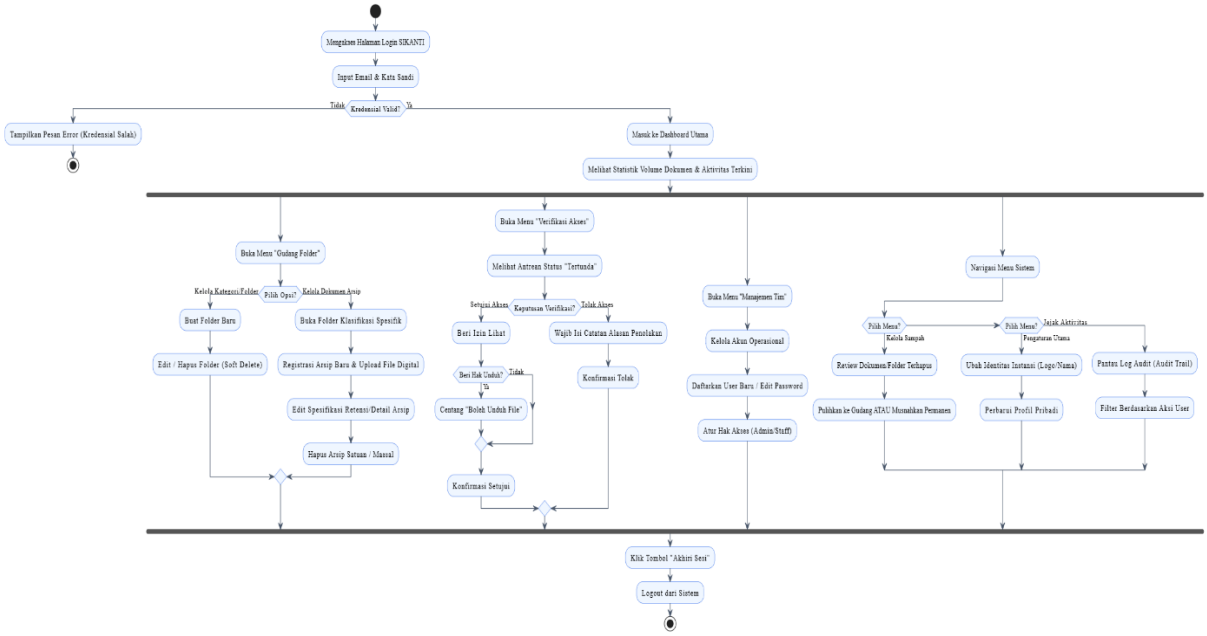

Figure 2. *Activity Diagram* Alur Kerja Sistem SIKANTI

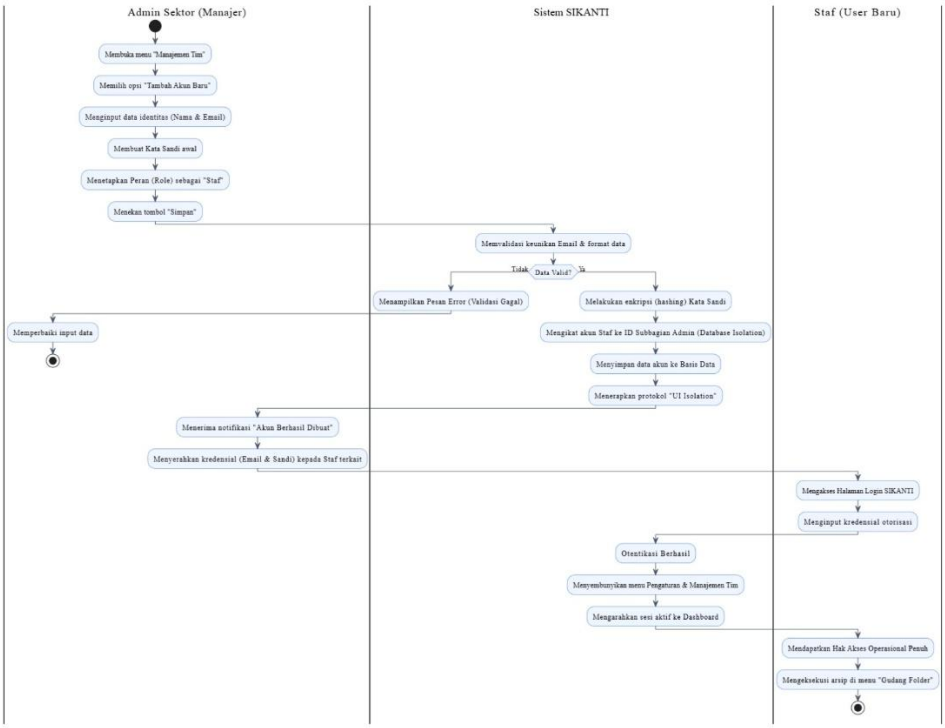

Figure 3. ***Activity Diagram*** **Alur Staf Terdaftar**

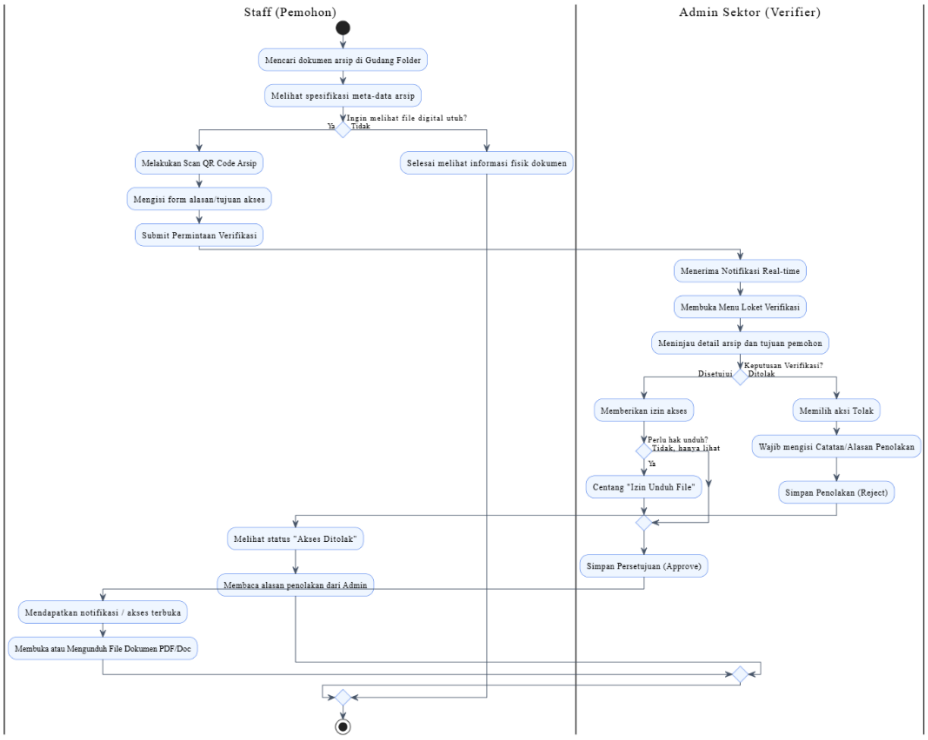

Figure 4. ***Activity Diagram*** **Alur Verifikasi Akses Dokumen (*QR Code*)**

2. Perancangan Basis Data *(Entity Relationship Diagram)*

Untuk menjamin integritas informasi, perancangan basis data dilakukan guna memastikan dokumen digital terhubung secara akurat dengan otorisasi pengguna, struktur klasifikasi lintas subbagian, mekanisme verifikasi, dan rekam jejak operasional. Struktur relasional antar-tabel ini divisualisasikan

melalui *Entity Relationship Diagram* (ERD). Penggunaan ERD bertujuan untuk mendeskripsikan model data logis berbasis entitas dan kardinalitas relasinya (Fathansyah, 2018). Berdasarkan arsitektur basis data aplikasi SIKANTI yang menerapkan isolasi tingkat unit (*database isolation*), terdapat tujuh entitas utama yang saling berelasi, yaitu:

