

BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Selama periode magang di Lembaga Konseling Keluarga Kreatif, posisi yang ditempati adalah sebagai IT Specialist pada divisi Media & IT dengan lingkup tugas pada pembangunan fitur blog di website. Pemberian tugas dilakukan oleh supervisor, dengan arahan dan instruksi kerja yang jelas sesuai kebutuhan proyek. Selama proses pengerjaan, progres pekerjaan dilaporkan secara berkala kepada supervisor guna memastikan proyek berjalan sesuai target dan standar kualitas yang telah ditentukan. CEO LK3 juga hadir di kantor sebanyak satu hingga tiga kali dalam seminggu untuk mengadakan sesi koordinasi bersama supervisor dan tim. Kehadiran ini bertujuan untuk memastikan bahwa arah kerja tim tetap sejalan dengan visi organisasi dan bahwa setiap bagian proyek dikawal dengan baik.

3.2 Tugas yang Dilakukan

Tabel 3.1. Internship Timeline Summary

Aktivitas	Status	Timeline - Start	Timeline - End
Blog list publik	Done	2025-04-01	2025-04-05
Blog detail viewer	Done	2025-04-06	2025-04-09
Menampilkan blog unggulan (featured)	Done	2025-04-10	2025-04-12
AJAX Search + Pagination publik	Done	2025-04-13	2025-04-15
CRUD Blog (Create, Edit, Delete)	Done	2025-04-16	2025-04-20
Upload gambar (cover image)	Done	2025-04-21	2025-04-22
Fitur pencarian admin + grup tanggal	Done	2025-04-23	2025-04-25
Pratinjau blog sebelum publish (preview)	Done	2025-04-26	2025-04-28
Pengujian, perbaikan bug, refactor	Done	2025-04-29	2025-05-01
Dokumentasi dan penyesuaian UI/UX	In Progress	2025-05-02	In Progress

Selama pelaksanaan kerja magang di Lembaga Konseling Keluarga Kreatif, berbagai tugas dalam pembangunan fitur di website telah dilakukan. Berikut adalah rincian tugas yang diberikan dalam proyek utama berdasarkan tabel 3.1:

1. Blog *list* publik

Pembuatan halaman utama blog yang menampilkan daftar artikel secara publik. Hal ini menjadi langkah awal untuk membangun kanal informasi yang dapat diakses oleh pengguna umum.

2. Blog detail *viewer*
Pengembangan halaman detail untuk setiap entri blog yang memungkinkan pengguna membaca isi lengkap dari artikel beserta metadata seperti tanggal, penulis, dan *cover image*.
3. Menampilkan blog unggulan (*featured*)
Implementasi logika untuk memilih dan menampilkan entri blog yang memiliki engagement tinggi (*views*, *shares*, dan kebaruan) sebagai konten unggulan di bagian *carousel* atau rekomendasi.
4. AJAX Search dan *Pagination* publik
Pengembangan fitur pencarian blog menggunakan AJAX agar hasil pencarian ditampilkan secara dinamis tanpa *me-refresh* halaman, termasuk sistem *pagination* yang mendukung kelancaran navigasi data.
5. CRUD blog
Implementasi fungsionalitas backend untuk membuat, mengedit, dan menghapus entri blog oleh admin, serta menyediakan tampilan antarmuka yang mendukung proses tersebut.
6. *Upload* gambar (*cover image*)
Penambahan fitur unggah gambar untuk setiap entri blog dengan validasi ukuran dan format *file*, serta penyimpanan path gambar di direktori publik.
7. Fitur pencarian admin dan grup tanggal
Pembuatan fitur pencarian pada halaman admin blog, lengkap dengan pengelompokan entri berdasarkan waktu publikasi seperti “Hari ini”, “Kemarin”, atau tanggal tertentu.
8. Pratinjau blog sebelum *publish* (*preview*)
Penambahan fitur preview agar admin dapat melihat tampilan blog sebelum dipublikasikan ke publik, memastikan isi dan tampilan telah sesuai.
9. Pengujian, perbaikan *bug*, *refactor*
Tahap evaluasi untuk memastikan sistem berjalan dengan baik, disertai perbaikan *bug*, pengoptimalan kode, dan refactoring guna menjaga kualitas proyek.
10. Dokumentasi dan penyesuaian UI/UX
Proses penyusunan dokumentasi teknis dan revisi desain antarmuka agar

tampilan dan pengalaman pengguna sesuai dengan standar kenyamanan dan kebutuhan lembaga. Saat ini masih dalam tahap pengerjaan (*In Progress*).

3.3 Uraian Pelaksanaan Magang

Tabel 3.2. Rangkuman Kegiatan Mingguan Selama Magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Mempelajari kebutuhan yang dibutuhkan & Mempersiapkan hal-hal untuk pembangunan aplikasi dengan Android Studio.
2	Melakukan transkrip video dari kelas pembelajar.
3	Melakukan transkrip video dari kelas pembelajar.
4	Membuat UI/UX aplikasi.
5	Membuat database aplikasi di Google Sheet serta mempelajari apa saja yang dibutuhkan untuk fitur blog di website LK3.
6	Membangun backend dari aplikasi, namun belum termasuk ke konfigurasi API dan database.
7	Melanjutkan fitur blog ke tahap pembuatan UI di Figma.
8	Memulai pembuatan UI/UX admin blog dengan menggunakan Laravel.
9	Melakukan transkrip video dari kelas pembelajar.
10	Melanjutkan pembuatan UI/UX untuk blog dan melakukan review terkait aplikasi bersama dengan supervisor.
11	Membuat backend untuk admin blog .

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

Tabel 3.2. Rangkuman Kegiatan Mingguan Selama Magang (Lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
12	Membuat backend untuk blog.
13	Melakukan revisi untuk admin blog.
14	Melakukan revisi untuk blog.
15	Melakukan input data kehadiran data pembelajar.
16	Mempersiapkan data yang dipakai untuk kemudian dikelola agar menemukan lembaga, organisasi masyarakat, sekolah, universitas, maupun gereja yang dapat diajak bekerja sama.
17	Melakukan input data terkait lembaga, organisasi masyarakat, sekolah, universitas, maupun gereja yang dapat diajak bekerja sama.
18	Membantu membuat surat untuk lembaga, organisasi masyarakat, sekolah, universitas, maupun gereja yang dapat diajak bekerja sama.
19	Membantu membuat surat untuk lembaga, organisasi masyarakat, sekolah, universitas, maupun gereja yang dapat diajak bekerja sama.
20	Melakukan input data untuk data-data pembelajar yang ikut acara "Your Mental Health Your Future".
21	Melakukan input data untuk data-data pembelajar yang ikut acara "Your Mental Health Your Future".
22	Melakukan input data untuk data-data pembelajar yang ikut acara "Your Mental Health Your Future".
23	Menyelesaikan pembangunan aplikasi, terutama dengan bagian API.

