

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Kegiatan magang dilaksanakan di PT Berkat Jocellyndo Abadi, sebuah perusahaan yang bergerak di bidang properti, mencakup layanan konsultasi properti dan agen properti. Posisi yang dijalani adalah sebagai *Full-stack Developer*, dengan tanggung jawab dalam pengembangan *website property listing* untuk mendukung kegiatan pemasaran properti perusahaan. Selama pelaksanaan magang, koordinasi dilakukan secara langsung dengan *Supervisor*, yaitu Bapak Frendy Harlina yang menjabat sebagai *Commissioner* di perusahaan. Koordinasi ini dilakukan untuk memastikan pekerjaan yang dilaksanakan sesuai dengan kebutuhan dan standar yang ditetapkan oleh perusahaan.

Tugas yang dikerjakan mencakup pengembangan di sisi *front-end* dan *back-end*. Pada bagian *front-end*, dilakukan pembangunan tampilan antarmuka pengguna (*UI*) *website* yang memudahkan pengguna dalam melihat *listing* properti, detail unit, galeri foto, serta fitur interaktif lainnya dengan desain yang responsif dan *user-friendly*. Teknologi yang digunakan mencakup *Next.js*, *React*, dan *Tailwind CSS* untuk memastikan tampilan yang menarik, mudah dinavigasi, serta penggunaan warna yang sesuai dan konsisten dengan identitas perusahaan. Sementara itu, pada bagian *back-end*, dikembangkan sistem manajemen data properti yang terintegrasi dengan *Supabase* sebagai *database* dan media penyimpanan gambar, sehingga data properti dapat ditampilkan secara dinamis di *website*. Integrasi ini memungkinkan data unit dan foto properti yang ditampilkan di *website* selalu terbarui secara otomatis tanpa perlu melakukan perubahan manual pada kode *website*.

3.2 Tugas yang Dilakukan

Pelaksanaan magang dimulai dengan mencari referensi mengenai *website* properti *listing* yang sudah ada sebagai acuan dalam menentukan struktur, fitur, serta tampilan yang sesuai dengan kebutuhan pengguna. Referensi ini mencakup berbagai elemen seperti tata letak halaman, penyajian informasi unit properti, hingga tampilan galeri foto.

Setelah itu, dilakukan pemilihan *library* dan *framework* yang akan digunakan dalam pengembangan website. Teknologi yang digunakan meliputi

React sebagai library utama, Next.js sebagai framework, Tailwind CSS untuk desain antarmuka, serta Supabase sebagai platform *back-end* yang digunakan untuk manajemen data dan penyimpanan media.

Setelah teknologi ditentukan, pembangunan struktur awal *website* dimulai dengan pembuatan halaman utama (*homepage*), halaman daftar properti (*property listing*), serta halaman detail unit. Proses pengembangan dilakukan secara bertahap, dimulai dari pembuatan komponen dasar seperti *navbar*, *footer*, dan tampilan unit properti, hingga penambahan fitur interaktif seperti galeri foto dan bagian yang menampilkan keunggulan atau kelebihan dari kawasan perumahan.

Pengelolaan data menggunakan layanan Supabase secara langsung, tanpa sistem login atau dashboard admin. Data unit dan gambar properti diunggah serta diperbarui melalui antarmuka yang disediakan oleh Supabase. Dengan pendekatan ini, perubahan data yang dilakukan melalui Supabase akan langsung terhubung dan ditampilkan secara dinamis di dalam *website*, tanpa perlu melakukan pembaruan manual pada kode.

Selama proses pengembangan, penulis secara rutin berkoordinasi dengan *Supervisor* untuk mendapatkan umpan balik dan arahan teknis. Hal ini memastikan bahwa fitur yang dibangun sesuai dengan harapan perusahaan serta memenuhi kebutuhan pengguna secara fungsional maupun estetika.

Selain proyek website properti, tugas lain yang dikerjakan adalah merancang dan mengembangkan proyek website undangan pernikahan online bernama Invitated. Proses dimulai dengan pembuatan desain menggunakan Figma untuk satu halaman utama (*homepage*) yang menampilkan jasa penjualan undangan online, serta tiga template undangan yang berbeda. Setiap template dirancang dengan tampilan unik dan menggunakan aset visual seperti bunga, hiasan, dan dekorasi yang memiliki animasi atau gerakan untuk mempercantik tampilan.

Setelah desain disetujui, dilakukan proses konversi desain menjadi kode menggunakan React dan Tailwind CSS. Masing-masing template diimplementasikan ke dalam bentuk website interaktif. Setelah proses pengembangan selesai, dilakukan tahapan *hosting* agar website undangan dapat diakses secara publik. Tugas ini tidak hanya melibatkan aspek teknis dalam pengkodean, tetapi juga penguasaan estetika desain dan animasi agar tampilan undangan terasa menarik dan profesional.

Selama proses pengembangan kedua proyek, dilakukan koordinasi secara berkala dengan *supervisor* untuk mendapatkan umpan balik dan arahan teknis. Hal ini memastikan bahwa fitur yang dibangun sesuai dengan harapan perusahaan serta

memenuhi kebutuhan pengguna secara fungsional maupun estetika.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan magang dilakukan selama 20 minggu, terhitung sejak tanggal 3 Februari 2025 hingga 3 Juli 2025. Uraian kegiatan magang setiap minggunya dapat dilihat pada Tabel 3.1.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu ke-	Pekerjaan yang dilakukan
1	Mempelajari framework yang akan digunakan serta melihat referensi website lain yang sudah ada.
2	Mulai membangun tampilan website <i>property listing</i> Perumahan Permata Mutiara Maja.
3	Perevisian tampilan serta membuat <i>database</i> (tipe unit, galeri, fasilitas) yang terintegrasi dengan <i>website</i> .
4	Menyelesaikan tampilan dan fungsionalitas website, serta memastikan data dari database terintegrasi dan tampil dengan benar.
5	Membuat desain awal website undangan pernikahan online menggunakan Figma dimulai dari <i>home page</i> .
6	Mendesain satu halaman template undangan pernikahan di Figma beserta pencarian aset visual (bunga, foto, dekorasi).
7	Menyelesaikan template undangan pertama dan melanjutkan pembuatan template kedua dengan elemen desain dan aset yang berbeda.
8	Melanjutkan pembuatan template ketiga undangan pernikahan. Memulai proyek website baru tentang unit apartemen dengan membuat komponen awal seperti <i>navbar</i> , <i>carousel</i> (gambar dari database), dan <i>section about</i> . Juga mulai mengerjakan komponen <i>card unit</i> (berisi gambar, nama unit, ukuran, kamar tidur, kamar mandi, harga) yang terhubung ke database.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu ke-	Pekerjaan yang dilakukan
9	Membuat bagian fasilitas pada website apartemen. Dilanjutkan dengan pembuatan bagian lokasi dan integrasi video YouTube, serta footer website. Setelah itu, kembali melanjutkan proyek website undangan pernikahan online dengan menambahkan bagian galeri dan lokasi. Kemudian membuat bagian RSVP dan <i>wedding wishes</i> pada satu template undangan.
10	Membuat bagian <i>gift</i> pada salah satu template undangan pernikahan. Kemudian melakukan revisi bertahap pada pergerakan animasi bunga yang ada pada template, termasuk revisi pada template pertama yang telah selesai sebelumnya. Di akhir minggu, dilakukan revisi warna pada template website untuk meningkatkan tampilan visual secara keseluruhan.
11	Melakukan kegiatan <i>self learning</i> untuk meningkatkan pemahaman teknis dan estetika desain. Di akhir minggu, melakukan submit revisi dan pengecekan oleh <i>supervisor</i> , serta melakukan revisi pada elemen hiasan (dekorasi) pada template website undangan pernikahan.
12	Melanjutkan revisi elemen bunga pada hiasan template website undangan pernikahan. Kegiatan dilanjutkan dengan <i>self learning</i> untuk memperdalam pemahaman teknis. Di akhir minggu dilakukan proses hosting atau peluncuran website undangan pernikahan.
13	Melanjutkan proses hosting atau peluncuran (<i>launch</i>) website undangan pernikahan. Setelah itu, mulai membuat template baru untuk website undangan, dimulai dari bagian awal yaitu halaman pembukaan undangan.
14	Membuat bagian utama pada template ke-tiga website undangan pernikahan, dimulai dari bagian paling atas. Kemudian menambahkan bagian nama pengantin, dilanjutkan dengan pembuatan bagian “Our Story” dan galeri foto pernikahan

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu ke-	Pekerjaan yang dilakukan
15	Melanjutkan penggerjaan template ketiga website undangan pernikahan, mencakup pembuatan bagian lokasi pernikahan, RSVP, serta bagian ucapan terima kasih di akhir undangan
16	Menyelesaikan bagian ucapan terima kasih pada template ketiga website undangan pernikahan. Setelah itu dilakukan proses pengumpulan kepada supervisor dan menunggu konfirmasi serta hasil review
17	Melakukan pengumpulan template website undangan pernikahan kepada supervisor untuk dilakukan peninjauan dan konfirmasi. Kegiatan ini bertujuan memastikan bahwa hasil pengembangan template sudah sesuai dengan kebutuhan dan standar yang telah disepakati, serta siap untuk mendapatkan masukan atau persetujuan dari supervisor
18	Melakukan revisi terhadap elemen animasi bunga dan hiasan pada website undangan pernikahan. Revisi ini difokuskan pada peningkatan estetika dan kelancaran animasi agar tampilan website menjadi lebih menarik dan sesuai dengan masukan dari supervisor. Hasil revisi kemudian diserahkan kembali untuk ditinjau
19	Melakukan beberapa kegiatan utama, yaitu mengikuti meeting online bersama supervisor, mencari asset-asset baru (seperti ikon, ilustrasi, atau elemen desain lainnya) untuk memperkaya tampilan template website undangan pernikahan, serta melakukan <i>self-learning</i> untuk meningkatkan pemahaman teknis dan estetika dalam pengembangan website tersebut.

3.4 Realisasi Proyek Website

Selama pelaksanaan magang di PT Berkat Jocellyndo Abadi, penulis telah menyelesaikan dua proyek utama dalam bentuk pengembangan website.

Proyek pertama adalah pembuatan website property listing untuk memfasilitasi penyajian informasi unit-unit properti yang ditawarkan oleh perusahaan. Website ini dirancang untuk menampilkan daftar properti, detail unit, galeri foto, serta informasi pendukung lainnya secara dinamis dan terstruktur.

Proyek kedua yang dikerjakan adalah perancangan website undangan pernikahan online bernama Invitated. Website ini mencakup perancangan antarmuka di Figma serta implementasi tiga template undangan dengan desain visual yang interaktif dan responsif. Selain itu, dibuat juga halaman utama (homepage) sebagai halaman promosi layanan pembuatan undangan digital. Setiap template dilengkapi dengan elemen dekoratif seperti bunga dan ornamen yang dianimasikan, serta telah melalui proses hosting untuk dapat diakses secara publik.

3.4.1 Perancangan Website Property Listing Perumahan Permata Mutiara Maja

Website property listing ini dikembangkan untuk menyajikan informasi unit-unit properti di kawasan Perumahan Permata Mutiara Maja secara terstruktur dan menarik. Informasi yang ditampilkan mencakup daftar rumah, detail spesifikasi, galeri foto, serta keunggulan lokasi perumahan. Halaman utama terdiri atas beberapa bagian, yaitu carousel, kartu unit properti, informasi tentang perumahan, daftar keunggulan, testimoni pengguna, dan footer. Seluruh desain antarmuka dibuat responsif dan ramah pengguna guna memberikan pengalaman akses yang nyaman di berbagai perangkat. Pengelolaan data dilakukan melalui Supabase, yang memungkinkan konten diperbarui secara dinamis melalui dashboard tanpa perlu melakukan perubahan langsung pada kode program.

Untuk memberikan gambaran yang lebih jelas mengenai struktur dan tampilan halaman utama *website property listing* Perumahan Permata Mutiara Maja, berikut disajikan beberapa tangkapan layar dari setiap bagian utama beserta potongan kode program yang relevan. Penjelasan singkat disertakan untuk menjelaskan fungsi dan tujuan masing-masing elemen dalam mendukung penyampaian informasi kepada pengguna.