1. Tabel subbagians: Merupakan entitas fundamental yang menjadi jangkar dari mekanisme *database isolation*. Tabel ini menyimpan data unit kerja pelaksana (seperti SDM, Humas, Hukum, Umum, Keuangan) melalui atribut nama_subbag dan kode_klasifikasi. Entitas ini berelasi dengan tabel users, kategoris, dan arsips untuk memastikan data tidak bocor lintas batas unit.
2. Tabel users: Entitas pengelola identitas dan kredensial autentikasi. Untuk mengeksekusi mekanisme *Role-Based Access Control* (RBAC), tabel ini dilengkapi atribut role bertipe ENUM ('Admin', 'Staff'). Adanya *Foreign Key* subbag_id pada tabel ini memastikan bahwa setiap akun yang didaftarkan melalui *closed-loop provisioning* langsung terikat secara permanen pada unit kerja administrator pembuatnya.
3. Tabel kategoris: Entitas master untuk menyusun pohon hierarki direktori (*folder tree*). Atribut utamanya meliputi nama_kategori dan deskripsi. Entitas ini juga diikat oleh subbag_id sehingga setiap unit kerja memiliki struktur Gudang Foldernya masing-masing secara independen.
4. Tabel arsips: Entitas sentral berskala besar yang menyimpan *metadata* dokumen digital. Tabel ini memiliki relasi *many-to-one* (banyak ke satu) dengan tabel kategoris (melalui kategori_id), tabel pengguna (user_id), dan tabel subbagian (subbag_id). Atribut spesifiknya sangat komprehensif untuk mendukung tata kelola JRA BPK, mencakup nomor_dokumen, nama_arsip, retensi_aktif, retensi_inaktif, nasib_akhir, lokasi_fisik, hingga direktori penyimpanan *soft file* (file_dokumen).
5. Tabel verifikasis: Entitas operasional yang bertindak sebagai mesin di balik gerbang verifikasi *QR Code*. Tabel ini menghubungkan dokumen (arsip_id) dengan pengguna peminjam (pemohon_id). Atributnya merekam tujuan_akses, status persetujuan (status ENUM: Tertunda, Disetujui, Ditolak), hak unduh (izin_unduh), serta catatan_admin yang memberikan jejak audit atas setiap keputusan otorisasi dokumen lintas unit.
6. Tabel riwayats: Entitas ini memfasilitasi fungsionalitas audit keamanan (*System Log*). Tabel ini mencatat secara otomatis setiap tindakan operasional di dalam sistem dengan merekam entitas pelaku (melalui user_id), jenis tindakan yang dilakukan (tipe_aksi), serta rincian kronologi tindakan (deskripsi).
7. Tabel pengaturans: Entitas skalar yang berdiri secara mandiri untuk menyimpan konfigurasi antarmuka dan identitas aplikasi secara global, meliputi atribut nama_aplikasi, sub_judul, dan logo_aplikasi.

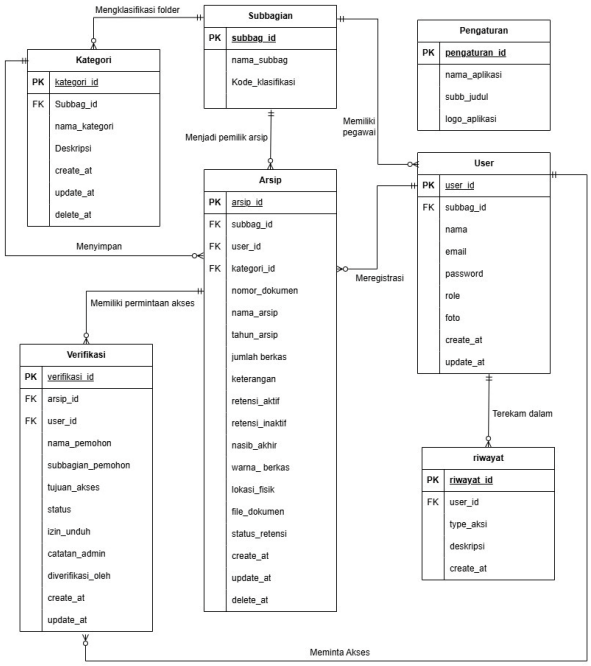

Figure 5. *Entity Relationship Diagram* Sistem SIKANTI

3 Pemodelan Arsitektur Sistem *(Class Diagram)*

Selain pemodelan basis data, perancangan arsitektur perangkat lunak secara logis dimodelkan menggunakan *Class Diagram*. Menurut Rosa & Shalahuddin (2018), *Class Diagram* merupakan diagram struktur statis yang mendeskripsikan representasi kelas, atribut, operasi (metode), serta hubungan antar-objek dalam sistem. Karena SIKANTI dikembangkan dengan pendekatan berorientasi objek (*Object-Oriented*) menggunakan *framework* Laravel, diagram ini memvisualisasikan bagaimana kelas-kelas *Model* berinteraksi secara programatis untuk mengeksekusi logika bisnis dan keamanan.

Sistem dimodelkan ke dalam tujuh kelas utama yang merepresentasikan entitas sistem secara utuh: Subbagian, User, Kategori, Arsip, Verifikasi, Riwayat, dan Pengaturan. Setiap kelas dibekali dengan atribut yang merepresentasikan struktur data, serta operasi spesifik (*methods*) yang mengendalikan alur logika sistem. Sebagai contoh:

1. Kelas User & Subbagian: Mengelola logika autentikasi dan otorisasi. Metode login dan logout pada kelas User bekerja secara integratif dengan kelas Subbagian untuk memastikan terciptanya sesi kerja yang terisolasi (*session isolation*) berdasarkan ID unit masing-masing.
2. Kelas Arsip & Kategori: Menangani manajemen direktori digital. Kelas Arsip memiliki metode fungsional seperti *uploadFile*, editMetadata, dan *generateQR*. Hubungan asosiasi antara kedua kelas ini memastikan integritas penempatan dokumen di dalam pohon hierarki folder yang tepat.
3. Kelas Verifikasi: Merupakan kelas krusial yang mengendalikan gerbang keamanan. Metode *requestAccess*, *approveAccess*, dan *rejectAccess* pada kelas ini mengatur sirkulasi peminjaman dokumen antar-unit dengan memvalidasi setiap permintaan akses melalui *QR Code*.
4. Kelas Riwayat & Pengaturan: Kelas Riwayat memiliki operasi logActivity() yang dipicu secara otomatis oleh sistem (*event-driven*) setiap kali metode pada kelas lain dieksekusi, guna menjaga akuntabilitas audit. Sementara kelas Pengaturan menyediakan metode updateAppIdentity() untuk mengelola visualisasi antarmuka sistem.

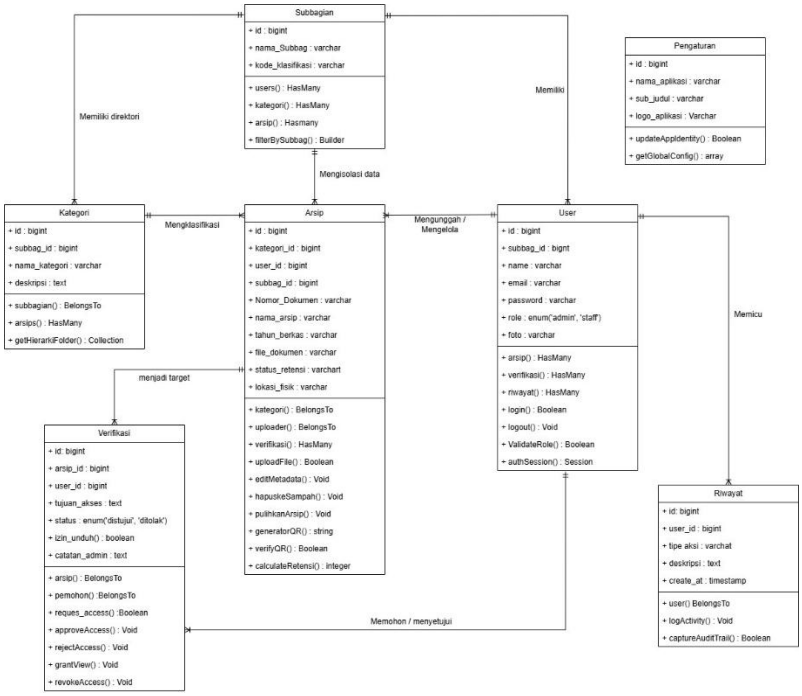

Figure 6. *Class Diagram* Sistem SIKANTI

4. Perancangan Antarmuka *(User Interface)*

Desain antarmuka difokuskan pada tingkat ketergunaan (*usability*) dan efisiensi interaksi manusia dan komputer (Shneiderman et al., 2017). Untuk menunjang tata kelola kearsipan tingkat institusional, antarmuka SIKANTI dirancang dengan pendekatan *SaaS-style minimalism* yang modern, bersih, dan *fully responsive*. Hal ini sejalan dengan prinsip bahwa desain antarmuka pengguna (UI) yang intuitif akan memudahkan interaksi dengan sistem, yang secara langsung berkontribusi pada peningkatan kualitas pengalaman pengguna (UX) secara keseluruhan (Halimah & Hamdani, 2025). Berbeda dengan sistem

konvensional yang statis, arsitektur visual SIKANTI menerapkan protokol *UI Isolation* yang beradaptasi secara dinamis menyesuaikan tingkat otoritas (*Role-Based Access Control*) dan unit kerja pengguna ketika sesi *login* dimulai.