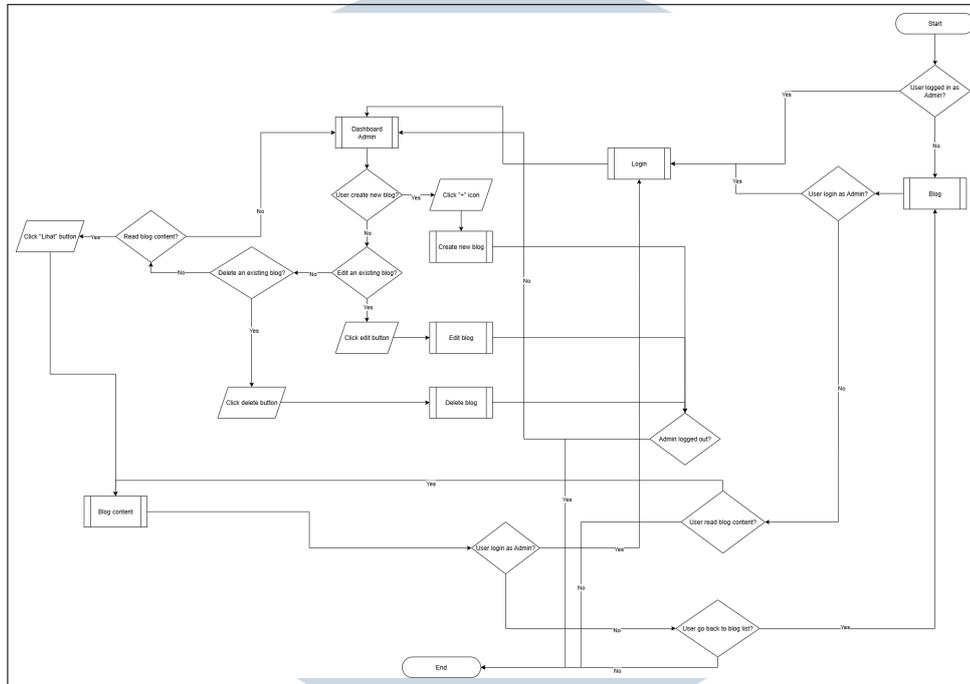
3.3.1 User Requirement

Berikut adalah kebutuhan pengguna (*user requirement*) yang dirancang berdasarkan kebutuhan lembaga serta fungsionalitas sistem blog yang diharapkan:

1. Administrator dapat membuat, mengedit, dan menghapus artikel blog. Sistem harus menyediakan antarmuka untuk mengelola konten blog secara fleksibel melalui fitur CRUD (*Create, Read, Update, Delete*).

2. Administrator dapat membaca dan menelusuri artikel berdasarkan tanggal publikasi. Blog harus menampilkan artikel secara kronologis dan memungkinkan Administrator menjelajah artikel dari kategori waktu seperti "Hari ini", "Kemarin", atau berdasarkan tanggal tertentu.
3. Sistem menyediakan fitur pencarian artikel secara real-time. Diperlukan fitur pencarian berbasis *AJAX* yang memungkinkan pengguna dan administrator menemukan artikel tanpa perlu memuat ulang halaman.
4. Sistem dapat menandai artikel sebagai unggulan. Artikel tertentu dapat diprioritaskan untuk tampil di bagian atas halaman (Hero section dan Featured Post) berdasarkan skor tertentu.
5. Administrator dapat mengunggah gambar sampul dan konten secara dinamis. Sistem harus mendukung fitur unggah gambar langsung dari editor untuk melengkapi artikel blog.
6. Pengguna publik dapat membaca artikel blog. Sistem harus menyajikan konten blog yang dapat diakses secara terbuka oleh masyarakat umum tanpa hambatan otentikasi.
7. Pengguna publik dapat menelusuri artikel unggulan dan terbaru. Halaman publik blog harus menampilkan bagian "Featured Post" dan daftar artikel terbaru agar pengguna mudah menemukan konten prioritas dan terbaru.
8. Pengguna publik dapat membaca deskripsi awal artikel sebelum membuka secara lengkap. Tampilan awal blog harus menyajikan cuplikan konten (preview) dari setiap artikel agar pengguna dapat menilai relevansi artikel sebelum membaca secara penuh.
9. Pengguna publik dapat melakukan pencarian berdasarkan judul atau isi artikel. Fitur pencarian berbasis *AJAX* harus tersedia di halaman publik untuk membantu pengguna menemukan konten secara cepat dan relevan.

3.3.2 Alur Fitur Blog



Gambar 3.1. Diagram Alur Sistem Blog

Alur kerja fitur blog pada Gambar 3.1 secara umum terbagi menjadi dua bagian utama, yaitu alur untuk administrator dan alur untuk pengguna publik. Berikut adalah penjelasan masing-masing alur berdasarkan diagram alir (flowchart):

1. Alur Administrator: Ketika seorang admin mengakses halaman login, langkah pertama adalah verifikasi status login sebagai admin. Jika belum login, maka admin akan diarahkan ke halaman *Login*. Setelah berhasil melakukan autentikasi, admin akan masuk ke halaman *Dashboard Admin*. Dari dashboard, admin dapat melakukan berbagai aktivitas manajemen blog:
 - (a) Jika admin ingin membuat blog baru, maka ia akan memilih opsi *Create new blog* dan mengklik ikon "+". Setelah itu, admin diarahkan ke halaman *Create new blog* untuk mengisi informasi artikel yang akan dipublikasikan.
 - (b) Jika ingin mengedit blog yang sudah ada, admin memilih opsi *Edit an existing blog*, lalu diarahkan ke halaman *Edit blog*.

- (c) Jika ingin menghapus blog yang ada, admin memilih opsi *Delete an existing blog*, kemudian menekan tombol *Delete*. Blog akan dihapus dari sistem.

Setelah selesai melakukan aksi manapun, admin dapat kembali ke dashboard. Jika admin memilih untuk logout, maka sistem akan mengakhiri sesi dan proses dinyatakan selesai.

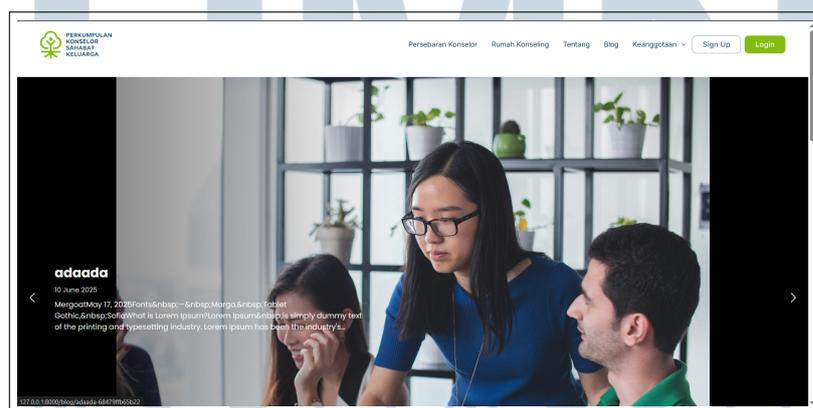
- 2. Alur Pengguna Publik: Jika pengguna tidak login sebagai admin, maka fitur blog akan langsung menampilkan halaman *Blog Publik*. Di halaman ini, pengguna dapat:

- (a) Menelusuri daftar artikel blog yang tersedia.
- (b) Mengklik tombol "*Lihat*" pada salah satu entri blog untuk membaca isi lengkap artikel tersebut.