A Struktur Database Website Perumahan Permata Mutiara Maja

Untuk mendukung fungsionalitas *website property listing* Perumahan Permata Mutiara Maja, digunakan beberapa tabel dalam *database* yang dirancang

dan diatur melalui Supabase. Struktur *database* ini berfungsi untuk menyimpan dan mengelola data gambar carousel, informasi unit properti yang tersedia, serta informasi deskriptif mengenai perumahan. Adapun tiga tabel utama yang digunakan dalam sistem ini adalah tabel carousel, products, dan about.

Tabel 3.2 menunjukkan struktur dari tabel `carousel`, yang digunakan untuk menyimpan data gambar-gambar yang akan ditampilkan pada bagian `carousel` di halaman utama website. Setiap gambar diidentifikasi dengan ID unik, memiliki URL gambar, serta mencatat waktu data tersebut ditambahkan ke dalam sistem.

Tabel 3.2. Struktur Tabel `carousel`

Nama Kolom	Tipe Data	Keterangan
<code>id</code>	<code>uuid</code>	Primary key
<code>image_url</code>	<code>text</code>	URL gambar <code>carousel</code>
<code>created_at</code>	<code>timestamp</code>	Waktu data dibuat

Tabel 3.3 merupakan struktur data yang digunakan untuk menyimpan informasi unit rumah yang tersedia di Perumahan Permata Mutiara Maja. Setiap baris dalam tabel ini merepresentasikan satu unit rumah, lengkap dengan detail seperti nama unit, deskripsi, harga, serta spesifikasi teknis seperti luas tanah, jumlah kamar, dan fasilitas lainnya. Informasi ini ditampilkan dalam bentuk kartu properti pada halaman utama website.

Tabel 3.3. Struktur Tabel `products`

Nama Kolom	Tipe Data	Keterangan
<code>id</code>	<code>uuid</code>	Primary key, identitas unik unit rumah
<code>name</code>	<code>text</code>	Nama unit rumah
<code>description</code>	<code>text</code>	Deskripsi singkat unit rumah
<code>price</code>	<code>numeric</code>	Harga unit rumah
<code>image_url</code>	<code>text</code>	URL gambar unit rumah
<code>created_at</code>	<code>timestamp</code>	Waktu data ditambahkan
<code>level_lantai</code>	<code>int2</code>	Jumlah lantai rumah
<code>luas_tanah</code>	<code>int2</code>	Luas tanah dalam m ²
<code>lebar_bangunan</code>	<code>int2</code>	Lebar bangunan rumah
<code>jumlah_kamar</code>	<code>int2</code>	Jumlah kamar tidur
<code>kamar_mandi</code>	<code>int2</code>	Jumlah kamar mandi
<code>carport</code>	<code>int2</code>	Jumlah carport/garasi

Tabel 3.4 digunakan untuk menyimpan data yang ditampilkan pada bagian "Tentang Perumahan" di halaman utama website. Informasi ini mencakup gambar

pendukung dan deskripsi singkat mengenai kawasan Perumahan Permata Mutiara Maja.

Tabel 3.4. Struktur Tabel about

Nama Kolom	Tipe Data	Keterangan
id	uuid	Primary key, identitas unik untuk data about
image_url	text	URL gambar pada bagian tentang perumahan
description	text	Deskripsi mengenai perumahan

B Carousel



Gambar 3.1. Carousel Website Perumahan Permata Mutiara Maja

Gambar 3.1 menunjukkan tampilan bagian carousel pada halaman utama website Perumahan Permata Mutiara Maja. Carousel ini digunakan untuk menampilkan beberapa gambar secara bergantian yang merepresentasikan kawasan perumahan, seperti lingkungan sekitar dan unit rumah. Konten gambar diambil dari tabel carousel pada Supabase, sehingga memudahkan pembaruan secara dinamis tanpa perlu mengubah kode secara manual.

```
1 return (
2   <div className="carousel-container">
3     {loading ? (
4       <p>Loading...</p>
5     ) : (
6       <>
```

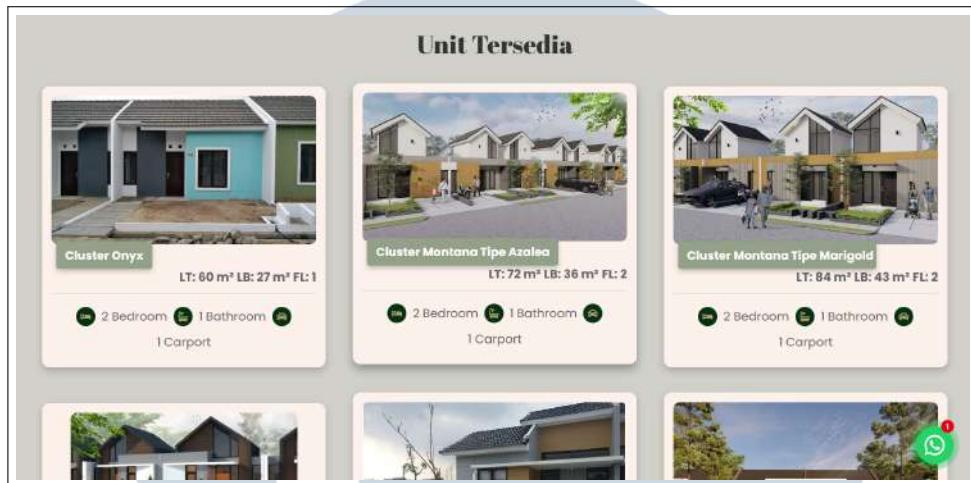
```
7      <div className="carousel-slide">
8          <img src={images[currentIndex]} alt={'Slide ${currentIndex + 1}'} className="carousel-image" />
9      </div>
10     <button className="carousel-button prev" onClick={prevSlide}>
11         &#10094;
12     </button>
13     <button className="carousel-button next" onClick={nextSlide}>
14         &#10095;
15     </button>
16     </>
17   ) }
18 </div>
19 ) ;
```

Kode 3.1: Carousel

Kode 3.1 merupakan potongan kode React yang digunakan untuk menampilkan fitur carousel pada *website* Perumahan Permata Mutiara Maja. Fitur ini memungkinkan pengguna melihat gambar properti secara bergantian. Komponen akan menampilkan pesan “Loading...” saat gambar masih dimuat. Setelah gambar tersedia, komponen akan menampilkan satu gambar dari array berdasarkan indeks saat ini (`currentIndex`), beserta dua tombol navigasi untuk berpindah ke gambar sebelumnya atau berikutnya. Setiap gambar diberi atribut `alt` yang menjelaskan urutan slide, dan elemen HTML diberi kelas untuk keperluan *styling*. Dengan implementasi ini, tampilan *website* menjadi lebih interaktif dan menarik bagi pengguna.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

C Unit Tersedia di Perumahan Permata Mutiara Maja



Gambar 3.2. Unit Tersedia di Perumahan Permata Mutiara Maja

Gambar 3.2 menampilkan bagian Unit Tersedia pada halaman utama website Perumahan Permata Mutiara Maja. Setiap unit ditampilkan dalam bentuk kartu (*card*) yang memuat informasi singkat seperti gambar unit, harga, dan status ketersediaan (tersedia, terjual, atau ditahan). Pengunjung dapat mengklik salah satu kartu untuk diarahkan ke halaman detail unit, yang menyajikan informasi lebih lengkap mengenai spesifikasi, denah, serta galeri foto dari unit tersebut. Seluruh data unit properti ini dikelola melalui tabel *products* di Supabase.

```
1 return (
2   <section className="product-section">
3     <h2 className="unit-tersedia abril-fatface-regular">Unit
4       Tersedia</h2>
5     <div className="product-container">
6       {loading ? (
7         <p>Loading products...</p>
8       ) : products.length > 0 ? (
9         products.map((product) => (
10           <div
11             key={product.id}
12             className="product-card"
13             onClick={() => handleCardClick(product.id)}
14           >
15             <div className="product-image-container">
16               <img
```

```

16         src={product.image_urls.length > 0 ? product.
17         image_urls[0].trim() : "placeholder.jpg"}
18         alt={product.name}
19         className="product-image"
20     />
21     <div className="product-header">
22       <h3 className="product-name">{product.name}</h3>
23     </div>
24     <div className="product-info">
25       <p className="product-dimensions">
26         <span>LT: {product.luas_tanah} m </span>
27         <span>LB: {product.lebar_bangunan} m </span>
28         <span>FL: {product.level_lantai}</span>
29       </p>
30       <p className="product-facilities">
31         <span className="facility-icon"><FaBed /></span>
32       > {product.jumlah_kamar} Bedroom
33         <span className="facility-icon"><FaBath /></span>
34       > {product.kamar_mandi} Bathroom
35         <span className="facility-icon"><FaCar /></span>
36       > {product.carport} Carport
37       </p>
38     </div>
39   </div>
40   )
41   ) : (
42     <p>Produk tidak tersedia.</p>
43   )
44 </div>
45 </section>
46 );

```

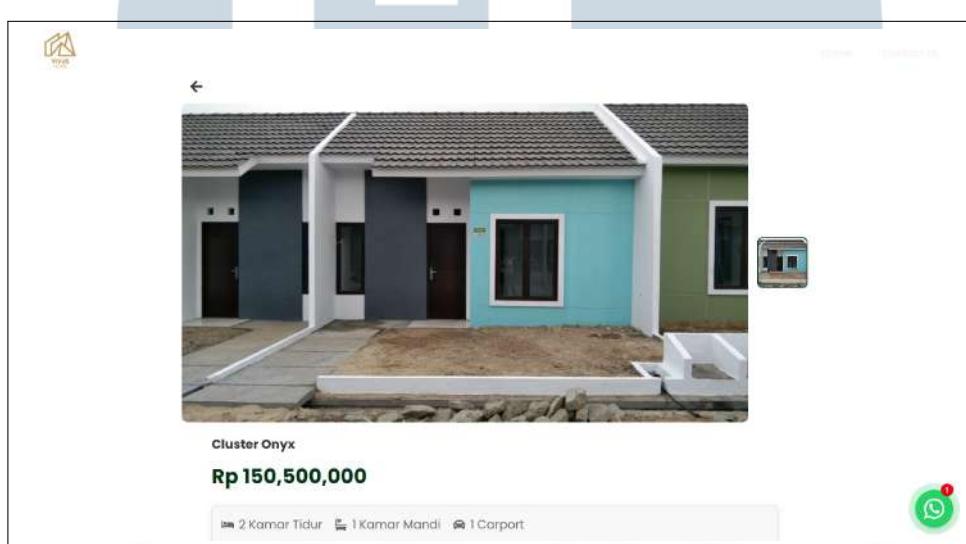
Kode 3.2: Unit Tersedia di Perumahan Permata Mutiara Maja

Kode 3.2 menampilkan kode React untuk bagian Unit Tersedia pada *website* Perumahan Permata Mutiara Maja. Komponen ini bertugas menampilkan daftar unit rumah yang tersedia dalam bentuk kartu. Jika data masih dimuat, maka teks “Loading products...” akan ditampilkan. Setelah data tersedia, komponen akan melakukan iterasi terhadap daftar *products* dan merender informasi unit rumah satu per satu.

Setiap kartu unit berisi gambar properti (jika tersedia), nama unit, dimensi

rumah seperti luas tanah (LT), lebar bangunan (LB), dan jumlah lantai (FL). Selain itu, disediakan juga informasi fasilitas seperti jumlah kamar tidur, kamar mandi, dan carport yang ditampilkan dengan ikon dari pustaka FontAwesome. Apabila gambar tidak tersedia, akan digantikan oleh gambar placeholder. Ketika pengguna mengklik salah satu kartu, fungsi `handleCardClick` akan dipanggil dengan ID unit tersebut, yang biasanya digunakan untuk navigasi ke halaman detail unit.

