

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Hasil Penelitian

Penelitian ini dilaksanakan di UPTD Puskesmas Panjalu yang beralamat di Desa Panjalu, Kec. Panjalu, Kabupaten Ciamis, Jawa Barat 46264. Penelitian ini berfokus pada hasil implementasi dan pengujian yang telah dikembangkan. Setelah melalui serangkaian tahapan dalam model pengembangan Sistem Informasi Puskesmas yang telah dibangun. Proses pengembangan sistem ini menggunakan model *Rapid Application Development* (RAD), yang terdiri dari empat fase utama: Perencanaan Syarat-syarat (*Requirements Planning*), Workshop Desain (*Design Workshop*), Implementasi (*Implementation*), dan Pengujian (*Testing*). Setiap fase akan diuraikan secara rinci dalam bab ini.

1. Perencanaan Syarat-Syarat

Pada tahap awal pengembangan, telah dilaksanakan proses perencanaan syarat-syarat untuk mengidentifikasi kebutuhan esensial dari sistem informasi yang akan dibangun. Proses ini, sebagaimana diuraikan pada BAB III, melibatkan analisis masalah dan analisis kebutuhan sistem yang didasarkan pada studi literatur terhadap regulasi yang berlaku serta observasi dan wawancara dengan pihak Puskesmas.

a. Hasil Analisis Masalah

Dari hasil pertemuan dan diskusi dengan pengguna, teridentifikasi beberapa permasalahan utama yang menjadi dasar pengembangan sistem. Tujuan utama sistem adalah untuk mendigitalisasikan proses pencatatan, pelaporan, dan penyebaran informasi kesehatan untuk mengatasi tantangan-tantangan berikut:

- 1) Proses pencatatan data, terutama untuk skrining lansia dan kegiatan posyandu, masih dilakukan secara manual menggunakan buku catatan. Hal ini meningkatkan risiko kesalahan input data (human error), redundansi, dan kehilangan data fisik.

- 2) Penyebaran informasi kepada masyarakat, seperti jadwal kegiatan atau pengumuman kesehatan, masih bergantung pada media konvensional seperti papan pengumuman di puskesmas, yang jangkauannya sangat terbatas dan kurang efektif.
- 3) Belum ada kanal digital yang terstruktur bagi masyarakat untuk berinteraksi dengan puskesmas, seperti untuk mengonfirmasi kehadiran pada sebuah kegiatan, yang penting untuk perencanaan logistik.

Analisis ini didukung oleh instrumen penelitian berupa lembar observasi dan angket wawancara yang menghasilkan pemahaman mendalam mengenai alur kerja dan kebutuhan informasi dari setiap calon pengguna.

b. Hasil Analisis Kebutuhan Sistem

Berdasarkan analisis masalah tersebut, berhasil dirumuskan spesifikasi kebutuhan sistem yang dibagi menjadi dua kategori utama, yaitu kebutuhan fungsional dan non-fungsional.

1) Kebutuhan Fungsional

Kebutuhan fungsional merupakan hasil identifikasi fitur-fitur utama yang wajib tersedia agar sistem dapat beroperasi sesuai tujuan. Berikut adalah kebutuhan fungsional yang telah ditetapkan:

- a) Sistem harus mampu mengelola semua informasi identitas puskesmas, yang mencakup data "Tentang Kami", Visi dan Misi, dan informasi legalitas lainnya.
- b) Sistem harus menyediakan fitur untuk mendokumentasikan dan menampilkan jenis-jenis layanan kesehatan yang tersedia di puskesmas,
- c) Sistem harus mendukung Indeks Layanan Primer yang telah diatur dalam permenkes 19 Tahun 2024
- d) Sistem harus memiliki modul untuk mengelola dan menyebarkan informasi kepada masyarakat secara dinamis, seperti Pengumuman, Jadwal Kegiatan Posyandu, dan Galeri Foto.

- e) Sistem harus menyediakan fungsionalitas untuk publikasi artikel dan konten edukasi kesehatan yang dapat diakses oleh publik.
 - f) Terakhir sistem harus mampu menampilkan informasi dasar seperti alamat, jam operasional, dan detail kontak puskesmas secara akurat.
- 2) Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah syarat teknis dan kualitas yang menjadi standar sistem. Hasil identifikasi kebutuhan non-fungsional adalah sebagai berikut:

- a) Sistem diwajibkan memiliki sistem autentikasi pengguna berbasis peran (role-based access) untuk membatasi hak akses. Peran yang diidentifikasi adalah Super Administrator, Penulis, Bidan, Ketua Posyandu, Anggota Posyandu, dan Masyarakat, untuk memastikan keamanan data.
- b) Sistem harus dapat diakses dengan baik melalui berbagai perangkat (desktop, tablet, dan smartphone). Oleh karena itu, antarmuka website publik (frontend) dan panel admin (backend) harus dirancang secara responsif.

2. Tahap Workshop Desain

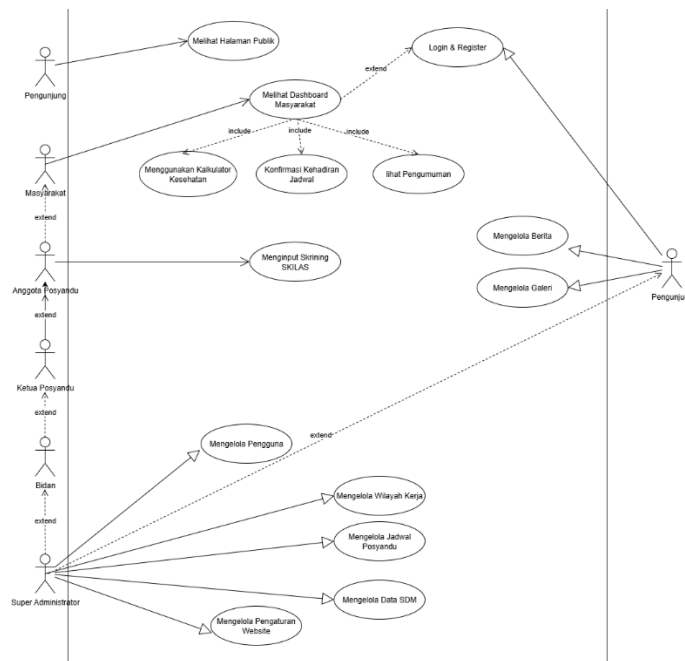
Pada tahap ini, dilakukan proses perancangan sistem secara iteratif dan kolaboratif, yang diwujudkan dalam bentuk *workshop desain*. Sesuai dengan model RAD, tahap ini fokus pada pembuatan prototipe visual dan fungsional yang kemudian dievaluasi dan disempurnakan berdasarkan masukan. Hasil dari tahap ini adalah sebuah desain arsitektur yang solid, antarmuka pengguna yang representatif, serta rancangan fitur yang detail.

a. Perancangan Arsitektur Sistem

Perancangan arsitektur sistem pada tahap ini diwujudkan dalam bentuk model Unified Modeling Language (UML) yang telah disesuaikan dengan kebutuhan sistem informasi Puskesmas Panjalu. Model ini digunakan untuk menggambarkan alur kerja, interaksi antar pengguna dengan sistem, serta hubungan antar komponen yang membentuk sistem secara keseluruhan.

Hasil perancangan mencakup beberapa diagram utama. *Use Case Diagram* telah dibuat untuk menggambarkan aktor yang terlibat, seperti superadmin, bidan, ketua posyandu, anggota posyandu, dan masyarakat, beserta interaksi mereka dengan fitur-fitur sistem. *Sequence Diagram* dihasilkan untuk menunjukkan urutan proses komunikasi antara pengguna, antarmuka, dan basis data pada berbagai skenario penggunaan. *Class Diagram* disusun untuk memodelkan struktur data, termasuk entitas, atribut, metode, serta hubungan antar kelas. Sementara itu, *Activity Diagram* digunakan untuk memvisualisasikan alur proses bisnis, mulai dari input data hingga publikasi informasi, sehingga memudahkan analisis alur kerja sistem.

1) *Use Case Diagram*



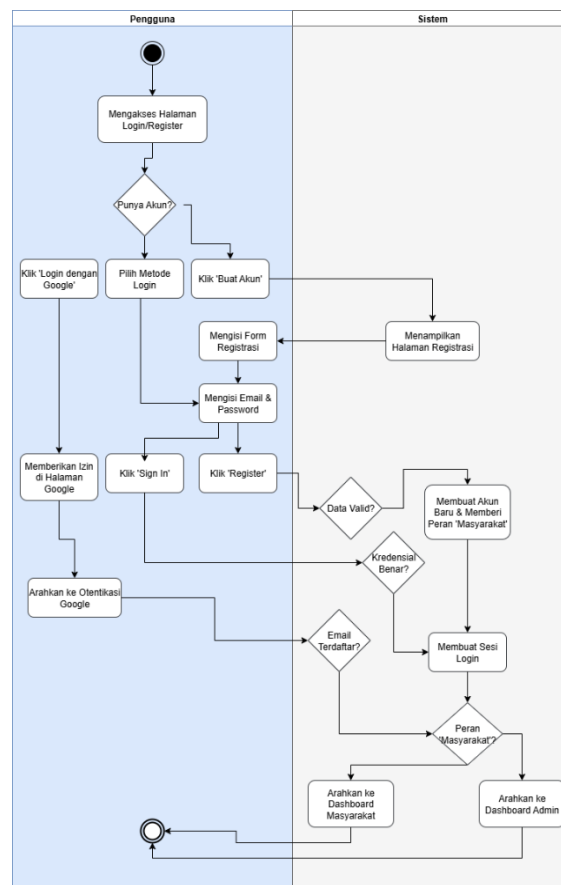
Gambar 4. 1 *Use Case Diagram*

Pertama, kita definisikan semua "pemain" dalam sistem kita, dari pengunjung biasa hingga Super Administrator. Kita akan menggunakan konsep pewarisan (generalisasi), di mana peran yang lebih tinggi otomatis memiliki semua kemampuan peran di bawahnya.

1) *Activity Diagram*

Diagram ini dibuat untuk menggambarkan alur kerja (*workflow*) dari proses-proses penting.

a) *Activity Diagram* Login dan register



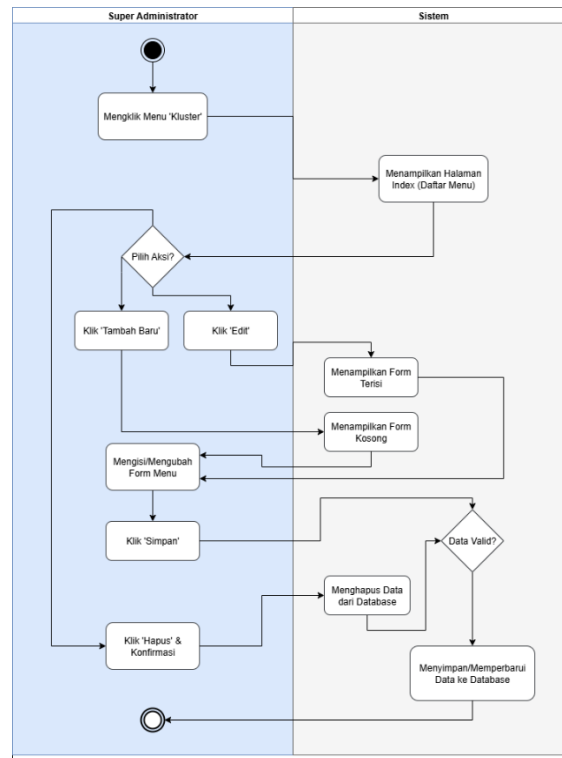
Gambar 4. 2 *Activity Diagram* Login dan Register

Diagram aktivitas ini menggambarkan alur kerja (*workflow*) yang terjadi saat seorang pengguna baru atau pengguna yang sudah ada berinteraksi dengan sistem otentikasi. Alur ini mencakup dua proses utama: Registrasi (pendaftaran akun baru) dan Login (masuk ke akun yang sudah ada).

Alur Registrasi Dimulai saat pengguna yang belum memiliki akun memilih untuk mendaftar. Mereka akan mengisi form yang berisi data diri (nama, email, password, serta pilihan desa dan dusun). Sistem kemudian akan memvalidasi data ini. Jika valid, sistem akan membuat akun baru, secara otomatis memberikan peran "Masyarakat", dan langsung mengarahkan pengguna ke dashboard mereka. Alur Login Dimulai saat pengguna yang sudah memiliki akun mencoba untuk masuk. Sistem menyediakan dua metode:

- Login Manual: Pengguna memasukkan email dan password. Sistem akan memverifikasi kredensial ini.
- Login dengan Google: Pengguna memilih untuk login menggunakan akun Google mereka. Sistem akan berkomunikasi dengan Google untuk otentikasi, lalu memeriksa apakah email Google tersebut sudah terdaftar di database. Jika proses login berhasil, sistem akan membuat sesi untuk pengguna dan mengarahkan mereka ke dashboard yang sesuai dengan perannya (Dashboard Masyarakat atau Dashboard Admin). Jika gagal (misalnya, password salah atau akun belum terdaftar), sistem akan menampilkan pesan error yang sesuai.

b) *Activity Diagram Kluster*



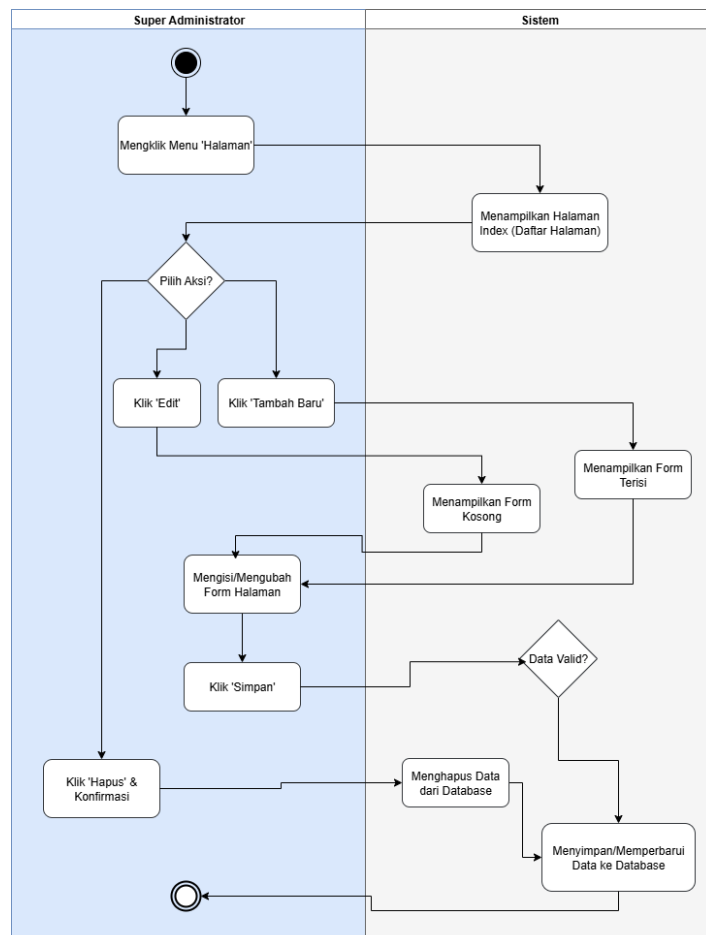
Gambar 4.3 *Activity Diagram Kluster*

Diagram aktivitas ini menggambarkan alur kerja yang dilakukan oleh seorang Super Administrator saat mengelola Kluster Menu, yaitu struktur navigasi utama yang ditampilkan di halaman depan website. Fitur ini memungkinkan admin untuk membuat, mengatur, memperbarui, dan menghapus item menu, termasuk menu induk dan sub-menunya (anak), untuk menciptakan sistem navigasi yang dinamis dan terstruktur. Alur kerja dari *Activity Diagram* ini diantaranya:

- Alur kerja dimulai saat admin mengakses halaman manajemen kluster, di mana sistem akan menampilkan daftar semua menu yang ada dalam format hirarkis. Admin kemudian dapat melakukan operasi CRUD standar:
- Menambah Menu yaitu Admin mengisi form untuk mendefinisikan menu baru, menentukan apakah itu menu induk atau sub-menu dari menu lain, dan menautkannya ke halaman internal atau URL eksternal

- Mengedit Menu yaitu Admin dapat mengubah detail dari menu yang sudah ada, seperti judul, urutan tampil, atau tautannya.
- Menghapus Menu Admin dapat menghapus item menu yang tidak lagi diperlukan.
- Setiap operasi akan divalidasi oleh sistem sebelum disimpan ke database, dan setelah berhasil, admin akan dikembalikan ke halaman daftar utama untuk melihat perubahan yang telah diterapkan

c) *Activity Diagram* Halaman



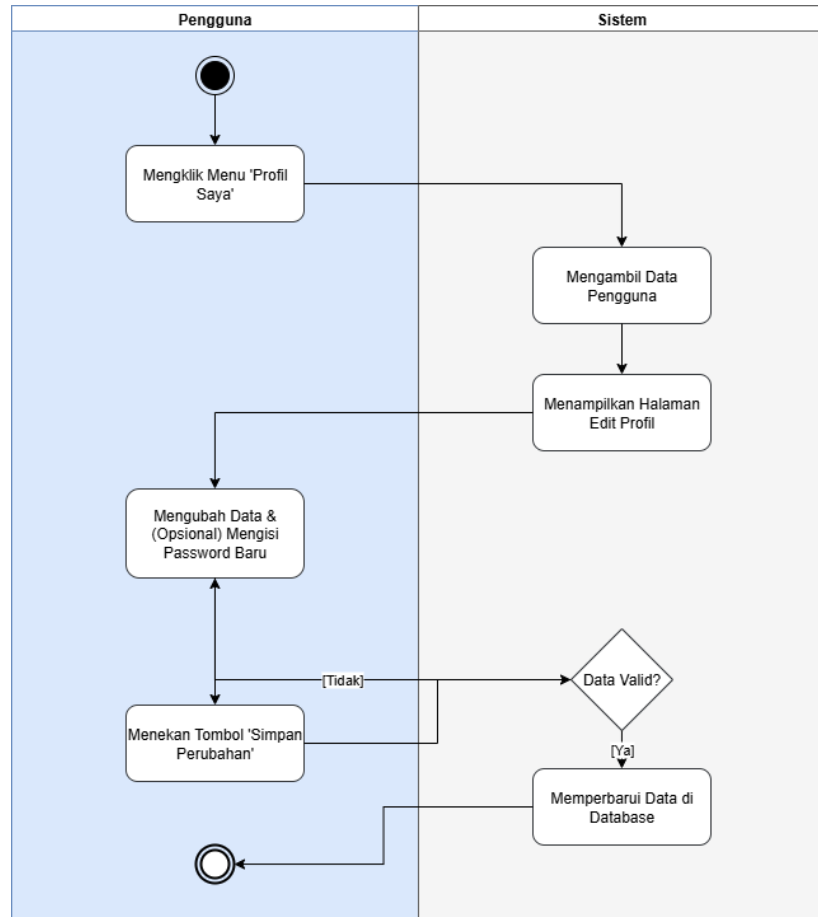
Gambar 4. 4 *Activity Diagram* Halaman

Diagram aktivitas ini menggambarkan alur kerja (workflow) yang dilakukan oleh seorang Super Administrator saat mengelola Halaman Statis. Halaman statis adalah konten

yang jarang berubah dan berfungsi untuk memberikan informasi fundamental tentang puskesmas, seperti profil, visi & misi, atau kontak. Fitur ini memungkinkan admin untuk membuat halaman baru dari nol, memperbarui konten yang sudah ada, dan menghapus halaman yang tidak lagi relevan. Alur kerja ini merupakan proses CRUD (Create, Read, Update, Delete) yang standar dan terstruktur:

Admin memulai dengan mengakses halaman manajemen, di mana sistem akan menampilkan daftar semua halaman yang telah dibuat. Kemudian Admin dapat membuat halaman baru dengan mengisi judul dan konten isi menggunakan editor teks yang kaya fitur, serta mengatur statusnya (apakah akan langsung dipublikasikan atau disimpan sebagai draf). Selain itu Admin dapat mengubah semua detail dari halaman yang sudah ada terakhir Admin dapat menghapus halaman yang tidak lagi diperlukan dari sistem. Setiap data yang dikirim akan divalidasi oleh sistem sebelum disimpan, dan setelah berhasil, admin akan selalu dikembalikan ke halaman daftar utama untuk melihat perubahan yang telah diterapkan, memastikan alur kerja yang efisien dan terkontrol.

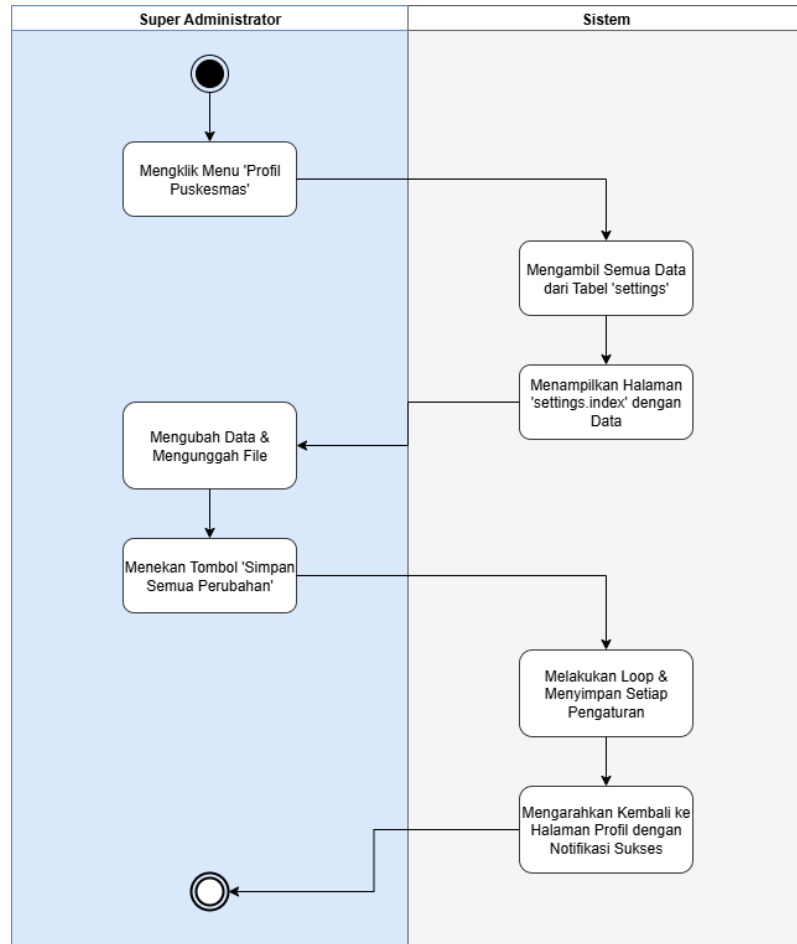
d) *Activity Diagram: Mengelola Profil Pengguna*



Gambar 4.5 *Activity Diagram* Mengelola Profil Pengguna

Diagram ini menggambarkan alur kerja langkah demi langkah saat seorang pengguna (baik itu admin, staf, maupun masyarakat) mengubah data profil mereka sendiri.

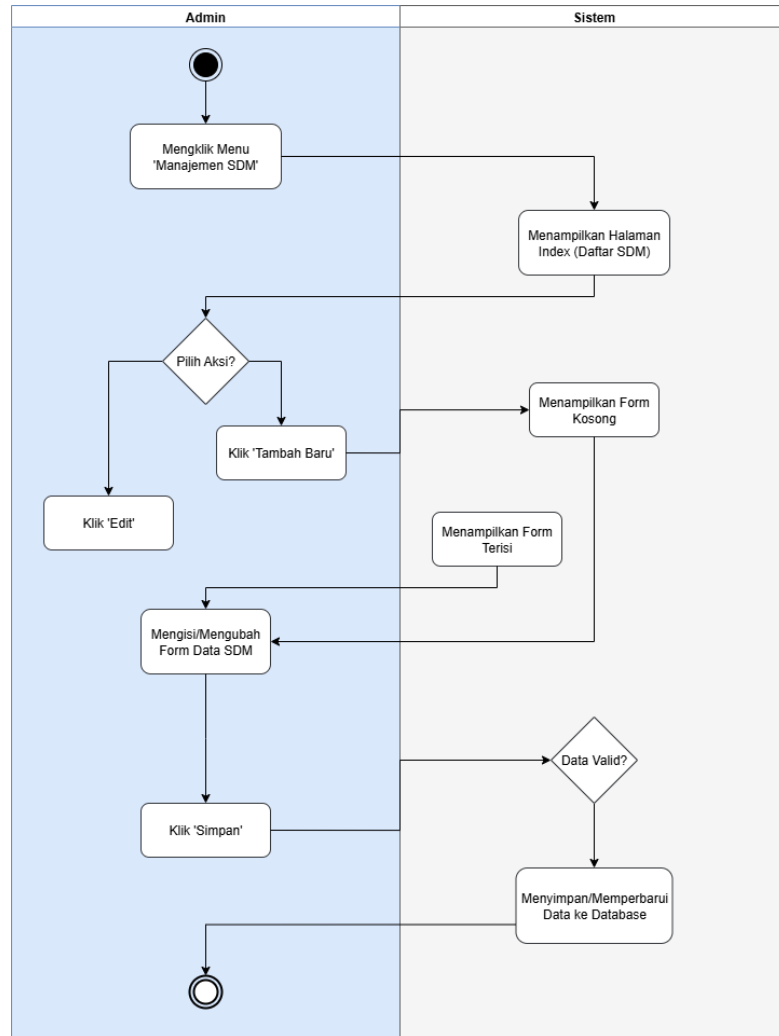
e) *Activity Diagram* Profil Puskesmas



Gambar 4. 6 *Activity Diagram* Profil Puskesmas

Diagram ini menggambarkan alur kerja langkah demi langkah saat seorang Super Administrator mengakses dan memperbarui informasi institusional Puskesmas, seperti nama, alamat, visi, misi, dan logo. Alur ini hanya terdiri dari dua aksi utama: menampilkan data yang ada (index) dan menyimpannya (update).

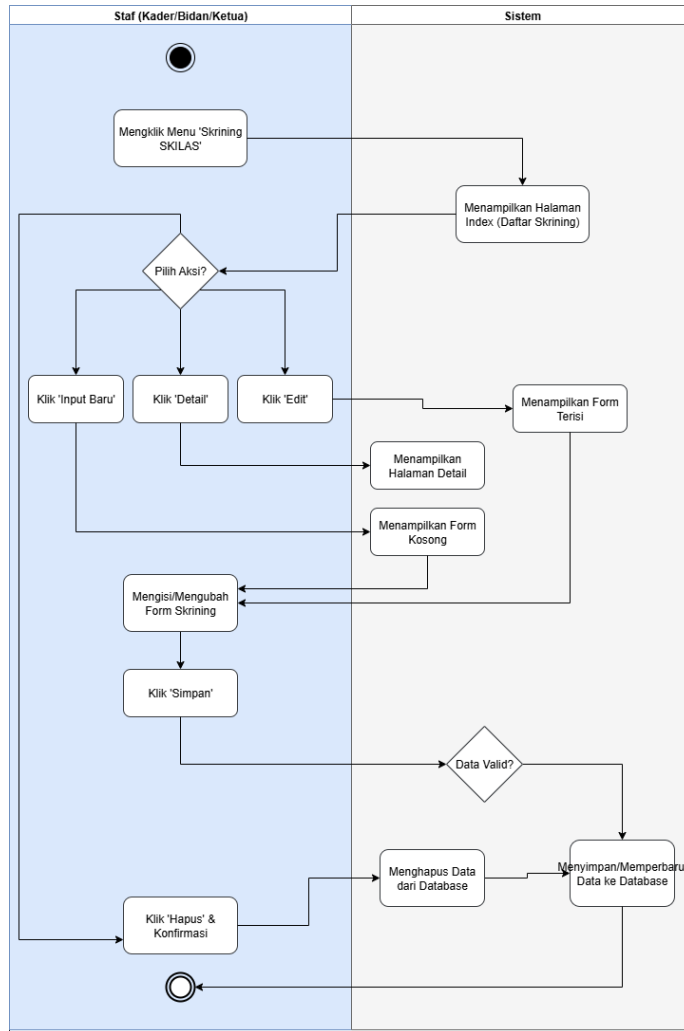
f) *Activity Diagram* Manajemen Pengguna



Gambar 4. 7 *Activity Diagram* Manajemen SDM

Diagram ini menggambarkan alur kerja lengkap yang dilakukan oleh seorang Admin (bisa Super Administrator atau Bidan) saat mengelola data Sumber Daya Manusia (SDM) atau Tenaga Kesehatan di dalam sistem. Diagram ini mencakup proses dari melihat daftar, menambah, memperbarui, hingga menghapus data.

g) Activiy Diagram Skirining SKILAS



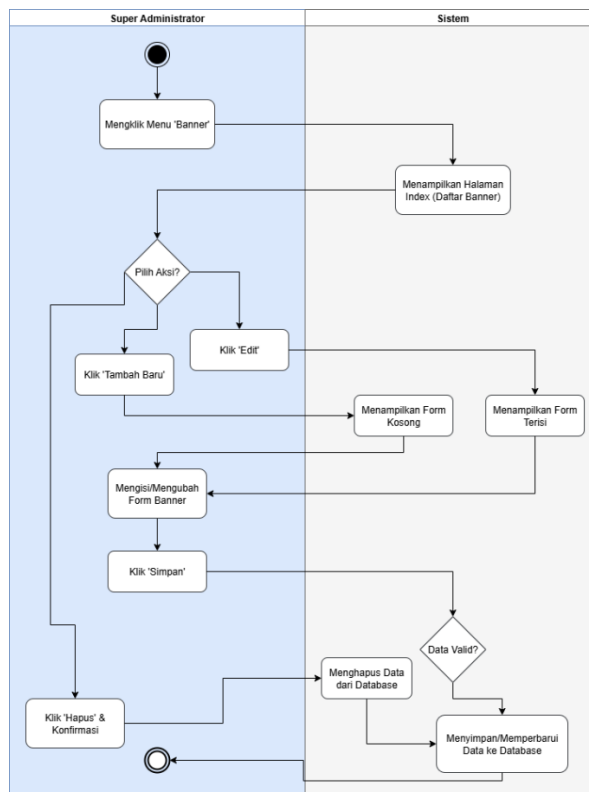
Gambar 4. 8 *Activity Diagram* Skirining SKILAS

Diagram aktivitas ini menggambarkan alur kerja (workflow) lengkap yang terjadi saat seorang staf (bisa Bidan, Ketua Posyandu, atau Anggota Posyandu) melakukan manajemen data pada modul Skirining SKILAS. Modul ini merupakan fitur inti untuk input data kesehatan lansia, yang mencakup proses melihat daftar data (index), menambah data baru (create & store), melihat rincian data (show), memperbarui data (edit & update), hingga menghapus data (destroy).