Setelah pengguna memilih untuk membaca artikel, mereka akan diarahkan ke halaman *Blog Content* untuk melihat isi lengkap dari blog yang dipilih. Proses selesai setelah pengguna selesai membaca artikel.

3.3.3 Blog - Implementasi UI/UX

1. Tampilan Publik Blog



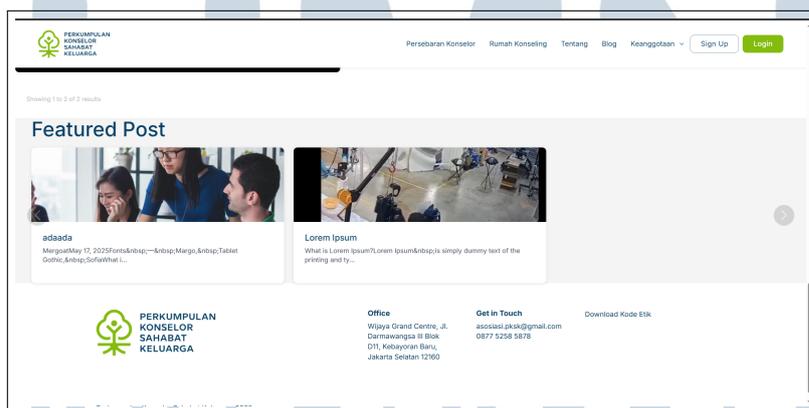
Gambar 3.2. Carousel Blog

Tampilan pada gambar 3.2 menunjukkan bagian atas halaman blog publik yang menampilkan fitur *carousel* atau *slider*. Carousel ini digunakan untuk menampilkan blog unggulan (*highlighted content*) secara visual dan interaktif. Blog-blog yang tampil di carousel dipilih berdasarkan skor gabungan dari jumlah *views*, *shares*, dan waktu publikasi (semakin baru, semakin tinggi skornya). Fitur ini memanfaatkan library *Swiper.js* untuk navigasi geser secara otomatis maupun manual.



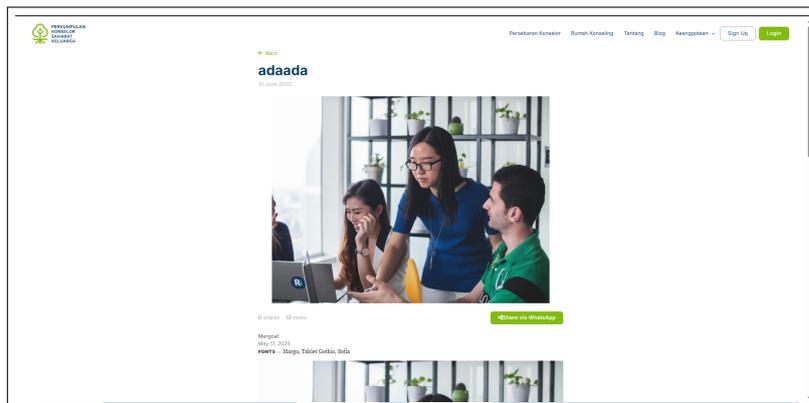
Gambar 3.3. Dashboard Blog

Gambar 3.3 menampilkan daftar blog terbaru secara kronologis, diurutkan berdasarkan waktu. Setiap blog ditampilkan dalam bentuk card, dilengkapi dengan gambar, judul, dan tombol "Read More". Pada gambar 3.2 juga terdapat search bar berbasis AJAX dan sistem pagination yang mendukung navigasi antar halaman tanpa perlu reload halaman penuh. Semua data ditampilkan dari basis data yang terhubung melalui controller LandingBlogController.



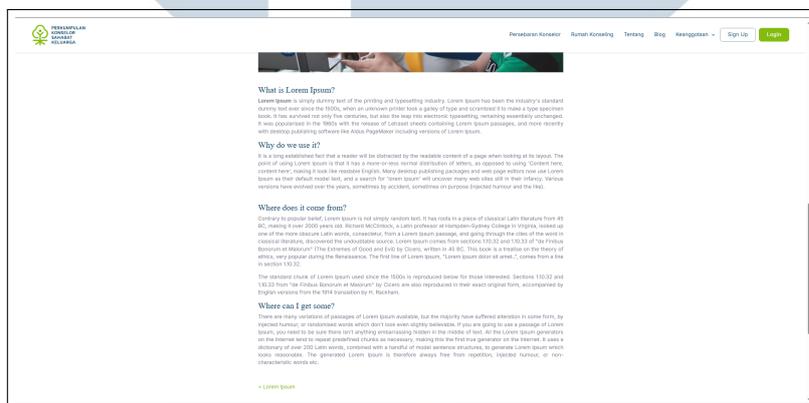
Gambar 3.4. Featured Post

Bagian pada gambar 3.4 menampilkan featured posts, yaitu blog-blog yang mendapatkan skor tertinggi (berdasarkan kombinasi views, shares, dan waktu publikasi). Tampilan di gambar 3.3 berada di bagian bawah halaman utama blog dan dirancang untuk menarik perhatian pengguna terhadap konten-konten pilihan. Tampilan disusun dalam bentuk card horizontal dan dilengkapi dengan slider navigasi.



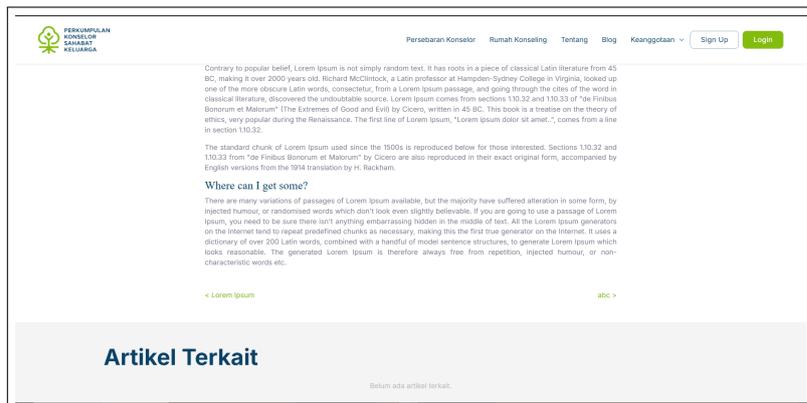
Gambar 3.5. Blog Content(1)

Gambar 3.5 merupakan tampilan awal halaman detail sebuah blog. Judul blog ditampilkan dengan ukuran besar, diikuti oleh tanggal publikasi. Di bawahnya, ditampilkan gambar cover blog secara penuh, sesuai dengan gambar yang diunggah oleh admin saat membuat blog. Terdapat pula informasi jumlah views dan shares, serta tombol untuk membagikan blog ke WhatsApp menggunakan integrasi ke URL WA.me.