D Halaman Detail Unit Perumahan Permata Mutiara Maja



Gambar 3.3. Halaman Detail Unit Perumahan Permata Mutiara Maja

Gambar 3.3 menampilkan tampilan Halaman Detail Unit yang ditampilkan setelah pengguna mengklik salah satu unit di halaman Unit Tersedia. Pada halaman ini, pengguna dapat melihat beberapa foto properti dalam bentuk galeri yang dapat digeser atau diperbesar untuk melihat detail visual unit. Selain galeri gambar, terdapat pula informasi penting lainnya seperti deskripsi unit, luas bangunan, luas tanah, jumlah kamar, harga, dan status ketersediaan. Tujuan dari halaman ini adalah memberikan gambaran menyeluruh kepada calon pembeli mengenai spesifikasi dan visual unit rumah yang ditawarkan. Seluruh data pada halaman ini juga diambil secara dinamis dari tabel `products` di Supabase.

```
1 return (
2   <div className="product-detail">
3     <div className="image-section">
4       <div className="image-container">
```

```

5           <button className="back-button" onClick={() => navigate
6             (-1)}>
7               <FaArrowLeft />
8           </button>
9           <img src={product.image_urls[currentImageIndex]} alt={product.name} />
10          </div>
11
12          {/* Thumbnail Gambar */}
13          <div className="thumbnail-container">
14            {product.image_urls.map((img, index) => (
15              <img
16                key={index}
17                src={img}
18                alt={'Thumbnail ${index}'}
19                className={'thumbnail ${currentImageIndex === index
20                  ? "active" : ""}'}
21                onClick={() => setCurrentImageIndex(index)}
22              />
23            )));
24          </div>
25        </div>
26
27        <div className="product-info">
28          <h2 className="product-title">{product.name}</h2>
29          <p className="product-price">Rp {product.price.
30            toLocaleString()}</p>
31          <div className="product-specs">
32            <p><FaBed /> {product.jumlah_kamar} Kamar Tidur</p>
33            <p><FaBath /> {product.kamar_mandi} Kamar Mandi</p>
34            <p><FaCar /> {product.carport} Carport</p>
35          </div>
36          <p className="product-description">
37            {product.description.split("\n").map((line, index) => (
38              <React.Fragment key={index}>
39                {line}
40                <br />
41              </React.Fragment>
42            )));
43          </p>
44        </div>
45
46        {/* Bagian Unit Lainnya */}

```

```

44     <div className="other-units">
45       <h3 className="other-units-title abril-fatface-regular">
46         Unit Lainnya</h3>
47       <div className="other-units-container">
48         {otherUnits.map((unit) => (
49           <div key={unit.id} className="other-unit-card"
50             onClick={() => navigate('/product/${unit.id}')}>
51               <img src={unit.image_url.split(",")[0]} alt={unit.
52                 name} />
53               <div className="other-unit-info">
54                 <h4>{unit.name}</h4>
55                 <p className="other-unit-price">Rp {unit.price.
56                   toLocaleString()}</p>
57               </div>
58             </div>
59           ) );

```

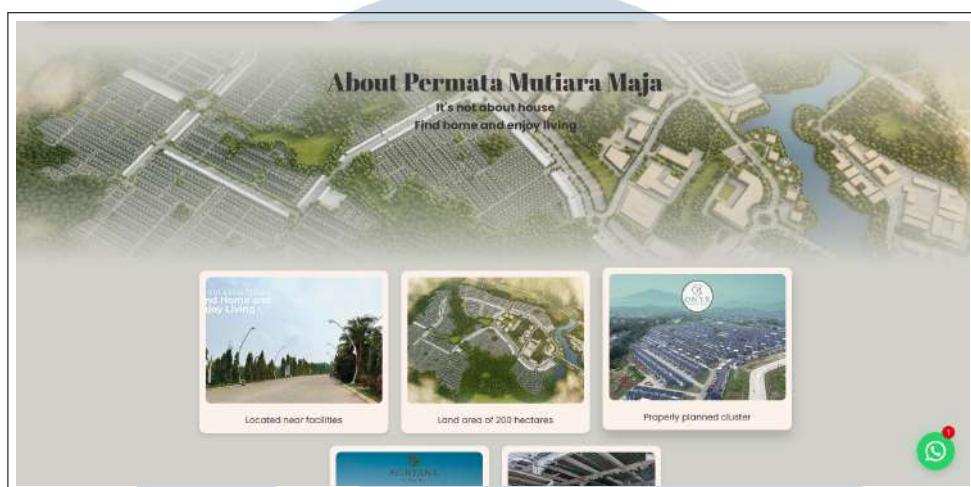
Kode 3.3: Halaman Detail Unit Perumahan Permata Mutiara Maja

Kode 3.3 menunjukkan kode untuk halaman detail unit perumahan Permata Mutiara Maja. Komponen ini bertugas menampilkan informasi lengkap dari satu unit yang dipilih.

Pada bagian atas terdapat tombol kembali yang menggunakan fungsi `navigate(-1)` untuk kembali ke halaman sebelumnya. Gambar utama unit diambil dari `product.image_urls[currentImageIndex]`, dan pengguna dapat memilih gambar lain melalui barisan thumbnail di bawahnya, yang akan mengganti gambar utama saat diklik.

Selanjutnya, bagian `product-info` menampilkan judul unit, harga dalam format lokal (`toLocaleString()`), serta spesifikasi seperti jumlah kamar tidur, kamar mandi, dan carport menggunakan ikon dari FontAwesome.

E Tentang Perumahan Permata Mutiara Maja



Gambar 3.4. Tentang Perumahan Permata Mutiara Maja

Gambar 3.4 menampilkan bagian Tentang pada halaman utama website Perumahan Permata Mutiara Maja. Bagian ini berisi moto perumahan yang menggambarkan nilai dan visi kawasan, serta dilengkapi beberapa gambar pendukung yang merepresentasikan suasana lingkungan sekitar. Setiap gambar disertai dengan deskripsi singkat untuk memperkuat narasi visual dan memberikan gambaran menyeluruh kepada pengunjung mengenai kualitas dan daya tarik perumahan tersebut.

Konten pada bagian ini dikelola melalui tabel `about` dalam Supabase, yang menyimpan informasi berupa URL gambar dan teks deskripsi yang ditampilkan secara dinamis di website.

```
1 return (
2   <section className="about-section">
3     /* Bagian Header + Gambar dengan Gradien */
4     <div className="about-image-wrapper">
5       <div className="about-text">
6         <h2 className="abril-fatface-regular">About Permata
Mutiara Maja</h2>
7         <h3>It's not about house</h3>
8         <h3>Find home and enjoy living</h3>
9       </div>
10      
11    </div>
```

```

12
13     /* Bagian Galeri dengan 3 Foto di Atas, 2 di Tengah */
14 <div className="about-gallery">
15     {aboutImages.length > 0 ? (
16         <>
17             <div className="row1">
18                 {aboutImages.slice(0, 3).map((item, index) => (
19                     <div key={item.id} className="about-card hidden"
ref={(el) => (aboutCardsRef.current[index] = el)}>
20                         <img src={item.image_url} alt="About" className
="about-card-image" />
21                         <p className="about-description">{item.
description}</p>
22                     </div>
23                 )));
24             </div>
25             <div className="row2">
26                 {aboutImages.slice(3, 5).map((item, index) => (
27                     <div key={item.id} className="about-card hidden"
ref={(el) => (aboutCardsRef.current[index + 3] = el)}>
28                         <img src={item.image_url} alt="About" className
="about-card-image" />
29                         <p className="about-description">{item.
description}</p>
30                     </div>
31                 )));
32             </div>
33         </>
34     ) : (
35         <p className="loading-text">Loading images...</p>
36     )
37     </div>
38 </section>
39 );

```

Kode 3.4: Tentang Perumahan Permata Mutiara Maja

Kode 3.4 menunjukkan kode untuk bagian “About” pada website Permata Mutiara Maja. Data gambar dan deskripsi pada galeri ini diambil dari Supabase, yaitu layanan backend yang digunakan sebagai database dan penyimpanan gambar. Gambar disimpan di Supabase Storage, lalu URL dan deskripsi tiap gambar diambil dan disimpan dalam array aboutImages. Setelah berhasil dimuat, gambar ditampilkan dalam dua baris layout galeri. Jika data belum tersedia, akan muncul

pesan “Loading images...”.

F Kelebihan dan Testimonial



Gambar 3.5. Kelebihan dan Testimonial Perumahan Permata Mutiara Maja

Gambar 3.5 menampilkan bagian Kelebihan dan Testimonial dari halaman utama website Perumahan Permata Mutiara Maja. Bagian kelebihan berisi poin-poin utama yang menjadi nilai jual kawasan perumahan, seperti lokasi strategis, fasilitas lengkap, lingkungan asri, dan lingkungan yang berkembang. Penyajian poin ini bertujuan untuk meyakinkan calon pembeli melalui informasi yang ringkas namun informatif.

Sementara itu, bagian testimonial menampilkan ulasan singkat dari konsumen atau penghuni yang telah membeli unit di perumahan tersebut. Testimoni disampaikan dalam bentuk kutipan yang memperkuat citra positif dan membangun kepercayaan pengunjung terhadap kualitas layanan dan produk yang ditawarkan. Seluruh bagian ini didesain secara visual menarik untuk mempertahankan perhatian pengguna dan meningkatkan kepercayaan terhadap brand.

```
1 return (
2   <section className="kelebihan-container">
3     {kelebihanList.map((item) => (
4       <div key={item.id} className="kelebihan-item">
5         <div className="icon">{item.icon}</div>
6         <p className="kelebihan-title">{item.title}</p>
7         <p className="kelebihan-desc">{item.desc}</p>
8       </div>
9     )));
10   </section>
```

```
11 );
```

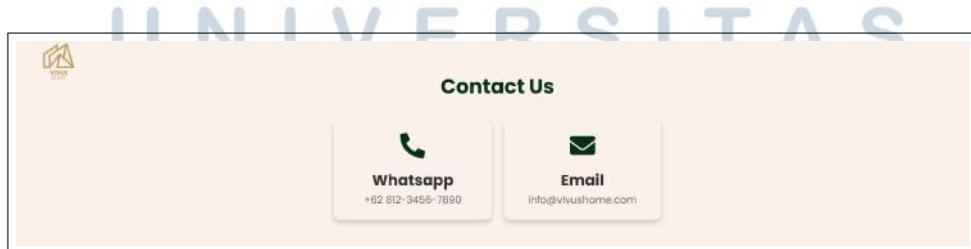
Kode 3.5: Kelebihan Perumahan Permata Mutiara Maja

```
1 return (
2   <section className="testimonial-section">
3     <h2 className="testimonial-title">Testimonial</h2>
4     <div className="testimonial-container">
5       {testimonials.map((testimonial, index) => (
6         <div className="testimonial-box" key={index}>
7           <FaUserCircle className="testimonial-icon" />
8           <div className="testimonial-content">
9             <FaQuoteLeft className="quote-icon" />
10            <p className="testimonial-text">{testimonial.text}</p>
11            <p className="testimonial-name">- {testimonial.name}</p>
12          </div>
13        </div>
14      )));
15    </div>
16  </section>
17);
```

Kode 3.6: Testimonial Perumahan Permata Mutiara Maja

Kode 3.5 dan 3.6 menampilkan komponen statis yang menampilkan daftar kelebihan dan testimoni Perumahan Permata Mutiara Maja. Data disusun dalam array JavaScript dan ditampilkan menggunakan `.map()`. Ikon diambil dari pustaka react-icons. Kode ini tidak terhubung dengan Supabase.

G Contact Us



Gambar 3.6. Contact Us

Gambar 3.6 menunjukkan tampilan bagian *Contact Us* pada website yang berisi informasi kontak utama berupa alamat *email* dan nomor telepon Vivus Home.

Bagian ini bertujuan untuk memudahkan pengunjung dalam menghubungi pihak pengembang apabila membutuhkan informasi lebih lanjut mengenai unit properti yang ditawarkan.

Desainnya dibuat sederhana dan jelas agar informasi kontak dapat diakses dengan cepat oleh pengguna. Penempatan bagian ini berada di bagian bawah halaman sebagai penutup dari keseluruhan tampilan utama website.