Alur kerja ini dimulai ketika staf membuka halaman utama modul, di mana mereka disajikan dengan daftar semua data skrining yang sudah ada. Dari sana, mereka memiliki beberapa pilihan aksi:

- Menambah Data: Staf akan diarahkan ke sebuah form komprehensif untuk mengisi semua detail pasien dan hasil skrining. Setelah data disimpan, sistem akan melakukan validasi sebelum menyimpannya ke database dan kembali ke halaman daftar.
- Melihat Detail: Staf dapat melihat rincian lengkap dari satu data skrining dalam format read-only.
- Mengedit Data: Alur ini mirip dengan menambah data, namun form akan terisi otomatis dengan data lama yang siap untuk diubah.
- Menghapus Data: Staf dapat menghapus data setelah melalui sebuah langkah konfirmasi untuk mencegah kesalahan.
- Setiap aksi yang berhasil akan selalu mengembalikan staf ke halaman daftar utama dengan sebuah notifikasi sukses, menciptakan sebuah siklus kerja yang efisien.

h) *Activity Diagram* Manajemen Layanan

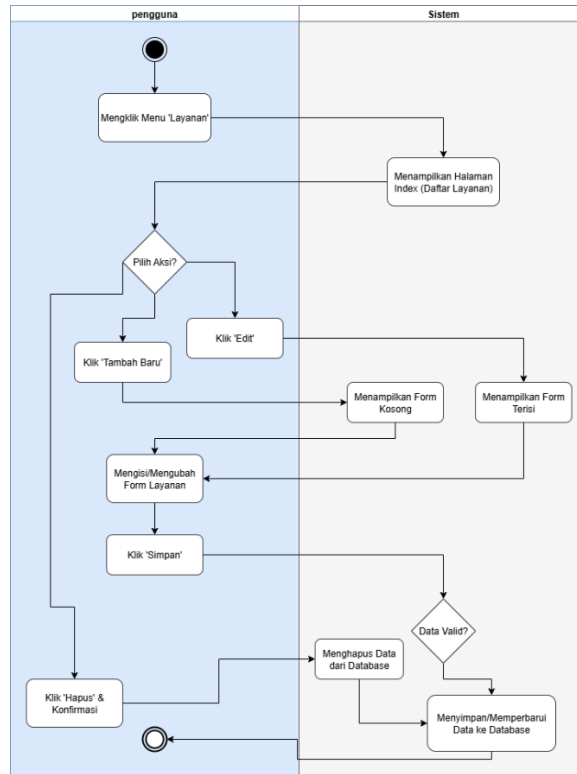


Gambar 4.9 *Activity Diagram* Manajemen Layanan

Diagram aktivitas ini menggambarkan alur kerja (workflow) yang dilakukan oleh seorang Super Administrator saat mengelola Layanan, yaitu gambar-gambar slider interaktif yang ditampilkan di bagian bawah sesudah banner halaman depan website. Fitur ini memungkinkan admin untuk menambah, memperbarui, mengatur urutan, dan menghapus gambar Layanan untuk menjaga tampilan beranda tetap segar dan informatif. Alur kerja ini merupakan proses CRUD (Create, Read, Update, Delete) standar:

Admin memulai dengan mengakses halaman manajemen Layanan, di mana sistem akan menampilkan semua Layanan yang ada beserta urutannya. Kemudian Admin dapat mengunggah gambar baru, menambahkan judul, tautan (link) jika diperlukan, serta menentukan urutan tampilnya. Tidak hanya itu Mengedit Layanan Admin dapat mengubah semua detail dari Layanan yang sudah ada, termasuk mengganti gambarnya terakhir Admin dapat menghapus Layanan yang sudah tidak relevan. Setiap data yang dikirim akan divalidasi oleh sistem sebelum disimpan, dan setelah berhasil, admin akan selalu dikembalikan ke halaman daftar utama untuk melihat perubahan yang telah diterapkan.

i) *Activity Diagram Layanan*



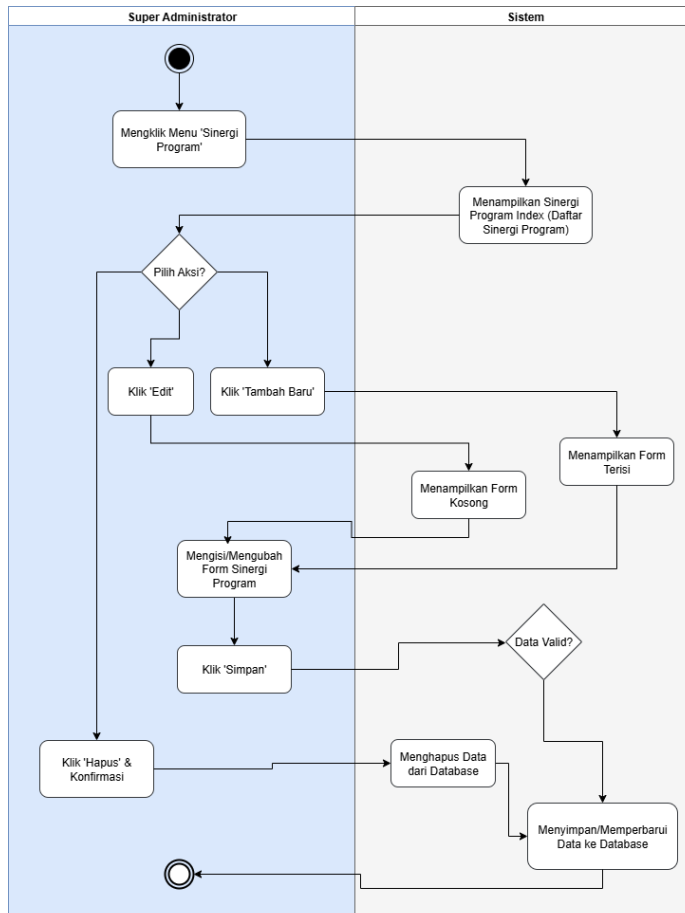
Gambar 4. 10 *Activity Diagram Layanan*

Diagram aktivitas ini menggambarkan alur kerja (workflow) yang dilakukan oleh seorang Super Administrator saat mengelola Layanan, yaitu gambar-gambar slider interaktif yang ditampilkan di bagian bawah sesudah banner halaman depan website. Fitur ini memungkinkan admin untuk menambah, memperbarui, mengatur urutan, dan menghapus gambar Layanan untuk menjaga tampilan beranda tetap segar dan informatif. Alur kerja ini merupakan proses CRUD (Create, Read, Update, Delete) standar:

Admin memulai dengan mengakses halaman manajemen Layanan, di mana sistem akan menampilkan semua Layanan yang ada beserta urutannya. Kemudian Admin dapat mengunggah gambar baru, menambahkan judul, tautan (link) jika diperlukan, serta menentukan urutan tampilnya. Tidak hanya itu Mengedit Layanan Admin dapat mengubah semua detail dari Layanan yang sudah ada, termasuk mengganti gambarnya terakhir Admin dapat menghapus Layanan yang sudah tidak relevan. Setiap data yang dikirim akan

divalidasi oleh sistem sebelum disimpan, dan setelah berhasil, admin akan selalu dikembalikan ke halaman daftar utama untuk melihat perubahan yang telah diterapkan.

j) *Activity Diagram Sinergi-Program*



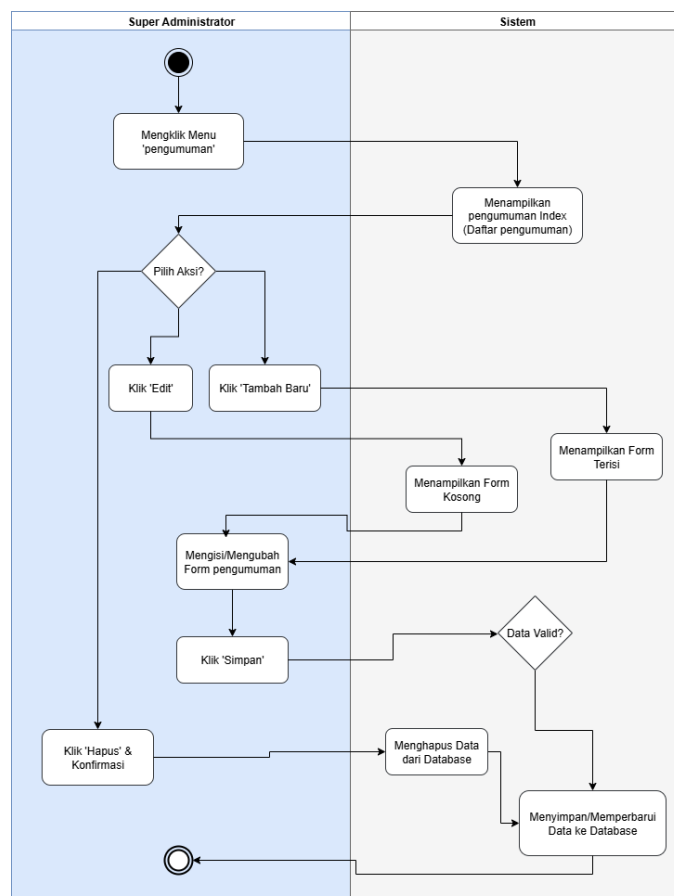
Gambar 4. 11 *Activity Diagram Sinergi Program*

Diagram aktivitas ini menggambarkan alur kerja (workflow) yang dilakukan oleh seorang Super Administrator saat mengelola Sinergi program, yaitu gambar-gambar slider interaktif yang ditampilkan di bagian atas halaman depan website. Fitur ini memungkinkan admin untuk menambah, memperbaiki, mengatur urutan, dan menghapus gambar sinergi program untuk menjaga tampilan beranda tetap segar dan informatif. Alur kerja ini merupakan proses CRUD (Create, Read, Update, Delete) standar:

Admin memulai dengan mengakses halaman manajemen sinergi program, di mana sistem akan menampilkan semua sinergi

program yang ada beserta urutannya. Kemudian Admin dapat mengunggah gambar baru, menambahkan judul, deskripsi singkat, dan tautan (link) jika diperlukan, serta menentukan urutan tampilnya. Tidak hanya itu Mengedit Sinergi program Admin dapat mengubah semua detail dari sinergi program yang sudah ada, termasuk mengganti gambarnya terakhir Admin dapat menghapus sinergi program yang sudah tidak relevan. Setiap data yang dikirim akan divalidasi oleh sistem sebelum disimpan, dan setelah berhasil, admin akan selalu dikembalikan ke halaman daftar utama untuk melihat perubahan yang telah diterapkan.

k) *Activity Diagram Pengumuman*



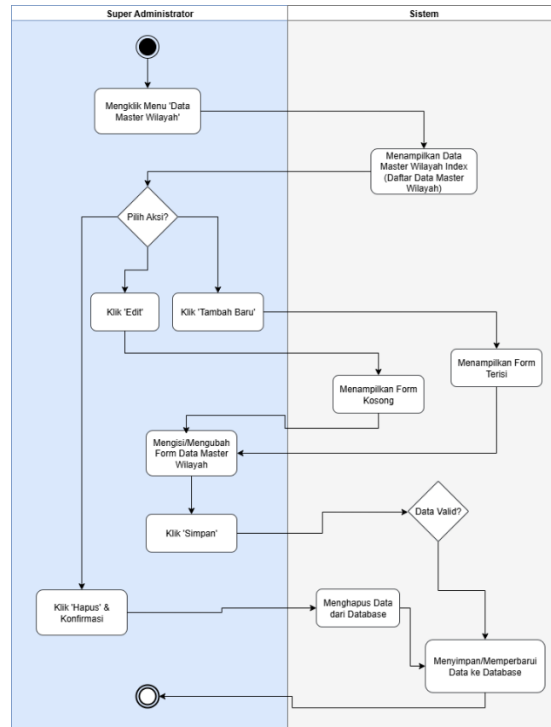
Gambar 4. 12 *Activity Diagram Pengumuman*

Diagram aktivitas ini menggambarkan alur kerja (workflow) yang dilakukan oleh seorang Super Administrator saat mengelola Pengumuman. Fitur ini berfungsi sebagai papan informasi digital,

memungkinkan admin untuk membuat, memperbarui, dan menghapus pengumuman yang akan ditampilkan kepada pengguna, baik di halaman depan maupun di dashboard masyarakat. Alur kerja ini merupakan proses CRUD (Create, Read, Update, Delete) yang terstruktur prosesnya Admin memulai dengan mengakses halaman manajemen pengumuman, di mana sistem akan menampilkan semua pengumuman yang ada. Kemudian Menambah Pengumuman Admin dapat membuat pengumuman baru dengan mengisi judul dan konten isi menggunakan editor teks, serta mengatur properti penting lainnya seperti tipe tampilan, status (draf atau publikasi), dan rentang tanggal tayang. Admin dapat mengubah semua detail dari pengumuman yang sudah ada. Admin dapat menghapus pengumuman yang sudah tidak relevan dari sistem.

Setiap data yang dikirim akan divalidasi oleh sistem sebelum disimpan, dan setelah berhasil, admin akan selalu dikembalikan ke halaman daftar utama untuk melihat perubahan yang telah diterapkan, memastikan pengelolaan informasi berjalan dengan efisien.

1) Activity Diagram Master Data

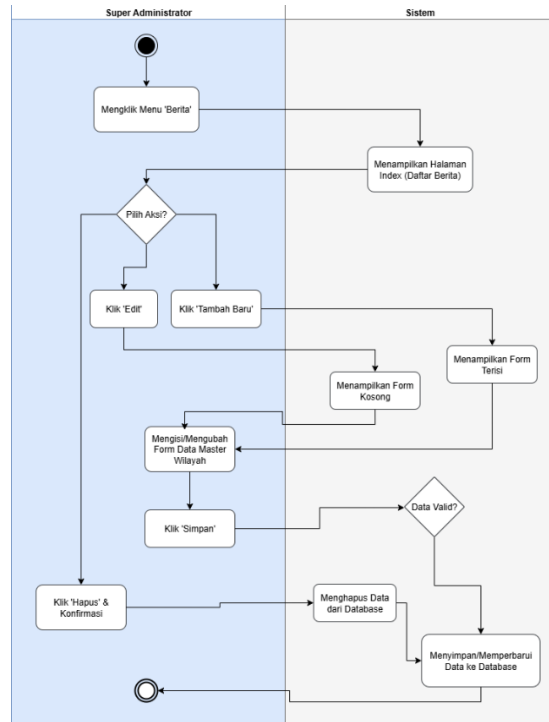


Gambar 4. 13 Activity Diagram Master Wilayah

Diagram aktivitas berikut menggambarkan alur kerja standar yang dilakukan oleh seorang Admin (Super Administrator atau Bidan) saat mengelola Data Master Wilayah. Alur kerja ini berlaku sama untuk beberapa modul, di antaranya adalah Manajemen Desa, Manajemen Dusun, Manajemen Pustu, dan Manajemen Posyandu. Fitur-fitur ini berfungsi sebagai fondasi data geografis dan operasional sistem. Alur kerja ini merupakan proses CRUD (Create, Read, Update, Delete) yang mana Admin memulai dengan mengakses halaman manajemen data master, di mana sistem akan menampilkan semua data yang relevan kemudian Admin dapat membuat entri data baru dengan mengisi form yang berisi atribut-atribut yang diperlukan, Admin dapat mengubah detail dari data yang sudah ada serta Admin dapat menghapus data yang sudah tidak relevan dari sistem.