Bagi Administrator Subbagian, layar utama (*dashboard*) menyajikan panel eksekutif interaktif yang memuat informasi *real-time* melalui representasi kartu statistik, seperti total volume folder, dokumen aktif, dan status penyusutan retensi. Halaman dashboard menampilkan ringkasan data metrik kearsipan, status penyusutan, dan visualisasi operasional administrator berdasarkan sesi login. Sementara itu, bagi pengguna tingkat Staf, antarmuka secara presisi menyembunyikan panel manajerial tersebut dan mengarahkan fokus ruang kerja secara eksklusif pada modul Gudang Folder. Fitur navigasi utama ditopang oleh *sidebar* cerdas yang menampilkan indikator Sektor Aktif pengguna (sebagai manifestasi visual dari *database isolation*), serta menyusun struktur direktori ke dalam bentuk pohon hierarki (*folder tree*) yang mengadopsi persis nomenklatur fisik BPK. Struktur navigasi aplikasi memisahkan arah informasi dan pembatasan akses modul antara tingkat Administrator dan pengguna tingkat Staf.

Untuk memfasilitasi temu balik informasi, antarmuka dilengkapi dengan bilah pencarian tunggal (*single search bar*) yang terintegrasi dengan algoritma *metadata* ganda guna mendukung pencarian dokumen secara instan berdasarkan kata kunci dan filter spesifik (Rahmansyah & Nopriani, 2025). Pengalaman pengguna (UX) semakin disempurnakan dengan dukungan notifikasi *toast* elegan sebagai umpan balik aksi sistem dan kemudahan navigasi registrasi dokumen secara terstruktur. Antarmuka pencarian sistem mengadopsi nomenklatur fisik lemari ordner ke dalam struktur pohon hierarki digital, yang diintegrasikan dengan bilah pencarian tunggal (single search bar) berbasis metadata ganda.

C. Implementasi Sistem *(Construction of Prototype)*

Tahap implementasi merupakan proses translasi dari hasil perancangan logis ke dalam bentuk kode program fungsional. Menurut Pressman & Maxim (2020), dalam pendekatan *prototyping*, tahap konstruksi difokuskan pada pengembangan representasi aspek-aspek perangkat lunak yang terlihat langsung oleh pengguna akhir. Prototipe SIKANTI direalisasikan menggunakan bahasa pemrograman PHP dengan *framework* Laravel yang mengadopsi pola arsitektur *Model-View-Controller* (MVC). Implementasi antarmuka dan fitur-fitur utama pada prototipe ini meliputi:

1. Gerbang Autentikasi Terisolasi (*Login Page*)

Halaman ini memvalidasi kredensial pengguna sebagai wujud implementasi prinsip kerahasiaan (*confidentiality*) guna mencegah akses ilegal (Rosa & Shalahuddin, 2018). SIKANTI menerapkan *closed-system architecture* di mana Akun Administrator Utama untuk kelima subbagian (SDM, Humas, Hukum, Umum, dan Keuangan) didaftarkan secara pra-produksi melalui mekanisme *database seeding*. Saat sistem diimplementasikan ke peladen produksi (*production server*), rute registrasi publik (seperti rute /register) dihapus secara permanen. Pendekatan *hardened deployment* ini secara proaktif meminimalkan area serangan (*attack surface*), menutup total celah eksploitasi pembuatan akun admin ilegal secara *remote*, dan mengunci akses sistem murni hanya untuk personel institusi yang telah terverifikasi.

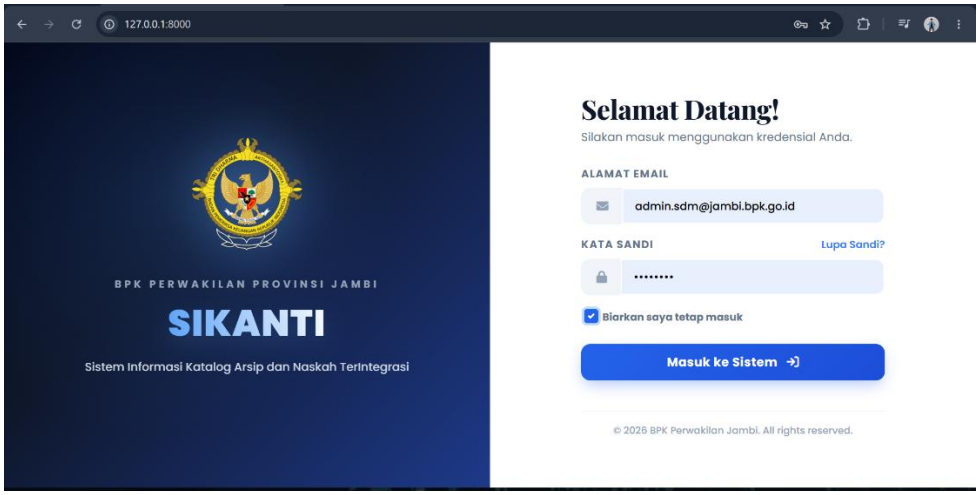

Figure 7. Tampilan *Login* SIKANTI

2. Beranda Eksekutif Terintegrasi (*Dashboard*)

Halaman *dashboard* bertindak sebagai pusat informasi visual (*data visualization*). Halaman ini menampilkan metrik kinerja kearsipan secara *real-time*, seperti kartu statistik volume folder, total dokumen aktif, status penyusutan (JRA), hingga lini masa aktivitas terkini. Menurut Shneiderman et al. (2017), antarmuka *dashboard* yang informatif sangat krusial untuk memberikan ringkasan status operasional secara cepat (*at-a-glance*) guna mendukung pengambilan keputusan manajerial yang akurat.

