RANCANG BANGUN APLIKASI *WEB BUILDER* BERBASIS LARAVEL FILAMENT UNTUK MENDUKUNG PENGELOLAAN WEBSITE ORGANISASI PERANGKAT DAERAH DI KABUPATEN KARIMUN

PROYEK AKHIR

Disusun Oleh:

Farhan Najedi

3312301019

Disusun untuk memenuhi salah satu syarat memperoleh gelar Ahli Madya Teknik Informatika

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNIK INFORMATIKA

POLITEKNIK NEGERI

BATAM

2026

KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat Allah SWT, Tuhan Yang Maha Pengasih dan Maha Penyayang yang telah melimpahkan segala rahmat, hidayah, dan inayah-Nya kepada penulis sehingga proposal tugas akhir ini yang berjudul "Rancang Bangun Aplikasi *Web Builder* Berbasis Laravel Filament Untuk Mendukung Pengelolaan Website Organisasi Perangkat Daerah Di Kabupaten Karimun" dapat penulis selesaikan dengan baik.

Penyusunan proposal tugas akhir ini tidak lepas dari bantuan, dukungan, dan bimbingan dari berbagai pihak. Oleh karena itu, pada kesempatan ini, penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Ibu Alena Uperiati, S.T., M.Cs., Dosen Pembimbing yang bersedia memberikan bimbingan selama proses penyusunan Tugas Akhir.
2. Bapak/Ibu pembimbing lapangan di Dinas Komunikasi Informatika Statistik dan Persandian Kabupaten Karimun yang telah memberikan ilmu serta bimbingan dan dukungan secara langsung.
3. Kedua orang tua beserta keluarga besar tercinta untuk doa restu, dukungan, dan motivasi yang selalu diberikan kepada Penulis.
4. Semua pihak yang tidak dapat disebutkan satu per satu, yang telah memberikan bantuan dan dukungan dalam penyusunan proposal ini.

Penulis menyadari bahwa proposal ini masih terdapat kekurangan dan keterbatasan. Oleh karena itu, penulis mengharapkan kritik serta saran demi perbaikan dan penyempurnaan di masa yang akan datang. Semoga tugas akhir ini dapat memberikan manfaat dan kontribusi positif bagi pengembangan Ilmu pengetahuan dan teknologi, khususnya di bidang Teknik Informatika.

DAFTAR ISI

KATA PENGANTAR ii
DAFTAR ISI iii
DAFTAR TABEL vi
DAFTAR GAMBAR vii
DAFTAR LAMPIRAN xi
ABSTRAK xii
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Tujuan 2
1.4 Batasan Masalah 2
1.5 Manfaat 3
1.6 Kolaborasi dan Sinergi dalam Pelaksanaan Proyek 4
2.1 Sistem Terkait 5
2.2 Landasan Teori 7
2.2.1 Sistem Informasi dan Aplikasi Berbasis Web 7
2.2.2 *Content Management System (CMS)* 8
2.2.3 Laravel 8
2.2.4 MySQL 8
2.2.5 Filament 9
2.2.6 Arsitektur *Multi-Tenant* 9
2.2.7 Organisasi Perangkat Daerah (OPD) 9
2.3 Metode Pengembangan Produk 10
BAB III ANALISIS DAN PERANCANGAN 13
3.1. Analisis Kebutuhan 13
3.1.1 Proses Bisnis Berjalan 13
3.1.2 Gambaran Umum Sistem 14
3.1.3 Aturan dan Batasan Teknis Sistem 15
3.2. Perancangan 16
3.2.1 Kebutuhan Fungsional 16
3.2.2 Kebutuhan Non-Fungsional 18
3.2.3 Diagram *Use Case* 19
3.2.4 Skenario *Use Case* 21

3.2.5 Activity diagram 44
3.2.6 ER Diagram 59
3.2.7 Perancangan Antarmuka (Wireframe) 60
BAB IV IMPLEMENTASI DAN PENGUJIAN 92
4.1. Hasil Implementasi 92
4.1.1 Implementasi Halaman Login 92
4.1.2 Implementasi Halaman Dasbor Super Admin 93
4.1.3 Implementasi Halaman Daftar Master OPD dan Tambah OPD 93
4.1.4 Implementasi Halaman Form Tambah Data OPD 94
4.1.5 Implementasi Halaman Manajemen Pengguna 94
4.1.6 Implementasi Halaman Tambah Akun Pengguna 95
4.1.7 Implementasi Halaman Manajemen Peran 95
4.1.8 Implementasi Halaman Form Konfigurasi Peran 96
4.1.9 Implementasi Halaman Dasbor Admin OPD 97
4.1.10 Implementasi Halaman Pengelolaan Pengumuman Admin OPD. 97
4.1.11 Implementasi Halaman Tambah Pengumuman Admin OPD 98
4.1.12 Implementasi Halaman Daftar Galeri Admin OPD 98
4.1.13 Implementasi Halaman Tambah Galeri Admin OPD 99
4.1.14 Implementasi Halaman Banner Admin OPD 99
4.1.15 Implementasi Halaman Tambah Banner Admin OPD 100
4.1.16 Implementasi Halaman Daftar Kategori Berita Admin 100
4.1.17 Implementasi Halaman Tambah Kategori Berita Admin 101
4.1.18 Implementasi Halaman Daftar Berita Admin 101
4.1.19 Implementasi Halaman Tambah Data Berita Admin 102
4.1.20 Implementasi Halaman Pengaturan OPD Admin 102
4.1.21 Implementasi Halaman Pengelolaan Profil OPD Admin 103
4.1.22 Implementasi Halaman Pengelolaan Konten Baru Admin 104
4.1.23 Implementasi Halaman Tambah Konten Halaman Baru Admin 104
4.1.24 Implementasi Halaman Pengelolaan Daftar Menu Baru Admin. 105
4.1.25 Implementasi Halaman Tambah Menu Baru Admin 105
4.1.26 Implementasi Halaman Publikasi Dokumen Admin 106
4.1.27 Implementasi Halaman Tambah Dokumen Admin 106
4.1.28 Implementasi Halaman Kategori Dokumen Admin 107
4.1.29 Implementasi Halaman Tambah Kategori Dokumen Admin 107
4.1.30 Implementasi Halaman Layanan Admin 107

4.1.31 Implementasi Halaman Tambah Layanan Admin 108
4.1.32 Implementasi Halaman Dasbor Pengguna 109
4.1.33 Implementasi Halaman Profil Pengguna 110
4.1.34 Implementasi Halaman Pengumuman Pengguna 111
4.1.35 Implementasi Halaman Detail Pengumuman 111
4.1.36 Implementasi Halaman Kategori Berita Pengguna 112
4.1.37 Implementasi Halaman Berita 113
4.1.38 Implementasi Halaman Detail Berita 114
4.1.39 Implementasi Halaman Layanan 115
4.1.40 Implementasi Halaman Detail Layanan Pengguna 115
4.1.41 Implementasi Halaman Galeri Pengguna 116
4.1.42 Implementasi Halaman Detail Galeri Pengguna 117
4.1.43 Implementasi Halaman Publikasi Dokumen 117
4.1.44 Implementasi Halaman Kategori Dokumen 118
4.1.45 Implementasi Halaman Menu baru Pengguna 119
4.1.46 Implementasi Halaman Detail Halaman Baru Pengguna 119
4.2. Pengujian *User Acceptance Testing* (UAT) 120
BAB V KESIMPULAN 121
5.1 Kesimpulan 122
5.2 Saran 122
DAFTAR PUSTAKA 124
DAFTAR LAMPIRAN 126

DAFTAR TABEL

Tabel 2. 1 Perbandingan Penelitian Sebelumnya 6
Tabel 3. 1 Kebutuhan Fungsional..... 17
Tabel 3. 2 Kebutuhan Non-Fungsional 18
Tabel 3. 3 Skenario Use Case Login 21
Tabel 3. 4 Skenario Use Case Mengelola Data Akun Admin OPD 22
Tabel 3. 5 Skenario Use Case Mengelola data OPD 24
Tabel 3. 6 Skenario Use Case Mengelola data profil OPD 26
Tabel 3. 7 Skenario Use Case Mengelola data berita OPD..... 27
Tabel 3. 8 Skenario Use Case Mengelola data kategori berita OPD..... 29
Tabel 3. 9 Skenario Use Case Mengelola data galeri OPD..... 30
Tabel 3. 10 Skenario Use Case Mengelola data kategori dokumen OPD..... 31
Tabel 3. 11 Skenario Use Case Mengelola layanan OPD 33
Tabel 3. 12 Skenario Use Case Mengelola Arsip Dokumen OPD..... 35
Tabel 3. 13 Skenario Use Case Mengelola banner dashboard OPD 36
Tabel 3. 14 Skenario Use Case Admin OPD Mengelola data Halaman OPD..... 38
Tabel 3. 15 Skenario Use Case Admin OPD Mengelola data Menu Halaman OPD 40
Tabel 3. 16 Skenario Use Case Admin OPD Mengelola data Identitas OPD 41
Tabel 3. 17 Skenario Usecase Super Admin Mengelola data Peran Pengguna..... 43
Tabel 4. 1 Tabel Pengujian User Acceptance Test(UAT) Aktor Super Admin .. 120
Tabel 4. 2 Tabel Pengujian User Acceptance Test(UAT) Aktor Admin OPD.... 121

DAFTAR GAMBAR

Gambar 2.1 Metode Waterfall 10
Gambar 3. 1 Flowchart Bisnis Saat Ini 14
Gambar 3. 2 Gambaran Umum Sistem 15
Gambar 3. 3 Usecase Diagram 20
Gambar 3. 4 Activity Diagram Login 45
Gambar 3. 5 Activity Diagram Mengelola Daftar Pengguna 46
Gambar 3. 6 Activity Diagram Mengelola Data OPD 47
Gambar 3. 7 Activity Diagram Mengelola Data Pengaturan OPD 48
Gambar 3. 8 Activity Diagram Mengelola Data Berita 49
Gambar 3. 9Activity Diagram Mengelola Data Kategori Berita 50
Gambar 3. 10 Activity Diagram Mengelola Data Galeri 51
Gambar 3. 11 Activity Diagram Mengelola Data Kategori Dokumen 52
Gambar 3. 12 Activity Diagram Mengelola Data Layanan OPD 53
Gambar 3. 13 Activity Diagram Mengelola Data Arsip Dokumen 54
Gambar 3. 14 Activity Diagram Mengelola Data Banner 55
Gambar 3. 15 Activity Diagram Mengelola Data Profil OPD 56
Gambar 3. 16 Activity Diagram Mengelola Data Tambah Menu Halaman OPD 57
Gambar 3. 17 Activity Diagram Mengelola Data Halaman Baru OPD 58
Gambar 3. 18 Activity Diagram Logout 59
Gambar 3. 19 ER Diagram 59
Gambar 3. 20 Wireframe Halaman Login 60
Gambar 3. 21 Wireframe Halaman Dashboard Super Admin 61
Gambar 3. 22 Wireframe Halaman Pengaturan OPD Superadmin 61
Gambar 3. 23 Wireframe Halaman Daftar OPD Superadmin 62
Gambar 3. 24 Wireframe Halaman Superadmin Tambah OPD 63
Gambar 3. 25 Wireframe Halaman Daftar Pengguna 63
Gambar 3. 26 Wireframe Antarmuka Tambah Pengguna 64
Gambar 3. 27 Wireframe Antarmuka Halaman Peran Pengguna 65
Gambar 3. 28 Wireframe Antarmuka Halaman Tambah Peran Pengguna 66
Gambar 3. 29 Wireframe Antarmuka Halaman Dasbor Admin OPD 67

Gambar 3. 30 Wireframe Antarmuka Halaman Daftar Pengumuman Admin 67
Gambar 3. 31 Wireframe Antarmuka Halaman Tambah Pengumuman 68
Gambar 3. 32 Wireframe Halaman Daftar Galeri Admin OPD 68
Gambar 3. 33 Wireframe Halaman Admin OPD Tambah Data Galeri.....69
Gambar 3. 34 Wireframe Halaman Data Banner Admin OPD 69
Gambar 3. 35 Wireframe Halaman Admin OPD Tambah Data Banner 70
Gambar 3. 36 Wireframe Halaman Daftar Kategori Berita Admin OPD 71
Gambar 3. 37 Wireframe Halaman Admin OPD Tambah Data Kategori Berita...71
Gambar 3. 38 Wireframe Halaman Daftar Berita Admin OPD 72
Gambar 3. 39 Wireframe Halaman Admin OPD Tambah Berita 72
Gambar 3. 40 Wireframe Halaman Pengaturan OPD Admin 73
Gambar 3. 41 Wireframe Halaman Admin Tambah Profil OPD 74
Gambar 3. 42 Wireframe Halaman Daftar Isi Halaman Baru Admin.....75
Gambar 3. 43 Wireframe Halaman Tambah Daftar Isi Halaman Baru Admin.....76
Gambar 3. 44 Wireframe Halaman Menu Navigasi Kustom Admin 77
Gambar 3. 45 Wireframe Halaman Publikasi Dokumen Admin.....77
Gambar 3. 46 Wireframe Halaman Tambah Dokumen Baru Admin.....78
Gambar 3. 47 Wireframe Halaman Kategori Dokumen Admin.....79
Gambar 3. 48 Wireframe Halaman Tambah Kategori Dokumen Admin 79
Gambar 3. 49 Wireframe Halaman Layanan OPD Admin 80
Gambar 3. 50 Wireframe Halaman Admin OPD Tambah Data Layanan 80
Gambar 3. 51 Wireframe Halaman Beranda Pengguna 81
Gambar 3. 52 Wireframe Halaman Profil Pengguna.....82
Gambar 3. 53 Wireframe Halaman Pengumuman Pengguna.....83
Gambar 3. 54 Wireframe Halaman Detail Pengumuman Pengguna 84
Gambar 3. 55 Wireframe Halaman Berita Pengguna 85
Gambar 3. 56 Wireframe Halaman Detail Berita 86
Gambar 3. 57 Wireframe Halaman Layanan Pengguna 87
Gambar 3. 58 Wireframe Halaman Galeri Pengguna.....87
Gambar 3. 59 Wireframe Halaman Detail Galeri Pengguna 88
Gambar 3. 60 Wireframe Halaman Publikasi Dokumen Pengguna 89
Gambar 3. 61 Wireframe Halaman Menu Baru Pengguna.....90

Gambar 3. 62 Wireframe Halaman Detail Daftar Isi Halaman Baru 91
Gambar 4. 1 Halaman Login 92
Gambar 4. 2 Halaman Dashboard Super Admin 93
Gambar 4. 3 Halaman Daftar OPD Superadmin 93
Gambar 4. 4 Halaman Tambah Data OPD Baru 94
Gambar 4. 5 Halaman Daftar Pengguna 94
Gambar 4. 6 Halaman Tambah Pengguna 95
Gambar 4. 7 Halaman Peran Pengguna 95
Gambar 4. 8 Halaman Tambah Peran Pengguna 96
Gambar 4. 9 Halaman Dasbor Admin OPD 97
Gambar 4. 10 Halaman Pengelolaan Pengumuman OPD 97
Gambar 4. 11 Halaman Tambah Pengumuman Admin OPD 98
Gambar 4. 12 Halaman Daftar Galeri Admin OPD 98
Gambar 4. 13 Halaman Tambah Galeri Admin OPD 99
Gambar 4. 14 Halaman Daftar Banner OPD 99
Gambar 4. 15 Halaman Tambah Data Banner Admin OPD 100
Gambar 4. 16 Halaman Kategori Berita Admin OPD 100
Gambar 4. 17 Halaman Tambah Kategori Berita Admin 101
Gambar 4. 18 Halaman Berita Admin 101
Gambar 4. 19 Halaman Tambah Data Berita Admin 102
Gambar 4. 20 Halaman Pengaturan OPD Admin 102
Gambar 4. 21 Halaman Pengelolaan Profil OPD 103
Gambar 4. 22 Halaman Pengelolaan Konten Baru 104
Gambar 4. 23 Halaman Tambah Konten Baru Admin 104
Gambar 4. 24 Halaman Pengelolaan Daftar Menu Baru Admin 105
Gambar 4. 25 Halaman Tambah Menu Baru Admin 105
Gambar 4. 26 Halaman Publikasi Dokumen Admin 106
Gambar 4. 27 Halaman Tambah Dokumen Admin 106
Gambar 4. 28 Halaman Kategori Dokumen Admin 107
Gambar 4. 29 Halaman Tambah Kategori Dokumen Admin 107
Gambar 4. 30 Halaman Layanan Admin 107
Gambar 4. 31 Halaman Tambah Layanan Admin 108

Gambar 4. 32 Halaman Beranda Website OPD 109
Gambar 4. 33 Halaman Profil OPD..... 110
Gambar 4. 34 Halaman Pengumuman OPD..... 111
Gambar 4. 35 Halaman Detail Pengumuman 111
Gambar 4. 36 Halaman Kategori Berita 112
Gambar 4. 37 Halaman Berita Pengguna 113
Gambar 4. 38 Halaman Detail Berita 114
Gambar 4. 39 Halaman Layanan 115
Gambar 4. 40 Halaman Detail Layanan Pengguna 115
Gambar 4. 41 Halaman Galeri Pengguna 116
Gambar 4. 42 Halaman Detail Galeri 117
Gambar 4. 43 Halaman Publikasi Dokumen 117
Gambar 4. 44 Halaman Kategori Dokumen 118
Gambar 4. 45 Halaman Menu Baru Pengguna 119
Gambar 4. 46 Halaman Detail Halaman Baru 119

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement 126
Lampiran B. Link Produk 126
Lampiran C. Dokumen Pengujian UAT 126

ABSTRAK

Perkembangan teknologi informasi dan komunikasi memberikan dampak signifikan terhadap penyelengaraan pemerintah daerah, khususnya dalam penyampaian informasi publik kepada masyarakat. Situs web (website) resmi instansi menjadi salah satu media penting bagi Organisasi Perangkat Daerah (OPD) dalam menyampaikan informasi secara cepat, transparan, dan mudah diakses. Namun dalam penerapannya, proses pembuatan dan pengelolaan website pada setiap OPD di Kabupaten Karimun masih menghadapi berbagai kendala, seperti keterbatasan sumber daya manusia yang memiliki kemampuan di bidang pengembangan website serta tingginya ketergantungan pada pihak ketiga dalam pembaruan dalam pemeliharaan konten. Berdasarkan permasalahan tersebut, diperlukan suatu sistem yang dapat mempermudah setiap OPD dalam membuat, mengelola, dan memelihara website instansi mereka secara mandiri. Dalam hal ini, Dinas Komunikasi Informatika Statistik dan Persandian (Diskominfo) Kabupaten Karimun berperan dalam memfasilitasi infrastruktur terpadu agar setiap admin OPD memiliki kendali penuh dan kemudahan dalam mengelola media informasinya masing-masing. Penelitian ini bertujuan untuk merancang dan membangun aplikasi *Web Builder* berbasis Laravel Filament yang memfasilitasi admin instansi dalam mengelola website secara fleksibel, mandiri, dan efisien dalam satu sistem terintegrasi. Metode pengembangan perangkat lunak yang digunakan adalah dengan metode pengembangan Waterfall yang meliputi tahap analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Aplikasi ini dikembangkan menggunakan bahasa pemrograman PHP serta MySQL sebagai sistem manajemen basis data. Hasil penelitian ini berupa aplikasi *Web Builder* yang menyediakan fitur pengelolaan berita, galeri, arsip dokumen, dan profil instansi melalui antarmuka admin yang responsif. Sistem ini diharapkan dapat meningkatkan efisiensi operasional admin OPD serta mengoptimalkan penyebaran informasi publik di Kabupaten Karimun.

Kata Kunci: Laravel Filament, Sistem Informasi, Website OPD, Sistem Informasi Pemerintahan, Digitalisasi Layanan Publik.

BAB I PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi dan komunikasi di era globalisasi saat ini telah membawa perubahan besar dalam berbagai aspek kehidupan, termasuk dalam tata kelola sistem pemerintahan daerah. Dinas Komunikasi Informatika Statistik dan Persandian (Diskominfo) kabupaten karimun, sebagai instansi pembina dan pengelola infrastruktur digital, dituntut untuk selalu beradaptasi guna meningkatkan efektivitas penyebaran informasi dan pengelolaan data secara digital. Penyediaan situs web bagi setiap Organisasi Perangkat Daerah (OPD) menjadi langkah strategis dalam mewujudkan keterbukaan informasi dan pelayanan publik modern.

Namun pada praktiknya, proses pembuatan, pengelolaan, dan pemeliharaan website masing-masing OPD kerap menghadapi kendala teknis. Kendala utama terletak pada keterbatasan sumber daya manusia (SDM) teknis di bidang pemrograman web pada masing-masing instansi. Akibatnya, setiap pengembangan website baru maupun pembaruan modul dan pemeliharaan rutin, pihak OPD masih sangat bergantung pada pengembang pihak ketiga (*vendor* eksternal). Ketergantungan ini berdampak langsung pada lambatnya proses pembaruan informasi publik, tingginya biaya pengadaan rutin, serta panjangnya alur birokrasi koordinasi.

Selain itu, sistem yang digunakan oleh masing-masing OPD masih bersifat terpisah atau *silo system* (sistem yang bekerja secara mandiri dan tidak terhubung satu sama lain). Hal ini mengakibatkan beban pengelolaan infrastruktur server dan domain menjadi tidak efisien. Diskominfo menghadapi kesulitan dalam memantau keaktifan website dan ketersediaan layanan publik secara terpadu.

Untuk mengatasi permasalahan tersebut, dibutuhkan solusi teknologi berupa Aplikasi *Web Builder* yang dirancang khusus agar admin di setiap OPD dalam membuat, mengelola, dan memelihara *website* resmi mereka masing-masing tanpa memerlukan keahlian pengodean (*coding*) yang mendalam. Dengan memanfaatkan *Framework* Laravel dan panel admin Filament, aplikasi ini diharapkan mampu meningkatkan efisiensi anggaran, mengurangi ketergantungan terhadap pihak eksternal, mempercepat proses pembaruan informasi publik, serta

mendukung terwujudnya Tata Kelola Pemerintahan Berbasis Elektronik (SPBE) yang lebih transparan, responsif, dan berkelanjutan di Kabupaten Karimun.

1.2 Rumusan Masalah

Berdasarkan Penjelasan diatas, dapat diidentifikasi beberapa permasalahan yang dihadapi oleh Dinas Komunikasi dan Informatika Kabupaten Karimun, yaitu:

1. Bagaimana merancang dan membangun aplikasi *web builder* yang memudahkan OPD di Kabupaten Karimun dalam mengelola informasi publik secara mandiri?
2. Bagaimana mengimplementasikan *Framework* Laravel dan Filament untuk menghasilkan sistem manajemen website secara mandiri oleh staff OPD?

1.3 Tujuan

Tujuan dari aplikasi ini adalah:

1. Merancang dan membangun aplikasi *web builder* berbasis web yang dapat digunakan oleh staff OPD untuk mempublikasikan dan memperbarui konten *website* intansi secara mandiri.
2. Mengimplementasikan *framework* Laravel dan Filament untuk menciptakan sistem manajemen konten untuk mendukung pengelolaan dan pembaruan data website secara mandiri oleh staff OPD. Menerapkan fitur pengelolaan berita, galeri, publikasi dokumen, dan profil instansi yang responsif menggunakan Laravel Filament guna meningkatkan efisiensi kerja di setiap OPD.

1.4 Batasan Masalah

Batasan masalah dari proyek ini adalah:

1. Pengembangan difokuskan pada penyediaan fitur standar situs web OPD yang bersifat informatif, yang mencakup pengelolaan profil instansi, berita, pengumuman, informasi layananan instansi, galeri foto, dan publikasi dokumen publik.
2. Sistem menerapkan arsitektur *multi-tenant* dengan pendekatan *single database shared schema*, yang memungkinkan Super Admin (Diskominfo)

mengalokasikan dan mengelola website untuk setiap OPD, sedangkan Admin OPD bertanggung jawab dalam mengelola konten website instansinya masing-masing melalui platform yang sama.

3. Sistem hanya berfokus pada manajemen konten website (CMS) secara mandiri dan tidak mencakup integrasi dengan sistem pemerintah lainnya seperti sistem e-government atau sistem layanan publik lainnya.
4. Pengembangan aplikasi menggunakan bahasa pemrograman PHP dengan *Framework* Laravel, panel administrasi Filament sebagai antarmuka pengelolaan sistem, serta MySQL sebagai sistem manajemen basis data.

1.5 Manfaat

Manfaat dari aplikasi ini adalah:

1. Penelitian ini membantu meningkatkan ilmu pengetahuan dengan menjadi referensi ilmiah tentang implementasi arsitektur *multi-tenant* pada lingkungan pemerintahan daerah. Tujuannya adalah untuk menciptakan infrastruktur aplikasi yang efisien dan terpusat.
2. Aplikasi ini membantu Dinas Komunikasi Informatika Statistik dan Persandian Kabupaten Karimun dalam mengawasi dan melakukan standarisasi situs web OPD melalui satu platform, sehingga mempermudah proses pemeliharaan sistem.
3. Proyek ini memberikan kemungkinan untuk setiap Organisasi Perangkat Daerah (OPD) dalam mengelola profil, berita, galeri, dan dokumen publik secara langsung tanpa harus bergantung pada pengembang eksternal setiap kali memerlukan perubahan konten web.
4. Hasil proyek ini dapat meningkatkan kualitas layanan informasi bagi masyarakat Kabupaten Karimun, sehingga setiap masyarakat dapat mengakses data publik yang akurat, transparan, dan seragam dari berbagai instansi pemerintahan secara cepat.
5. Proyek ini berkontribusi terhadap pencapaian tujuan pembangunan berkelanjutan *Sustainable Development Goals* (SGDs) Nomor 9 mengenai Industri, Inovasi, dan Infrastruktur melalui pembangunan infrastruktur

digital yang berkelanjutan serta efisiensi penggunaan sumber daya teknologi di Tingkat daerah Kabupaten Karimun.

6. Penelitian ini juga mendukung SGDs (*Sustainable Development Goals*) Nomor 16 tentang Perdamaian, Keadilan, dan Kelembagaan yang tangguh. Tujuannya adalah memperkuat akses publik terhadap informasi serta mendorong transparansi dan akuntabilitas pada setiap lembaga pemerintahan di Kabupaten Karimun.

1.6 Kolaborasi dan Sinergi dalam Pelaksanaan Proyek

Selama pelaksanaan proyek, kolaborasi dilakukan dengan berbagai pihak untuk memastikan solusi yang dihasilkan relevan dan berkualitas. Meskipun proyek ini dilaksanakan secara individu, terdapat sinergi yang kuat dengan berbagai pihak untuk memastikan sistem yang dikembangkan sesuai dengan kebutuhan. Berikut bentuk kolaborasi yang saya lakukan:

1. Melakukan diskusi dengan pembimbing lapangan di Dinas Komunikasi Informatika Statistik dan Persandian Kabupaten Karimun. Diskusi ini saya lakukan untuk memantau kemajuan proyek serta mendapat arahan teknis terkait standarisasi website pemerintah daerah dan kebijakan tata kelola informasi di lingkungan Kabupaten Karimun.
2. Melakukan koordinasi dan wawancara dengan tim teknis Diskominfo Kabupaten Karimun dalam hal penyiapan lingkungan pengembangan, manajemen domain atau *subdomain* untuk setiap OPD, serta memastikan kesiapan infrastruktur untuk implementasi pengembangan proyek.
3. Melakukan konsultasi dengan dosen pembimbing untuk memastikan kesesuaian metodologi penelitian, ketepatan implementasi *framework* Laravel dan Filament, serta kelengkapan dokumentasi laporan sesuai dengan standar akademik.

Bentuk kolaborasi ini menunjukkan bahwa terdapat sinergi yang kuat antara pihak akademis, instansi pemerintahan, dan pengguna akhir. Keterlibatan aktif dari pembimbing lapangan dan dosen menjadi bagian penting dalam menjamin keberhasilan dan kualitas aplikasi yang dikembangkan.

BAB II TINJAUAN PUSTAKA

2.1 Sistem Terkait

Pada bagian ini disajikan beberapa solusi atau sistem yang berkaitan dengan pengelolaan website. Tinjauan ini bertujuan untuk memberikan gambaran mengenai fitur, teknologi, serta kelebihan dan keterbatasan dari sistem yang telah ada. Dengan adanya tinjauan ini, diharapkan dapat menjadi dasar dalam pengembangan sistem yang diusulkan agar lebih sesuai dengan kebutuhan pengguna.

1. Sistem CMS *Website* Pondok Pesantren Raudatul Musthofa (Fathurrohmi et al., 2025) sistem ini menyediakan fitur pengelolaan artikel, kategori berita, profil instansi, registrasi penulis, serta manajemen komentar. Keunggulan dari penelitian tersebut terletak pada dokumentasi yang lengkap serta penerapan metode *Pragmatic Extreme Programming* (PXP) yang adaptif terhadap kebutuhan pengguna. Namun, sistem ini dikembangkan secara khusus untuk satu institusi, yaitu pondok pesantren, sehingga belum ditujukan untuk mengelola beberapa institusi dalam satu platform terpusat.
2. Sistem E-katalog *Multi-Tenant* KatalogQu (Abdillah, 2025) menggunakan Laravel, MySQL, dan *library* QR Code. Aplikasi ini memiliki fitur registrasi *tenant*, manajemen produk, pembuatan QR Code, dan *dashboard* analisis. Kelebihan utama sistem ini adalah penerapan arsitektur *multi-tenant* yang memungkinkan banyak penyewa menggunakan satu sistem dengan basis data terintegrasi namun data tetap terisolasi secara aman. Namun, fokus penelitian tersebut masih terbatas pada sektor komersial UMKM dan belum diterapkan untuk memfasilitasi kebutuhan pengelolaan serta publikasi informasi publik pada instansi pemerintahan daerah.
3. Pengembangan Portal PPID (Surfina dan Ma'sum, 2026) menggunakan WordPress, Elementor, dan MySQL. Fitur yang dibangun mencakup autentikasi admin, manajemen dokumen PPID, pengunggahan laporan format PDF, konfigurasi menu navigasi, hingga monitoring permohonan informasi publik. Sistem ini memberikan kemudahan bagi administrator

non-teknis dalam memutakhirkan data tanpa keahlian pemrograman. Namun, penelitian tersebut menggunakan WordPress sebagai CMS dengan dukungan plugin, sehingga pengembangan dan kustomisasi sistem masih bergantung pada ekosistem WordPress dan komponen tambahan yang digunakan.

4. Sistem SIMPEL Larizmu (Fitrianto dan Sulistyo, 2026) menerapkan *Role Based Access Control* (RBAC) pada sistem *multi-tenant* menggunakan Laravel sebagai *framework* pengembangan dan Filament Shield untuk mengatur peran, hak akses pengguna, dan isolasi data antar-tenant. Kelebihannya terletak pada keamanan dan pengelolaan akses dalam sistem terpusat, sedangkan fokusnya masih pada pengelolaan data dan pelaporan organisasi, bukan pembangunan website publik. Penelitian ini memiliki kesamaan pada penggunaan Laravel, Filament, *multi-tenant*, dan pengelolaan hak akses. Perbedaannya, penelitian yang dikembangkan berfokus pada Web Builder untuk mengelola website masing-masing OPD dengan pemisahan data berdasarkan opd_id dalam satu basis data.

Tabel perbandingan penelitian sebelumnya dengan yang akan dilakukan dapat dilihat pada tabel 2.1

Tabel 2. 1 Perbandingan Penelitian Sebelumnya

Pengembang/ Tahun	Nama Aplikasi	Fitur Utama	Teknologi	Kelebihan	Keterbatasan
Fathurrohmi et al. (2025)	CMS Website Pondok Pesantren Raudlatul Musthofa	Manajemen artikel, kategori berita, profil, registrasi penulis, admin, dan komentar	Laravel, MySQL, bootstrap	Menggunakan metode PXP yang fleksibel dan adaptif terhadap kebutuhan pengguna serta menyediakan fitur pengelolaan konten yang cukup lengkap	Dikembangkan hanya untuk lingkungan pesantren sehingga belum ditujukan untuk pengelolaan serta publikasi informasi website instansi.
Abdilllah (2025)	Website E-Kataalog *multi-tenant* KatalogQu	Registrasi tenant, manajemen produk, pembuatan QR Code, dan dashboard analisis	Laravel, MySQL, Library QR Code	Menggunakan arsitektur *multi-tenant* sehingga satu sistem bisa digunakan banyak penyewa dengan ruang	Aplikasi hanya berfokus pada sektor komersial (E-Katalog UMKM) dan belum menyentuh kebutuhan

				data yang terisolasi	pengelolaan website mandiri pada instansi pemerintahan daerah.
Surfina & Ma'sum (2026)	Pengembangan Portal PPID	Autentikasi admin, manajemen dokumen PPID, unggah laporan pdf, pengaturan navigasi menu, dan monitoring permintaan informasi	Wordpress (CMS), Elementor, dan MySQL	Memudahkan pemutakhiran data oleh administrator tanpa keahlian pemrograman mendalam	Pengembangan menggunakan WordPress sebagai CMS sehingga kustomisasi sistem bergantung pada platform dan komponen tambahan yang digunakan
Fitrianto & Sulistyo (2026)	SIMPEL Larizmu	Manajemen data tenant, pengelolaan pengguna, pembatasan hak akses berdasarkan peran dan tenant, serta pelaporan	Laravel, filament shield, MySQL	Menerapkan *multi-tenant* dan RBAC sehingga akses pengguna dapat dibatasi sesuai peran dan tenant dalam sistem terpusat	Berfokus pada pengeloaan data dan pelaporan organisasi, bukan pada pembangunan dan pengelolaan website publik masing-masing tenant.
Penelitian ini	Web Builder OPD	Pengelolaan Profil OPD, berita, kategori berita, pengumuman, layanan, galeri, dokumen publik, konfigurasi website	Laravel, filament, MySQL, *multi-tenant, Single Database Shared Schema*	Memungkinkan Diskominfo mengelola beberapa website OPD melalui satu sistem terpusat, sedangkan masing-masing Admin OPD dapat mengelola konten website sesuai kewenangannya	Pengembangan difokuskan pada website informatif OPD dan belum mencakup integrasi dengan sistem pelayanan publik atau sistem pemerintahan lainnya.

2.2 Landasan Teori

2.2.1 Sistem Informasi dan Aplikasi Berbasis Web

Web merupakan kumpulan informasi yang saling terhubung dan dapat memuat berbagai jenis konten seperti teks, gambar, video, serta elemen interaktif. Informasi tersebut dapat diakses oleh pengguna melalui peramban web dengan menggunakan jaringan internet (Christian &

Voutama, 2024). Dalam penelitian ini, konsep berbasis web diterapkan pada aplikasi manajemen *website* Organisasi Perangkat Daerah (OPD). Penerapan berbasis web memungkinkan pengelolaan konten dan informasi setiap OPD dilakukan melalui sistem yang terintegrasi dengan infrastruktur terpusat. Dengan demikian, pengelolaan *website* OPD dapat dilakukan melalui satu sistem tanpa memerlukan pengembangan aplikasi secara terpisah untuk setiap OPD.

2.2.2 *Content Management System (CMS)*

Content Management System (CMS) merupakan platform atau perangkat lunak yang digunakan untuk mengatur dan mengelola konten pada website, serta mempermudah pengguna dalam melakukan pengelolaan dan pengembangan website tanpa harus mendalami bahasa pemrograman (Pamungkas dan Azis, 2022). Dalam penelitian ini, konsep CMS diterapkan melalui *Admin Panel* berbasis Filament yang memungkinkan administrator OPD mengelola konten website, sedangkan struktur dan pengembangan aplikasi dikelola secara terpusat oleh teknisi IT Diskominfo.

2.2.3 Laravel

Laravel merupakan *framework* aplikasi web berbasis PHP yang bersifat *open-source* dan mendukung pengembangan aplikasi web secara cepat dan mudah. Laravel juga menyediakan fitur-fitur modern yang membantu pengembang dalam membangun aplikasi serta menghasilkan kode yang lebih singkat dan mudah dipahami (Melyani et al., 2023). Dalam penelitian ini, Laravel dimanfaatkan sebagai fondasi utama untuk membangun aplikasi manajemen *Website* Organisasi Perangkat Daerah (OPD).

2.2.4 MySQL

MySQL adalah sistem manajemen basis data relasional (*Relational Database Management System*) yang menggunakan perintah standar SQL untuk menyimpan, mengakses, dan mengelola data secara efisien (Arif, 2023). MySQL mendukung pengelolaan data secara terstruktur sehingga dapat digunakan sebagai basis data dalam pengembangan aplikasi berbasis

web. Penggunaan MySQL juga memungkinkan data disimpan dan dikelola secara terpusat sesuai dengan kebutuhan sistem. Dalam penelitian ini, MySQL digunakan sebagai basis data terpusat dengan arsitektur *single database shared schema* untuk menyimpan dan mengelola data setiap OPD dalam sistem *multi-tenant*, sehingga data dari masing-masing OPD dapat dikelola berdasarkan identitas dan hak aksesnya.

2.2.5 Filament

Filament merupakan *admin panel* berbasis Laravel yang digunakan untuk membangun antarmuka administrasi aplikasi secara cepat dan terstruktur. Filament menyediakan fitur pengelolaan data melalui operasi *create, read, updata,* dan *delete* (CRUD) serta mendukung pembuatan antarmuka yang responsif, sehingga dapat mempermudah proses pengembangan dan pengelolaan data dalam aplikasi (Sudarsono dan Vebriandi, 2025). Dalam penelitian ini, Filament digunakan sebagai *admin panel* untuk Admin Diskominfo dan Admin OPD dalam mengelola data serta konten website sesuai dengan hak akses masing-masing.

2.2.6 Arsitektur *Multi-Tenant*

Menurut Athallah dan Ayunda (2026), arsitektur *Multi-Tenant* merupakan pendekatan arsitektur yang memungkinkan satu sistem digunakan oleh beberapa *tenant* dengan menerapkan isolasi data antar-*tenant*. Dalam penelitian ini, strategi yang digunakan adalah *single database shared schema*, yaitu seluruh data dari setiap Organisasi Perangkat Daerah (OPD) disimpan dalam satu database dan dibedakan secara logis menggunakan identitas unik (opd_id) pada setiap tabel terkait. Strategi ini digunakan untuk mendukung pengelolaan beberapa website OPD melalui satu sistem terpusat serta mempermudah pemeliharaan dan pembaruan struktur basis data oleh Diskominfo Kabupaten Karimun.

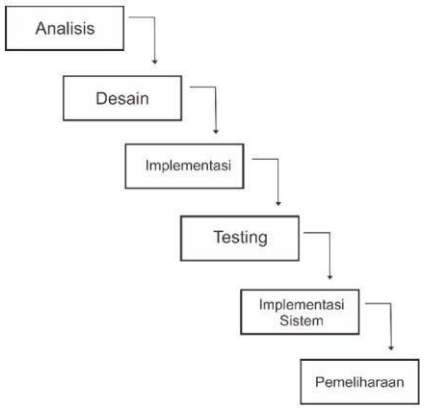
2.2.7 Organisasi Perangkat Daerah (OPD)

Organisasi Perangkat Daerah (OPD) memiliki peran penting dalam penyelenggaraan pemerintahan daerah, khususnya dalam pelaksanaan otonomi daerah, pelayanan public, dan Pembangunan wilayah. Menurut

Mozin dan Abdussamad (2026), menjelaskan bahwa OPD memiliki peran strategis dalam mendukung keberhasilan pembangunan dan pelayanan publik di tingkat daerah. Efektivitas kinerja OPD dipengaruhi oleh berbagai aspek, seperti akuntabilitas publik, pengelolaan anggaran, serta pemanfaatan teknologi informasi. Dalam penelitian ini, OPD menjadi pengguna aplikasi *Web Builder* yang bertugas mengelola dan menyajikan informasi instansinya melalui website masing-masing OPD.

2.3 Metode Pengembangan Produk

Untuk merancang dan membangun aplikasi *web builder*, penulis menggunakan metode waterfall. Metode waterfall dipilih karena pendekatannya yang terstruktur dan sistematis, dengan setiap tahapannya dilakukan secara berurutan, sebelum melanjutkan ke tahap berikutnya. Dalam metode waterfall ada beberapa tahapan yang harus dilalui yaitu analisis (*analysis*), desain (*design*), implementasi (*implementation*), uji coba (*testing*), dan pemeliharaan (*maintenance*) (Supriadi et al., 2025). Yang dapat dilihat pada gambar 2.1.



Gambar 2.1 Metode Waterfall
(Sumber: Andriansah et al., 2025)

a. Analisis kebutuhan

Tahap analisis kebutuhan merupakan tahapan yang bertujuan untuk mengidentifikasi, mengumpulkan, dan mendokumentasikan seluruh spesifikasi serta fungsi yang wajib dimiliki oleh sistem. Pada penelitian ini, analisis kebutuhan dilakukan melalui wawancara dan observasi pada Diskominfo Kabupaten Karimun sebagai pihak yang mengelola website

OPD secara terpusat. Analisis difokuskan untuk mengidentifikasi permasalahan pengelolaan website OPD, kebutuhan pengelolaan konten, pembagian hak akses pengguna, serta batasan sistem yang akan dikembangkan. Hasil analisis digunakan sebagai dasar dalam menentukan kebutuhan fungsional dan nonfungsional aplikasi *Web Builder*.

b. Desain

Berdasarkan analisis kebutuhan, dilakukan perancangan struktur dan alur sistem dari aplikasi *Web Builder.* Tahapan desain ini meliputi pembuatan diagram alir (flowchart), perancangan struktur tabel pada basis data MySQL melalui Entity Relationship Diagram (ERD), serta pembuatan rancangan antarmuka aplikasi menggunakan wireframe.

c. Implementasi

Pada tahap implementasi, rancangan sistem diterjemahkan ke dalam kode program. Pada penelitian ini Aplikasi dibangun menggunakan *framework* Laravel dengan MySQL sebagai basis data dan Filament sebagai *admin panel*. Aplikasi ini menerapkan pendekatan *single database shared schema*, yaitu seluruh data *tenant* disimpan dalam satu basis data dan diisolasi berdasarkan identitas *tenant.* Pendekatan ini digunakan untuk mendukung pengelolaan beberapa website OPD secara terpusat dalam satu aplikasi.

d. Pengujian

Tahap pengujian sistem dilakukan untuk memastikan bahwa fungsi dan fitur yang telah diimplementasikan berjalan sesuai dengan kebutuhan yang telah ditentukan. Pada penelitian ini, pengujian dilakukan menggunakan metode *User Acceptance Testing* (UAT). UAT digunakan untuk mengetahui penerimaan pengguna terhadap sistem berdasarkan fungsi dan fitur yang dikembangkan. Pengujian melibatkan pengguna sistem, yaitu *Super Admin* Diskominfo dan *Admin* OPD, sebagai pihak yang menggunakan aplikasi dalam proses pengelolaan website.

e. Pemeliharaan

Tahap pemeliharaan dilakukan setelah aplikasi diimplementasikan dan digunakan pada lingkungan server. Pada penelitian ini, tahap

pemeliharaan difokuskan pada pengawasan infrastruktur server terpusat pada Diskominfo, pemeliharaan konsistensi data dengan strategi *single database shared schema*, pembaruan *framework* Laravel dan *admin panel* Filament secara berkala, serta optimalisasi kueri basis data untuk memastikan isolasi data antar-*tenant* berdasarkan atribut opd_id agar sistem tetap berjalan stabil dan berkelanjutan.

BAB III ANALISIS DAN PERANCANGAN

3.1. Analisis Kebutuhan

Sesuai dengan tahapan analisis kebutuhan pada metode Waterfall yang telah dijelaskan pada Bab II, bagian ini menjelaskan kebutuhan sistem secara menyeluruh, dimulai dari pemahaman terhadap kondisi saat ini hingga rancangan awal sistem yang akan diusulkan guna mengatasi permasalahan pengelolaan website OPD di Kabupaten Karimun.

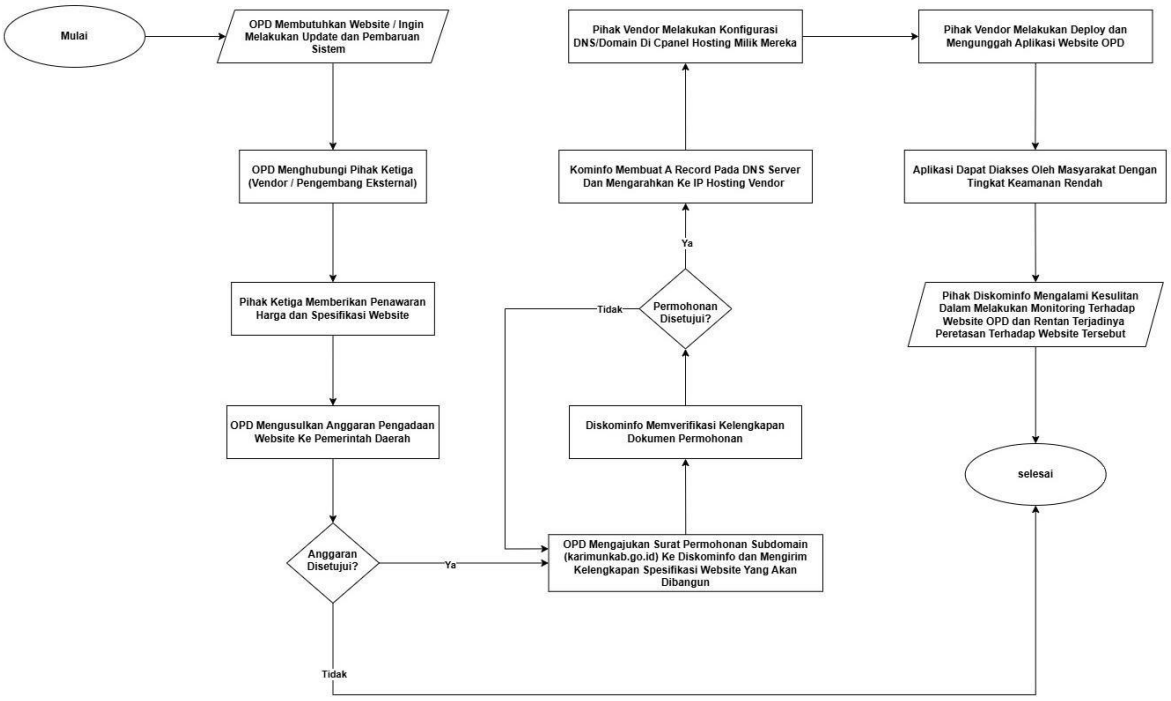
3.1.1 Proses Bisnis Berjalan

Berdasarkan hasil observasi dan wawancara yang dilakukan pada tahap analisis kebutuhan, berikut diuraikan proses bisnis pengelolaan *website* OPD yang saat ini masih dilakukan secara konvensional dan belum terintegrasi dalam satu sistem terpusat, sebelum adanya *Website Builder App*. Setiap Organisasi Perangkat Daerah (OPD) yang membutuhkan website harus mengandalkan pihak eksternal untuk membangun, memelihara, dan mengembangkan website.

