

BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Posisi yang dijabat selama pelaksanaan magang di Vision+ Academy yaitu sebagai *Software Developer Intern* khususnya divisi *frontend* yang berada langsung dibawah *Frontend Department*. Pada pelaksanaan magang ini untuk setiap proyek dikerjakan oleh 2 orang penanggung jawab yang dibagi menjadi 1 orang divisi *frontend* dan 1 orang di divisi *backend*. Adapun untuk penanggung jawab untuk setiap proyek tergantung pada tingkat kepentingan dan skala proyek yang ada. Koordinasi proyek dilakukan melalui berbagai cara seperti *chat Whatsapp*, *Gitlab*, *Jira Software* dan *update progress daily*. Koordinasi dilakukan untuk memantau perkembangan kemajuan proyek dan juga untuk tempat diskusi jika ada kendala dalam tahap pengerjaan.

3.2 Tugas yang Dilakukan

Tugas yang didapat selama pelaksanaan program kerja magang ini yaitu membangun sebuah halaman *Frequently Ask Question* untuk digunakan pada situs web *visionplus.id*. Proyek ini bertujuan untuk mengembangkan halaman FAQ yang bersifat dinamis dan mendukung fitur multibahasa, sehingga memudahkan pengguna dalam mengakses informasi sesuai kebutuhan, sekaligus memungkinkan pihak admin atau pengguna internal untuk melakukan analisis terhadap performa konten yang disajikan. Melalui sistem ini, pengguna dapat lebih mudah menemukan solusi atas permasalahan umum tanpa harus menghubungi layanan pelanggan secara langsung, sementara tim internal dapat meninjau efektivitas artikel melalui fitur analitik yang tersedia.

Proyek ini mencakup pengembangan beberapa halaman penting, seperti *homepage* (halaman utama), *dashboard admin* untuk pengelolaan konten, serta form laporan untuk meninjau atau melaporkan informasi jika ada permasalahan yang belum tersedia pada konten FAQ. Berikut ini merupakan fitur-fitur utama yang tersedia yaitu:

1. *Article Management*: Membuat, mengedit, menghapus, mengarsipkan, dan mengkategorikan artikel melalui dashboard.

2. *Multi-language Support*: Mengaitkan variasi bahasa untuk setiap artikel; menetapkan *fallback* (bahasa default) jika terjemahan tidak tersedia.
3. *Engagement Metrics*: Mencatat dan menampilkan metrik seperti *view count*, *like and dislike count* pada setiap artikel.
4. *Admin Dashboard*: Antarmuka untuk admin dalam memantau metrik, memfilter berdasarkan kategori/bahasa, dan mengidentifikasi artikel yang perlu diperbarui.
5. *User-Facing UI*: Tampilan responsif daftar/detail FAQ, pemilih bahasa, serta tombol suka.

3.3 Uraian Pelaksanaan Magang

Dalam pengembangan halaman FAQ, digunakan berbagai teknologi pendukung, khususnya pada sisi *frontend*. Salah satunya adalah penerapan Nuxt yang dikombinasikan dengan TypeScript, yang memungkinkan pembuatan antarmuka pengguna yang responsif, dinamis, dan lebih terstruktur. Untuk memastikan tampilan yang konsisten dan fleksibel di berbagai perangkat, digunakan juga Tailwind CSS, sebuah *framework* CSS modern yang mempercepat proses desain antarmuka dan memudahkan pengembangan tampilan yang adaptif dan modular. Pelaksanaan kerja magang diuraikan seperti pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu

Minggu Ke -	Pekerjaan yang dilakukan
1	Mempelajari standar penulisan kode web yang mencakup prinsip dasar, penggunaan <i>framework</i> , konsep komponen, serta struktur file dan aplikasi, sebagai dasar dalam pengembangan antarmuka.
2	Kegiatan yang dilakukan berfokus pada pembelajaran <i>Git Workflow</i> , <i>Web Deployment Workflow</i> , dan <i>Development SOP</i> . Materi mencakup penggunaan <i>branch</i> , <i>commit</i> , dan <i>pull request</i> , proses build dan pengelolaan <i>environment variable</i> untuk <i>deployment</i> , serta pemahaman SOP pengembangan seperti alur kerja tim dan pembagian tugas.
Lanjut pada halaman berikutnya	

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu (Lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
3	<p>Kegiatan yang dilakukan mencakup pembelajaran integrasi dengan <i>Backend API</i>, termasuk <i>request-response HTTP</i>, pengelolaan data, dan penanganan <i>error</i>. Selain itu, dipelajari alur percakapan pada sistem <i>CCare</i> dan <i>Chatbot Vision+</i>, seperti logika interaksi, pemetaan intent, dan pengalihan ke agen. Juga dilakukan perancangan halaman FAQ multibahasa yang mencakup pembuatan sitemap untuk navigasi dan <i>flowchart</i> untuk alur interaksi pengguna dan admin.</p>
4	<p>Kegiatan yang dilakukan mencakup pembelajaran integrasi dengan <i>Backend API</i> serta pemahaman alur chat pada sistem <i>CCare</i> dan <i>Chatbot Vision+</i>. Materi integrasi API meliputi cara melakukan <i>request</i> dan <i>response</i> menggunakan metode <i>HTTP</i>, pengelolaan data dari <i>backend</i> ke antarmuka pengguna, serta penanganan <i>error</i>. Sementara itu, pada bagian <i>CCare</i> dan <i>Chatbot</i>, dipelajari alur komunikasi otomatis antara pengguna dan sistem, termasuk logika percakapan, pengalihan ke agen manusia, dan pemetaan intent pengguna. Pembelajaran ini bertujuan untuk membangun sistem yang interaktif, responsif, dan terintegrasi dengan layanan backend secara efisien.</p>
5	<p>Mempelajari <i>Generative AI</i> terkait dengan pembangunan <i>Chatbot</i> yang meliputi pemahaman konsep dasar dan jenis model generatif, pengenalan model populer seperti <i>GPT</i> untuk <i>text generation</i>, teknik pengolahan dataset dan <i>preprocessing input</i>, serta penerapan model dalam menghasilkan respons otomatis. Pembelajaran juga mencakup proses <i>fine-tuning</i> dan penggunaan <i>prompt</i> untuk meningkatkan relevansi jawaban <i>chatbot</i> sesuai konteks percakapan.</p>
<p>Lanjut pada halaman berikutnya</p>	

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu (Lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
6	Kegiatan difokuskan pada task <i>mini project</i> bertema Widget MPI. Pekerjaan diawali dengan proses inialisasi proyek dan perancangan tampilan antarmuka pengguna. Selain itu, dilakukan instalasi <i>framework</i> Nuxt.js sebagai fondasi pengembangan, serta implementasi Pinia untuk kebutuhan <i>state management</i> . Setelah struktur dasar terbentuk, pengerjaan proyek dilanjutkan dengan pengembangan elemen-elemen awal tampilan dan alur dasar aplikasi.
7	Pengembangan proyek Widget MPI difokuskan pada pembuatan tampilan halaman <i>landing page</i> . Setelah itu, tim menerima API dari backend untuk diintegrasikan ke dalam proyek. Kegiatan dilanjutkan dengan proses integrasi API serta pengembangan logika frontend agar data yang diperoleh dari API dapat ditampilkan secara dinamis dan interaktif di antarmuka pengguna.
8	Perkembangan proyek yaitu perbaikan tampilan agar lebih sesuai dengan konten yang tersedia. Penyesuaian ini mencakup styling dan layout agar lebih responsif dan informatif. Selanjutnya, dilakukan finalisasi proyek secara keseluruhan dan dilakukan <i>weekly report</i> untuk mengevaluasi yang sudah dilakukan selama seminggu.
9	Dimulai dengan <i>sprint plan</i> untuk proyek halaman FAQ yang mencakup penentuan <i>goal</i> yang ingin dicapai serta melakukan <i>build up new project</i> . Selanjutnya dilakukan pembuatan <i>repository</i> , perancangan kerangka <i>landing page</i> untuk <i>user</i> , dan <i>import</i> aset dari <i>Figma</i> . Setelah struktur dasar terbentuk, dilakukan pengembangan tampilan <i>landing page</i> serta halaman Ajukan Pengaduan dan Lacak Pengaduan, lengkap dengan navigasi antar halaman. Di akhir minggu, halaman pengaduan disempurnakan dengan penambahan <i>dummy data</i> yang disimpan di <i>store</i> menggunakan Pinia, kemudian ditutup dengan <i>weekly report</i> sebagai bentuk evaluasi pekerjaan selama satu minggu.
Lanjut pada halaman berikutnya	

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu (Lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
10	<p>Pengembangan difokuskan pada tampilan fitur "<i>Find by Topic</i>" dan rekomendasi FAQ menggunakan <i>dummy data</i>. Dilanjutkan dengan penambahan data subtopik serta penyesuaian tipe data artikel agar dapat menyimpan ID subtopik, dan pembuatan halaman untuk menampilkan artikel berdasarkan topik yang dipilih. Tampilan halaman ditingkatkan agar juga menampilkan subtopik dan tampil lebih responsif. Selain itu, ditambahkan fitur tombol untuk mengganti bahasa, perbaikan <i>dropdown</i> pada <i>form</i> pengaduan, serta penyesuaian tampilan daftar topik untuk mode <i>mobile</i>. Di akhir minggu, dilakukan refactor komponen menjadi berbasis <i>atomic design</i> agar lebih reusable, serta merapikan struktur kode. Lalu ditutup dengan pelaksanaan <i>weekly report</i>.</p>
11	<p>Pengembangan diarahkan pada pembuatan <i>base component</i> untuk <i>button</i> agar dapat <i>reusable</i>, sekaligus mengganti struktur <i>button</i> yang ada dengan komponen tersebut. Selain itu, dilakukan perapihan struktur folder agar sesuai dengan konsep <i>atomic design</i>. Kemudian dibuat <i>base component</i> tambahan seperti <i>input</i>, <i>dropdown</i>, dan <i>upload file</i>, serta penyesuaian kode agar seluruh <i>form</i> menggunakan komponen tersebut. Di akhir minggu, dibuat <i>base component</i> untuk fitur <i>search</i>, perapihan tampilan, dan push ke <i>branch</i> sementara. Kegiatan ditutup dengan <i>weekly report</i> sebagai evaluasi mingguan.</p>
12	<p>Penambahan tombol kontak <i>WhatsApp</i> di <i>navbar</i> serta penyesuaian tampilan halaman <i>show article by topicId</i> untuk mode <i>mobile</i>. Selain itu, dibuat <i>base component</i> untuk navigasi utama dan diperbaiki tampilan <i>hamburger navbar</i>, serta halaman "Lacak Pengaduan" dan "Ajukan Pengaduan" agar responsif di perangkat <i>mobile</i>. Pengembangan juga mencakup pembuatan desain sementara untuk halaman admin <i>dashboard</i>. Kegiatan ditutup dengan <i>weekly report</i>, serta perapihan kode dan tampilan pada sisi pengguna.</p>
Lanjut pada halaman berikutnya	