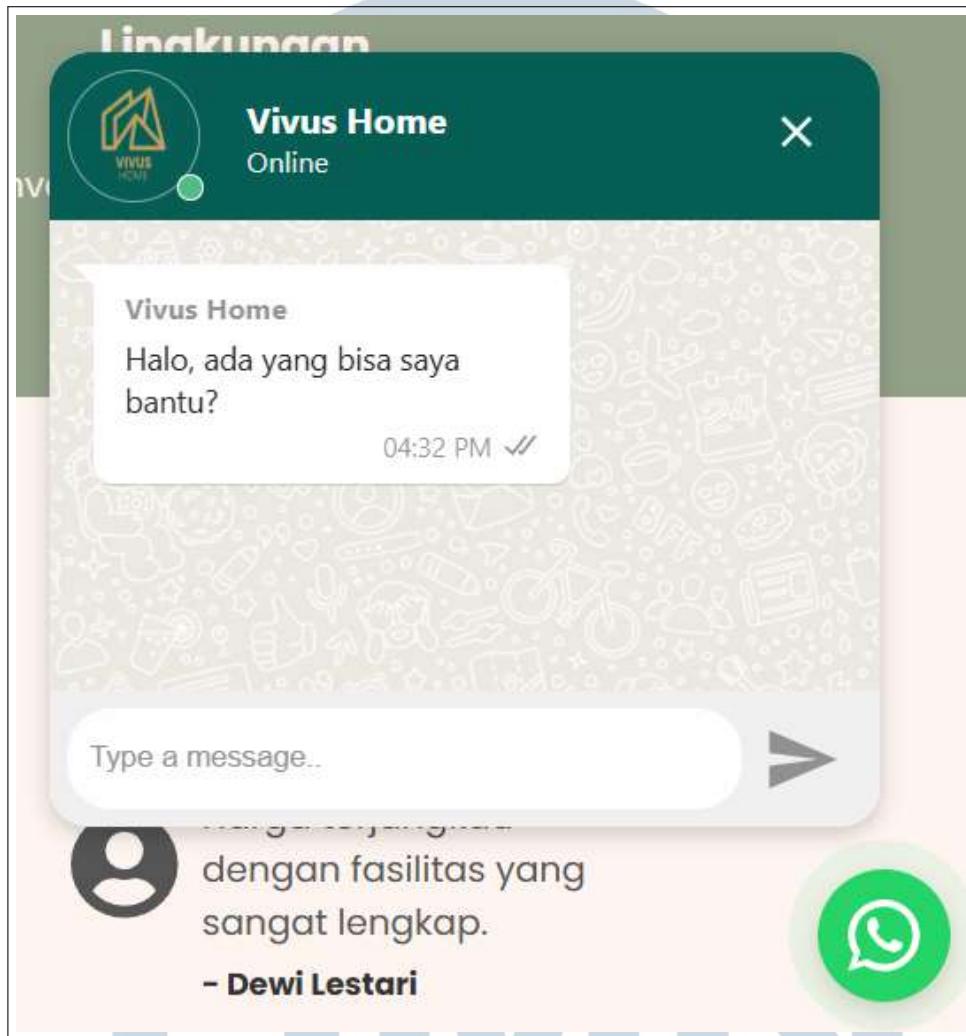
```
1 return (
2     <div className="contact-container">
3         <h1 className="contact-title">Contact Us</h1>
4         <div className="contact-cards">
5             <div className="contact-card">
6                 <FaPhoneAlt className="contact-icon" />
7                 <h2>Whatsapp</h2>
8                 <p>+62 812-3456-7890</p>
9             </div>
10            <div className="contact-card">
11                <FaEnvelope className="contact-icon" />
12                <h2>Email</h2>
13                <p>info@vivushome.com</p>
14            </div>
15        </div>
16    </div>
17);
```

Kode 3.7: Contact Us

Kode 3.7 menampilkan komponen ContactUs yang dibuat dengan React. Komponen ini menampilkan dua informasi kontak—WhatsApp dan email—masing-masing dalam bentuk kartu yang berisi ikon, judul, dan detail kontak. Ikon diambil dari pustaka react-icons.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

H Floating Chat WhatsApp



Gambar 3.7. Tombol Floating Chat Whatsapp

Gambar 3.7 menunjukkan tampilan tombol *floating chat* WhatsApp yang ditambahkan pada halaman utama untuk memudahkan pengunjung website dalam menghubungi pihak pengelola secara langsung melalui WhatsApp. Tombol ini melayang di sudut kanan bawah layar, sehingga selalu terlihat dan dapat diakses kapan saja tanpa mengganggu konten utama.

Ketika pengguna mengirimkan pesan, pengguna akan diarahkan secara otomatis ke aplikasi WhatsApp atau halaman web WhatsApp dengan nomor tujuan yang telah ditentukan. Fitur ini meningkatkan kenyamanan interaksi serta mempercepat proses komunikasi antara calon pembeli dan pengelola properti.

```

1 <div className="whatsapp-container">
2   <FloatingWhatsApp
3     phoneNumber={phoneNumber}
4     accountName="Vivus Home"
5     avatar="/assets/fulllogo.png"
6     chatMessage="Halo, ada yang bisa saya bantu?"
7     statusMessage="Online"
8     chatboxHeight={350} // Perbesar chatbox agar lebih
9       nyaman
10      allowClickAway={true} // Bisa ditutup dengan klik
11      di luar
12      className="custom-whatsapp-widget"
13    />
14  </div>

```

Kode 3.8: Floating Chat Whatsapp

Kode 3.8 memperlihatkan kode implementasi tombol floating chat WhatsApp pada website. Komponen FloatingWhatsApp digunakan untuk menampilkan widget yang mengarah langsung ke nomor WhatsApp tertentu. Properti seperti phoneNumber, accountName, avatar, dan chatMessage disesuaikan untuk mencerminkan identitas bisnis (Vivus Home). Fitur allowClickAway memungkinkan pengguna menutup chat dengan mengklik area luar, sedangkan chatboxHeight mengatur tinggi jendela chat agar lebih nyaman digunakan.

3.4.2 Perancangan Website Property Listing Apartemen Aerium Residence

Website property listing ini dikembangkan untuk menampilkan unit-unit apartemen yang tersedia di Aerium Residence. Informasi disajikan secara terstruktur dan menarik guna mendukung promosi digital properti.

Halaman utama (*homepage*) terdiri dari beberapa bagian utama, yaitu carousel, moto Aerium, kartu unit (*property cards*), fasilitas, lokasi, integrasi video YouTube, dan footer. Seluruh elemen dirancang dengan antarmuka yang responsif dan mudah digunakan. Pengelolaan data dilakukan melalui Supabase untuk memastikan setiap konten dapat diperbarui secara dinamis tanpa mengubah kode secara manual.

Di bawah ini ditampilkan dokumentasi visual dari masing-masing bagian halaman utama. Setiap gambar disertai dengan penjelasan untuk menggambarkan fungsi dan perannya dalam keseluruhan tampilan website. Selain itu, disertakan

juga cuplikan kode yang digunakan untuk membangun bagian-bagian tersebut, guna memberikan gambaran teknis mengenai implementasinya.

A Struktur Database Website Apartemen Aerium Residence

Untuk mendukung fungsionalitas website property listing Apartemen Aerium Residence, digunakan beberapa tabel dalam database yang dikelola melalui Supabase. Struktur database ini dirancang untuk menyimpan data gambar carousel utama, informasi masing-masing unit apartemen, serta daftar fasilitas yang tersedia. Adapun empat tabel utama yang digunakan dalam sistem ini adalah tabel `users`, `apart`, `apart_images`, dan `facilities`.

Tabel 3.5 berikut menunjukkan struktur tabel `users` yang digunakan untuk menyimpan data pengguna yang telah terautentikasi dalam sistem. Tabel ini merupakan bagian dari sistem manajemen autentikasi bawaan Supabase, yang secara otomatis menyimpan informasi penting terkait identitas pengguna. Data dalam tabel ini digunakan untuk keperluan otorisasi akses, seperti membedakan antara pengguna umum dan administrator dalam dashboard admin. Informasi yang tercatat meliputi ID pengguna, peran, alamat email, hingga kata sandi yang telah dienkripsi untuk menjaga keamanan sistem. Struktur ini memungkinkan penerapan autentikasi berbasis peran (role-based authentication) dalam pengelolaan konten website.

Tabel 3.5. Struktur Tabel `users`

Nama Kolom	Tipe Data	Keterangan
<code>instance_id</code>	<code>uuid</code>	ID instansi, digunakan untuk identifikasi multi-tenant
<code>id</code>	<code>uuid</code>	Primary key, identitas unik pengguna
<code>aud</code>	<code>varchar</code>	Audience (pengguna aplikasi), nilai ini tergantung konfigurasi autentikasi
<code>role</code>	<code>varchar</code>	Peran pengguna dalam sistem
<code>email</code>	<code>varchar</code>	Alamat email pengguna
<code>encrypted_password</code>	<code>varchar</code>	Kata sandi pengguna yang telah dienkripsi

Tabel 3.6 berikut menunjukkan struktur tabel `apart` yang digunakan untuk menyimpan gambar-gambar utama yang ditampilkan pada bagian carousel halaman utama website Apartemen Aerium Residence. Tabel ini menyimpan URL gambar serta informasi waktu unggahan untuk keperluan pengelolaan konten secara dinamis.

Tabel 3.6. Struktur Tabel apart

Nama Kolom	Tipe Data	Keterangan
id	int8	Primary key, identitas unik setiap gambar
created_at	timestamp with time zone	Waktu data dibuat atau diunggah
image_url	varchar	URL gambar carousel utama

Tabel 3.7 berfungsi untuk menyimpan data setiap unit apartemen yang tersedia di Aerium Residence. Setiap entri dalam tabel ini mencakup informasi yang cukup lengkap mengenai unit, seperti nama unit (name), ukuran dalam meter persegi (ukuran), jumlah kamar tidur (bedroom), jumlah kamar mandi (bathroom), serta jumlah kamar pembantu jika ada (maidroom). Selain itu, tabel ini juga menyimpan gambar unit melalui kolom image_url, harga unit (harga), dan deskripsi tambahan mengenai unit (desc) yang dapat digunakan untuk menampilkan informasi penting kepada calon pembeli. Kolom created_at merekam waktu unggah data, sementara kolom id menjadi identitas unik dari setiap unit. Data pada tabel ini digunakan untuk menampilkan bagian *property cards* pada halaman utama, serta mendukung tampilan halaman detail ketika pengguna mengakses informasi lebih lanjut mengenai suatu unit.

Tabel 3.7. Struktur Tabel apart_images

Nama Kolom	Tipe Data	Keterangan
id	int8	Primary key, identitas unik unit apartemen
created_at	timestamp with time zone	Waktu data dibuat atau diunggah
name	varchar	Nama atau tipe unit apartemen
ukuran	varchar	Luas unit dalam m ²
bedroom	int4	Jumlah kamar tidur
bathroom	int4	Jumlah kamar mandi
maidroom	int4	Jumlah kamar pembantu
image_url	varchar	URL gambar unit apartemen
harga	varchar	Harga unit apartemen
desc	text	Deskripsi tambahan mengenai unit

Tabel 3.8 berikut merupakan struktur tabel facilities yang digunakan untuk menyimpan daftar gambar fasilitas yang tersedia pada apartemen. Data ini

digunakan untuk ditampilkan dalam galeri fasilitas pada halaman utama website. Setiap entri berisi URL gambar dan informasi waktu unggahan untuk pengelolaan dinamis melalui dashboard admin Supabase.

Tabel 3.8. Struktur Tabel facilities

Nama Kolom	Tipe Data	Keterangan
id	int8	Primary key, identitas unik setiap fasilitas
created_at	timestamp with time zone	Waktu saat data fasilitas ditambahkan ke database
image_url	varchar	URL gambar fasilitas yang ditampilkan di website

B Carousel



Gambar 3.8. Carousel

Gambar 3.8 menunjukkan komponen carousel yang berada di bagian paling atas halaman utama website Apartemen Aerium Residence. Carousel ini berfungsi sebagai elemen pembuka yang menampilkan gambar-gambar visual menarik dari hunian untuk menarik perhatian pengunjung sejak awal.