1. Ketergantungan pada pihak eksternal: Setiap instansi yang membutuhkan pembaruan konten atau penyesuaian halaman harus menghubungi pengembang pihak ketiga, sehingga proses publikasi informasi menjadi lambat dan membutuhkan biaya pemeliharaan tambahan
2. *Silo system*: setiap instansi memiliki aplikasi terpisah pada server hosting yang berbeda-beda. Hal ini mengakibatkan tingginya biaya infrastruktur dan pemantauan oleh Diskominfo.
3. Keterbatasan Teknis: Tidak semua OPD memiliki staf khusus yang mempunyai keahlian pemrograman, sehingga dibutuhkan antarmuka yang lebih sederhana untuk mengelola *website* instansinya.

Secara keseluruhan, kondisi tersebut menyebabkan proses pengembangan website menjadi kurang efisien, baik dari segi waktu maupun efisiensi anggaran, karena setiap perubahan konten maupun pengembangan fitur memerlukan biaya tambahan serta prosedur koordinasi yang cukup panjang. Akibatnya, proses penyebaran informasi menjadi tidak optimal, sulit diperbarui secara cepat, serta menimbulkan ketergantungan yang tinggi terhadap pihak eksternal, sehingga proses

digitalisasi layanan pemerintahan daerah menjadi terhambat dan sulit berjalan secara efektif serta berkelanjutan.

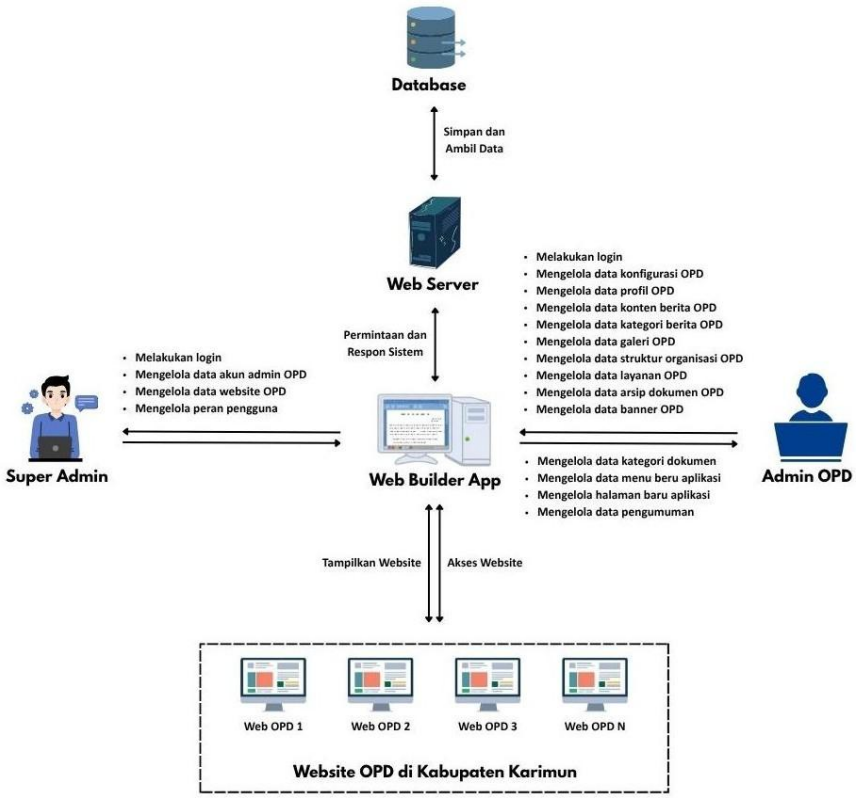


Gambar 3. 1 Flowchart Bisnis Saat Ini

3.1.2 Gambaran Umum Sistem

Aplikasi *Web Builder* yang dikembangkan merupakan sebuah sistem informasi berbasis *web* dengan arsitektur *multi-tenant* yang dirancang untuk mempermudah setiap Organisasi Perangkat Daerah (OPD) di lingkungan Pemerintah Kabupaten Karimun dalam membangun, mengelola, dan memelihara situs resmi instansi secara mandiri dan fleksibel.

Sistem ini hadir untuk menggantikan model pengelolaan lama yang bergantung pada pihak ketiga (*vendor* eksternal). Melalui platform ini, Dinas Komunikasi Informatika Statistik dan Persandian (Diskominfo) bertindak sebagai penyedia dan pengelola infrastruktur terpadu, sementara setiap instansi diberikan hak akses mandiri untuk mengelola portal informasi publiknya tanpa harus menguasai keahlian pemrograman web. Contoh gambaran umum sistem dapat dilihat pada Gambar 3.2 berikut ini.



Gambar 3. 2 Gambaran Umum Sistem

3.1.3 Aturan dan Batasan Teknis Sistem

Sebagai upaya menjaga keteraturan pengelolaan website dan keamanan data pada sistem *Web Builder*, sistem menerapkan beberapa aturan dan Batasan teknis sebagai berikut:

1. Sistem membedakan hak akses berdasarkan peran pengguna, yaitu Super Admin Diskominfo Kabupaten Karimun dan Admin OPD. Super admin memiliki kewenangan untuk mengelola data OPD dan pengguna, sedangkan Admin OPD hanya dapat mengelola konten website milik OPD yang bersangkutan.
2. Sistem menerapkan arsitektur *multi-tenant* dengan *Single Database Shared Schema.* Data setiap OPD dipisahkan berdasarkan identitas (opd_id) sehingga proses pengelolaan dan penyajian data hanya mengacu pada OPD yang sesuai.
3. Admin OPD hanya dapat mengelola konten yang menjadi bagian dari website OPD-nya, meliputi profil instansi, berita, pengumuman, layanan,

galeri, dan dokumen publik. Admin OPD tidak memiliki akses untuk mengelola data maupun konten milik OPD lain.

3.2. Perancangan

Tahap perancangan (design) merupakan fase kedua dalam metode *waterfall* yang berfungsi menerjemahkan hasil analisis kebutuhan ke dalam rancangan arsitektur, struktur data, alur fungsional, dan antarmuka pengguna secara rinci sebelum memasuki tahap implementasi kode program. Perancangan ini bertujuan untuk menghasilkan rancangan sistem *Web Builder* yang sesuai dengan kebutuhan pengelolaan website OPD, mencakup pembagian hak akses antara Admin OPD dan Super Admin Diskominfo Kabupaten Karimun, serta pemisahan data setiap OPD dalam sistem terpusat.

Rancangan sistem pada penelitian ini mencakup penentuan spesifikasi kebutuhan fungsional dan non-fungsional, perancangan diagram *usecase* beserta skenario interaksi pengguna, pemodelan alur kerja sistem menggunakan *Activity Diagram*, pemodelan basis data relasional menggunakan *Entity Relationship Diagram* (ERD), serta perancangan antarmuka visual melalui *Wireframe*.

Berikut adalah daftar definisi yang digunakan pada bagian perancangan aplikasi ini:

1. FRSA-XX: Kode Kebutuhan Fungsional Super Admin untuk pengguna tingkat Super Admin.
2. FRAO-XX: Kode Kebutuhan Fungsional untuk pengguna tingkat pengelola instansi (Admin OPD).
3. NFR-XX: Kode kebutuhan non-fungsional untuk menjamin kualitas, keamanan, dan keandalan operasional sistem.
4. UC-XX: Kode skenario *use case* yang mendefinisikan interaksi spesifik antara pengguna dan sistem.

3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang menggambarkan fungsi utama yang harus diselesaikan oleh sistem agar dapat berjalan sesuai tujuan yang

diharapkan. Berikut adalah daftar kebutuhan fungsional yang akan diimplementasikan, berdasarkan peran pengguna yang disajikan pada tabel 3.1

Tabel 3. 1 Kebutuhan Fungsional

Peran	Kode	Kebutuhan Fungsional
Super Admin	FRSA-01	Super Admin dapat membuat dan mengelola data admin OPD.
	FRSA-02	Super Admin dapat membuat dan mengelola data OPD.
	FRSA-03	Super Admin dapat mengelola peran pengguna OPD.
	FRSA-04	Super Admin dapat melakukan logout untuk mengakhiri penggunaan sistem dengan aman.
Admin OPD	FRAO-01	Admin OPD dapat menambah dan mengelola data profil OPD.
	FRAO-02	Admin OPD dapat menambah dan mengelola data berita.
	FRAO-03	Admin OPD menambah dan mengelola data kategori berita.
	FRAO-04	Admin OPD dapat menambah dan mengelola data galeri OPD.
	FRAO-05	Admin OPD dapat menambah dan mengelola data kategori dokumen OPD.
	FRAO-06	Admin OPD dapat menambah dan mengelola data layanan OPD.

	FRAO-07	Admin OPD dapat menambah dan mengelola data publikasi dokumen OPD.
	FRAO-08	Admin OPD dapat menambah dan mengelola data banner pada dashboard halaman utama website.
	FRAO-09	Admin OPD dapat menambah dan mengelola data identitas OPD pada website.
	FRAO-10	Admin OPD dapat menambah dan mengelola data Halaman Baru pada website.
	FRAO-11	Admin OPD dapat menambah dan mengelola data Tambah Menu Baru pada dashboard halaman utama website.
	FRAO-12	Admin OPD dapat melakukan logout untuk mengakhiri penggunaan sistem dengan aman.

3.2.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan yang berkaitan dengan kualitas dan batasan sistem dalam mendukung kinerja dan penggunaan aplikasi. Berikut adalah kebutuhan non-fungsional yang disajikan pada Tabel 3.2

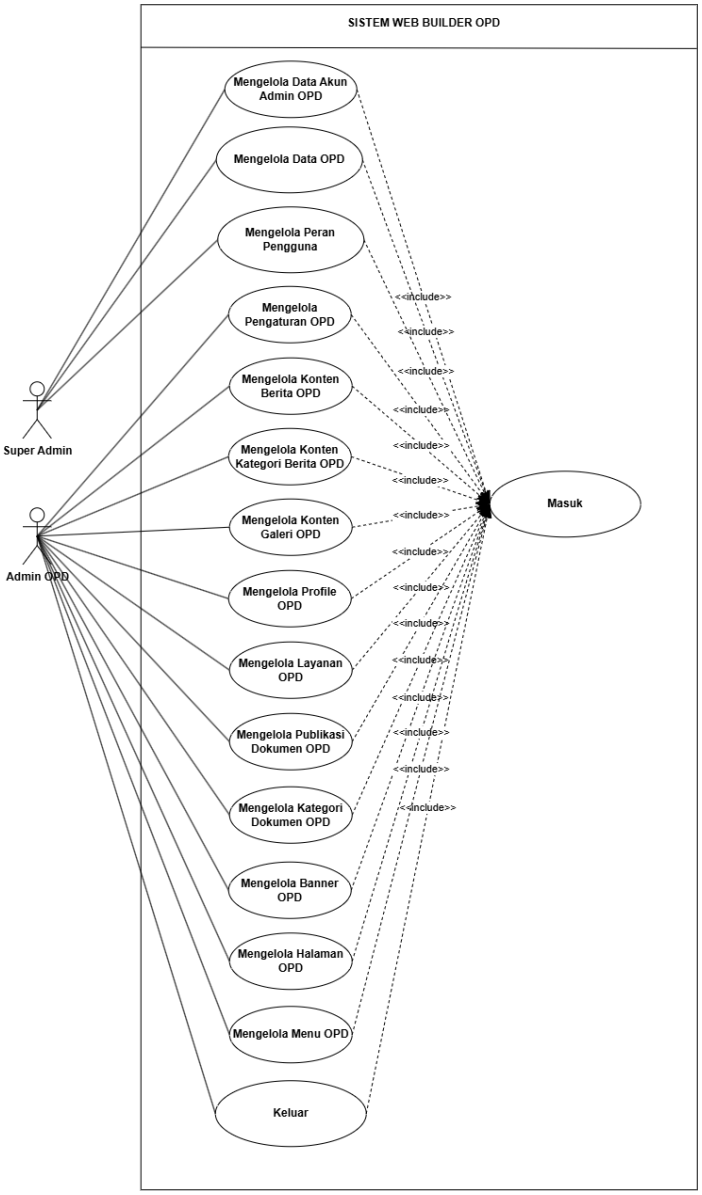
Tabel 3. 2 Kebutuhan Non-Fungsional

Nama	Kode	Kebutuhan Non-Fungsional aplikasi
Security	NFR-01	Sistem harus menerapkan autentikasi untuk membatasi akses ke halaman administrasi.

Reliability	NFR-02	Sistem harus menjaga konsistensi data antara pengguna, OPD, dan data konten yang dikelola.
Kompatibilitas	NFR-03	Aplikasi harus dapat berjalan dengan lancar dan memiliki tampilan yang responsif pada berbagai perangkat (*browser desktop* maupun *mobile*) agar informasi dapat diakses oleh masyarakat kapan saja tanpa kendala teknis.
Usabilitas	NFR-04	Antarmuka admin harus menggunakan bahasa Indonesia yang baik dan benar serta desain yang intuitif sehingga dapat dioperasikan dengan mudah oleh pengguna non-teknis (staf OPD) tanpa memerlukan keahlian pemrograman.

3.2.3 Diagram *Use Case*

Diagram *Use Case* ini menggambarkan interaksi antara pengguna (aktor) dengan sistem.



Gambar 3. 3 Usecase Diagram

Gambar 3.3 merupakan *usecase* diagram yang menggambarkan interaksi antara dua aktor utama, yaitu *Super Admin* dan *Admin OPD*, dengan Sistem *Web Builder*. *Super Admin* memiliki akses penuh terhadap pengelolaan sistem yang dapat mengelola data akun *Admin* OPD, mengelola Data OPD (*website* OPD) yang terdaftar dalam sistem, dan mengelola pembagian hak akses melalui *role* dan *permission*.

Admin OPD berperan sebagai pengelola konten pada website milik masing-masing OPD. Setelah melakukan proses masuk ke dalam sistem, *Admin OPD* dapat mengelola berbagai data dan informasi pada website OPD, meliputi profil OPD, berita, kategori berita, banner, galeri, layanan yang tersedia, data publikasi

dokumen, kategori dokumen, pengaturan OPD, pengelolaan halaman dan menu OPD untuk menyesuaikan struktur serta informasi yang ditampilkan pada website masing-masing instansi, serta banner pada website OPD.

Seluruh fungsi pengelolaan dilakukan melalui satu platform terintegrasi dengan pembatasan akses berdasarkan *role*, *permission*, dan identitas OPD pengguna. Dengan demikian, *Admin* OPD dapat mengelola data website instansinya sesuai dengan hak akses yang diberikan, sedangkan *Super Admin* dapat melakukan pengelolaan dan pengawasan secara terpusat terhadap data OPD yang terdaftar dalam sistem. Penerapan pengaturan hak akses tersebut memungkinkan kewenangan setiap pengguna disesuaikan dengan kebutuhan pengelolaan sistem. Sistem ini dirancang untuk mendukung efisiensi pengelolaan website OPD serta mengurangi ketergantungan terhadap pihak eksternal dalam proses pembaruan dan pemeliharaan informasi website OPD.

3.2.4 Skenario *Use Case*

a. *Usecase Scenario Login*

Tabel 3. 3 Skenario Use Case Login

Skenario *Use Case* 1 : Login	
Nomor	UC-01
Nama	Login
Aktor Utama	Super Admin, Admin OPD
Preconditions	Pengguna belum login ke sistem
Postconditions	Pengguna berhasil masuk ke sistem sesuai dengan data OPD masing-masing
Main Success Scenario	1. Pengguna mengakses URL login sistem 2. Pengguna memasukkan *Email* dan *Password*. 3. Pengguna menekan tombol “Login”. 4. Sistem mem-validasi kredensial pengguna. 5. Data login valid dan sesuai dengan akun yang terdaftar.

	6. Sistem mengizinkan akses dan menampilkan dashboard sesuai dengan peran pengguna masing-masing.
Alternative Flow (Extension)	4.1. Jika pengguna memasukkan email dan password yang salah sistem akan menampilkan pesan "email dan password yang dimasukkan salah". 4.2. Jika sistem memeriksa data auntentikasi dan menemukan bahwa tidak ada data email dan password yang dimasukkan, maka sistem akan menampilkan pesan "harap masukkan email dan password".

Tabel 3.3 merupakan *use case scenario* login yang digunakan oleh aktor superadmin dan Admin OPD untuk mengakses sistem sesuai dengan hak akses masing-masing pengguna.

b. *Usecase Scenario* Mengelola Data Akun Admin OPD

Tabel 3. 4 Skenario Use Case Mengelola Data Akun Admin OPD

Skenario *Use Case* 2 : Mengelola Data Akun Admin OPD	
Nomor	UC-02
Nama	Mengelola Data Akun Admin OPD
Aktor Utama	Super Admin
Preconditions	Super Admin telah login ke dalam sistem aplikasi dan berada pada halaman "Pengguna".
Postconditions	Data Admin OPD berhasil ditambahkan, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin mengakses menu halaman "Pengguna" pada sidebar. 2. Sistem menampilkan daftar Pengguna yang sudah ada pada sistem. 3. Admin menekan tombol "tambah" untuk menambah data Admin OPD.

	4. Sistem menampilkan halaman tambah pengguna. 5. Admin menginput Nama, Email, Kata sandi, OPD, dan Role. 6. Admin menekan tombol "Buat". 7. Sistem mem-verifikasi data dan memastikan bahwa *roles* yang diinput sesuai dengan akses pengguna. 8. Sistem menyimpan data akun *Admin OPD* ke tabel *Users.* 9. Admin memilih pengguna dari daftar *Users* dan menekan tombol "edit" untuk memperbarui data. 10. Admin mengubah nama email atau password, lalu menekan tombol "simpan". 11. Sistem memperbarui data pengguna pada database dan menampilkan pesan "data pengguna berhasil disimpan". 12. Admin memilih pengguna dari daftar pengguna dan menekan tombol "hapus". 13. Sistem menghapus data dari database dan menampilkan pesan "data berhasil dihapus".
Alternative Flow (Extension)	3.1. Jika sistem menemukan data yang diinput kosong dan belum diisi, sistem akan menolak proses penyimpanan data, dan menampilkan pesan validasi. 3.2. Jika sistem menemukan bahwa email pengguna sudah digunakan oleh akun lain, sistem menolak penyimpanan dan

	menampilkan pesan bahwa akun telah digunakan. 8.1. Jika terjadi kegagalan database atau sistem, sistem akan membatalkan proses penyimpanan, dan data tidak ditambahkan. Sistem akan menampilkan pesan kegagalan. 12.1. Jika sistem gagal menghapus data, data akan tetap tersimpan, dan sistem menampilkan pesan kegagalan penghapusan.

Tabel 3.4 merupakan *use case scenario* mengelola data akun Admin OPD. Superadmin dapat menambah, memperbarui dan menghapus data akun Admin OPD.

c. *Usecase Scenario Superadmin* mengelola data OPD

Tabel 3. 5 Skenario Use Case Mengelola data OPD

Skenario *Use Case* 3 : Mengelola data OPD	
Nomor	UC-03
Nama	Mengelola data OPD
Aktor Utama	Super Admin
Preconditions	Admin berada dihalaman OPD
Postconditions	Data OPD berhasil dibuat, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu “OPD” pada sidebar halaman. 2. Sistem menampilkan halaman OPD dan daftar OPD yang sudah ada. 3. Admin menekan tombol “tambah” untuk menambah data OPD baru. 4. Admin mengisi nama opd, *slug* (otomatis), dan deskripsi, lalu menekan tombol “buat”.

	5. Sistem menyimpan data OPD ke database dan menampilkan pesan “data berhasil disimpan”. 6. Admin memilih data OPD dan menekan tombol “edit”. 7. Admin mengubah data deskripsi, lalu menekan tombol “simpan”. 8. Sistem memperbarui data pada database dan menampilkan pesan “data berhasil disimpan”. 9. Admin memilih data OPD dan menekan tombol “hapus”. 10. Sistem menghapus data OPD dari database dan menampilkan pesan “data berhasil dihapus”.
Alternative Flow (Extension)	4.1. admin tidak mengisi nama atau deskripsi. Sistem menampilkan pesan “harap isi bidang ini”. 4.2. admin memasukkan nama yang sudah digunakan maka sistem akan menampilkan pesan “gagal menyimpan, data ini sudah tersedia”. 8.1. terjadi kesalahan saat menyimpan data. Sistem menampilkan pesan “gagal menyimpan data”. 9.1. Admin memilih OPD yang ingin dihapus, jika sistem gagal menghapus data, data OPD tetap tersedia, sistem menampilkan pesan kegagalan penghapusan.

Tabel 3.5 merupakan *Usecase Scenario Superadmin* mengelola data *website OPD* yang memungkinkan superadmin menambah, memperbarui dan menghapus data website OPD.

d. *Usecase Scenario Admin OPD* mengelola identitas profil OPD

Tabel 3. 6 Skenario Use Case Mengelola data profil OPD

Skenario *Use Case* 4 : Mengelola data profil OPD	
Nomor	UC-04
Nama	Mengelola data profil OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman profil OPD
Postconditions	Data profil OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu "OPD *configs*" pada sidebar halaman. 2. Sistem menampilkan halaman OPD dan data OPD konfigurasi yang ada. 3. Admin menekan tombol "tambah" untuk menambah data baru. 4. Admin mengisi nama opd, *slug* (otomatis), logo, Alamat, nomor telepon, email, Alamat URL sosial media, lalu menekan tombol "buat". 5. Sistem menyimpan data identitas OPD ke database dan menampilkan pesan "data berhasil disimpan". 6. Admin memilih data identitas dan menekan tombol "edit". 7. Admin mengubah data alamat, lalu menekan tombol "simpan". 8. Sistem memperbarui data pada database dan menampilkan pesan "data berhasil disimpan". 9. Admin memilih data identitas OPD dan menekan tombol "hapus".

	10. Sistem menghapus data OPD dari database dan menampilkan pesan "data berhasil dihapus".
Alternative Flow (Extension)	4.1. Jika data tidak lengkap sistem akan menolak untuk menyimpan data, dan memberikan pesan "lengkapi kolom berikut". 4.2. Jika data tidak sesuai format sistem akan membatalkan penyimpanan data. 7.1. Jika sistem gagal menyimpan perubahan, data lama tetap tersimpan, dan sistem menampilkan pesan kegagalan.

Tabel 3.6 merupakan *Usecase Scenario Admin OPD* mengelola data Profil *OPD* yang memungkinkan Admin menambah serta memperbarui data Profil OPD.

e. *Usecase Scenario* Admin OPD mengelola berita OPD

Tabel 3. 7 Skenario Use Case Mengelola data berita OPD

Skenario *Use Case 5* : Mengelola data berita OPD	
Nomor	UC-05
Nama	Mengelola data berita OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman berita OPD
Postconditions	Data berita OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu "Berita" pada sidebar halaman. 2. Sistem menampilkan halaman berita OPD dan data berita yang sudah ada. 3. Admin menekan tombol "tambah" untuk menambah data berita OPD baru.

	4. Admin mengisi kategori berita, judul, *slug* (otomatis), dan isi konten berita, lalu menekan tombol “buat”. 5. Sistem menyimpan data berita OPD ke database dan menampilkan pesan “data berhasil disimpan”. 6. Admin memilih data berita dan menekan tombol “edit”. 7. Admin mengubah data isi berita, lalu menekan tombol “simpan”. 8. Sistem memperbarui data pada database dan menampilkan pesan “data berhasil disimpan”. 9. Admin memilih data berita OPD dan menekan tombol “hapus”. 10. Sistem menghapus data berita OPD dari database dan menampilkan pesan “data berhasil dihapus”.
Alternative Flow (Extension)	4.1. Jika field wajib belum diisi, sistem menolak menyimpan data, dan menampilkan pesan validasi. 4.2. Jika sistem gagal menyimpan data, data berita tidak ditambahkan dan sistem menampilkan pesan kesalahan, dan Kembali ke halaman tambah berita. 9.1. Jika sistem gagal menghapus data berita, berita tetap tersedia di sistem, dan sistem menampilkan pesan kegagalan.

Tabel 3.7 merupakan skenario *use case admin OPD* mengelola berita OPD, seperti menambah, memperbarui, dan menghapus data berita yang dijelaskan melalui serangkaian langkah dalam skenario normal.

f. *Usecase Scenario* Admin OPD Mengelola Data Kategori Berita OPD

Tabel 3. 8 Skenario Use Case Mengelola data kategori berita OPD

Skenario *Use Case* 6 : Mengelola data kategori berita OPD	
Nomor	UC-06
Nama	Mengelola data kategori berita OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data kategori berita OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu "Kategori Berita" pada sidebar halaman. 2. Sistem menampilkan halaman kategori berita OPD dan data kategori yang sudah ada. 3. Admin menekan tombol "tambah" untuk menambah data kategori berita OPD baru. 4. Admin mengisi Kategori berita, OPD, dan *slug* (otomatis), lalu menekan tombol "buat". 5. Sistem menyimpan data kategori berita OPD ke database dan menampilkan pesan "data berhasil disimpan". 6. Admin memilih data kategori berita dan menekan tombol "edit". 7. Admin mengubah nama kategori berita, lalu menekan tombol "simpan". 8. Sistem memperbarui data pada database dan menampilkan pesan "data berhasil disimpan". 9. Admin memilih data kategori berita OPD dan menekan tombol "hapus".

	10. Sistem menghapus data kategori berita OPD dari database dan menampilkan pesan “data berhasil dihapus”.
Alternative Flow (Extension)	4.1. Jika nama kategori berita kosong, sistem menampilkan pesan kesalahan “harap diisi”. 4.1. Jika sistem gagal menyimpan data, data tidak ditambahkan, dan sistem menampilkan pesan kegagalan. 8.1. Jika data yang diubah telah tersedia pada sistem, sistem akan menampilkan pesan data sudah digunakan.

Tabel 3.8 merupakan skenario *use case* mengelola kategori berita OPD, yang memungkinkan Admin OPD menambah, memperbarui, dan menghapus data Kategori Berita OPD yang dijelaskan melalui serangkaian langkah dalam skenario normal.

g. *Usecase Scenario* Admin OPD mengelola Data Galeri OPD

Tabel 3. 9 Skenario Use Case Mengelola data galeri OPD

Skenario *Use Case* 7 : Mengelola data galeri OPD	
Nomor	UC-07
Nama	Mengelola data galeri OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data galeri OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu “Galeri” pada sidebar halaman. 2. Sistem menampilkan halaman galeri OPD dan data galeri yang sudah ada. 3. Admin menekan tombol “tambah” untuk menambah data galeri OPD baru.

	4. Admin mengisi judul, OPD, gambar, deskripsi, dan *slug* (otomatis), lalu menekan tombol “buat”. 5. Sistem menyimpan data galeri OPD ke database dan menampilkan pesan “data berhasil disimpan”. 6. Admin memilih data galeri dan menekan tombol “edit”. 7. Admin mengubah judul atau deskripsi galeri, lalu menekan tombol “simpan”. 8. Sistem memperbarui data pada database dan menampilkan pesan “data berhasil disimpan”. 9. Admin memilih data galeri OPD dan menekan tombol “hapus”. 10. Sistem menghapus data galeri OPD dari database dan menampilkan pesan “data berhasil dihapus”.
Alternative Flow (Extension)	4.1. Jika pengguna tidak menginput judul atau mengosongkan data galeri. Sistem menampilkan pesan “Harap isi bidang ini”. 4.2. Jika data gambar yang dimasukkan tidak memenuhi validasi, sistem akan menolak data gambar, dan sistem menampilkan pesan. 7.1. Jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan “Gagal menyimpan data”.

Tabel 3.9 merupakan skenario *use case Admin OPD* mengelola galeri OPD yang memungkinkan Admin OPD menambah, memperbarui, dan menghapus data Galeri OPD yang dijelaskan melalui serangkaian langkah dalam skenario normal.

h. *Usecase Scenario* Admin OPD mengelola Kategori Dokumen OPD

Tabel 3. 10 Skenario Use Case Mengelola data kategori dokumen OPD

Skenario *Use Case* 8 : Mengelola data kategori dokumen OPD	
Nomor	UC-08
Nama	Mengelola data kategori dokumen OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data kategori dokumen OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu "Kategori Dokumen" pada sidebar halaman admin. 2. Sistem menampilkan halaman kategori dokumen OPD dan data kategori yang sudah ada. 3. Admin menekan tombol "tambah" untuk menambah data kategori dokumen baru. 4. Admin mengisi Kategori dokumen, dan *slug* (otomatis), lalu menekan tombol "buat". 5. Sistem menyimpan data kategori dokumen ke database dan menampilkan pesan "data berhasil disimpan". 6. Admin memilih data kategori dokumen dan menekan tombol "edit". 7. Admin mengubah nama kategori berita, lalu menekan tombol "simpan". 8. Sistem memperbarui data pada database dan menampilkan pesan "data berhasil disimpan".

	9. Admin memilih data kategori dokumen dan menekan tombol “hapus”. 10. Sistem menghapus data kategori dokumen dari database dan menampilkan pesan “data berhasil dihapus”.
Alternative Flow (Extension)	4.1. jika pengguna tidak menginput judul kategori dokumen. Sistem menampilkan pesan “Harap isi bidang ini”. 7.1. jika data yang diubah sudah tersedia pada sistem, sistem menampilkan pesan, “nama kategori sudah tersedia”. 7.2. jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan “Gagal menyimpan data”.

Tabel 3.10 merupakan skenario *use case* mengelola Data kategori Dokumen OPD yang memungkinkan Admin OPD untuk menambah, memperbarui, dan menghapus data kategori dokumen yang dijelaskan melalui serangkaian langkah dalam skenario normal.

i. *Usecase Scenario* Admin OPD mengelola Layanan OPD

Tabel 3. 11 Skenario Use Case Mengelola layanan OPD

Skenario *Use Case* 9 : Mengelola layanan OPD	
Nomor	UC-09
Nama	Mengelola data layanan OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data layanan OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu “layanan” pada sidebar halaman.

	2. Sistem menampilkan halaman layanan OPD dan data layanan yang sudah ada. 3. Admin menekan tombol "tambah" untuk menambah data layanan OPD baru. 4. Admin mengisi nama layanan, OPD, dan deskripsi, lalu menekan tombol "buat". 5. Sistem menyimpan data layanan OPD ke database dan menampilkan pesan "data berhasil disimpan". 6. Admin memilih data layanan dan menekan tombol "edit". 7. Admin mengubah judul atau deskripsi layanan, lalu menekan tombol "simpan". 8. Sistem memperbarui data pada database dan menampilkan pesan "data berhasil disimpan". 9. Admin memilih data layanan OPD dan menekan tombol "hapus". 10. Sistem menghapus data layanan OPD dari database dan menampilkan pesan "data berhasil dihapus".
Alternative Flow (Extension)	4.1. jika pengguna tidak menginput salah satu data wajib. Sistem menampilkan pesan "Harap isi bidang ini". 4.2. jika sistem gagal melakukan penyimpanan data, data layanan tidak disimpan dan sistem menampilkan pesan kegagalan.

	7.1. jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan “Gagal menyimpan data”.

Tabel 3.11 merupakan skenario *use case* mengelola layanan OPD yang memungkinkan admin OPD untuk menambah, memperbarui, dan menghapus data layanan OPD yang dijelaskan melalui serangkaian langkah dalam skenario normal.

j. *Usecase Scenario* Admin OPD Mengelola Data Arsip Dokumen OPD

Tabel 3. 12 Skenario Use Case Mengelola Arsip Dokumen OPD

Skenario *Use Case* 10 : Mengelola Arsip Dokumen OPD	
Nomor	UC-010
Nama	Mengelola data Arsip Dokumen OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data Arsip Dokumen OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu “Arsip Dokumen” pada sidebar halaman. 2. Sistem menampilkan halaman Arsip Dokumen OPD dan data dokumen yang sudah ada. 3. Admin menekan tombol “tambah” untuk menambah data OPD baru. 4. Admin mengisi nama, OPD, dan dokumen yang diperlukan, lalu menekan tombol “buat”. 5. Sistem menyimpan data dokumen OPD ke database dan menampilkan pesan “data berhasil disimpan”. 6. Admin memilih data dokumen dan menekan tombol “edit”.

	7. Admin mengubah judul atau deskripsi dokumen, lalu menekan tombol "simpan". 8. Sistem memperbarui data pada database dan menampilkan pesan "data berhasil disimpan". 9. Admin memilih data dokumen OPD dan menekan tombol "hapus". 10. Sistem menghapus data dokumen OPD dari database dan menampilkan pesan "data berhasil dihapus".
Alternative Flow (Extension)	4.1. jika pengguna tidak menginput salah satu data wajib. Sistem menampilkan pesan "Harap isi bidang ini". 4.2. Jika dokumen yang dimasukkan tidak sesuai dengan maximum size yang telah ditetapkan sistem akan menampilkan pesan "ukuran terlalu besar", dan pengguna memasukkan dokumen yang telah memenuhi syarat. 7.1. jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan "Gagal menyimpan data".

Tabel 3.12 merupakan skenario *use case* mengelola arsip dokumen OPD, yang memungkinkan Admin menambah, memperbarui, dan menghapus data publikasi dokumen yang dijelaskan melalui serangkaian langkah dalam skenario normal.

k. *Usecase Scenario* Admin OPD Mengelola Banner OPD

Tabel 3. 13 Skenario Use Case Mengelola banner dashboard OPD

Skenario *Use Case* 11 : Mengelola banner dashboard OPD	
Nomor	UC-011

Nama	Mengelola banner dashboard OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data banner OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu “Banner” pada sidebar halaman. 2. Sistem menampilkan halaman banner OPD dan data banner yang sudah ada. 3. Admin menekan tombol “tambah” untuk menambah data banner baru. 4. Admin mengisi Teks pada banner, slogan/subjudul, gambar banner, serta keterangan untuk non-aktif atau aktifkan banner, dan deskripsi, lalu menekan tombol “buat”. 5. Sistem menyimpan data banner ke database dan menampilkan pesan “data berhasil disimpan”. 6. Admin memilih data banner dan menekan tombol “edit”. 7. Admin mengubah subjudul atau gambar, lalu menekan tombol “simpan”. 8. Sistem memperbarui data pada database dan menampilkan pesan “data berhasil disimpan”. 9. Admin memilih data banner dan menekan tombol “hapus”.

	10. Sistem menghapus data banner dari database dan menampilkan pesan "data berhasil dihapus".
Alternative Flow (Extension)	4.1. jika pengguna tidak menginput salah satu data wajib. Sistem menampilkan pesan "Harap isi bidang ini". 4.2. jika data gambar tidak sesuai, sistem menampilkan pesan validasi, dan pengguna memasukkan data gambar yang sesuai. 7.1. jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan "Gagal menyimpan data". 9.1. jika sistem gagal menghapus data, data tetap tersedia, dan sistem menampilkan pesan kegagalan.

Tabel 3.13 merupakan skenario *use case* mengelola Banner OPD yang memungkinkan admin OPD menambah, memperbarui, dan menghapus data Banner yang dijelaskan melalui serangkaian langkah dalam skenario normal.

l. *Usecase Scenario* Admin OPD mengelola data Halaman OPD

Tabel 3. 14 Skenario Use Case Admin OPD Mengelola data Halaman OPD

Skenario *Use Case* 12 : Mengelola data Halaman OPD	
Nomor	UC-012
Nama	Mengelola data Halaman OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data Halaman OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu "Halaman" pada sidebar halaman admin.

	2. Sistem menampilkan data halaman OPD dan data halaman yang sudah ada. 3. Admin menekan tombol "tambah" untuk menambah data halaman OPD baru. 4. Admin mengisi data halaman yang dibutuhkan untuk membuat halaman baru, dan *slug* (otomatis), lalu menekan tombol "buat". 5. Sistem menyimpan data halaman ke database dan menampilkan pesan "data berhasil disimpan". 6. Admin memilih data halaman baru dan menekan tombol "edit". 7. Admin mengubah nama halaman, lalu menekan tombol "simpan". 8. Sistem memperbarui data pada database dan menampilkan pesan "data berhasil disimpan". 9. Admin memilih data halaman dan menekan tombol "hapus". 10. Sistem menghapus data Halaman dari database dan menampilkan pesan "data berhasil dihapus".
Alternative Flow (Extension)	4.1. jika pengguna tidak menginput judul Halaman. Sistem menampilkan pesan "Harap isi bidang ini". 5.1. jika penyimpanan gagal, data halaman tidak ditambahkan, dan sistem menampilkan pesan validasi.

	7.1. jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan “Gagal menyimpan data”.

Tabel 3.14 merupakan skenario *use case* Admin OPD mengelola Data Halaman OPD yang memungkinkan admin menambah, memperbarui, dan menghapus data Halaman OPD yang dijelaskan melalui serangkaian langkah dalam skenario normal.

m. *Usecase Scenario* Admin OPD mengelola Data Menu Halaman OPD

Tabel 3. 15 Skenario Use Case Admin OPD Mengelola data Menu Halaman OPD

Skenario *Use Case* 13 : Mengelola data Menu Halaman OPD	
Nomor	UC-013
Nama	Mengelola data Menu Halaman OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data Menu Halaman OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu “Menu Halaman” pada sidebar halaman admin. 2. Sistem menampilkan data Menu halaman OPD dan data Menu halaman yang sudah ada. 3. Admin menekan tombol “tambah” untuk menambah data Menu halaman OPD baru. 4. Admin mengisi data Menu halaman yang dibutuhkan untuk membuat halaman baru, dan *slug* (otomatis), lalu menekan tombol “buat”.

	5. Sistem menyimpan data Menu halaman ke database dan menampilkan pesan "data berhasil disimpan". 6. Admin memilih data Menu halaman baru dan menekan tombol "edit". 7. Admin mengubah nama Menu halaman, lalu menekan tombol "simpan". 8. Sistem memperbarui data pada database dan menampilkan pesan "data berhasil disimpan". 9. Admin memilih data Menu halaman dan menekan tombol "hapus". 10. Sistem menghapus data Menu Halaman dari database dan menampilkan pesan "data berhasil dihapus".
Alternative Flow (Extension)	4.1. jika pengguna tidak menginput judul Menu Halaman. Sistem menampilkan pesan "Harap isi bidang ini". 4.2. jika pengguna memasukkan data yang tidak valid, sistem akan menampilkan pesan validasi, dan data tidak disimpan. 7.1. jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan "Gagal menyimpan data".

Tabel 3.15 skenario *use case* Admin OPD mengelola Data Menu Halaman OPD yang memungkinkan admin menambah, memperbarui, dan menghapus data menu halaman yang dijelaskan melalui serangkaian langkah dalam skenario normal.

n. *Usecase Scenario* Admin OPD mengelola Data Identitas OPD

Tabel 3. 16 Skenario Use Case Admin OPD Mengelola data Identitas OPD

Skenario *Use Case* 14 : Mengelola data Identitas OPD	
Nomor	UC-014
Nama	Mengelola data Identitas OPD
Aktor Utama	Admin OPD
Preconditions	Admin sudah login dan berada di halaman dashboard OPD
Postconditions	Data Identitas OPD berhasil ditambah, diperbarui, dan dihapus dari sistem.
Main Success Scenario	1. Admin memilih menu "Pengaturan OPD" pada sidebar halaman admin. 2. Sistem menampilkan data Identitas OPD. 3. Admin mengisi data Identitas OPD yang dibutuhkan untuk membuat Identitas OPD, dan *slug* (otomatis), lalu menekan tombol "buat". 4. Sistem menyimpan data Identitas OPD ke database dan menampilkan pesan "data berhasil disimpan". 5. Admin memilih data Identitas OPD dan menekan tombol "edit". 6. Admin mengubah nama Alamat OPD, lalu menekan tombol "simpan". 7. Sistem memperbarui data Identitas OPD pada database dan menampilkan pesan "data berhasil disimpan".
Alternative Flow (Extension)	4.1. jika pengguna tidak menginput Nama OPD. Sistem menampilkan pesan "Harap isi bidang ini". 4.2. jika pengguna memasukkan data yang tidak sesuai format, sistem menolak perubahan, dan menampilkan pesan validasi.

	7.1. jika terjadi kegagalan sistem atau error pada koneksi database sistem menampilkan pesan “Gagal menyimpan data”.

Tabel 3.16 merupakan skenario *use case* Admin OPD mengelola Data Identitas OPD yang memungkinkan admin menambah, memperbarui, dan menghapus data Identitas OPD yang dijelaskan melalui serangkaian langkah dalam skenario normal.

o. *Usecase Scenario* Super Admin mengelola Data Peran Pengguna

Tabel 3. 17 Skenario Usecase Super Admin Mengelola data Peran Pengguna

Skenario *Use Case* 15 : Mengelola data Peran Pengguna	
Nomor	UC-015
Nama	Mengelola data Peran Pengguna
Aktor Utama	Super Admin
Preconditions	Admin sudah login dan berada di halaman dashboard S*uper Admin.*
Postconditions	Data Peran Pengguna berhasil ditetapkan.
Main Success Scenario	1. Super Admin memilih menu “peran” pada sidebar halaman admin. 2. Sistem menampilkan data peran pengguna yang tersimpan dalam sistem. 3. Super Admin menambah nama peran yang ingin ditetapkan pada akun pengguna. 4. Super Admin memilih data dan status aksi yang dapat dilakukan oleh pengguna. 5. Super admin menekan tombol simpan. 6. Sistem melakukan validasi terhadap data peran dan hak akses yang dipilih. 7. Sistem menyimpan data pengguna.

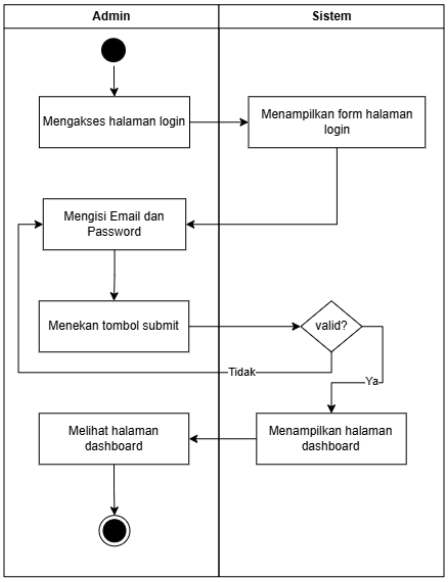
	8. Sistem menampilkan pesan data peran pengguna berhasil ditambahkan. 9. Sistem menampilkan data peran pengguna yang berhasil diperbarui.
Alternative Flow (Extension)	5.1. jika pengguna memilih tombol batal, sistem membatalkan proses perubahan, dan data tidak akan diubah atau disimpan. 6.1. jika super admin belum memilih aksi atau hak akses untuk peran tersebut, sistem menampilkan pesan peringatan, dan data peran belum disimpan. 7.1. jika terjadi kegagalan pada proses penyimpanan, sistem membatalkan proses penyimpanan, dan data peran tidak diubah. Sistem menampilkan peran validasi.

Tabel 3.17 merupakan skenario *usecase* mengelola data peran pengguna yang dilakukan oleh *Super Admin*. Melalui fitur ini, *Super Admin* dapat menambahkan, menetapkan, dan mengelola data peran (*role*) beserta hak akses (*permission*) yang dimiliki oleh setiap pengguna dalam sistem. Pengaturan ini bertujuan untuk memastikan bahwa setiap pengguna hanya dapat mengakses serta mengelola fitur yang sesuai dengan kewenangannya dalam pengelolaan aplikasi.

3.2.5 Activity diagram

Activity diagram digunakan untuk menggambarkan alur kerja (workflow) yang lebih detail dari proses-proses bisnis yang ada di dalam sistem. Untuk memodelkan sistem yang diusulkan, berikut ini adalah activity diagram yang dirancang untuk aplikasi ini.

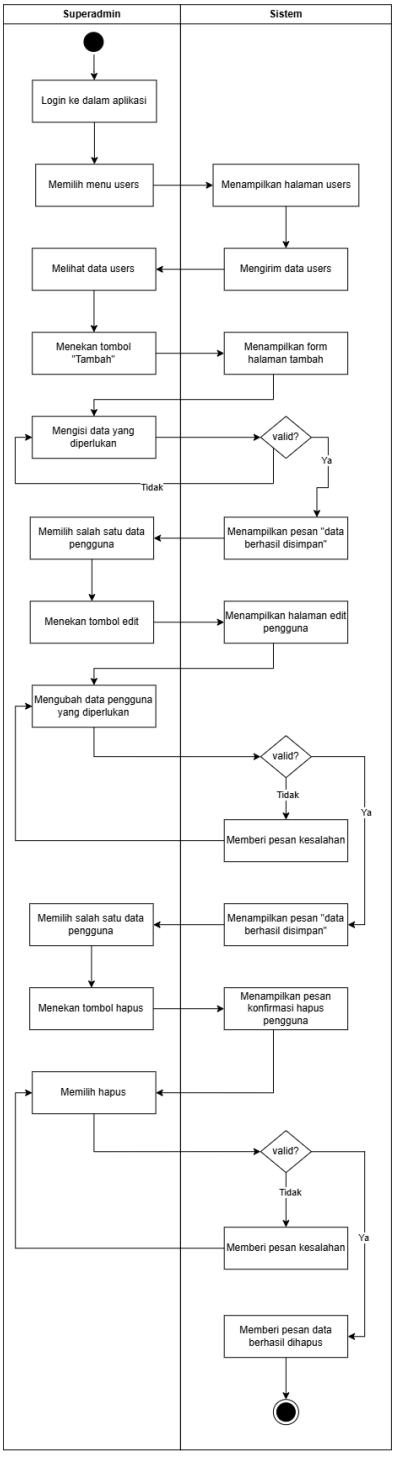
a. *Activity Diagram Login*



Gambar 3. 4 Activity Diagram Login

Activity Diagram Login menggambarkan alur pengguna saat mengakses halaman login, mengisi form data login yaitu pada kolom (email dan password), lalu sistem memverifikasi kredensial dan akses role pengguna serta menampilkan halaman utama jika berhasil.