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu (Lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
13	<p>Kegiatan diawali dengan <i>sprint plan</i> untuk menentukan target yang ingin dicapai dalam periode sprint berikutnya, serta melanjutkan pengembangan <i>landing page</i> untuk halaman admin <i>dashboard</i>. Pengembangan dilanjutkan dengan pembuatan <i>sidebar</i> dan komponen <i>card</i> sebagai tampilan <i>overview</i> konten. Selain itu, dibuat halaman manajemen artikel yang dilengkapi dengan <i>dropdown</i> untuk menambahkan konten seperti topik, subtopik, dan artikel, serta <i>custom search component</i>. Tampilan untuk fitur <i>edit</i> dan <i>delete</i> artikel juga mulai dikembangkan, bersamaan dengan penyempurnaan tampilan <i>overview</i> artikel. Di luar proyek FAQ, dalam kegiatan <i>Event Timnas Day</i> dengan memantau traffic siaran selama pertandingan berlangsung dan melakukan observasi langsung bersama tim <i>Customer Service</i> untuk mengidentifikasi kendala yang dialami pelanggan.</p>
14	<p>Pengembangan masih difokuskan pada halaman admin <i>dashboard</i> dengan penambahan tampilan <i>analytic card</i> untuk konten. Selain itu juga dilakukan evaluasi terkait hasil observasi selama <i>Event Timnas Day</i>, termasuk pemantauan <i>traffic</i> siaran dan kendala yang dihadapi pelanggan. Selama menunggu data dari <i>backend</i>, dilakukan perapihan <i>styling</i>, struktur kode, serta <i>file</i> aset proyek. Di akhir minggu, tampilan halaman <i>user</i> dan admin <i>dashboard</i> kembali diperbaiki agar lebih rapi dan konsisten. Kegiatan ditutup dengan <i>weekly report</i> sebagai evaluasi, sekaligus persiapan integrasi data <i>backend</i> yang sudah tersedia untuk minggu berikutnya.</p>
Lanjut pada halaman berikutnya	

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Tabel 3.1. Pekerjaan yang dilaksanakan setiap minggu (Lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
15	Kegiatan dimulai dengan <i>sprint plan</i> untuk menentukan target yang ingin dicapai dalam satu periode <i>sprint</i> ke depan. Selanjutnya dilakukan pengembangan tampilan halaman manajemen topik dan subtopik. Pengembangan dilanjutkan dengan pembuatan tampilan manajemen konten serta fitur filter, sekaligus integrasi API untuk halaman <i>user</i> . Sistem juga mulai dihubungkan dengan API untuk menampilkan daftar artikel berdasarkan topik, artikel rekomendasi berdasarkan jumlah <i>like</i> terbanyak, serta fitur pencarian. Di akhir minggu, dilakukan integrasi API untuk mendukung fitur translasi artikel multibahasa dan dilakukan <i>weekly report</i> untuk evaluasi <i>progress</i> yang dilakukan.

Sebelum dilakukan pengembangan, halaman FAQ pada *website* Vision+ masih terbatas baik dari segi fitur maupun antarmuka. Seperti yang ditunjukkan pada Tabel 3.2, yaitu belum tersedia fitur pencarian, daftar topik yang lengkap, sistem tanya-jawab, umpan balik artikel, formulir pengaduan, pelacakan status, serta pilihan bahasa yang memadai. Selain itu, belum ada halaman admin untuk pengelolaan konten. Kekurangan ini mendorong pengembangan halaman FAQ yang lebih interaktif dan fungsional, dengan penambahan fitur-fitur utama seperti pencarian dinamis, pengelompokan topik, sistem pengaduan, pemilihan bahasa, dan *dashboard* admin.

Tabel 3.2. List Sebelum dan Target Pengembangan *Website* Halaman FAQ

Sebelum Pengembangan	Target Pengembangan
Tidak terdapat fitur pencarian pertanyaan, topik, atau kendala.	Menambahkan kolom pencarian untuk memudahkan pengguna menemukan informasi yang dibutuhkan.
Topik bantuan hanya mencakup beberapa menu terbatas.	Menampilkan daftar lengkap topik dan subtopik yang terorganisir dan mudah dinavigasi.
Lanjut pada halaman berikutnya	

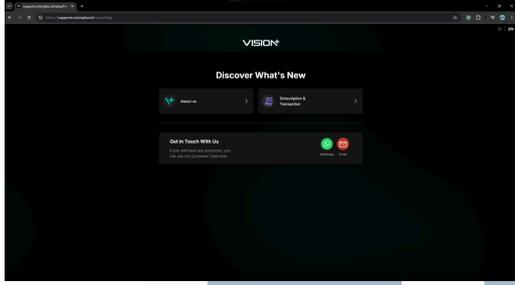
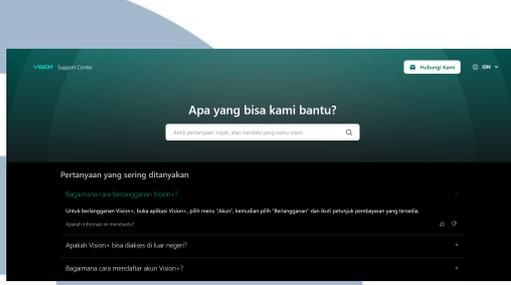
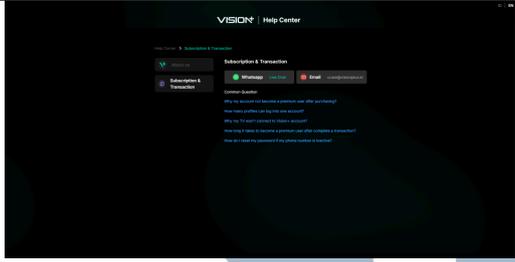
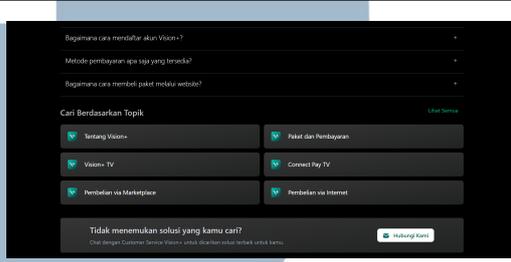
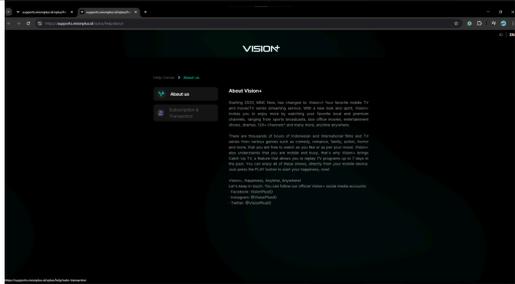
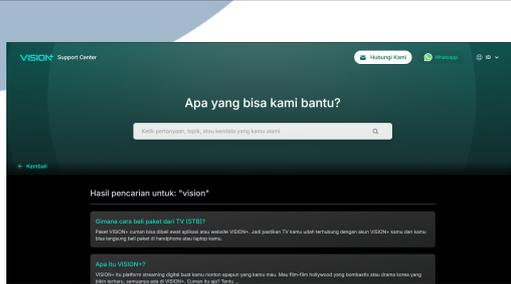
Tabel 3.2. List Sebelum dan Target Pengembangan *Website* Halaman FAQ (Lanjutan)

Sebelum Pengembangan	Target Pengembangan
Tidak ada sistem pertanyaan dan jawaban yang bisa dibuka dan dibaca pengguna.	Menyediakan format tanya-jawab untuk setiap subtopik yang dapat dibuka dan dibaca dengan mudah.
Tidak tersedia fitur pemberian <i>feedback</i> .	Menambahkan tombol <i>feedback</i> pengguna untuk setiap artikel.
Tidak tersedia formulir pengaduan.	Menambahkan fitur pengajuan pengaduan melalui formulir langsung di platform.
Tidak dapat melacak status pengaduan.	Menambahkan fitur pelacakan pengaduan dengan ID Tiket atau Email.
Hanya tersedia pilihan bahasa terbatas.	Menyediakan fitur pemilihan bahasa di bagian antarmuka.
Belum tersedia halaman admin untuk mengelola pembaruan konten	Menyediakan <i>role</i> admin dan halaman <i>dashboard</i> untuk mengelola pembaruan konten

Tabel 3.3 menyajikan beberapa perbandingan tampilan halaman FAQ pada *website* Vision+ sebelum dan sesudah dilakukan pengembangan. Pada bagian (a) merupakan tampilan awal halaman FAQ yang masih sederhana, sedangkan pada bagian (b) menampilkan versi halaman yang lebih informatif dan interaktif, dengan penambahan kolom pencarian serta penyusunan konten yang lebih terstruktur.

Bagian (c) memperlihatkan halaman versi lama yang menampilkan daftar artikel namun belum tertata dengan baik, sementara bagian (d) menunjukkan versi pengembangan yang sudah mengelompokkan artikel berdasarkan subtopik dan lebih mudah dinavigasi. Adapun bagian (e) menampilkan salah satu artikel sebelum pengembangan dengan tampilan yang belum mendukung pengalaman baca yang optimal. Sebaliknya, bagian (f) memperlihatkan hasil pengembangan fungsi pencarian yang memungkinkan pengguna untuk menemukan artikel secara langsung berdasarkan kata kunci. Perbandingan ini menunjukkan peningkatan signifikan baik dari sisi antarmuka maupun fungsionalitas yang mendukung kemudahan penggunaan.

Tabel 3.3. Perbandingan Tampilan Halaman FAQ Sebelum dan Setelah Pengembangan

Sebelum Pengembangan	Setelah Pengembangan
 <p>(a)</p>	 <p>(b)</p>
 <p>(c)</p>	 <p>(d)</p>
 <p>(e)</p>	 <p>(f)</p>

Proses pengembangan halaman FAQ multibahasa pada *website* Vision+ melibatkan beberapa komponen utama seperti *sitemap*, dan *flowchart*. *Sitemap* digunakan untuk menggambarkan struktur navigasi antar halaman seperti *homepage*, daftar FAQ, *detail* FAQ, dan *dashboard admin*. Sedangkan *Flowchart* digunakan untuk menjelaskan alur interaksi pengguna, mulai dari pencarian pertanyaan, pemilihan bahasa, hingga pengelolaan konten oleh admin.

3.3.1 Perancangan Sistem

Sistem ini dibuat menjadi beberapa halaman seperti *homepage*, *form report*, *tracking* dan *dashboard analytic*. Sistem ini juga membagi hak akses untuk 2 jenis pengguna seperti *User* dan *Admin* untuk mengelola keseluruhan konten. Berikut ini

merupakan *user requirements* yang telah dikumpulkan oleh *project manager*.

1. *User Requirement*

- a. Dapat mencari pertanyaan, topik, atau kendala melalui kolom pencarian.
- b. Dapat melihat daftar topik dan subtopik bantuan yang tersedia.
- c. Dapat membuka dan membaca jawaban dari setiap pertanyaan atau subtopik.
- d. Dapat menilai apakah informasi yang diberikan membantu atau tidak.
- e. Dapat melihat daftar pertanyaan yang sering ditanyakan (FAQ).
- f. Dapat mengajukan pengaduan melalui formulir yang tersedia.
- g. Dapat melakukan pelacakan pengaduan dengan memasukkan ID Tiket atau *Email*.
- h. Terdapat fitur pemilihan bahasa.

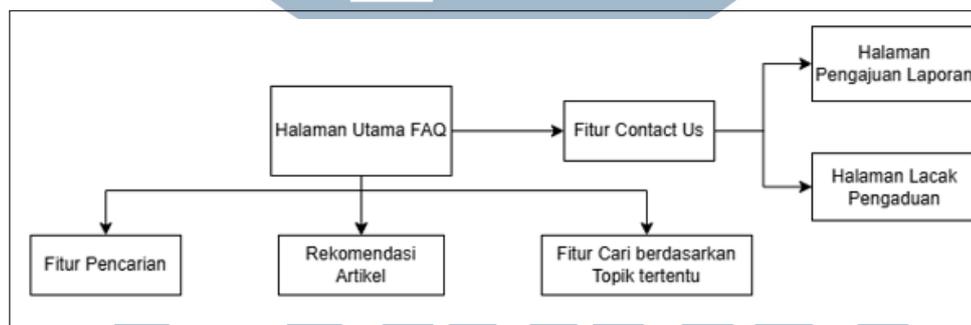
2. *Admin Requirement*

- a. Dapat melihat dan menganalisis data artikel FAQ melalui *dashboard analytics*.
- b. Dapat mengekspor data analitik kedalam format CSV atau Excel
- c. Dapat membuat kategori, folder, dan artikel baru melalui dashboard admin.
- d. Dapat mengedit kategori, folder, dan artikel yang sudah ada.
- e. Dapat menghapus kategori, folder, dan artikel yang tidak lagi relevan (dengan konfirmasi jika terdapat dependensi).
- f. Dapat mengarsipkan kategori, folder, dan artikel agar tidak ditampilkan ke publik namun tetap tersimpan.
- g. Dapat menambahkan varian bahasa (multi-language) untuk kategori, folder, dan artikel tertentu.
- h. Dapat menggunakan fitur drag-and-drop untuk menyusun ulang urutan kategori, folder, dan artikel.
- i. Dapat memfilter artikel berdasarkan kategori, bahasa, status (*draft/publish/arsip*), atau *engagement (view/like/dislike)*.