Setiap operasi akan divalidasi oleh sistem sebelum disimpan, dan setelah berhasil, admin akan selalu dikembalikan ke halaman daftar utama untuk melihat perubahan yang telah diterapkan.

m) *Activity Diagram* Berita

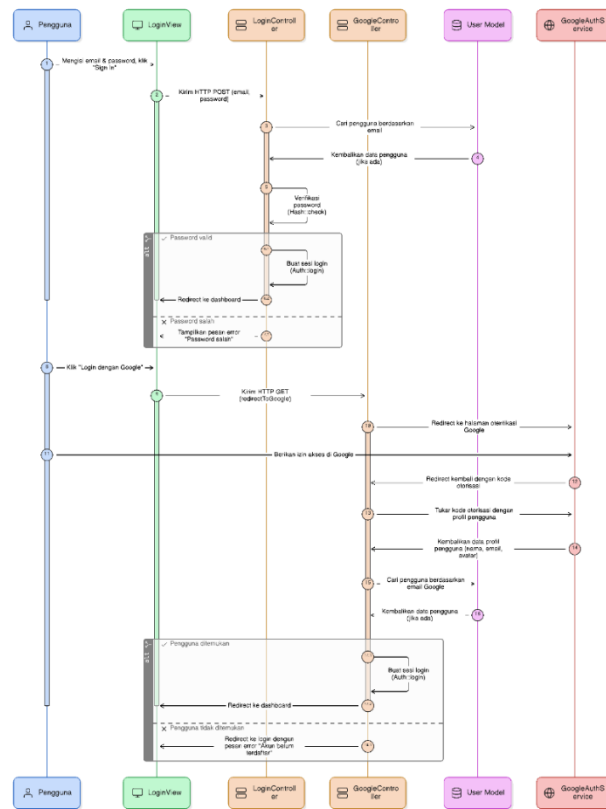


Gambar 4. 14 *Activity Diagram* Berita

2) *Sequence Diagram*

Sequence Diagram menggambarkan urutan "percakapan" atau pesan antar komponen sistem untuk menyelesaikan satu tugas.

a) *Sequence Diagram* Login/Register



Gambar 4. 15 *Sequence Diagram* Login dan Register

Diagram sekuens (urutan) ini menggambarkan interaksi teknis antar komponen sistem saat seorang pengguna mencoba untuk masuk (login) ke dalam aplikasi. Alur ini mencakup dua skenario utama yang disediakan oleh sistem: Login Manual menggunakan email dan password, serta Login dengan Google yang memanfaatkan otentikasi eksternal. Diagram ini menunjukkan bagaimana View (antarmuka pengguna) berkomunikasi dengan Controller yang berbeda untuk setiap metode login, dan bagaimana Controller tersebut kemudian berinteraksi dengan Model dan layanan eksternal untuk memverifikasi identitas pengguna sebelum memberikan akses ke dashboard. Objek/Komponen yang Terlibat:

- Pengguna yaitu Aktor yang memulai interaksi.
- LoginView yaitu Representasi dari halaman login (auth/login.blade.php).

- LoginController yaitu Controller yang menangani logika untuk login manual.
- GoogleController yaitu Controller yang menangani logika untuk login via Google.
- User (Model) yaitu Representasi dari tabel users yang berinteraksi dengan database.
- GoogleAuthService yaitu Representasi dari layanan otentikasi eksternal milik Google.

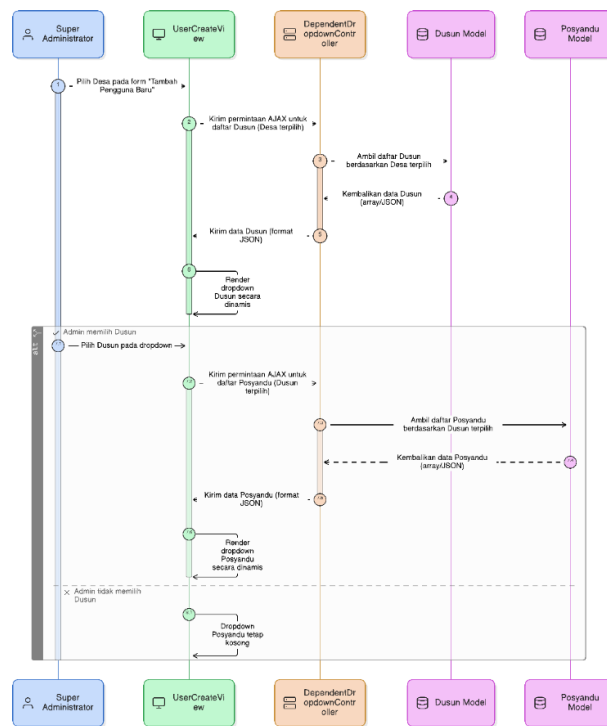
Tabel 4. 1 Skenario dari Squence Diagram

Skenario 1 melakukan Login Manual (Email & Password)		
Pengguna	LoginView	Mengisi email & password, lalu menekan tombol "Sign In".
LoginView	LoginController	Mengirimkan HTTP POST Request ke method store dengan data email dan password.
LoginController	User (Model)	Meminta untuk mencari pengguna berdasarkan alamat email yang diberikan.
User (Model)	LoginController	Mengembalikan data pengguna yang ditemukan (jika ada).
LoginController	LoginController	Melakukan verifikasi internal: Hash::check(password_input, password_database).
LoginController	LoginController	Jika verifikasi berhasil, membuat sesi login: Auth::login(user).
LoginController	LoginView	Mengirimkan respons Redirect ke halaman dashboard yang sesuai.
Skenario 2: Login dengan Google		
Pengguna	LoginView	Menekan tombol "Login dengan Google".

LoginView	GoogleController	Mengirimkan HTTP GET Request ke method redirectToGoogle.
GoogleController	GoogleAuthService	Mengarahkan (redirect) browser pengguna ke halaman otentikasi Google.
Pengguna	GoogleAuthService	Memberikan izin akses di halaman Google.
GoogleAuthService	GoogleController	Mengarahkan kembali pengguna ke aplikasi kita (method handleGoogleCallback) sambil membawa "tiket" atau kode otorisasi.
GoogleController	User (Model)	Meminta untuk mencari pengguna di database kita berdasarkan alamat email yang diterima dari Google.
GoogleAuthService	GoogleController	Mengembalikan data profil pengguna.
GoogleController	GoogleAuthService	Menukarkan "tiket" tersebut dengan data profil pengguna (nama, email, avatar).
User (Model)	GoogleController	Mengembalikan data pengguna yang ditemukan (jika ada).
GoogleController	GoogleController	Melakukan keputusan internal: "Apakah pengguna ini ditemukan di database kita?".
GoogleController	GoogleController	Jika ditemukan, membuat sesi login: Auth::login(user).
GoogleController	LoginView	Mengirimkan respons Redirect ke halaman dashboard yang sesuai.

GoogleController	LoginView	[Jika Tidak Ditemukan] Mengirimkan respons Redirect kembali ke halaman login dengan pesan error "Akun belum terdaftar".
------------------	-----------	--

b) *Sequence Diagram Wilayah*



Gambar 4. 16 *Sequence Diagram Wilayah*

Diagram sekuens (urutan) ini menggambarkan interaksi teknis antar komponen sistem saat seorang Super Administrator menggunakan fitur dropdown dinamis pada form "Tambah Pengguna Baru". Alur ini menunjukkan bagaimana permintaan asynchronous (AJAX) diproses dari antarmuka pengguna (View) hingga ke database (melalui Model) dan kembali lagi, tanpa perlu me-refresh halaman. Diagram ini dibagi menjadi dua fase utama yang berantai:

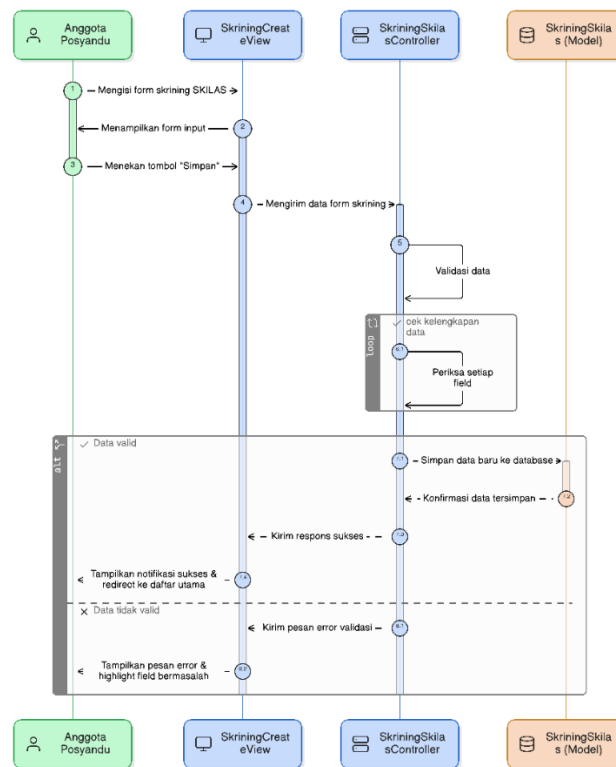
Fase 1 adalah Memilih Desa dimulai saat admin memilih sebuah Desa. View akan mengirimkan permintaan ke Controller

untuk mendapatkan daftar Dusun yang sesuai. Controller kemudian berinteraksi dengan Model untuk mengambil data dari database dan mengirimkannya kembali ke View dalam format JSON.

Fase 2 adalah Memilih Dusun dimulai setelah dropdown Dusun terisi. Saat admin memilih sebuah Dusun, proses yang sama terulang kembali. View mengirim permintaan baru untuk mendapatkan daftar Posyandu yang sesuai, dan hasilnya digunakan untuk mengisi dropdown Posyandu. Diagram ini secara efektif menunjukkan implementasi arsitektur Model-View-Controller (MVC) dalam menangani interaksi pengguna yang dinamis. Objek/Komponen yang Terlibat:

- Super Administrator yaitu Aktor yang memulai interaksi.
- UserCreateView yaitu Representasi dari antarmuka pengguna file users/create.blade.php.
- DependentDropdownController yaitu Controller yang bertugas menerima permintaan AJAX dan mengatur logika bisnis.
- Dusun Model yaitu Representasi dari tabel dusuns yang berinteraksi dengan database.
- Posyandu Model yaitu Representasi dari tabel posyandus

c) *Sequence Diagram* Skrining SKILAS



Gambar 4. 17 *Sequence Diagram* Skrining SKILAS

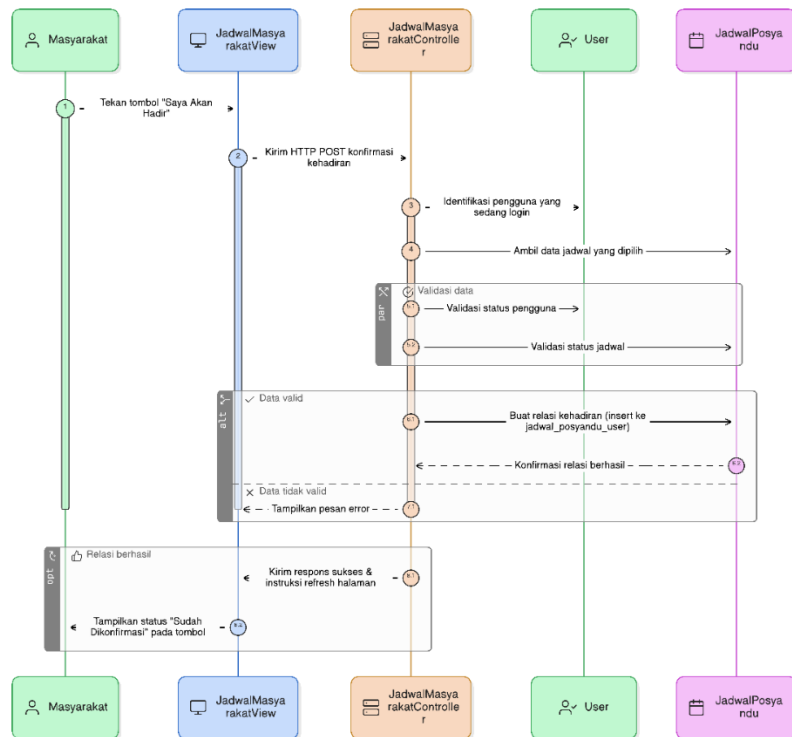
Diagram sekuens (urutan) ini menggambarkan interaksi teknis antar komponen sistem saat seorang Anggota Posyandu (Kader) melakukan tugas utamanya: menginput data baru hasil Skrining SKILAS. Alur ini menunjukkan proses Create data yang fundamental dalam arsitektur Model-View-Controller (MVC).

"Percakapan" dimulai saat Kader mengisi form skrining di antarmuka pengguna (View). Setelah menekan tombol "Simpan", View akan mengirimkan semua data dari form ke Controller. Controller kemudian bertindak sebagai "manajer", pertama dengan memvalidasi semua data yang masuk untuk memastikan kelengkapannya. Jika data valid, Controller akan memerintahkan Model untuk membuat catatan baru dan menyimpannya secara permanen ke dalam database. Terakhir, sistem akan memberikan respons sukses dan mengarahkan Kader

kembali ke halaman daftar utama. Objek/Komponen yang Terlibat:

- Anggota Posyandu: Aktor yang memulai interaksi.
- SkringingCreateView: Representasi dari antarmuka pengguna (file skringing-skilas/create.blade.php).
- SkringingSkilasController: Controller yang bertugas menerima permintaan dan mengatur logika bisnis.
- SkringingSkilas (Model): Representasi dari tabel skringing_skilas yang berinteraksi langsung dengan database.

d) Sequence Diagram Konfirmasi Kehadiran Jadwal



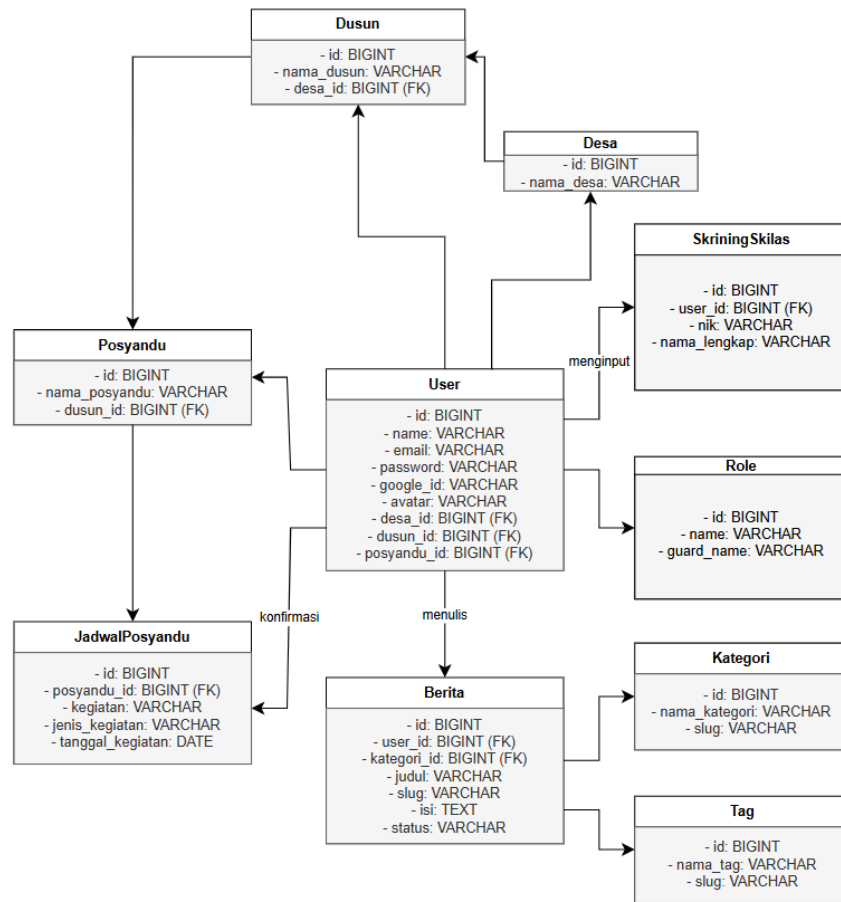
Gambar 4. 18 *Sequence Diagram* Konfirmasi Kehadiran Jadwal

Diagram sekuens (urutan) ini menggambarkan interaksi teknis antar komponen sistem saat seorang pengguna Masyarakat melakukan aksi konfirmasi kehadiran untuk sebuah Jadwal Posyandu. Alur ini menunjukkan bagaimana sebuah aksi sederhana dari antarmuka pengguna (View) diproses oleh Controller untuk memperbarui relasi data many-to-many di dalam database. Interaksi dimulai saat pengguna menekan tombol "Saya

Akan Hadir" di halaman jadwal. Aksi ini mengirimkan sebuah permintaan HTTP POST ke Controller. Controller kemudian mengidentifikasi pengguna yang sedang login dan jadwal yang dipilih. Setelah itu, ia memerintahkan Model User dan JadwalPosyandu untuk membuat "ikatan" atau catatan baru di dalam tabel penghubung (`jadwal_posyandu_user`). Setelah berhasil, sistem akan memberikan respons sukses dan mengarahkan pengguna kembali ke halaman jadwal, di mana tombol yang tadi diklik kini akan berubah status menjadi "Sudah Dikonfirmasi". Objek/Komponen yang Terlibat:

- Masyarakat yaitu Aktor yang memulai interaksi.
- JadwalMasyarakatView yaitu Representasi dari antarmuka pengguna (file `masyarakat/jadwal/index.blade.php`).
- JadwalMasyarakatController yaitu Controller yang bertugas menerima permintaan konfirmasi.
- User (Model) yaitu Representasi dari pengguna yang sedang login.
- JadwalPosyandu (Model) yaitu Representasi dari jadwal yang dikonfirmasi.