Gambar 3.6. Blog Content(2)

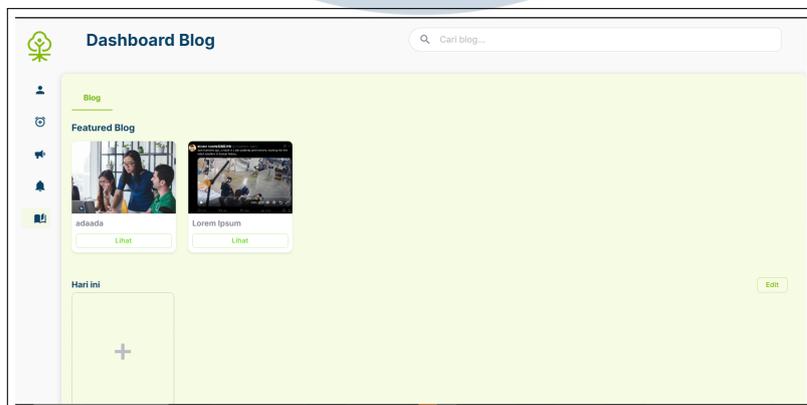
Melanjutkan dari gambar 3.6 sebelumnya, gambar 3.5 memperlihatkan isi konten blog yang telah diformat menggunakan *rich text editor*. Konten ditampilkan menggunakan sintaks Blade directive `{!! $post->content !!}` yang memungkinkan HTML dari editor untuk dirender secara langsung di halaman. Tampilan ini mendukung struktur paragraf, heading, hyperlink, serta penyisipan gambar, sehingga mempermudah admin dalam menyusun artikel blog yang kaya dan terstruktur secara visual.



Gambar 3.7. Blog Content(3)

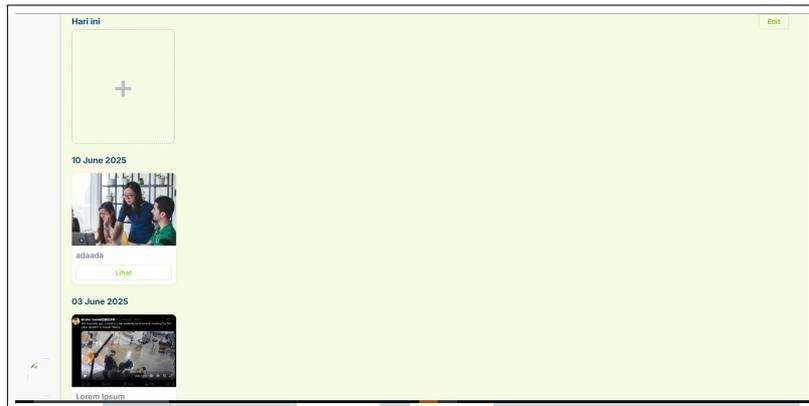
Gambar 3.7 yaitu berupa tampilan lanjutan dari konten blog yang berada di bagian akhir artikel. Di sini juga terlihat bagian navigasi antar artikel, yakni tombol "Previous" dan "Next" untuk berpindah ke blog sebelumnya atau berikutnya, sesuai dengan urutan ID di database. Di bagian paling bawah terdapat bagian "Artikel Terkait" yang menampilkan blog lain dengan kata kunci judul serupa, sebagai hasil dari algoritma related post berdasarkan kemiripan teks pada judul blog.

2. Dashboard Admin Blog



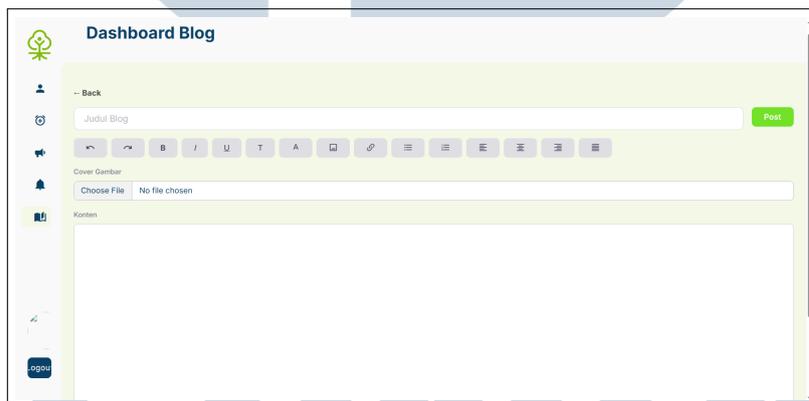
Gambar 3.8. Dashboard Admin(1)

Tampilan di gambar 3.8 merupakan halaman awal dari dashboard blog untuk admin. Pada bagian kiri layar terdapat panel navigasi yang memungkinkan admin berpindah ke berbagai modul lainnya seperti blog, pengaturan, dan logout. Di bagian tengah, terdapat daftar blog yang telah dipublikasikan, termasuk fitur "Featured Blog" di bagian atas. Blog yang termasuk featured dipilih berdasarkan skor tertinggi dan ditandai secara visual untuk membedakannya dari blog biasa.



Gambar 3.9. Dashboard Admin(2)

Gambar 3.9 memperlihatkan daftar lengkap blog yang ditampilkan berdasarkan waktu publikasi, dengan tanggal diurutkan dari yang terbaru ke terlama. Tiap entri blog ditampilkan dalam bentuk kartu (card) yang berisi judul, tanggal, dan gambar sampul. Tersedia pula tombol "+", yang digunakan untuk menambahkan blog baru ke sistem. Tampilan 3.8 juga mendukung tampilan responsif dan interaktif.



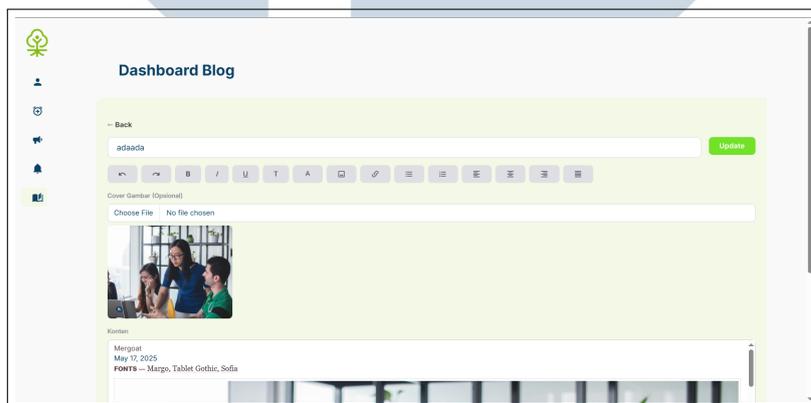
Gambar 3.10. Create Blog

Gambar 3.10 ialah tampilan halaman pembuatan blog baru. Tersedia form input untuk judul blog, pilihan cover image, dan konten blog. Editor teks disesuaikan dengan kebutuhan penulisan rich text layaknya Microsoft Word, dengan toolbar untuk styling seperti bold, italic, underline, pengaturan font, insert link, dan insert image. Terdapat pula tombol "Post" untuk mengirim blog ke database dan menampilkannya di halaman publik.