- j. Dapat memberikan rekomendasi artikel yang sering ditanyakan ke halaman utama FAQ berdasarkan engagement metrics (view count, like count, atau rekomendasi agen/AI).
- k. Dapat melihat daftar artikel yang paling banyak dilihat atau disukai oleh pengguna untuk mengetahui artikel yang perlu diperbarui atau dikembangkan lebih lanjut.

A *Sitemap User*

Setelah pengguna mengakses halaman pusat bantuan Vision+, seperti pada Gambar 3.1, sistem akan menampilkan halaman utama yang berisi kolom pencarian (*search*) di bagian atas. Fitur ini memungkinkan pengguna mencari pertanyaan berdasarkan kata kunci, topik, atau kendala yang dialami secara cepat dan efisien. Di bawahnya, terdapat daftar pertanyaan yang sering ditanyakan (FAQ) serta kategori topik seperti "Tentang Vision+", "Paket dan Pembayaran", atau "VISION+ TV".



Gambar 3.1. *Sitemap User*

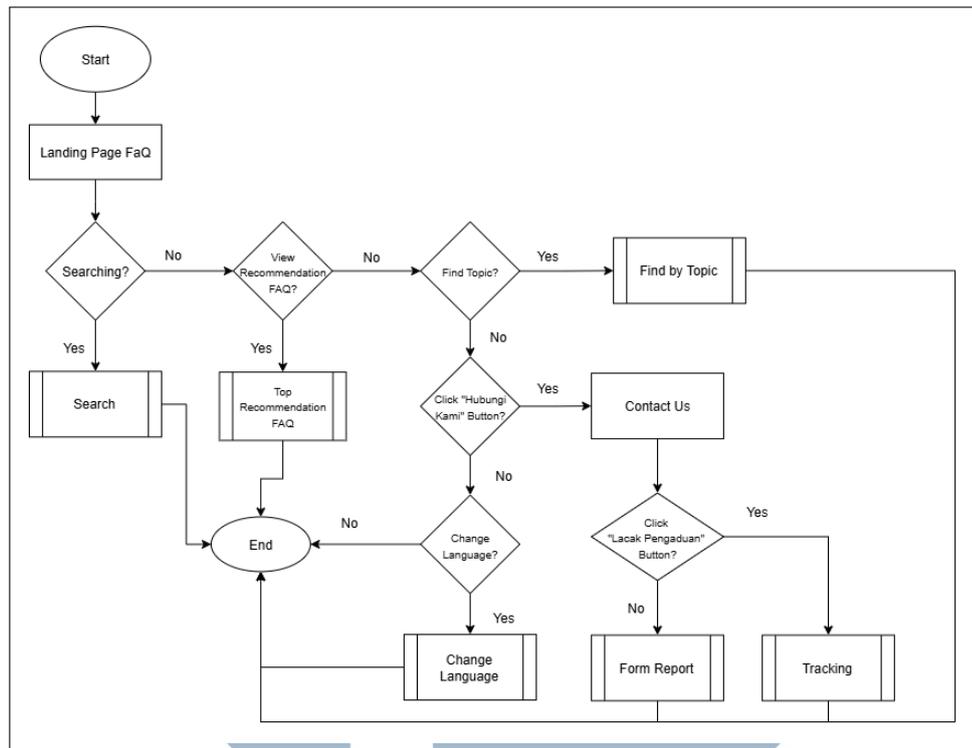
Pengguna dapat memilih topik untuk melihat daftar pertanyaan dan jawaban yang relevan. Sistem juga mendukung fitur multi bahasa, yang memungkinkan pengguna memilih bahasa sesuai preferensi, guna meningkatkan kenyamanan dalam memahami informasi. Jika pengguna tidak menemukan solusi, tersedia tombol "Hubungi Kami" yang mengarahkan ke halaman formulir pengaduan atau pelacakan pengaduan yang telah diajukan, lengkap dengan opsi untuk melampirkan bukti dan memilih kategori kendala yang sesuai.

B *Flowchart Halaman User*

Setelah pengguna mengakses halaman utama FAQ, sistem menampilkan *Homepage* yang berisi beberapa pilihan utama, termasuk *Top Recommendation* FAQ yang menampilkan pertanyaan yang paling sering ditanyakan, serta daftar topik yang tersedia. Jika pengguna memilih untuk menjelajahi topik, sistem akan mengambil *Topic ID* berdasarkan pilihan yang diambil, kemudian menampilkan daftar FAQ yang relevan sesuai dengan *Topic ID* tersebut. Data topik ini diambil dari *database* yang menyimpan seluruh kategori topik yang tersedia.

Selain itu, pengguna juga dapat memanfaatkan fitur *Search* jika ingin mencari informasi secara spesifik. Saat melakukan pencarian, sistem akan memeriksa apakah pengguna sedang mencari. Jika ya, pengguna diarahkan ke halaman pencarian. Jika tidak, sistem menampilkan rekomendasi FAQ teratas. Apabila pengguna tidak menemukan solusi yang sesuai dari daftar FAQ, pengguna dapat memilih tombol *Contact Us*. Opsi ini akan mengarahkan pengguna ke dua pilihan: halaman *Form Report* untuk mengirimkan keluhan atau pertanyaan baru, dan halaman *Tracking* untuk melacak status laporan yang telah dikirim sebelumnya. Dengan alur pada Gambar 3.2 ini, sistem memberikan fleksibilitas bagi pengguna untuk mencari solusi secara mandiri maupun melakukan pelaporan masalah secara langsung.



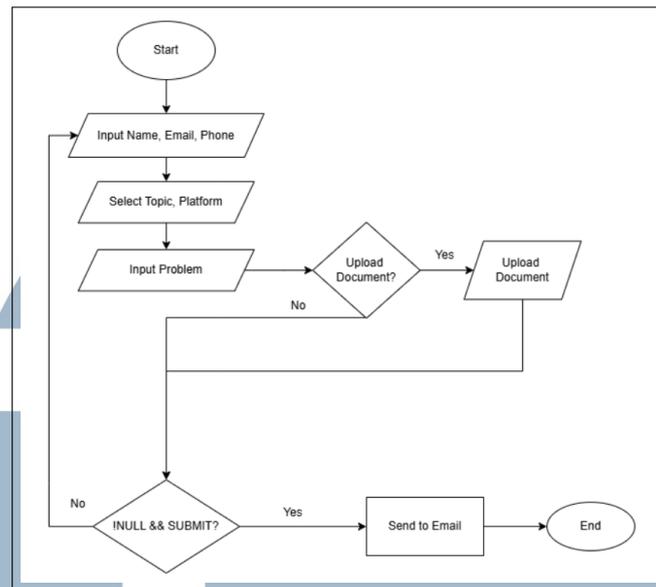


Gambar 3.2. *Flowchart Halaman User*

B.1 *Flowchart Modul Form Report*

Setelah pengguna memilih menu *Contact Us*, sistem akan menampilkan halaman formulir pengaduan yang harus diisi oleh pengguna. Formulir ini mencakup beberapa kolom input, antara lain Nama, Email yang bisa dihubungi, serta Email atau Nomor Telepon yang terdaftar di Vision+. Setelah itu, pengguna diminta untuk memilih kategori kendala serta *platform* yang digunakan, lalu menjelaskan secara rinci permasalahan yang dialami. Untuk memperjelas laporan, pengguna juga memiliki opsi untuk mengunggah dokumen pendukung seperti tangkapan layar atau bukti transfer, dengan format gambar atau PDF dan ukuran maksimal 1MB.

Setelah semua input diverifikasi dan tidak ada data yang kosong, sistem akan mengirimkan laporan tersebut ke email pusat layanan pelanggan untuk ditindaklanjuti lebih lanjut. Dengan alur yang ada pada Gambar 3.3 ini, sistem memastikan bahwa setiap laporan yang masuk bersifat valid, lengkap, dan dapat diproses secara efisien oleh tim terkait.



Gambar 3.3. Flowchart Modul Form Report

B.2 Flowchart Modul Tracking

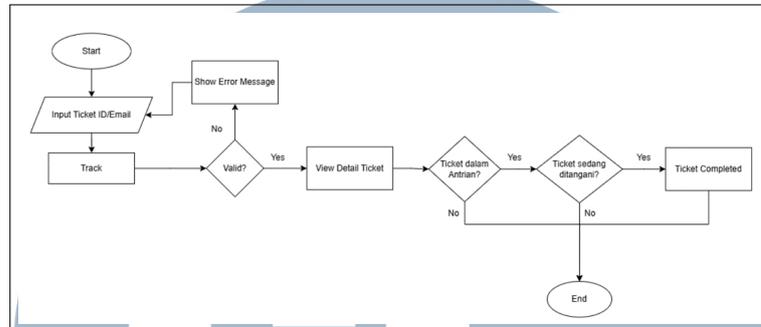
Setelah pengguna membuka halaman pelacakan tiket, sistem akan meminta pengguna untuk mengisi informasi berupa Ticket ID atau alamat email yang digunakan saat pengajuan tiket. Setelah input dimasukkan, pengguna menekan tombol Track untuk memulai proses pelacakan. Sistem kemudian akan melakukan validasi terhadap data yang dimasukkan. Jika data tidak valid, sistem akan menampilkan *error message* dan pengguna diminta untuk memeriksa kembali input yang dimasukkan.

Namun, jika data yang dimasukkan valid, sistem akan menampilkan halaman Detail Ticket yang memuat status tiket pengguna. Pada tahap ini, sistem akan memeriksa status tiket tersebut dan menampilkannya dalam salah satu dari tiga kemungkinan status, yaitu:

- Tiket dalam Antrian, jika laporan masih menunggu untuk diproses
- Tiket sedang ditangani, jika laporan sudah dalam proses penanganan oleh tim terkait
- Tiket *Completed*, jika laporan telah diselesaikan dan ditutup.

Dengan alur yang ada pada Gambar 3.4 ini, pengguna dapat dengan mudah mengetahui posisi atau perkembangan terakhir dari laporan yang telah diajukan, tanpa harus menghubungi layanan pelanggan secara langsung. Sistem pelacakan

ini dirancang untuk memberikan transparansi dan efisiensi dalam penanganan tiket.



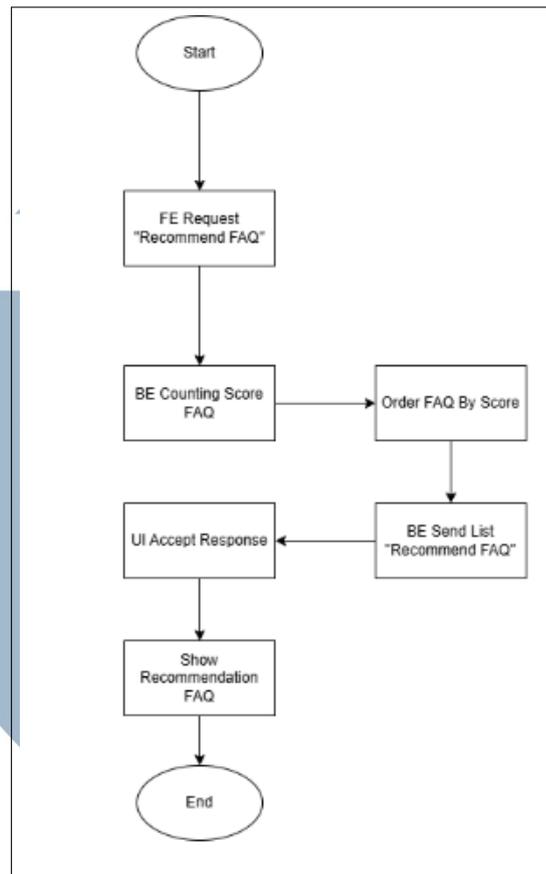
Gambar 3.4. *Flowchart Modul Tracking*

B.3 *Flowchart Modul Top Recommendation FAQ*

Setelah pengguna membuka halaman *Support Center*, sistem secara otomatis menampilkan daftar pertanyaan yang sering diajukan (*Frequently Asked Questions* atau FAQ) tanpa perlu melalui menu tambahan. Daftar ini langsung disajikan dalam bentuk *accordion*, di mana masing-masing pertanyaan dapat diklik untuk melihat jawaban atau detail informasinya. Ketika pengguna mengklik salah satu pertanyaan, sistem akan menampilkan isi jawaban di bawah pertanyaan tersebut, lengkap dengan informasi tambahan seperti tautan ke artikel yang relevan atau tombol untuk menandai apakah informasi tersebut membantu.