3) Class Diagram



Gambar 4. 19 *Class Diagram*

Diagram Kelas (Class Diagram) ini berfungsi sebagai cetak biru arsitektur untuk keseluruhan Sistem Informasi Puskesmas. Diagram ini memvisualisasikan struktur fundamental dari database dan model aplikasi, menunjukkan setiap entitas utama sebagai sebuah "kelas", lengkap dengan atribut (kolom tabel) dan hubungan antar kelas tersebut. Diagram ini menyoroti beberapa hubungan kunci:

- b) Hubungan Pengguna dan Peran yaitu Menunjukkan hubungan many-to-many antara User dan Role yang dikelola oleh package Spatie, yang memungkinkan sistem hak akses yang fleksibel.
- c) Hubungan Wilayah Kerja yaitu Menggambarkan struktur hirarkis dari Desa yang memiliki banyak Dusun, dan Dusun yang memiliki banyak Posyandu.

- d) Hubungan Konten yaitu Menunjukkan bagaimana Berita dibuat oleh seorang User dan dikelompokkan berdasarkan Kategori serta Tag.
- e) Hubungan Interaktif yaitu Memvisualisasikan hubungan many-to-many antara User (Masyarakat) dan JadwalPosyandu untuk mencatat data konfirmasi kehadiran.

Secara keseluruhan, diagram ini memberikan gambaran teknis yang komprehensif tentang bagaimana data diorganisir dan saling terhubung di dalam sistem.

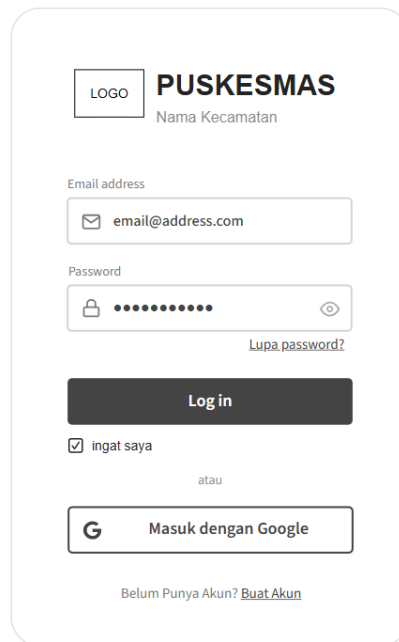
b. Perancangan Antarmuka Pengguna

Pada tahap ini, perancangan antarmuka pengguna dilakukan untuk merealisasikan hasil rancangan *Wireframe* yang telah disusun pada tahap sebelumnya. Perancangan dilakukan dengan tujuan untuk menghasilkan tampilan sistem informasi Puskesmas Panjalu yang responsif, mudah digunakan, serta mampu memberikan pengalaman pengguna (user experience) yang baik.

Hasil perancangan antarmuka dituangkan ke dalam bentuk halaman-halaman utama sistem, meliputi halaman beranda, halaman login, halaman dashboard, serta halaman-halaman modul seperti pengelolaan data skrining SKILAS, manajemen pengguna, dan publikasi informasi. Setiap halaman dirancang dengan memperhatikan tata letak yang konsisten, penggunaan elemen navigasi yang jelas, serta penyusunan komponen yang memudahkan pengguna dalam menemukan informasi atau melakukan tindakan.

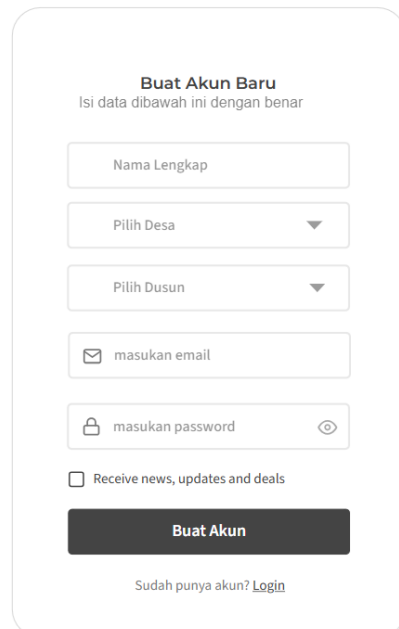
Dalam implementasinya, perancangan ini telah menghasilkan tampilan awal sistem yang sesuai dengan rancangan *Wireframe* sebelumnya. Elemen-elemen seperti menu, tombol aksi, form input, dan tabel data ditempatkan secara terstruktur, sehingga alur interaksi pengguna dapat berjalan lancar. Selain itu, pemilihan warna dan tipografi disesuaikan dengan identitas visual Puskesmas Panjalu, sehingga memberikan kesan profesional dan mudah dikenali oleh pengguna.

1) Wireframe Login dan Register



The wireframe shows a login interface for 'PUSKESMAS'. At the top left is a 'LOGO' placeholder. To its right is the title 'PUSKESMAS' and the subtitle 'Nama Kecamatan'. Below this is an 'Email address' field containing 'email@address.com'. Underneath is a 'Password' field with a lock icon, a series of dots, and an eye icon to toggle visibility. A link 'Lupa password?' is positioned below the password field. A dark 'Log in' button is centered below the password field. Below the button is a checked checkbox labeled 'ingat saya'. The word 'atau' is centered below the checkbox. A 'Masuk dengan Google' button with the Google 'G' logo is centered below 'atau'. At the bottom, the text 'Belum Punya Akun? [Buat Akun](#)' is centered.

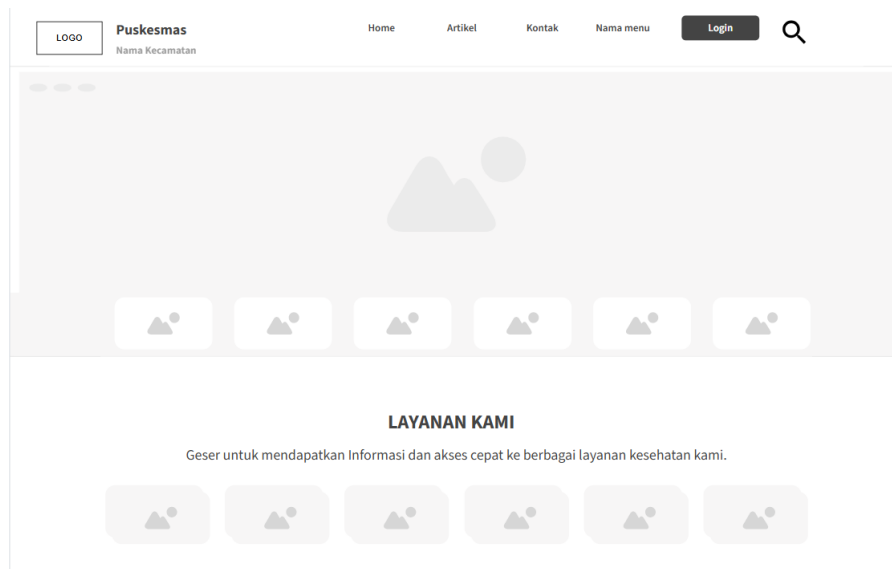
Gambar 4. 20 Wireframe Login



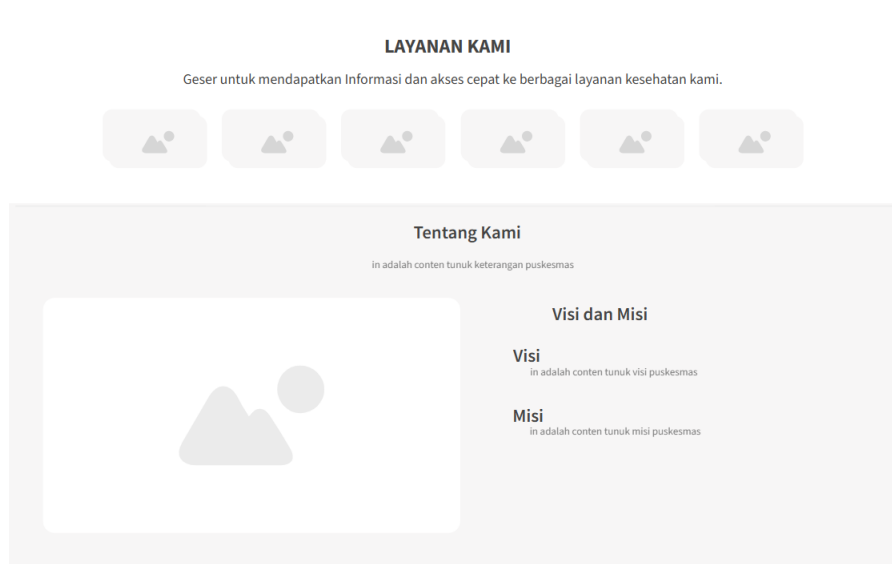
The wireframe shows a registration interface titled 'Buat Akun Baru' with the instruction 'Isi data dibawah ini dengan benar'. The form contains several fields: 'Nama Lengkap', 'Pilih Desa' (a dropdown menu), 'Pilih Dusun' (a dropdown menu), 'masukan email' (with an envelope icon), and 'masukan password' (with a lock icon and an eye icon). Below these fields is an unchecked checkbox labeled 'Receive news, updates and deals'. A dark 'Buat Akun' button is centered below the checkbox. At the bottom, the text 'Sudah punya akun? [Login](#)' is centered.

Gambar 4. 21 Wireframe Legister

2) Wireframe Tampilan Depan/Landing Page



Gambar 4. 22 Wireframe Tampilan Depan Landing Page

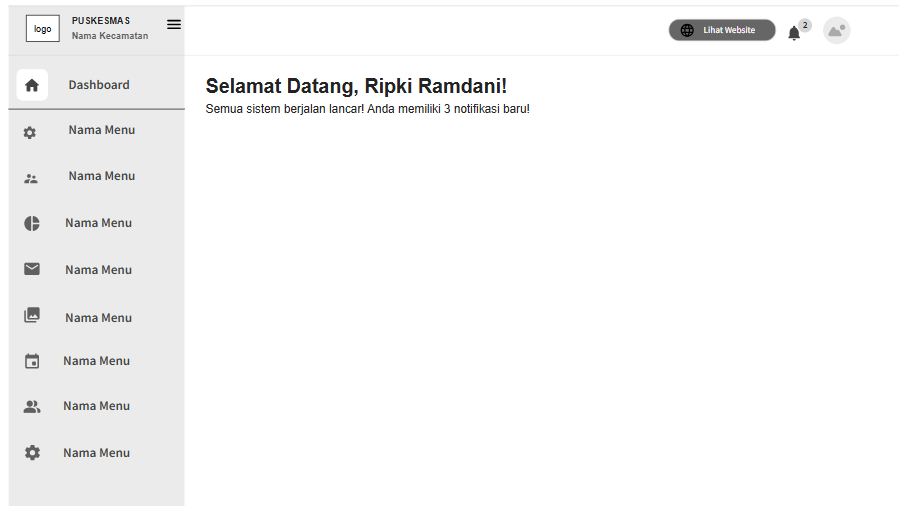


Gambar 4. 23 Wireframe Layanan dan Tentang Kami

Wireframe tampilan depan dirancang menggunakan template Newspaper yang kemudian dikustomisasi agar sesuai dengan identitas visual dan kebutuhan informasi Puskesmas Panjalu. Tampilan ini terdiri dari header berisi logo, nama instansi, dan menu navigasi; main content yang menampilkan berita terbaru, pengumuman, dan agenda, jadwal posyandu dan informasi tambahan; serta footer berisi kontak dan alamat. Penyesuaian

dilakukan pada skema warna, tata letak, dan elemen grafis agar tampil profesional dan mudah diakses pengguna.

3) Dashboard Sistem informasi



Gambar 4. 24 *Wirefame* Dahnboard Superadmin



Gambar 4. 25 *Wirefame* Dashboard Masyarakat

Wireframe dashboard dirancang menggunakan template Skydash yang kemudian dikustomisasi sesuai kebutuhan sistem informasi Puskesmas Panjalu. Dashboard berfungsi sebagai pusat kendali bagi pengguna sesuai peran masing-masing, seperti superadmin, bidan, atau ketua posyandu, serta masyarakat. Tampilan terdiri dari sidebar navigasi untuk mengakses modul, header dengan informasi akun dan notifikasi, serta main content yang menampilkan ringkasan data, grafik, dan tabel sesuai modul

yang diakses. Penyesuaian dilakukan pada menu, ikon, dan warna agar konsisten dengan identitas visual puskesmas.

4) Kluster

The wireframe shows a web application interface for adding a new cluster. On the left is a sidebar with a logo and a menu containing 'Dashboard' and several 'Nama Menu' items. The main content area is titled 'Tambah Kluster Baru' and contains the following form fields: 'Judul Menu' (text input), 'Menu Induk (Parent)' (dropdown menu with 'Pilih' text), 'Hubungkan ke halaman' (dropdown menu with 'Pilih' text), 'link Manual/URL' (text input), and 'Urutan' (dropdown menu with 'Pilih' text). At the bottom of the form are two buttons: 'simpan' (dark grey) and 'Batal' (light grey).

Gambar 4. 26 Wireframe Kluster

Kluster berfungsi untuk membuat dan mengatur kategori utama konten yang akan muncul pada menu navigasi (*navbar*) website. Kategori ini dapat berupa layanan kesehatan, berita, galeri, atau informasi lain sesuai kebutuhan puskesmas. Pengaturan kluster memudahkan pengelompokan informasi agar pengunjung dapat menemukan konten yang mereka cari secara lebih terstruktur.