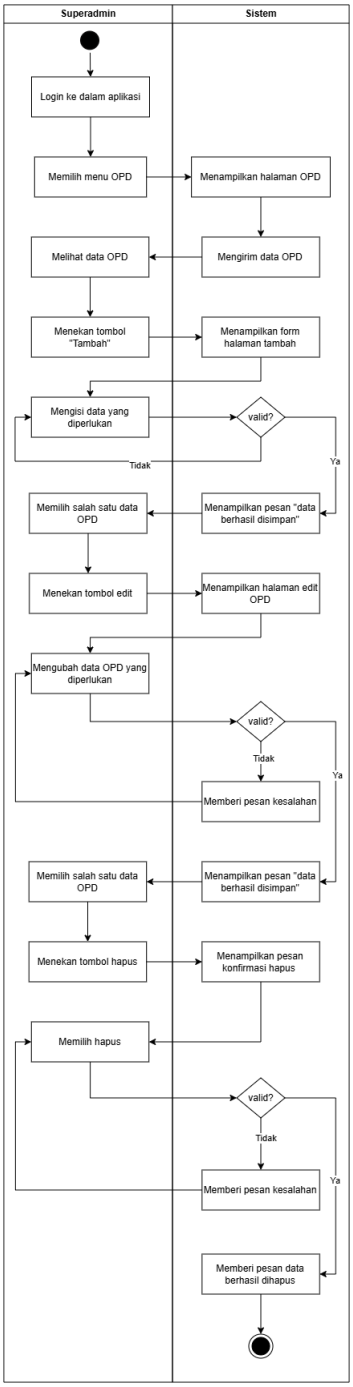
b. *Activity Diagram* Mengelola Daftar Pengguna



Gambar 3. 5 Activity Diagram Mengelola Daftar Pengguna

Gambar 3.5 menggambarkan alur pengguna mengelola daftar pengguna mulai dari membuat, memperbarui data, dan menghapus data pengguna. Setelah itu sistem memperoses perubahan yang dilakukan serta menampilkan hasilnya secara langsung.

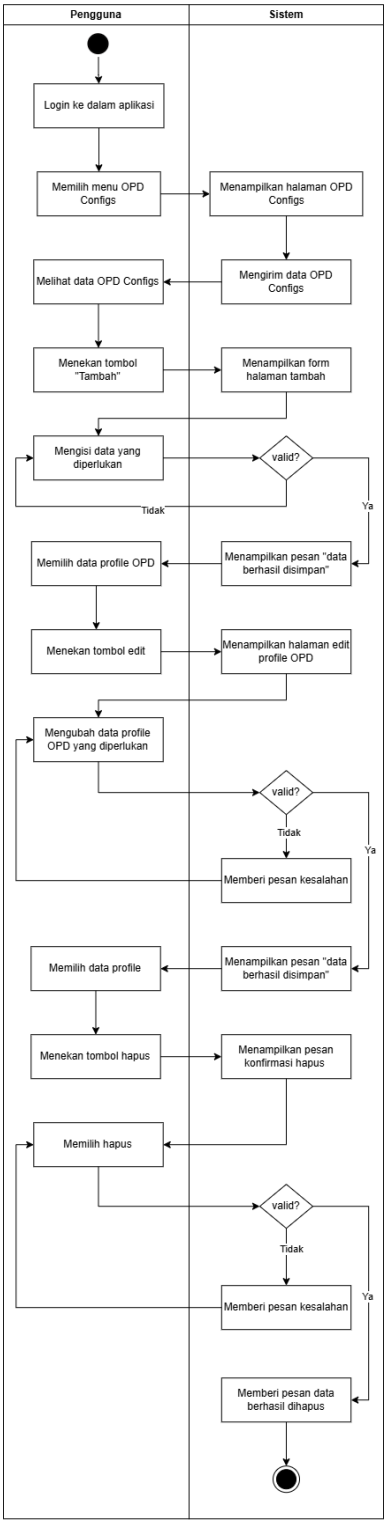
c. *Activity Diagram* Mengelola Data OPD



Gambar 3. 6 Activity Diagram Mengelola Data OPD

Gambar 3.6 menggambarkan alur pengguna saat melakukan tambah, memperbarui, dan menghapus data website opd yang dilakukan. Serta sistem menyimpan perubahan yang dilakukan dan menampilkan data yang tersedia.

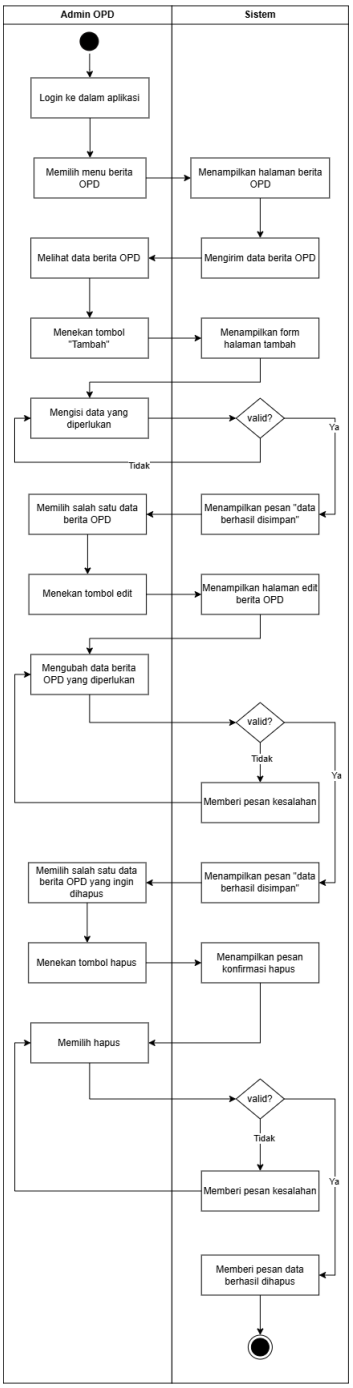
d. *Activity Diagram* Mengelola Data Pengaturan OPD



Gambar 3. 7 Activity Diagram Mengelola Data Pengaturan OPD

Pada gambar 3.7 menggambarkan alur Admin menambah, memperbarui, dan menghapus data Pengaturan OPD, dan sistem memproses data dan perubahan yang terjadi serta menampilkan informasi data Pengaturan OPD.

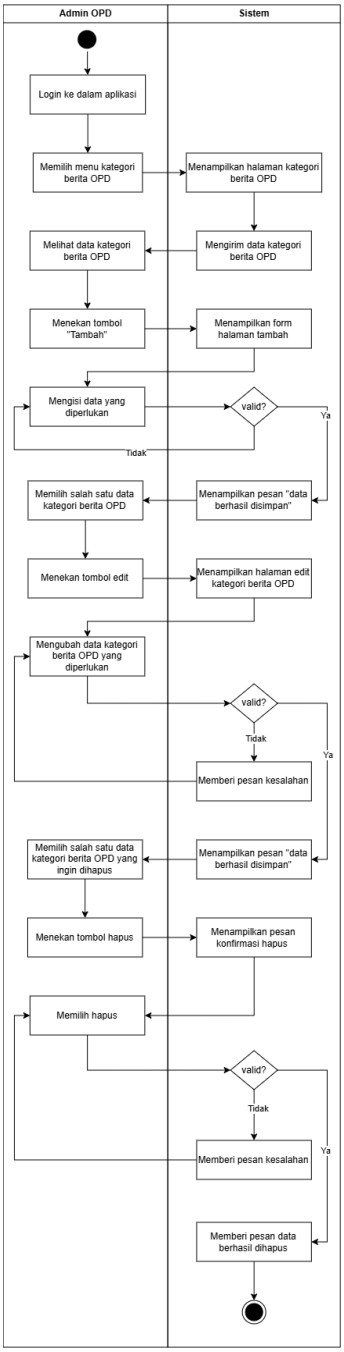
e. *Activity Diagram* Mengelola Data Berita



Gambar 3. 8 Activity Diagram Mengelola Data Berita

Gambar 3.8 mengilustrasikan alur Admin menambah, memperbarui, dan menghapus data berita OPD, lalu sistem memproses data dan perubahan yang dilakukan serta menampilkan informasi halaman berita OPD.

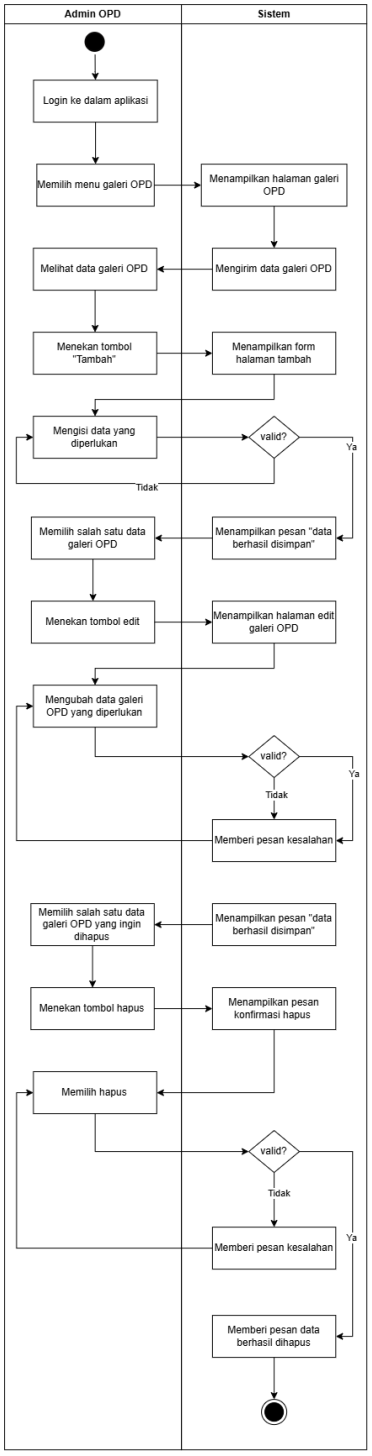
f. *Activity Diagram* Mengelola Data Kategori Berita



Gambar 3. 9Activity Diagram Mengelola Data Kategori Berita

Pada gambar 3.9 mengilustrasikan alur Admin menambah, memperbarui, dan menghapus data kategori berita OPD. Lalu sistem menyimpan perubahan yang dilakukan dan menampilkan data kategori berita yang tersedia pada halaman kategori berita.

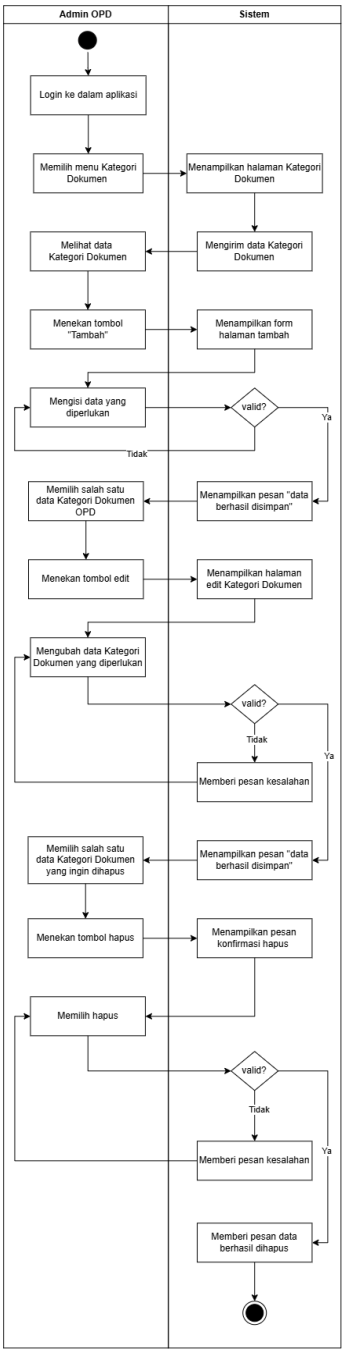
g. *Activity Diagram* Mengelola Data Galeri



Gambar 3. 10 Activity Diagram Mengelola Data Galeri

Pada gambar 3.10 mengilustrasikan alur proses Admin menambah, memperbarui, dan menghapus data galeri OPD pada halaman Galeri. Lalu sistem memperoses perubahan yang dilakukan, dan menampilkan daftar data galeri yang tersedia pada halaman Galeri.

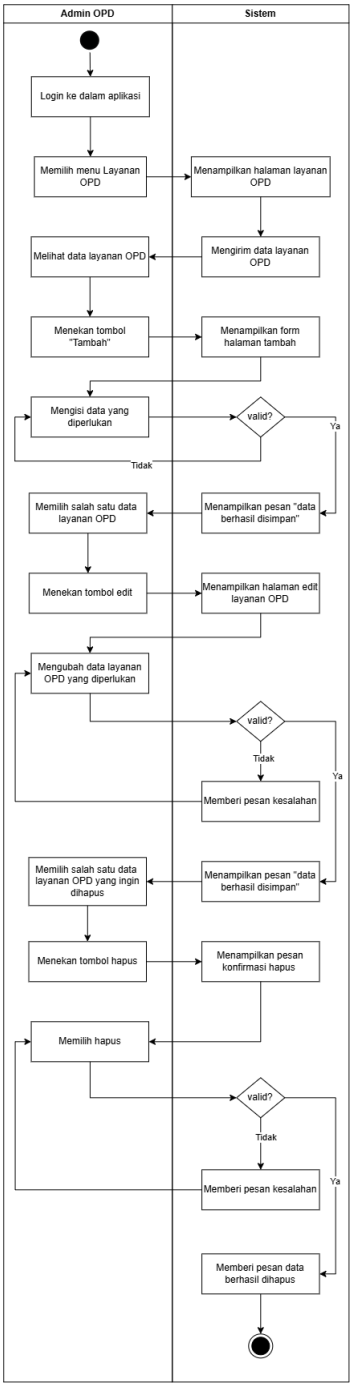
h. *Activity Diagram* Admin Mengelola Kategori Dokumen



Gambar 3. 11 Activity Diagram Mengelola Data Kategori Dokumen

Pada Gambar 3.11 mengilustrasikan proses Admin menambah, memperbarui, dan menghapus data Kategori Dokumen. Sistem memperoses perubahan yang dilakukan oleh admin, setelah itu sistem menampilkan data yang tersedia pada halaman Kategori Dokumen.

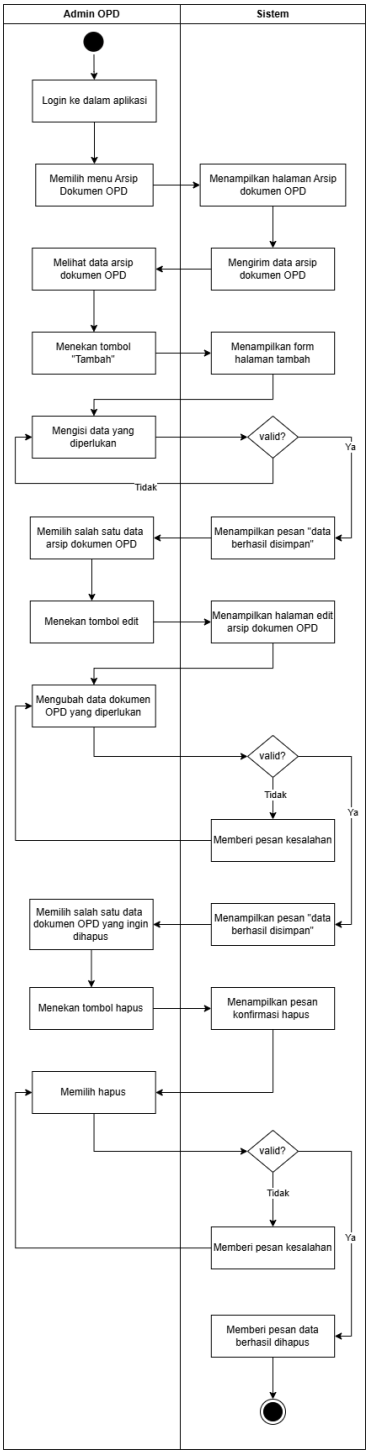
i. *Activity Diagram* Admin Mengelola Data Layanan OPD



Gambar 3. 12 Activity Diagram Mengelola Data Layanan OPD

Pada gambar 3.12 mengilustrasikan alur proses Admin menambah, memperbarui, dan menghapus data layanan OPD. Lalu sistem memperoses dan meyimpan perubahan yang dilakukan dan menampilkan data layanan yang tersedia pada halaman layanan OPD.

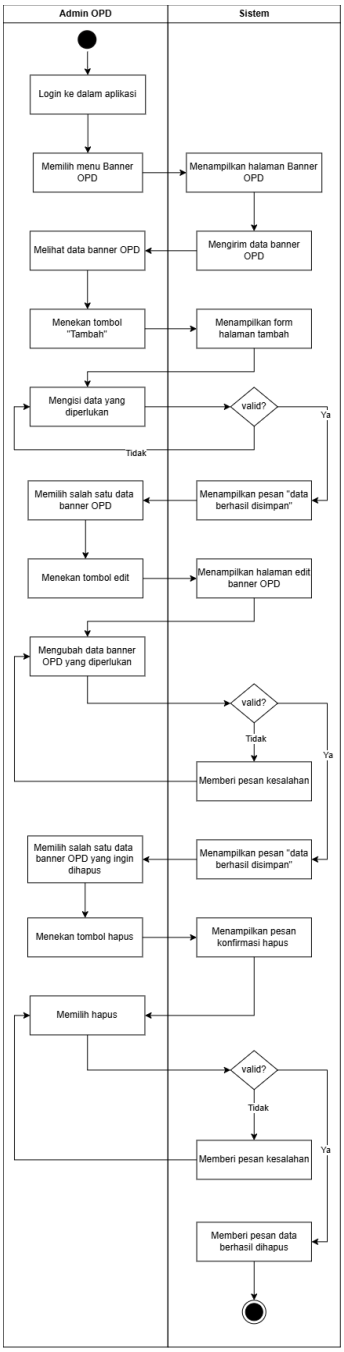
j. *Activity Diagram* Admin Mengelola Data Arsip Dokumen OPD



Gambar 3. 13 Activity Diagram Mengelola Data Arsip Dokumen

Pada gambar 3.13 mengilustrasikan alur proses Admin menambah, memperbarui, dan menghapus data arsip dokumen OPD. Sistem memperoses dan menyimpan perubahan yang dilakukan dan menampilkan daftar asrip dokumen pada halaman arsip dokumen.

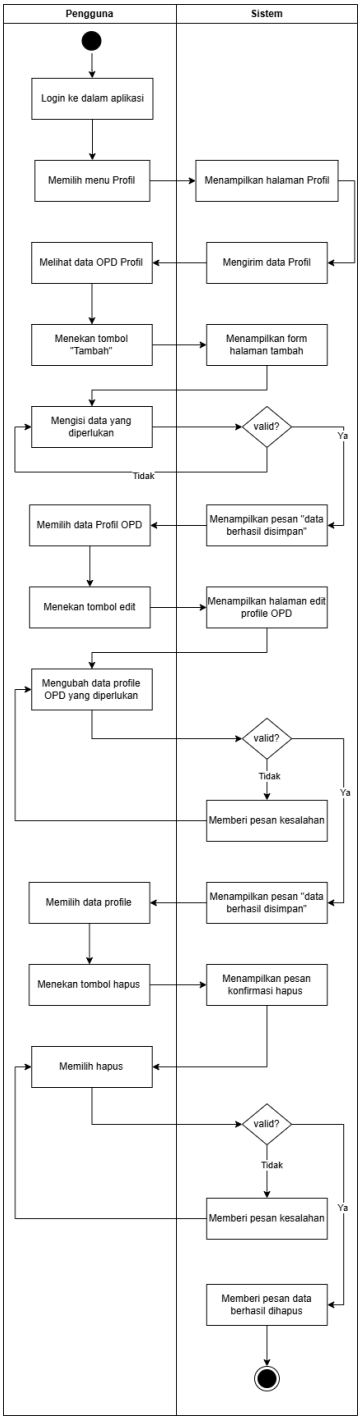
k. *Activity Diagram* Admin Mengelola Data Banner OPD



Gambar 3. 14 Activity Diagram Mengelola Data Banner

Pada gambar 3.14 mengilustrasikan alur proses Admin menambah, memperbarui, dan menghapus data banner pada halaman dashboard website. Sistem memproses dan menyimpan perubahan yang dilakukan. Lalu menampilkan data banner yang ditambahkan pada halaman pengguna.

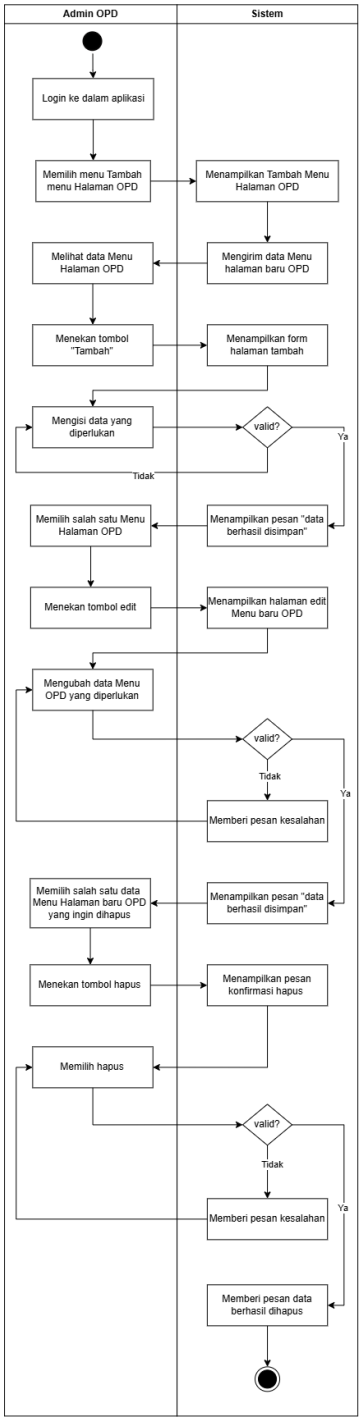
l. *Activity Diagram* Admin Mengelola Data Profil OPD



Gambar 3. 15 Activity Diagram Mengelola Data Profil OPD

Pada gambar 3.15 mengilustrasikan alur proses Admin menambah, memperbarui, dan menghapus data Profil pada website OPD. Sistem memproses dan menyimpan perubahan yang dilakukan. Lalu menampilkan data Profil yang ditambahkan pada halaman pengguna.

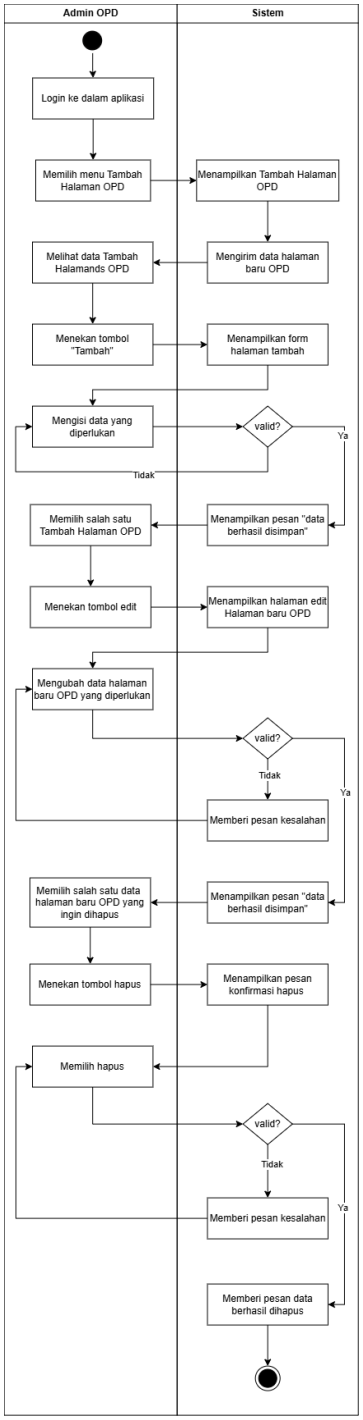
m. *Activity Diagram* Admin Mengelola Menu Halaman Baru OPD



Gambar 3. 16 Activity Diagram Mengelola Data Tambah Menu Halaman OPD

Pada gambar 3.16 mengilustrasikan alur proses Admin menambah, memperbarui, dan menghapus data Menu Halaman pada website OPD. Sistem memproses dan menyimpan perubahan yang dilakukan. Lalu menampilkan data Menu Halaman yang ditambahkan pada halaman pengguna.

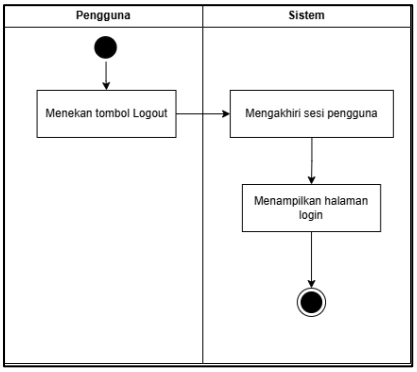
n. *Activity Diagram* Admin Mengelola Data Tambah Halaman Baru



Gambar 3. 17 Activity Diagram Mengelola Data Halaman Baru OPD

Pada gambar 3.17 mengilustrasikan alur proses Admin menambah, memperbarui, dan menghapus data Halaman Baru pada website OPD. Sistem memproses dan menyimpan perubahan yang dilakukan. Lalu menampilkan data Halaman baru yang ditambahkan pada halaman pengguna.

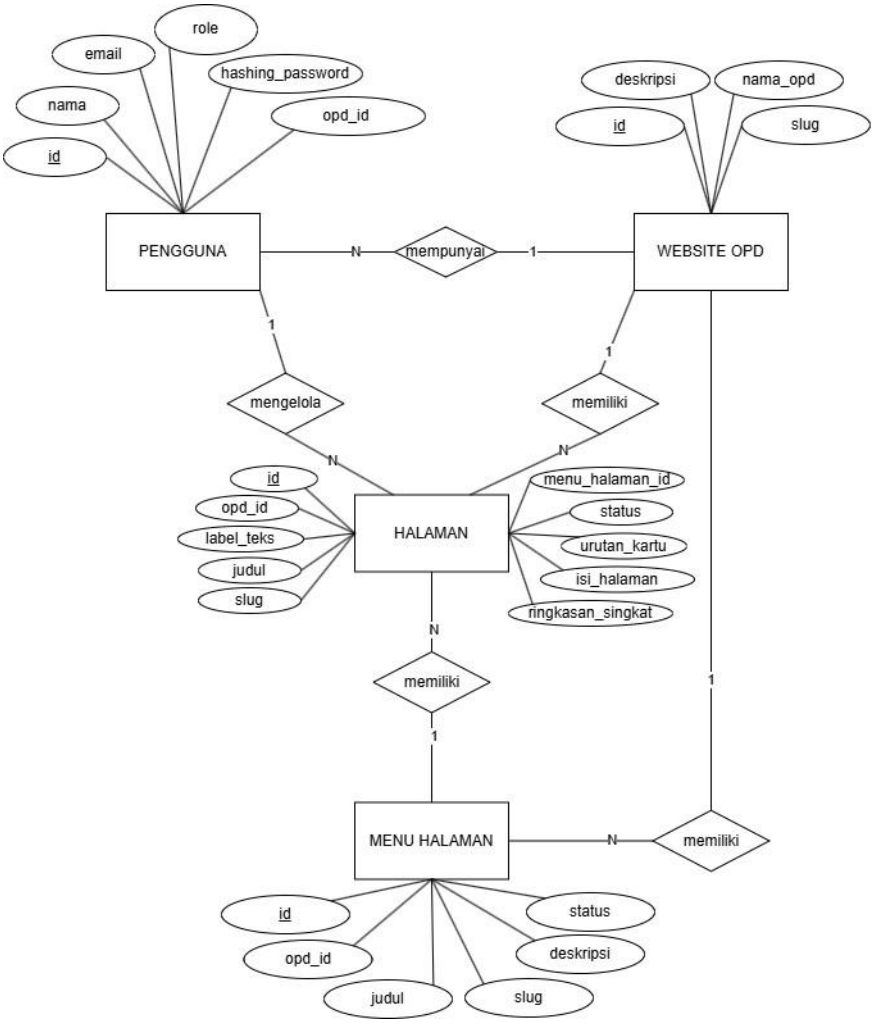
o. *Activity Diagram Logout*



Gambar 3. 18 Activity Diagram Logout

Pada Gambar 3.18 mengilustrasikan alur proses pengguna melakukan logout, lalu sistem memverifikasi sesi pengguna, dan menutup akses pengguna dari sistem.

3.2.6 ER Diagram



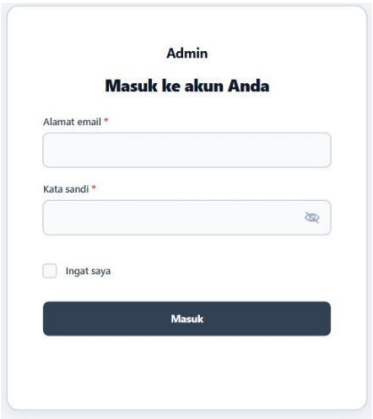
Gambar 3. 19 ER Diagram

Rancangan basis data pada aplikasi *web builder* digambarkan menggunakan *Entity Relationship Diagram* (ERD) seperti ditunjukkan pada gambar 19. ERD tersebut menggambarkan struktur entitas, atribut, serta hubungan yang digunakan dalam sistem untuk mendukung pengelolaan website Organisasi Perangkat Daerah (OPD) di Kabupaten Karimun. Entitas yang ditampilkan meliputi PENGGUNA, WEBSITE OPD, HALAMAN, dan MENU Halaman. Diagram ini menggambarkan empat entitas utama, yaitu Pengguna, Website OPD, Halaman, dan Menu Halaman, beserta atribut-atribut yang dimiliki oleh masing-masing entitas. Selain itu, diagram ini menunjukkan hubungan antarentitas dan kardinalitasnya, seperti relasi antara Pengguna dengan Website OPD dan Halaman, relasi antara Halaman dengan OPD dan Menu Halaman dalam proses pengelolaan halamannya. ER Diagram ini dirancang untuk memastikan pengelolaan data yang terstruktur, saling terintegrasi, serta mendukung proses pengelolaan website Organisasi Perangkat Daerah (OPD) di Kabupaten Karimun berbasis web.

3.2.7 Perancangan Antarmuka (Wireframe)

Perancangan antarmuka akan divisualisasikan dalam bentuk wireframe. Wireframe adalah kerangka dasar tata letak halaman web yang menunjukkan struktur website, posisi elemen, dan alur navigasi tanpa detail desain grafis.

3.2.7.1 Wireframe Halaman Login

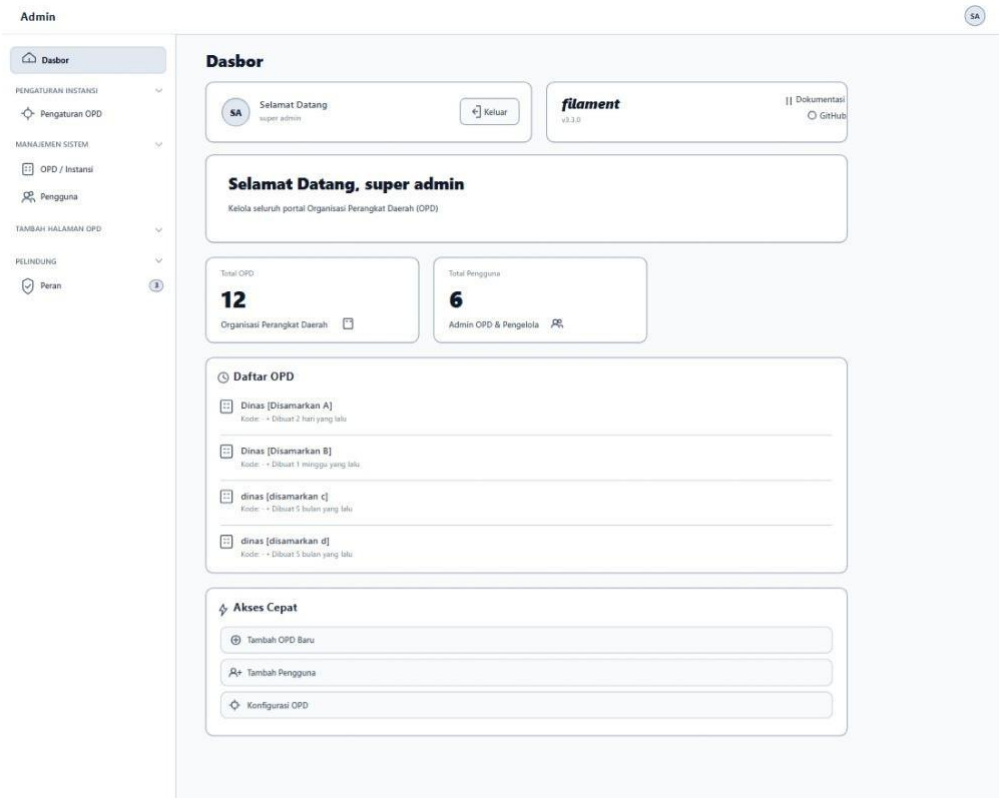


Gambar 3. 20 Wireframe Halaman Login

Gambar 3.20 merupakan Halaman Login yang berfungsi sebagai gerbang masuk pengguna ke dalam sistem. Halaman ini menyediakan formulir autentikasi yang terdiri dari kolom Email dan Password untuk memverifikasi identitas pengguna. Setelah data diisi, pengguna dapat menekan tombol

“masuk” untuk masuk ke dalam sistem dan diarahkan ke halaman dashboard sesuai hak akses masing-masing pengguna.

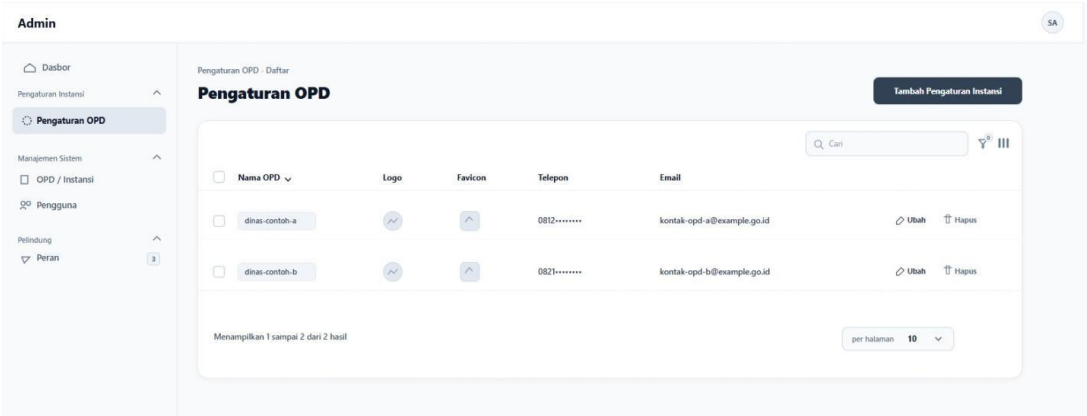
3.2.7.2 Wireframe Halaman Beranda Super Admin



Gambar 3. 21 Wireframe Halaman Dashboard Super Admin

Halaman Dasbor merupakan tampilan utama yang yang menjadi pusat kendali untuk memantau ringkasan data serta mengakses fungsi-fungsi manajemen utama secara cepat.

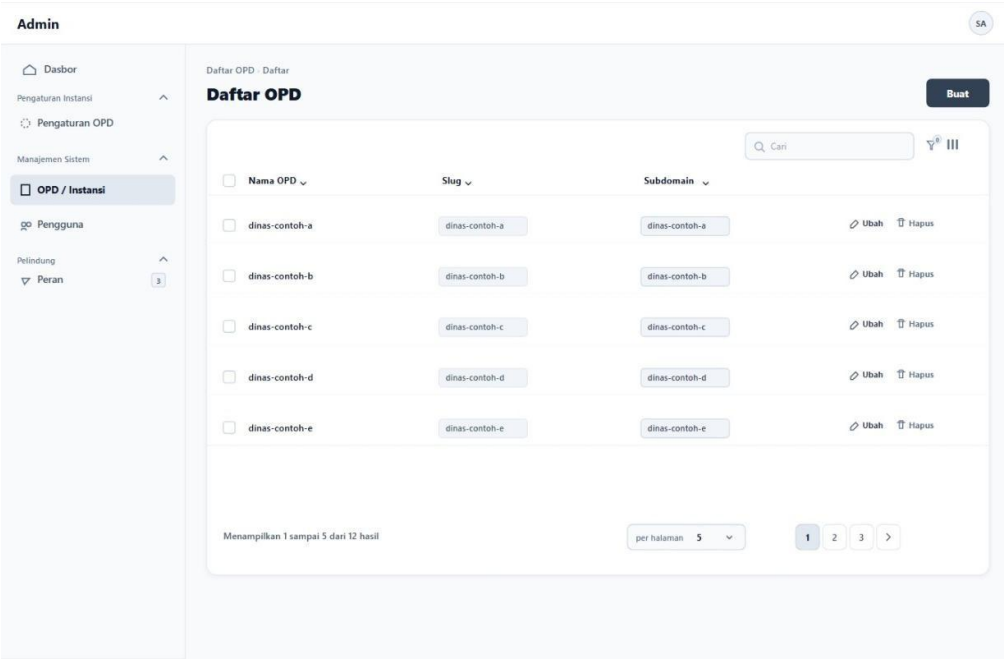
3.2.7.3 Wireframe Halaman Pengaturan OPD Super Admin



Gambar 3. 22 Wireframe Halaman Pengaturan OPD Superadmin

Halaman Pengaturan OPD berfungsi sebagai modul manajemen konfigurasi dan identitas visual website bagi setiap Organisasi Perangkat Daerah (OPD) yang terdaftar didalam sistem. Pada halaman ini, administrator dapat melihat daftar konfigurasi aktif, menambahkan konfigurasi baru, memperbarui identitas OPD, maupun menghapus data pengaturan yang tidak diperlukan.

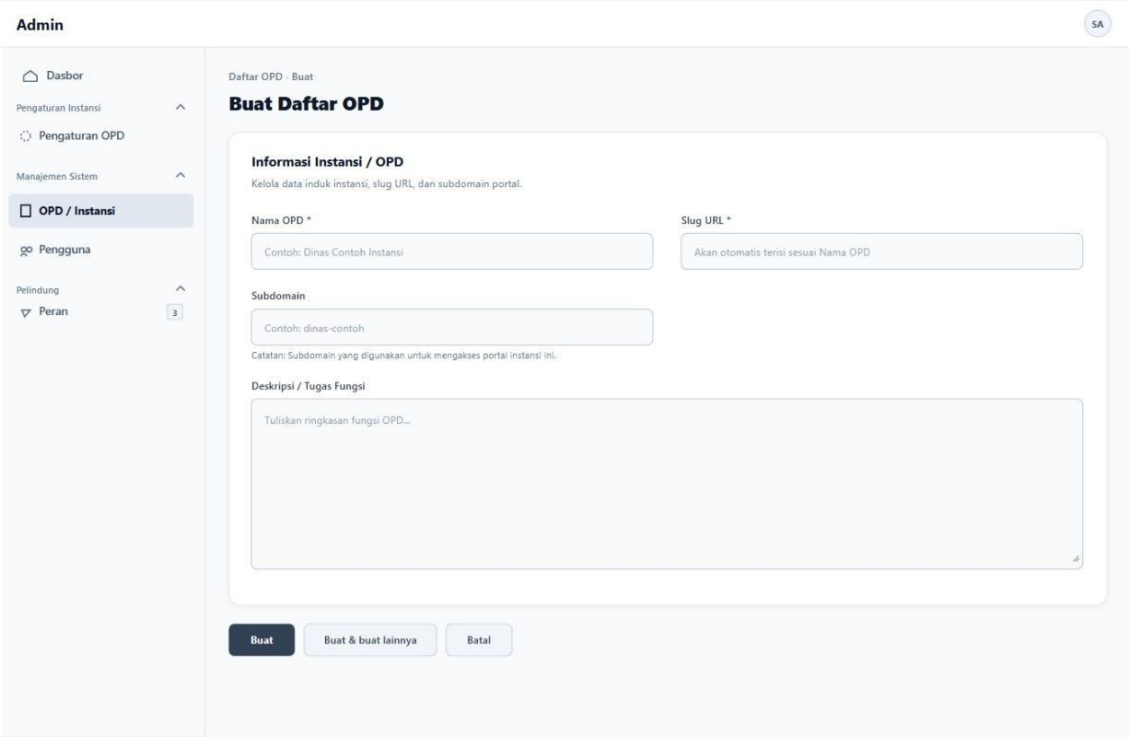
3.2.7.4 Wireframe Halaman Daftar OPD Superadmin



Gambar 3. 23 Wireframe Halaman Daftar OPD Superadmin

Rancangan antarmuka Halaman Daftar OPD berfungsi sebagai modul pengelola data master seluruh Organisasi Perangkat Daerah (OPD) yang terdaftar didalam arsitektur sistem *multi-tenant*. Melalui antarmuka ini, Super Admin memiliki otoritas penuh untuk mendaftarkan instansi baru, mengelola identitas penamaan *slug* serta memantau seluruh entitas instansi yang aktif di dalam sistem.

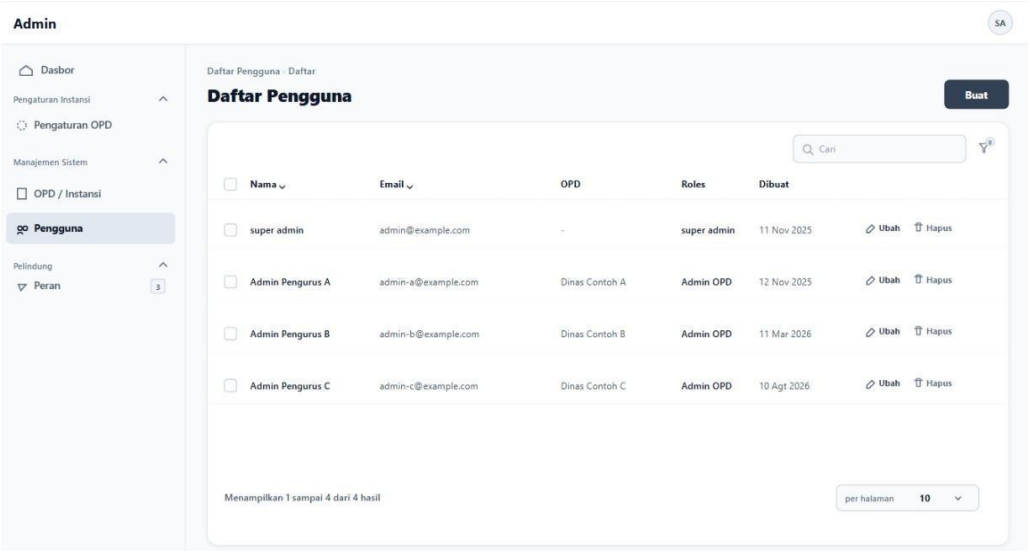
3.2.7.5 Wireframe Halaman Superadmin Tambah OPD



Gambar 3. 24 Wireframe Halaman Superadmin Tambah OPD

Halaman tambah OPD berfungsi sebagai formulir input data untuk mendaftarkan dan menginisialisasi entitas website OPD baru ke dalam sistem. Melalui formulir ini, Super Admin mengonfigurasi identitas dasar instansi, penamaan jalur URL sistem (*slug*) serta ringkasan tugas dan fungsi pokok instansi terkait.

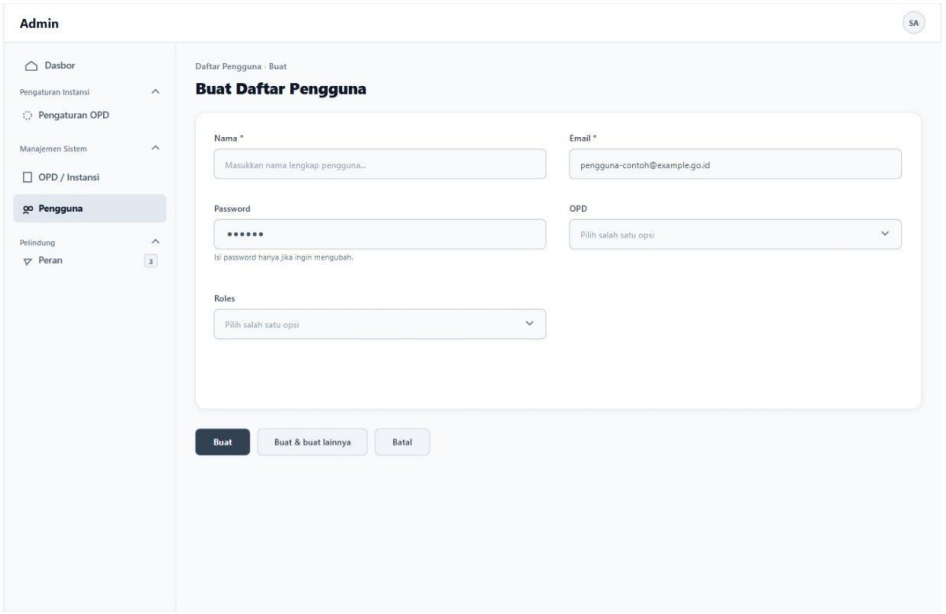
3.2.7.6 Wireframe Halaman Daftar Pengguna Superadmin



Gambar 3. 25 Wireframe Halaman Daftar Pengguna

Rancangan antarmuka halaman Daftar Pengguna berfungsi sebagai modul manajemen akun autentikasi dan otorisasi hak akses pengguna ke dalam panel administratif. Antarmuka ini memberikan peran kepada Super Admin untuk mengontrol data seluruh akun pengguna, memetakan instansi OPD yang dikelola oleh Admin OPD terkait, serta mengelola pembaruan maupun penghapusan kredensial pengguna di dalam sistem.