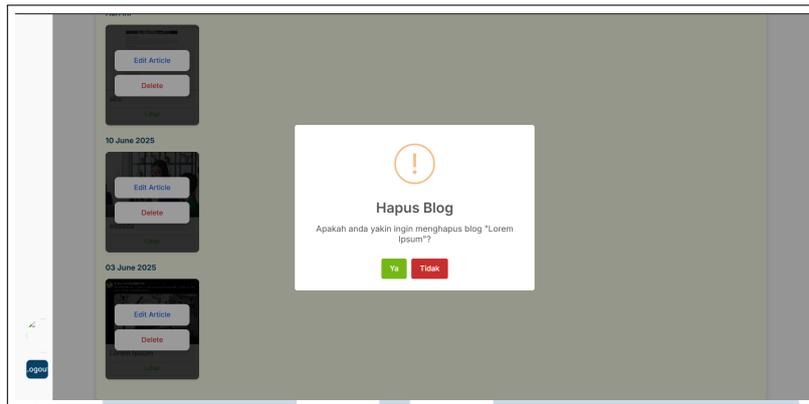
Gambar 3.11. Edit dan Delete

Gambar 3.11 menunjukkan kondisi ketika admin mengaktifkan mode edit. Ketika mode ini diaktifkan, akan muncul overlay pada setiap card blog yang memberikan dua pilihan: Edit Article dan Delete. Tombol-tombol ini memungkinkan admin melakukan tindakan cepat tanpa harus berpindah halaman terlebih dahulu. Fitur ini meningkatkan efisiensi pengelolaan blog dari sisi administrator.



Gambar 3.12. Edit Page

Gambar 3.12 menunjukkan tampilan halaman edit blog. Formulirnya hampir serupa dengan halaman Create Blog, namun dengan data yang telah terisi. Admin dapat mengubah judul, cover image, dan konten blog sesuai kebutuhan. Tombol “Update” disediakan untuk menyimpan perubahan dan menggantikan konten lama dengan yang baru di tampilan publik.

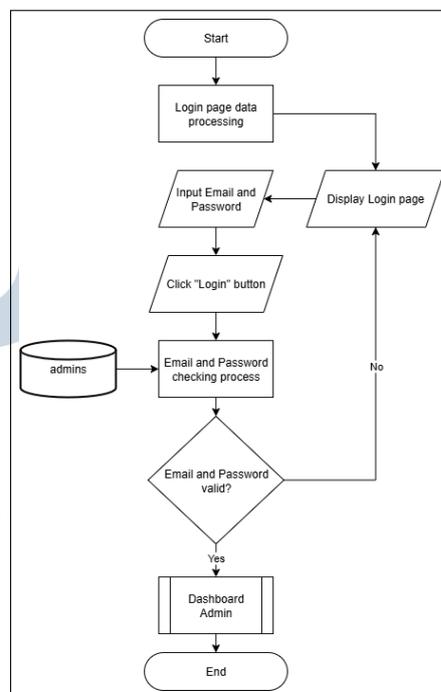


Gambar 3.13. Delete

Tampilan pada gambar 3.13 memperlihatkan konfirmasi saat admin mengklik tombol Delete. Dialog konfirmasi ditampilkan dengan pertanyaan “Apakah anda yakin ingin menghapus blog ini?” serta dua pilihan: ”Ya” untuk melanjutkan penghapusan dan ”Tidak” untuk membatalkannya. Fitur ini dilengkapi dengan notifikasi visual menggunakan SweetAlert2 agar proses penghapusan tidak dilakukan secara tidak sengaja.

3.3.4 Blog - Implementasi Backend

1. Flowchart Modul Login



Gambar 3.14. Flowchart Modul Login

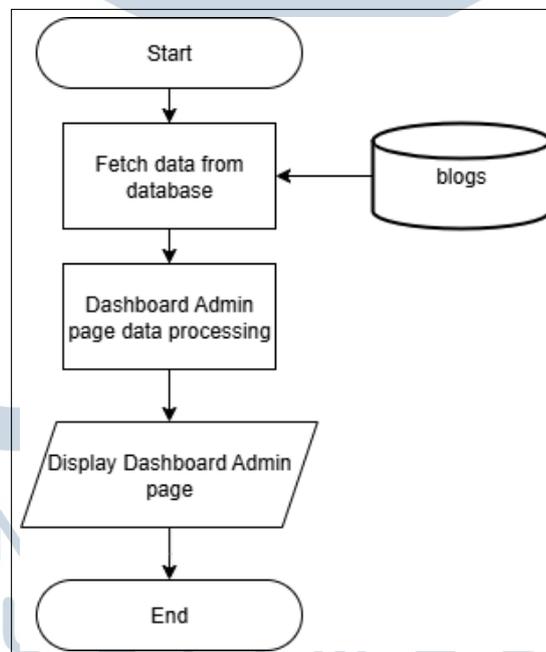
Flowchart pada Gambar 3.14 menunjukkan alur proses autentikasi yang harus dilalui oleh admin sebelum dapat mengakses dashboard admin. Proses dimulai ketika halaman login diproses dan ditampilkan oleh sistem.

Pengguna kemudian diminta untuk mengisi kolom email dan kata sandi pada form login. Setelah input data dilakukan, pengguna menekan tombol *Login* untuk memulai proses autentikasi. Sistem kemudian akan melakukan pengecekan terhadap data yang dimasukkan dengan mencocokkannya ke dalam basis data admins.

Jika kombinasi email dan kata sandi ditemukan valid dan sesuai dengan data yang tersimpan di dalam basis data, maka pengguna akan diarahkan menuju halaman *Dashboard Admin*. Namun, apabila data tidak sesuai, maka sistem akan menampilkan kembali halaman login beserta notifikasi kesalahan untuk memberikan kesempatan kepada pengguna dalam menginput ulang kredensial yang benar.

Fitur ini dirancang untuk memastikan hanya pengguna yang telah terdaftar sebagai admin yang dapat mengakses bagian pengelolaan fitur blog. Validasi dilakukan secara aman, dan prosesnya telah mendukung pengembalian ke halaman awal apabila input tidak sesuai, guna menghindari akses tidak sah dan menjaga integritas sistem.

2. Flowchart Modul Dashboard Admin



Gambar 3.15. Flowchart Modul Dashboard Admin

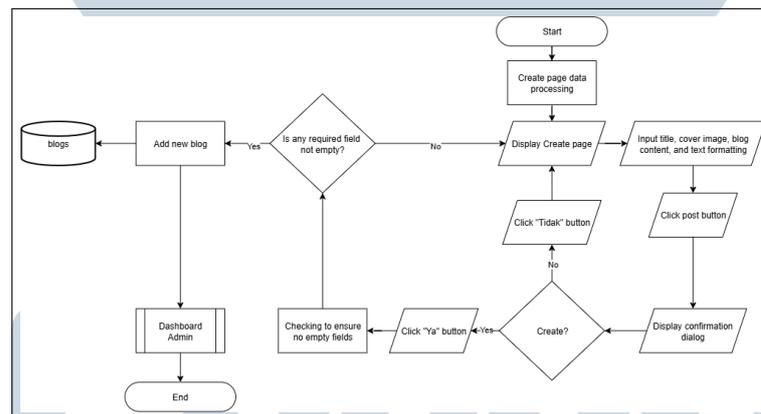
Flowchart pada Gambar 3.15 menggambarkan proses alur kerja sistem ketika seorang admin mengakses halaman utama dashboard setelah berhasil login. Proses dimulai dengan pengolahan data yang diperlukan untuk menampilkan halaman dashboard admin.

Sistem akan mengambil seluruh data yang berkaitan dengan entri blog dari basis data *blogs*. Data ini mencakup informasi seperti judul blog, tanggal pembuatan, status unggulan, jumlah tayangan, jumlah dibagikan, dan lain-lain. Setelah proses pengambilan dan pengolahan data selesai, sistem akan menyusun data tersebut ke dalam tampilan yang dapat dibaca dan dikelola oleh admin.