Pada bagian atas gambar, terdapat teks promosi seperti slogan atau keunggulan utama apartemen yang ditampilkan secara melayang. Penempatan teks ini bertujuan untuk memberikan informasi singkat dan langsung mengenai nilai jual dari properti, serta memperkuat kesan profesional dan elegan dari tampilan website.

```

1  return (
2    <div className="relative w-full h-[30vh] sm:h-[70vh] md:h-
screen overflow-hidden">
3      <img
4        src={images[currentIndex]}
5        alt="Apartment"
6        className="w-full h-full object-cover transition-opacity
duration-1000"
7      />
8
9      <div className="absolute inset-0 bg-gradient-to-b from-
black/50 via-transparent to-[#F3F2EE]"></div>
10
11     <div className="absolute top-[15%] left-4 sm:left-10 right-
4 sm:right-10 flex justify-between items-start w-[90%] sm:w-
[80%]">
12       <div className="text-white max-w-[60%] sm:max-w-[50%] md:
max-w-[40%]">
13         <h1 className="text-lg font-[Abril_Fatface] sm:text-2xl
md:text-4xl lg:text-5xl tracking-wide drop-shadow-lg leading-
tight font-baskervville">
14           <span className="block sm:inline">AERIUM</span>{ " " }
15           <span className="block sm:inline">RESIDENCE</span>
16         </h1>
17
18         <div className="w-16 sm:w-20 h-1 bg-white my-2 sm:my-4">
19           </div>
20
21         <p className="text-xs sm:text-lg md:text-xl lg:text-2xl
font-light tracking-wide">
22           DISCOVER YOUR <br className="hidden sm:block" /> VERY
23           OWN WAY OF LIVING <br className="hidden sm:block" />
24           HARMONIOUSLY
25         </p>
26       </div>
27
28       <div className="bg-white/20 backdrop-blur-md p-3 sm:p-5
lg:p-6 rounded-lg text-white
29                     max-w-[200px] sm:max-w-[250px] lg:max-w-
[350px] shadow-lg border border-white/30">
30         <h2 className="text-sm sm:text-md lg:text-xl text-red-
500 font-bold">      Promo</h2>

```

```

28         <ul className="text-xs sm:text-sm lg:text-md mt-2 space
29           -y-1">
30             <li>    Free DP 15%*</li>
31             <li>    Diskon Hingga 20%*</li>
32         </ul>
33         <p className="mt-2 text-[10px] sm:text-xs lg:text-sm
34           italic">* Syarat & Ketentuan Berlaku</p>
35     </div>
36   </div>
37
38   <div className="absolute bottom-4 left-1/2 transform -
39     translate-x-1/2 flex gap-2">
40     {images.map((_, index) => (
41       <div
42         key={index}
43         className={`${'w-2 h-2 rounded-full ${
44           index === currentIndex ? "bg-white" : "bg-white/50"
45           }`}>
46       </div>
47     )));
48   </div>
49 </div>
50 );

```

Kode 3.9: Carousel

Kode 3.9 menunjukkan komponen Carousel yang menampilkan gambar latar dari database Supabase tabel apart. Gambar terbaru diambil berdasarkan created_at, lalu ditampilkan sebagai background dengan teks promosi di atasnya. Di sebelah kiri terdapat judul dan slogan, sementara di kanan terdapat kotak promo transparan berisi daftar penawaran menarik seperti diskon dan gratis DP. Desain menggunakan Tailwind CSS dan efek gradien overlay untuk estetika.



C Tentang Apartemen Aerium Residence



Gambar 3.9. Tentang Apartemen Aerium Residence

Gambar 3.9 menampilkan bagian Tentang Aerium Residence pada halaman utama. Pada bagian ini, terdapat moto "Living the Lightness of Life" beserta deskripsi singkat yang menggambarkan Aerium sebagai hunian modern yang menyatu dengan alam, menjadi tempat beristirahat dari hiruk pikuk kota, dan menciptakan momen berharga bersama keluarga.

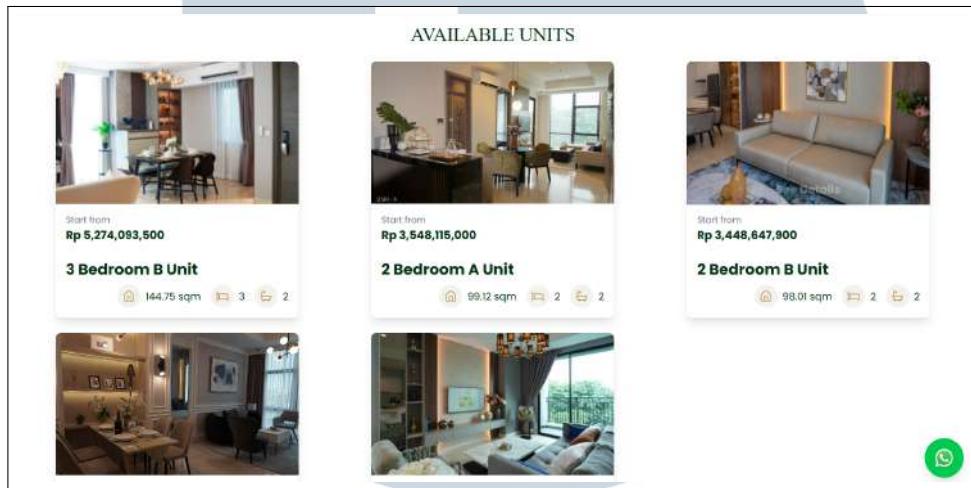
```
1 <div className="w-full md:w-1/2 text-left flex-grow">
2     <h2 className="font-[Baskerville] text-2xl sm:text-3xl
3         lg:text-4xl font-bold text-[#033414] mb-4">
4         Living the Lightness of Life
5     </h2>
6     <p className="text-gray-700 leading-relaxed">
7         Aerium is a one-of-a-kind celebration of life and
8         nature, where the
9             modern sophistication and the beauty of earth
10            seamlessly come together
11            in perfect harmony.
12        </p>
13        <p className="mt-4 text-gray-700 leading-relaxed">
14            This extraordinary retreat from the city's constant
15            hustle and bustle
16            transforms into a sanctuary, a place where you and
17            your loved ones
18            create precious moments that turn into lasting
19            memories.
20        </p>
21    </div>
```

Kode 3.10: Tentang Apartemen Aerium Residence

Kode 3.10 menampilkan komponen About yang menjelaskan konsep hunian Aerium Residence. Bagian kiri menampilkan dua foto unit apartemen yang

ditumpuk dengan efek bayangan untuk kesan elegan. Sementara di kanan, terdapat teks deskriptif mengenai gaya hidup dan ketenangan yang ditawarkan Aerium. Komponen ini dibuat dengan Next.js dan Tailwind CSS, serta memanfaatkan fitur Image dari Next.js untuk optimasi gambar.

D Unit Tersedia di Apartemen Aerium Residence



Gambar 3.10. Unit Tersedia di Apartemen Aerium Residence

Gambar 3.10 menampilkan daftar unit apartemen yang tersedia di Aerium Residence dalam bentuk kartu informasi. Setiap kartu memuat gambar unit, tipe, harga, dan fitur utama seperti jumlah kamar dan luas bangunan. Pengunjung dapat mengklik salah satu kartu untuk diarahkan ke halaman detail unit yang memuat informasi lebih lengkap. Data pada bagian ini diambil secara dinamis dari tabel `apart_images` di Supabase dan dapat dikelola melalui dashboard admin tanpa perlu mengubah kode secara langsung.

```
1 return (
2     <div className="container mx-auto px-6 py-12">
3         <h2 className="font-[Abril_Fatface] uppercase text-3xl text-[#033414] mb-6 text-center">
4             Available Units
5         </h2>
6
7         <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-6 place-items-center">
8             {units.map((unit, index) => (
9                 <motion.div
```

```

10         key={unit.id}
11         onClick={() => handleClick(unit.id)}
12         className="relative bg-white shadow-lg rounded-lg
13         overflow-hidden w-full max-w-sm cursor-pointer group
14         transition-transform duration-300"
15         initial={{ opacity: 0, scale: 0.9 }}
16         whileInView={{ opacity: 1, scale: 1 }}
17         viewport={{ once: false, amount: 0.2 }}
18         transition={{ duration: 0.5, delay: index * 0.1 }}
19     >
20     <img
21         src={unit.image_url[0]}
22         alt={unit.name}
23         className="w-full h-56 object-cover transition-
24         transform duration-300 group-hover:scale-105"
25     />
26
27     <div className="absolute inset-0 group-hover:bg-
28         opacity-40 transition-all duration-300 flex items-center
29         justify-center">
30         <p className="text-white text-lg font-bold opacity
31         -0 group-hover:opacity-100 transition-opacity duration-300">
32             See Details
33         </p>
34     </div>
35
36     <div className="p-4">
37         <p className="text-sm text-gray-500">Start from</p>
38         <h2 className="text-lg font-bold text-[#033414]">
39             Rp {unit.harga.toLocaleString()}
40         </h2>
41
42         <div className="mt-6 text-left w-full max-w-[800px]
43         ">
44             <h1 className="text-[#033414] text-2xl font-bold">
45                 {unit.name}</h1>
46                 <p className="text-gray-500 mt-2">{unit.desc}</p>
47             </div>
48
49             <div className="flex flex-wrap items-center gap-x-4
50             gap-y-2 text-gray-600 mt-2 justify-end">
51                 <div className="flex items-center gap-2">
52                     <div className="bg-[#f3f2ee] p-2 rounded-full">
```

```

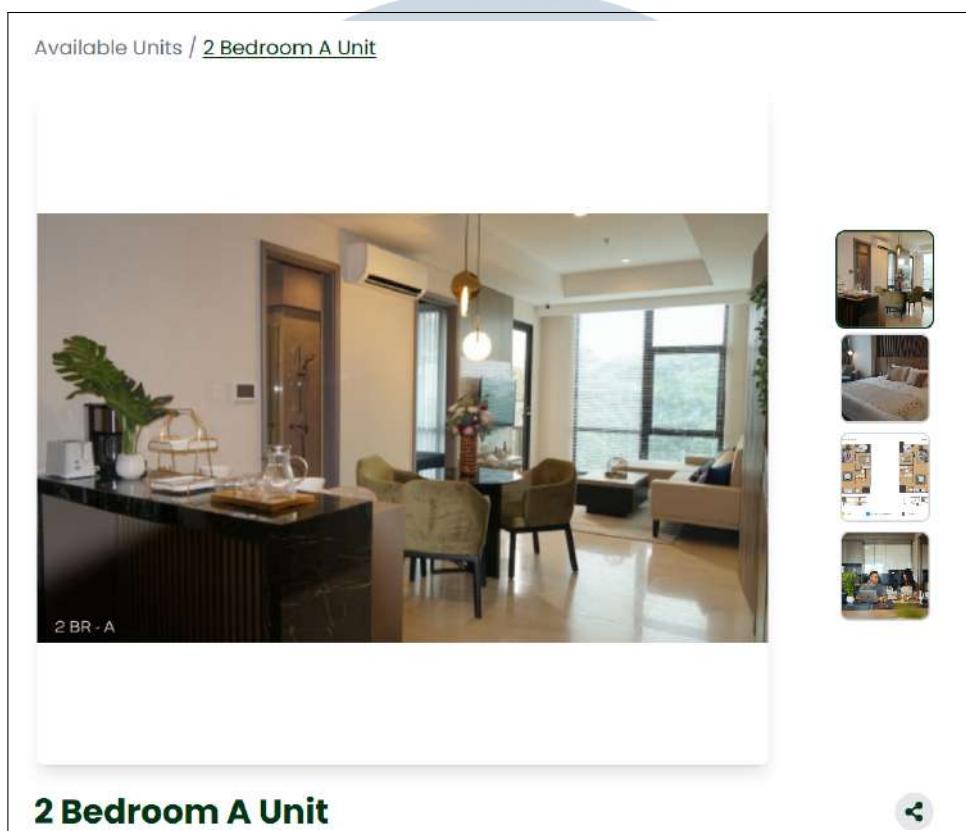
44         <Home size={20} className="text-[#bfa162]" />
45     </div>
46     <span className="text-[#033414]">{unit.ukuran}
47     sqm</span>
48     </div>
49     <div className="flex items-center gap-2">
50       <div className="bg-[#f3f2ee] p-2 rounded-full">
51         <Bed size={20} className="text-[#bfa162]" />
52       </div>
53       <span className="text-[#033414]">{unit.bedroom
54     }</span>
55     </div>
56     <div className="flex items-center gap-2">
57       <div className="bg-[#f3f2ee] p-2 rounded-full">
58         <Bath size={20} className="text-[#bfa162]" />
59       </div>
60       <span className="text-[#033414]">{unit.bathroom
61     }</span>
62     </div>
63   </div>
64   </motion.div>
65 ) ) }
66 </div>
67 </div>
68 ) ;

```

Kode 3.11: Unit Tersedia di Apartemen Aerium Residence

Kode 3.11 menunjukkan kode komponen `Unit` yang menampilkan daftar unit apartemen Aerium Residence secara dinamis dari database Supabase (`apart_images`). Setiap kartu unit menampilkan gambar, harga, nama, dan informasi spesifikasi seperti luas, jumlah kamar tidur, dan kamar mandi, lengkap dengan ikon dari pustaka `lucide-react`. Animasi transisi ditambahkan dengan Framer Motion, dan fitur klik pada kartu akan menavigasi pengguna ke halaman detail unit menggunakan Next.js router.