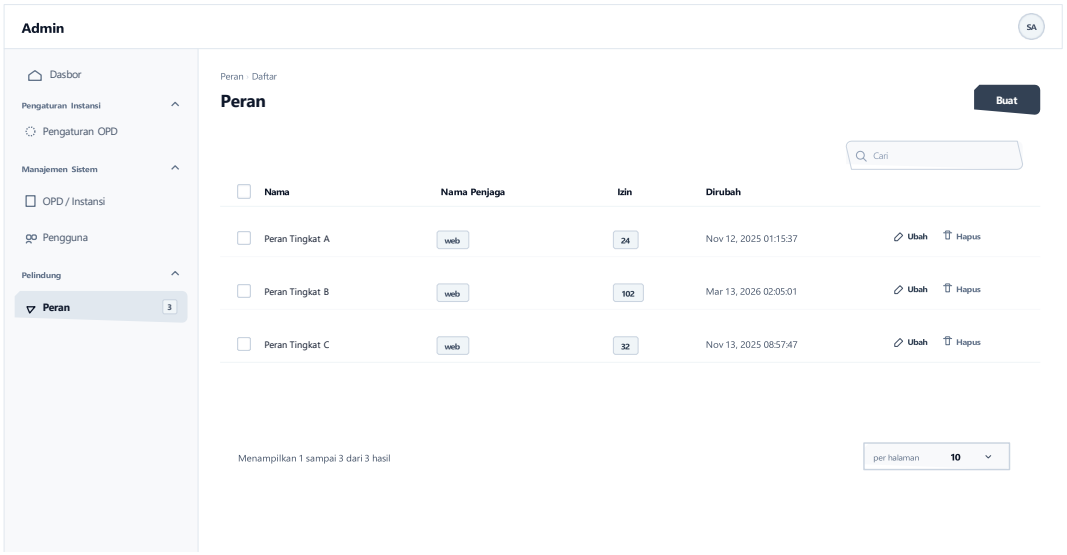
3.2.7.7 Wireframe Halaman Superadmin Tambah Pengguna



Gambar 3. 26 Wireframe Antarmuka Tambah Pengguna

Halaman antarmuka Tambah Pengguna berfungsi sebagai formulir untuk melakukan registrasi untuk membuat kredensial akun pengguna baru yang akan diberikan hak akses ke dalam panel administratif sistem. Melalui antarmuka ini, Super Admin dapat mendaftarkan akun Admin OPD, mengatur kata sandi, memetakan instansi OPD yang berada di bawah wewenang pengguna, serta menyematkan peran (*roles*) sesuai dengan tanggung jawab pengguna.

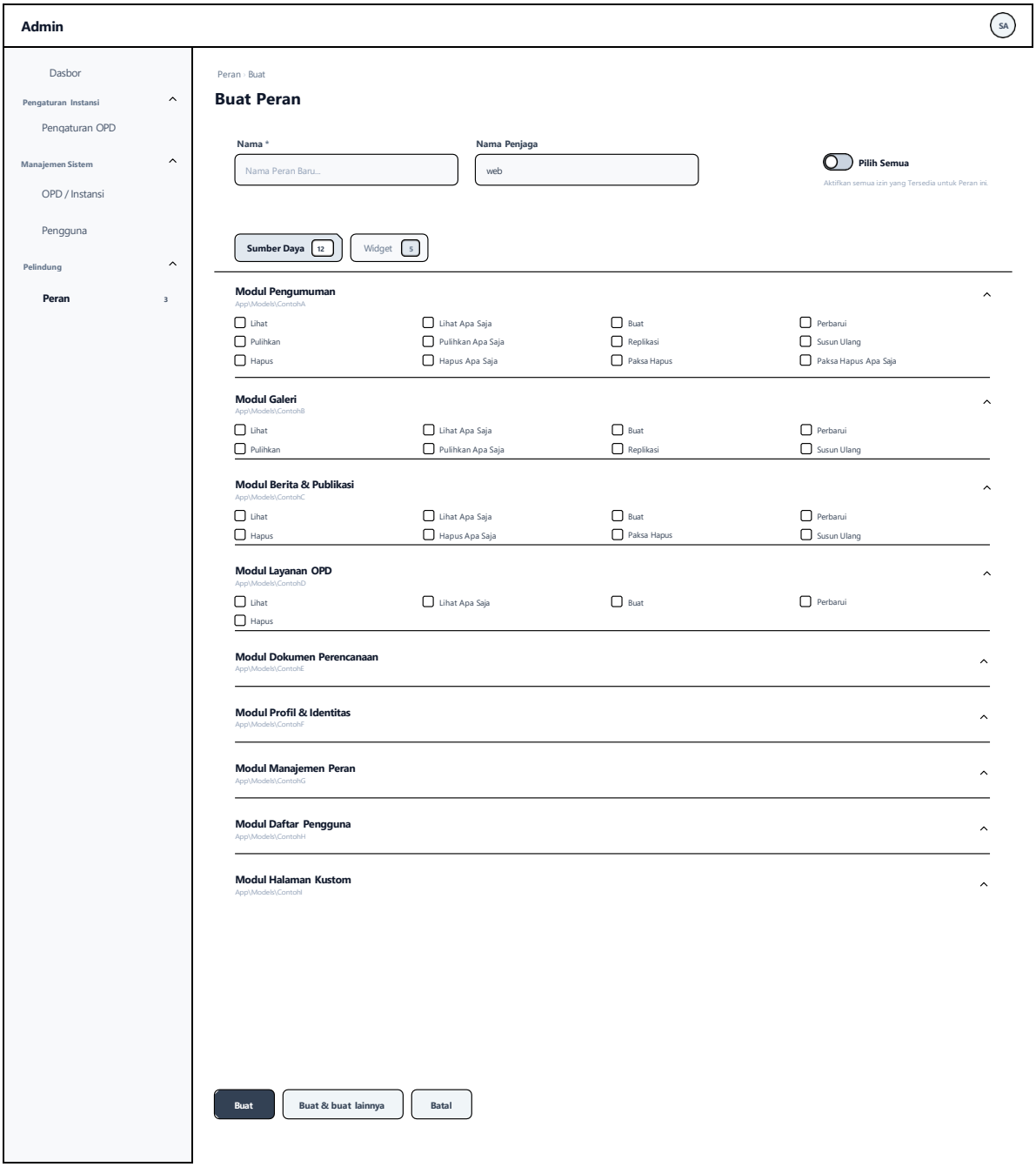
3.2.7.8 Wireframe Halaman Superadmin Tambah Peran Pengguna



Gambar 3. 27 Wireframe Antarmuka Halaman Peran Pengguna

Rancangan antarmuka Halaman Peran berfungsi sebagai manajemen pada tingkat hak akses dan perizinan sistem (*permission*). Melalui antarmuka ini, Super Admin memiliki kendali untuk mengelola daftar peran pengguna, memantau jumlah izin (*permission*) yang di tetapkan ke masing-masing peran, serta melakukan konfigurasi hak akses secara dinamis guna menjaga keamanan operasional aplikasi.

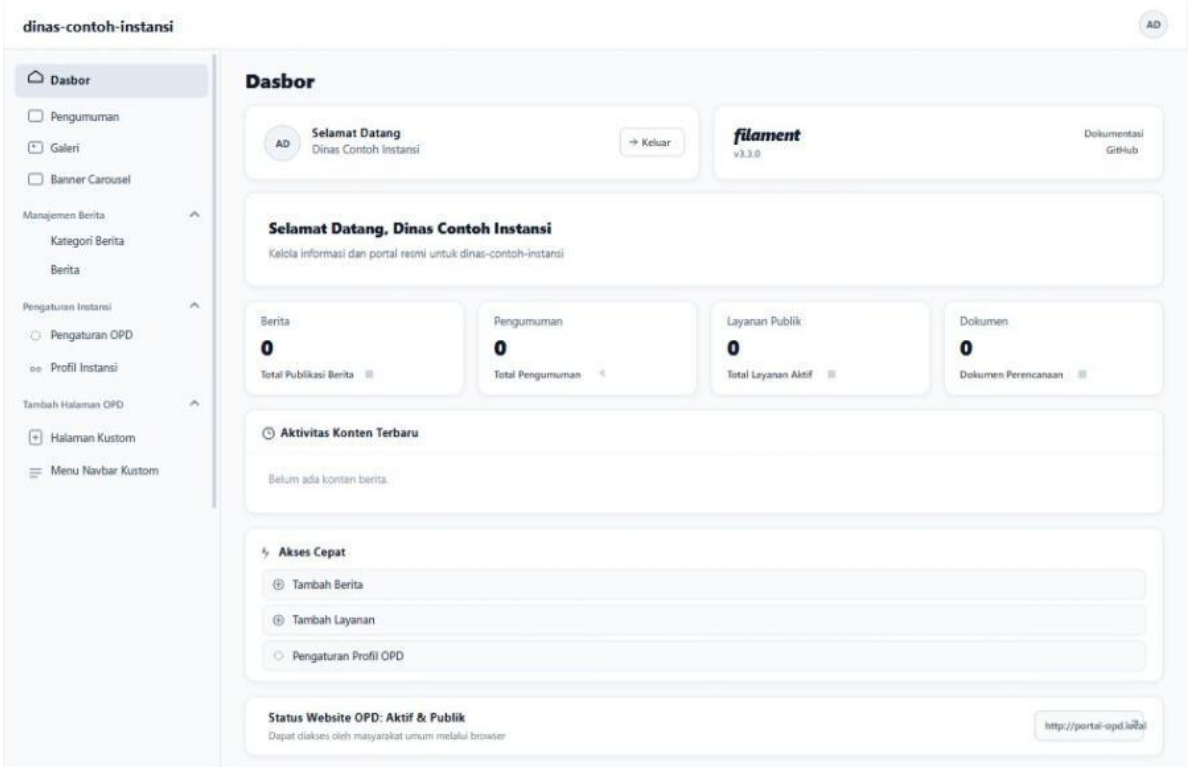
3.2.7.9 Wireframe Halaman Superadmin Tambah Peran Pengguna



Gambar 3. 28 Wireframe Antarmuka Halaman Tambah Peran Pengguna

Rancangan antarmuka Halaman Tambah Peran berfungsi sebagai konfigurasi otorisasi dan kontrol akses berbasis peran. Melalui formulir ini, Super Admin dapat mendefinikan kelompok peran baru serta mengatur secara glanular hak izin operasional terhadap setiap modul dan komponen dasbor yang ada di dalam sistem.

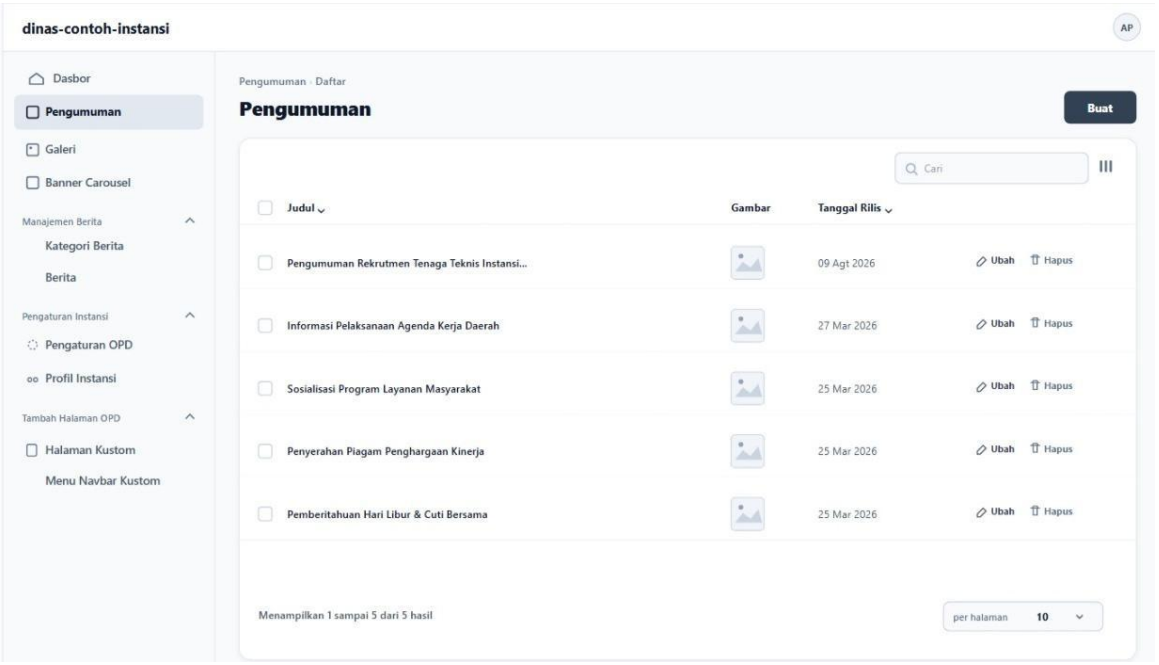
3.2.7.10 Wireframe Halaman Dashboard Admin OPD



Gambar 3. 29 Wireframe Antarmuka Halaman Dasbor Admin OPD

Antarmuka halaman Dasbor Admin OPD berfungsi sebagai pusat kendali operasional bagi administrator instansi terkait. Halaman ini mengelola data-data konten pada website OPD yang bersangkutan.

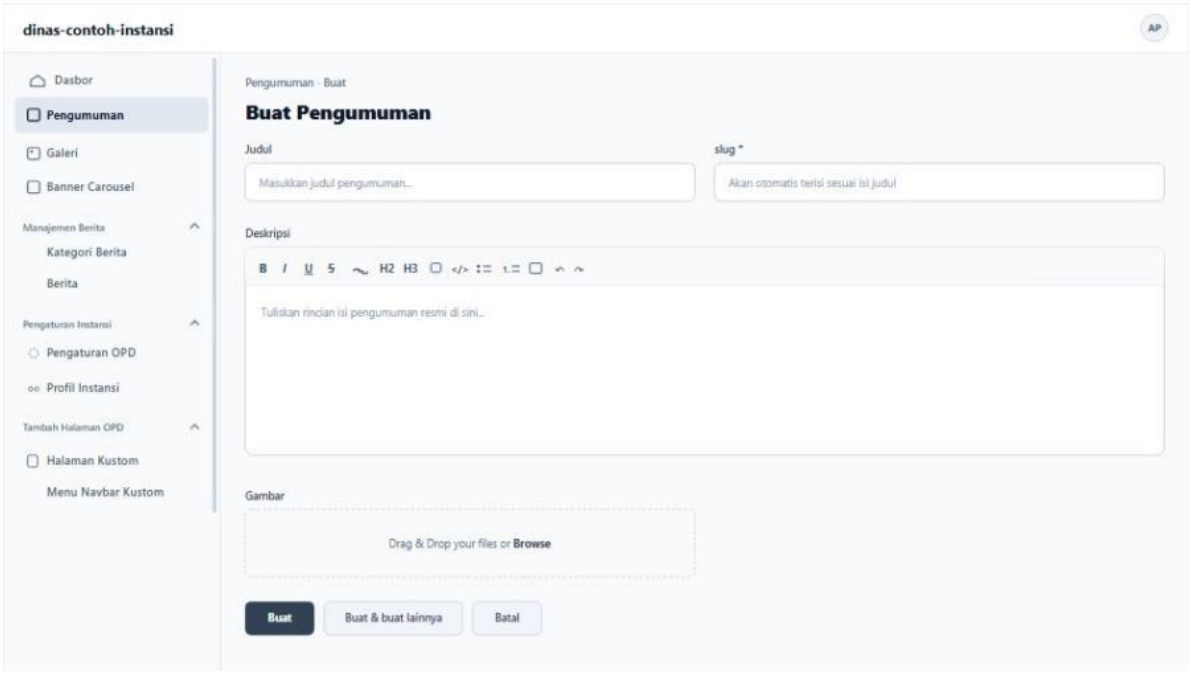
3.2.7.11 Wireframe Halaman Daftar Pengumuman Admin OPD



Gambar 3. 30 Wireframe Antarmuka Halaman Daftar Pengumuman Admin

Antarmuka Halaman Pengumuman berfungsi sebagai modul publikasi pemberitahuan penting yang diterbitkan oleh Organisasi Perangkat Daerah (OPD) kepada masyarakat umum melalui portal website.

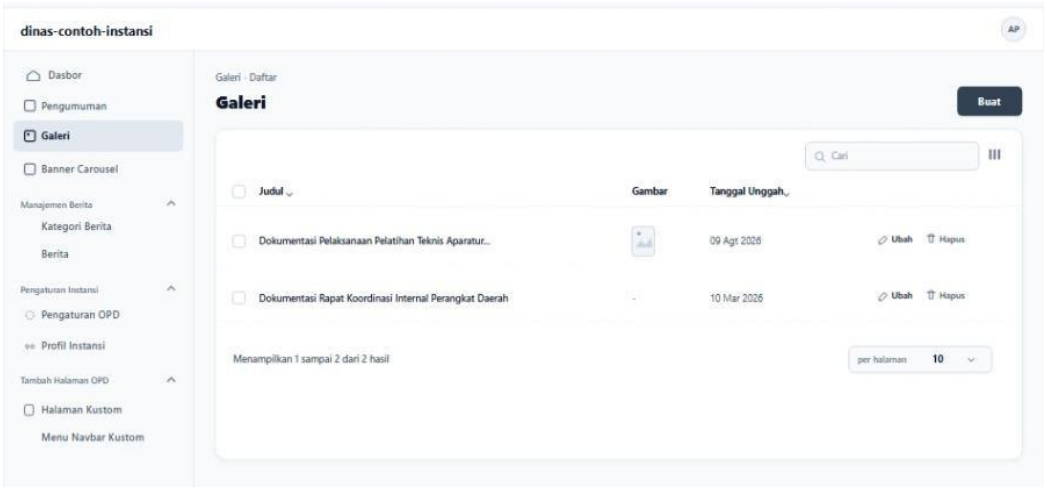
3.2.7.12 Wireframe Halaman Tambah Pengumuman Admin OPD



Gambar 3. 31 Wireframe Antarmuka Halaman Tambah Pengumuman

Antarmuka halaman Tambah Pengumuman berfungsi sebagai formulir untuk menerbitkan informasi atau pemberitahuan bagi administrator OPD. halaman ini berisi penginputan judul pengumuman, pembuatan *slug* tautan otomatis, penyusunan narasi dengan format teks, serta pengunggahan gambar.

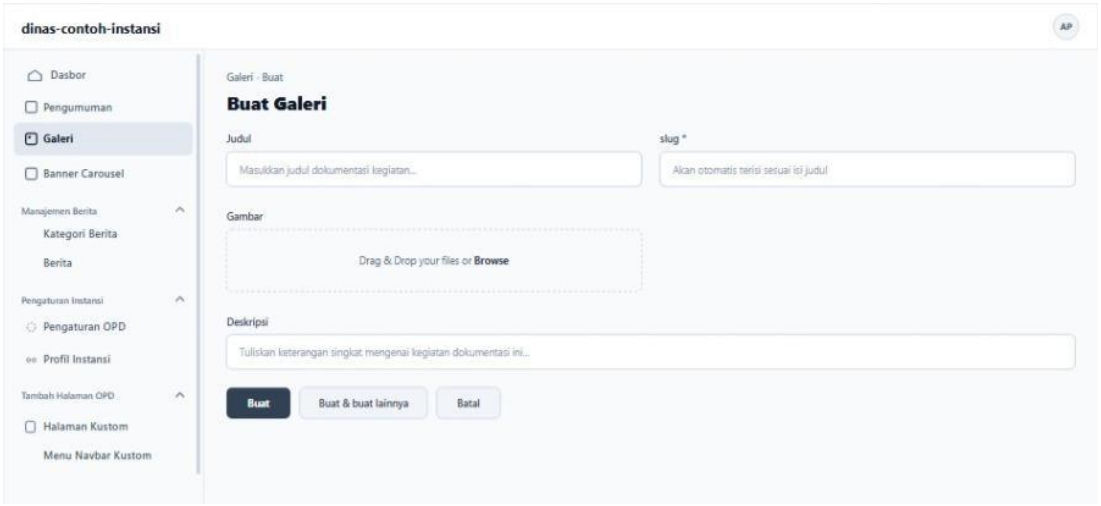
3.2.7.13 Wireframe Halaman Galeri Admin OPD



Gambar 3. 32 Wireframe Halaman Daftar Galeri Admin OPD

Antarmuka halaman Galeri berfungsi sebagai modul pengelolaan dokumentasi visual kegiatan, liputan agenda kerja, dan foto kegiatan yang dipublikasikan oleh Organisasi Perangkat Daerah (OPD) kepada masyarakat luas melalui portal website. Data yang disajikan terisolasi secara otomatis berdasarkan identitas instansi, sehingga administrator hanya dapat mengelola album dokumentasi milik instansinya sendiri.

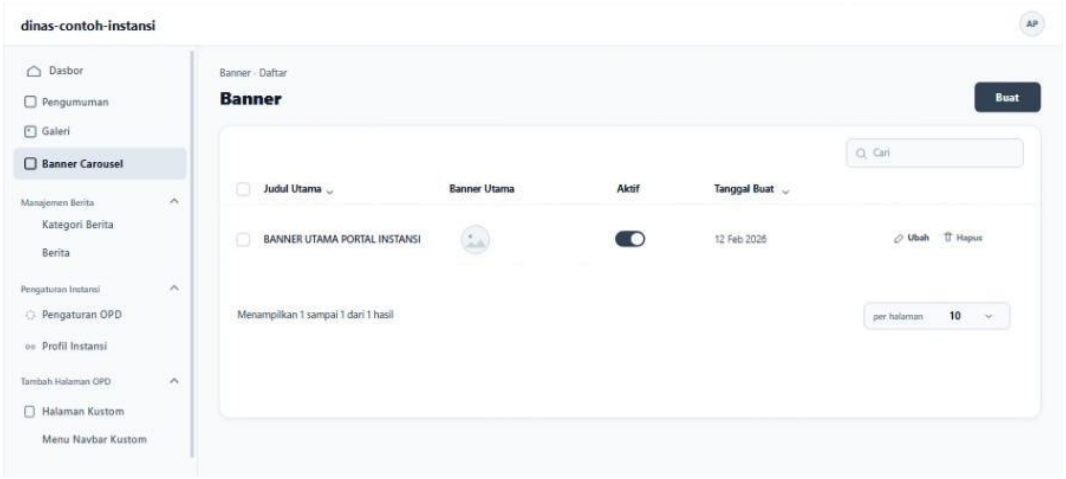
3.2.7.14 Wireframe Halaman Tambah Galeri Admin OPD



Gambar 3. 33 Wireframe Halaman Admin OPD Tambah Data Galeri

Antarmuka Halaman Tambah Galeri berfungsi sebagai formulir untuk mengunggah dan mempublikasikan dokumentasi visual kegiatan instansi baru ke portal publik OPD. formulir ini dapat digunakan untuk menambah judul kegiatan, gambar, dan penulisan deskripsi penjelasan yang bersifat informatif.

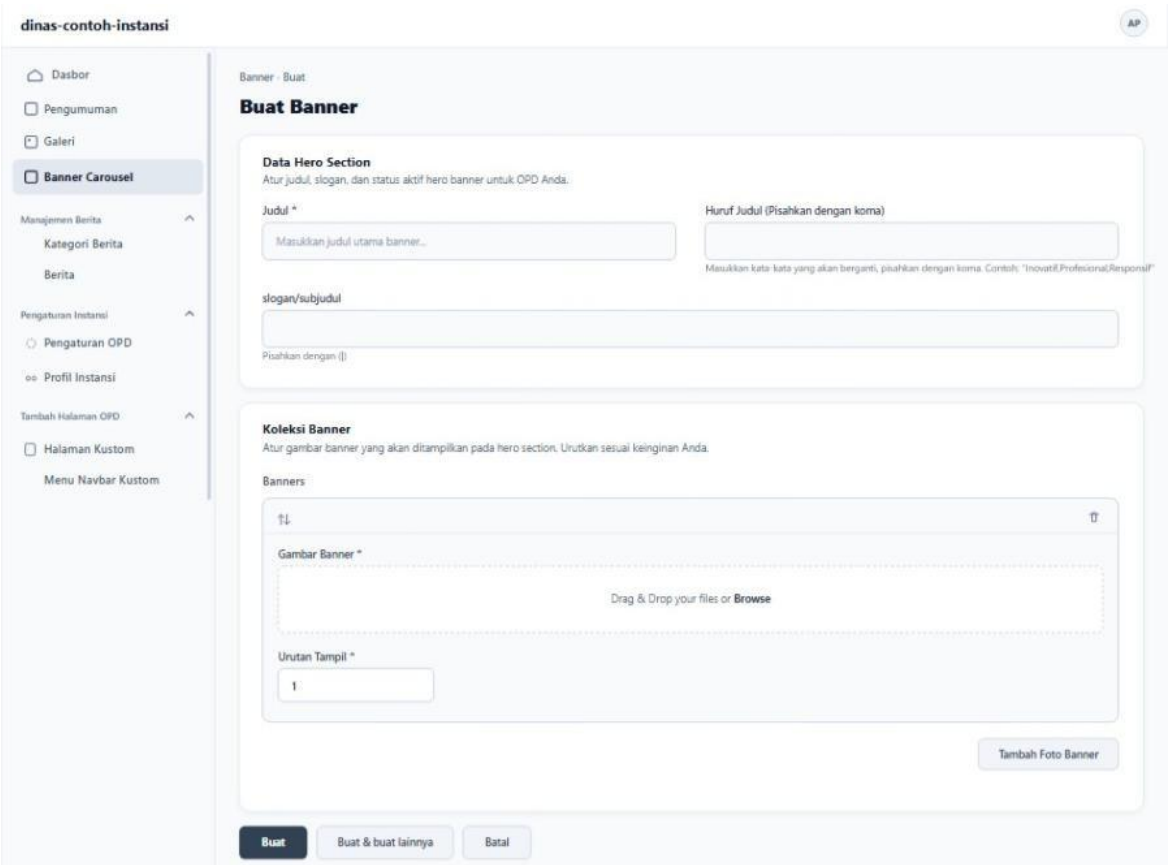
3.2.7.15 Wireframe Halaman Banner Admin OPD



Gambar 3. 34 Wireframe Halaman Data Banner Admin OPD

Antarmuka halaman Banner berfungsi sebagai modul pengelolaan spanduk visual yang ditampilkan pada bagian teratas beranda portal publik OPD. Modul ini memberikan akses kepada admin untuk mempublikasikan data seperti promosi program unggulan, maklumat pelayanan, serta foto spanduk visual secara interaktif dengan kendali status tampil dinamis.

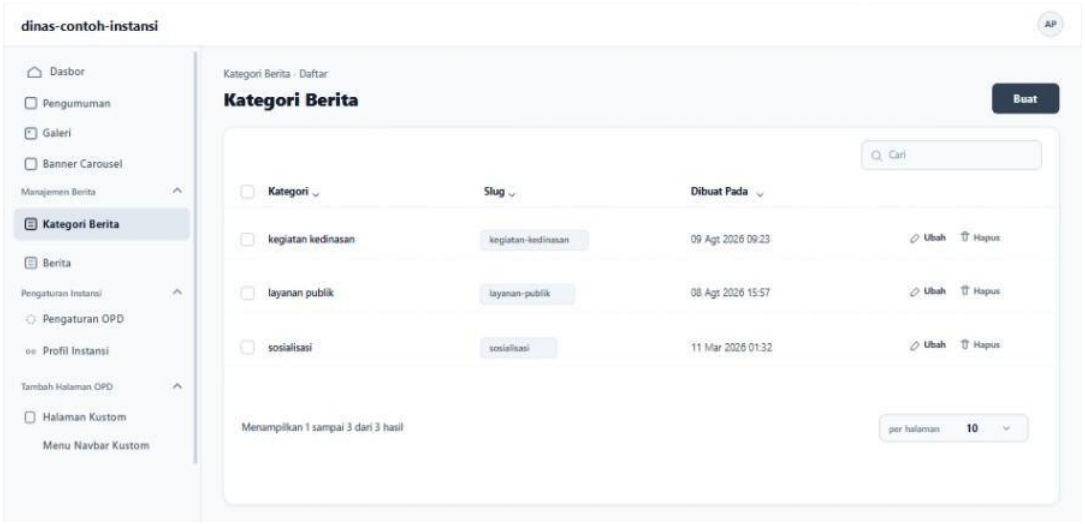
3.2.7.16 Wireframe Halaman Tambah Banner Admin OPD



Gambar 3. 35 Wireframe Halaman Admin OPD Tambah Data Banner

Antarmuka Halaman Tambah Banner berfungsi sebagai formulir konfigurasi spanduk dinamis yang dapat digunakan oleh Admin OPD untuk mengatur teks judul, huruf judul, slogan, kendali status aktivasi banner, serta pengelolaan gambar multi-item.

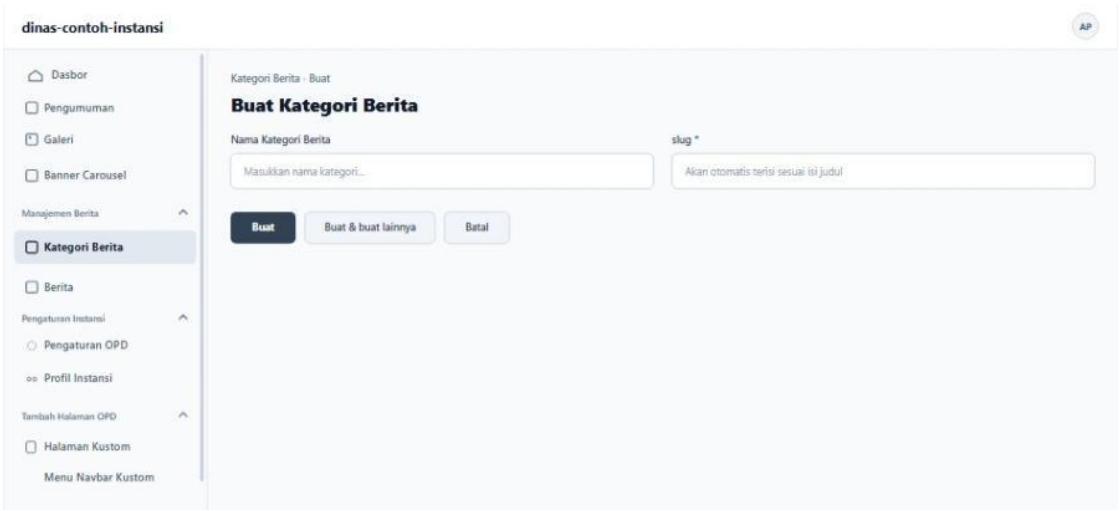
3.2.7.17 Wireframe Halaman Kategori Berita Admin OPD



Gambar 3. 36 Wireframe Halaman Daftar Kategori Berita Admin OPD

Antarmuka halaman Kategori Berita berfungsi sebagai modul untuk mengelola pengelompokan data berita yang diterbitkan oleh OPD. melalui antarmuka ini admin dapat membuat klasifikasi topik berita guna mempermudah pengorganisasian arsip informasi dan penelusuran artikel bagi pengunjung portal publik.

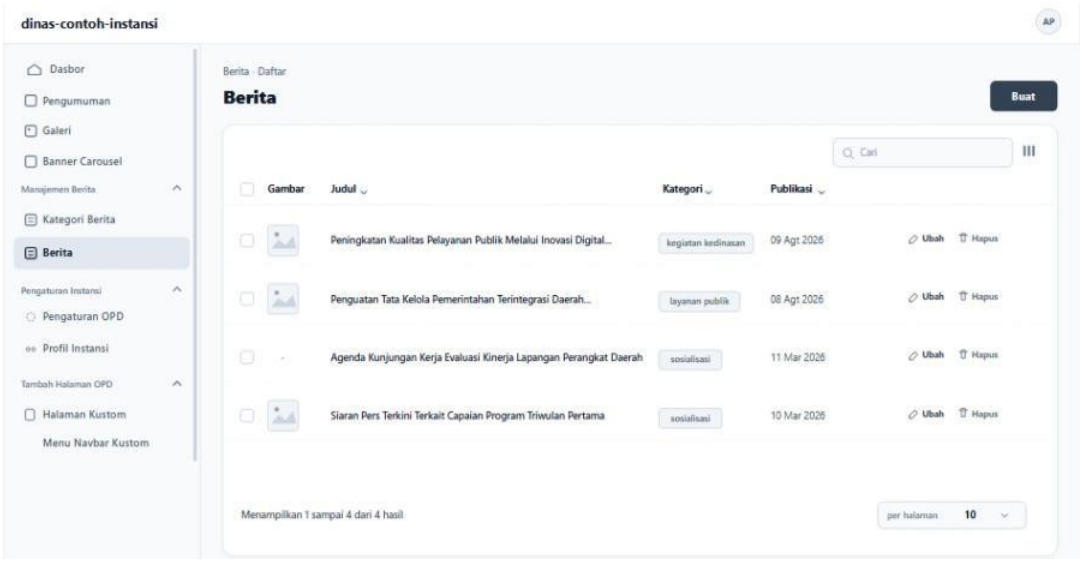
3.2.7.18 Wireframe Halaman Admin Tambah Data Kategori Berita



Gambar 3. 37 Wireframe Halaman Admin OPD Tambah Data Kategori Berita

Rancangan antarmuka Halaman Tambah kategori Berita berfungsi sebagai modul formulir untuk mendefinisikan klasifikasi atau topik artikel baru bagi Admin OPD. formulir ini memberikan akses kepada Admin OPD untuk mengisi data nama berita serta kontrol penyimpanan data secara berulang.

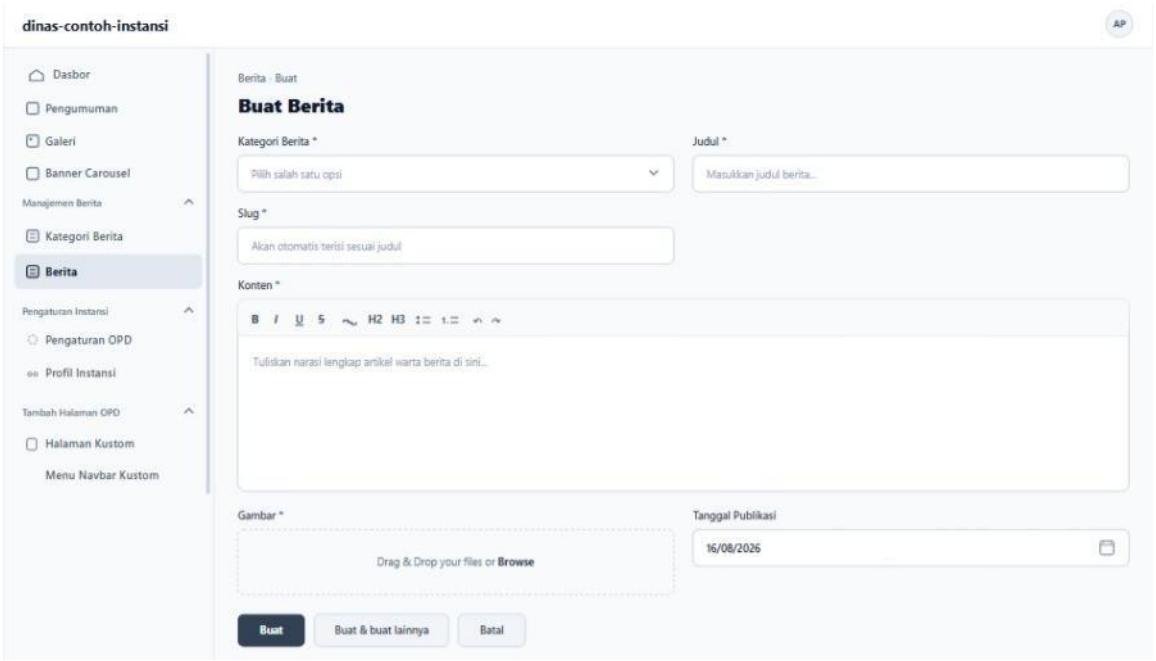
3.2.7.19 Wireframe Halaman Daftar Berita Admin OPD



Gambar 3. 38 Wireframe Halaman Daftar Berita Admin OPD

Rancangan antarmuka Halaman Berita berfungsi sebagai modul pengelolaan dan publikasi seluruh berita resmi, liputan program kerja, serta siaran pers yang diterbitkan oleh OPD kepada masyarakat. Seluruh data pada antarmuka ini terikat secara otomatis pada instansi terkait, yang memungkinkan Admin mengelola publikasi artikel berita beserta gambar berita dan memilih kategori secara terisolasi.

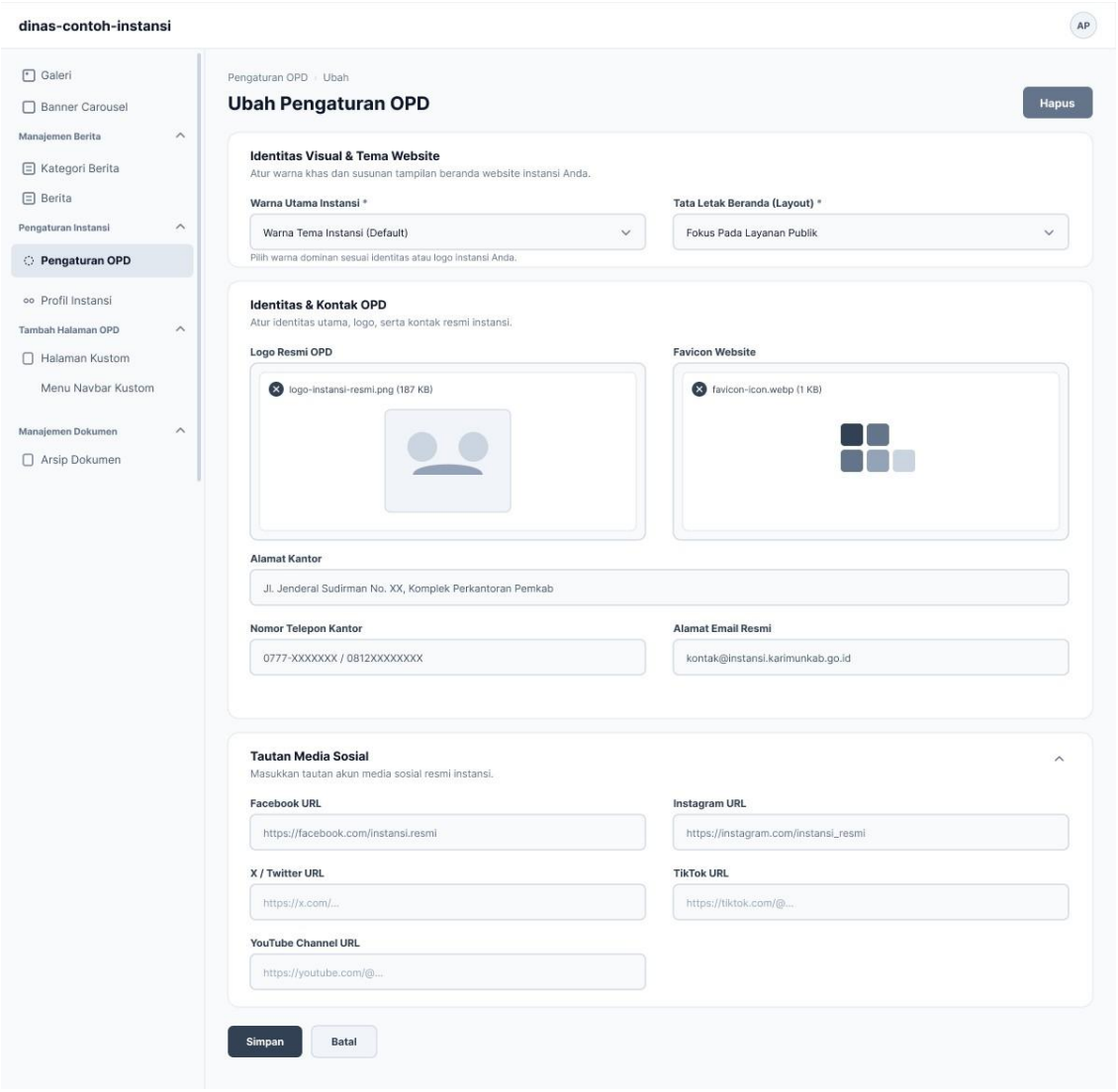
3.2.7.20 Wireframe Halaman Admin OPD Tambah Berita



Gambar 3. 39 Wireframe Halaman Admin OPD Tambah Berita

Rancangan antarmuka Halaman Tambah Berita berfungsi sebagai modul formulir untuk menambah data berita resmi bagi Admin OPD. Formulir ini memungkinkan Admin OPD untuk melakukan tambah berita dengan menentukan kategori berita, pengisian judul berita, penambahan *slug* otomatis, teks narasi deskripsi berita, serta gambar sampul berita.

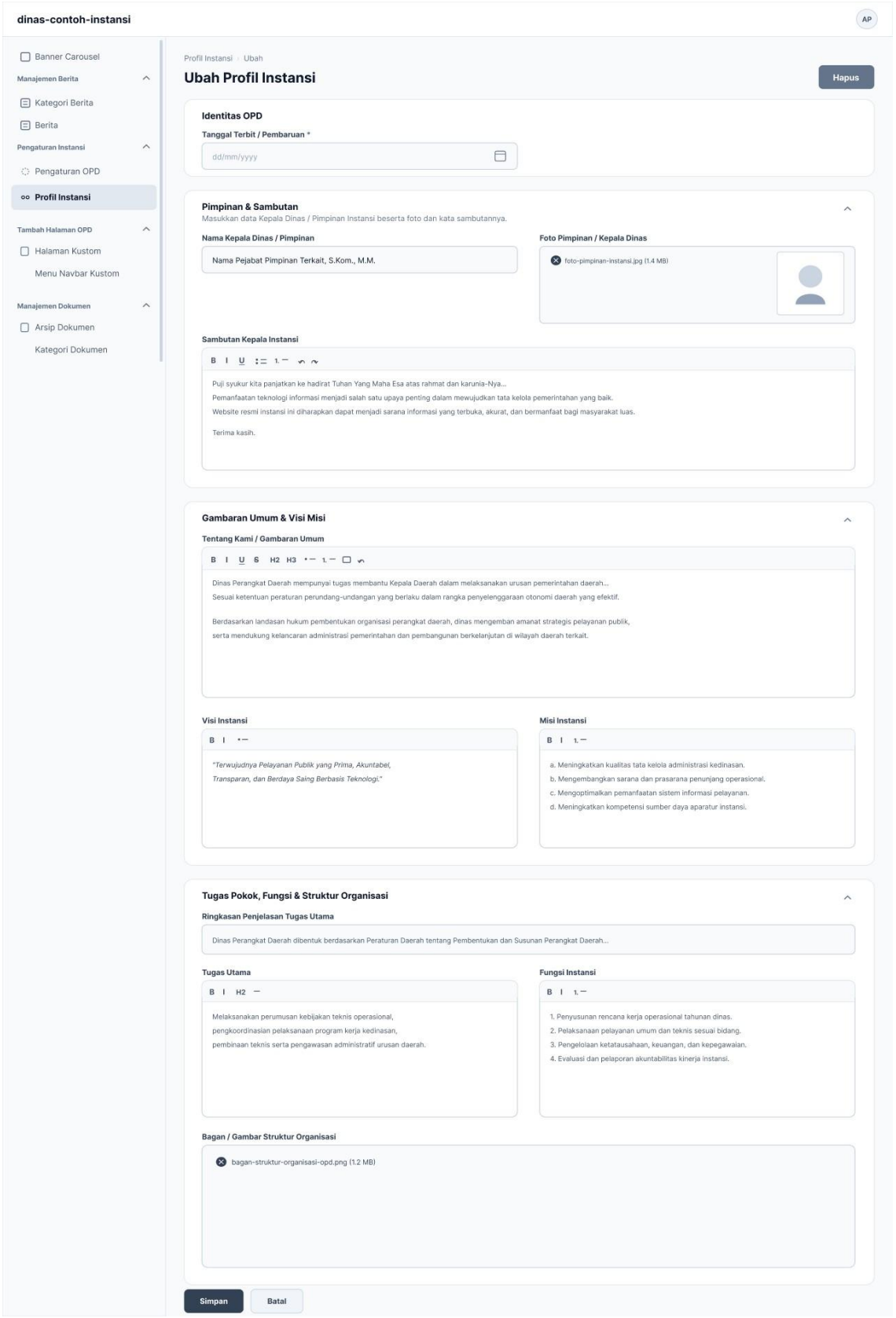
3.2.7.21 Wireframe Halaman Pengaturan OPD Admin OPD



Gambar 3. 40 Wireframe Halaman Pengaturan OPD Admin

Antarmuka halaman Pengaturan OPD berfungsi sebagai modul pengaturan identitas global dan tata letak beranda portal website OPD. Melalui formulir ini, Admin OPD dapat melakukan kostumisasi warna visual, mengunggah asset identitas resmi kedinasan, memperbarui kontak operasional kantor, serta mengintegrasikan saluran media sosial resmi pemerintah daerah.

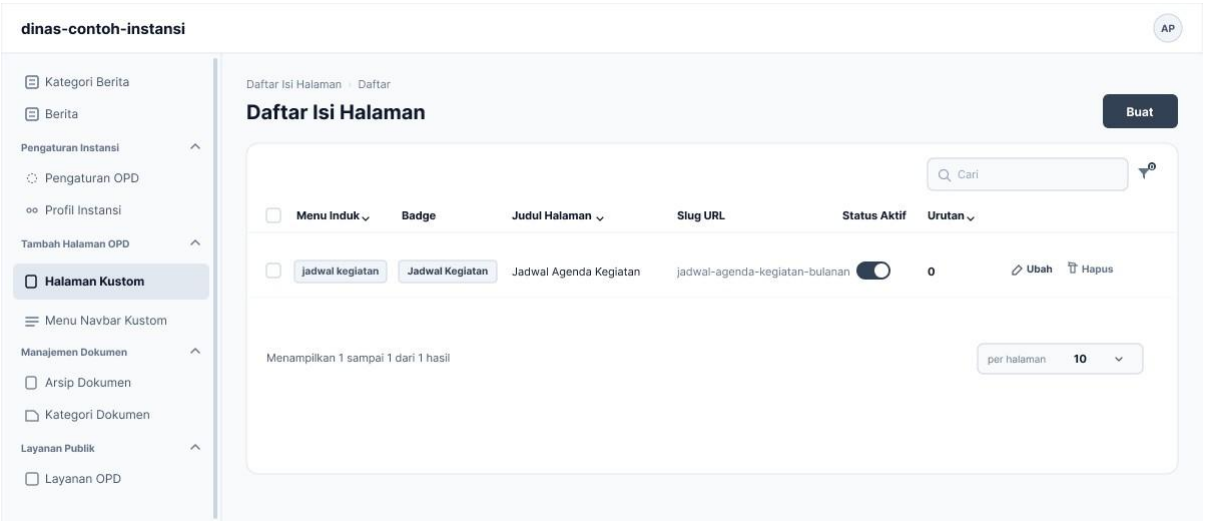
3.2.7.22 Wireframe Halaman Admin Tambah Profil Instansi



Gambar 3. 41 Wireframe Halaman Admin Tambah Profil OPD

Antarmuka halaman Profil Instansi berfungsi sebagai modul konfigurasi profil kelembagaan lengkap bagi OPD. formulir ini memfasilitasi pengisian identitas kepemimpinan kedinasan, kata sambutan pejabat, profil gambaran umum instansi, perumusan visi dan misi, rincian tugas pokok dan fungsi, serta pengunggahan bagan struktur organisasi kedinasan yang dipublikasikan pada portal informasi publik.