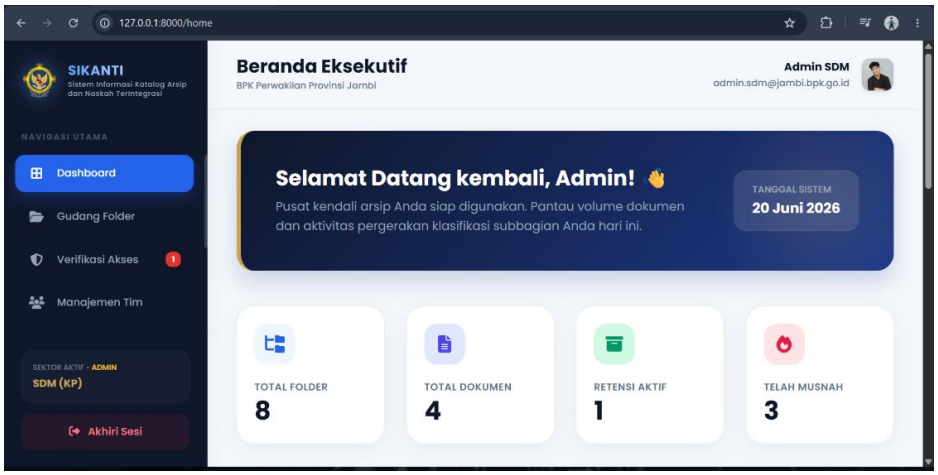

Figure 8. *Dashboard* Utama SIKANTI

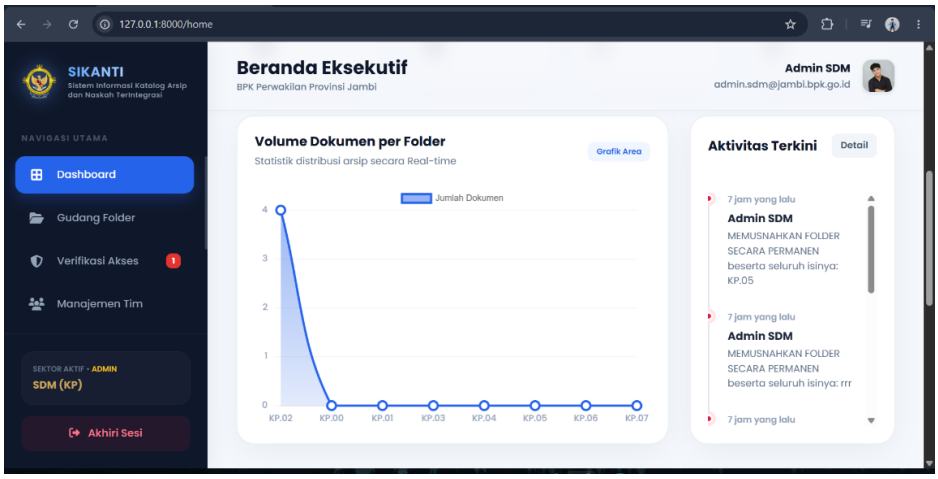

Figure 9. *Dashboard* Utama SIKANTI

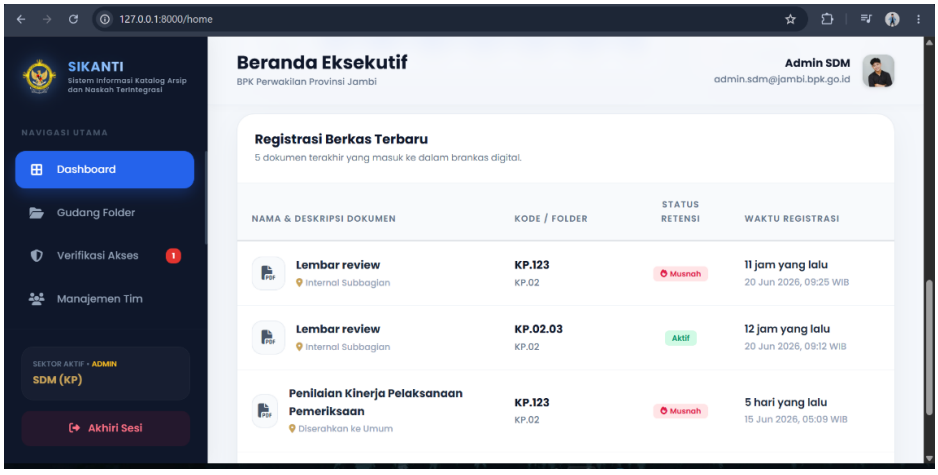

Figure 10. *Dashboard* Utama SIKANTI

3. Gudang Folder (*Hierarchical Directory, Folder Content, & Database Isolation*)

Fitur ini mereplikasi ruang arsip fisik ke dalam pohon hierarki direktori (*folder tree*) menggunakan nomenklatur klasifikasi resmi BPK (seperti KP, KU, HM, HK, dan UM) (Muhidin & Winata, 2016). Melalui mekanisme *database isolation*, sistem menyaring hak akses secara otomatis sehingga setiap subbagian hanya dapat melihat direktori internalnya. Di dalam struktur ini, implementasi folder isi dirancang untuk menampung lembar dokumen digital (*file* spesifik) secara teratur. Akun Staf (*User*) yang telah didaftarkan oleh Administrator melalui skenario *closed-loop provisioning* diberikan otoritas operasional penuh (*full access*) terhadap menu Gudang Folder beserta seluruh sub-folder dan dokumen di dalamnya. Hal ini memungkinkan Staf untuk mengeksekusi tindakan penelusuran, pengunggahan naskah terstruktur, hingga pembaruan dokumen secara mandiri, sekaligus mengoptimalkan rasio penemuan kembali (*finding ratio*) informasi pada unit kerja terkait tanpa mengorbankan integritas data lintas subbagian.

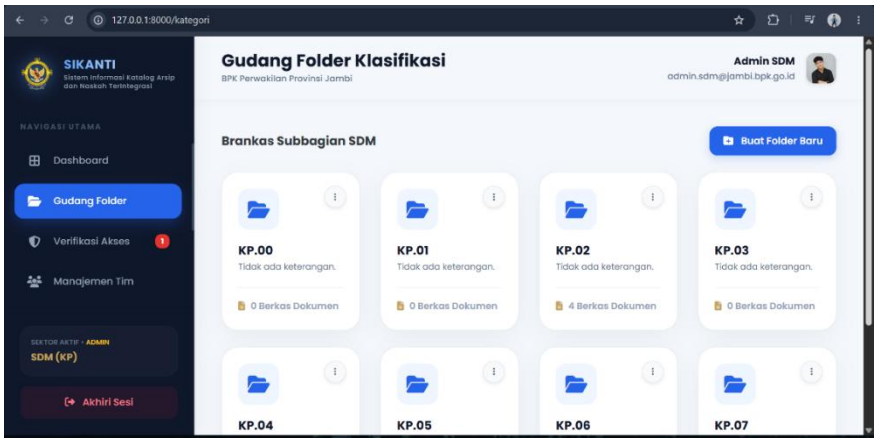

Figure 11. Gudang Folder SIKANTI

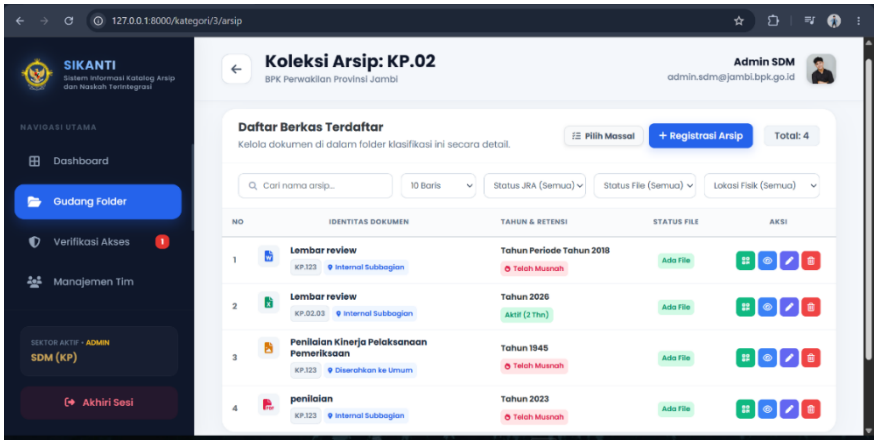

Figure 12. Folder Isi Sistem SIKANTI

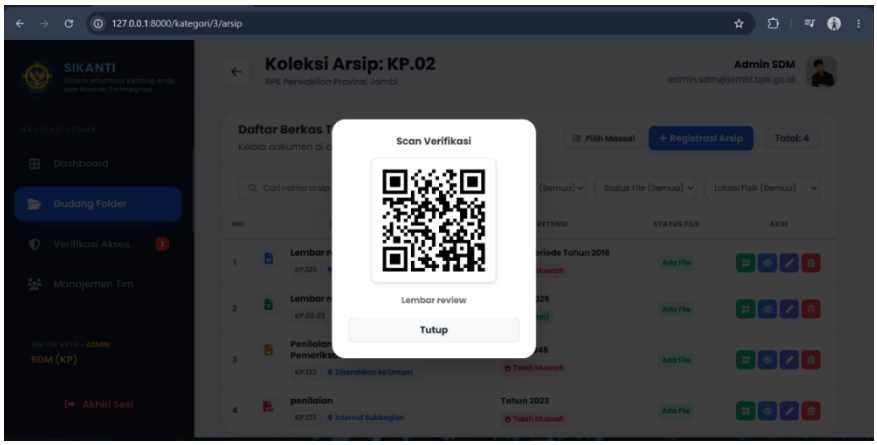

Figure 13. Kode Verifikasi QR Sistem SIKANTI

4. Gerbang Verifikasi Akses (*QR Code Gateway*)

Fitur ini merealisasikan mekanisme otorisasi akses menggunakan perangkat pemindai. Administrator dibekali antarmuka khusus untuk memantau permintaan izin akses yang masuk secara *real-time*. Sesuai dengan prinsip *gatekeeping*, admin memegang kendali mutlak untuk menyetujui atau menolak permintaan akses dokumen berdasarkan urgensi pemohon (Sutirman et al., 2016).