Seluruh proses ini dilakukan secara otomatis dan responsif seperti pada Gambar 3.5, memungkinkan pengguna untuk langsung menemukan jawaban dari permasalahan umum yang sering terjadi tanpa harus menghubungi layanan pelanggan. Dengan penyajian yang sederhana dan informatif ini, sistem memudahkan pengguna untuk menavigasi informasi secara cepat dan efisien.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

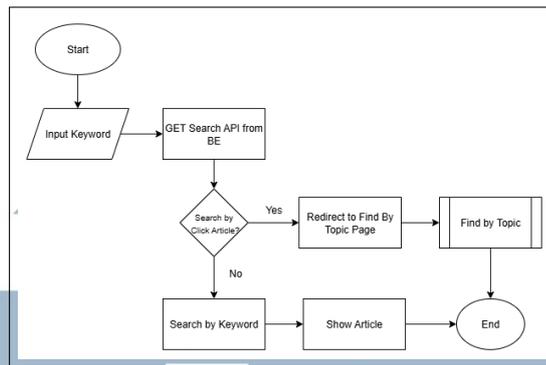


Gambar 3.5. Flowchart Modul Top Recommendation FAQ

B.4 Flowchart Modul Search

Flowchart pada Gambar 3.6 ini menggambarkan alur logika sistem pencarian artikel dengan dua metode berbeda, yaitu pencarian berdasarkan klik artikel dan pencarian berdasarkan kata kunci. Proses diawali saat pengguna memasukkan kata kunci, lalu sistem mengambil data dari *Search API backend*. Setelah data diterima, sistem memeriksa apakah pencarian berasal dari klik artikel.

Jika pencarian dilakukan melalui klik artikel, pengguna akan diarahkan ke halaman *"Find By Topic"* untuk menampilkan artikel sesuai topik yang dipilih. Jika tidak, sistem langsung menampilkan hasil pencarian berdasarkan kata kunci yang dimasukkan. Kedua jalur pencarian tersebut berakhir pada tahap *"Show Article"*, di mana artikel ditampilkan kepada pengguna. Alur ini memberikan fleksibilitas bagi pengguna untuk menemukan artikel baik melalui navigasi topik maupun pencarian langsung.

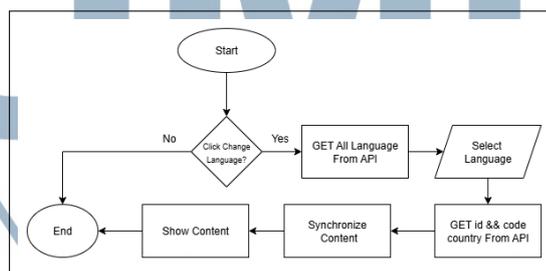


Gambar 3.6. *Flowchart Modul Search*

B.5 *Flowchart Modul Change Language*

Flowchart pada Gambar 3.7 ini menggambarkan alur logika sistem manajemen multi-bahasa pada konten aplikasi. Proses diawali dengan pengecekan apakah pengguna mengklik opsi untuk mengubah bahasa. Jika tidak, sistem langsung menampilkan konten dalam bahasa *default* dan proses berakhir. Namun, jika pengguna ingin mengubah bahasa, sistem akan mengambil seluruh data bahasa dari API untuk kemudian ditampilkan dalam bentuk pilihan kepada pengguna.

Setelah pengguna memilih bahasa, sistem mengambil ID dan kode negara dari API, lalu melakukan sinkronisasi konten sesuai bahasa yang dipilih. Terakhir, sistem menampilkan konten dalam bahasa yang baru. Alur ini memungkinkan dukungan multi-bahasa secara dinamis, sehingga konten aplikasi dapat disesuaikan dengan preferensi bahasa pengguna.



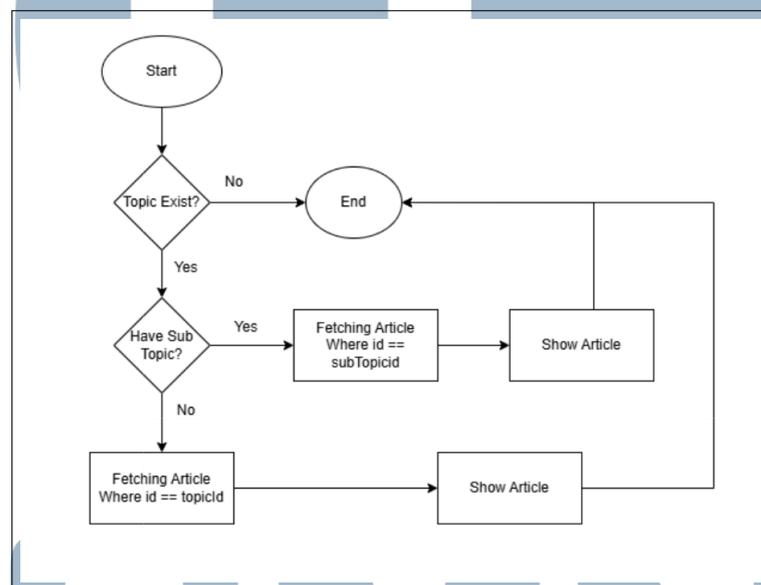
Gambar 3.7. *Flowchart Modul Change Language*

B.6 *Flowchart Modul Find by Topic*

Flowchart ini menggambarkan alur logika dalam menampilkan artikel berdasarkan topik dan subtopik yang dipilih oleh pengguna. Proses dimulai

dengan pemeriksaan apakah topik yang dimaksud tersedia. Jika tidak ada, maka proses dihentikan. Jika topik tersedia, sistem akan memeriksa apakah topik tersebut memiliki subtopik. Jika terdapat subtopik, sistem akan mengambil artikel berdasarkan 'subTopicId'; jika tidak, maka artikel diambil berdasarkan 'topicId'.

Setelah proses pengambilan data berdasarkan ID yang sesuai, artikel akan ditampilkan kepada pengguna. Alur ini memastikan bahwa sistem dapat menampilkan informasi yang relevan secara dinamis, baik berdasarkan topik utama maupun subtopiknya, sehingga pengalaman pengguna menjadi lebih terstruktur dan informatif.



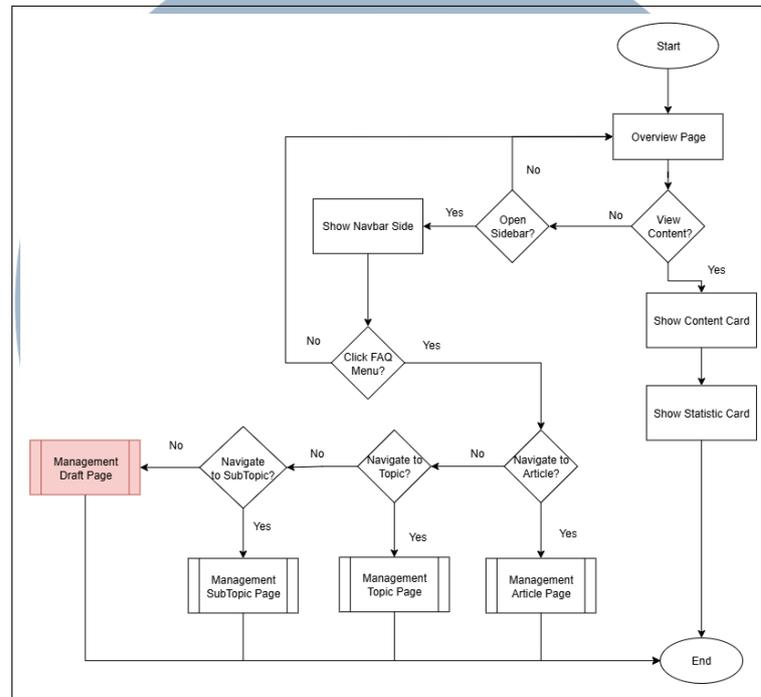
Gambar 3.8. *Flowchart* Modul *Find by Topic*

C *Flowchart* Halaman Admin

C.1 *Flowchart* Halaman Overview

Pada Gambar 3.9, *flowchart* ini menggambarkan alur navigasi pada halaman *Overview* dalam sistem *dashboard* admin. Proses diawali dari halaman *Overview*, di mana pengguna dapat melihat jumlah konten yang dimiliki baik itu Topik, Sub Topik, Artikel dan *Draft*. Selain itu, pengguna juga memiliki opsi untuk membuka Sidebar. Setelah sidebar ditampilkan, pengguna dapat mengakses menu FAQ, yang kemudian menyediakan tiga opsi navigasi: ke halaman *Topic*, *SubTopic*, *Article*, atau *Draft*. Apabila pengguna memilih salah satu dari tiga navigasi tersebut, maka sistem akan mengarahkan ke halaman manajemen yang sesuai, yaitu halaman

Management Topic Page, Management SubTopic Page, Management Article Page atau Management Draft Page. Alur ini memberikan fleksibilitas bagi admin dalam mengelola berbagai jenis konten dari satu titik pusat navigasi.



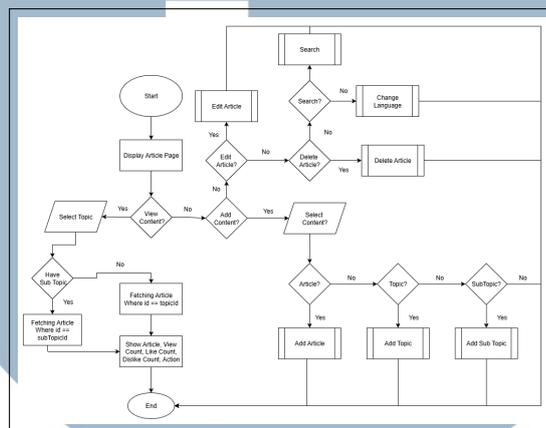
Gambar 3.9. Flowchart Halaman Overview

C.2 Flowchart Management Article Page

Pada Gambar 3.10 flowchart ini menggambarkan alur interaksi pengguna (administrator) dalam mengelola konten pada halaman Manajemen Artikel di *dashboard Support Center*. Alur dimulai dengan tampilan halaman artikel. Dari sini, pengguna memiliki beberapa opsi utama. Pertama, dapat memilih untuk "View Content?" dengan menyeleksi topik. Jika topik memiliki sub-topik, sistem akan mengambil artikel berdasarkan ID sub-topik; jika tidak, sistem akan mengambil artikel berdasarkan ID topik. Setelah itu, artikel akan ditampilkan beserta metrik seperti *View Count, Like Count, Dislike Count*, dan kolom *Action* untuk manajemen lebih lanjut.