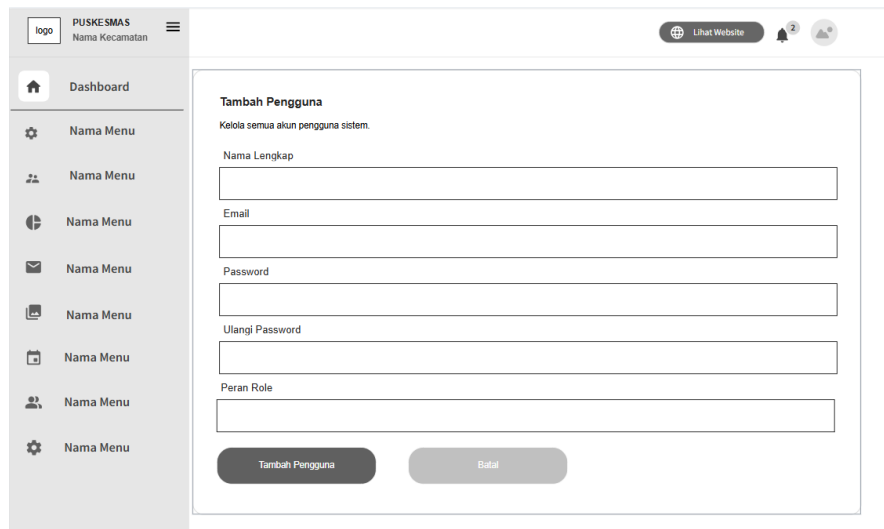
5) Halaman

The wireframe shows a web application interface for adding a new page. On the left is a sidebar with a logo and a menu containing 'Dashboard' and several 'Nama Menu' items. The main content area is titled 'Tambah Halaman Baru' and contains the following form fields: 'Judul Halaman' (text input) and 'Konten Halaman' (large text area). To the right of the main form are three separate panels: 'Pengaturan Publikasi' with a 'Status' dropdown menu (showing 'draft') and 'simpan'/'Batal' buttons; 'Induk Menu' with a 'Hubungkan Ke Menu' dropdown menu (showing 'Tidak Terhubung'); and 'Gambar Unggulan' with an 'Upload Gambar' dropdown menu (showing 'Upload').

Gambar 4. 27 Wirefame Tambah Halaman

Halaman digunakan untuk membuat, mengedit, dan menghapus konten halaman pada website puskesmas. Konten dapat berupa teks, gambar, atau kombinasi keduanya sesuai kebutuhan informasi yang akan disampaikan kepada masyarakat. Melalui modul ini, administrator dapat menambahkan halaman baru, seperti profil layanan, jadwal kegiatan, artikel kesehatan, atau pengumuman khusus, sehingga website selalu terbaru dan relevan.

6) Tambah Pengguna



The wireframe shows a web application interface for adding a user. The header includes a logo, the text 'PUSKESMAS Nama Kecamatan', and a 'Lihat Website' button. A sidebar menu on the left contains 'Dashboard' and several 'Nama Menu' items. The main content area is titled 'Tambah Pengguna' and contains the following fields: 'Nama Lengkap', 'Email', 'Password', 'Ulangi Password', and 'Peran Role'. At the bottom of the form are two buttons: 'Tambah Pengguna' and 'Batal'.

Gambar 4. 28 *Wireframe* Tambah Pengguna

Wireframe ini digunakan untuk menambahkan akun pengguna baru pada sistem informasi Puskesmas Panjalu. Data yang diinput meliputi nama pengguna, kata sandi, konfirmasi kata sandi, dan peran (role) pengguna. Setiap peran memiliki pengaturan tambahan yang berbeda. Pengaturan ini memastikan setiap pengguna memiliki akses sesuai wilayah dan tanggung jawabnya.

- a) Bidan, memilih nama desa
- b) Ketua Posyandu, memilih nama desa, dusun, dan posyandu.
- c) Anggota Posyandu, sama seperti ketua posyandu, yaitu menyematkan desa, dusun, dan posyandu.
- d) Masyarakat, menyematkan nama desa dan dusun.
- e) Superadmin dan Penulis, hanya menambahkan nama, kata sandi, dan peran tanpa pengaturan tambahan.

7) Tambah Sumber Daya Manusia

The wireframe shows a web interface for adding new health resources. On the left is a sidebar menu with 'Dashboard' at the top and several 'Nama Menu' items below. The main content area is titled 'Tambah Data SDM Baru' and includes a sub-header 'Kelola semua akun pengguna sistem.' Below this are several input fields: 'Nama Lengkap', 'NIP / NIK (Opsional)', 'Jabatan/Profesi', 'Spesialisasi', and 'Photo' with an 'Upload' button. At the bottom of the form are two buttons: 'Tambah Pengguna' and 'Batal'.

Gambar 4. 29 *Wireframe* Tambah Sumber Daya Manusia

Wireframe ini digunakan untuk menambahkan data tenaga kesehatan atau pegawai puskesmas. Informasi yang dimasukkan dapat mencakup nama pegawai, jabatan, unit kerja, serta informasi pendukung lain sesuai kebutuhan.

8) Jadwal Posyandu

The wireframe shows a web interface for adding a new health activity schedule. On the left is a sidebar menu with 'Dashboard' at the top and several 'Nama Menu' items below. The main content area is titled 'Tambah Jadwal Baru' and includes a sub-header 'Gambar Banner' with a dropdown menu showing 'pilih posyandu'. Below this are two 'Jenis Kegiatan' dropdown menus, also showing 'pilih posyandu'. There are also input fields for 'Tanggal Kegiatan' (08/12/2025), 'Waktu Mulai' (08/12/2025), and 'Waktu Selesai' (08/12/2025). A large text area for 'Keterangan' is at the bottom. At the bottom of the form are two buttons: 'Simpan' and 'Batal'.

Gambar 4. 30 *Wireframe* Tambah Jadwal Posyandu

Wireframe jadwal posyandu berfungsi untuk mengatur dan menyebarkan informasi jadwal pelayanan Posyandu di wilayah kerja Puskesmas Panjalu. Petugas dapat menginput tanggal, waktu, dan lokasi kegiatan secara rinci, termasuk memilih wilayah desa, dusun, dan posyandu terkait. Informasi ini membantu masyarakat mengetahui waktu layanan kesehatan seperti imunisasi, pemeriksaan ibu hamil, balita, maupun lansia. Tata letak modul memungkinkan penjadwalan dilakukan dengan cepat, dilengkapi pilihan kategori kegiatan agar jadwal dapat dikelompokkan sesuai jenis layanan. Data yang telah disimpan akan secara otomatis tampil di website resmi sehingga masyarakat dapat mengaksesnya kapan saja.

9) Pengumuman

The wireframe shows a dashboard interface for adding a new announcement. On the left is a sidebar with a 'Dashboard' link and several 'Nama Menu' items. The main area is titled 'Tambah Pengumuman Baru'. It contains a form with a 'Judul Pengumuman' text input, an 'Isi Pengumuman' text area, and a 'Pengaturan' panel. The 'Pengaturan' panel includes a 'Tipe Tampilan' dropdown set to 'Inpo Biasa', 'Tanggal Mulai' and 'Tanggal Selesai' date pickers both set to '08/12/2025', and a 'Status' dropdown set to 'Draft'. A 'simpan' button is at the bottom of the settings. Below that is an optional 'Lampiran (Opsional)' section with an 'Upload File (PDF, Word, Gambar)' button and an 'Upload' input field.

Gambar 4. 31 *Wireframe* Tambah Pengumuman

Wireframe tambah pengumuman digunakan untuk membuat dan mempublikasikan informasi resmi Puskesmas Panjalu kepada masyarakat melalui website. Pada tampilan ini, pengguna dapat mengisi judul pengumuman dan isi pengumuman sebagai informasi utama. Di sisi kanan form tersedia pengaturan tambahan, seperti tipe tampilan yang menentukan format penyajian, tanggal mulai dan tanggal selesai yang mengatur periode penayangan, serta status pengumuman yang dapat diatur menjadi draft atau publish. Selain

itu, terdapat fitur unggah lampiran opsional dalam format PDF, Word, atau gambar, yang memungkinkan pengumuman dilengkapi dengan dokumen pendukung.

10) Berita

The wireframe shows a dashboard for 'PUSKESMAS Nama Kecamatan'. The main content area is titled 'Tambah Berita Baru'. It features a form with two main input areas: 'Judul Berita' (Title) and 'Isi Pengumuman' (Content). To the right of the main form, there are several configuration panels: 'Pengaturan Publikasi' (Publication Settings) with a 'Status' dropdown set to 'Draft', a 'Tanggal publikasi' (Publication Date) field set to '08/12/2025', and a 'simpan' (Save) button; 'Pilih Kategori' (Select Category) with an 'Upload' dropdown; 'Pilih Tags' (Select Tags) with an empty list; and 'Lampiran (Opsional)' (Optional Attachment) with an 'Upload File (PDF, Word, Gambar)' label and an 'Upload' dropdown. The dashboard includes a sidebar with 'Dashboard' and several 'Nama Menu' items, and a top navigation bar with 'Lihat Website', a notification bell, and a user profile icon.

Gambar 4. 32 *Wireframe* Tambah Berita

Wireframe tambah berita digunakan untuk mempublikasikan informasi, kegiatan, atau pengumuman penting Puskesmas Panjalu kepada masyarakat secara daring melalui website resmi. Pada bagian utama, pengguna dapat mengisi judul berita sebagai identitas informasi yang akan dipublikasikan, serta menulis isi pengumuman secara detail. Di sisi kanan form tersedia pengaturan publikasi yang mencakup pilihan status berita, seperti *draft* atau *publish*, serta tanggal publikasi untuk menentukan kapan berita tersebut ditayangkan. Selain itu, pengguna dapat memilih kategori berita untuk mengelompokkan informasi sesuai topik, menambahkan *tags* agar berita lebih mudah ditemukan melalui pencarian, dan mengunggah lampiran opsional dalam format PDF, Word, atau gambar sebagai pendukung informasi.

11) Manajemen Puskesmas

The wireframe shows a web application interface for managing a Puskesmas. It features a sidebar menu on the left with options: Dashboard, Nama Menu (repeated 8 times), and a gear icon. The main content area is divided into three sections: 1. 'Informasi Umum' (General Information) with input fields for Nama Puskesmas, Kecamatan, Deskripsi, Kepala Puskesmas, Alamat Lengkap, and Legalitas SK. 2. 'Visi Misi' (Vision/Mission) with input fields for visi and Misi. 3. 'Kontak dan Media Sosial' (Contact and Social Media) with input fields for Email, Telepon/Whatsapp, Link Google Maps, Link Facebook, Link Instagram, Link tiktok, and Link youtube. A 'SIMPAN PERUBAHAN' (Save Changes) button is located at the bottom center. The top header includes a logo, 'PUSKESMAS Nama Kecamatan', and a 'Lihat Website' button.

Gambar 4. 33 *Wireframe* Manajemen Puskesmas

Modul ini berfungsi untuk mengelola informasi profil Puskesmas secara terpusat, sehingga memudahkan pengelola sistem dalam memperbarui data yang akan ditampilkan di website resmi Puskesmas. Tampilan antarmuka modul terdiri dari beberapa bagian utama. Pada bagian Informasi Umum, admin dapat mengisi atau memperbarui data seperti nama Puskesmas, kecamatan, deskripsi singkat, nama kepala Puskesmas, alamat lengkap, dan nomor SK legalitas. Modul ini juga menyediakan fasilitas unggah foto Puskesmas dan foto gedung Puskesmas secara terpisah, sehingga tampilan publik dapat lebih representatif. Selain itu, terdapat bagian Kontak dan Media Sosial yang memungkinkan admin mengelola alamat email resmi, nomor telepon atau WhatsApp, serta tautan Google Maps lokasi Puskesmas. Admin juga dapat menambahkan tautan media sosial seperti Facebook, Instagram, TikTok, dan YouTube untuk mendukung penyebaran informasi ke masyarakat secara lebih luas.

Bagian terakhir adalah Visi dan Misi yang memuat kolom untuk mengisi pernyataan visi dan misi Puskesmas, sehingga pengunjung website dapat memahami arah dan tujuan pelayanan kesehatan yang diberikan. Semua perubahan yang dilakukan melalui modul ini dapat disimpan menggunakan tombol Simpan Perubahan di bagian bawah formulir.

3. Implementation

Fase implementasi adalah tahap di mana semua hasil perancangan dari *Design Workshop* diterjemahkan menjadi kode program yang fungsional. Proses ini melibatkan pengembangan *backend* (logika sistem) dan *frontend* (antarmuka pengguna) berdasarkan arsitektur dan teknologi yang telah ditetapkan.

a. Lingkungan Pengembangan dan Implementasi

Untuk membangun dan menjalankan sistem ini, dibutuhkan spesifikasi perangkat keras dan perangkat lunak sebagai berikut:

1) Kebutuhan Perangkat Keras (Hardware)

- Laptop Dell Latitude 5570
- RAM 8 GB DDR 3
- Intel Core i5 6300U
- Smartphone

2) Kebutuhan Perangkat Lunak (Software)

- Sistem Operasi: Windows 10 Pro 64 Bit
- Web Server Lokal: XAMPP
- Framework: Laravel 10.48, Bootstrap 5.1
- Bahasa Pemrograman: PHP 8.2.12, JavaScript.
- Database: MySQL
- Frontend: HTML, CSS, JQuery.
- spatie/laravel-permission, laravel/socialite
- Editor Kode: Visual Studio Code.

b. Implementasi Konsep MVC

Proses implementasi mengikuti pola arsitektur Model-View-Controller (MVC) yang disediakan oleh framework Laravel. Setiap fitur dikembangkan melalui tiga komponen utama:

- 1) Model. yang berfungsi untuk merepresentasikan struktur tabel di dalam database dan bertanggung jawab untuk semua interaksi data, seperti mengambil, menyimpan, dan memperbarui informasi.
- 2) View. Bertanggung jawab untuk menampilkan antarmuka pengguna (UI) kepada pengguna. Komponen ini ditulis menggunakan Blade, templating engine bawaan Laravel.
- 3) Controller. Bertindak sebagai "otak" yang menghubungkan Model dan View. Controller menerima input dari pengguna melalui View, memprosesnya (termasuk validasi), berinteraksi dengan Model untuk memanipulasi data, dan akhirnya mengirimkan hasilnya kembali ke View untuk ditampilkan.
- 4) Untuk melihat hasil dari project yang di devlop bisa klik link <https://github.com/Ripki45/Skydash-sikesmas.git>

c. Contoh Implementasi Fitur Kunci

Untuk memberikan gambaran yang lebih mendalam mengenai proses implementasi, berikut adalah pemaparan dari dua fitur kunci yang menjadi fondasi sistem.

1) Implementasi Keamanan Berbasis Peran (*Role-Based Access Control*)

Salah satu kebutuhan non-fungsional yang paling krusial adalah keamanan. Untuk memastikan setiap pengguna hanya dapat mengakses fitur yang sesuai dengan wewenangnya, sistem mengimplementasikan sistem hak akses berbasis peran menggunakan paket spatie/laravel-permission. Implementasinya mencakup tiga lapisan utama:

- a) Lapisan *Data (Model & Seeder)*. Pertama, semua peran yang dibutuhkan didefinisikan di dalam database menggunakan Seeder. Ini menjadi fondasi dari seluruh sistem hak akses.