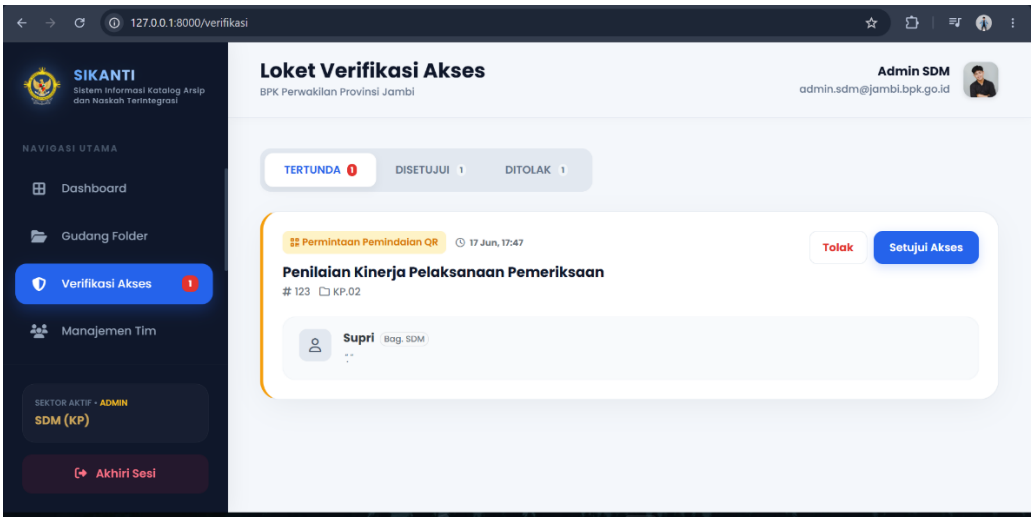

Figure 14. Gerbang Verifikasi Akses Sistem SIKANTI

5. Manajemen Tim (*Closed-Loop Provisioning*)

Fitur ini merealisasikan skenario *Role-Based Access Control* (RBAC) melalui mekanisme pendaftaran akun secara tertutup. Administrator Subbagian memiliki otoritas untuk mendaftarkan akun Staf pelaksana, mereset kata sandi, dan menentukan batasan hak akses. Pengaturan peran yang ketat sangat esensial untuk menjamin segregasi tugas dan mencegah tumpang tindih kewenangan dalam pengelolaan data sensitif (Gunawan et al., 2026).

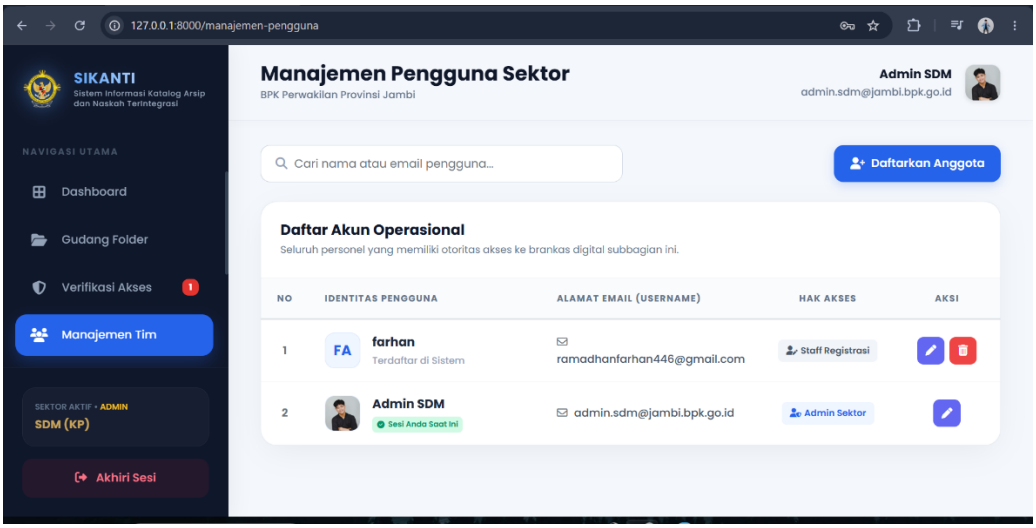

Figure 15. Manajemen Pengguna Sistem SIKANTI

6. Kelola Sampah (*Data Recovery & Retention*)

Fitur ini mengakomodasi siklus hidup dokumen atau penyusutan arsip digital. Dokumen yang dihapus tidak langsung musnah, melainkan ditampung sementara di modul "Kelola Sampah". Hal ini dirancang sebagai langkah mitigasi pemulihan data (*data recovery*) apabila terjadi galat manusia (*human error*), sekaligus mendukung implementasi Jadwal Retensi Arsip (JRA) instansi (Pressman & Maxim, 2020) .

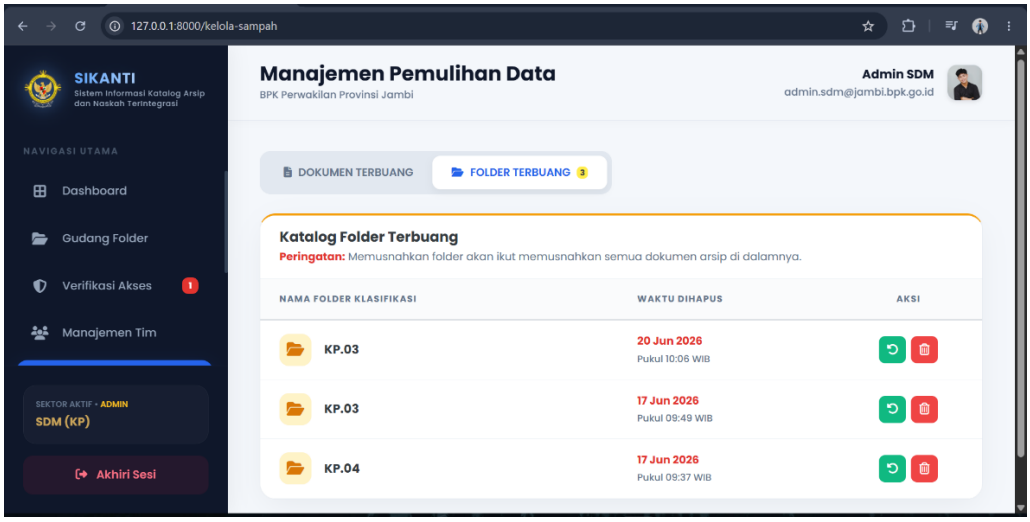

Figure 16. Kelola Sampah Sistem SIKANTI

7. Jejak Aktivitas (*System Audit Trail*)

Halaman riwayat berfungsi sebagai *audit trail* yang merekam setiap rekam jejak operasional di dalam sistem secara kronologis. Pencatatan ini mencakup identitas pelaku, jenis tindakan (*Create, Update, Delete*), dan stempel waktu yang presisi. Pendokumentasian aktivitas ini esensial untuk menjamin transparansi, akuntabilitas, dan deteksi dini terhadap anomali akses basis data (Fathansyah, 2018).