Kedua, pengguna dapat melakukan "Search?" untuk mencari konten. Jika search dilakukan, hasil pencarian akan ditampilkan, memungkinkan pengguna untuk langsung berinteraksi dengan artikel terkait. Ketiga, pengguna memiliki opsi untuk "Add Content?". Jika ya, sistem akan meminta pengguna untuk "Select

Content?” apakah akan menambahkan artikel, topik, atau sub-topik. Berdasarkan pilihan tersebut, alur akan mengarah ke proses masing-masing. Terakhir, terdapat fungsionalitas untuk “Change Language” yang merupakan aksi terpisah dari alur utama manajemen konten. Untuk aksi manajemen artikel individual, pengguna dapat memilih untuk “Edit Article” atau “Delete Article” dari tampilan artikel. Seluruh proses akan berakhir setelah tindakan pengguna selesai atau kembali ke tampilan awal halaman artikel.

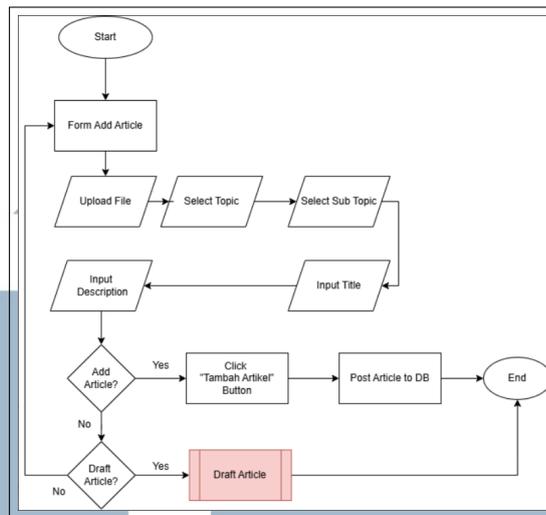


Gambar 3.10. Flowchart Management Article Page

1. Flowchart Add Article

Pada Gambar 3.11, *flow* ini dimulai dari tampilan *Form Add Article*, di mana admin melakukan input berupa pemilihan Topik dan Sub Topik, kemudian mengisi Judul serta Deskripsi dari artikel. Setelah semua data diisi, sistem memberikan opsi kepada admin untuk memutuskan apakah artikel tersebut langsung ditambahkan sebagai artikel terpublikasi atau tidak. Jika dipilih untuk ditambahkan, admin akan menekan tombol Tambah Artikel, lalu sistem akan menyimpan artikel tersebut ke dalam database dan proses berakhir.

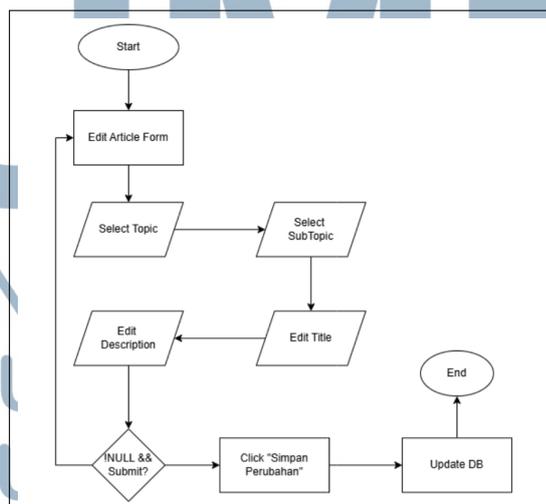
Namun, jika artikel belum ingin dipublikasikan, sistem akan menanyakan apakah artikel ingin disimpan sebagai *draft*. Jika ya, maka artikel akan disimpan dalam status *Draft*, dan jika tidak, maka proses akan kembali ke awal untuk pengisian ulang. Alur ini memberikan fleksibilitas bagi admin untuk memilih antara langsung mempublikasikan artikel atau menyimpannya terlebih dahulu sebagai *draft*. Untuk saat ini fitur *Draft* belum tersedia.



Gambar 3.11. *Flow Add Article*

2. *Flowchart Edit Article*

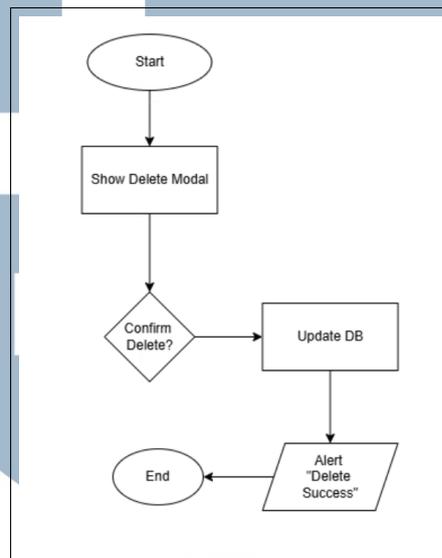
Pada Gambar 3.12 *flowchart* ini menggambarkan proses *edit* artikel oleh admin. Dari *Edit Article Form*, admin dapat memilih topik dan subtopik terkait, lalu mengubah judul dan deskripsi artikel. Setelah validasi berhasil dan tombol "Simpan Perubahan" diklik, data akan diperbarui dalam *database*. Proses ini terhubung dengan bagian *Click "Edit Button"?* dan *"Edit Article"* pada *flowchart* utama, dan memungkinkan admin memperbarui konten FAQ secara dinamis tanpa intervensi pengembang.



Gambar 3.12. *Flowchart Edit Article*

3. *Flowchart Delete Article*

Pada Gambar 3.13, *flowchart* ini menjelaskan proses penghapusan artikel dari sistem. Ketika admin memilih menghapus sebuah artikel, sistem akan menampilkan *modal* konfirmasi. Jika penghapusan dikonfirmasi, data akan diperbarui di *database* dan sistem menampilkan notifikasi sukses. *Flow* ini merupakan bagian dari *node Delete Article* dalam *flow* utama, dan berfungsi untuk menjaga kebersihan serta relevansi konten di dalam sistem FAQ.



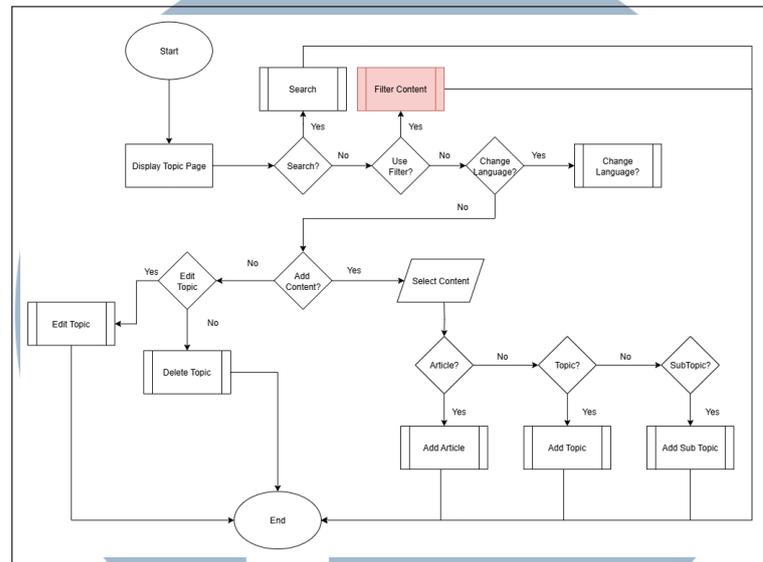
Gambar 3.13. *Flowchart Delete Article*

C.3 *Flowchart Management Topic Page*

Pada Gambar 3.14 *flowchart* ini menggambarkan alur interaksi pengguna (administrator) pada halaman Manajemen Topik di *dashboard Support Center*, yang berfokus pada pengelolaan topik utama. Proses dimulai dengan tampilan halaman daftar topik. Dari tampilan ini, pengguna dapat mengambil beberapa tindakan. Pertama, pengguna memiliki opsi untuk "Use Filter?". Jika ya, alur akan mengarah ke proses "Filter Content", yang memungkinkan penyaringan daftar topik berdasarkan kriteria tertentu. Jika tidak, atau setelah *filtering*, pengguna juga bisa memilih untuk "Change Language?".

Kedua, pengguna dapat melakukan "Search?". Jika search dipilih, alur akan mengarah ke proses "Search" untuk mencari topik berdasarkan *keyword*. Ketiga, ada opsi untuk "Add Content?". Jika ya, sistem akan meminta pengguna untuk "Select Content?" apakah akan menambahkan Artikel, Topik, atau Sub Topik. Berdasarkan pilihan tersebut, alur akan mengarah ke proses masing-masing. Keempat, pengguna dapat memilih untuk "Edit Topic" atau "Delete Topic".

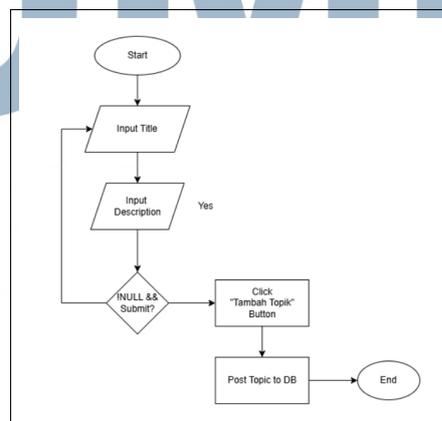
dari daftar topik yang ada, yang akan mengarah ke *form* pengeditan atau proses penghapusan topik. Semua alur ini dirancang untuk memberikan kontrol penuh kepada administrator dalam mengelola struktur topik konten.



Gambar 3.14. *Flowchart Management Topic Page*

1. *Flowchart Add Topic*

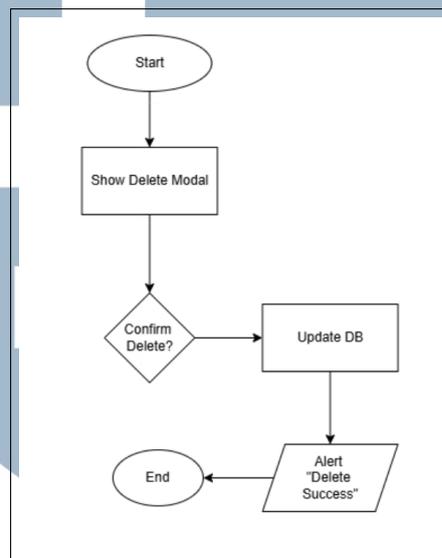
Pada Gambar 3.15, *flowchart* ini dimulai dari admin melakukan *input* untuk judul dari topik, dan deskripsi untuk topik tersebut. Setelah semua data diisi, sistem akan mengecek apakah semua *input* sudah terisi dan admin akan menekan tombol "Tambah Topik", lalu sistem akan menyimpan topik tersebut ke dalam *database* dan proses berakhir.



Gambar 3.15. *Flowchart Add Topic*

2. *Flowchart Delete Topic*

Pada Gambar 3.16, *flowchart* ini menjelaskan proses penghapusan topik dari sistem. Ketika admin memilih menghapus sebuah topik, sistem akan menampilkan modal konfirmasi. Jika penghapusan dikonfirmasi, data akan diperbarui di *database* dan sistem menampilkan notifikasi sukses. Sistem ini berfungsi untuk menjaga kebersihan serta relevansi konten di dalam sistem FAQ.