E Halaman Detail Unit Apartemen Aerium Residence



Gambar 3.11. Halaman Detail Unit Apartemen Aerium Residence

Gambar 3.11 menunjukkan tampilan halaman detail untuk salah satu unit apartemen di Aerium Residence. Halaman ini dirancang untuk memberikan informasi lengkap dan menyeluruh mengenai unit yang dipilih oleh pengunjung. Di dalamnya terdapat tampilan gambar utama unit, deskripsi mendetail, spesifikasi teknis seperti jumlah kamar tidur, kamar mandi, luas bangunan, serta informasi tambahan seperti harga.

Tampilan ini juga memuat galeri tambahan yang menampilkan berbagai sudut interior dan eksterior unit, memberikan pengalaman visual yang lebih mendalam kepada calon pembeli. Seluruh informasi yang ditampilkan pada halaman ini diambil dari *database* Supabase, khususnya dari tabel *apart_images*.

```
1 return (
2   <div className="container mx-auto px-6 py-16 flex flex-col items-center">
3     <div className="w-full max-w-[800px] mb-6 mt-16 text-left">
```

```

4      <p className="text-[#033414] text-lg">
5        <span
6          onClick={() => router.back()}
7          className="cursor-pointer text-gray-500 hover:
underline"
8        >
9          Available Units
10         </span>{ " " }
11         <span className="text-gray-500"/></span>{ " " }
12         <span className="underline text-[#033414]">{unit.name
} </span>
13       </p>
14     </div>
15
16     <div className="flex flex-col lg:flex-row items-center
justify-center gap-8">
17       <motion.div
18         key={currentIndex}
19         className="w-full max-w-[650px] h-[300px] sm:h-[400px]
lg:h-[600px] rounded-lg overflow-hidden shadow-lg"
20         initial={{ opacity: 0.8, scale: 0.95 }}
21         animate={{ opacity: 1, scale: 1 }}
22         transition={{ duration: 0.5 }}
23       >
24         <div className="w-full h-full bg-white flex justify-
center items-center">
25           <img
26             src={unit.image_url[currentIndex]}
27             alt={`Image ${currentIndex + 1}`}
28             className="w-full h-full object-contain"
29           />
30         </div>
31       </motion.div>
32
33       <div className="hidden lg:flex flex-col gap-2 max-h-[400
px] ml-8">
34         {unit.image_url.map((img, index) => (
35           <motion.div
36             key={index}
37             className={`${'w-20 h-20 rounded-lg overflow-hidden
border-2 cursor-pointer transition-all ${
currentIndex === index ? "border-[#033414] scale
-110" : "border-gray-300"}`}>

```

```

39         } `}
40         onClick={() => setCurrentIndex(index)}
41         whileHover={{ scale: 1.1 }}
42       >
43         <img
44           src={img}
45           alt={`Thumbnail ${index}`}
46           className="w-full h-full object-cover"
47         />
48       </motion.div>
49     )) }
50   </div>
51 </div>
52
53 <div className="lg:hidden w-full mt-4">
54   <div className="grid grid-cols-3 sm:grid-cols-4 gap-2 px
-2">
55     {unit.image_url.map((img, index) => (
56       <motion.div
57         key={index}
58         className={`${'w-full h-20 rounded-lg overflow-hidden
border-2 cursor-pointer transition-all ${
59           currentIndex === index ? "border-[#033414] scale
-110" : "border-gray-300"
60         }`}
61         onClick={() => setCurrentIndex(index)}
62         whileHover={{ scale: 1.1 }}
63       >
64         <img
65           src={img}
66           alt={`Thumbnail ${index}`}
67           className="w-full h-full object-cover"
68         />
69       </motion.div>
70     )) }
71   </div>
72 </div>
73
74 <div className="mt-6 text-left w-full max-w-[800px] flex
items-center justify-between">
75   {/* Nama Unit */}
76   <h1 className="text-[#033414] text-3xl font-bold">{unit.
name}</h1>

```

```

77
78     /* Button Share */
79     <button
80         onClick={handleShare}
81         className="bg-gray-200 p-2 rounded-full hover:bg-gray
82 -300 transition"
83         >
84             <FaShareAlt className="text-[#033414] text-lg" />
85         </button>
86     </div>
87
88     /* Deskripsi Unit */
89     <div className="mt-4 text-gray-500 max-w-[800px]">
90         <p className="leading-relaxed text-base">{unit.desc}</p>
91     </div>
92
93     <div className="mt-6 bg-gray-200 p-6 rounded-lg w-full max-
94 w-[800px]">
95         <h2 className="text-xl font-semibold mb-4">Unit Details</
96 h2>
97         <div className="flex flex-wrap gap-6">
98             {/* Ukuran */}
99             <div className="flex items-center gap-2">
100                 <div className="bg-[#f3f2ee] p-2 rounded-full">
101                     <FaHome className="text-[#bfa162]" />
102                 </div>
103                 <span className="text-[#033414]">{unit.ukuran} sqm</
104 span>
105             </div>
106             {/* Bedrooms */}
107             <div className="flex items-center gap-2">
108                 <div className="bg-[#f3f2ee] p-2 rounded-full">
109                     <FaBed className="text-[#bfa162]" />
110                 </div>
111                 <span className="text-[#033414]">{unit.bedroom}</span>
112             </div>
113             {/* Bathrooms */}
114             <div className="flex items-center gap-2">
115                 <div className="bg-[#f3f2ee] p-2 rounded-full">
116                     <FaBath className="text-[#bfa162]" />
117                 </div>
118                 <span className="text-[#033414]">{unit.bathroom}</span>
119             </div>

```

```

115     Bathroom</ span>
116     </div>
117     { /* Maidroom */
118       unit.maidroom && (
119         <div className="flex items-center gap-2">
120           <div className="bg-[#f3f2ee] p-2 rounded-full">
121             <FaBroom className="text-[#bfa162]" />
122           <span className="text-[#033414]">{unit.maidroom}
123         Maidroom</ span>
124         </div>
125       ) }
126     </div>

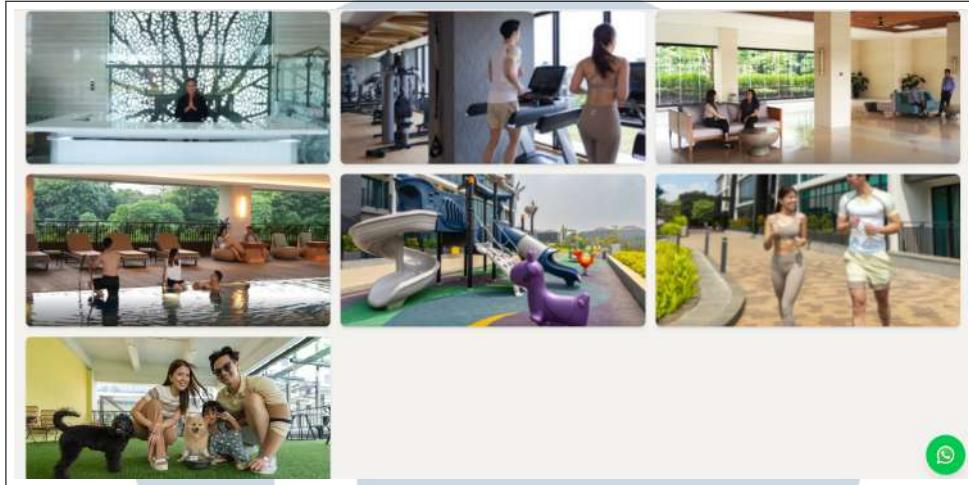
```

Kode 3.12: Halaman Detail Unit Apartemen Aerium Residence

Kode 3.12 menunjukkan kode halaman detail unit Apartemen Aerium Residence yang menampilkan informasi lengkap mengenai satu unit apartemen berdasarkan ID dari URL. Data unit diambil dari tabel apart_images di Supabase dan mencakup gambar, nama, ukuran, jumlah kamar tidur dan kamar mandi, serta ruang pembantu jika tersedia. Halaman ini dilengkapi dengan fitur galeri gambar utama dan thumbnail yang dapat dipilih, breadcrumb navigasi, tombol berbagi menggunakan Web Share API, serta daftar unit lain yang masih tersedia. Desainnya responsif dan interaktif dengan animasi menggunakan Framer Motion, serta mendukung navigasi antar detail unit secara dinamis melalui router.push.



F Fasilitas Apartemen Aerium Residence



Gambar 3.12. Fasilitas Apartemen Aerium Residence

Gambar 3.12 menampilkan bagian Fasilitas Apartemen Aerium Residence yang berada di halaman utama website. Pada bagian ini, pengguna dapat melihat berbagai fasilitas unggulan yang ditawarkan oleh Aerium, seperti area bermain anak, kolam renang, ruang serbaguna, gym, dan fasilitas hijau lainnya.

Fasilitas-fasilitas ini ditampilkan dalam bentuk galeri gambar yang ditata rapi dan responsif, sehingga tetap nyaman diakses baik melalui perangkat desktop maupun mobile. Setiap gambar mewakili satu jenis fasilitas yang tersedia di lingkungan apartemen.

Data visual ini diambil secara dinamis dari storage Supabase dan dikelola melalui tabel facilities. Admin dapat menambahkan atau menghapus gambar fasilitas kapan saja melalui dashboard, tanpa perlu menyentuh kode program, sehingga menjaga fleksibilitas pengelolaan konten.

```
1 return (
2   <div>
3     <h2 className="font-[Abril_Fatface] uppercase text-3xl text-
4       -center text-[#033414] mb-2">
5       Facilities
6     </h2>
7     <div className="grid grid-cols-2 md:grid-cols-3 gap-4">
8       {images.map((item, index) => (
9         <motion.img
10           key={index}
11           src={item.image_url}>
```

```
11     alt={'Facility ${index + 1}'}
12     className="w-full h-48 object-cover rounded shadow"
13     initial={{ opacity: 0, scale: 0.9 }}
14     animate={{ opacity: 1, scale: 1 }}
15     transition={{ duration: 0.4, delay: index * 0.1 }}
16   />
17   )) }
18 </div>
19 </div>
20 );
```

Kode 3.13: Fasilitas Apartemen Aerium Residence

Kode 3.13 menampilkan kode komponen halaman fasilitas Apartemen Aerium Residence yang berfungsi untuk menampilkan ikon-ikon fasilitas beserta galeri gambar fasilitas yang tersimpan di Supabase. Komponen ini menampilkan ikon dari pustaka lucide-react seperti lounge, kolam renang, gym, dan lain-lain secara responsif menggunakan grid. Selain itu, gambar-gambar fasilitas ditampilkan dalam bentuk galeri dinamis yang diambil dari tabel facilities di Supabase.