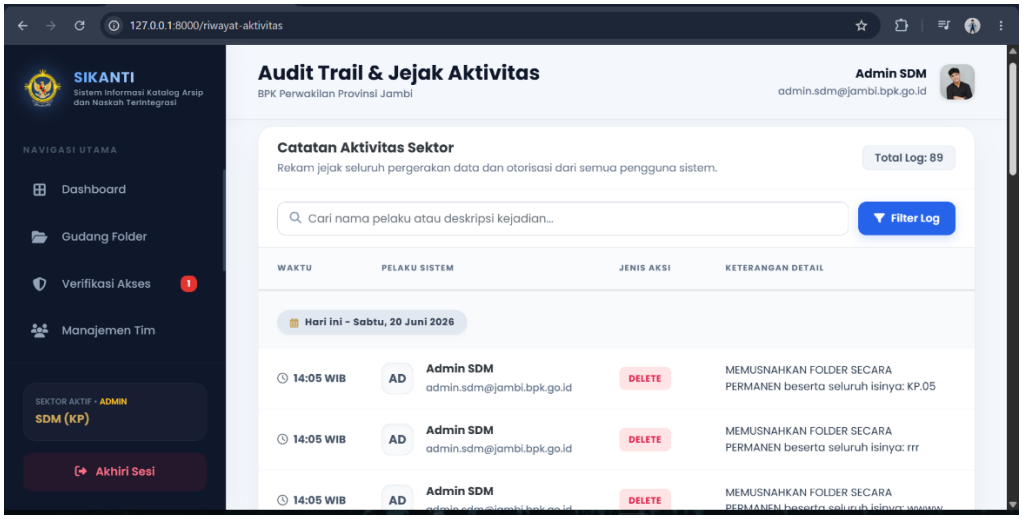

Figure 17. Riwayat *(System Log)* Sistem SIKANTI

8. Pengaturan Identitas Aplikasi dan Profil (*System Customization*)

Modul pengaturan memberikan fleksibilitas bagi administrator untuk melakukan modifikasi identitas visual aplikasi seperti perubahan nama dan logo aplikasi secara dinamis tanpa perlu mengubah kode sumber (*hard-coding*). Selain mengelola parameter global sistem, fitur ini juga diintegrasikan dengan manajemen profil personal tingkat manajerial, di mana administrator dapat memperbarui foto profil serta melakukan pergantian kata sandi (*password*). Fungsionalitas pembaruan sandi secara mandiri ini merupakan langkah preventif yang krusial dalam pemeliharaan keamanan kredensial (*account security hygiene*) guna memitigasi risiko kebocoran akses di lingkungan instansi (Rosa & Shalahuddin, 2018).

Figure 18. Pengaturan Sistem Sistem SIKANTI

9. Antarmuka Aksesibilitas Pegawai (*Mobile View-Only Interface*)

Setelah pegawai melakukan pemindaian Kode QR melalui perangkat seluler (*smartphone*), sistem akan menampilkan antarmuka khusus yang bersifat *mobile-friendly*. Jika permintaan akses disetujui oleh Administrator, pegawai dapat melihat naskah digital melalui penampil dokumen terintegrasi. Implementasi ini menerapkan *UI Isolation* yang sangat ketat, di mana tampilan pada perangkat pegawai dibatasi hanya pada hak tinjau (*view-only*), tanpa navigasi manajerial, dan secara otomatis memblokir fungsi unduh (*download*) guna mencegah penyalahgunaan informasi di luar lingkup otoritas (Shneiderman et al., 2017).

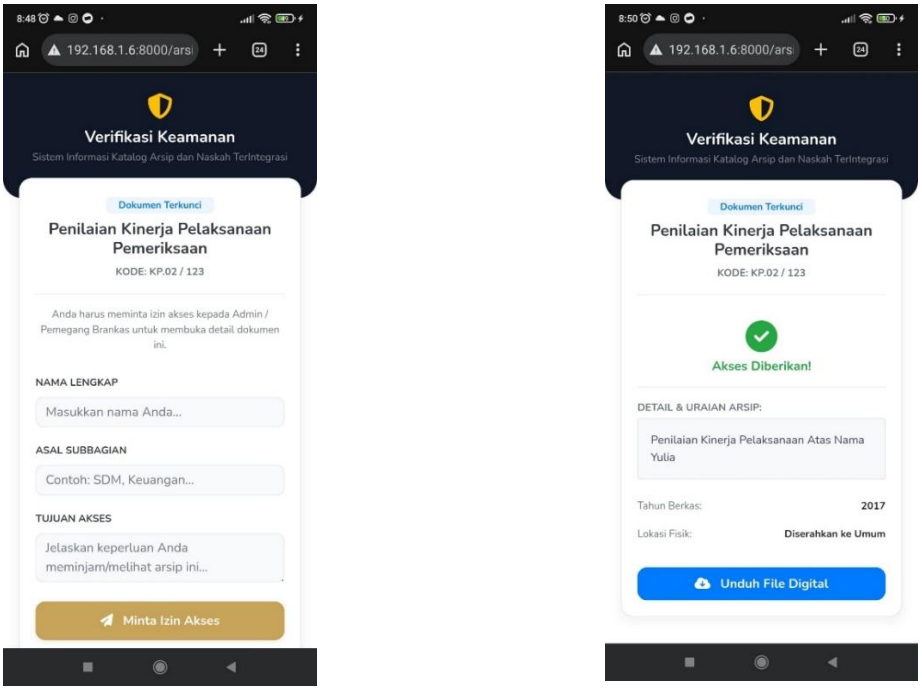

Figure 19. Antarmuka Aksesibilitas Pegawai Sistem SIKANTI

10. Antarmuka Ruang Kerja Staf (*Staff User Interface*)

Sebagai wujud implementasi *UI Isolation*, navigasi pada akun level Staf dibatasi secara eksklusif pada dua menu utama: *Dashboard* dan *Gudang Folder* (Shneiderman et al., 2017). Meskipun akses manajerial

tingkat atas disembunyikan, Staf yang terdaftar tetap diberikan otoritas operasional penuh (*full access*) pada modul *Gudang Folder*. Hal ini memberikan keleluasaan bagi pengguna untuk mengeksekusi siklus pengelolaan arsip secara mandiri khusus di dalam lingkup direktori subbagiannya masing-masing.

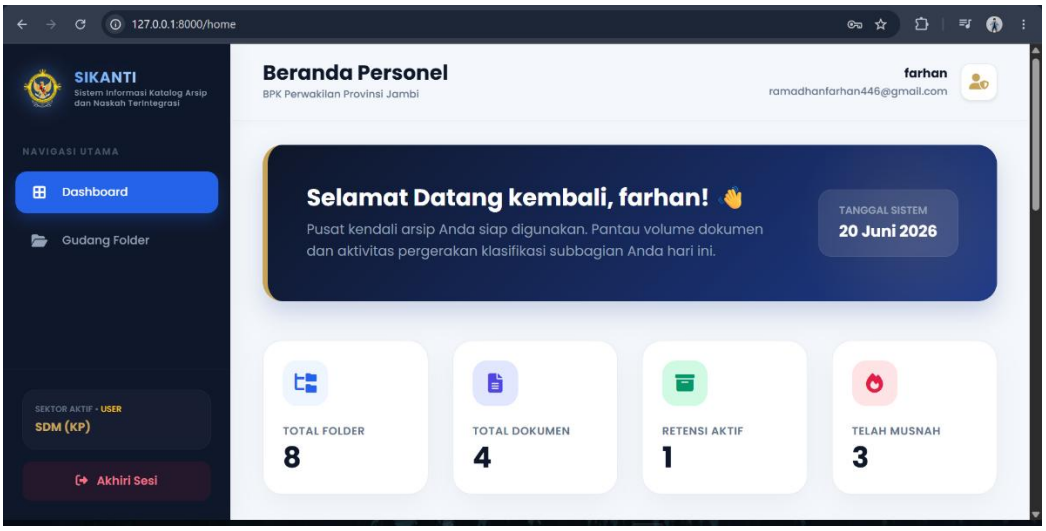

Figure 20. Antarmuka Pengguna Level Staf SIKANTI

D. Hasil Pengujian Sistem (Tabel Black-Box)

Pengujian fungsionalitas aplikasi SIKANTI dilakukan menggunakan metode *Black-Box Testing* untuk memastikan seluruh modul, fitur keamanan, dan navigasi antarmuka berjalan sesuai spesifikasi kebutuhan pengguna tanpa adanya galat (*error*). Pengujian ini mencakup simulasi operasi harian administrator maupun batasan akses untuk level staf.