Halaman dashboard kemudian ditampilkan dalam antarmuka yang interaktif, di mana admin dapat melihat daftar artikel blog yang telah dibuat, memilih untuk mengedit, menghapus, atau membuat artikel baru, serta melihat statistik konten secara langsung. Tampilan ini dirancang responsif agar dapat diakses dari berbagai perangkat dan memudahkan admin dalam melakukan manajemen konten.

Alur ini menjadi pusat kendali utama bagi administrator dalam mengelola seluruh aktivitas blog, karena dari halaman inilah seluruh fitur CRUD (Create, Read, Update, Delete) dapat diakses dengan mudah dan cepat.

3. Flowchart Modul *Create New Blog*



Gambar 3.16. Flowchart Modul Create New Blog

Flowchart pada Gambar 3.16 menggambarkan alur proses pembuatan entri blog baru oleh admin melalui dashboard. Proses dimulai dengan pemrosesan data untuk menampilkan halaman Create, di mana admin disediakan formulir untuk mengisi berbagai elemen blog, seperti judul, gambar sampul (cover image), konten utama blog, serta melakukan pemformatan teks apabila diperlukan.

Setelah seluruh isian diisi, admin menekan tombol Post untuk memproses permintaan pembuatan blog. Sebelum blog benar-benar ditambahkan ke dalam sistem, akan muncul dialog konfirmasi sebagai langkah verifikasi tambahan untuk menghindari entri yang tidak disengaja.

Jika admin menyetujui dengan mengklik tombol Ya, maka sistem akan melakukan pengecekan terhadap seluruh isian formulir untuk memastikan tidak ada bidang wajib yang kosong. Proses validasi ini sangat penting

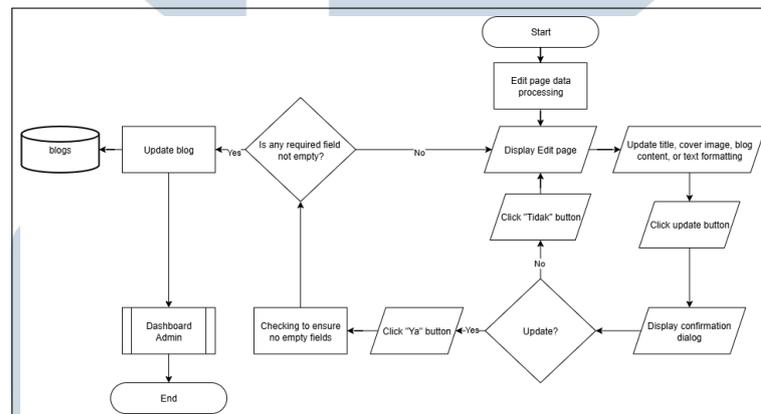
untuk menjaga integritas data dan memastikan bahwa seluruh entri yang masuk telah memenuhi standar minimum kelengkapan.

Apabila seluruh data telah valid, sistem akan menambahkan entri baru ke dalam basis data blogs dan menghubungkannya dengan akun admin yang membuatnya. Relasi antara entri blog dan akun admin dilakukan dengan pendekatan satu-ke-banyak (one-to-many), artinya satu admin dapat membuat banyak blog, namun setiap entri blog hanya dapat dimiliki oleh satu admin.

Setelah proses penambahan blog berhasil, pengguna akan diarahkan kembali ke halaman Dashboard Admin sebagai konfirmasi bahwa entri telah berhasil dipublikasikan. Sebaliknya, apabila pada tahap konfirmasi admin menekan tombol Tidak, maka proses dibatalkan dan pengguna tetap berada di halaman Create untuk dapat melakukan revisi terlebih dahulu sebelum memposting ulang.

Penerapan alur ini bertujuan untuk menjaga kualitas entri blog serta memberikan kontrol penuh kepada admin selama proses pembuatan blog berlangsung, baik dari segi konten, format, maupun kelengkapan data.

4. Flowchart Modul Edit Blog



Gambar 3.17. Flowchart Modul Edit Blog

Flowchart pada Gambar 3.17 menjelaskan alur proses pengeditan entri blog yang dilakukan oleh admin dari dalam sistem manajemen blog. Proses ini memungkinkan admin untuk memperbarui informasi pada blog yang telah dipublikasikan sebelumnya.

Alur dimulai dari tahap Edit page data processing yang bertanggung jawab untuk menyiapkan data blog yang akan diedit dan menampilkannya dalam halaman edit. Setelah halaman ditampilkan, admin diberikan akses untuk memperbarui informasi seperti judul, gambar sampul (cover image), konten utama blog, serta melakukan pemformatan teks (misalnya italic, bold, dan pengaturan paragraf).

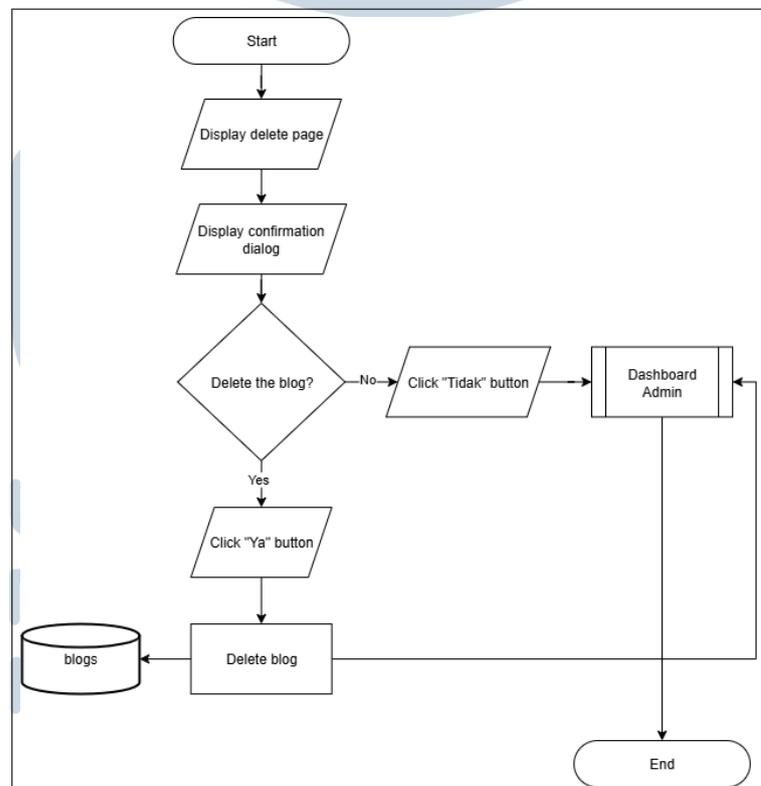
Setelah perubahan dilakukan, admin dapat menekan tombol Update yang akan memunculkan dialog konfirmasi. Jika admin menekan tombol Ya dalam dialog konfirmasi tersebut, maka sistem akan melakukan pengecekan terhadap seluruh bidang isian untuk memastikan bahwa tidak ada data penting yang dibiarkan kosong.