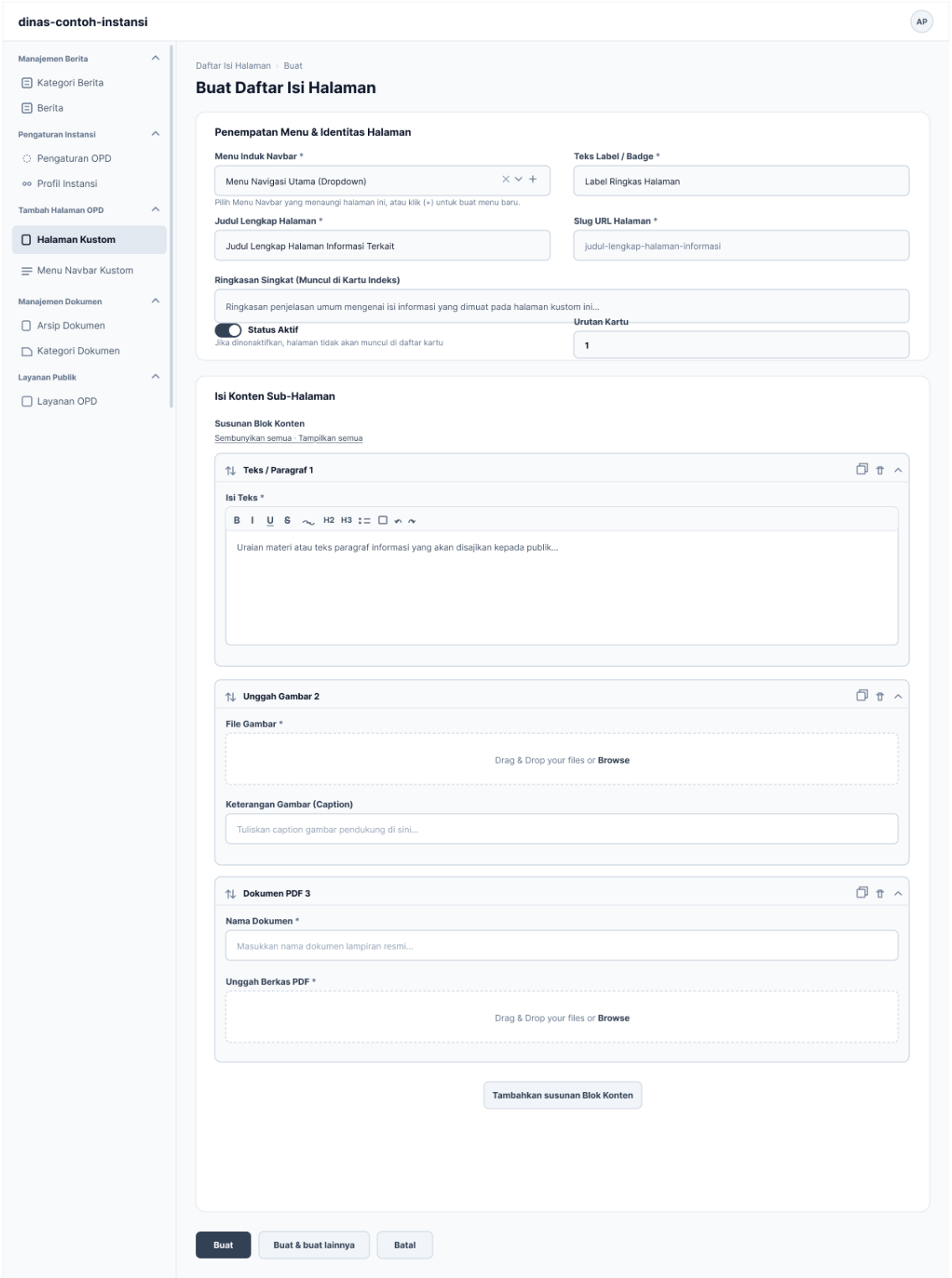
3.2.7.23 Wireframe Daftar Halaman Baru Admin



Gambar 3. 42 Wireframe Halaman Daftar Isi Halaman Baru Admin

Antarmuka halaman Daftar isi halaman berfungsi sebagai modul manajemen tabel seluruh sub-halaman kustom yang dibuat oleh Admin OPD. Modul ini memberikan kemudahan bagi pengelola portal untuk memantau pemetaan halaman terhadap menu navigasi, memeriksa teks label penanda, mengontrol status publikasi penayangan secara instan, serta mengatur urutan posisi tampilan kartu pada portal publik instansi.

3.2.7.24 Wireframe Halaman Admin OPD Tambah Halaman Baru

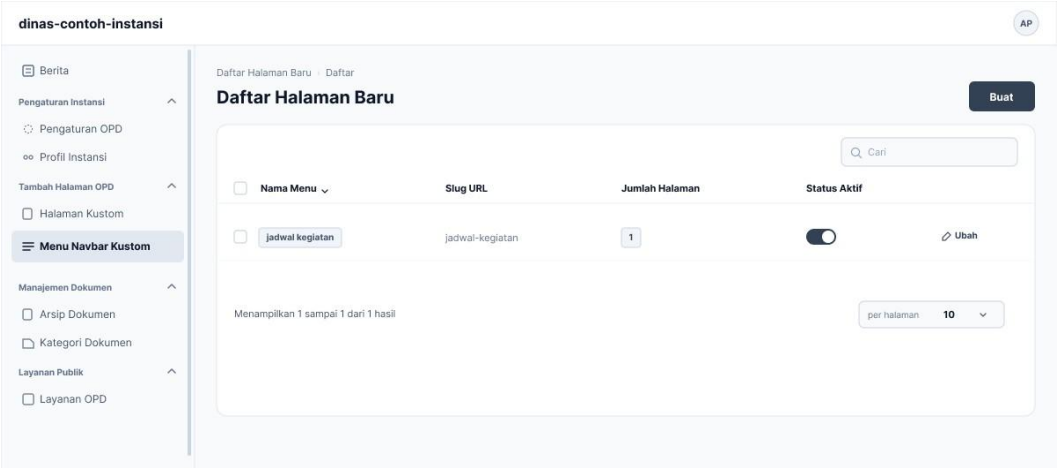


Gambar 3. 43 Wireframe Halaman Tambah Daftar Isi Halaman Baru Admin

Rancangan antarmuka Halaman Tambah Daftar Isi Halaman berfungsi sebagai modul pembuat laman kustom dinamis bagi OPD. formulir ini memfasilitasi penempatan sub-halaman pada menu navigasi utama, pembuatan identitas dan

slug otomatis halaman, penyusunan ringkasan kartu indeks, serta isi konten multi-blok secara fleksibel yang mendukung elemen teks paragraf, gambar dan dokumen.

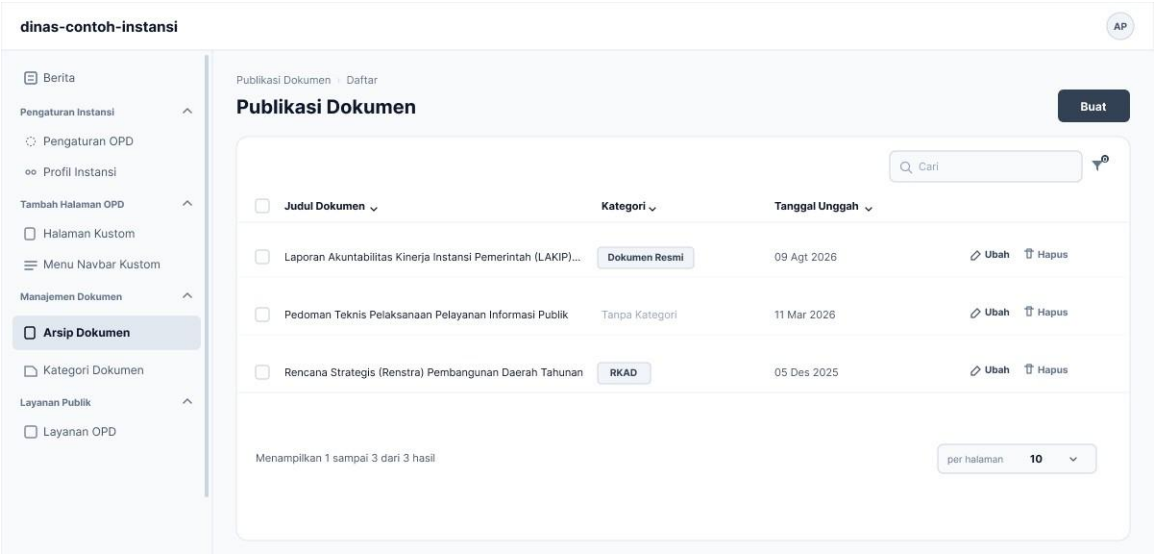
3.2.7.25 Wireframe Halaman Daftar Menu Halaman Baru Admin OPD



Gambar 3. 44 Wireframe Halaman Menu Navigasi Kustom Admin

Antarmuka Halaman Menu Navbar Kustom berfungsi sebagai modul konfigurasi menu navigasi pada bagian *navbar*. Modul ini memfasilitasi pembuatan kategori menu baru untuk menaungi Kumpulan sub-halaman kustom, memantau jumlah halaman yang terasosiasi, serta mengatur status aktif tayang menu secara terpusat.

3.2.7.26 Wireframe Halaman Publikasi Dokumen Admin

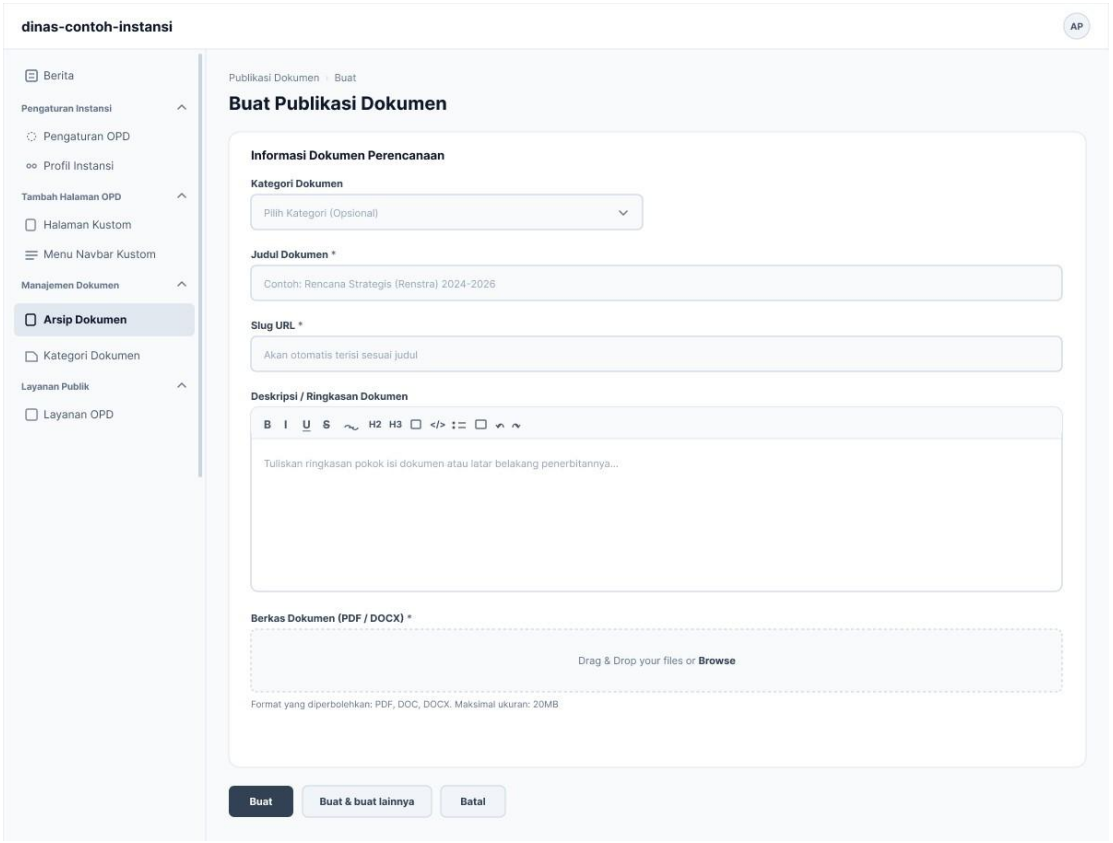


Gambar 3. 45 Wireframe Halaman Publikasi Dokumen Admin

Antarmuka Halaman Publikasi Dokumen berfungsi sebagai modul pengelolaan berkas unduhan resmi, laporan kinerja, rencana strategis (Renstra), peraturan

daerah, serta dokumen perencanaan publik yang diterbitkan oleh OPD. Antarmuka ini dirancang secara terisolasi agar Admin OPD dapat mengarsipkan berkas digital, mengelompokkannya berdasarkan klasifikasi dokumen, dan menyediakan sarana unduhan transparan bagi masyarakat.

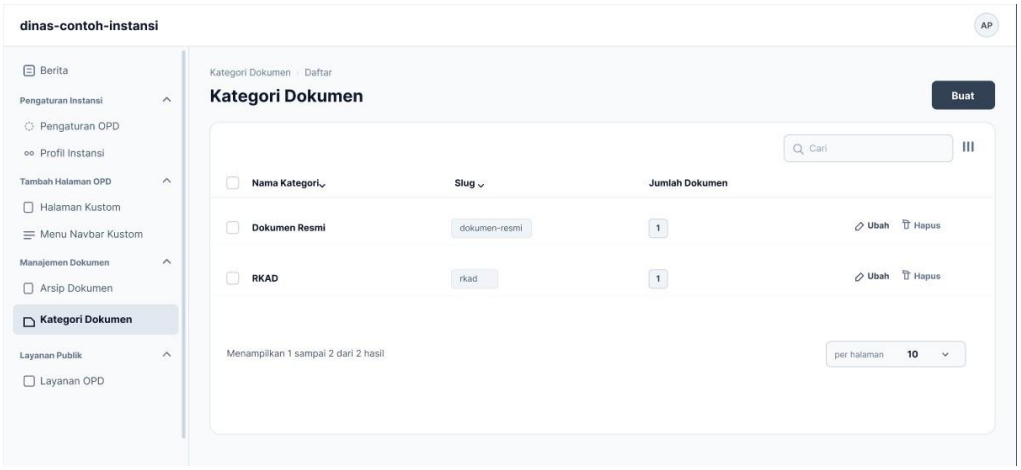
3.2.7.27 Wireframe Halaman Admin OPD Tambah Dokumen Baru



Gambar 3. 46 Wireframe Halaman Tambah Dokumen Baru Admin

Antarmuka halaman Tambah Dokumen berfungsi sebagai formulir untuk mengunggah dan mempublikasikan berkas dokumen resmi baru bagi Admin OPD. Formulir ini memfasilitasi pengelompokan klasifikasi kategori dokumen, pengisian *slug* URL otomatis, penulisan deskripsi atau ringkasan dokumen dengan editor teks, serta pengunggahan berkas digital dengan batas ukuran yang ditentukan.

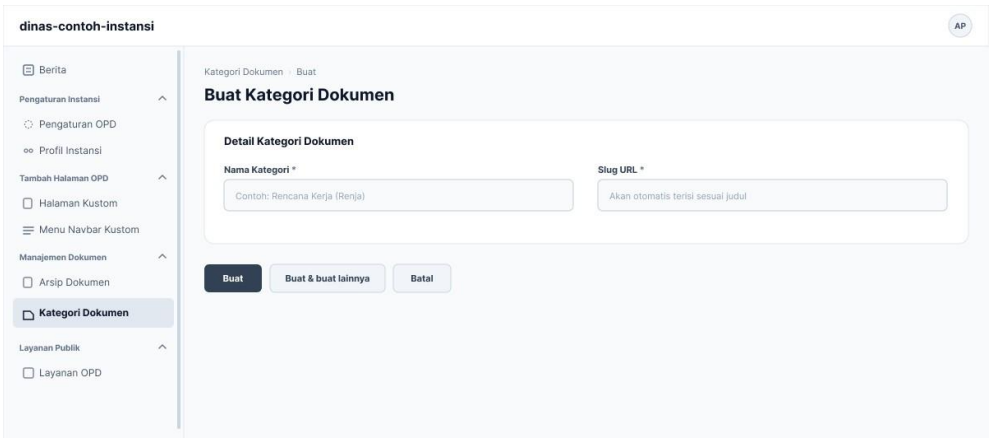
3.2.7.28 Wireframe Halaman Kategori Dokumen Admin



Gambar 3. 47 Wireframe Halaman Kategori Dokumen Admin

Antarmuka halaman Kategori Dokumen berfungsi sebagai halaman yang menampilkan daftar kategori dokumen yang tersedia dan pengelompokan berkas dokumen publik yang dikelola oleh Admin OPD. modul ini memfasilitasi pembuatan kategori dokumen, memantau jumlah berkas dokumen yang terasosiasi dalam tiap kategori secara otomatis, serta menyediakan pengaturan pengorganisasian dokumen unduhan portal publik.

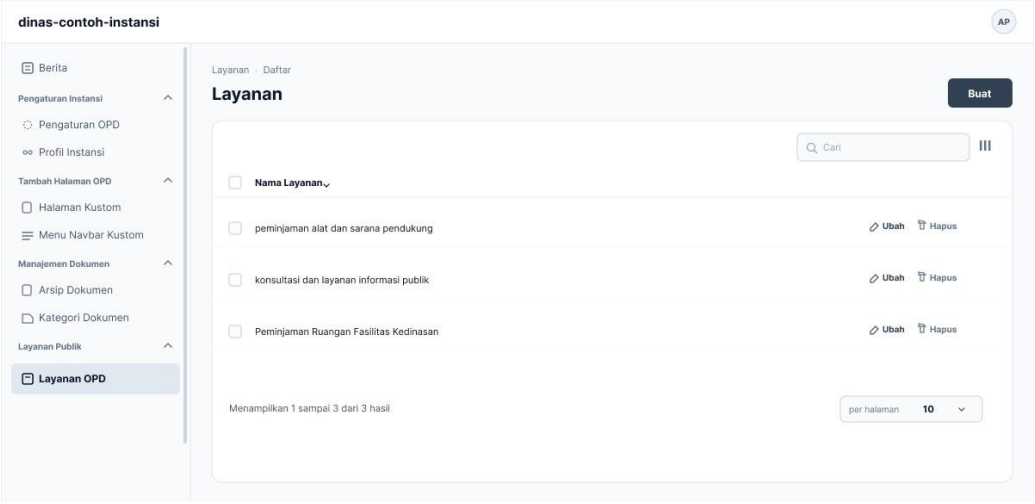
3.2.7.29 Wireframe Halaman Admin Tambah Kategori Dokumen



Gambar 3. 48 Wireframe Halaman Tambah Kategori Dokumen Admin

Antarmuka Halaman Tambah Kategori Dokumen berfungsi sebagai formulir untuk menambah data dan mengelompokkan berkas dokumen publik baru bagi Admin OPD. Formulir ini memfasilitasi penamaan kelompok dokumen serta kemudahan penyimpanan data klasifikasi secara berulang.

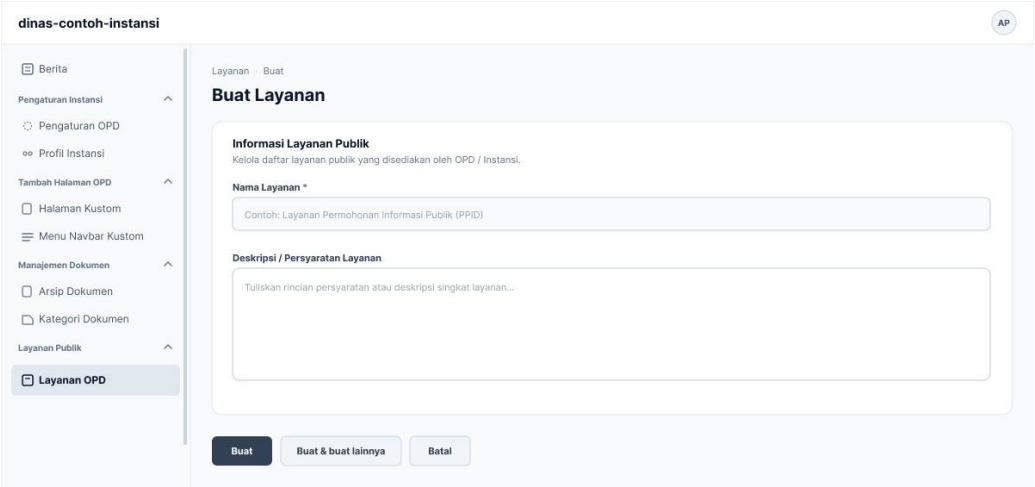
3.2.7.30 Wireframe Daftar Halaman Layanan Admin



Gambar 3. 49 Wireframe Halaman Layanan OPD Admin

Antarmuka halaman Layanan OPD berfungsi sebagai modul pengelolaan daftar layanan publik terpadu yang disediakan oleh Organisasi Perangkat Daerah (OPD). Modul ini memberikan fasilitas bagi Admin OPD untuk mempublikasikan ragam fasilitas yang disediakan oleh instansi, namun hanya sebagai informasi kepada masyarakat.

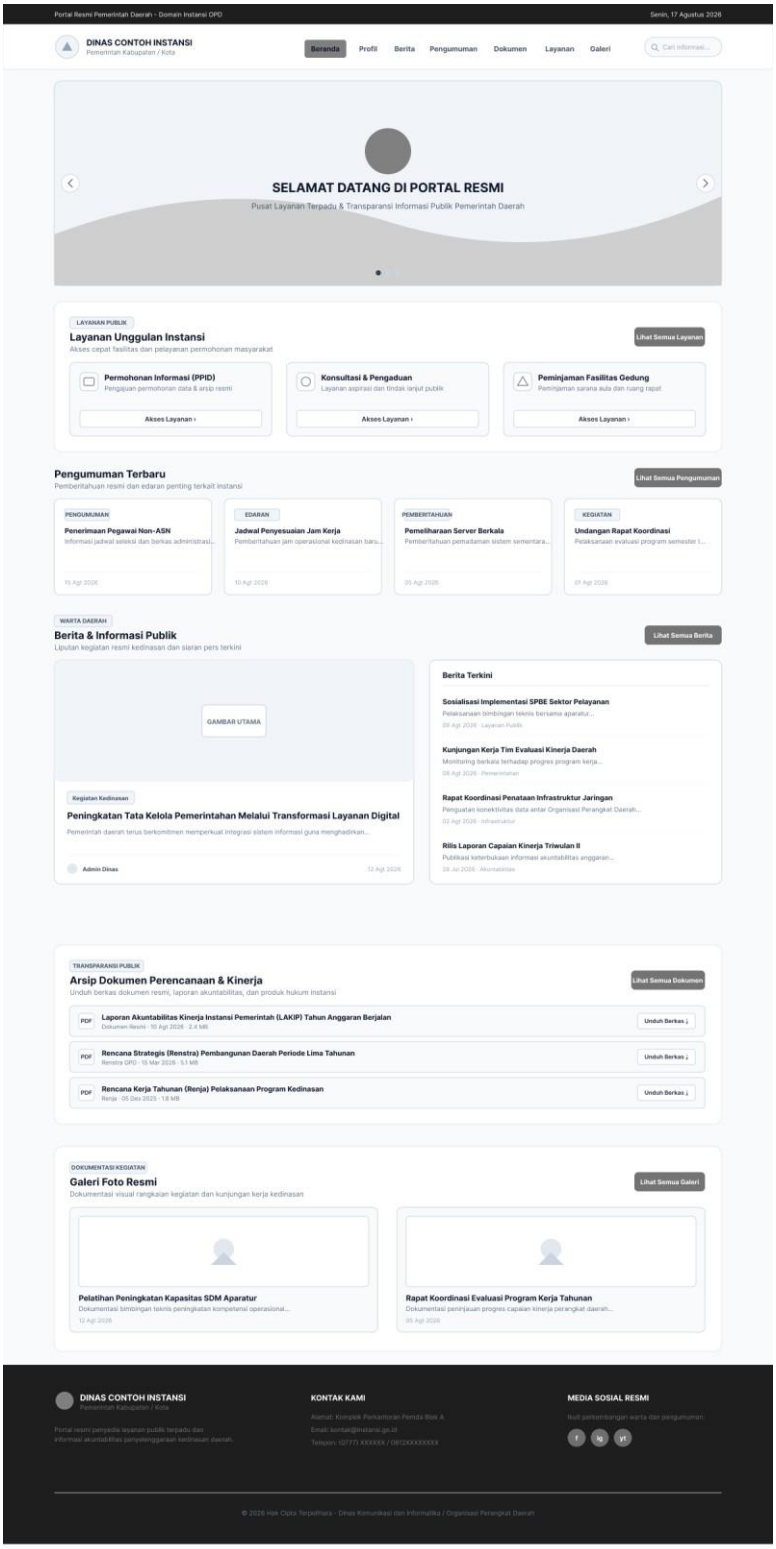
3.2.7.31 Wireframe Halaman Admin Tambah Layanan



Gambar 3. 50 Wireframe Halaman Admin OPD Tambah Data Layanan

Antarmuka Halaman Tambah Layanan berfungsi sebagai formulir untuk mendaftarkan rincian fasilitas dan program pelayanan publik baru yang diselerenggarakan oleh OPD. Formulir ini memberikan akses kepada Admin OPD untuk menambah data layanan seperti judul layanan, serta petunjuk berupa deskripsi.

3.2.7.32 Wireframe Halaman Beranda Pengguna

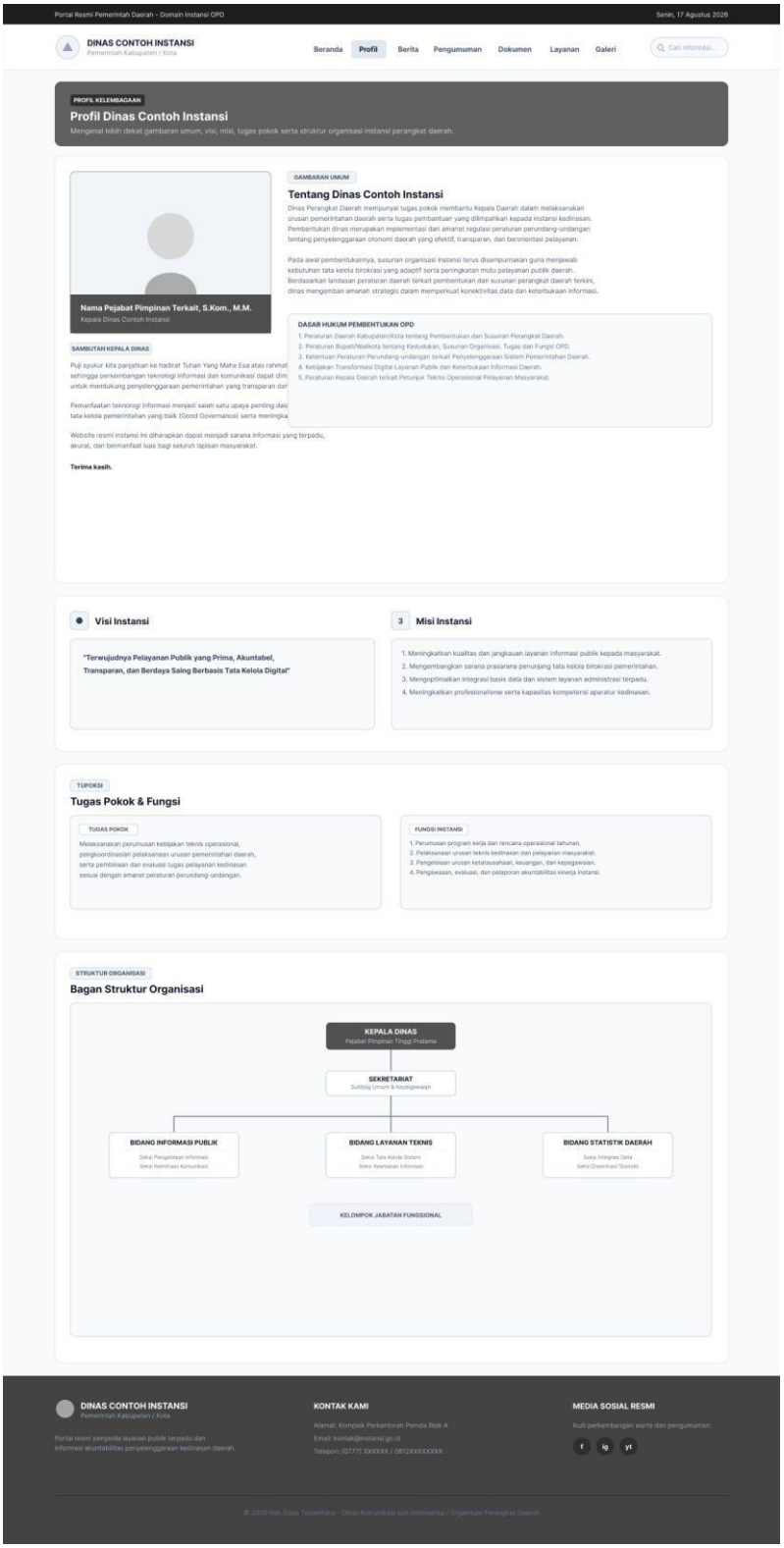


Gambar 3. 51 Wireframe Halaman Beranda Pengguna

Antarmuka halaman beranda merupakan halaman utama saat pengguna mengakses informasi. Halaman ini mengintegrasikan seluruh konten dinamis yang dikelola melalui panel admin OPD secara terisolasi, menyajikan informasi

kedinasan, layanan publik unggulan, berita terkini, pengumuman penting, dokumen transparansi kinerja, dan dokumentasi kegiatan resmi OPD.

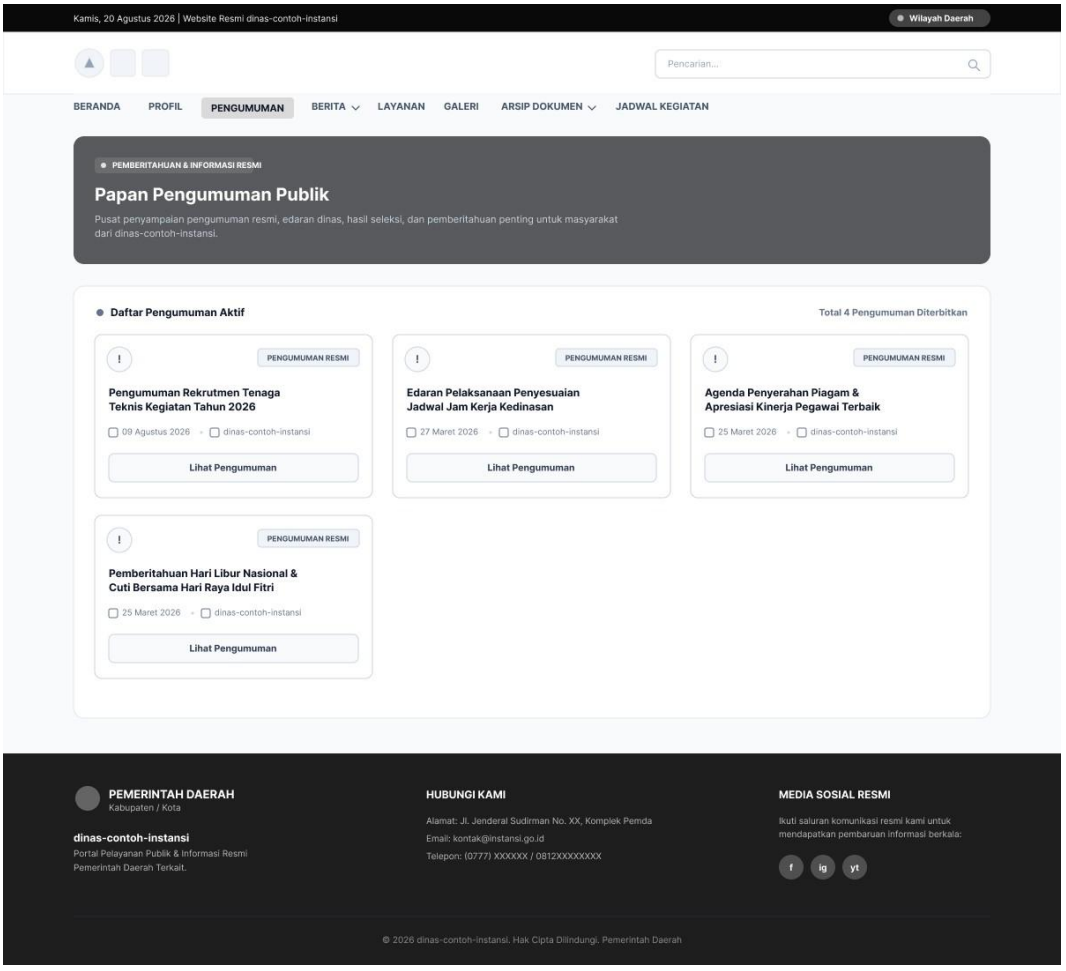
3.2.7.33 Wireframe Halaman Profil Pengguna



Gambar 3. 52 Wireframe Halaman Profil Pengguna

Antarmuka halaman Profil instansi berfungsi sebagai modul presentasi kelembagaan kedinasan yang dapat diakses oleh masyarakat pada portal resmi OPD. Halaman ini menampilkan seluruh data profil instansi yang dikonfigurasikan melalui panel Admin OPD secara dinamis, meliputi kata sambutan pimpinan beserta foto resmi, gambaran umum sejarah dan landasan hukum instansi, rumusan visi dan misi pembangunan daerah, rincian tugas pokok dan fungsi, serta bagan struktur organisasi kedinasan.

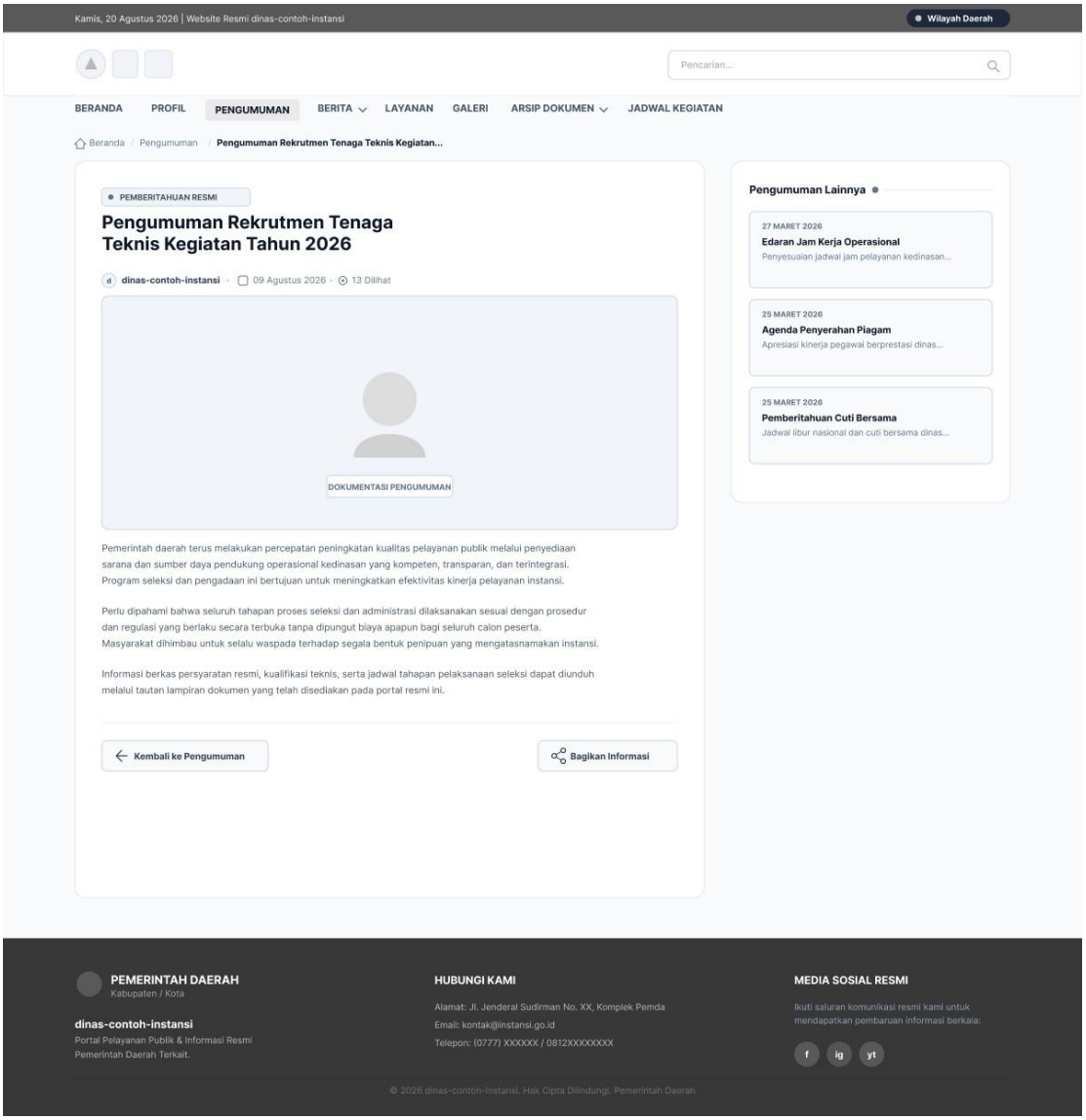
3.2.7.34 Wireframe Halaman Pengumuman Pengguna



Gambar 3. 53 Wireframe Halaman Pengumuman Pengguna

Antarmuka Halaman Pengumuman Pengguna berfungsi sebagai modul untuk menampilkan informasi resmi kedinasan, pengumuman rekruitmen atau seleksi, serta pemberitahuan penting lainnya yang ditujukan bagi masyarakat umum. Halaman ini menyajikan seluruh pengumuman aktif yang diterbitkan melalui panel Admin OPD secara terisolasi.

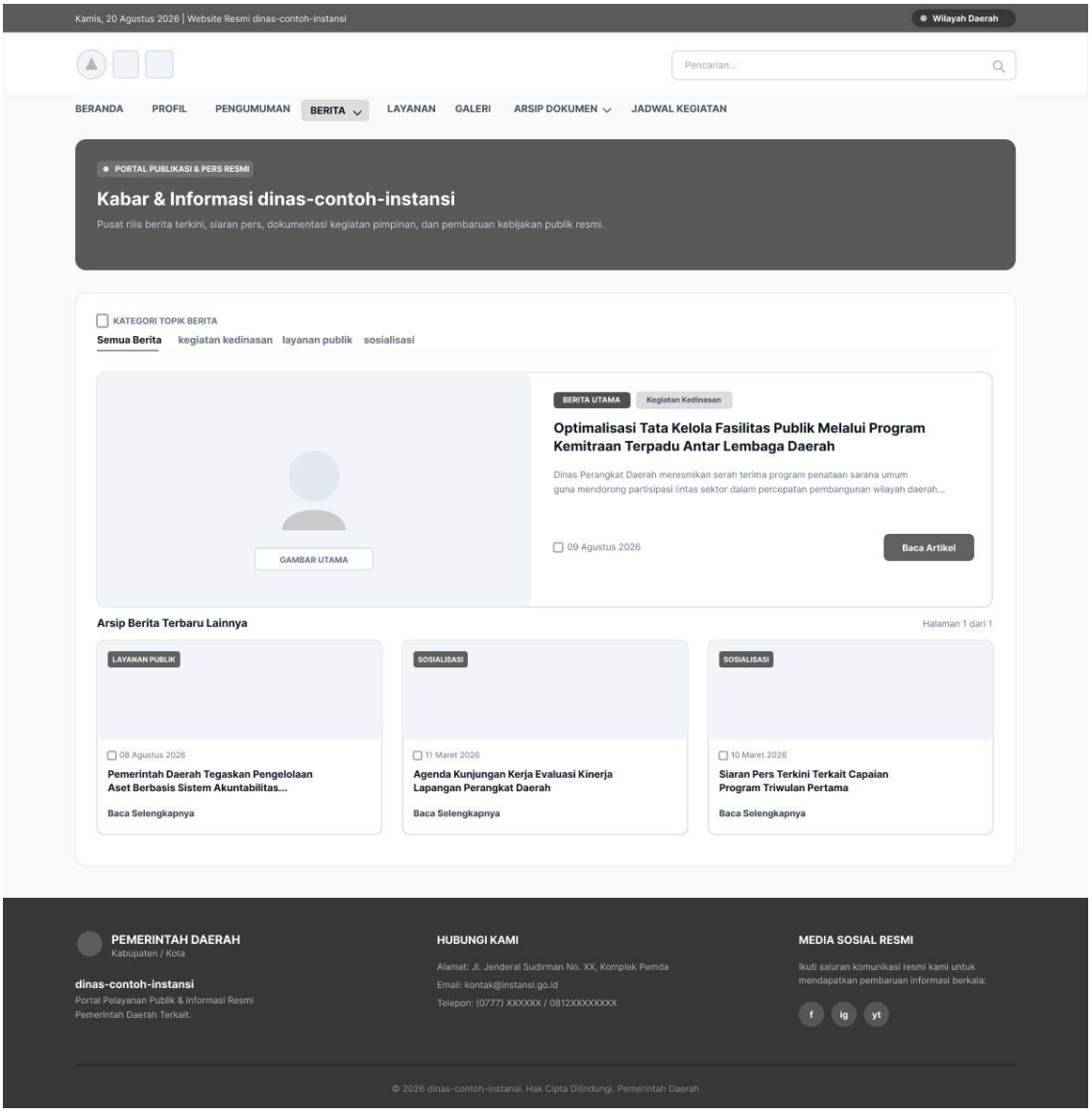
3.2.7.35 Wireframe Detail Pengumuman Pengguna Pengguna



Gambar 3. 54 Wireframe Halaman Detail Pengumuman Pengguna

Antarmuka halaman Detail Pengumuman Pengguna berfungsi sebagai modul untuk menampilkan informasi detail dari instansi terkait. Halaman ini menyajikan seluruh data lengkap yang berisi judul, tanggal penerbitan, gambar, serta deskripsi.

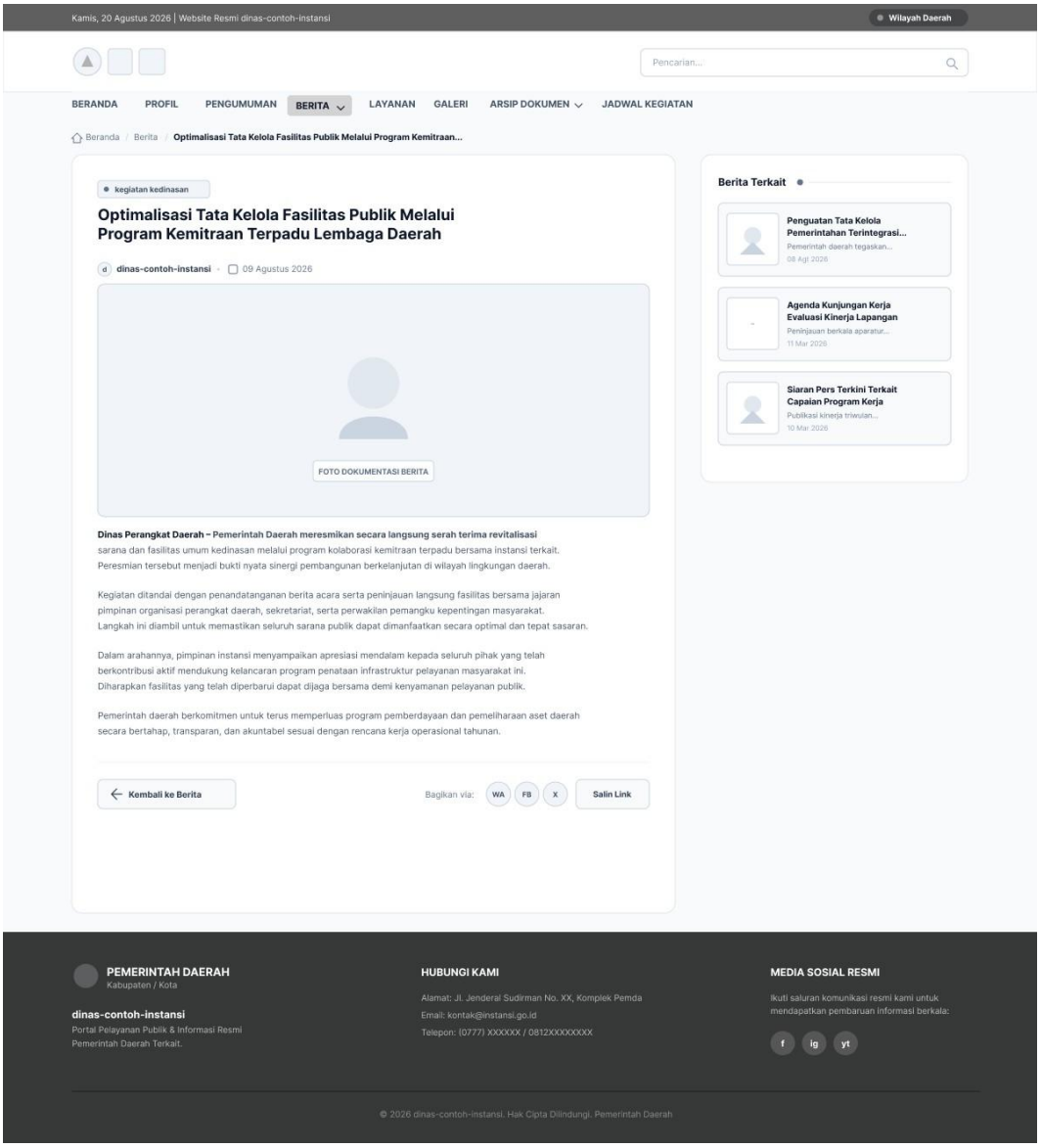
3.2.7.36 Wireframe Halaman Berita Pengguna



Gambar 3. 55 Wireframe Halaman Berita Pengguna

Antarmuka Halaman Berita berfungsi sebagai portal informasi publik resmi, liputan kegiatan dinas, siaran pers, dan pembaruan kebijakan bagi masyarakat luas. Antarmuka ini mengintegrasikan seluruh artikel berita yang dikelola dan dipublikasikan melalui panel Admin OPD secara terisolasi. Halaman pengumuman ini dilengkapi dengan tab navigasi penyaringan berita berdasarkan topik kategori berita, sorotan berita utama, serta daftar berita lain.