G Lokasi Apartemen Aerium Residence



Gambar 3.13. Lokasi Apartemen Aerium Residence

Gambar 3.13 menampilkan lokasi strategis Apartemen Aerium Residence yang dikelilingi fasilitas umum seperti sekolah, supermarket, pusat perbelanjaan, dan rumah sakit. Tampilan ini juga dilengkapi dengan gambar peta lokasi yang jika diklik akan mengarahkan pengguna ke Google Maps untuk melihat posisi apartemen secara lebih detail.

```

1 return (
2     <div>
3         <h2 className="font-[Abril_Fatface] uppercase text-3xl text-
4             -center text-[#033414] mb-2">
5             Facilities
6         </h2>
7         <div className="grid grid-cols-2 md:grid-cols-3 gap-4">
8             {images.map((item, index) => (
9                 <motion.img
10                    key={index}
11                    src={item.image_url}
12                    alt={`Facility ${index + 1}`}
13                    className="w-full h-48 object-cover rounded shadow"
14                    initial={{ opacity: 0, scale: 0.9 }}
15                    animate={{ opacity: 1, scale: 1 }}
16                    transition={{ duration: 0.4, delay: index * 0.1 }}
17                />
18            ))})
19         </div>
20     </div>
21 );

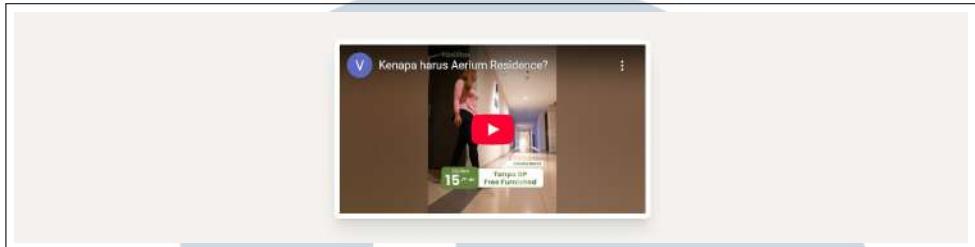
```

Kode 3.14: Lokasi Apartemen Aerium Residence

Kode 3.14 menampilkan potongan kode yang digunakan untuk menampilkan informasi lokasi Apartemen Aerium Residence. Komponen ini menyajikan daftar lokasi penting di sekitar apartemen, seperti institusi pendidikan, rumah sakit, pusat perbelanjaan, dan pasar swalayan. Pengguna dapat memilih kategori lokasi yang ingin ditampilkan, dan daftar tempat akan berubah sesuai pilihan. Selain itu, bagian kanan antarmuka menampilkan gambar peta lokasi dan tombol interaktif "Go to Map" yang mengarahkan pengguna ke Google Maps untuk navigasi lebih lanjut. Desain responsif dan interaktif ini meningkatkan kemudahan pengguna dalam mengetahui aksesibilitas fasilitas publik di sekitar Aerium.

MULTIMEDIA
NUSANTARA

H Video Apartemen Aerium Residence



Gambar 3.14. Video Apartemen Aerium Residence

Gambar 3.14 menunjukkan integrasi video promosi dari YouTube mengenai Apartemen Aerium Residence. Video ini memberikan gambaran visual terkait suasana apartemen, termasuk area luar dan dalam bangunan, untuk mendukung penyampaian informasi secara lebih interaktif kepada calon pembeli.

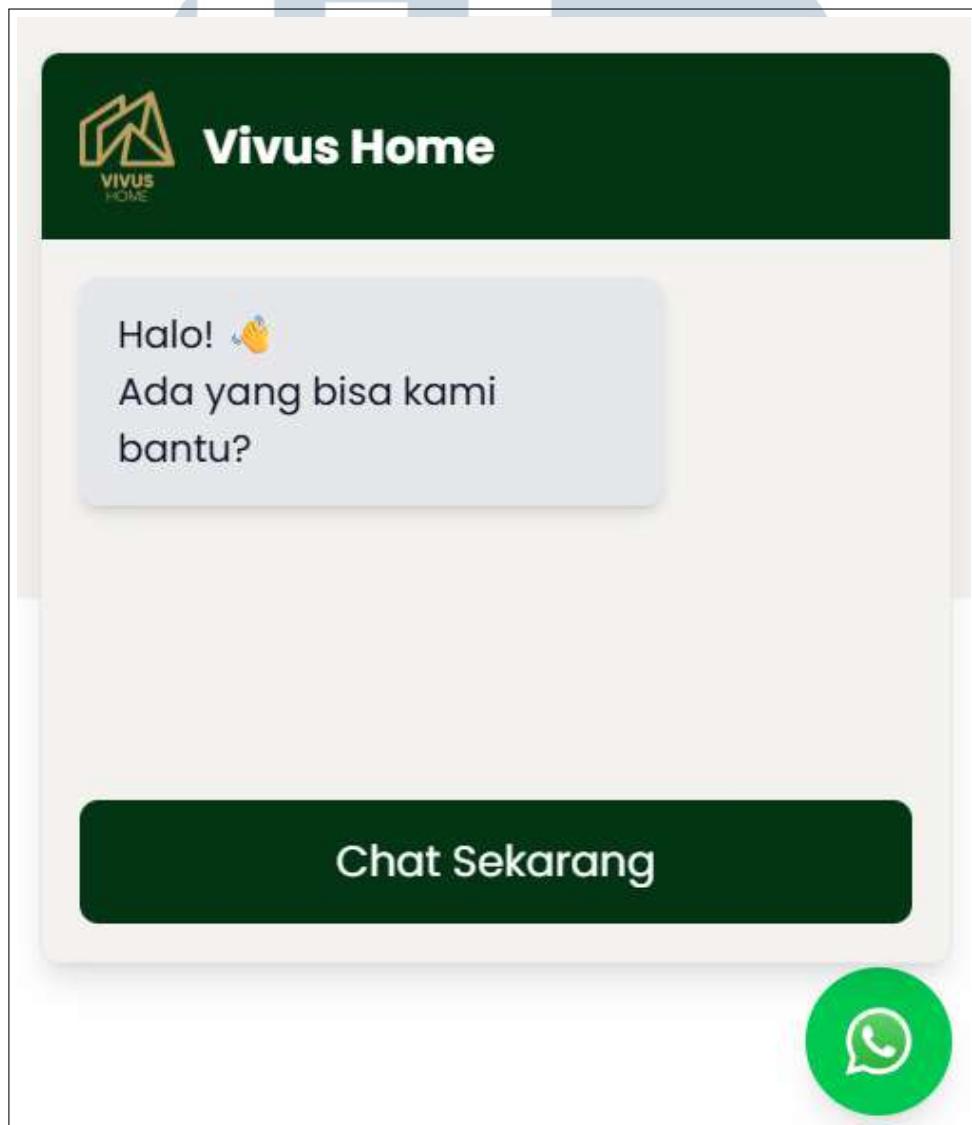
```
1 return (
2     <div className="w-full bg-[#f3f2ee] py-12 flex items-center
3         justify-center">
4         <div className="w-[80%] max-w-[500px] aspect-video bg-white
5             p-2 rounded-lg shadow-xl">
6             <ReactPlayer
7                 url="https://youtube.com shorts/Ijni9lWpE9U?feature=
8                 share"
9                 controls={true}
10                width="100%"
11                height="100%"
12            />
13        </div>
14    </div>
15);
16;
```

Kode 3.15: Video Apartemen Aerium Residence

Kode 3.15 menunjukkan kode komponen yang digunakan untuk menampilkan video promosi Apartemen Aerium Residence. Komponen ini menggunakan *library* react-player secara dinamis untuk menghindari rendering di sisi server (SSR). Video ditampilkan dalam sebuah kotak responsif dengan rasio aspek video, dibungkus dalam tampilan bergaya modern menggunakan Tailwind CSS. Dengan integrasi kontrol pemutar video YouTube secara langsung di halaman, pengguna dapat dengan mudah melihat cuplikan visual apartemen tanpa harus

keluar dari situs, sehingga meningkatkan pengalaman interaktif dan informatif bagi calon pembeli.

I Floating Chat Whatsapp



Gambar 3.15. Floating Chat Whatsapp

Gambar 3.15 menunjukkan fitur *Floating Chat WhatsApp* yang diterapkan pada halaman utama website Apartemen Aerium Residence. Fitur ini berbentuk ikon WhatsApp melayang di pojok kanan bawah layar, yang selalu terlihat meskipun pengguna menggulir halaman.

```

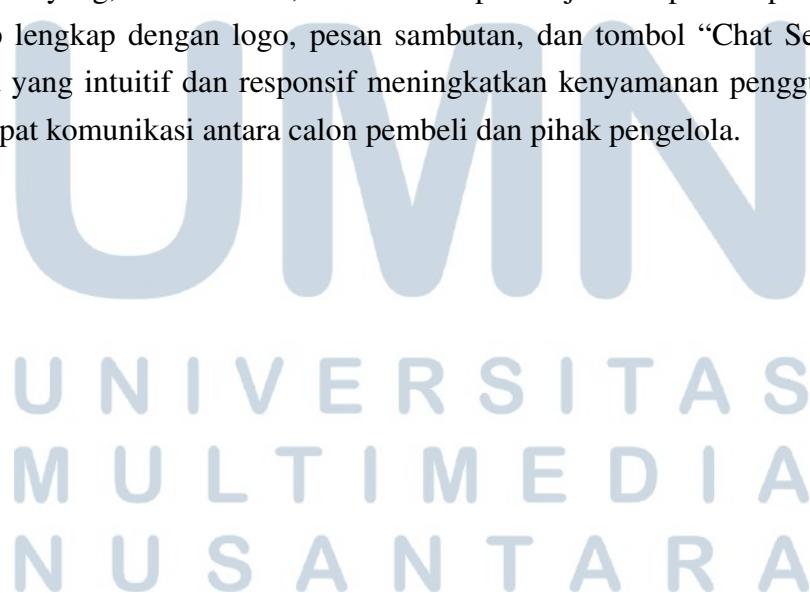
1 | return (
2 |   <div className="fixed bottom-4 right-4 z-50 flex flex-col
3 |     items-end">
4 |   <button
5 |     onClick={() => setIsOpen(!isOpen)}
6 |     className="bg-green-500 text-white p-4 rounded-full
7 |     shadow-lg hover:bg-green-600 transition flex items-center
8 |     justify-center"
9 |     >
10 |       <Image
11 |         src="/assets/whatsapp.svg"
12 |         alt="WhatsApp"
13 |         width={30}
14 |         height={30}
15 |       />
16 |     </button>
17 |
18 |     {isOpen && (
19 |       <div className="absolute bottom-16 right-0 bg-[#f3f2ee]
20 |         shadow-lg rounded-lg w-96 h-96 border border-gray-200 overflow
21 |         hidden animate-fade-in flex flex-col">
22 |         {/* Header */}
23 |         <div className="bg-[#033414] text-white flex items-
24 |           center p-4 rounded-t-lg">
25 |           <Image
26 |             src="/assets/fulllogo_transparent.png"
27 |             alt="Vivus Home Logo"
28 |             width={40}
29 |             height={40}
30 |             className="mr-3"
31 |           />
32 |           <span className="font-semibold text-xl">Vivus Home</
33 |             span>
34 |         </div>
35 |
36 |         <div className="flex-1 p-4 flex items-start">
37 |           <div className="bg-gray-200 text-gray-900 px-4 py-3
38 |             rounded-lg shadow-md max-w-[70%]">
39 |             <p>Halo!</p>
40 |             <p>Ada yang bisa kami bantu?</p>
41 |           </div>
42 |         </div>
43 |       </div>
44 |     )
45 |   )
46 | )

```

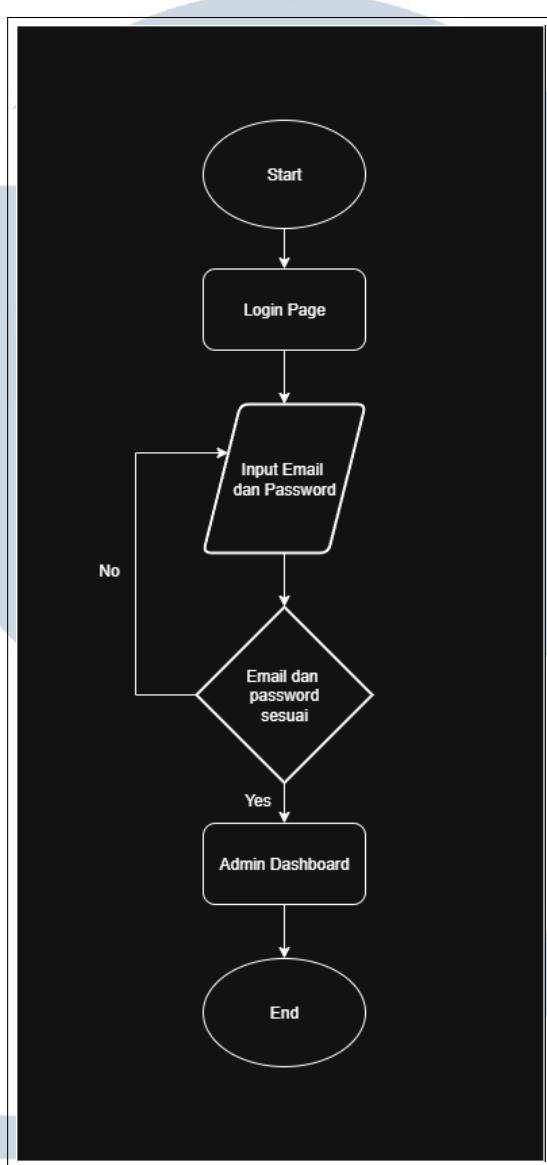
```
36 <div className="p-4">
37   <a
38     href={`https://wa.me/${phoneNumber}?text=${message}`}
39     target="_blank"
40     rel="noopener noreferrer"
41     className="block bg-[#033414] text-white text-
42       center py-3 rounded-lg hover:bg-green-600 transition text-lg"
43     >
44       Chat Sekarang
45     </a>
46   </div>
47 </div>
48 </div>
49 ) ;
```

Kode 3.16: Floating Chat Whatsapp

Kode 3.16 memperlihatkan kode untuk fitur *Floating Chat WhatsApp* yang terintegrasi pada website Apartemen Aerium Residence. Fitur ini memungkinkan pengunjung situs untuk menghubungi pihak pengelola secara langsung melalui WhatsApp dengan sekali klik. Komponen ini memiliki tombol melayang di pojok kanan bawah yang, ketika diklik, akan menampilkan jendela percakapan bergaya WhatsApp lengkap dengan logo, pesan sambutan, dan tombol “Chat Sekarang”. Desainnya yang intuitif dan responsif meningkatkan kenyamanan pengguna serta mempercepat komunikasi antara calon pembeli dan pihak pengelola.