Table 1. Hasil Pengujian *Black-Box* Sistem SIKANTI

No	Modul/Fitur Utama	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status
1	Autentikasi	Login menggunakan email & sandi terdaftar	Sistem memvalidasi kredensial & mengarahkan ke Dashboard sesuai role.	Sesuai	Valid
2	Hardenend Deployment	Mengakses rute /register secara langsung via UR	Sistem menolak akses dan mengembalikan pesan error / 404.	Sesuai	Valid
3	Database Isolation	Login Admin SDM & melihat isi Gudang Folder	Sistem hanya menampilkan direktori KP milik Subbagian SDM.	Sesuai	Valid
4	UI Isolation	Login menggunakan akun level Staf	Menu Pengaturan, Manajemen Tim, & Sampah disembunyikan otomatis.	Sesuai	Valid
5	Registrasi Arsip	Mengunggah soft file dokumen beserta metadata JRA	Data tersimpan ke basis data & file terunggah ke *storage.*	Sesuai	Valid

No	Modul/Fitur Utama	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status
6	Verifikasi QR	Pegawai luar memindai QR Code & mengirim form izin	Notifikasi *real-time* muncul di layar Admin pemilik arsip.	Sesuai	Valid
7	Otoritas Akses	Admin menyetujui izin dengan opsi "Lihat Saja"	Halaman dokumen di HP pegawai terbuka tanpa tombol unduh.	Sesuai	Valid
8	Manajemen Tim	Admin mendaftarkan akun Staf baru (*Closed-loop*)	Akun baru berhasil dibuat & otomatis terikat ke subbagiannya.	Sesuai	Valid
9	Kelola Sampah	Menghapus dokumen (*soft delete*) & melakukan *restore*	Dokumen masuk ke *Recycle Bin* & dapat dipulihkan kembali.	Sesuai	Valid
10	Jejak Aktivitas	Melakukan aksi tambah, edit, atau hapus arsip	Sistem mencatat log aktivitas (pelaku, aksi, stempel waktu).	Sesuai	Valid
11	Pengaturan Utama	Mengubah logo instansi atau kata sandi profil	Tampilan identitas aplikasi & kredensial berhasil diperbarui.	Sesuai	Valid

Hasil Pengujian *Black-Box* Sistem SIKANTI (Triatama et al., 2026).

Berdasarkan pengujian pada Tabel 1, seluruh fungsionalitas sistem, baik operasional maupun pengamanan akses, dinyatakan valid dan berhasil berjalan sesuai dengan perancangan arsitektur sistem.

E. Uji Temu Balik Informasi (*Information Retrieval*)

Klaim efektivitas percepatan pencarian dokumen diukur melalui pengujian perbandingan waktu penemuan kembali arsip (*retrieval time*) antara metode konvensional (pencarian ordner fisik) dengan pencarian digital pada SIKANTI. Pengujian dilakukan secara acak terhadap 5 sampel dokumen naskah dinas.

Table 2. Hasil Uji Perbandingan Waktu Temu Balik Arsip

No	Kode / Jenis Arsip	Pencarian Manual Fisik	Pencarian SIKANTI	Persentase Efisiensi
1	KP.02 / Lembar Disposisi	12,5 Menit	2,1 Detik	99,72%
2	KP.04 / Bezetting Pegawai	15,0 Menit	3,4 Detik	99,62%
3	KP.05 / Kenaikan Gaji Berkala	8,0 Menit	1,8 Detik	99,62%
4	KP.06 / Surat Keterangan Bekerja	11,2 Menit	2,5 Detik	99,62%
5	KP.07 / Penyampaian Konsep Surat Tugas	14,0 Menit	2,9 Detik	99,65%
	Rata-Rata	**12,14 Menit**	**2,54 Detik**	**99,65%**

Hasil Uji Perbandingan Waktu Temu Balik Arsip (Triatama et al., 2026).

Data pada Tabel 2 menunjukkan bahwa SIKANTI secara objektif mampu meningkatkan kecepatan penemuan dokumen dari rata-rata 12,14 menit menjadi hanya 2,54 detik per pencarian. Peningkatan efisiensi sebesar 99,65% ini membuktikan bahwa algoritma pencarian metadata ganda sistem beroperasi secara optimal.

F. Evaluasi Penerapan Metode *Prototyping*

Penggunaan metode *prototyping* memfasilitasi komunikasi iteratif antara pengembang dan pengguna akhir. Umpan balik nyata (*real feedback*) yang diperoleh melalui kuesioner evaluasi dari para staf dan administrator langsung diimplementasikan sebagai dasar perbaikan (*refinement*). Pendekatan ini dilakukan untuk memastikan kesesuaian kinerja sistem dengan rutinitas operasional instansi, mengoptimalkan pelaporan manajerial, serta menekan potensi resistensi adaptasi pengguna baru.

Table 3. Matriks Iterasi Pengembangan Prototipe SIKANTI

Iterasi	Tahapan Prototyping	Masukan / *Feedback* Pengguna	Perbaikan / *Action Refinement*
Ke-1	Quick Design & Initial Prototype	Pengguna tingkat staf merasa bingung jika tata letak direktori folder dibuat dinamis tanpa pola nomenklatur instansi yang baku.	Mengonstruksi struktur pohon direktori (*folder tree*) yang persis mereplikasi nomenklatur fisik pada lemari arsip BPK (seperti KP, KU, HM).
Ke-2	Construction & Evaluation	Pengguna tingkat staf merasa terganggu dengan banyaknya navigasi menu manajerial yang tampil di layar tetapi tidak dapat diakses.	Menerapkan skenario *UI Isolation* dengan menyembunyikan menu Pengaturan, Jejak Aktivitas, dan Manajemen Tim secara otomatis pada sesi akun level Staf.
Ke-3	Performance & Feature Refinement	Pengguna tingkat staf merasa terganggu dengan banyaknya navigasi menu manajerial yang tampil di layar tetapi tidak dapat diakses.	Menerapkan skenario *UI Isolation* dengan menyembunyikan menu Pengaturan, Jejak Aktivitas, dan Manajemen Tim secara otomatis pada sesi akun level Staf.
Ke-4	Final Delivery & User Onboarding	Pengguna menyarankan peningkatan kecepatan dan stabilitas aplikasi saat mengunggah arsip berukuran besar. Pengguna juga merekomendasikan penambahan fitur unduh (*download*) rekap arsip untuk memudahkan pelaporan berjenjang.	Mengoptimalkan konfigurasi peladen (*upload_max_filesize*) untuk stabilitas transfer *file* besar, serta merancang fungsionalitas ekspor data rekapitulasi arsip guna mendukung kebutuhan *monitoring* manajerial.

Matriks Iterasi Pengembangan Prototipe SIKANTI (Triatama et al., 2026).

G. Evaluasi Penerimaan Pengguna (*User Acceptance Testing*)

Pengujian akhir dilakukan melalui *User Acceptance Testing* (UAT) untuk mengukur tingkat kelayakan dan penerimaan aplikasi. Instrumen pengujian menggunakan kuesioner berskala Likert (1–5) yang diujikan kepada 10 responden representatif di lingkungan Sekretariat BPK Perwakilan Provinsi Jambi (terdiri dari Administrator Subbagian, Staf Pelaksana, dan Pengelola IT).

Table 4. Hasil Evaluasi *User Acceptance Testing* (UAT)

No	Indikator Evaluasi	Skor Rata-Rata (Skala 5)	Presentase Kelayakan	Kategori
1	Usability (Ketergunaan): Kerapian antarmuka & kemudahan penelusuran hierarki direktori.	4,13	82,50%	Sangat Layak
2	Security (Keamanan): Efektivitas kontrol otorisasi QR Code & isolasi data lintas subbagian.	4,13	82,50%	Sangat Layak
3	Efficiency (Efisiensi): Kecepatan sistem pencarian & pemantauan otomatis masa retensi (JRA).	4,38	87,50%	Sangat Layak
4	Adaptability (Adaptasi): Kesesuaian sistem dengan prosedur tata kelola kearsipan & kelayakan produksi.	4,38	87,50%	Sangat Layak
Rata-Rata Keseluruhan		**4,25**	**85,00%**	**Sangat Layak**

Hasil Evaluasi *User Acceptance Testing* (UAT) (Triatama et al., 2026).

Hasil perhitungan evaluasi UAT menghasilkan tingkat persentase penerimaan keseluruhan sebesar 85,00% (kategori Sangat Layak). Hal ini menegaskan bahwa sistem SIKANTI tidak hanya memenuhi standar keamanan teknis, tetapi juga secara praktis dapat diterima dengan baik dan dinilai siap untuk diimplementasikan secara penuh (*production*) guna mendukung tata kelola kearsipan digital instansi.