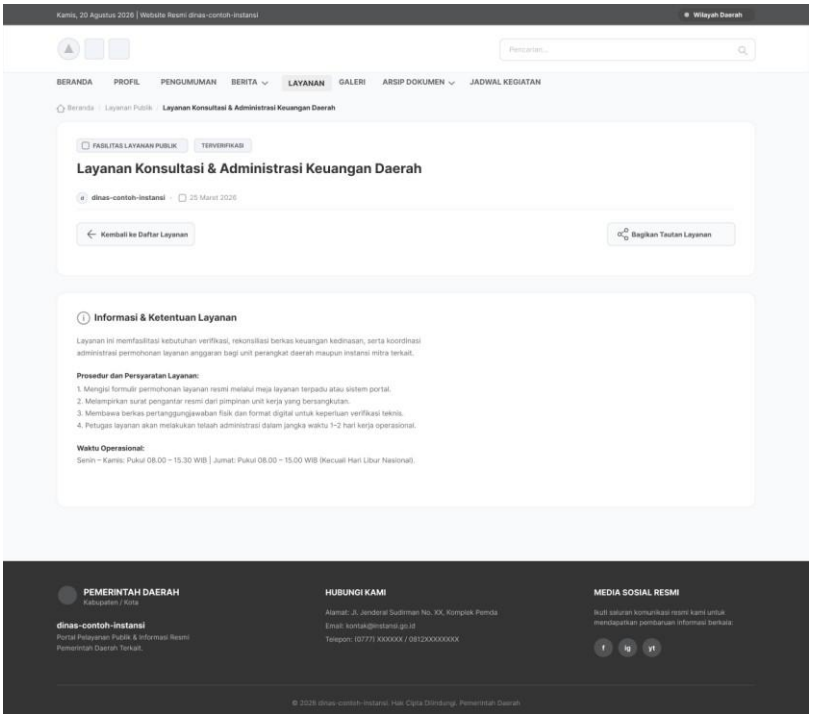
3.2.7.37 Wireframe Detail Berita Pengguna



Gambar 3. 56 Wireframe Halaman Detail Berita

Antarmuka halaman detail berita menampilkan data detail berita berupa nama instansi yang melakukan penambahan berita, tanggal dipublikasikannya berita, kategori berita, gambar, deskripsi mengenai detail berita, serta pada bagian sebelah kanan terdapat daftar berita lainnya yang dimiliki oleh instansi.

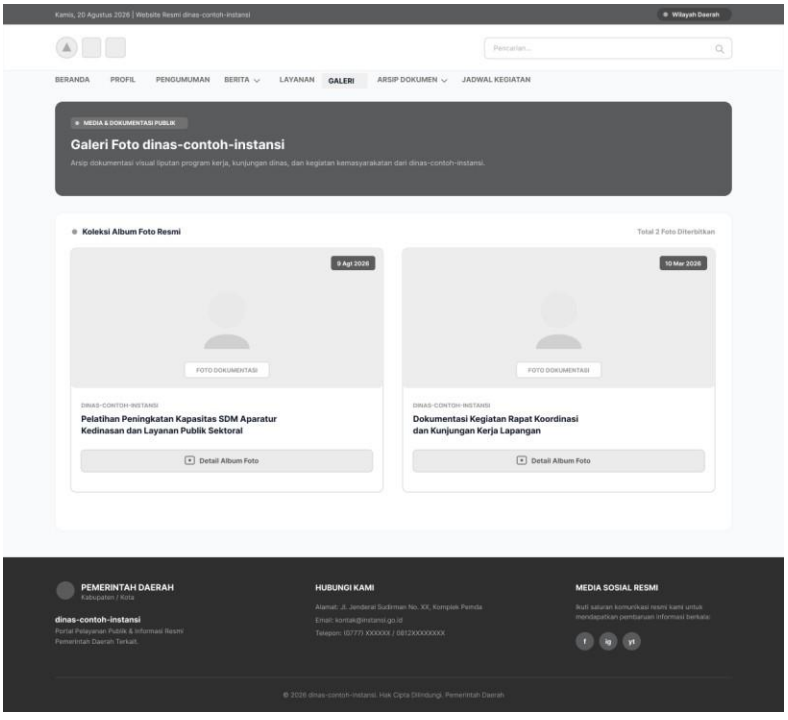
3.2.7.38 Wireframe Layanan Pengguna



Gambar 3. 57 Wireframe Halaman Layanan Pengguna

Rancangan antarmuka halaman Layanan menampilkan data daftar layanan yang tersedia yang dimiliki oleh instansi terkait. Data Layanan menampilkan judul layanan, tanggal publikasi, serta deskripsi layanan instansi terkait.

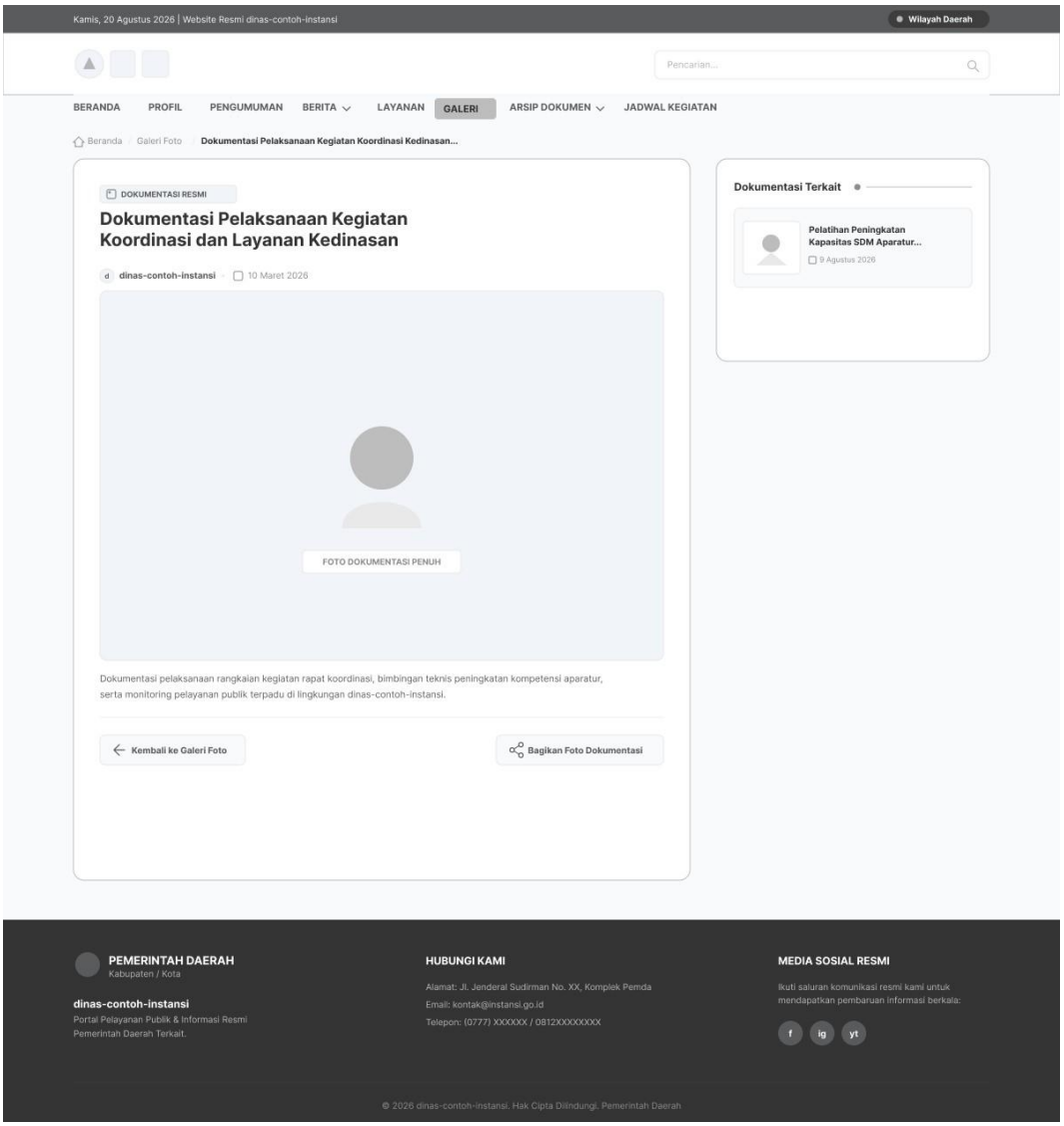
3.2.7.39 Wireframe Halaman Galeri Pengguna



Gambar 3. 58 Wireframe Halaman Galeri Pengguna

Antarmuka Galeri menampilkan daftar galeri yang dipublikasikan oleh instansi terkait, baik itu berupa galeri kegiatan dinas, galeri foto, agenda kegiatan pimpinan, dan galeri lainnya yang dapat didokumentasikan oleh masing-masing instansi.

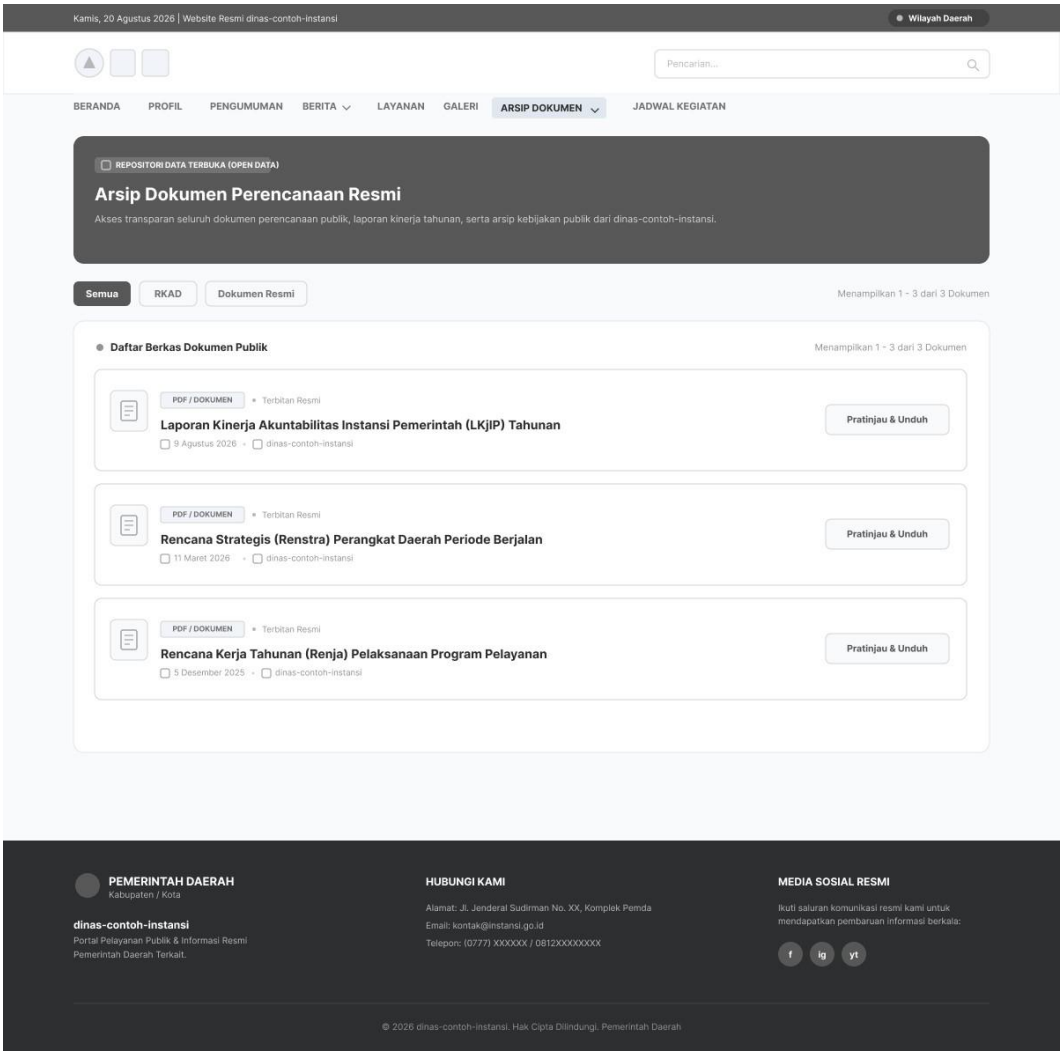
3.2.7.40 Wireframe Halaman Detail Galeri Pengguna



Gambar 3. 59 Wireframe Halaman Detail Galeri Pengguna

Antarmuka Detail Galeri menampilkan data galeri yang tersimpan pada sistem. Halaman ini memuat informasi berupa judul galeri, nama instansi, tanggal upload, foto galeri, beserta deskripsi mengenai detail galeri yang diupload oleh masing-masing instansi.

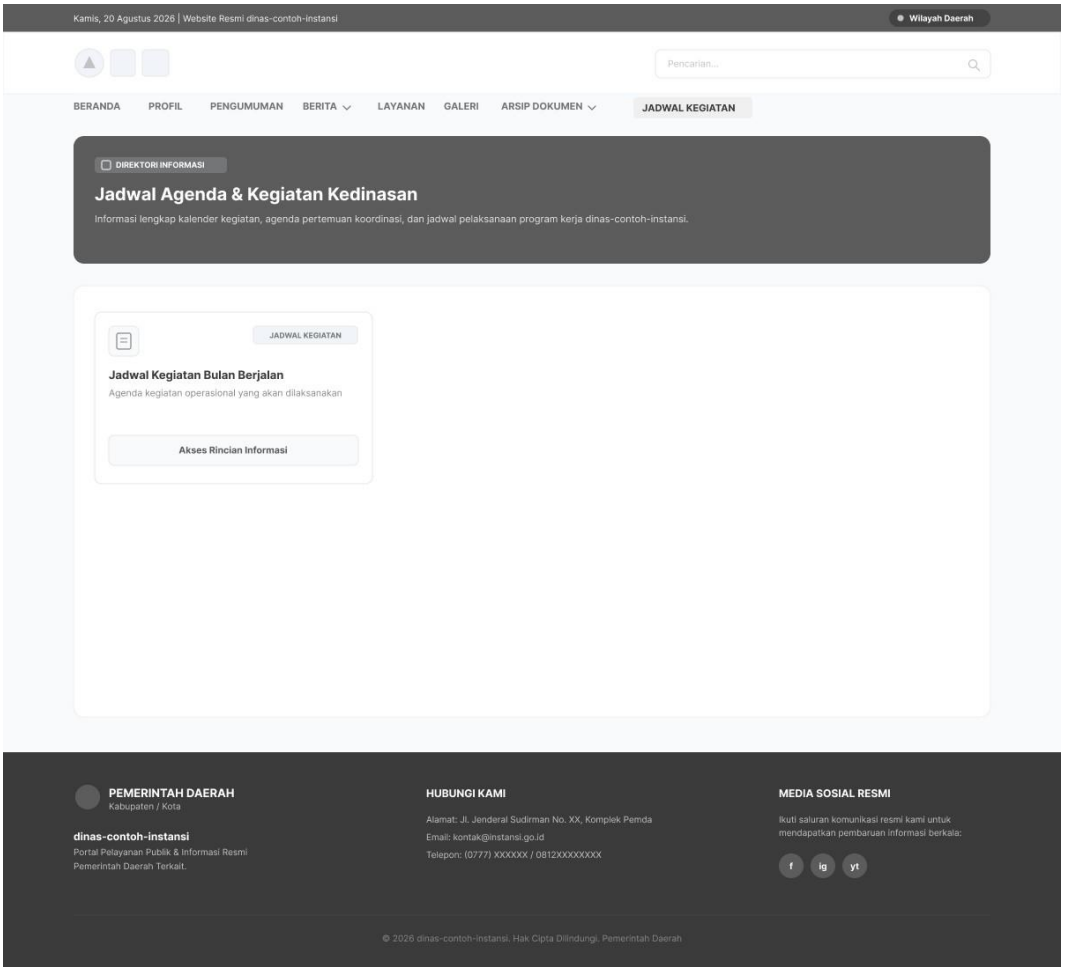
3.2.7.41 Wireframe Halaman Publikasi Dokumen Pengguna



Gambar 3. 60 Wireframe Halaman Publikasi Dokumen Pengguna

Antarmuka halaman Publikasi Dokumen menampilkan daftar seluruh data dokumen publik yang dimiliki oleh OPD terkait. Data yang ditampilkan berurutan berdasarkan tanggal penambahan dokumen dilakukan, serta terdapat navigasi menu yang menyaring data dokumen berdasarkan kategori dokumen.

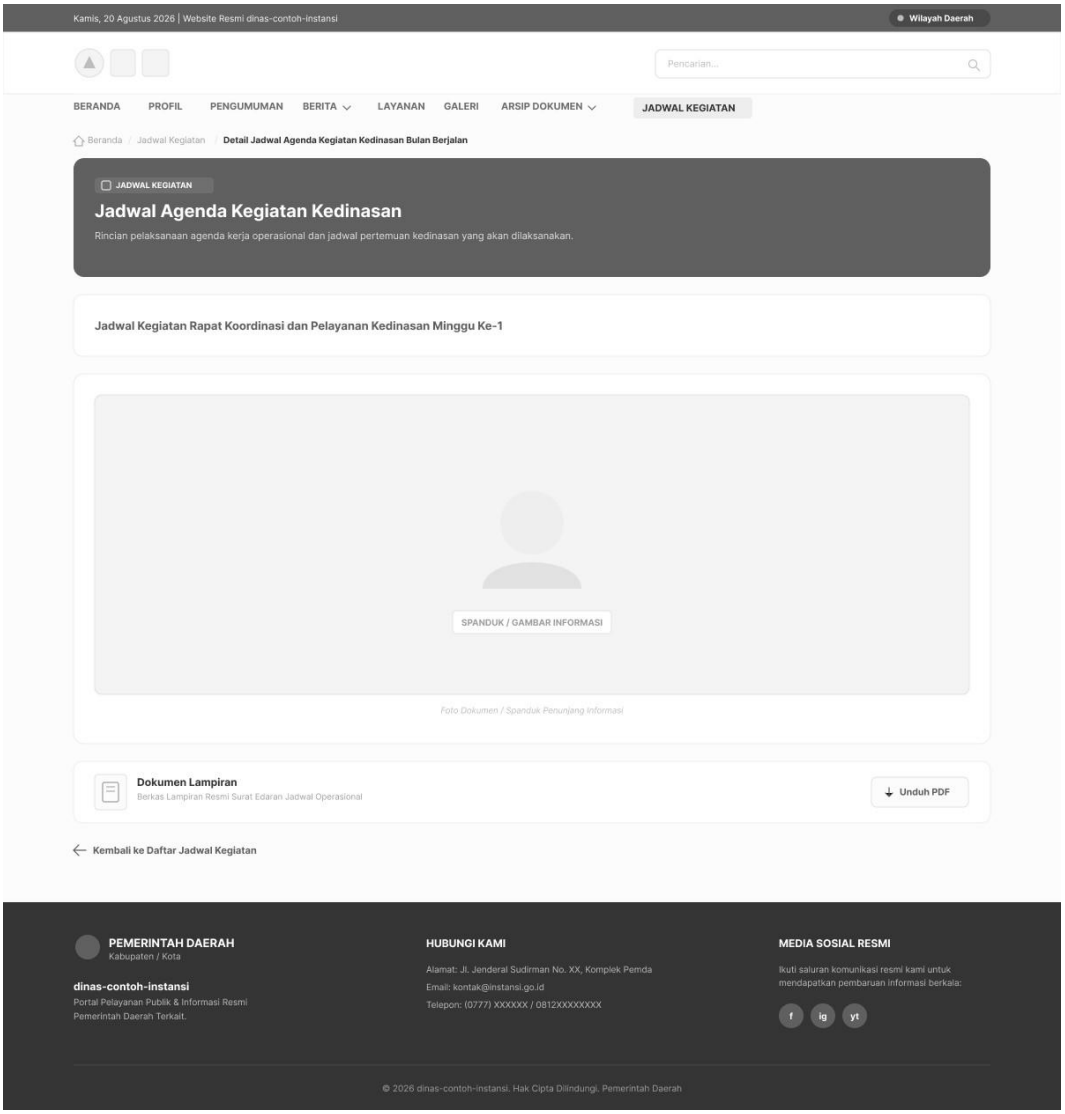
3.2.7.42 Wireframe Halaman Menu Baru Pengguna



Gambar 3. 61 Wireframe Halaman Menu Baru Pengguna

Antarmuka Halaman Menu Tambahan berfungsi sebagai representasi penayangan modul halaman mandiri yang ditambah secara dinamis melalui fitur halaman kustom dan menu navigasi pada panel Admin OPD. modul ini menampilkan informasi yang diambil dari data penambahan menu baru.

3.2.7.43 Wireframe Halaman Detail Halaman Baru Pengguna



Gambar 3. 62 Wireframe Halaman Detail Daftar Isi Halaman Baru

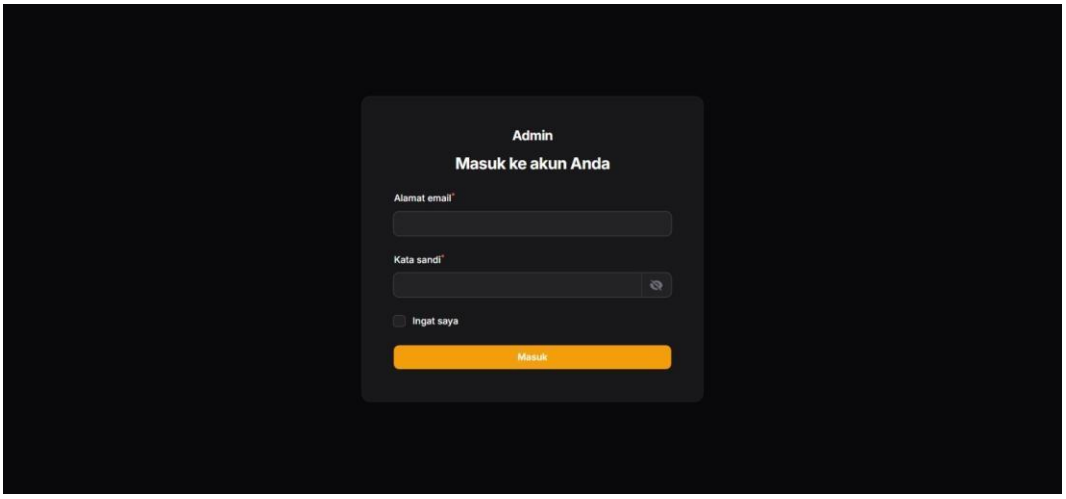
Antarmuka Halaman Detail Konten Halaman Kustom merupakan modul penayangan informasi dari konten halaman yang ditambah secara dinamis oleh Admin OPD.

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1. Hasil Implementasi

Implementasi adalah proses penerapan rancangan sistem ke dalam aplikasi berbasis web menggunakan *framework* Laravel, panel admin Filament, PHP, dan sistem manajemen basis data MySQL. Aplikasi *Web Builder* ini menyediakan mekanisme pengelolaan bertingkat dengan pemisahan hak akses berbasis peran serta arsitektur *multi-tenant*. Superadmin dapat bertugas mengelola data entitas website OPD baru, mengelola akun pengguna, mengatur pembagian hak akses pengguna, serta memantau operasional seluruh portal instansi secara terpusat. Admin OPD dapat melakukan autentikasi ke dalam panel Admin untuk mengelola identitas profil instansi, mempublikasikan berita dan kategorinya, mengunggah dokumentasi kegiatan melalui halaman galeri, menerbitkan pengumuman resmi, mengarsipkan dokumen transparansi kinerja, mengelola direktori layanan publik, mengatur *banner* utama website, serta merancang struktur navigasi dan kustom halaman secara dinamis.

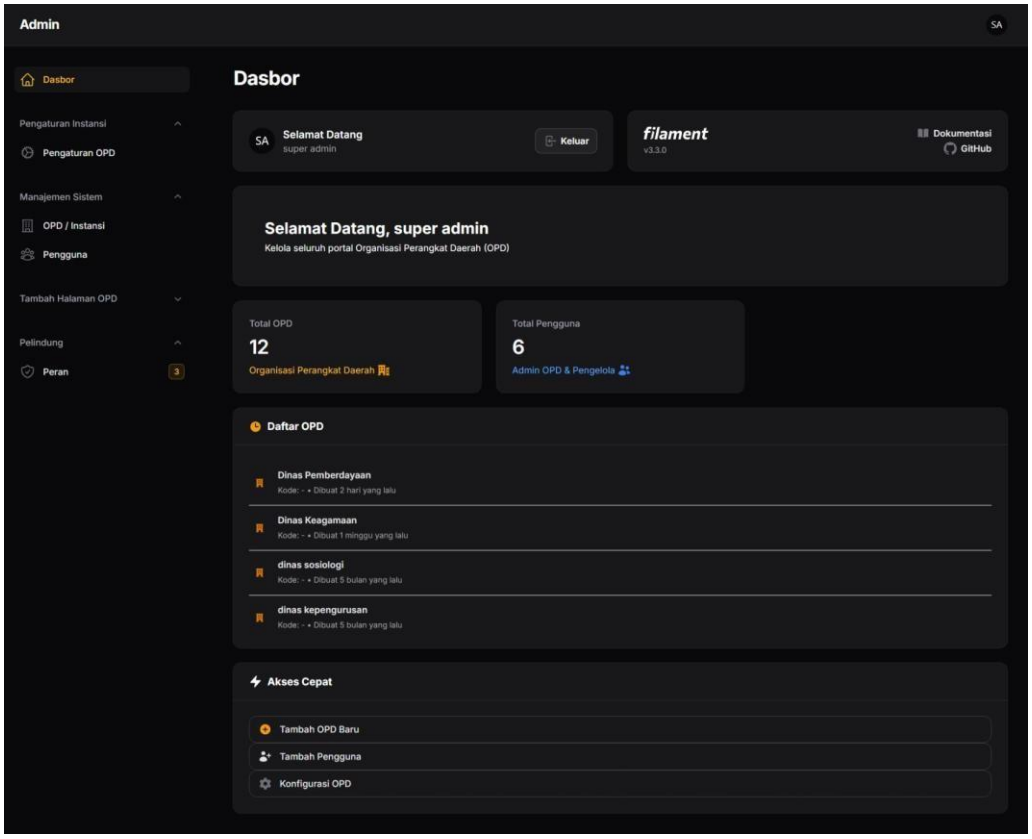
4.1.1 Implementasi Halaman Login



Gambar 4. 1 Halaman Login

Halaman login merupakan gerbang masuk bagi pengguna untuk mengakses panel Admin sesuai dengan perannya masing-masing (Superadmin atau Admin OPD). Sistem memvalidasi kredensial email dan kata sandi yang telah di-enkripsi.

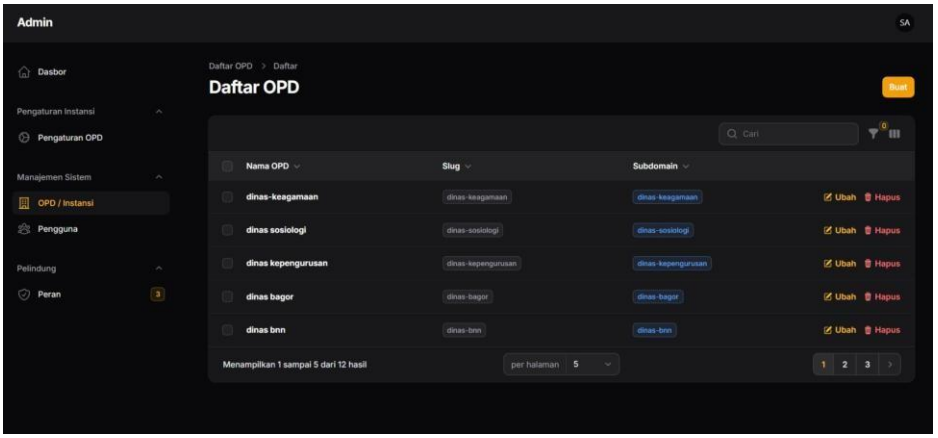
4.1.2 Implementasi Halaman Dasbor Super Admin



Gambar 4. 2 Halaman Dashboard Super Admin

Halaman darbor Superadmin berfungsi sebagai pusat pengawasan seluruh sistem. Halaman ini menampilkan ringkasan total OPD yang terdaftar, jumlah pengguna, serta tombol akses cepat untuk registrasi OPD baru.

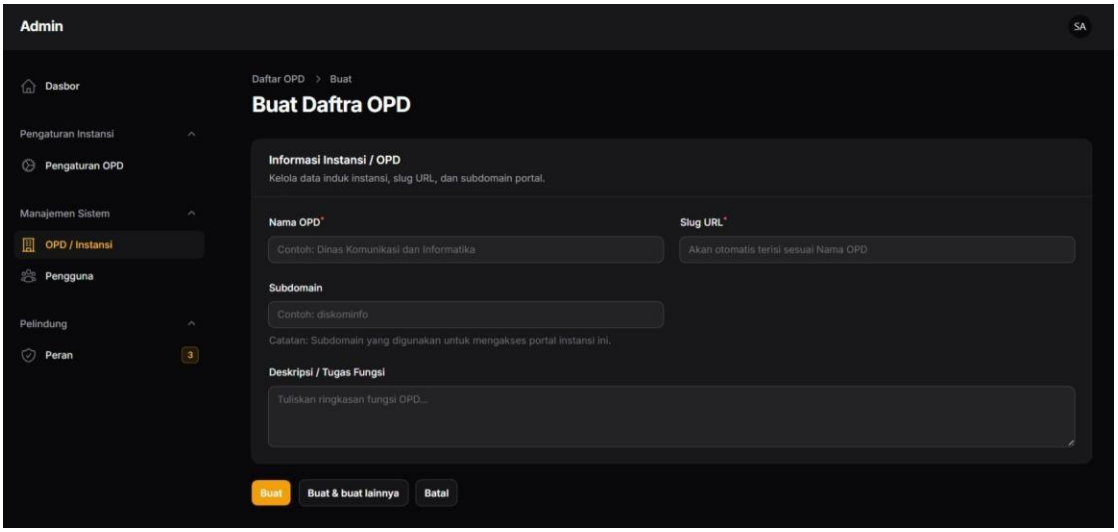
4.1.3 Implementasi Halaman Daftar Master OPD dan Tambah OPD



Gambar 4. 3 Halaman Daftar OPD Superadmin

Halaman master OPD memuat data seluruh instansi yang terdaftar di dalam sistem. Tabel ini dilengkapi kolom nama instansi, *slug* URL, serta tombol aksi ubah dan hapus.

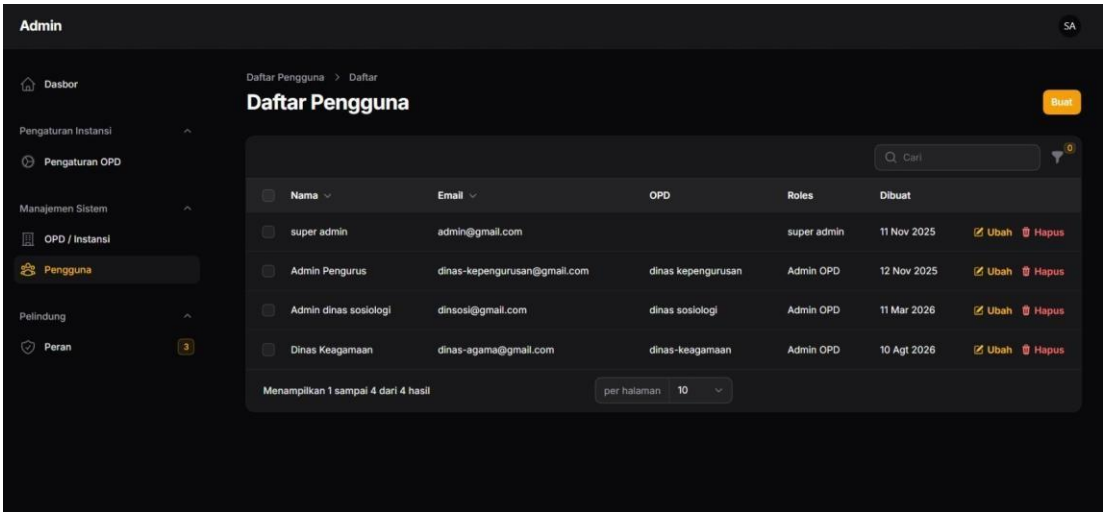
4.1.4 Implementasi Halaman Form Tambah Data OPD



Gambar 4. 4 Halaman Tambah Data OPD Baru

Halaman tambah Data OPD menunjukkan formulir pendaftaran OPD baru. Superadmin dapat mengisi nama instansi pada kolom Nama OPD, di mana sistem akan mengenerasi *Slug* URL serta mengisi deskripsi atau tugas fungsi. Setelah data selesai diinput, superadmin dapat menekan tombol "buat" untuk merekam dan menyimpan data ke dalam sistem atau tombol "batal" untuk membatalkan proses pengisian formulir.

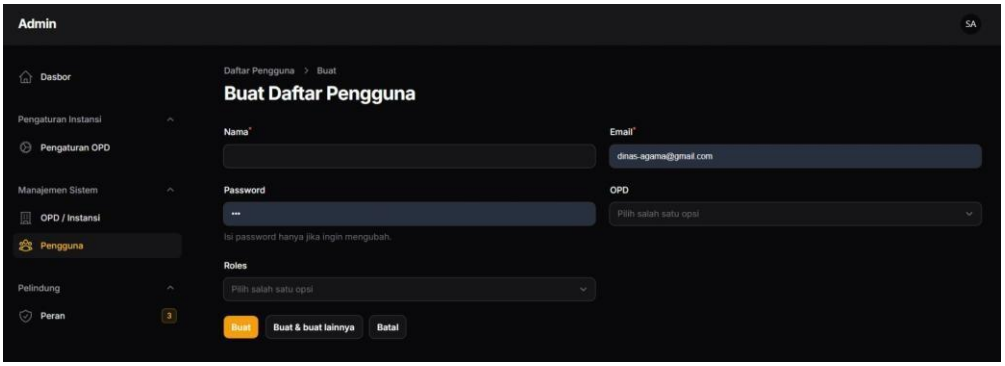
4.1.5 Implementasi Halaman Manajemen Pengguna



Gambar 4. 5 Halaman Daftar Pengguna

Halaman Daftar Pengguna menampilkan tabel daftar seluruh akun pengguna sistem. Informasi yang disajikan mencakup nama lengkap, alamat email, instansi OPD yang dinaungi, penetapan peran, serta tanggal pembuatan akun.

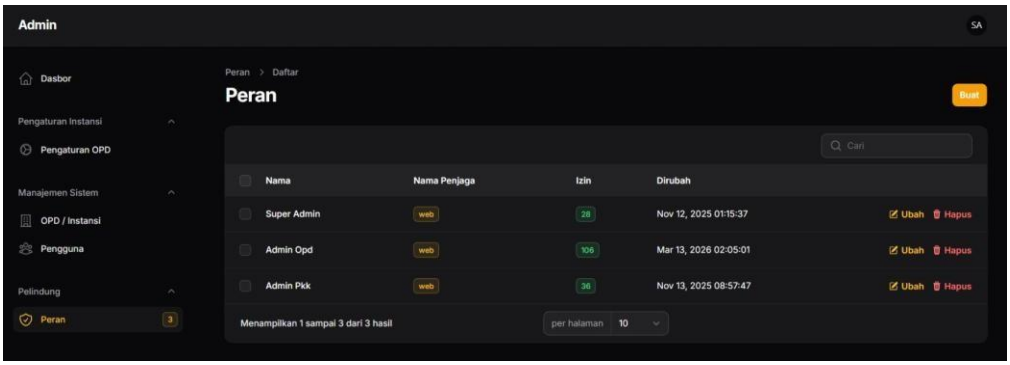
4.1.6 Implementasi Halaman Tambah Akun Pengguna



Gambar 4. 6 Halaman Tambah Pengguna

Halaman Tambah Pengguna memperlihatkan formulir pendaftaran pengguna baru. Melalui antarmuka ini Superadmin dapat mendaftarkan identitas nama, email, kata sandi, serta mengikat akun tersebut ke dalam salah satu instansi melalui dropdown OPD dan menetapkan peran pada pengguna tersebut.

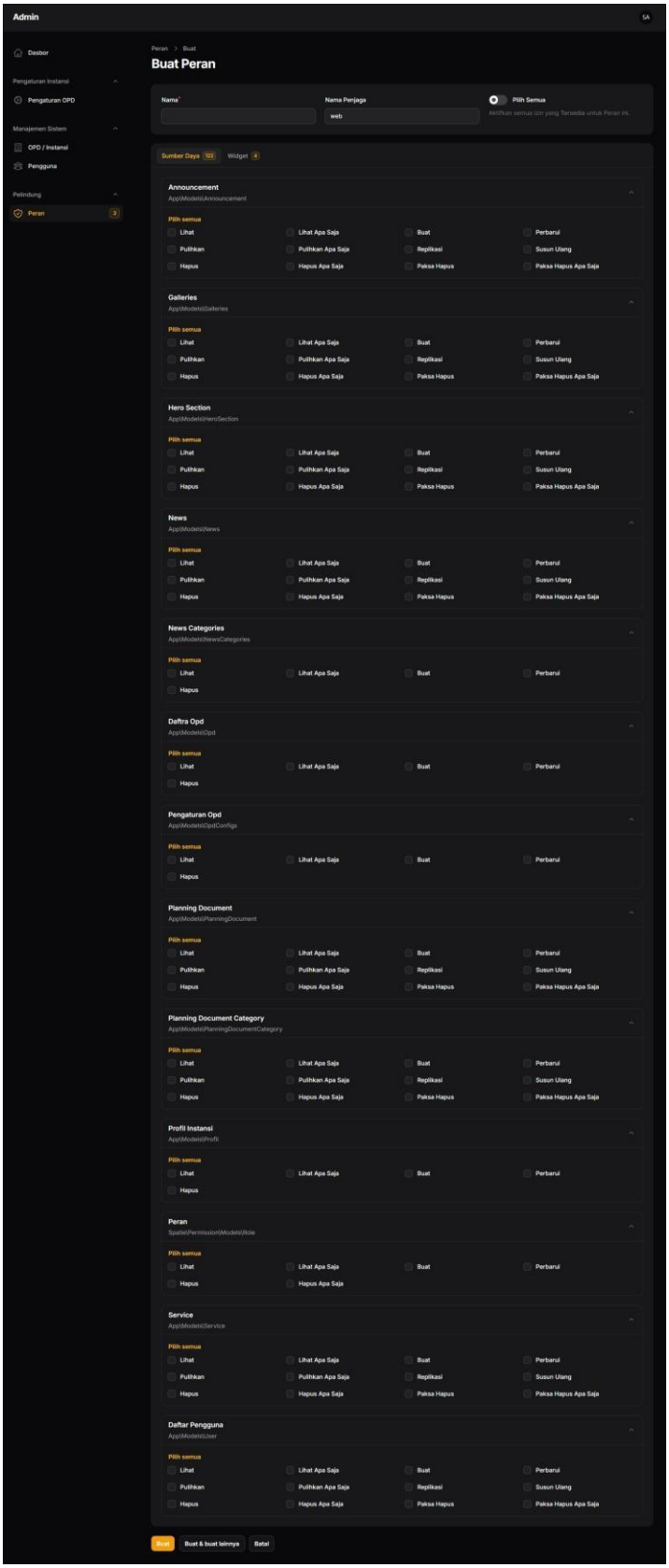
4.1.7 Implementasi Halaman Manajemen Peran



Gambar 4. 7 Halaman Peran Pengguna

Halaman Peran pengguna menunjukkan antarmuka pengelolaan kelompok peran. Halaman ini menampilkan nama peran, autentikasi, dan ringkasan jumlah izin yang terpasang pada masing-masing peran di dalam aplikasi.

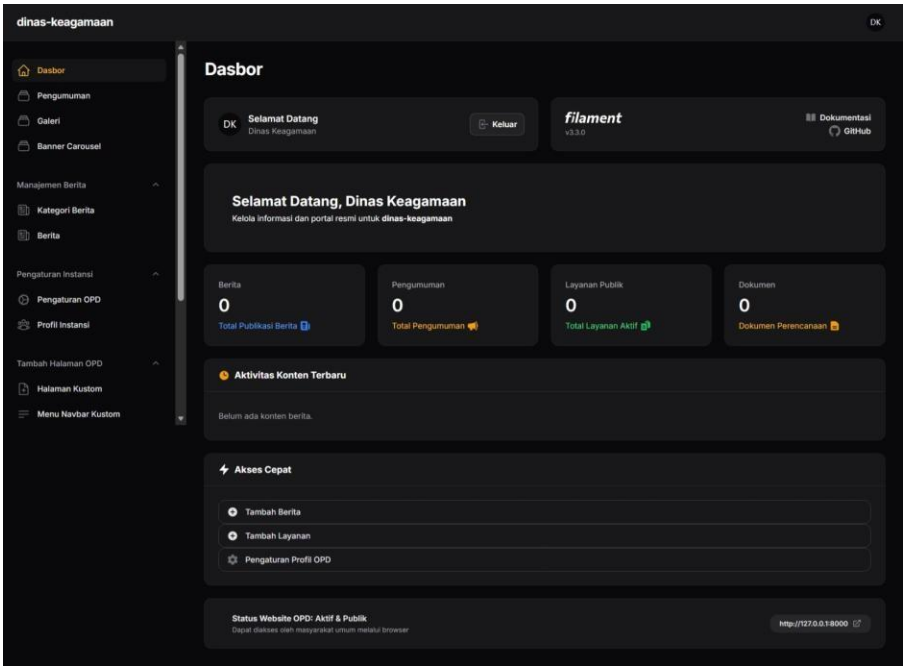
4.1.8 Implementasi Halaman Form Konfigurasi Peran



Gambar 4. 8 Halaman Tambah Peran Pengguna

Halaman Tambah Peran pengguna menampilkan formulir pembuatan dan konfigurasi izin peran. Formulir ini menyediakan checkbox untuk setiap modul sistem dengan opsi tindakan seperti lihat, buat, perbarui, dan hapus.

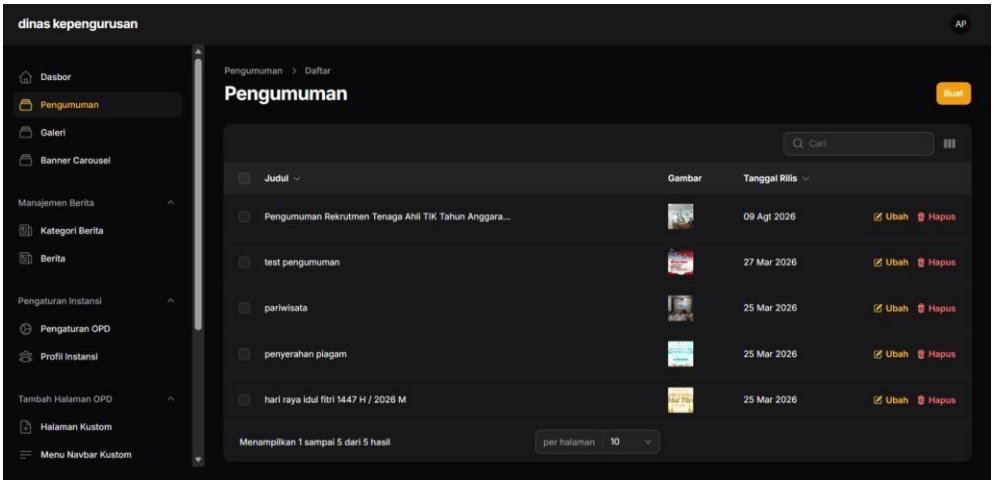
4.1.9 Implementasi Halaman Dasbor Admin OPD



Gambar 4. 9 Halaman Dasbor Admin OPD

Halaman Dasbor Admin OPD memuat ringkasan statistik total publikasi publikasi berita, pengumuman, layanan aktif, dan publikasi dokumen milik instansi terkait. Halaman ini dilengkapi dengan pintasan tombol posting cepat serta tautan langsung untuk melihat pratinjau portal publik instansi yang sedang aktif.

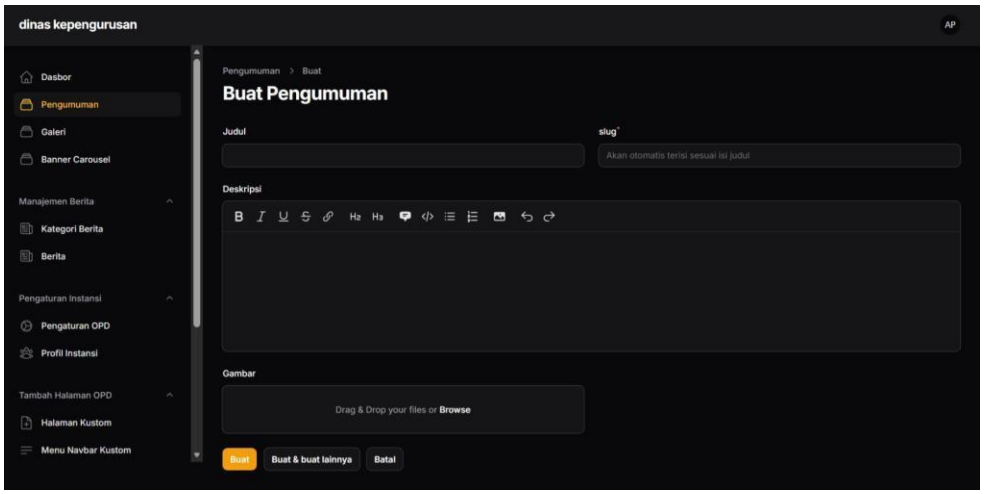
4.1.10 Implementasi Halaman Pengelolaan Pengumuman Admin OPD



Gambar 4. 10 Halaman Pengelolaan Pengumuman OPD

Halaman Pengelolaan Pengumuman menampilkan tabel daftar pengumuman yang memuat kolom judul pengumuman, gambar pendukung, dan tanggal publikasi.