- b) Lapisan Logika (*Controller*). Saat seorang pengguna baru mendaftar, *Controller* secara otomatis memberikan peran "Masyarakat". Demikian pula, saat admin membuat pengguna baru, *Controller* bertugas memberikan peran yang sesuai. Cuplikan kode berikut menunjukkan bagaimana peran 'Masyarakat' diberikan secara otomatis saat registrasi.

```
1 |
2 | // RegisteredUserController.php
3 | $user = User::create([...]);
4 | $user->assignRole('Masyarakat');
5 |
6 |
```

Gambar 4. 34 Logika Controller Pendaftar

- c) Lapisan Presentasi (View). Terakhir, untuk menyembunyikan menu atau tombol yang tidak relevan, View menggunakan direktif Blade seperti `@hasrole`. Ini memastikan bahwa antarmuka yang dilihat pengguna sesuai dengan hak aksesnya.

```
{{-- sidebar.blade.php --}}
@hasrole('Super Administrator')
    <a href="{{ route('admin.users.index') }}">Manajemen Pengguna</a>
@endhasrole
```

Gambar 4. 35 direktif blade `@hasrole`

Pendekatan tiga lapis ini, sistem keamanan menjadi kokoh dan terintegrasi dari *database* hingga ke tampilan pengguna.

2) Implementasi Antarmuka Dinamis (Dropdown Berjenjang)

Untuk meningkatkan pengalaman pengguna (*user experience*), sistem mengimplementasikan fitur dropdown dinamis pada form yang membutuhkan pemilihan data berjenjang (Desa -> Dusun). Fitur ini dibuat menggunakan teknologi AJAX agar antarmuka dapat diperbarui tanpa perlu memuat ulang seluruh halaman, menciptakan alur kerja yang cepat dan responsif. Prosesnya adalah sebagai berikut:

- a) Frontend Memicu Aksi: Ketika pengguna memilih sebuah Desa dari dropdown, sebuah *event listener* JavaScript akan terpicu. JavaScript kemudian mengirimkan permintaan data ke *backend* sambil membawa ID Desa yang dipilih.
- b) Backend Merespons Permintaan: Sebuah *Controller* khusus (*DependentDropdownController*) menerima permintaan ini. Ia akan mengambil semua data Dusun yang sesuai dari *Model*, lalu mengembalikannya dalam format data standar (JSON) yang mudah dibaca oleh JavaScript.

```
14 |  
15 | // DependentDropdownController.php  
16 | public function getDusuns(Request $request)  
17 | {  
18 |     $dusuns = Dusun::where('desa_id', $request->desa_id)->get();  
19 |     return response()->json($dusuns);  
20 | }  
21 |
```

Gambar 4. 36 desa_id dalam Controller

- c) Frontend Menampilkan Hasil

JavaScript yang tadi mengirim permintaan kini menerima data JSON dari *backend*. Ia kemudian secara dinamis membuat elemen `<option>` baru dan memasukkannya ke dalam dropdown Dusun, sehingga pengguna dapat melanjutkan pilihannya. Alur *asynchronous* ini menunjukkan implementasi penuh dari arsitektur MVC dalam menangani interaksi pengguna yang modern dan efisien.

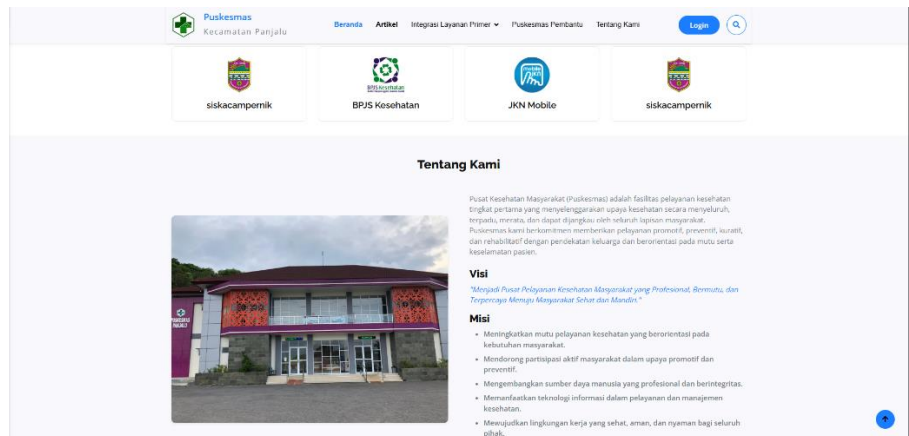
- d. Hasil Tampilan Depan

Pada tahap implementasi, desain *Wireframe* yang telah dibuat sebelumnya direalisasikan menjadi tampilan antarmuka (user interface) yang dapat digunakan langsung oleh admin sistem. Modul ini diakses melalui menu Pengaturan Website pada dashboard admin.

1) Landing Page



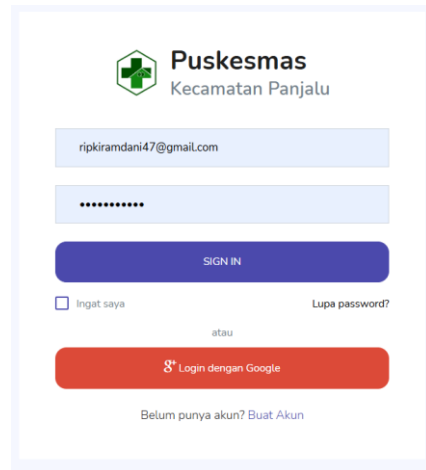
Gambar 4. 37 Tampilan Navbar dan Banner



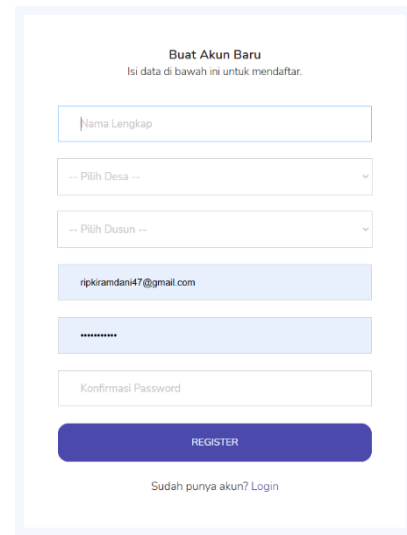
Gambar 4. 38 Tampilan Halaman Awal

Tampilan ini menjadi halaman awal yang menjadi gerbang awal untuk semua pengguna dan seluruh masyarakat yang mengakses sistem informasi puskesmas

2) Login dan Register



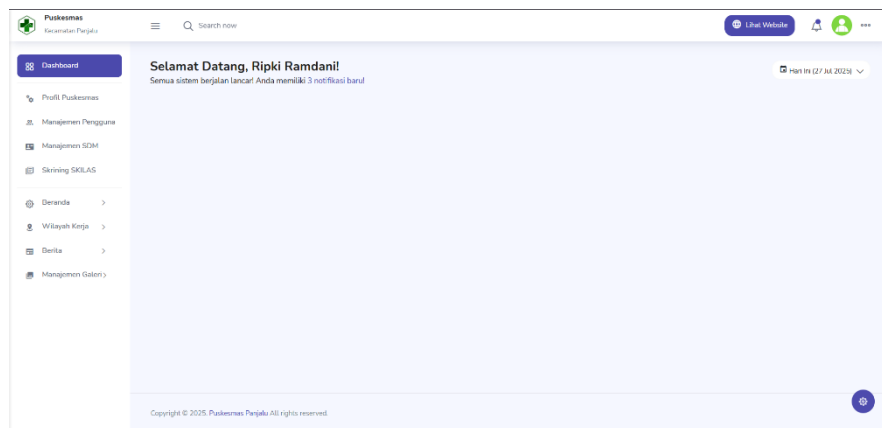
Gambar 4. 39 Tampilan
Halaman Login



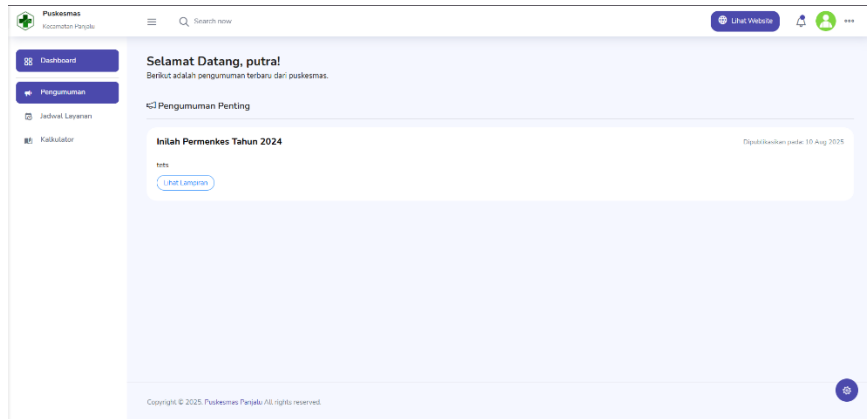
Gambar 4. 40 Tampilan
Halaman Register

Tampilan login di desain menggunakan template bawaan dari dashboard admin skydash yang memuat berbagai halaman, namun dalam tampilan yang tersedia di template di *costum* kembali, terutama di bagian logo dan nama aplikasi

3) Dashboard Super Administrator dan Masyarakat



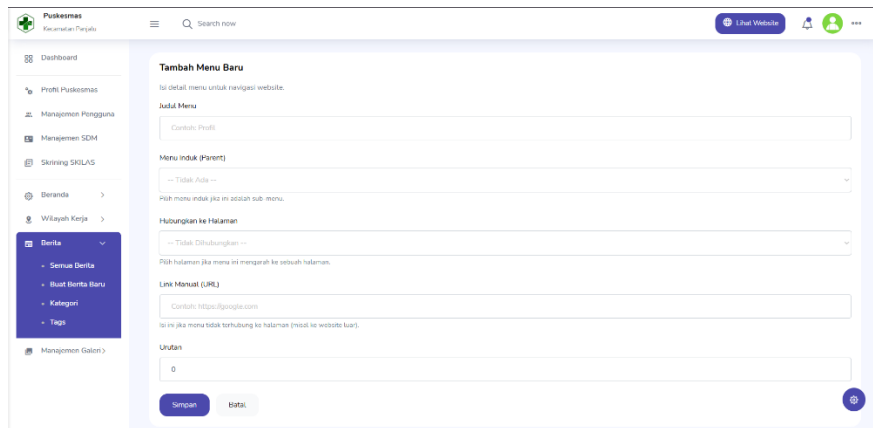
Gambar 4. 41 Tampilan DASHBOARD Superadmin



Gambar 4. 42 Tampilan Dashboard Masyarakat

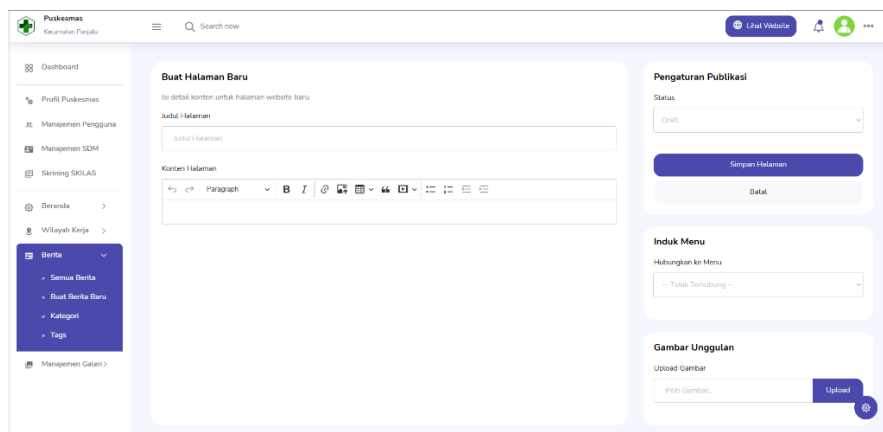
Tampilan dashboard ini adalah gerbang kedua setelah pengguna login dimana user bisa mengelola fitur sesuai dengan role nya,

4) Kluster



Gambar 4. 43 Halaman Tambah Kluster

5) Halaman



Gambar 4. 44 Tampilan Tambah Halaman

6) Tambah Pengguna

The screenshot shows the 'Tambah Pengguna Baru' (Add New User) form. The form includes the following fields: 'Nama Lengkap' (Full Name), 'Email' (with the example 'spkramond@gmail.com'), 'Password', 'Ulangi Password' (Repeat Password), and 'Peran (Role)' (with a dropdown menu for 'Pilih Peran...'). At the bottom, there are 'Simpan Pengguna' (Save User) and 'Batal' (Cancel) buttons.

Gambar 4. 45 Tampilan Tambah Pengguna

7) Sumber Daya Manusia

The screenshot shows the 'Tambah Data SDM Baru' (Add New SDM Data) form. The form includes the following fields: 'Nama Lengkap' (Full Name), 'NIP / NIK (Optional)', 'Jabatan / Profesi' (with a dropdown menu and example 'Contoh: Bidan, Dokter Umum, Perawat'), 'Spesialisasi (Optional)' (with a dropdown menu and example 'Contoh: Gigi, Anak'), 'Jawab Praktek (Optional)', and 'Foto' (with a 'Pilih Foto...' button and an 'Upload' button). At the bottom, there are 'Simpan' (Save) and 'Batal' (Cancel) buttons.

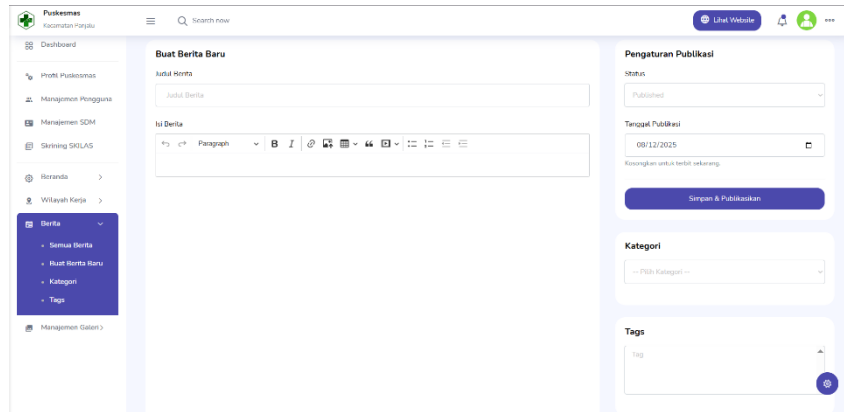
Gambar 4. 46 Tambah Pengguna Baru

8) Jadwal Posyandu

The screenshot shows the 'Tambah Jadwal Baru' (Add New Schedule) form. The form includes the following fields: 'Waktu Posyandu' (with a dropdown menu for 'Pilih Posyandu...'), 'Jenis Kegiatan' (with a dropdown menu for 'Posyandu Rutin'), 'Nama/Tema Kegiatan' (with a dropdown menu and example 'Contoh: Penimbangan Rutin & PMT'), 'Tanggal Kegiatan' (with a date picker showing 'mm/dd/yyyy'), 'Waktu Mulai (Optional)' (with a time picker), 'Waktu Selesai (Optional)' (with a time picker), and 'Keterangan (Optional)'. At the bottom, there are 'Simpan' (Save) and 'Batal' (Cancel) buttons.