Langkah validasi ini penting untuk memastikan bahwa entri blog yang diperbarui tetap memenuhi standar kualitas dan kelengkapan yang telah ditetapkan sistem. Apabila semua data telah valid, sistem akan melakukan pembaruan terhadap entri blog dalam basis data blogs.

Sebaliknya, apabila pada tahap konfirmasi admin memilih opsi Tidak, maka perubahan tidak akan disimpan dan pengguna akan tetap berada di halaman Edit. Setelah proses pembaruan berhasil, admin akan diarahkan kembali ke halaman Dashboard Admin sebagai penanda bahwa proses edit telah selesai dan blog telah diperbarui.

Relasi antara entri blog dan admin tetap dipertahankan, sehingga seluruh perubahan tetap dapat ditelusuri berdasarkan identitas admin yang melakukan revisi. Proses ini juga menjaga konsistensi konten dan memberikan fleksibilitas kepada admin dalam melakukan perbaikan, pembaruan informasi, atau penyesuaian konten sesuai kebutuhan.

5. Flowchart Modul *Delete* Blog



Gambar 3.18. Flowchart Modul Delete Blog

Flowchart pada Gambar 3.18 menggambarkan alur proses penghapusan entri blog oleh admin dari dalam sistem. Proses ini dimulai dari Delete page data processing yang berfungsi untuk menyiapkan halaman konfirmasi penghapusan berdasarkan data entri blog yang dipilih oleh admin.

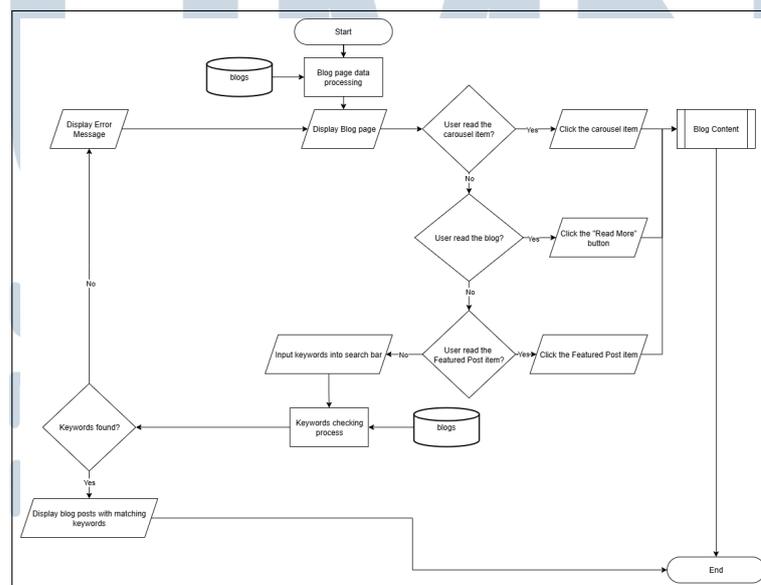
Setelah halaman delete ditampilkan, sistem akan memunculkan confirmation dialog untuk memastikan bahwa tindakan penghapusan benar-benar diinginkan. Hal ini merupakan bagian dari mekanisme pengamanan untuk menghindari penghapusan yang tidak disengaja, mengingat bahwa penghapusan bersifat permanen dan akan menghapus data dari basis data blogs.

Pada dialog konfirmasi tersebut, admin diberikan dua pilihan: memilih tombol Ya untuk melanjutkan proses penghapusan atau memilih Tidak untuk membatalkan tindakan. Jika admin memilih Tidak, maka sistem akan langsung mengarahkan kembali ke halaman Dashboard Admin tanpa melakukan perubahan apa pun terhadap data blog.

Namun, jika admin memilih Ya, sistem akan melanjutkan proses ke tahap berikutnya, yaitu Delete blog. Pada tahap ini, sistem akan menghapus entri blog yang bersangkutan dari tabel penyimpanan utama dan memastikan bahwa entri tersebut tidak lagi dapat diakses oleh pengguna publik.

Setelah proses penghapusan selesai, sistem akan menavigasi kembali ke halaman Dashboard Admin sebagai tanda bahwa proses telah berakhir. Keberadaan proses konfirmasi ini menegaskan pentingnya kontrol administratif dan akurasi dalam pengelolaan konten, serta menghindari risiko kehilangan data akibat kesalahan operasional.

6. Flowchart Modul Blog



Gambar 3.19. Flowchart Modul Blog

Flowchart pada Gambar 3.19 menjelaskan alur interaksi pengguna publik ketika mengakses halaman blog. Proses dimulai dari tahap pemrosesan data blog (blog page data processing), yang bertugas memuat seluruh entri blog yang tersedia dari basis data untuk ditampilkan ke pengguna dalam tampilan daftar blog publik.

Setelah proses pemrosesan data selesai, sistem akan menampilkan halaman blog utama (Display Blog page) yang berisi berbagai entri blog. Di halaman ini, pengguna diberikan beberapa opsi untuk mengeksplorasi konten, termasuk melalui fitur carousel, daftar blog utama, dan artikel unggulan (featured post).

Pertama, sistem memeriksa apakah pengguna memilih untuk membaca artikel dari carousel. Jika ya, maka pengguna akan diarahkan langsung ke halaman konten blog yang dimaksud. Apabila tidak, sistem melanjutkan pengecekan apakah pengguna memilih salah satu entri blog dari daftar umum. Jika pengguna mengklik tombol Read More, maka sistem akan menampilkan halaman konten dari entri blog tersebut.

Jika pengguna tidak memilih dari daftar umum, sistem kembali memeriksa apakah pengguna tertarik pada artikel unggulan yang ditampilkan secara khusus. Klik pada item unggulan ini juga akan mengarahkan ke halaman konten blog.