Simpulan

Berdasarkan hasil perancangan, pengujian, dan evaluasi, dapat disimpulkan bahwa penerapan metode *prototyping* berhasil diimplementasikan dalam membangun Sistem Informasi Katalog Arsip dan Naskah TerIntegrasi (SIKANTI) pada Sekretariat BPK Perwakilan Provinsi Jambi. Hasil pengujian *Black-Box* menunjukkan bahwa seluruh fungsionalitas operasional dan fitur keamanan sistem telah berjalan valid sesuai dengan spesifikasi kebutuhan pengguna. Dari aspek kinerja operasional, implementasi sistem temu balik informasi (*information retrieval*) berbasis metadata ganda terbukti secara terukur mampu meningkatkan efisiensi pencarian arsip, yakni memangkas rata-rata waktu penelusuran dari 12,14 menit secara manual menjadi 2,54 detik (tingkat efisiensi 99,65%).

Dari aspek tata kelola keamanan, integrasi arsitektur *Role-Based Access Control* (RBAC), *Hardened Deployment*, *Database Isolation*, dan *UI Isolation* yang dikombinasikan dengan gerbang verifikasi *QR Code* terbukti efektif dalam memperketat kontrol akses sirkulasi dokumen serta meminimalkan risiko kebocoran informasi sensitif lintas subbagian. Keberhasilan transisi alih media ini juga didukung oleh tingginya tingkat penerimaan pengguna. Berdasarkan hasil pengujian *User Acceptance Testing* (UAT), sistem SIKANTI memperoleh persentase kelayakan sebesar 85,00% (Sangat Layak), yang mengonfirmasi bahwa replikasi nomenklatur fisik ke dalam bentuk *folder tree* digital berhasil meminimalkan kurva pembelajaran (*learning curve*) dan menekan resistensi adaptasi pengguna.

Sebagai rekomendasi untuk menjamin keberlanjutan ekosistem kearsipan digital ini, instansi disarankan untuk menetapkan Standar Operasional Prosedur (SOP) baku terkait registrasi metadata dan

otomatisasi retensi arsip. Selain itu, program literasi teknologi yang berkelanjutan perlu dilaksanakan guna menyelaraskan kecakapan sumber daya manusia dengan kapabilitas sistem yang telah dibangun.

Pengakuan

Penulis mengucapkan puji syukur dan terima kasih yang sebesar-besarnya kepada Badan Pemeriksa Keuangan Perwakilan Provinsi Jambi, khususnya pimpinan dan seluruh staf Subbagian Sumber Daya Manusia (SDM), atas izin penelitian, fasilitas kelengkapan data, serta waktu yang diluangkan selama proses observasi dan evaluasi sistem. Ucapan terima kasih juga ditujukan kepada segenap sivitas akademika Universitas Islam Negeri Sulthan Thaha Saifuddin Jambi atas bimbingan akademis dan dukungan institusional yang diberikan selama masa penelitian ini berlangsung. Penghargaan setinggi-tingginya turut disampaikan kepada seluruh anggota tim peneliti dan pengembang (M. Farhan Ramadhan, M. Reno Novriansyah, Rayhan Syawal, Noviana Safitri, dan Nanda Septa Lian Sari) atas dedikasi serta kolaborasi yang solid dalam merealisasikan sistem SIKANTI dari tahap perancangan arsitektur hingga implementasi akhir.

Ucapan terima kasih secara khusus ditujukan kepada:

1. Bapak Aditya Wahyu Dewanggajati S.E., M.E., selaku mentor magang, yang telah memberikan bimbingan intensif, arahan strategis, masukan teknis, serta bantuan dalam proses analisis sistem yang sangat berharga bagi kelancaran pengembangan aplikasi ini.
2. M. Gusti Maulana, S.T., atas dedikasi dan kontribusi aktifnya dalam proses pengembangan aplikasi Sistem SIKANTI, serta atas arahan dan saran konstruktif yang sangat membantu dalam penyempurnaan dan keberhasilan aplikasi ini.
3. Bapak Heri Gunawan S.E., selaku staf Subbagian IT & Umum, atas wawasan teknis dan penjelasannya yang komprehensif mengenai tata kelola kearsipan instansi yang sangat membantu dalam perancangan alur keamanan sistem.

Penelitian dan pengembangan sistem ini didanai secara mandiri (dana pribadi) oleh para peneliti selama pelaksanaan program magang di Badan Pemeriksa Keuangan (BPK) Perwakilan Provinsi Jambi, dan tidak menerima pendanaan berupa hibah dari lembaga donor atau pihak luar manapun.

Daftar Pustaka

Ebizeen, H., & Cahyono, A. D. (2025). Perancangan Sistem Informasi Manajemen Kearsipan Digital Kecamatan Belantikan Raya Menggunakan Metode Prototype. *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, *14*(3), 2247–2264.

Fathansyah. (2018). *Basis data: Revisi ketiga* (3rd ed.). Bandung: Informatika.

Giawa, Y., Raudhah, Abdy, S., & Satria Alasi, T. (2026). Perancangan Sistem Informasi Surat Masuk Dan Surat Keluar Secara Multiuser Menggunakan Metode Prototype Pada Kantor Desa Fondrakoraya Kabupaten Nias Selatan. *Jurnal Informatika Logika*, *3*(1), 14–20.

Gunawan, A., Mulyanto, A., & Iskandar, R. (2026). Perancangan Role Based Access Control Pada Sistem Informasi Pengelolaan Data Pmb Berbasis Web. *Jurnal Informatika SIMANTIK*, *11*(1), 37–41.

Halimah, N., & Hamdani, A. (2025). Perancangan Ui Dan Ux E Arsip Surat Studi Kasus Kecamatan Suboh. *Akiratech: Journal of Computer and Electrical Engineering*, *2*(3), 116–126. https://doi.org/10.63935/akiratech.v2i3.227

Iqbal, M., Daraba, D., & Indrayani, E. (2024). Implementasi Kebijakan Tentang Pedoman Penerapan Sistem Informasi Kearsipan Dinamis Terintegrasi Dalam Upaya Pengelolaan Naskah Dinas Elektronik Di Sekretariat Daerah Kota Bekasi. *Jurnal Kajian Pemerintah (JKP)*, *10*(2), 109–122.

Muhidin, S. A., & Winata, H. (2016). *Manajemen kearsipan: Untuk organisasi publik, bisnis, sosial, politik, dan kemasyarakatan*. Bandung: Pustaka Setia.

Pemerintah Republik Indonesia. (2009). Undang-Undang Republik Indonesia Nomor 43 Tahun 2009 tentang Kearsipan. In *Lembaran Negara Republik Indonesia Tahun 2009 Nomor 152* (No. 43). Indonesia.

Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: a practitioner's approach* (9th ed.). McGraw-Hill Education.

Rahmansyah, A., & Nopriani, F. (2025). Perancangan User Interface Sistem E-Arsip Menggunakan Metode Human Centered Design Pada Komplek Pergudangan Perusahaan Umum Bulog Palembang. *Prosiding Seminar Nasional Sains dan Teknologi Seri III Fakultas Sains dan Teknologi*, *2*(1), 555–569.

Rosa, A. S., & Shalahuddin, M. (2018). *Rekayasa perangkat lunak: Terstruktur dan berorientasi objek* (Revisi). Bandung: Penerbit Informatika.

Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2017). *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (6th ed.). Pearson.

Sugiyono. (2019). *Metode penelitian kuantitatif, kualitatif, dan R&D*. Bandung: Alfabeta.

Sutirman, Wijayanti, N. S., & Purwanto. (2016). Studi Tentang Implementasi Sistem Manajemen Arsip Elektronik Pada Kantor Pemerintahan Kota Yogyakarta. *Efisiensi - Kajian Ilmu Administrasi*, *14*(1), 70–97.

Yogopriyatno, J., Nursanty, & Roeliana, L. (2024). Kesiapan Implementasi Sistem Informasi Kearsipan Dinamis Terintegrasi (Srikandi) Di Kota Bengkulu. *Jurnal Administrasi dan Kesekretarisan*, *9*(1), 31–44.