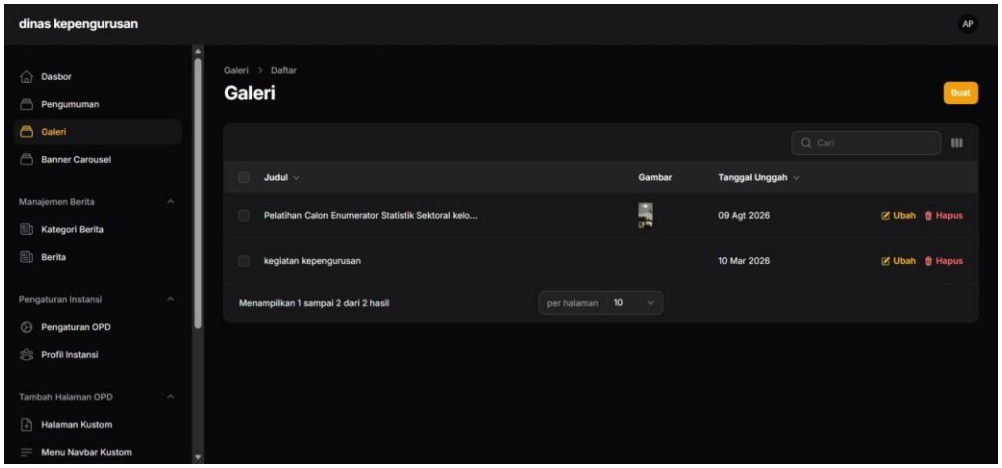
4.1.11 Implementasi Halaman Tambah Pengumuman Admin OPD



Gambar 4. 11 Halaman Tambah Pengumuman Admin OPD

Halaman Tambah Pengumuman menampilkan formulir pembuatan data pengumuman baru yang menyediakan kolom isian judul, auto-*slug*, editor teks deskripsi isi pemberitahuan, serta media unggahan berkas gambar.

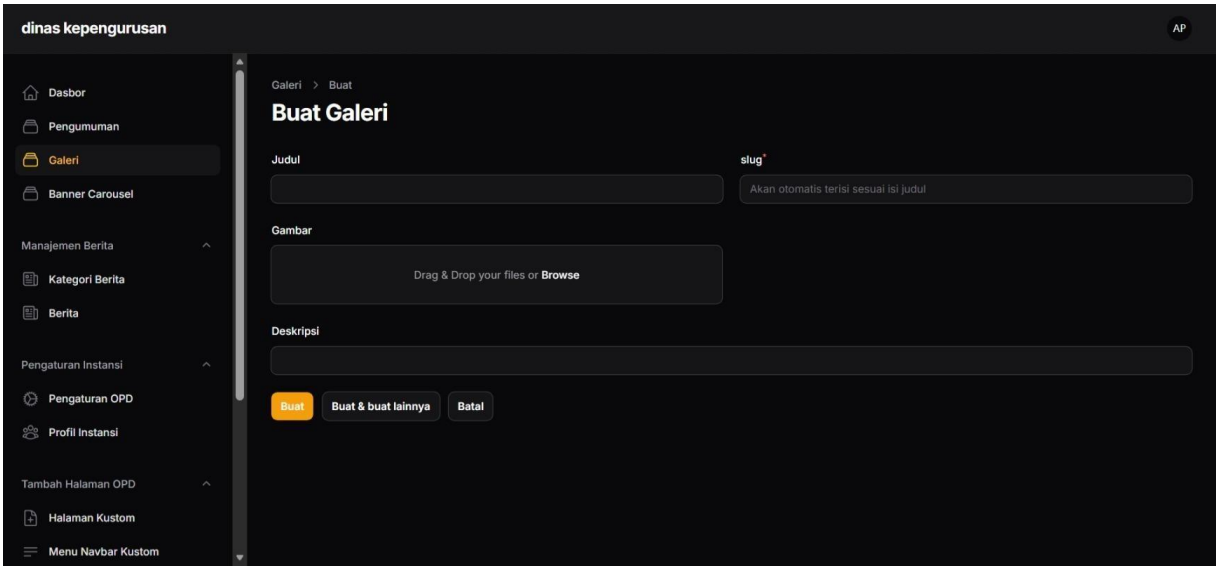
4.1.12 Implementasi Halaman Daftar Galeri Admin OPD



Gambar 4. 12 Halaman Daftar Galeri Admin OPD

Halaman Daftar Galeri menampilkan tabel daftar album galeri kegiatan dinas yang memuat judul dokumentasi, pratinjau foto, dan tanggal unggahan, serta terdapat tombol navigasi untuk melakukan perubahan dan menghapus data galeri. Pada bagian atas kanan terdapat tombol untuk menambah data galeri baru.

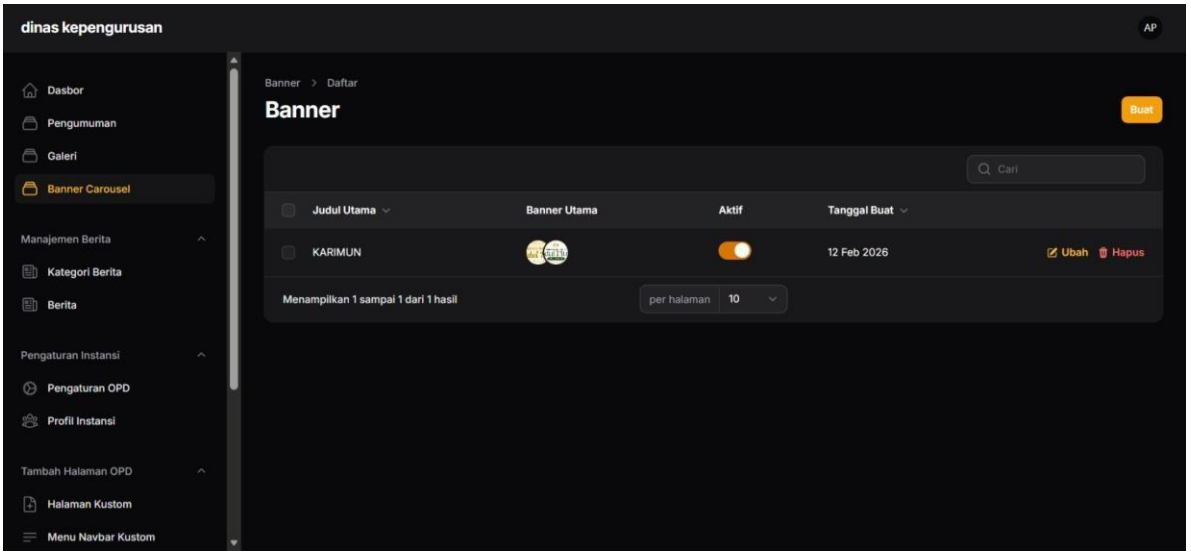
4.1.13 Implementasi Halaman Tambah Galeri Admin OPD



Gambar 4. 13 Halaman Tambah Galeri Admin OPD

Halaman Tambah Galeri memperlihatkan formulir input dokumentasi baru dengan kolom isian judul, area unggahan berkas foto, dan kolom deskripsi. Setelah admin OPD selesai mengisi data baru, admin OPD dapat menekan tombol “buat” untuk menambah data, dan “batal” untuk membatalkan aksi.

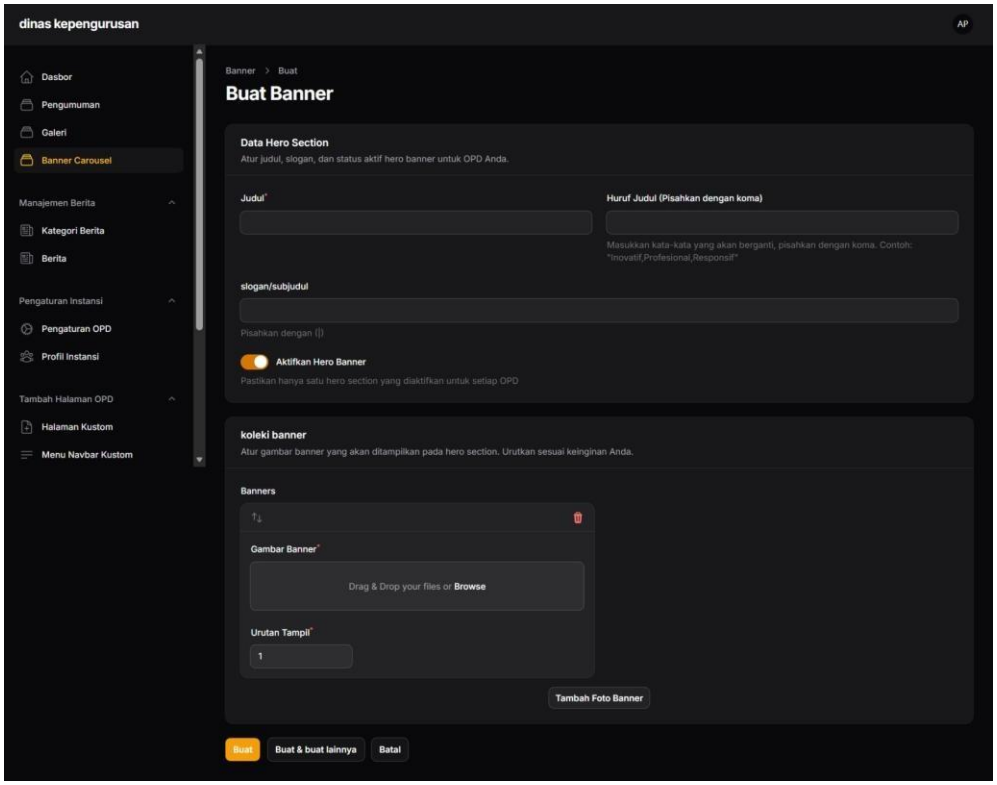
4.1.14 Implementasi Halaman Banner Admin OPD



Gambar 4. 14 Halaman Daftar Banner OPD

Halaman Daftar Banner memperlihatkan tabel daftar banner instansi yang memuat informasi teks judul, pratinjau gambar, status aktif, dan tanggal pembuatan. Pada bagian atas kanan aplikasi terdapat tombol “buat” untuk menambah data banner.

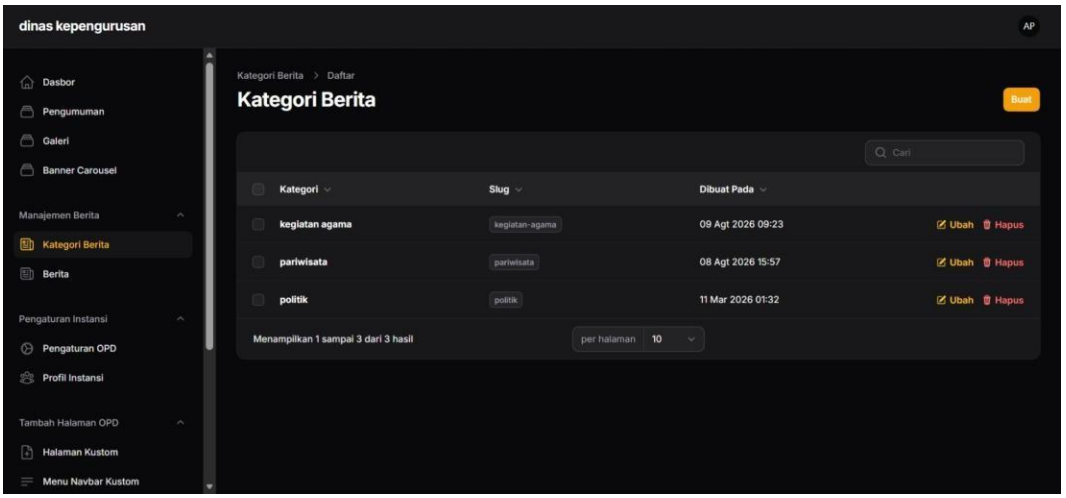
4.1.15 Implementasi Halaman Tambah Banner Admin OPD



Gambar 4. 15 Halaman Tambah Data Banner Admin OPD

Halaman Tambah Data Banner menampilkan formulir pembuatan banner baru. Pada halaman formulir ini admin dapat mengisi judul utama banner, variasi kata, slogan subjudul, serta mengunggah beberapa berkas gambar menggunakan fitur repeater serta penentuan urutan tampil.

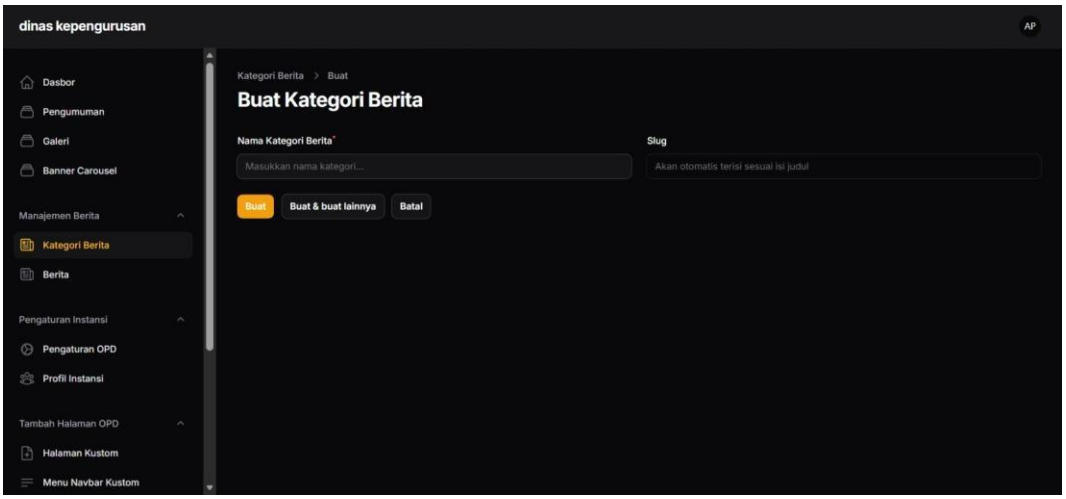
4.1.16 Implementasi Halaman Daftar Kategori Berita Admin



Gambar 4. 16 Halaman Kategori Berita Admin OPD

Halaman Kategori Berita menampilkan tabel daftar kategori berita yang dimiliki oleh OPD terkait. Pada halaman kategori berita terdapat informasi seperti nama kategori, tanggal dipublikasikan, serta terdapat aksi untuk melakukan proses “ubah” dan “hapus” kategori berita.

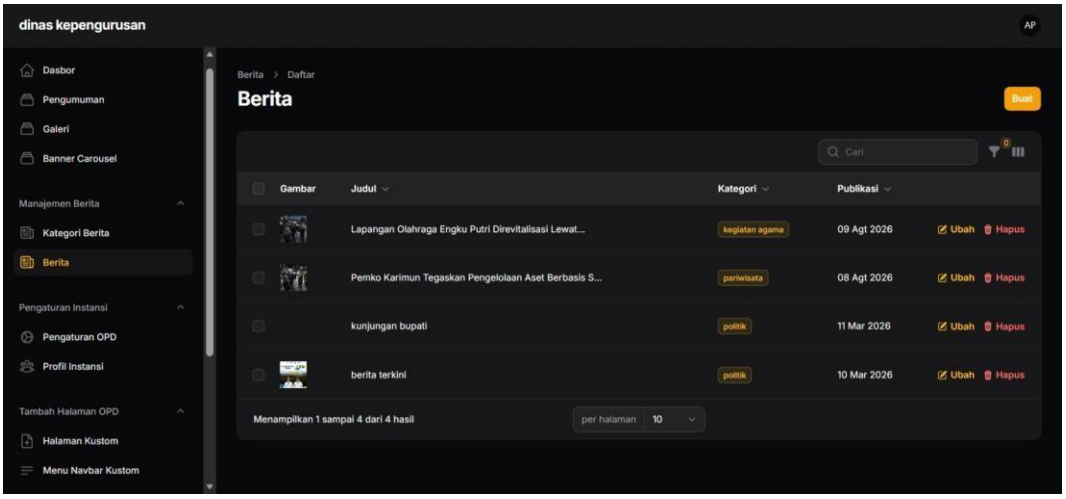
4.1.17 Implementasi Halaman Tambah Kategori Berita Admin



Gambar 4. 17 Halaman Tambah Kategori Berita Admin

Halaman Tambah Kategori Berita menampilkan formulir untuk menambah data kategori berita. Admin dapat mengisi data seperti Nama kategori untuk mengelompokkan data Berita berdasarkan Kategori yang ditambah.

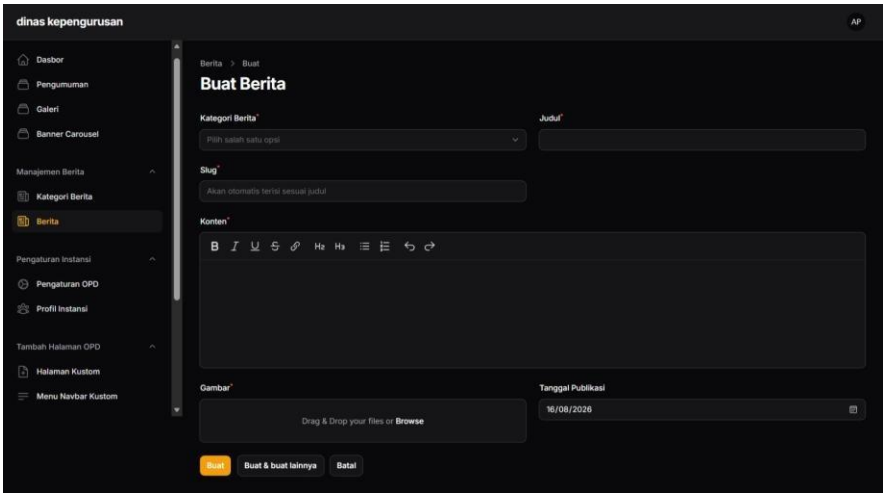
4.1.18 Implementasi Halaman Daftar Berita Admin



Gambar 4. 18 Halaman Berita Admin

Halaman Daftar Berita menampilkan data daftar berita yang dimiliki oleh OPD terkait. Pada tabel berita terdapat informasi berita seperti, judul berita, gambar, kategori berita, dan tanggal publikasi berita.

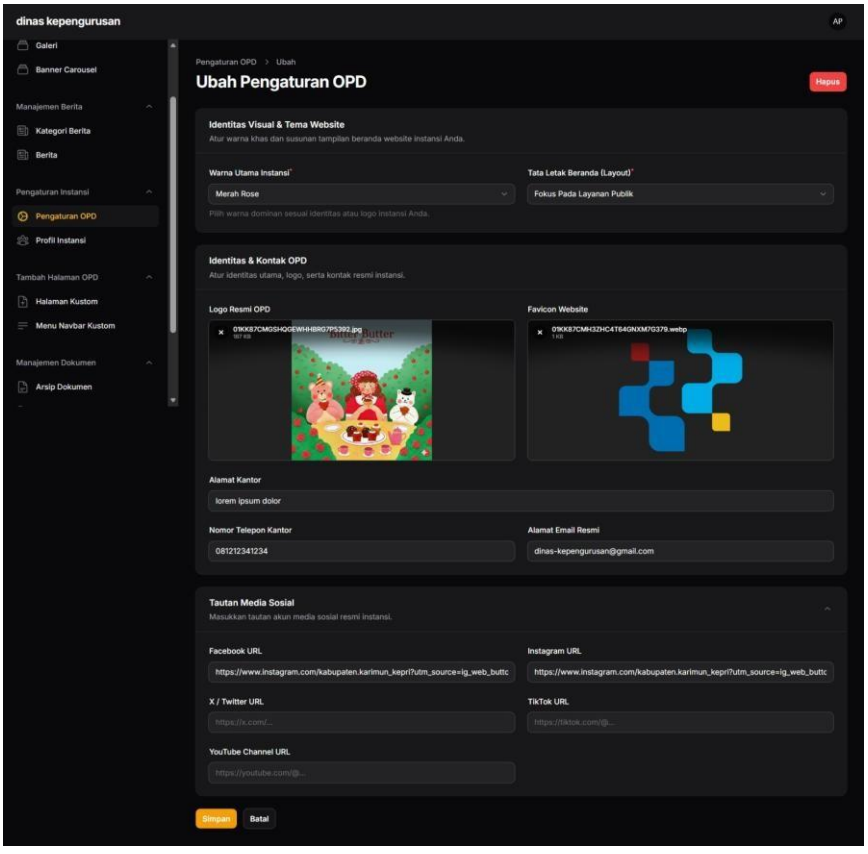
4.1.19 Implementasi Halaman Tambah Data Berita Admin



Gambar 4. 19 Halaman Tambah Data Berita Admin

Halaman Tambah Data Berita menampilkan formulir untuk menambah data berita baru. Pada halaman Tambah Berita admin dapat menambah berita dengan mengisi data seperti judul berita, memilih kategori berita berdasarkan kategori yang dimiliki oleh OPD terkait, tanggal terbit berita, gambar, serta isi berita.

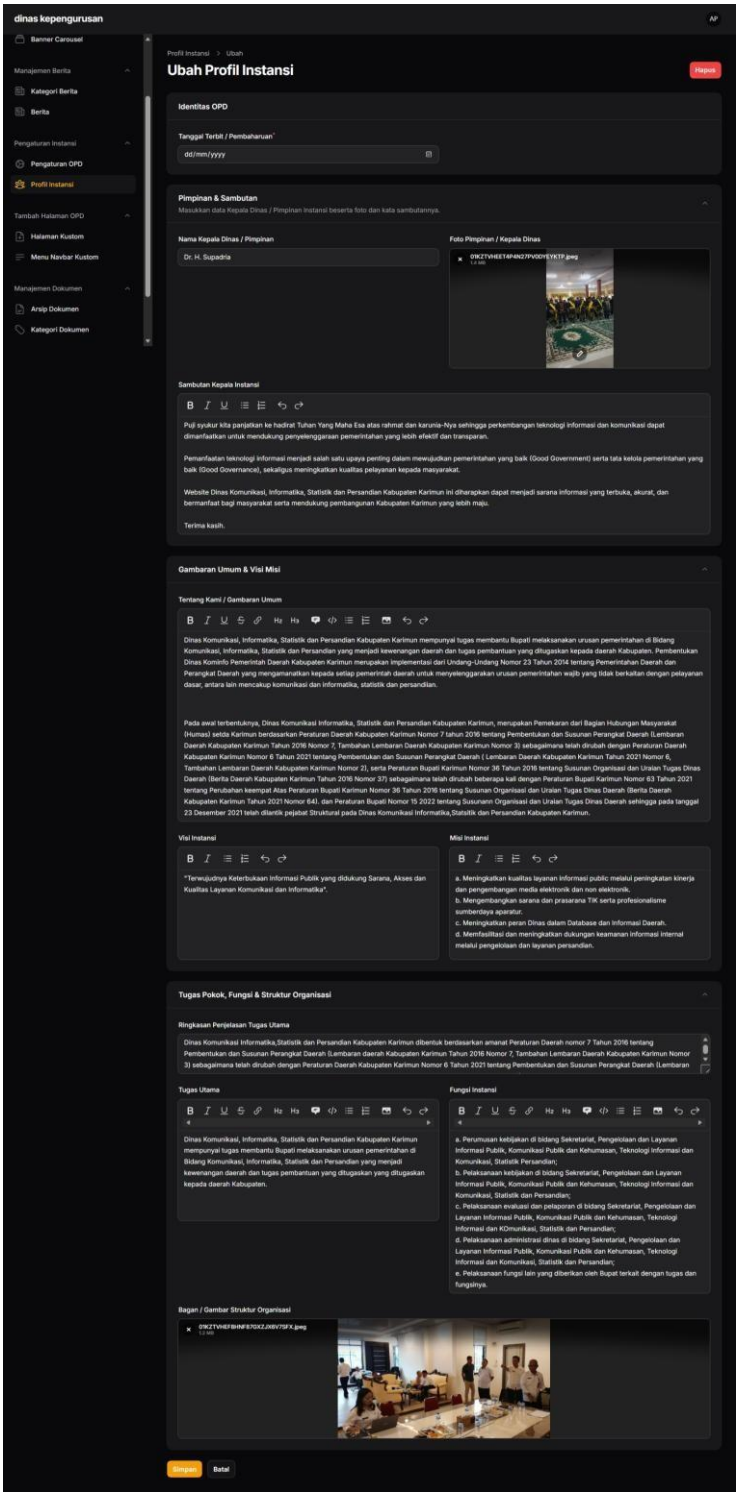
4.1.20 Implementasi Halaman Pengaturan OPD Admin



Gambar 4. 20 Halaman Pengaturan OPD Admin

Halaman Pengaturan OPD admin menampilkan formulir konfigurasi website OPD. Admin dapat memilih tata letak beranda, mengunggah asset foto logo dinas, memperbarui alamat kantor, email, kontak, dan tautan media sosial.

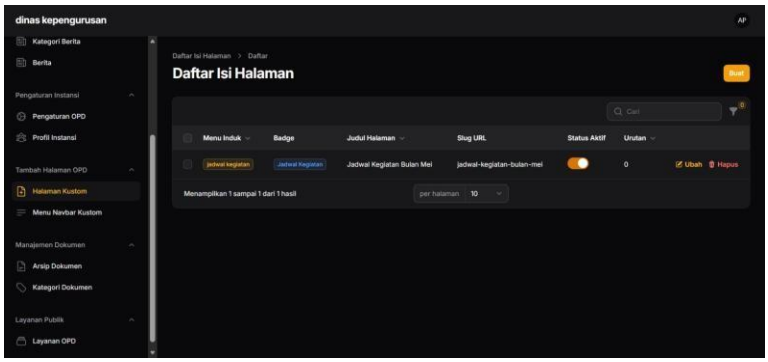
4.1.21 Implementasi Halaman Pengelolaan Profil OPD Admin



Gambar 4. 21 Halaman Pengelolaan Profil OPD

Halaman Pengelolaan Profil OPD menunjukkan formulir pengelolaan profil kelembagaan lengkap. Formulir ini memfasilitasi pengisian profil OPD nama dan foto kepala dinas, kata sambutan, uraian latar belakang instansi, visi dan misi, rincian tugas pokok dan fungsi, serta pengunggahan gambar bagan struktur organisasi kedinasan.

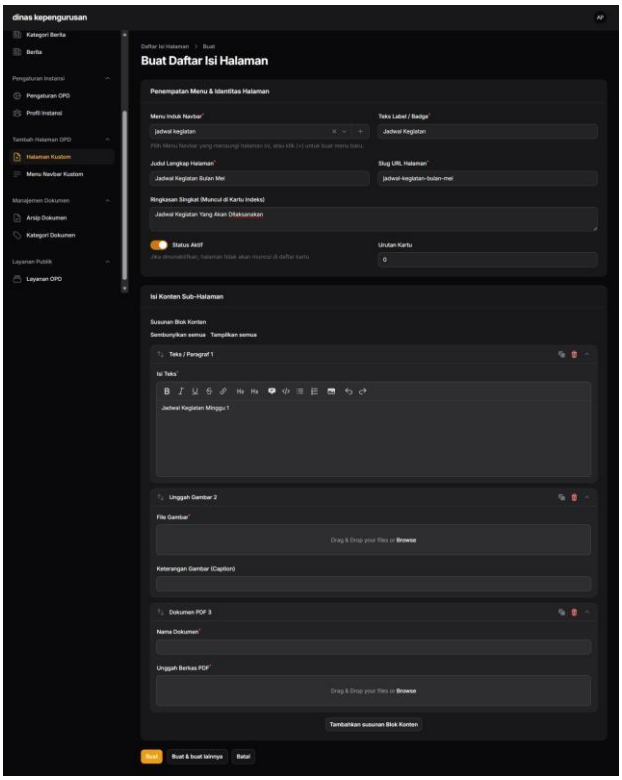
4.1.22 Implementasi Halaman Pengelolaan Konten Baru Admin



Gambar 4. 22 Halaman Pengelolaan Konten Baru

Halaman Pengelolaan Konten Baru menampilkan tabel manajemen sub-halaman kustom yang menampilkan pemetaan menu induk, teks badge, judul halaman, status tayang, dan urutan di kartu indeks dengan default “1”.

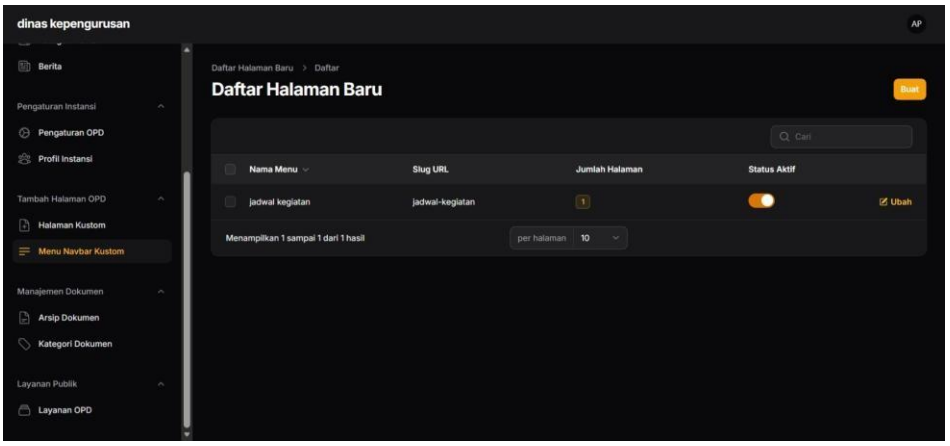
4.1.23 Implementasi Halaman Tambah Konten Halaman Baru Admin



Gambar 4. 23 Halaman Tambah Konten Baru Admin

Halaman Tambah Konten Baru menampilkan formulir penambahan data baru untuk menambah halaman baru. Pada formulir ini admin dapat memetakan halaman ke menu navigasi induknya, mengatur identitas halaman, serta menyusun konten halaman menggunakan blok-blok komponen fleksibel.

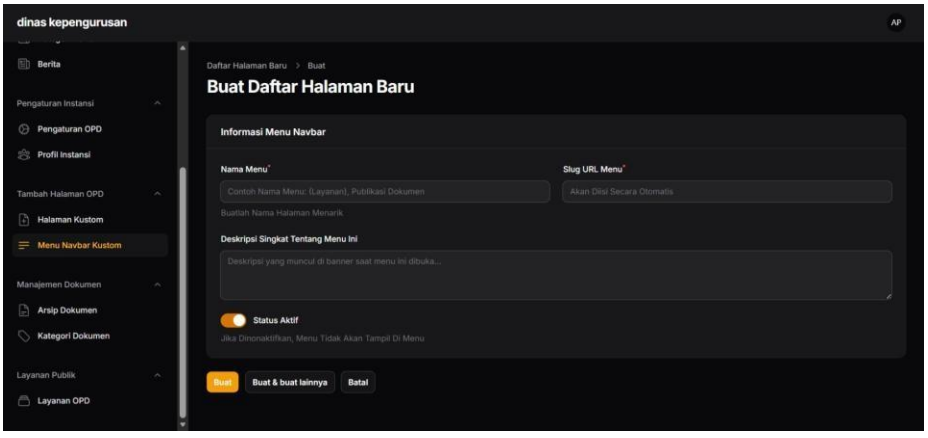
4.1.24 Implementasi Halaman Pengelolaan Daftar Menu Baru Admin



Gambar 4. 24 Halaman Pengelolaan Daftar Menu Baru Admin

Halaman Pengelolaan Daftar Menu Baru menampilkan tabel yang memuat informasi nama menu, jumlah halaman yang tersedia pada menu tersebut, serta status pada menu tersebut. Pada tabel tersebut terdapat aksi untuk melakukan ubah data menu.

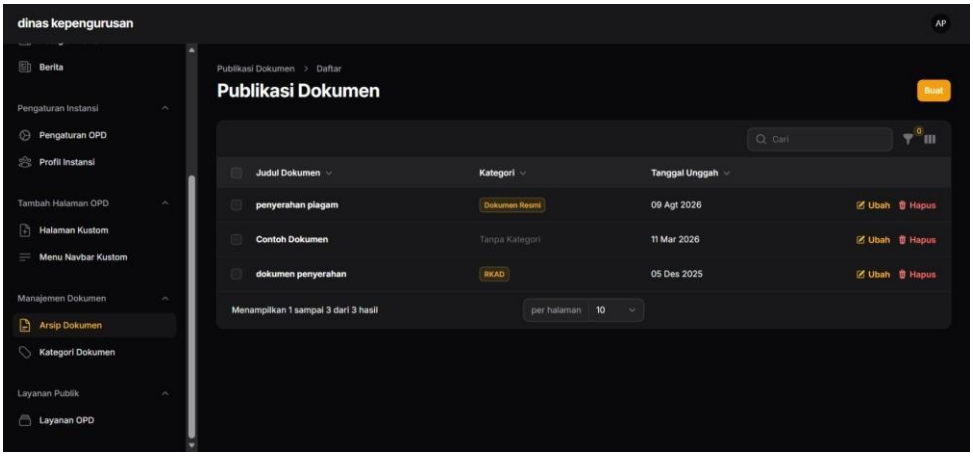
4.1.25 Implementasi Halaman Tambah Menu Baru Admin



Gambar 4. 25 Halaman Tambah Menu Baru Admin

Halaman Tambah Menu Baru menampilkan formulir untuk menambah data menu navigasi baru. Pada formulir ini Admin dapat mengisi kolom nama menu kustom, *slug* URL, deskripsi singkat tentang menu ini, serta status aktivasi menu.

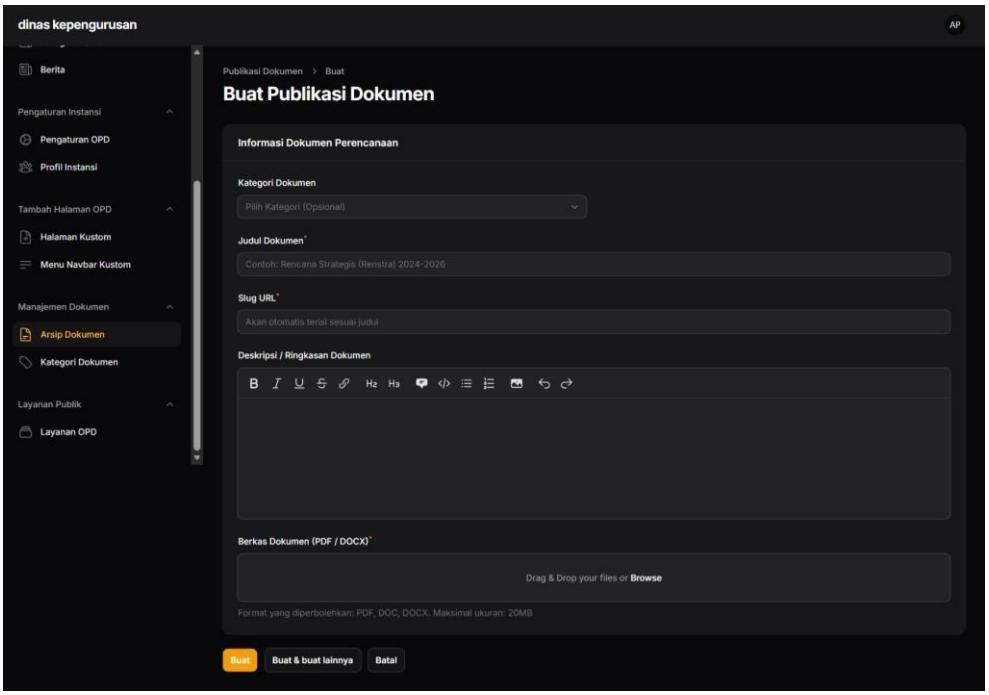
4.1.26 Implementasi Halaman Publikasi Dokumen Admin



Gambar 4. 26 Halaman Publikasi Dokumen Admin

Halaman Publikasi Dokumen menampilkan tabel dokumen yang memuat judul berkas resmi, kategori dokumen, tanggal publikasi, serta tombol aksi untuk melakukan ubah dan hapus data dokumen.

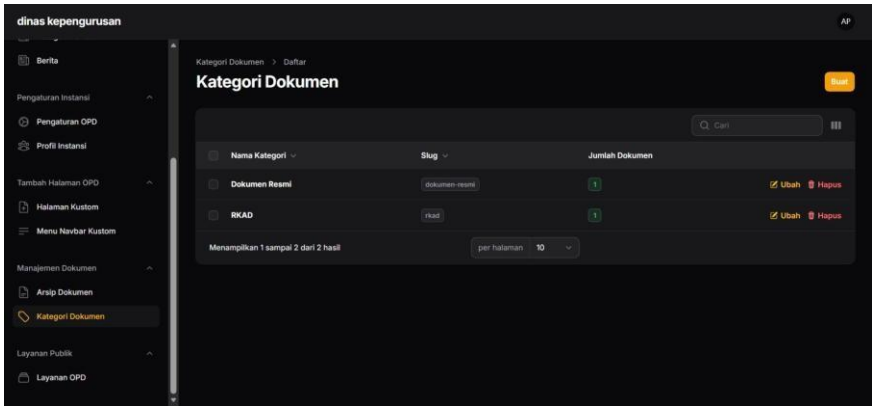
4.1.27 Implementasi Halaman Tambah Dokumen Admin



Gambar 4. 27 Halaman Tambah Dokumen Admin

Halaman Tambah Dokumen menampilkan formulir untuk mengunggah dokumen publik. Pada formulir ini admin dapat memilih jenis kategori dokumen, mengisi judul dokumen, mengisi deskripsi ringkas terkait dokumen, dan mengunggah berkas digital dengan format pdf dengan ukuran berkas maksimal.

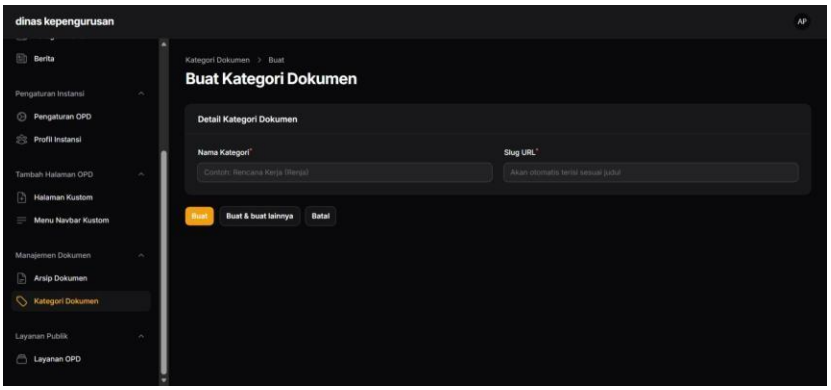
4.1.28 Implementasi Halaman Kategori Dokumen Admin



Gambar 4. 28 Halaman Kategori Dokumen Admin

Halaman Kategori Dokumen menampilkan tabel klasifikasi kategori dokumen seperti nama kategori dokumen, jumlah dokumen yang tersedia berdasarkan kategori dokumen tersebut, serta tombol aksi untuk melakukan ubah dan hapus.

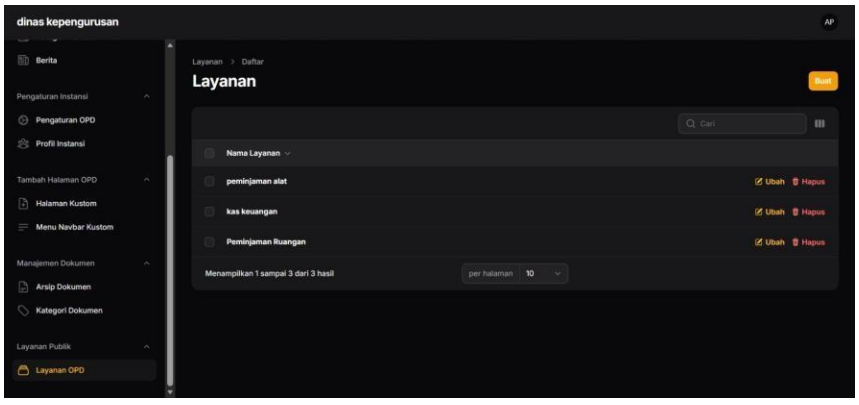
4.1.29 Implementasi Halaman Tambah Kategori Dokumen Admin



Gambar 4. 29 Halaman Tambah Kategori Dokumen Admin

Halaman Tambah Kategori Dokumen menampilkan formulir penambahan kategori arsip dokumen baru dengan kolom input nama kategori dokumen.

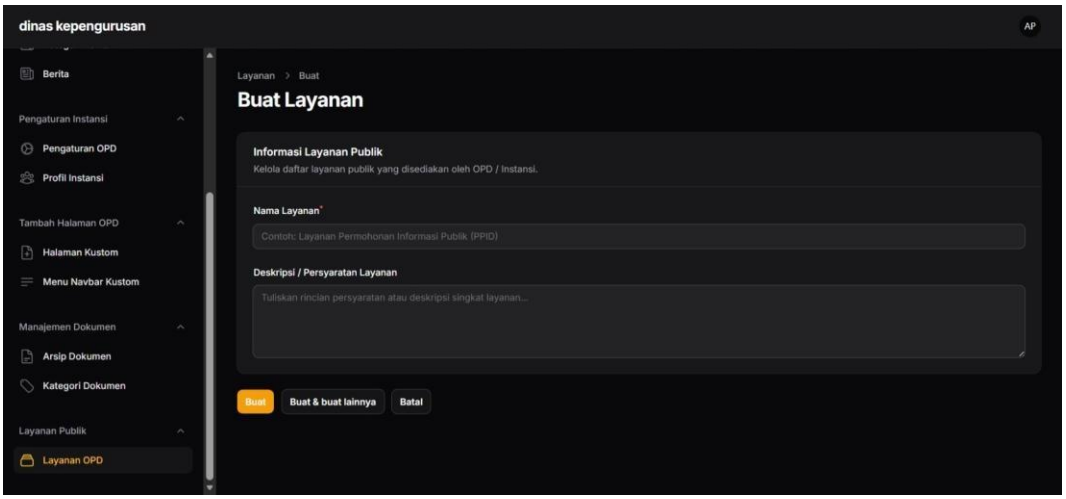
4.1.30 Implementasi Halaman Layanan Admin



Gambar 4. 30 Halaman Layanan Admin

Halaman Layanan menampilkan tabel daftar master layanan publik yang memuat nama program layanan dan tombol aksi pengubahan data dan hapus data layanan.

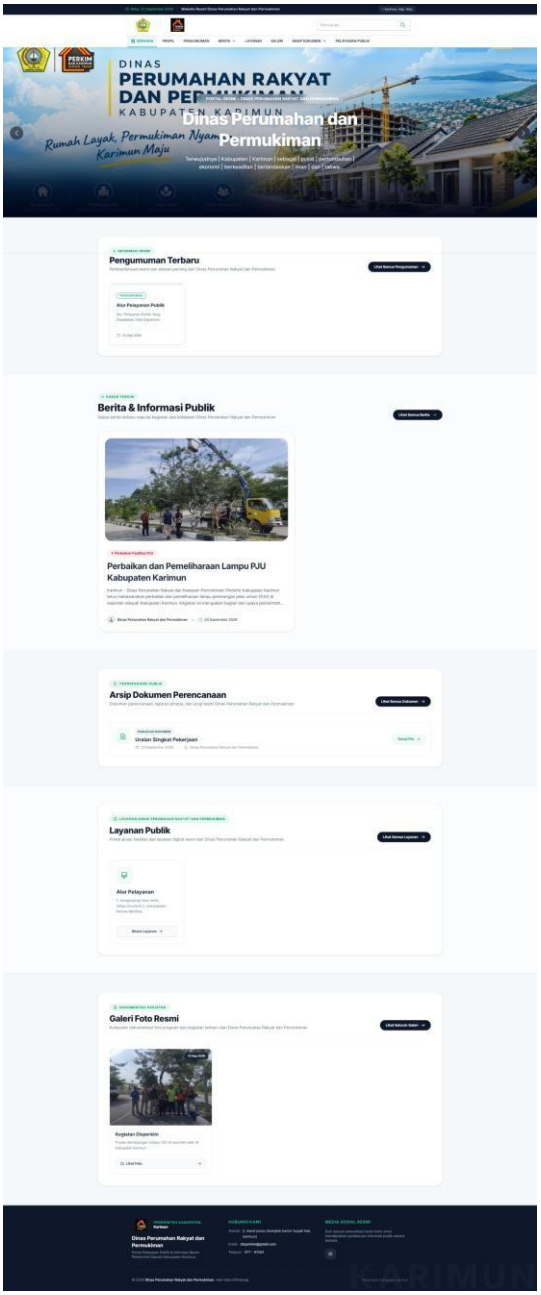
4.1.31 Implementasi Halaman Tambah Layanan Admin



Gambar 4. 31 Halaman Tambah Layanan Admin

Halaman Layanan Admin menampilkan formulir pendaftaran layanan baru dengan kolom isian nama Layanan, serta area penulisan deskripsi atau kriteria persyaratan berkas pengajuan.