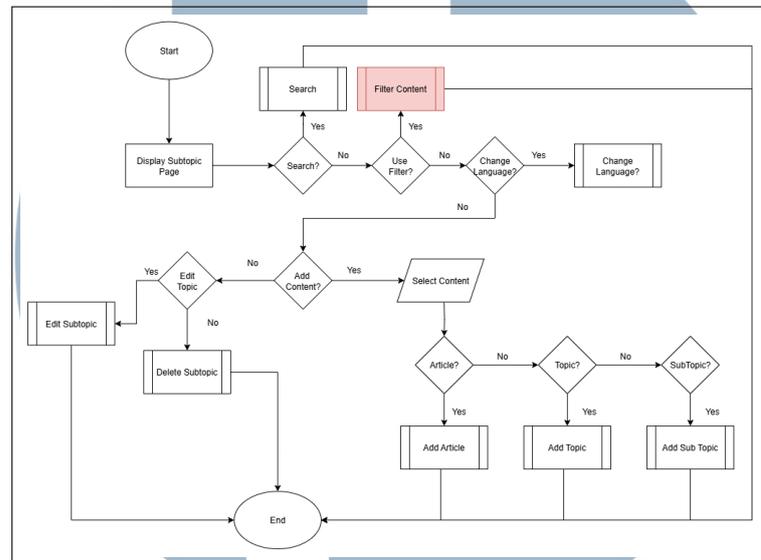
Gambar 3.16. *Flowchart Delete Topic*

C.4 *Flowchart Management Sub Topic Page*

Pada Gambar 3.17 *flowchart* ini menggambarkan alur interaksi pengguna (administrator) pada halaman Manajemen Sub Topik di *dashboard Support Center*, yang berfokus pada pengelolaan topik utama. Proses dimulai dengan tampilan halaman daftar sub topik. Dari tampilan ini, pengguna dapat mengambil beberapa tindakan. Pertama, pengguna memiliki opsi untuk "Use Filter?". Jika ya, alur akan mengarah ke proses "Filter Content", yang memungkinkan penyaringan daftar topik berdasarkan kriteria tertentu. Jika tidak, atau setelah *filtering*, pengguna juga bisa memilih untuk "Change Language?".

Kedua, pengguna dapat melakukan "Search?". Jika search dipilih, alur akan mengarah ke proses "Search" untuk mencari sub topik berdasarkan *keyword*. Ketiga, ada opsi untuk "Add Content?". Jika ya, sistem akan meminta pengguna untuk "Select Content?" apakah akan menambahkan Artikel, Topik, atau Sub Topik. Berdasarkan pilihan tersebut, alur akan mengarah ke proses masing-masing.

Keempat, pengguna dapat memilih untuk "Edit Sub Topic" atau "Delete Sub Topic" dari daftar sub topik yang ada, yang akan mengarah ke *form* pengeditan atau proses penghapusan sub topik. Semua alur ini dirancang untuk memberikan kontrol penuh kepada administrator dalam mengelola struktur sub topik konten.

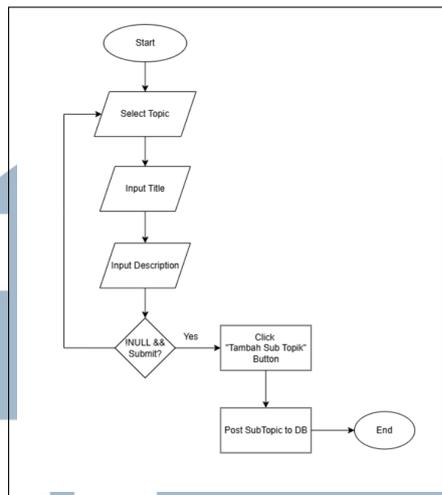


Gambar 3.17. Flowchart Management Sub Topic Page

1. Flowchart Add Sub Topic

Pada Gambar 3.18, *flowchart* ini dimulai dari admin melakukan *select* topik yang diinginkan, lalu memberi *input* untuk judul dari sub topik, dan deskripsi untuk sub topik tersebut. Setelah semua data diisi, sistem akan mengecek apakah semua *input* sudah terisi dan admin akan menekan tombol "Tambah Sub Topik", lalu sistem akan menyimpan sub topik tersebut ke dalam *database* dan proses berakhir.

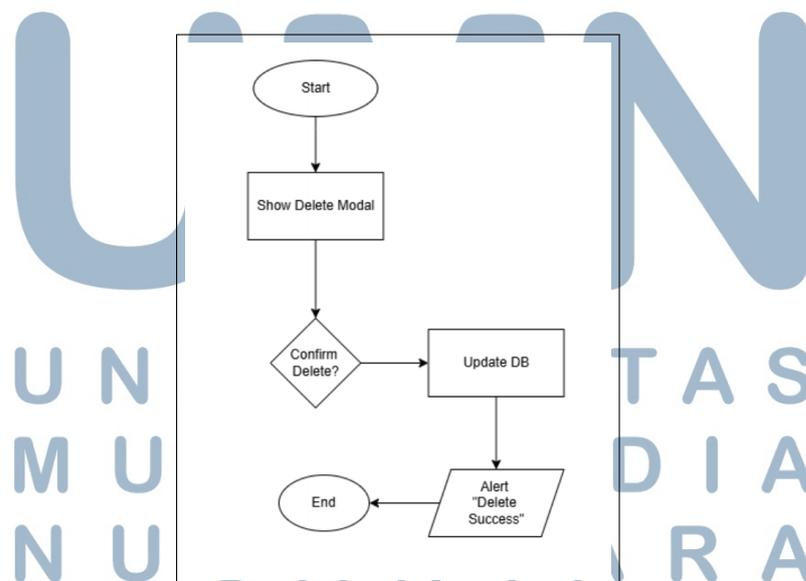
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.18. *Flowchart Add Sub Topic*

2. *Flowchart Delete Sub Topic*

Pada Gambar 3.19, *flow* ini menjelaskan proses penghapusan sub topik dari sistem. Ketika admin memilih menghapus sebuah sub topik, sistem akan menampilkan *modal* konfirmasi. Jika penghapusan dikonfirmasi, data akan diperbarui di *database* dan sistem menampilkan notifikasi sukses. Sistem ini berfungsi untuk menjaga kebersihan serta relevansi konten di dalam sistem FAQ.



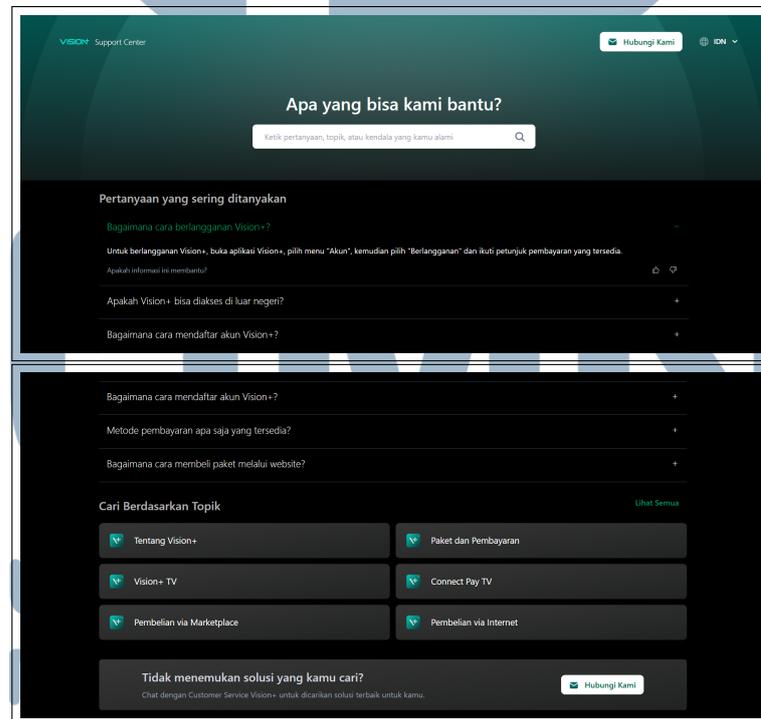
Gambar 3.19. *Flowchart Delete Sub Topic*

3.4 Implementasi

A Halaman *User*

Pada Gambar 3.20 ini merupakan implementasi dari halaman *Home User*. Untuk halaman ini, pengguna dapat mencari solusi atas kendala yang dialami melalui kolom pencarian dengan memasukkan kata kunci berupa pertanyaan, topik, atau masalah umum. Di bawah kolom pencarian, sistem akan menampilkan pertanyaan yang sering ditanyakan (FAQ) secara langsung, dan pengguna dapat mengklik salah satu pertanyaan untuk melihat jawabannya secara lengkap.

Selain itu, pengguna juga dapat menelusuri informasi berdasarkan kategori melalui bagian "Cari Berdasarkan Topik", seperti Tentang Vision+, Paket dan Pembayaran, Vision+ TV, dan lainnya. Jika pertanyaan atau kendala yang dihadapi tidak ditemukan, pengguna dapat menghubungi *Customer Service* secara langsung dengan menekan tombol "Hubungi Kami" yang tersedia di bagian atas dan bawah halaman.

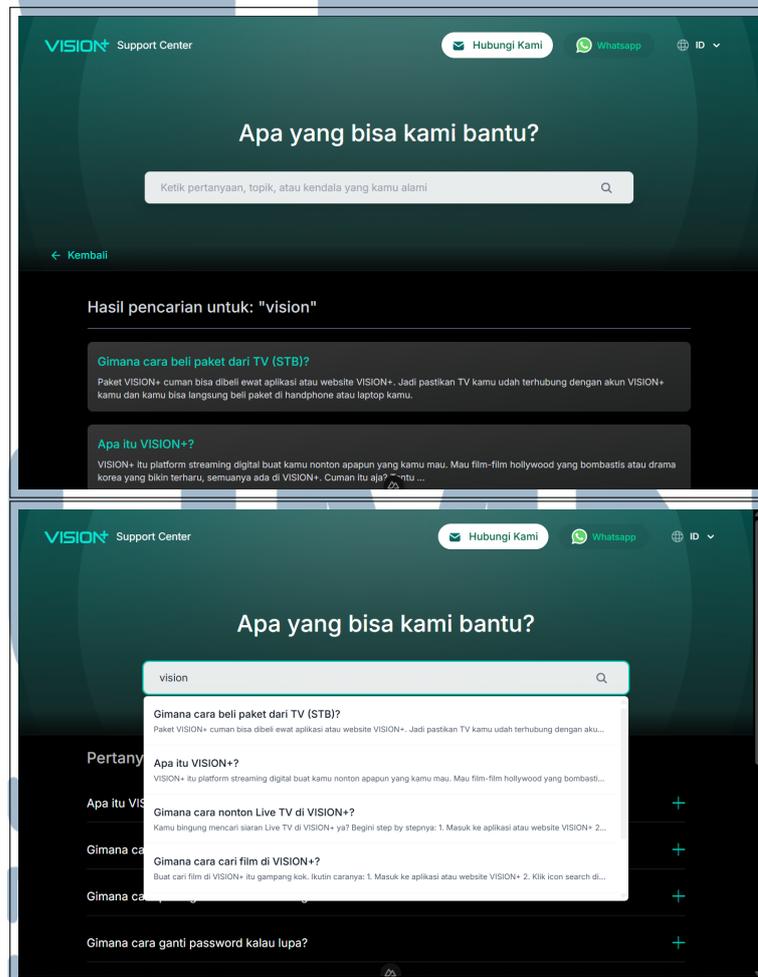


Gambar 3.20. Halaman *User*

A.1 Halaman *Search Result*

Pada Gambar 3.21 ini merupakan implementasi dari fitur *Search* dimana ketika *user* melakukan *search* sistem akan melakukan *redirect* menuju halaman *Search Result* yang akan menampilkan data artikel yang dikirim oleh API sesuai dengan *keyword* yang diinput. Jika *user* memilih salah satu artikel maka akan dibawa ke halaman *Find by Topic* yang akan menampilkan artikel sesuai dengan artikel yang dipilih.

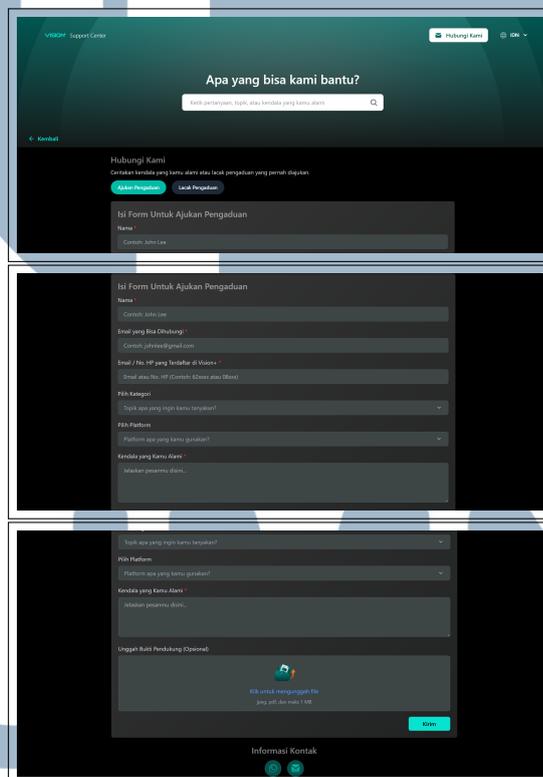
Lalu jika *user* melakukan pencarian berdasarkan klik langsung artikel yang muncul pada *dropdown* maka sistem akan langsung melakukan *direct* ke halaman *Find by Topic* yang akan menampilkan artikel yang dipilihnya.