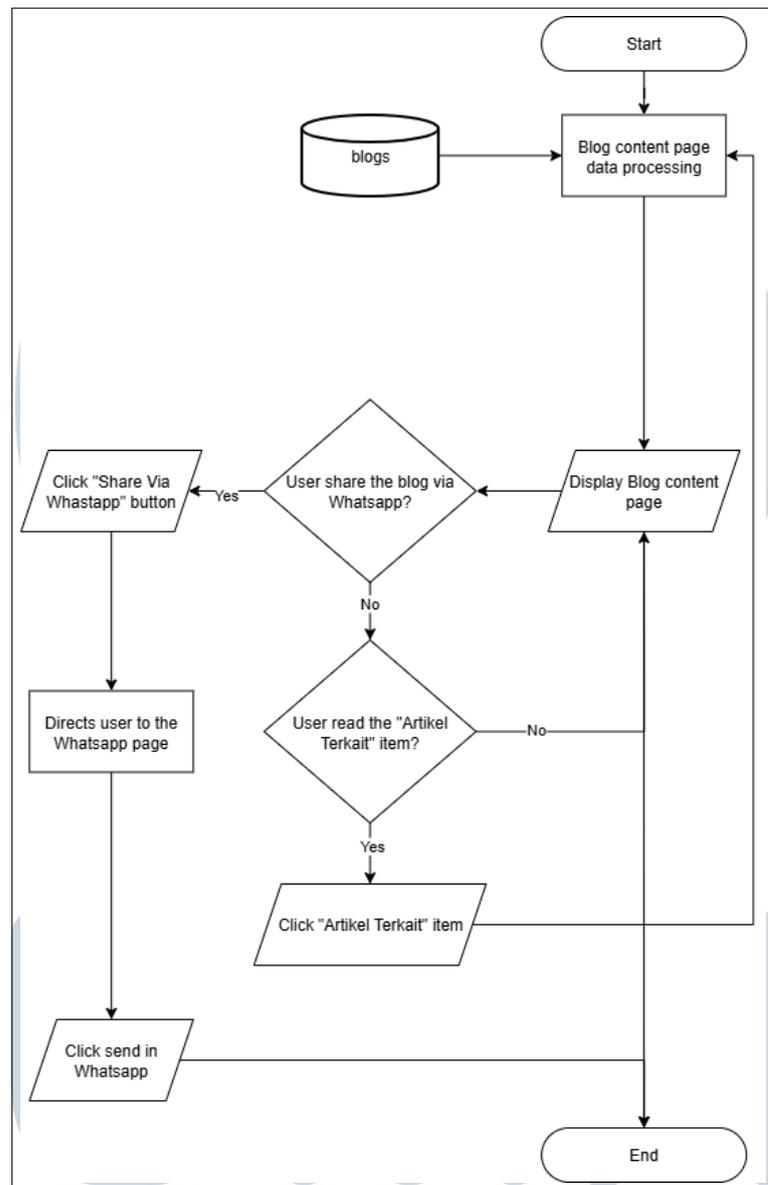
Apabila pengguna tidak melakukan interaksi terhadap ketiga jalur eksplorasi tersebut (carousel, daftar umum, atau unggulan), maka sistem mengarahkan pengguna untuk menggunakan kolom pencarian. Pengguna dapat memasukkan kata kunci yang relevan ke dalam search bar. Sistem kemudian akan memproses pencarian tersebut melalui tahapan pemeriksaan kata kunci (keywords checking process) dengan mencocokkan input pengguna terhadap data blog yang ada di dalam basis data.

Jika hasil pencarian menghasilkan kecocokan, maka sistem akan menampilkan daftar entri blog yang sesuai dengan kata kunci yang dimasukkan. Sebaliknya, jika tidak ditemukan kecocokan, sistem akan menampilkan pesan kesalahan atau informasi bahwa tidak ada hasil yang ditemukan.

Seluruh proses ini didesain agar pengguna memiliki banyak jalur eksplorasi yang fleksibel dan intuitif terhadap konten blog. Fitur carousel, artikel unggulan, dan daftar utama memfasilitasi pengguna untuk menjelajahi konten yang direkomendasikan, sedangkan fitur pencarian memungkinkan pengalaman yang lebih personal melalui masukan kata kunci spesifik.

Pendekatan ini tidak hanya meningkatkan kenyamanan navigasi pengguna, tetapi juga berkontribusi pada user engagement yang lebih tinggi karena pengguna dapat menemukan konten yang relevan dengan minat mereka melalui berbagai cara. Flowchart ini mencerminkan strategi penyajian konten yang mendukung aksesibilitas, relevansi, dan efisiensi dalam sistem blog yang dibangun.

7. Flowchart Modul Blog *Content*



Gambar 3.20. Flowchart Modul Blog Content

Flowchart pada Gambar 3.20 menggambarkan alur sistem pada halaman konten blog publik, yang difokuskan pada bagaimana pengguna dapat mengakses detail artikel melalui berbagai jalur interaksi yang tersedia. Proses dimulai dari tahap pemrosesan data blog (blog page data processing), di mana sistem mengambil data dari basis data blog untuk ditampilkan pada halaman utama.

Setelah data berhasil dimuat, sistem menampilkan halaman blog (Display Blog page), yang menyajikan konten dalam berbagai format seperti carousel artikel pilihan, daftar artikel terbaru, dan entri unggulan (featured post).

Dari sini, pengguna memiliki beberapa opsi eksplorasi yang digambarkan sebagai jalur keputusan dalam flowchart.

Pertama, sistem akan memeriksa apakah pengguna membaca konten dari komponen carousel. Jika pengguna mengklik salah satu item carousel, maka sistem akan menampilkan halaman Blog Content yang berisi detail artikel tersebut. Jika tidak, maka sistem akan memeriksa apakah pengguna membaca entri dari daftar utama blog dengan menekan tombol Read More. Aksi ini juga akan membawa pengguna ke halaman konten blog.

Jika pengguna tidak memilih dari carousel maupun daftar utama, sistem akan memverifikasi apakah pengguna mengakses artikel dari entri unggulan. Klik pada bagian Featured Post akan membawa pengguna ke detail artikel.

Apabila ketiga opsi eksplorasi konten tersebut tidak dipilih, maka sistem akan menyediakan jalur alternatif berupa kolom pencarian. Pengguna dapat memasukkan kata kunci tertentu ke dalam search bar untuk menemukan konten yang relevan. Sistem akan menjalankan proses pemeriksaan kata kunci (Keywords checking process) dengan mencocokkan input terhadap konten yang ada di dalam basis data blog.

Jika kata kunci yang dimasukkan tidak ditemukan dalam sistem, maka sistem akan menampilkan pesan kesalahan (Display Error Message) yang menunjukkan bahwa pencarian tidak menghasilkan hasil yang relevan. Sebaliknya, jika kata kunci ditemukan, maka sistem akan menampilkan daftar artikel yang sesuai dengan kata kunci tersebut (Display blog posts with matching keywords).

Seluruh jalur dalam flowchart ini dirancang untuk mendukung fleksibilitas dan efisiensi dalam pencarian serta pembacaan konten oleh pengguna. Fitur carousel dan entri unggulan berfungsi sebagai konten prioritas atau rekomendasi, sedangkan kolom pencarian memberikan kontrol penuh kepada pengguna untuk menjelajahi konten secara spesifik. Desain ini mencerminkan penerapan prinsip user-centered design dalam pengembangan antarmuka blog publik.

Dengan sistem ini, pengalaman membaca blog menjadi lebih dinamis, responsif, dan ramah pengguna, yang pada akhirnya berkontribusi pada peningkatan tingkat keterlibatan (engagement) pembaca terhadap konten blog yang disediakan.

3.4 Kendala dan Solusi yang Ditemukan

Selama melakukan kegiatan magang di Lembaga Konseling Keluarga Kreatif, terdapat beberapa kendala dan dari kendala tersebut, ditemukan juga beberapa solusi untuk menyelesaikannya.

3.4.1 Kendala

1. Terdapat kendala dalam proses komunikasi dengan supervisor, yang berdampak pada kurang maksimalnya penyampaian dan tindak lanjut terhadap progres pekerjaan.
2. Adanya adaptasi yang diperlukan dikarenakan perubahan prioritas proyek.

3.4.2 Solusi

1. Mengupayakan komunikasi yang lebih proaktif dengan supervisor, seperti menyusun laporan perkembangan secara teratur dan menyampaikan pertanyaan atau hambatan teknis secara jelas, agar supervisi dan tindak lanjut terhadap progres pekerjaan dapat berjalan lebih efektif.
2. Menyesuaikan strategi kerja dengan melakukan manajemen waktu dan prioritas secara sistematis, agar setiap perubahan arah proyek dapat direspons secara fleksibel tanpa mengganggu kelangsungan tugas utama yang sedang berjalan.