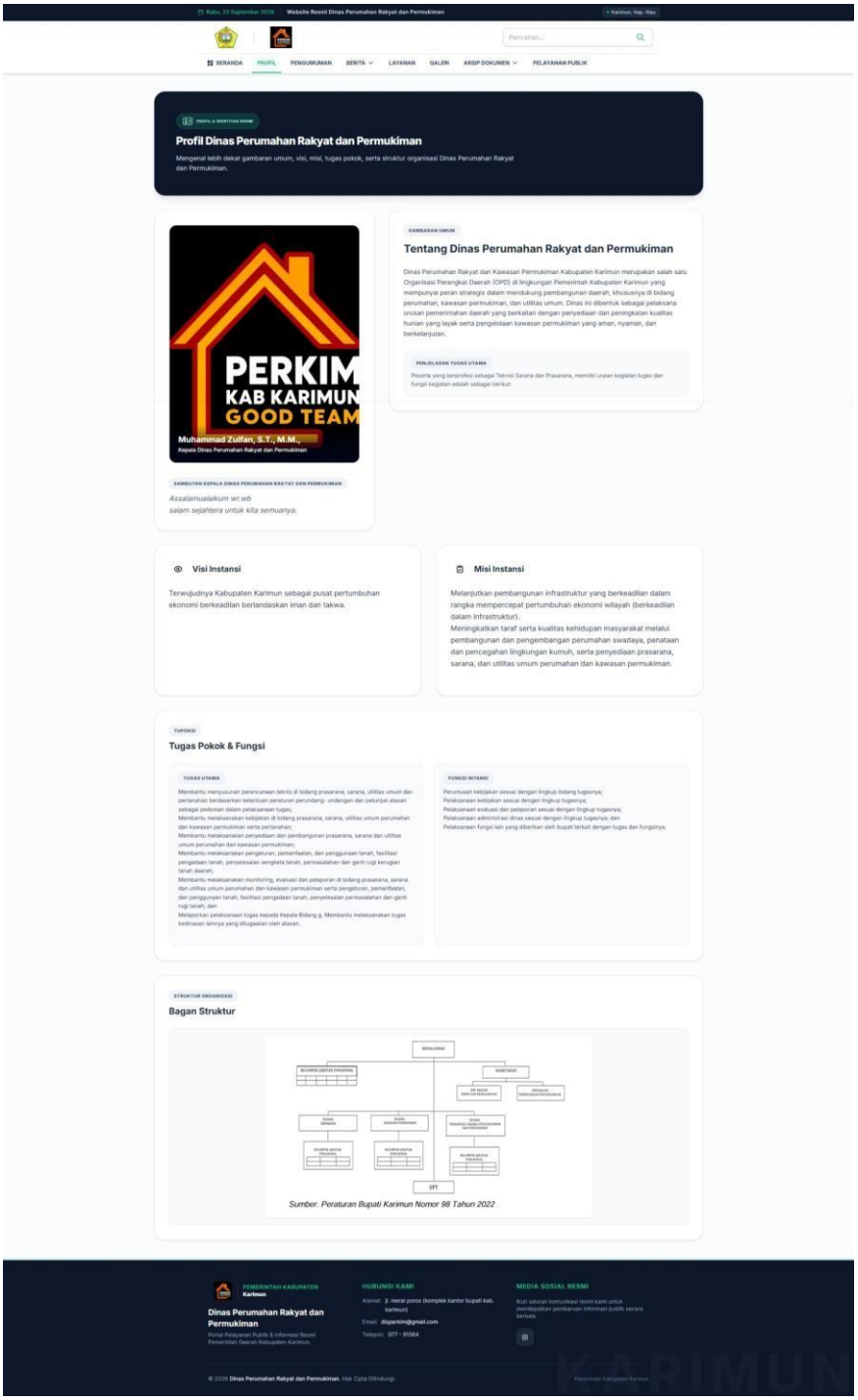
4.1.32 Implementasi Halaman Dasbor Pengguna



Gambar 4. 32 Halaman Beranda Website OPD

Gambar 4.32 menampilkan halaman beranda portal publik OPD yang diakses oleh masyarakat umum. Halaman ini mengintegrasikan seluruh data dinamis yang dikelola oleh Admin OPD, meliputi *header navigasi*, *hero banner slider*, layanan publik unggulan, pengumuman baru, sorotan berita utama, arsip dokumen kinerja, galeri kegiatan dinas, dan informasi kontak atau media sosial instansi pada bagian footer.

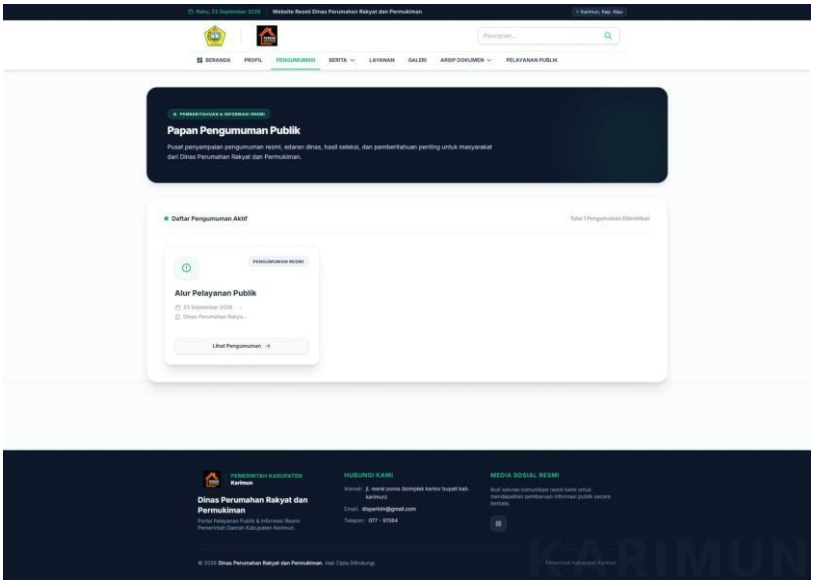
4.1.33 Implementasi Halaman Profil Pengguna



Gambar 4. 33 Halaman Profil OPD

Gambar 4.33 menampilkan antarmuka halaman profil kelembagaan OPD. Halaman ini menyajikan informasi lengkap mengenai pimpinan dinas, sejarah dan gambaran umum pembentukan instansi, visi dan misi instansi, rincian tugas pokok dan fungsi (Tupoksi), serta visualisasi bagan struktur Organisasi Perangkat Daerah.

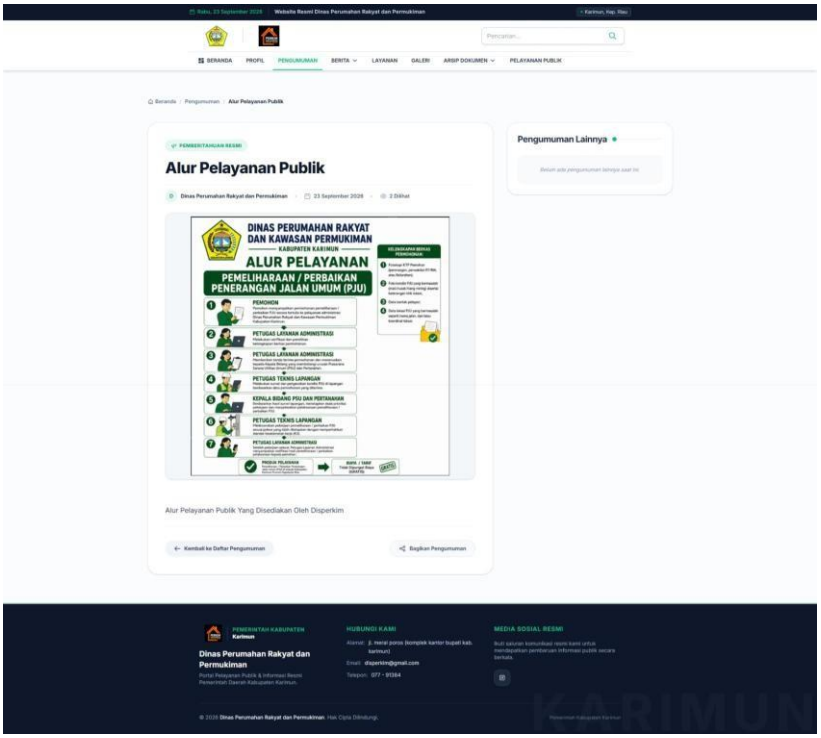
4.1.34 Implementasi Halaman Pengumuman Pengguna



Gambar 4. 34 Halaman Pengumuman OPD

Gambar 4.34 menampilkan antarmuka pengumuman publik pada website OPD. Halaman ini memuat daftar pengumuman resmi kedinasan yang sedang aktif dalam tampilan kartu responsif, yang dilengkapi dengan label judul pengumuman, tanggal *publish*, dan tombol aksi detail.

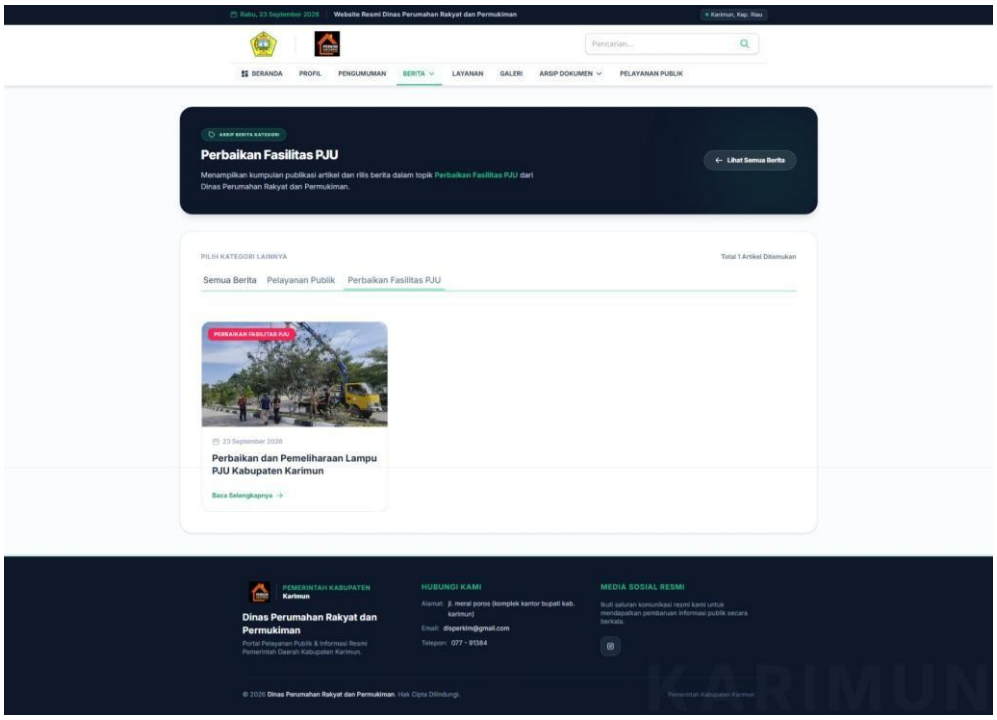
4.1.35 Implementasi Halaman Detail Pengumuman



Gambar 4. 35 Halaman Detail Pengumuman

Gambar 4.35 menampilkan halaman isi detail pengumuman resmi. Halaman ini menyajikan judul lengkap, tanggal terbit, jumlah tayangan pembaca, foto dokumentasi lampiran, isi narasi pengumuman, tombol kembali, serta daftar pengumuman lainnya yang tampil pada bagian bilah samping.

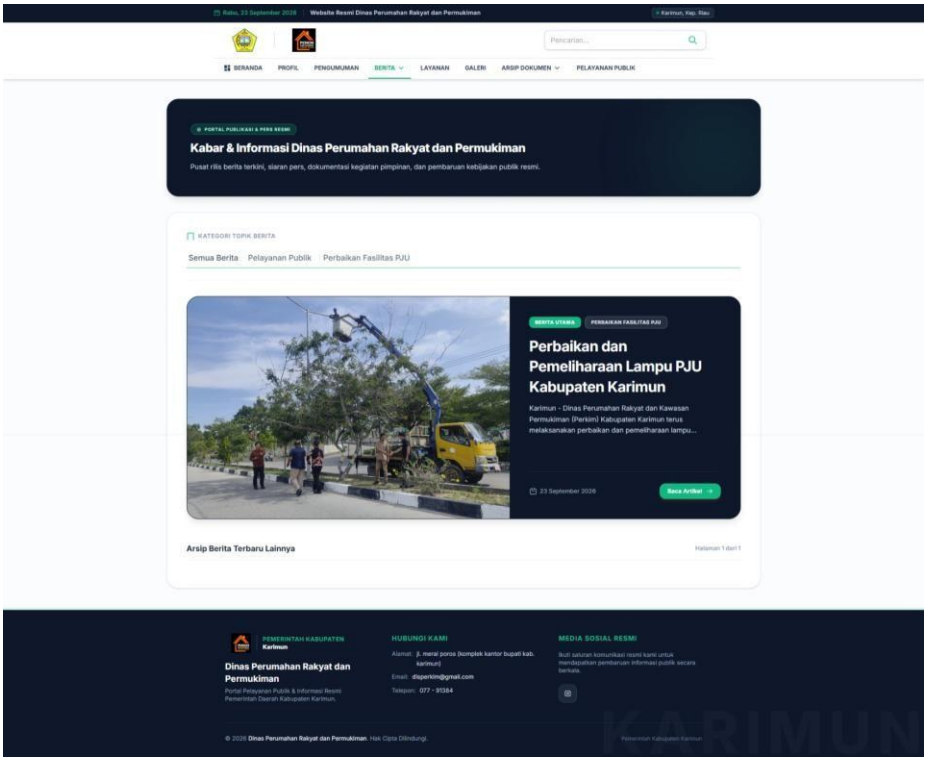
4.1.36 Implementasi Halaman Kategori Berita Pengguna



Gambar 4. 36 Halaman Kategori Berita

Gambar 4.36 menampilkan halaman kategori berita pengguna. Halaman ini menampilkan informasi terkait kategori berita yang tersedia pada halaman berita. Halaman ini mempermudah pengguna dalam mencari berita berdasarkan kategori.

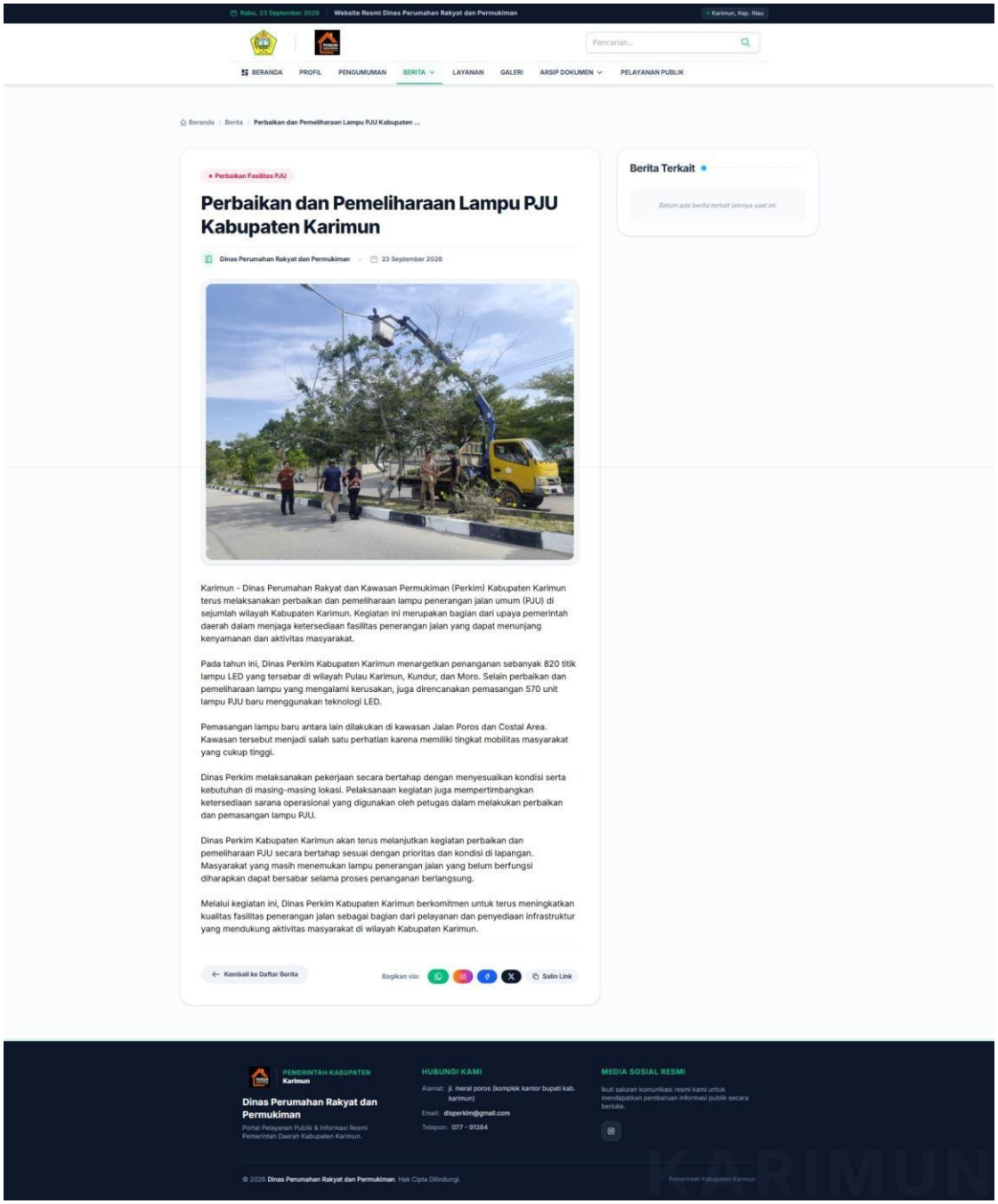
4.1.37 Implementasi Halaman Berita



Gambar 4. 37 Halaman Berita Pengguna

Gambar 4.37 menampilkan halaman portal berita publik yang dilengkapi dengan penyaringan dan pengelompokan berita berdasarkan kategori topik yang tersedia, serta kartu sorotan berita utama.

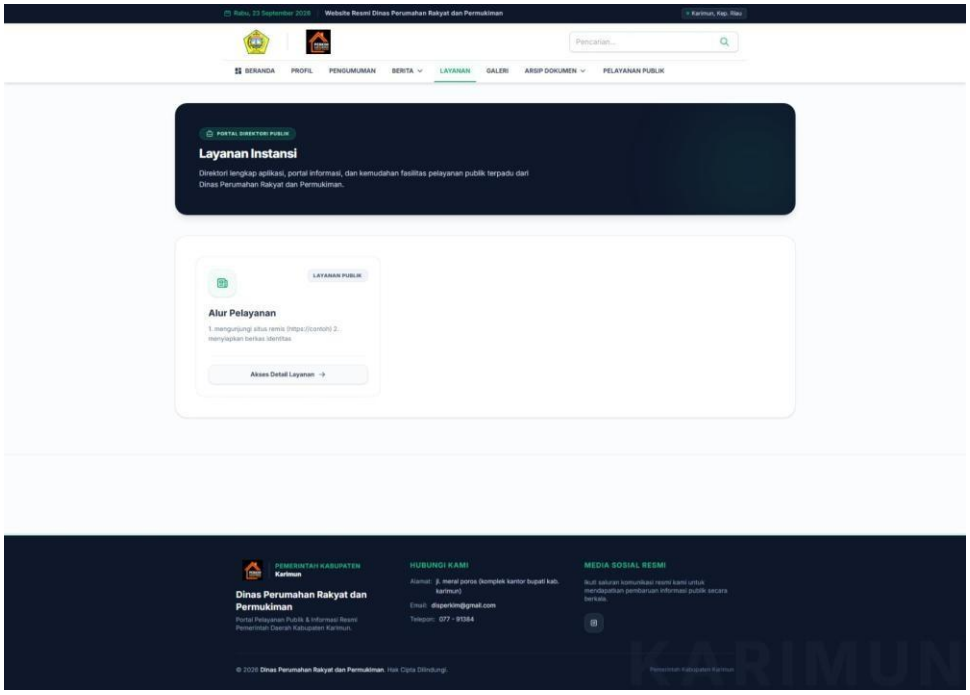
4.1.38 Implementasi Halaman Detail Berita



Gambar 4. 38 Halaman Detail Berita

Gambar 4.38 menampilkan rincian informasi berita. Pada halaman ini terdapat foto utama berita, narasi lengkap artikel, tanggal terbit, serta tombol bagikan ke media sosial, serta rekomendasi artikel berita terkait lainnya.

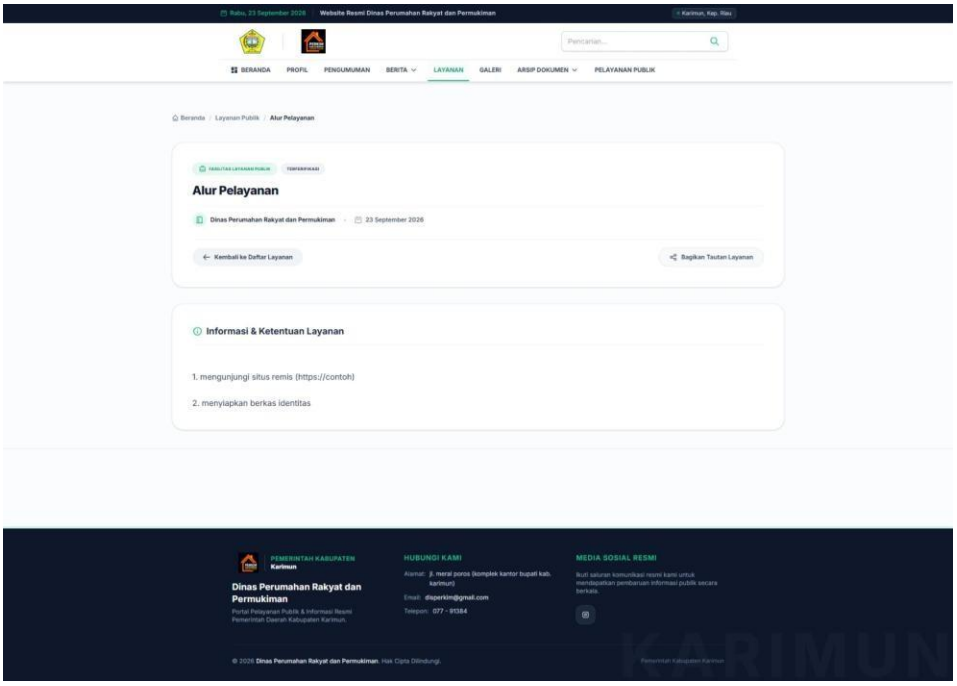
4.1.39 Implementasi Halaman Layanan



Gambar 4. 39 Halaman Layanan

Gambar 4.39 menampilkan halaman layanan pengguna. Pada halaman ini setiap layanan disajikan dalam bentuk kartu ringkas yang memuat nama fasilitas layanan, deskripsi singkat, dan tombol navigasi menuju detail layanan.

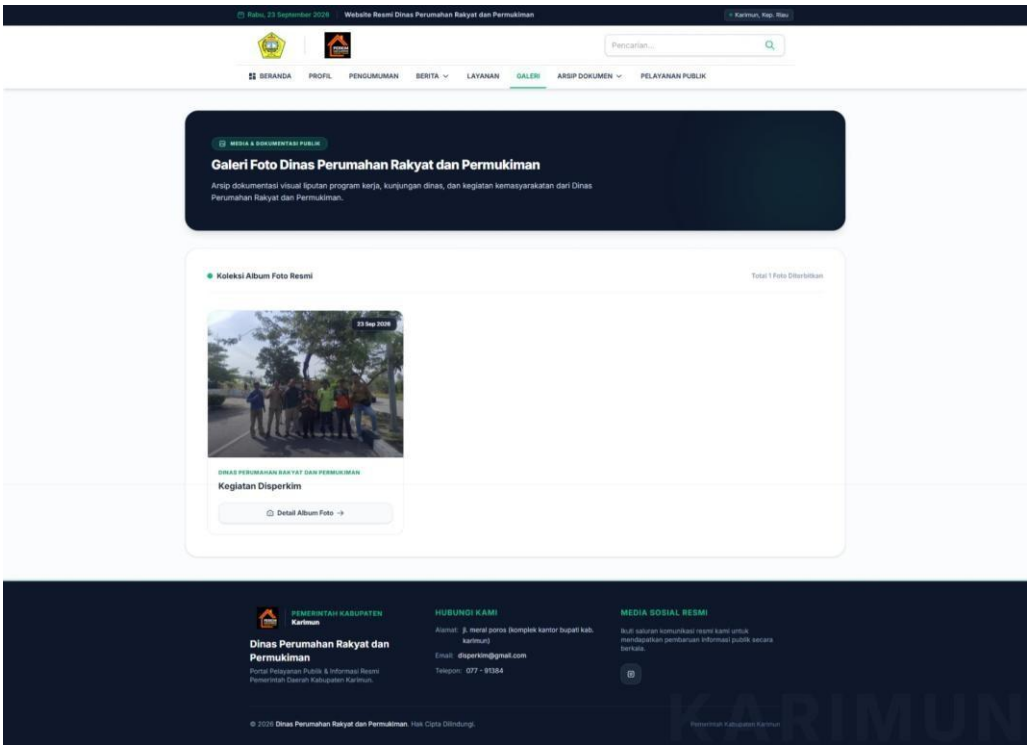
4.1.40 Implementasi Halaman Detail Layanan Pengguna



Gambar 4. 40 Halaman Detail Layanan Pengguna

Gambar 4.40 menampilkan halaman rincian informasi dan petunjuk teknis layanan kedinasan. Halaman ini memuat kartu identitas layanan di bagian atas serta kartu informasi ketentuan yang merinci prosedur mekanisme pengajuan, berkas persyaratan, dan jadwal jam operasional layanan dinas.

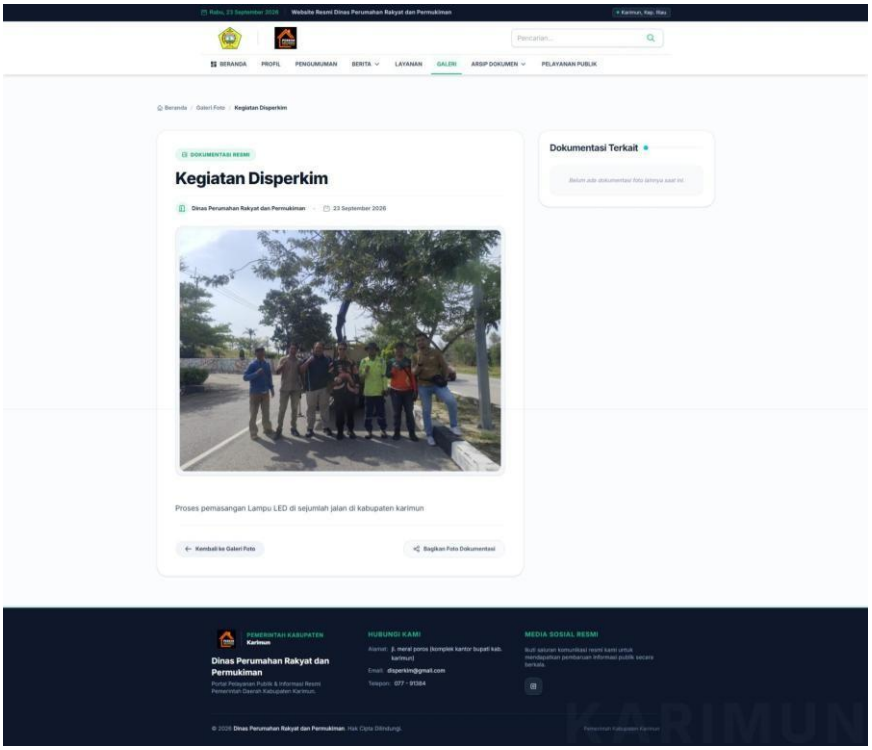
4.1.41 Implementasi Halaman Galeri Pengguna



Gambar 4. 41 Halaman Galeri Pengguna

Gambar 4.41 menampilkan album koleksi dokumentasi visual kegiatan kedinasan. Halaman ini menyajikan kisi kartu album foto resmi lengkap dengan pratinjau gambar, lencana tanggal pelaksanaan kegiatan, dan tombol akses menuju detail album.

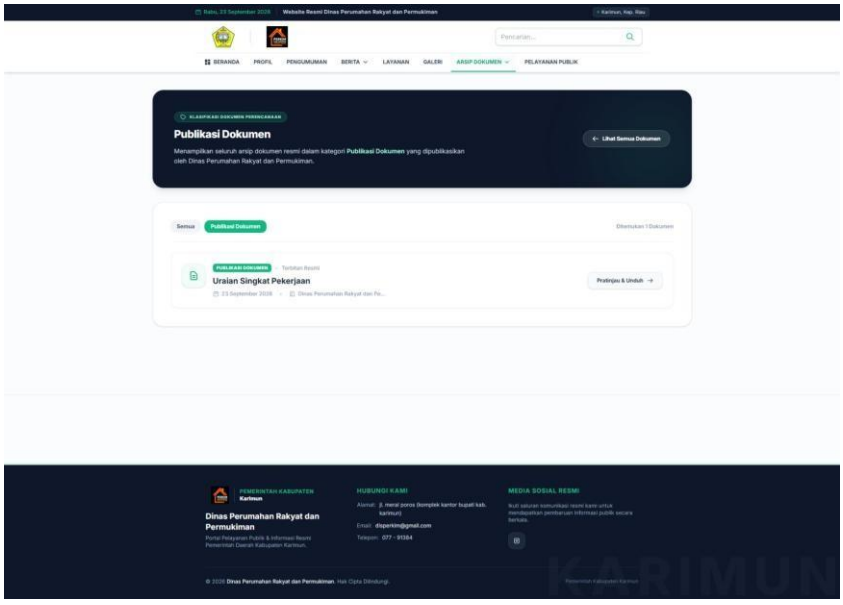
4.1.42 Implementasi Halaman Detail Galeri Pengguna



Gambar 4. 42 Halaman Detail Galeri

Gambar 4.42 menampilkan foto dokumentasi dengan ukuran penuh. Halaman ini dilengkapi dengan *caption* penjelasan kegiatan, tanggal diunggah, tombol bagikan dokumentasi, serta daftar dokumentasi foto lainnya pada bilah samping.

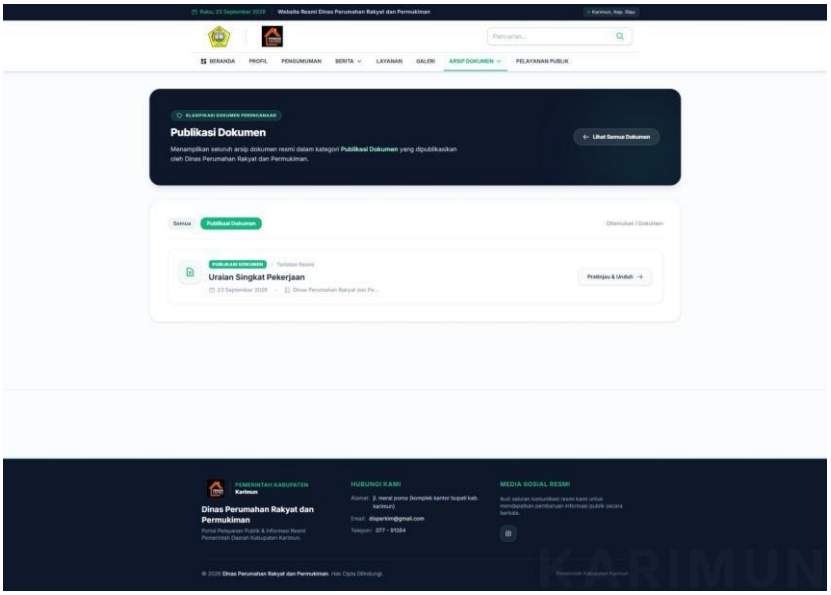
4.1.43 Implementasi Halaman Publikasi Dokumen



Gambar 4. 43 Halaman Publikasi Dokumen

Gambar 4.43 menampilkan repositori data terbuka dokumen perencanaan dan akuntabilitas kinerja instansi. Halaman ini dilengkapi dengan bilah filter berdasarkan kategori dokumen dan tombol unduh berkas dokumen.

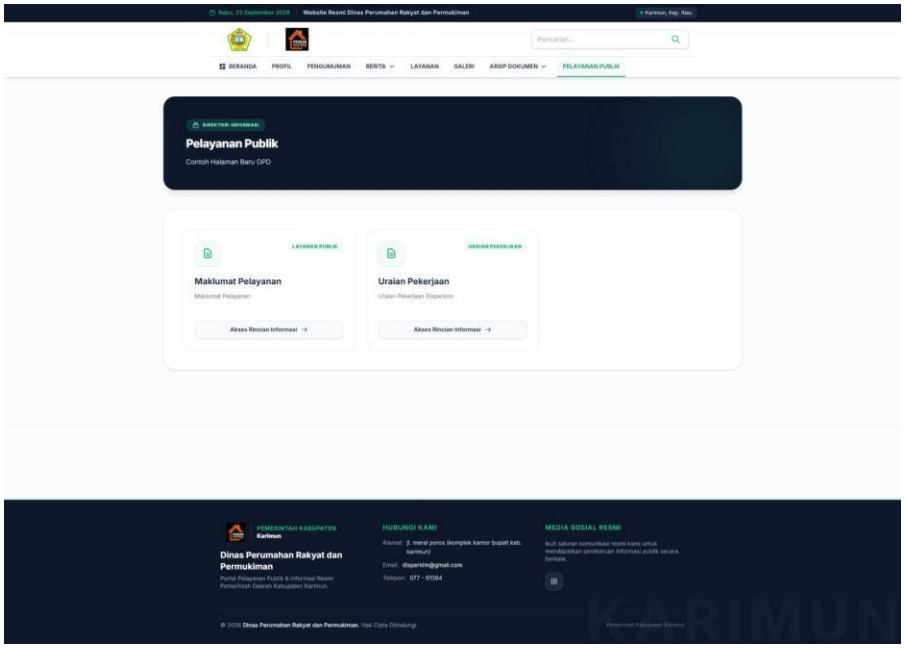
4.1.44 Implementasi Halaman Kategori Dokumen



Gambar 4. 44 Halaman Kategori Dokumen

Gambar 4.44 menampilkan halaman kategori dokumen. Halaman ini berfungsi sebagai pengelompokan data dokumen yang dipisahkan berdasarkan kategori dokumen yang tersedia.

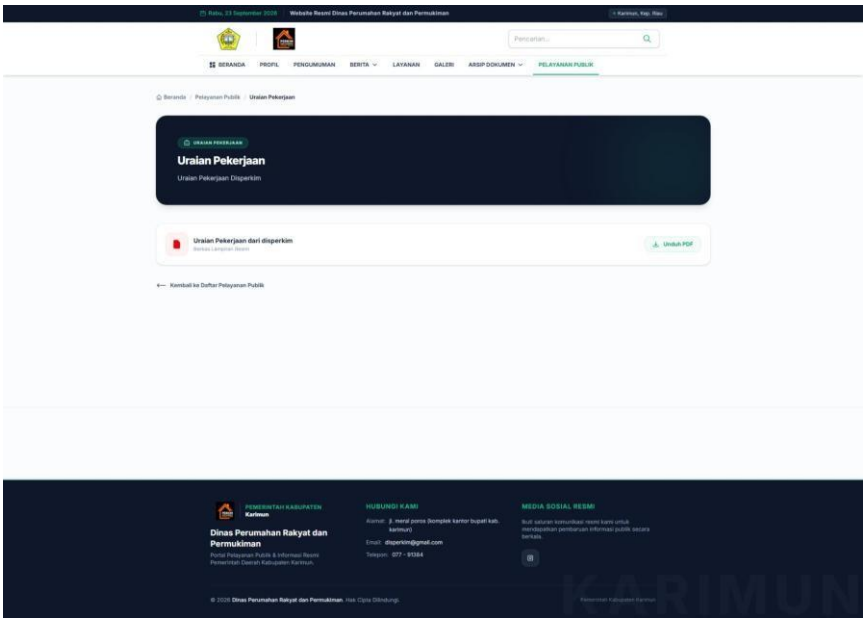
4.1.45 Implementasi Halaman Menu baru Pengguna



Gambar 4. 45 Halaman Menu Baru Pengguna

Gambar 4.45 menampilkan implementasi penayangan modul menu mandiri yang di buat secara dinamis melalui fitur aplikasi. Pada menu tersebut terdapat beberapa data yang dapat di kelola oleh Admin OPD secara dinamis melalui menu pengelolaan menu aplikasi.

4.1.46 Implementasi Halaman Detail Halaman Baru Pengguna



Gambar 4. 46 Halaman Detail Halaman Baru

Gambar 4.46 menampilkan halaman detail konten kustom dinamis yang ditampilkan secara dinamis melalui fitur admin. Tampilan halaman ini dapat memuat konten gambar, teks, dan berkas yang dilengkapi tombol unduh.

4.2. Pengujian *User Acceptance Testing* (UAT)

Pengujian *User Acceptance Testing* (UAT) dilakukan untuk memastikan bahwa seluruh fitur *Web Builder* (Sistem Pengelolaan Website OPD) berbasis Web telah berfungsi sesuai dengan kebutuhan pengguna. Pengujian ini dilakukan oleh Bapak Kamaludin, S. T. sebagai Pembimbing Lapangan dan pengelola Administrator sistem, sebagai representasi Super Admin. Selain itu, pengujian juga dilakukan terhadap perwakilan staf pengelola dari Dinas Perumahan Rakyat dan Permukiman sebagai representasi Admin OPD.

Pada proses pengujian, masing-masing penguji melakukan validasi terhadap fitur sesuai dengan hak akses yang diberikan oleh sistem, seperti login, pembuatan akun pengguna, pembuatan website OPD secara dinamis, pengelolaan konfigurasi secara dinamis, pengelolaan profil OPD, pengelolaan berita, pengelolaan layanan, pengelolaan pengumuman, serta implementasi pengelolaan menu dan halaman baru OPD secara dinamis.

Hasil pengujian kemudian dibandingkan dengan kebutuhan fungsional sistem untuk memastikan bahwa setiap fitur telah berjalan dengan baik dan sesuai dengan yang diharapkan. Hasil *User Acceptance Testing* (UAT) dapat dilihat pada Daftar Lampiran C.

Tabel 4. 1 Tabel Pengujian User Acceptance Test(UAT) Aktor Super Admin

Kode FR	Use Case	Skenario Pengujian	Hasil Yang Diharapkan	Status
KF-01	Melakukan Login	Menginput *username & password* akun Super Admin, lalu memilih Masuk.	Sistem mengautentikasi data dan mengarahkan ke Dashboard Super Admin.	Lulus
KF-02	Super Admin mengelola data akun admin OPD	Menambah, memperbarui, dan menghapus data akun Admin OPD.	Data Akun Admin OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-03	Super Admin mengelola data OPD	Menambah, memperbarui, dan menghapus data OPD.	Data OPD berhasil diproses dan tersimpan dalam basis data.	Lulus

KF-04	Mengelola Peran Pengguna Admin OPD	Menambah, memperbarui, dan menghapus peran Pengguna.	Peran pengguna berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-05	Keluar	Keluar / *Logout* dari aplikasi	Sistem memproses permintaan dan mengarahkan ke halaman Login	Lulus

Tabel 4. 2 Tabel Pengujian User Acceptance Test(UAT) Aktor Admin OPD

	Use Case	Skenario Pengujian	Hasil Yang Diharapkan	Status
KF-01	Melakukan Login	Menginput *username & password* akun Admin, lalu memilih Login.	Sistem mengautentikasi data dan mengarahkan ke Dashboard Admin.	Lulus
KF-02	Mengelola Identitas OPD	Menambah, memperbarui, dan menghapus identitas OPD.	Data Identitas OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-03	Mengelola Data Konten Berita OPD	Menambah, memperbarui, dan menghapus data Berita OPD.	Data Berita OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-04	Mengelola Data Kategori Berita OPD	Menambah, memperbarui, dan menghapus data Kategori Berita OPD.	Data Kategori Berita OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-05	Mengelola Data Galeri OPD	Menambah, memperbarui, dan menghapus data Galeri OPD.	Data Galeri OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-06	Mengelola Data Profil OPD	Menambah, memperbarui, dan menghapus data Profil OPD.	Data Profil OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-07	Mengelola Data Layanan OPD	Menambah, memperbarui, dan menghapus data Layanan OPD.	Data Layanan OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-08	Mengelola Data Publikasi Dokumen OPD	Menambah, memperbarui, dan menghapus data Publikasi Dokumen OPD.	Data Publikasi Dokumen OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-09	Mengelola Data Kategori Dokumen OPD	Menambah, memperbarui, dan menghapus data Kategori Dokumen OPD.	Data Kategori Dokumen OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-10	Mengelola Data Bannner OPD	Menambah, memperbarui, dan menghapus data Banner OPD.	Data Banner OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-11	Mengelola Data Tambah Halaman Baru OPD	Menambah, memperbarui, dan menghapus data halaman OPD.	Data Halaman Baru OPD berhasil diproses dan tersimpan dalam basis data.	Lulus
KF-12	Mengelola Data Tambah Menu baru pada halaman OPD	Menambah, memperbarui, dan menghapus data tambah menu baru pada halaman OPD.	Data Menu Halaman Baru OPD berhasil diproses dan tersimpan dalam basis data.	Lulus

BAB V KESIMPULAN

5.1 Kesimpulan

Berdasarkan seluruh tahapan pengembangan yang meliputi analisis kebutuhan, perancangan, implementasi, hingga pengujian terhadap aplikasi *Web Builder* (Sistem Pengelolaan Website OPD) berbasis web pada Dinas Komunikasi Informatika Statistik dan Persandian Kabupaten Karimun, diperoleh beberapa kesimpulan yang dirumuskan berdasarkan tujuan yang telah ditetapkan, yaitu:

1. Aplikasi *Web Builder* berbasis web telah berhasil dirancang dan dibangun untuk memfasilitasi staf Organisasi Perangkat Daerah (OPD) di Kabupaten Karimun dalam membuat, mengelola, dan mempublikasikan konten website resmi instansi secara dinamis.
2. Implementasi Framework Laravel yang dipadukan dengan panel administrasi Filament dan basis data MySQL terbukti efektif menciptakan sistem manajemen konten (*Content Management System*) yang intuitif dan responsif. Sistem ini berhasil menyediakan fitur standar pengelolaan profil instansi, publikasi berita beserta kategorinya, penyebaran pengumuman resmi, dokumentasi galeri foto, arsip dokumen publik, direktori informasi layanan kedinasan, hingga pembuatan halaman dan menu navigasi kustom yang mampu meningkatkan efisiensi operasional staf OPD di Kabupaten Karimun.

5.2 Saran

Berdasarkan hasil pelaksanaan proyek akhir ini, terdapat beberapa saran yang dapat dijadikan pertimbangan untuk pengembangan sistem lebih lanjut maupun implementasi di lapangan:

1. Menambahkan fleksibilitas pada modul pembangunan halaman berupa fitur penyusunan tata letak berbasis *drag-and-drop* serta variasi tema visual (*template*) beranda yang lebih beragam agar setiap OPD dapat menyesuaikan tampilan portalnya secara beragam.
2. Membangun antarmuka pemrograman aplikasi untuk memfasilitasi pertukaran data secara otomatis dengan sistem pemerintahan digital lain, seperti sistem informasi pelayanan publik terpusat lainnya (*e-government*).

3. Menerapkan modul *Activity Log Audit Trail* yang lebih mendalam untuk merekam riwayat perubahan konten oleh staf admin, serta mengaktifkan autentikasi dua faktor pada panel login untuk memperketat standar keamanan data instansi pemerintah daerah.

DAFTAR PUSTAKA

Christian, C., & Voutama, A. (2024). RANCANG BANGUN APLIKASI SISTEM INFORMASI INVENTARIS BERBASIS WEBSITE. *Jurnal Informatika Dan Teknik Elektro Terapan*, *12*(2), 1500–1509. http://dx.doi.org/10.23960/jitet.v12i2.4259

Pamungkas, R., & Azis, Muh. N. L. (2022). Implementasi Content Management System Wordpress Sebagai Sistem Informasi Publik Berbasis Web (Studi Kasus Website Kelurahan Bendo Kabupaten Magetan). *UNIMPA Press*, 15–16. https://eprint.unipma.ac.id/287/1/BUku%20Pak%20Rido%20Terbaru.pdf

Melyani, R. I., Rosita, & Aji, S. (2023). Pengembangan Sistem Informasi Penggajian Berbasis Web Menggunakan Framework Laravel dengan Metode Agile Software Development. *Jurnal Sistem Informasi Akuntansi (JASIKA)*, *3*(1), 31–36. https://doi.org/10.31294/jasika.v3i01.2195

Arif, S. M. (2023). Pembuatan website informasi & pendaftaran webinar umum menggunakan PHP & MYSQL. *Journal of Information System, Applied, Management, Accounting and Research*, *7*(3), 789-796. http://journal.stmikjayakarta.ac.id/index.php/jisamar

Sudarsono, E., & Vebriandi, M. Y. (2025). Implementasi Framework Laravel Filament pada Sistem CRM untuk Optimalisasi Data Pelanggan dan Program Loyalitas Poin di Toko Branding Telemarco. *Journal of Innovation And Future Technology*, *7*(1), 23-34. https://ejournal.lppm-unbaja.ac.id/index.php/iftech/article/download/3839/1831

Athallah, M. I., & Ayunda, A. T. (2026). Model Arsitektur Multi-Tenant Berbasis Role-Based Access Control pada Sistem Manajemen Waralaba. *Progresif: Jurnal Ilmiah Komputer*, *22*(1), 39-49. http://dx.doi.org/10.35889/progresif.v22i1.3374

Mozin, S. Y., & Abdussamad, S. N. (2026). Efektivitas Kinerja Organisasi Pada Perangkat Daerah. *Jurnal Kolaboratif Sains*, *9*(1), 1711-1717. https://jurnal.unismuhpalu.ac.id/index.php/JKS/article/download/10365/6969

Supriadi, E., Nurcahyo, W., & Faizah, N. M. (2025). Perancangan aplikasi sistem informasi wisata alam di Kota Pandeglang, Provinsi Banten, berbasis web dengan metode waterfall menggunakan PHP dan MySQL. *Jurnal Ilmu Komputer Dan Teknologi Informasi*, *2*(1), 22-32. https://doi.org/10.35870/jikti.v2i1.1342

Fathurrohim, A., Iskandar, J., & Sari, Y. K. (2025). Pengembangan sistem *Content Management System* (CMS) website pondok pesantren menggunakan framework Laravel. *Jurnal Informatika dan Teknik Elektro Terapan (JITET), 13*(3). http://dx.doi.org/10.23960/jitet.v13i3.7070

Surfina, J., & Ma'sum, H. (2026). Pengembangan website Pejabat Pengelola Informasi dan Dokumentasi berbasis WordPress. *Jurnal Manajemen Informatika Jayakarta,* *6*(2), 119–126. https://doi.org/10.52362/jmijayakarta.v6i2.2341

Abdillah, I. A. (2025). Rancang Bangun dan Pengembangan Website E-Katalog Multi-Tenant Katalogqu untuk Mendukung Transformasi Digital Bisnis menggunakan Laravel Berbasis Qrcode di PT. Era Cipta Digital [Laporan

kerja praktik, Universitas Medan Area]. Digital Repository Universitas Medan Area.
https://repositori.uma.ac.id/handle/123456789/28840

Fitrianto, D., & Sulistyo, W. Y. (2026). Penerapan Role-Based Access Control untuk Keamanan Data Multi-Tenant SIMPEL Lazismu. *Insect (Informatics and Security): Jurnal Teknik Informatika*, *12*(01), 1-10. https://doi.org/10.33506/insect.v12i01.5009

DAFTAR LAMPIRAN

Lampiran A. Dokumen Proses Pengumpulan Requirement

https://drive.google.com/drive/folders/1ITtGmOTVmkhmlojLKmiLscxcyFB4joVJ?usp=sharing

Lampiran B. Link Produk

https://github.com/farhannajedi/builder-opd-app.git

Lampiran C. Dokumen Pengujian UAT

https://drive.google.com/drive/folders/1U9n9TnLO5ihXhORraoCu7CyutCmh75cu?usp=sharing