Gambar 3.21. Halaman *Search Result*

A.2 Halaman *Form Report*

Pada Gambar 3.22 untuk halaman ini, pengguna dapat mengajukan pengaduan terkait kendala yang dialami saat menggunakan layanan Vision+. Terdapat dua pilihan utama, yaitu Ajukan Pengaduan dan Lacak Pengaduan. Ketika pengguna memilih untuk mengajukan pengaduan, sistem akan menampilkan form isian yang harus dilengkapi. Setelah seluruh data diisi, pengguna dapat menekan tombol "Kirim" untuk mengajukan pengaduan. Halaman ini bertujuan untuk memfasilitasi pengguna yang tidak menemukan solusi melalui FAQ yang tersedia, agar tetap mendapatkan bantuan langsung dari tim *Customer Service* Vision+.



The image shows three sequential screenshots of a web form for reporting a complaint on the Vision+ support center. The top screenshot shows the search bar and the 'Hubungi Kami' (Contact Us) button. The middle screenshot shows the 'Isi Form Untuk Ajukan Pengaduan' (Fill out the form to report a complaint) section, which includes fields for Name, Email (for follow-up), and a dropdown menu for 'Pilih Platform' (Select Platform). The bottom screenshot shows the 'Kendala yang Kamu Alami' (The problem you are experiencing) section, which includes a dropdown menu for 'Pilih Platform' and a 'Kirim' (Send) button.

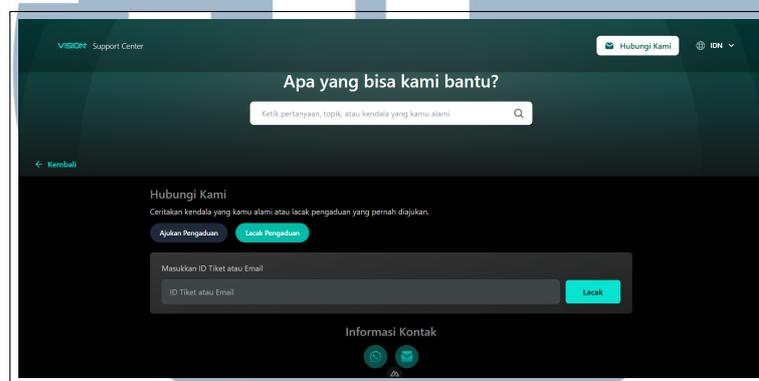
Gambar 3.22. Halaman *Form Report*

A.3 Halaman *Tracking*

Pada Gambar 3.23 untuk halaman ini, pengguna dapat melacak status pengaduan yang sebelumnya telah diajukan. Pengguna cukup memilih tombol "Lacak Pengaduan", lalu memasukkan ID Tiket atau alamat Email yang digunakan saat pengaduan dikirimkan. Setelah itu, pengguna dapat menekan tombol "Lacak" untuk melihat informasi detail terkait status penanganan pengaduan yang

bersangkutan.

Fitur ini disediakan agar pengguna dapat memantau perkembangan penanganan masalah tanpa perlu menghubungi *Customer Service* secara langsung, sehingga proses komunikasi menjadi lebih efisien dan transparan. Halaman ini merupakan bagian dari sistem layanan mandiri yang dirancang untuk meningkatkan pengalaman dan kepuasan pengguna. Namun saat ini untuk fungsional fitur "Lacak Pengaduan" masih dalam tahap perencanaan sehingga belum siap untuk dikembangkan.

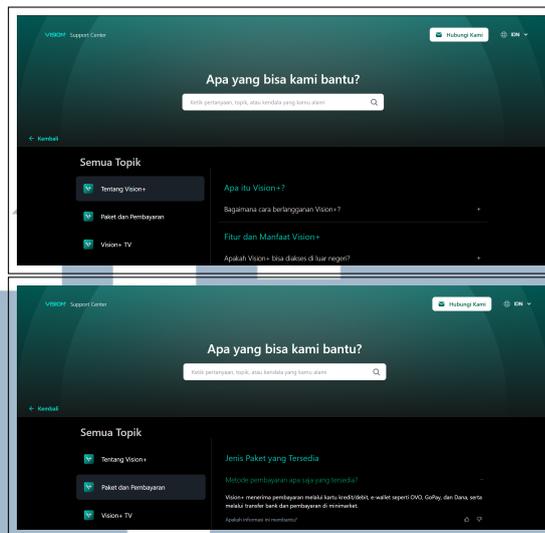


Gambar 3.23. Halaman *Tracking*

A.4 Halaman *Find by Topic*

Pada Gambar 3.24 untuk halaman ini menampilkan daftar artikel yang tersedia pada sistem *Vision+ Support Center*. Pengguna dapat melihat dan memilih kategori/topik tertentu seperti "Tentang Vision+", "Paket dan Pembayaran", atau "Vision+ TV". Setelah memilih salah satu topik, sistem akan menampilkan daftar pertanyaan dan jawaban yang termasuk dalam kategori tersebut pada bagian kanan layar.

Pertanyaan disusun berdasarkan subtopik atau di dalam topik utama, seperti "Apa itu Vision+?" atau "Metode pembayaran apa saja yang tersedia?". Setiap pertanyaan dapat diklik untuk menampilkan jawaban lengkapnya secara langsung tanpa membuka halaman baru, sehingga memudahkan pengguna mendapatkan informasi secara cepat dan efisien. Halaman ini dirancang untuk membantu pengguna menjelajahi informasi berdasarkan kategori tertentu apabila *user* tidak menggunakan fitur pencarian di halaman utama.

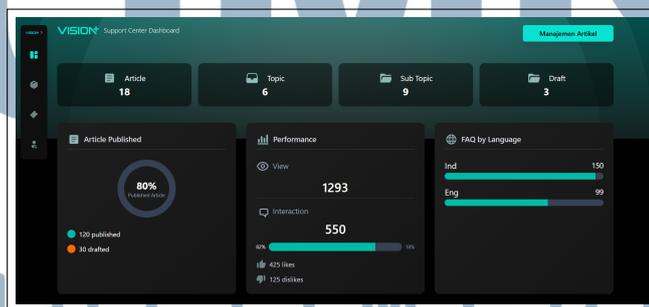


Gambar 3.24. Halaman *Find by Topic*

B Halaman Admin

B.1 Halaman *Overview*

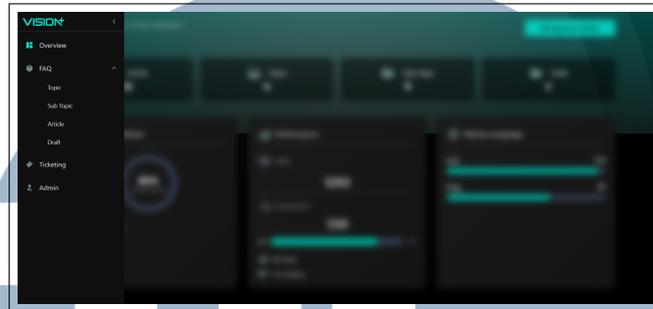
Pada gambar 3.25 ini merupakan implementasi dari halaman *Overview* pada sistem *Support Center Vision+*. Halaman ini menampilkan ringkasan informasi penting seperti jumlah artikel, topik, subtopik, dan artikel yang masih dalam status *draft*. Tampilan ini dirancang dengan antarmuka yang sederhana dan modern menggunakan tema gelap serta elemen visual berbasis kartu untuk memudahkan pembacaan data.



Gambar 3.25. Halaman *Overview*

Dashboard ini juga menampilkan beberapa statistik tambahan seperti performa artikel dan distribusi FAQ berdasarkan bahasa, sehingga admin dapat memantau aktivitas dan perkembangan konten secara umum. Navigasi *sidebar*

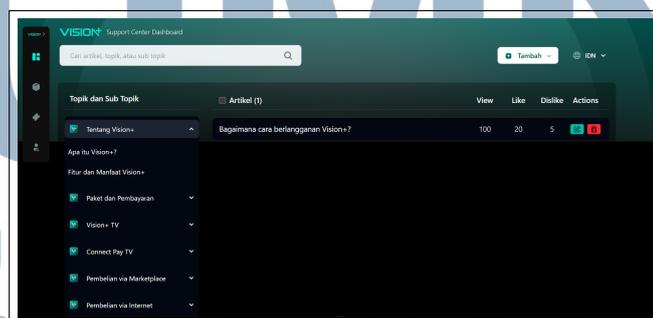
disediakan di sisi kiri untuk mengakses berbagai fitur manajemen lainnya seperti pada Gambar 3.26.



Gambar 3.26. *Sidebar Navigation*

B.2 Management Article Page

Pada Gambar 3.27 merupakan implementasi dari halaman Artikel pada sistem *Support Center Vision+*. Halaman ini menampilkan informasi yang tersedia, terlihat daftar "Topik dan Sub Topik" di sisi kiri, di mana pengguna dapat menelusuri kategori konten. Di sisi kanan, "Artikel (1)" menunjukkan detail artikel yang terkait dengan topik yang dipilih, lengkap dengan metrik *View*, *Like*, dan *Dislike*. Adanya kolom *Actions* dengan ikon *edit* dan *hapus* menunjukkan fungsionalitas untuk mengelola artikel secara langsung dari tampilan ini. Desain antarmuka yang bersih dan terorganisir ini memberikan kemudahan navigasi bagi pengguna untuk memantau dan mengelola konten dukungan.

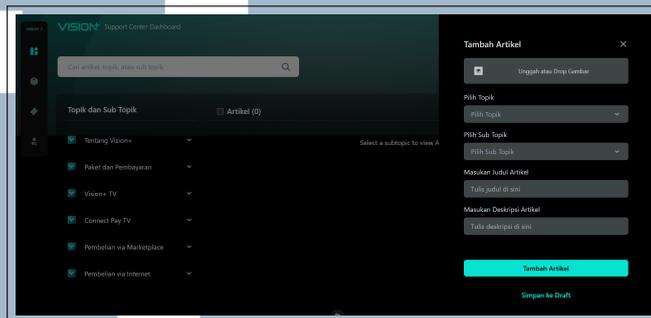


Gambar 3.27. *Management Article Page*

1. Add Article

Pada Gambar 3.28, fitur "Tambah Artikel" memungkinkan administrator untuk menambahkan konten FAQ baru ke dalam sistem. *Interface* ini dilengkapi dengan area unggah *file*, dropdown untuk memilih Topik dan Sub

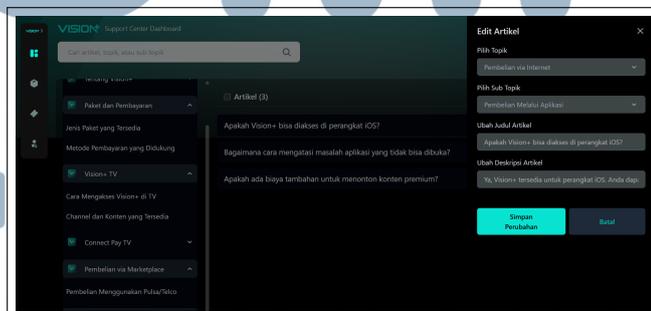
Topik, serta *field* untuk memasukkan judul dan deskripsi artikel. Pengguna dapat langsung mempublikasikan artikel dengan tombol "Tambah Artikel" berwarna hijau atau menyimpannya sebagai *draft* menggunakan tombol "Simpan ke Draft" jika konten belum siap dipublikasikan. Fitur ini dirancang dengan tampilan yang sederhana namun fungsional untuk memudahkan pengelolaan konten FAQ secara efisien.