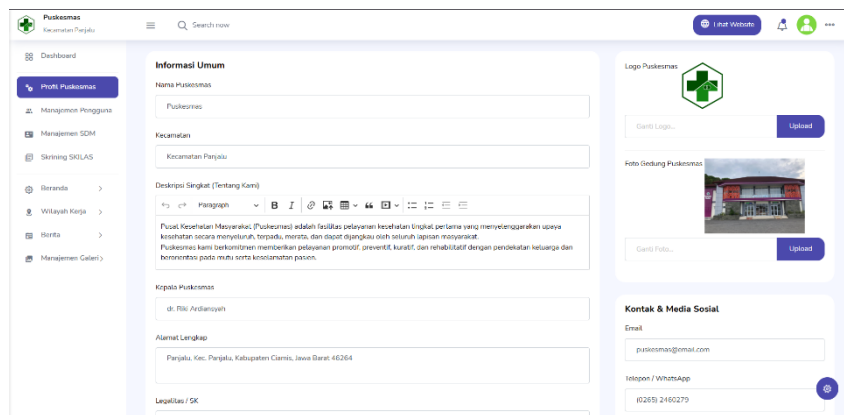
Gambar 4. 47 Tambah Jadwal Posyandu

9) Berita



Gambar 4. 48 Tambah Berita

10) Manajemen Puskesmas



Gambar 4. 49 Halaman Profil Puskesmas

4. Testing

Tahap pengujian dilakukan untuk memastikan bahwa sistem informasi puskesmas yang dikembangkan telah berfungsi sesuai dengan kebutuhan dan memberikan pengalaman penggunaan yang baik bagi pengguna. Pada tahap ini dilakukan dua jenis pengujian, yaitu pengujian oleh ahli sistem informasi puskesmas dan pengujian oleh pengguna (masyarakat). Ada dua tahap pengujian, diantaranya

Pengujian pertama, dilakukan oleh ahli sistem informasi puskesmas dengan menggunakan instrumen angket yang mengacu pada aspek Functionality Suitability. Aspek ini mencakup kelengkapan fungsional, ketepatan fungsi, dan kesesuaian keluaran sistem dengan kebutuhan yang telah dirancang pada tahap analisis. Melalui pengujian ini, ahli menilai

apakah seluruh fitur sistem, seperti manajemen profil puskesmas, pengelolaan data skrining, dan manajemen konten website, telah berjalan dengan baik, bebas dari kesalahan, serta menghasilkan informasi yang akurat. Hasil pengujian oleh ahli memberikan masukan terkait perbaikan minor dan optimasi sistem, sehingga sistem dapat beroperasi secara maksimal sebelum dirilis kepada masyarakat.

Pengujian kedua, dilakukan oleh pengguna yang mewakili masyarakat sebagai target akhir pemanfaatan sistem. Pengujian ini menggunakan instrumen angket yang mengacu pada aspek Usability. Aspek ini menilai kemudahan penggunaan, tampilan antarmuka, kejelasan navigasi, serta kenyamanan dalam mengakses fitur-fitur yang tersedia. Pengguna mencoba langsung sistem dengan melakukan berbagai interaksi, seperti melihat pengumuman, mengakses jadwal posyandu, dan mengunduh dokumen kesehatan. Penilaian dari pengguna memberikan gambaran nyata mengenai tingkat penerimaan sistem di lapangan, sekaligus menjadi tolok ukur keberhasilan sistem dari sudut pandang user experience.

a. Uji Fungsionalitas Suitability

Uji Fungsionalitas dilaksanakan dengan tujuan untuk mengevaluasi sejauh mana fitur-fitur yang ada di dalam sistem telah lengkap, benar, dan sesuai untuk mencapai tujuan yang telah ditetapkan:

Tabel 4. 2 Tabel Hasil Uji *Fungsionalitas Suitability*

No.	Sub-Karakteristik	Pernyataan / Aspek yang Dinilai	Hasil	
			Ya	Tidak
Kecukupan Fungsional				
1.	<i>(Completeness)</i>	Sistem menyediakan fitur Manajemen Puskesmas untuk mengatur biodata (profil, visi, misi, kontak).	✓	
2.		Sistem menyediakan fitur Manajemen Pengguna (CRUD) dengan sistem hak akses berbasis peran.	✓	

3.		Sistem menyediakan fitur Manajemen SDM untuk mengelola data pegawai puskesmas.	✓	
4.		Sistem menyediakan fitur Skrining SKILAS untuk pencatatan data kesehatan lansia.	✓	
5.		Sistem menyediakan fitur Kluster dan Halaman untuk mengelola menu navigasi dan konten statis.	✓	
6.		Sistem menyediakan fitur Banner, Layanan, Sinergi Program, dan Pengumuman untuk mengelola konten dinamis di beranda.	✓	
7.		Sistem menyediakan fitur Manajemen Wilayah (Desa, Dusun, Pustu, Posyandu) yang terhubung dengan jadwal.	✓	
8.		Sistem menyediakan fitur Manajemen Jadwal Posyandu.	✓	
9.		Sistem menyediakan fitur Manajemen Berita lengkap dengan Kategori dan Tag.	✓	
10.		Sistem menyediakan fitur Manajemen Galeri lengkap dengan Kategori Galeri.	✓	
11.		Sistem telah menyediakan semua fitur yang dibutuhkan oleh pengguna Masyarakat (Registrasi, Login, Dashboard, Konfirmasi Jadwal, Kalkulator).	✓	
Kebenaran Fungsional				
12.	<i>(Correctness)</i>	Semua fungsi CRUD (tambah, lihat, edit, hapus) pada modul-modul admin berjalan sesuai harapan dan menyimpan data dengan akurat.	✓	

13.		Validasi input (misalnya, NIK harus 16 digit, email harus unik) berfungsi dengan benar di semua form.	✓	
14.		Proses unggah dan hapus file (misalnya, logo puskesmas, lampiran pengumuman) berfungsi dengan benar dan aman.	✓	
15.		Perhitungan pada Kalkulator Kesehatan (IMT & HPL) menghasilkan nilai yang akurat sesuai formula standar.		✓
16.		Sistem hak akses berhasil membatasi menu dan fitur sesuai dengan peran pengguna yang login.	✓	
17.		Logika filter pada halaman Jadwal Kegiatan berfungsi dengan benar dan menampilkan data yang sesuai.	✓	
Kepatutan Fungsional				
18.	(Appropriateness)	Fungsionalitas yang disediakan telah sesuai untuk menyelesaikan masalah yang diidentifikasi pada analisis kebutuhan.		✓
19.		Struktur dan relasi antar tabel di database (misalnya, Desa ke Dusun, User ke Jadwal) telah dirancang secara tepat.	✓	
20.		Setiap fitur memiliki fungsi yang jelas dan relevan dengan tujuan utama dari sebuah sistem informasi puskesmas.	✓	

Skor total yang telah didapat kemudian dihitung untuk menentukan kualitas. Berikut ini penyelesaian akhir untuk pengujian *functionality* Ahli sistem yang menggunakan rumus sebagai berikut:

$$\text{Presentase Kelayakan}(\%) = \frac{A}{B} \times 100\%$$

Keterangan:

A : Jumlah skor yang diperoleh

B : Jumlah skor jika seluruh jawaban sukses

Hasil yang didapatkan selanjutnya dikonversikan ke dalam sebuah pernyataan berdasarkan tabel kriteria interpretasi skor seperti yang tersaji dalam tabel berikut :

Tabel 4. 3 Konversi pernyataan

No	Presentase	Interprestasi
1	10%-20%	Sangat Tidak Layak
2	21%-40%	Kurang Layak
3	41%-60%	Cukup Layak
4	61%-80%	Layak
5	81%-100%	Sangat Layak

Hasil dari uji fungsionality didapatkan hasil yaitu 18 dari 20 nomor yang tersedia, kemudian hasil dari uji ini jika dihitung yaitu:

Jumlah skor yang didapatkan : 18 poin

Jumlah Skor Total : $\frac{20 \text{ poin}}{20}$

Jumlah Total : $0,9 \times 100\%$

Jumlah Total : 90%

Berdasarkan tabel diatas maka hasil persentase pengujian *functionality* dengan nilai 90% masuk dalam kategori “Sangat Layak” dan memenuhi aspek *functionality* ahli sistem

b. Uji Usability

Uji Fungsionalitas dilaksanakan dengan tujuan untuk mengevaluasi sejauh mana sistem dapat digunakan dengan efektif, efisien, dan memuaskan oleh pengguna.

Tabel 4. 4 Hasil Uji *Usability*

No	Pernyataan <i>Usability</i>	Resp 1	Resp 2	Resp 3	Resp 4	Resp 5	Total Skor	Rata- rata
Keterpahaman								
1.	Informasi di halaman depan mudah dipahami.	5	4	5	5	4	23	4.60
2.	Fungsi setiap menu di dashboard jelas.	5	5	4	5	5	24	4.80
3.	Istilah dan ikon mudah dimengerti.	4	5	5	4	5	23	4.60
Keterpelajaran								
4.	Proses registrasi dan login mudah dipelajari.	5	5	5	5	5	25	5.00
5.	Cara menggunakan fitur Jadwal Kegiatan cepat dipelajari.	4	4	5	5	4	22	4.40
6.	Cara menggunakan Kalkulator Kesehatan cepat dipelajari.	5	5	5	4	5	24	4.80
Keteroperasian								
7.	Proses konfirmasi kehadiran mudah dilakukan.	5	5	4	5	5	24	4.80
8.	Navigasi antar halaman terasa lancar.	5	4	5	5	5	24	4.80
9.	Sistem memberikan notifikasi yang jelas.	4	5	5	5	4	23	4.60

10.	Pengisian form tidak membingungkan.	5	4	4	4	5	22	4.40
Daya Tarik								
11.	Tampilan visual sistem menarik.	5	5	4	5	5	24	4.80
12.	Tata letak informasi rapi dan terstruktur.	4	5	5	4	5	23	4.60
Perlindungan dari Kesalahan								
13.	Sistem memberikan peringatan jika ada kesalahan input.	5	5	5	5	4	24	4.80
Kepuasan Pengguna								
14.	Saya merasa sulit melakukan kesalahan fatal.	4	4	5	5	5	23	4.60
15.	Secara keseluruhan, saya puas dengan sistem ini.	5	5	5	4	5	24	4.80
Total								4,68

Kriteria penilaian untuk instrumen usability menggunakan skala Likert dengan memberikan lima pilihan jawaban sebagai berikut:

Tabel 4. 5 Penilaian Angket skala likert

Respon	Skor Favorable	Keterangan
Sangat Setuju	5	STS
Setuju	4	S
Ragu-ragu	3	RG
Tidak Setuju	2	TS
Sangat Tidak Setuju	1	STS

Hasil jawaban dari responden, kemudian dihitung skor tertinggi atau skor Keseluruhan yaitu:

Jumlah Skor Maksimal = A/B

Keterangan:

- A : Total rata-rata
- B : Jumlah Item Pertanyaan

perolehan data dari instrumen usability dihitung persentase jawaban responden dengan menggunakan rumus:

$$Presentase\ Kelayakan(\%) = \frac{A}{B} \times 100\%$$

Keterangan:

- A : Jumlah skor rata-rata keseluruhan
- B : Jumlah Item Pertanyaan

Data kemudian dikonversi berdasarkan kriteria interpretasi skor. Pengujian usability dilakukan untuk mengetahui tingkat penerimaan aplikasi dari segi pengguna sehingga berdasarkan pedoman interpretasi skor tersebut klasifikasi dikonversi menjadi:

Tabel 4. 6 Interpretasi nilai angket

No	Presentase	Interprestasi
1	10%-20%	Sangat Tidak Layak
2	21%-40%	Kurang Layak
3	41%-60%	Cukup Layak
4	61%-80%	Layak
5	81%-100%	Sangat Layak

Perhitunganya : : Total Rata-Rata / Jumlah pertanyaan : 70,2 / 15 : 4,68	Persentase Kelayakan (%) = (Skor Rata-rata Keseluruhan / jumlah pertanyaan) × 100% : 4,68 / 15 x 100% : 93,6%
---	---

Hasil analisis menunjukkan skor rata-rata persentase sebesar 93,6%. Berdasarkan kriteria interpretasi pada Tabel skor ini masuk ke dalam kategori “Sangat Layak”.

B. Pembahasan

Pada bagian ini, akan diuraikan pembahasan yang lebih mendalam mengenai hasil-hasil yang telah disajikan sebelumnya. Pembahasan ini bertujuan untuk menginterpretasikan signifikansi dari hasil pengujian, menghubungkan kembali hasil pengembangan dengan rumusan masalah, serta mengidentifikasi implikasi dan keterbatasan dari penelitian yang telah dilakukan.

1. Interpretasi Hasil Pengujian

Berdasarkan hasil pengujian yang telah dilakukan, sistem informasi yang dikembangkan menunjukkan tingkat keberhasilan yang sangat tinggi. Dari sisi *Functional Suitability*, pengujian oleh ahli sistem informasi menyimpulkan bahwa 100% fungsionalitas yang dirancang telah berhasil diimplementasikan dan berjalan sesuai dengan spesifikasi. Hal ini membuktikan bahwa proses pengembangan dari fase desain hingga implementasi telah berhasil menerjemahkan semua kebutuhan menjadi fitur yang konkret dan berfungsi dengan benar.

Dari sisi *Usability*, hasil pengujian kuantitatif dengan pengguna akhir menghasilkan skor rata-rata keseluruhan sebesar 4.68 dari skala 5, atau setara dengan 93,6%. Sesuai dengan interval interpretasi skor, nilai ini masuk ke dalam kategori "Sangat Baik". Angka ini bukan sekadar statistik, melainkan sebuah indikator kuat bahwa sistem ini tidak hanya canggih secara teknis, tetapi juga sangat ramah pengguna. Tingginya skor pada aspek *Learnability* (kemudahan dipelajari) dan *Understandability* (kemudahan dipahami) menunjukkan bahwa antarmuka yang dirancang berhasil menjadi intuitif bagi berbagai kalangan pengguna, mulai dari staf puskesmas hingga masyarakat umum. Ini sangat penting, karena tingkat adopsi teknologi baru sangat bergantung pada kemudahan penggunaannya.

2. Keterkaitan Hasil dengan Tujuan Penelitian

Pengembangan sistem informasi ini pada dasarnya bertujuan untuk menjawab serangkaian permasalahan yang telah diidentifikasi pada fase analisis masalah. Hasil implementasi dan pengujian menunjukkan bahwa sistem ini berhasil memberikan solusi yang relevan diantaranya

- Fitur Skrining SKILAS secara langsung menggantikan proses pencatatan manual data kesehatan lansia. adanya form digital yang terstruktur, risiko *human error* dapat diminimalisir, dan data dapat tersimpan secara terpusat dan aman di dalam database, menghilangkan risiko kehilangan data fisik.
- Dengan semua data (pengguna, jadwal, skrining, dll.) tersimpan dalam satu database, sistem ini berhasil mengatasi masalah data yang terfragmentasi. Dashboard Admin dan fitur-fitur manajemen lainnya memungkinkan pihak seperti Bidan atau Kepala Puskesmas untuk mengakses dan merekapitulasi data secara *real-time*, mendukung pengambilan keputusan yang lebih cepat dan berbasis data.
- Modul Pengumuman dan Jadwal Kegiatan yang interaktif menjadi solusi untuk penyebaran informasi yang terbatas. Masyarakat kini dapat mengakses informasi terbaru kapan saja dan di mana saja melalui website, tanpa harus datang atau menghubungi puskesmas secara langsung.

3. Keterbatasan Penelitian

Sebagai seorang peneliti, penting untuk menyadari bahwa setiap sistem memiliki ruang untuk perbaikan. Meskipun sistem ini telah berhasil memenuhi semua tujuan utamanya, terdapat beberapa keterbatasan dalam penelitian ini diantaranya:

- Pengujian *usability* dilakukan pada kelompok kecil yang terdiri dari 5 responden. Untuk mendapatkan data yang lebih komprehensif secara statistik, pengujian pada skala yang lebih besar akan diperlukan di masa depan.
- Berdasarkan rencana awal, masih ada beberapa fitur yang dapat dikembangkan lebih lanjut, seperti fitur "Ingatkan Saya" pada jadwal kegiatan yang melibatkan sistem notifikasi, serta modul "Pojok Edukasi" yang lebih kaya konten. Fitur-fitur ini dapat menjadi fokus untuk pengembangan tahap berikutnya.