Gambar 3.28. Form Add Article

2. Edit Article

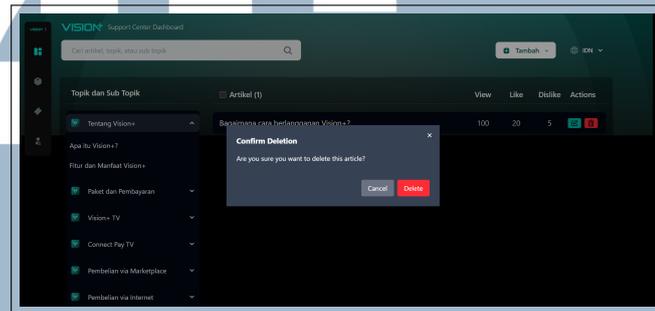
Pada Gambar 3.29, fitur "Edit Artikel" pada halaman ini memungkinkan administrator untuk memodifikasi artikel FAQ yang sudah ada dalam sistem. Panel ini menampilkan informasi artikel yang sedang diedit, termasuk isi dari artikel tersebut. Setelah melakukan perubahan yang diperlukan, admin dapat menyimpan perubahan dengan tombol "Simpan Perubahan" berwarna hijau atau membatalkan proses pengeditan dengan tombol "Batal". Fitur ini memungkinkan pengelola konten untuk memperbarui informasi FAQ secara mudah tanpa perlu membuat artikel baru.



Gambar 3.29. Form Edit Article

3. Delete Article

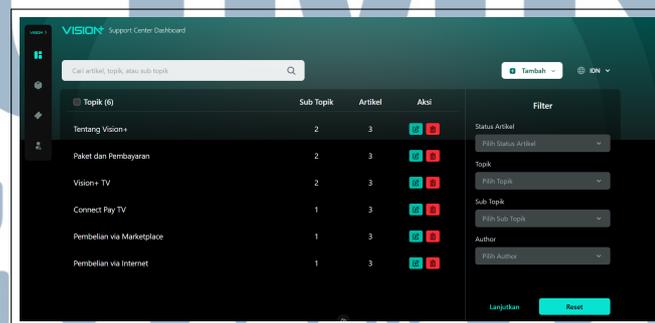
Pada Gambar 3.30, fitur "Hapus Artikel" berfungsi sebagai pengaman saat administrator akan menghapus sebuah artikel FAQ. Fitur ini muncul setelah admin mengklik ikon tempat sampah yang berada di samping artikel yang ingin dihapus. Dialog konfirmasi digunakan untuk memastikan admin tidak menghapus artikel secara tidak sengaja.



Gambar 3.30. Modal Delete Article

B.3 Management Topic Page

Pada Gambar 3.31 merupakan implementasi dari halaman topik pada *Support Center Vision+* dengan fokus untuk melakukan manajemen topik. Pada Kolom Topik(6) menunjukkan kategori dari topik yang tersedia, dan setiap topik akan menunjukkan metrik sub topik dan artikel yang terikat oleh topik tersebut. Untuk fitur Filter pada sisi kanan menyediakan opsi untuk menyaring daftar topik berdasarkan berbagai kriteria, seperti status artikel, topik, sub topik, dan penulis, yang sangat membantu dalam mengelola basis pengetahuan yang besar.

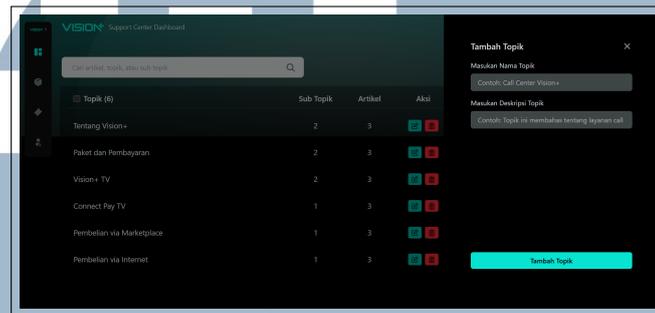


Gambar 3.31. Management Topic Page

1. Add Topic

Pada Gambar 3.32 ini merupakan implementasi dari fitur tambah topik yang berfungsi untuk administrator saat ingin menambah topik. Pada *form* ini

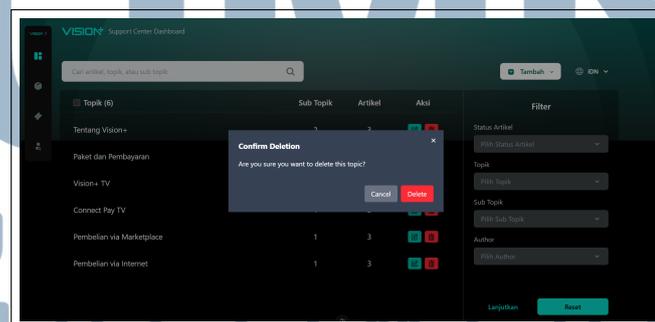
tersedia beberapa input yang dibutuhkan lalu jika sudah selesai mengisinya, admin dapat menekan tombol "Tambah Topik" untuk menyimpan topik baru ke dalam sistem. Fitur ini esensial dalam membangun dan memperluas struktur basis pengetahuan, memungkinkan kategorisasi konten dukungan secara modular.



Gambar 3.32. Form Add Topic

2. Delete Topic

Pada Gambar 3.33 ini, merupakan implementasi dari fitur hapus topik yang berfungsi untuk melakukan penghapusan topik. Fitur ini akan muncul dalam bentuk *modal* setelah admin mengklik tombol logo *delete*, dialog konfirmasi digunakan untuk memastikan admin tidak menghapus topik secara tidak sengaja. Pilihan "Cancel" dan "Delete" memungkinkan pengguna untuk mempertimbangkan kembali atau melanjutkan penghapusan, menjaga konsistensi dalam prosedur keamanan data di seluruh sistem.

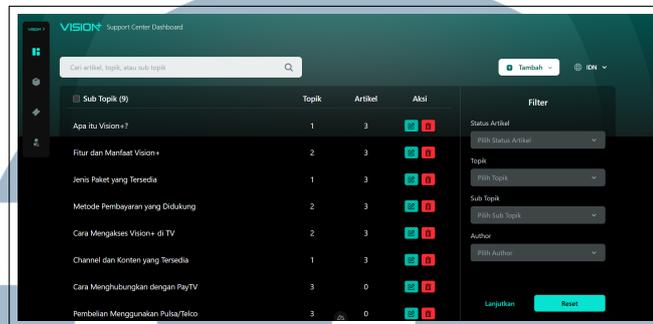


Gambar 3.33. Modal Delete Topic

B.4 Management Sub Topic Page

Pada Gambar 3.34 merupakan implementasi dari halaman sub topik yang berfokus untuk pengelolaan topik utama. Pada kolom sub topik terdapat *list* sub

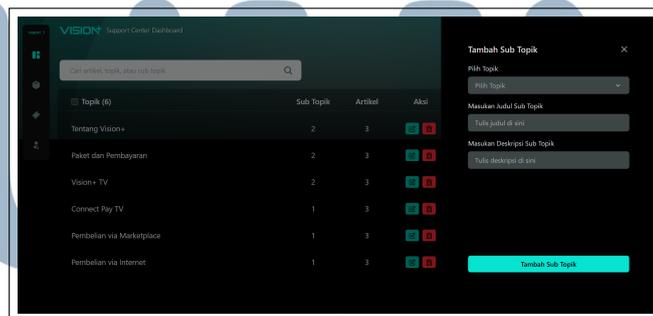
topik yang tersedia lalu akan menunjukkan metrik jumlah artikel yang terkait dengan sub topik tersebut dan juga dapat melihat sub topik ini bagian dari topik.



Gambar 3.34. Management Sub Topic Page

1. Add Sub Topic

Pada Gambar 3.35 ini, menunjukkan implementasi dari fitur tambah sub topik yang berfungsi untuk administrator yang ingin menambah sub topik dengan mengisi beberapa input yang tersedia pada *form*. Jika sudah selesai melakukan input maka langsung menekan tombol "Tambah Sub Topik" untuk menyimpan sub topik baru ke dalam sistem. Desain ini menunjukkan bagaimana sistem mendukung pembentukan struktur konten yang terorganisir dengan baik, memfasilitasi navigasi yang intuitif bagi pengguna akhir.

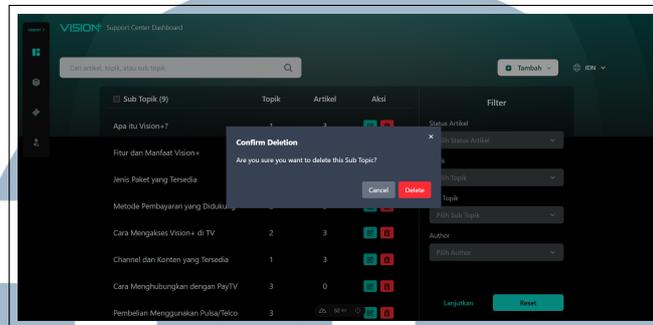


Gambar 3.35. Form Add Sub Topic

2. Delete Sub Topic

Pada Gambar 3.36 ini, merupakan implementasi dari fitur hapus sub topik yang berfungsi untuk melakukan penghapusan sub topik. Fitur ini akan muncul dalam bentuk *modal* setelah admin mengklik tombol logo *delete*, dialog konfirmasi digunakan untuk memastikan admin tidak menghapus topik secara tidak sengaja. Pilihan "Cancel" dan "Delete" memungkinkan

pengguna untuk mempertimbangkan kembali atau melanjutkan penghapusan, menjaga konsistensi dalam prosedur keamanan data di seluruh sistem.



Gambar 3.36. Delete Sub Topic

3.5 Spesifikasi Sistem

Berikut adalah spesifikasi dari *software* yang digunakan dalam pembuatan halaman FAQ pada *website* Vision+.

1. *Gitlab*
2. *Brave*
3. *Google Chrome*
4. *Visual Studio Code*
5. *Figma*
6. *Diagrams.net*
7. *Jira Software*

Berikut merupakan spesifikasi *hardware* yang digunakan.

1. *Operating Systems: Windows*
2. *Random Access Memory (RAM): 16 GB*
3. *Storage: 500 GB (SSD)*
4. *Processor: 12th Gen Intel(R) Core(TM) i5-12500H*

3.6 Kendala dan Solusi yang Ditemukan

3.6.1 Kendala

Berikut ini merupakan kendala yang dialami selama pengerjaan magang ini.

1. Kemunduran jadwal mulai magang yang membuat pengerjaan project kedepannya akan mengalami sedikit waktu nantinya.
2. Perbedaan tenggat waktu yang bervariasi untuk setiap tugas yang diberikan
3. Adanya pengalihan tugas dikarenakan *project* utama sedang terkena *blocked* yang mengakibatkan pengerjaan menjadi tertunda.
4. Terdapat kendala saat integrasi dengan *Backend* karena logika sistem nya yang cukup kompleks dan juga relasi *database* yang berbeda.

3.6.2 Solusi

Berikut ini merupakan solusi yang ditemukan untuk kendala yang dialami.

1. Menentukan prioritas tugas yang dapat dikerjakan terlebih dahulu dengan pertimbangan durasi waktu.
2. Jika ada tugas yang terkena *blocked* maka melakukan inisiasi dahulu agar ketika sudah dilanjutkan *project* tidak tertinggal jauh.
3. Menjaga komunikasi dan koordinasi dengan keseluruhan tim agar proses berjalannya *project* tidak adanya miskomunikasi.
4. Memfokuskan pada fitur utama agar dapat terselesaikan terlebih dahulu agar *project* tetap bergerak maju.

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