J Halaman Login

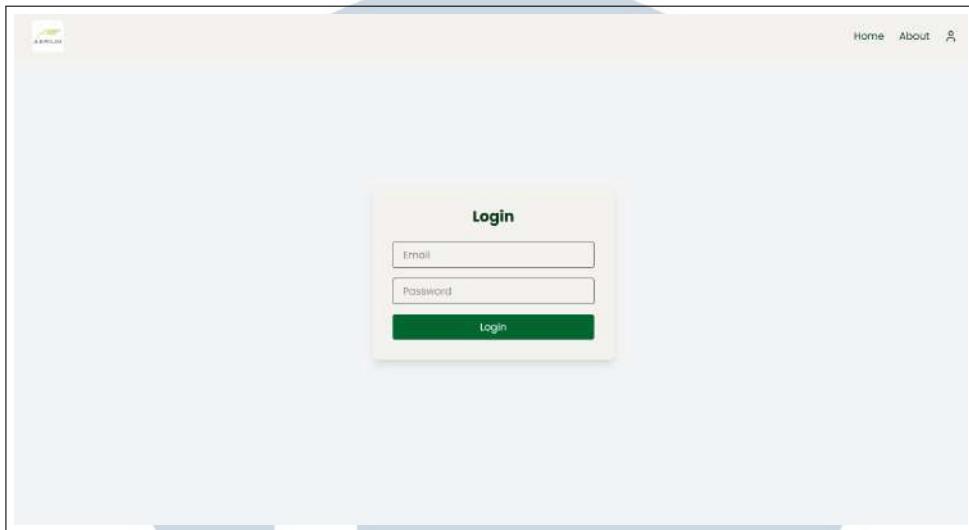


Gambar 3.16. Diagram Alur Sistem Login

Gambar 3.16 menunjukkan alur proses autentikasi *admin* sebelum mengakses *dashboard admin*. Proses dimulai dari pengguna yang diarahkan ke halaman login. Pada halaman ini, pengguna diminta untuk memasukkan *email* dan kata sandi.

Jika *email* dan *password* yang dimasukkan tidak sesuai, sistem akan menolak akses dan pengguna tetap berada di halaman login. Sebaliknya, apabila kredensial yang dimasukkan valid dan cocok dengan data pada tabel *users*, maka pengguna akan diarahkan ke *dashboard admin*. Proses ini berakhir setelah admin

berhasil masuk ke sistem dan mendapatkan hak akses penuh terhadap manajemen konten *website*.



Gambar 3.17. Halaman Login

Gambar 3.17 menampilkan antarmuka halaman *login* yang digunakan oleh *admin* untuk mengakses *dashboard* pengelolaan konten pada *website* Apartemen Aerium Residence. Halaman ini dirancang dengan tampilan yang sederhana dan fokus, menampilkan dua kolom isian utama, yaitu alamat *email* dan kata sandi.

Form *login* ini terintegrasi langsung dengan sistem autentikasi Supabase, di mana akun pengguna telah disimpan dalam tabel *users*. Setelah proses *login* berhasil, pengguna dengan peran *admin* akan diarahkan ke halaman *dashboard* untuk melakukan pengelolaan data seperti gambar *carousel*, unit apartemen, dan fasilitas. Sistem *login* ini juga dilengkapi dengan validasi sisi klien untuk memastikan bahwa kolom *email* dan *password* tidak dikirim dalam keadaan kosong.

```
1 return (
2   <div className="min-h-screen flex items-center justify-center
3     bg-gray-100 p-4">
4     <div className="bg-[#f3f2ee] p-8 rounded-lg shadow-lg w-
5       full max-w-sm">
6       <h1 className="text-2xl text-[#033414] font-semibold text
7         -center mb-6">Login</h1>
8
9       {errorMsg && (
10         <p className="text-red-500 text-sm mb-4 text-center">{
11           errorMsg
12         </p>
13       ) }
14   </div>
15 </div>
```

```

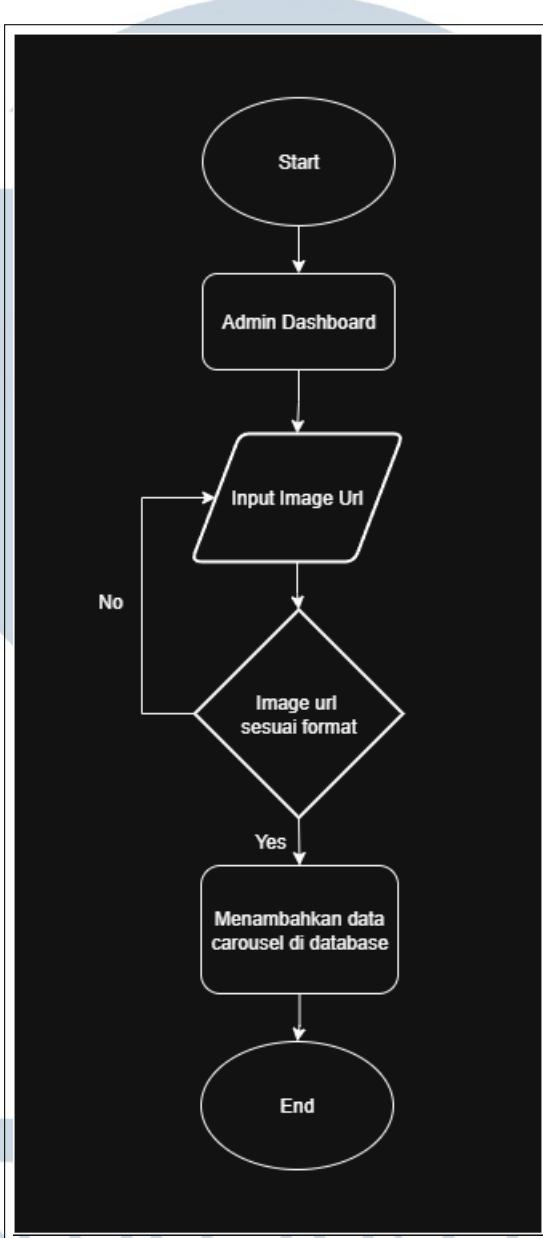
9
10    <form onSubmit={handleLogin} className="space-y-4">
11      <input
12        type="email"
13        placeholder="Email"
14        className="w-full px-4 py-2 border rounded focus:
outline-none"
15        value={email}
16        onChange={(e) => setEmail(e.target.value)}
17        required
18      />
19      <input
20        type="password"
21        placeholder="Password"
22        className="w-full px-4 py-2 border rounded focus:
outline-none"
23        value={password}
24        onChange={(e) => setPassword(e.target.value)}
25        required
26      />
27      <button
28        type="submit"
29        className="w-full bg-green-800 text-white py-2
rounded hover:bg-green-900"
30      >
31        Login
32      </button>
33    </form>
34  </div>
35</div>
36) ;

```

Kode 3.17: Halaman Login

Kode 3.17 menunjukkan cuplikan kode halaman *login admin* pada website Apartemen Aerium Residence. Komponen ini terdiri dari memasukkan *email* dan *password*, serta tombol *submit*. Data dikendalikan melalui *state* lokal menggunakan *setEmail* dan *setPassword*. Form akan menjalankan fungsi *handleLogin* saat klik tombol *submit*, dan pesan *error* ditampilkan jika proses *login* gagal.

K Halaman Admin Dashboard Ubah Carousel

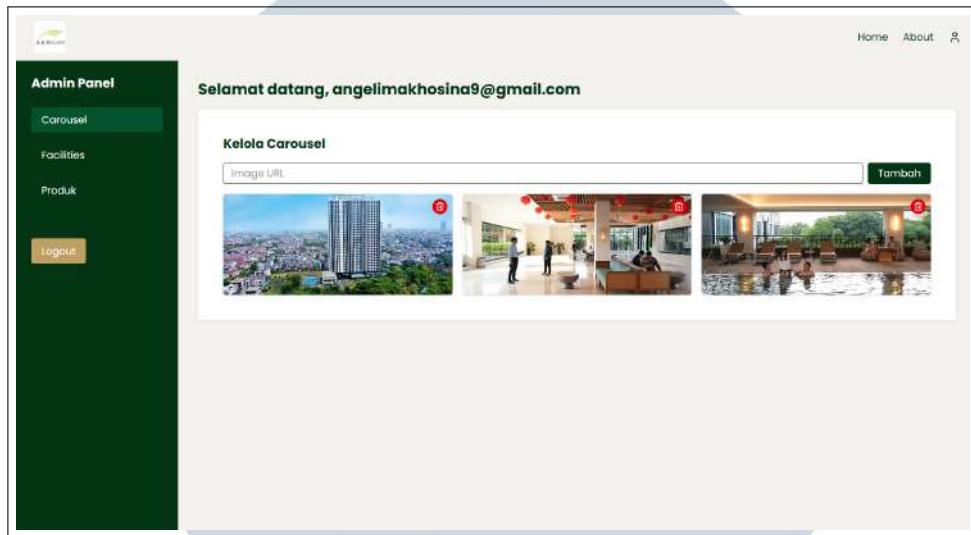


Gambar 3.18. Diagram Alur Sistem Ubah Carousel

Gambar 3.18 menunjukkan alur proses ketika admin menambahkan gambar baru ke bagian *carousel* melalui *dashboard admin*. Proses dimulai dari tampilan *dashboard*, di mana admin dapat menginput URL gambar yang ingin ditambahkan.

Sistem kemudian akan memverifikasi apakah URL yang dimasukkan valid dan sesuai dengan format yang diizinkan (misalnya URL mengarah ke file gambar berekstensi .jpg, .png, dll). Jika URL tidak sesuai, sistem akan meminta *admin*

untuk menginput ulang. Namun, jika valid, data gambar akan disimpan ke dalam database Supabase, dan gambar akan langsung muncul di tampilan carousel utama situs web.



Gambar 3.19. Halaman Admin Dashboard Ubah Carousel

Gambar 3.19 menunjukkan tampilan halaman "Kelola Carousel" pada *dashboard admin website* Apartemen Aerium Residence. Halaman ini memungkinkan *admin* yang telah *login* untuk menambah dan menghapus gambar yang ditampilkan pada *carousel* halaman utama.

Fitur yang tersedia meliputi input URL gambar, tombol "Tambah", dan daftar gambar yang telah diunggah. Setiap gambar memiliki ikon tempat sampah berwarna merah untuk menghapus gambar dari *database* Supabase.

```
1 return (
2   <div className="p-4">
3     <h2 className="text-xl text-[#033414] font-bold mb-4">
4       Kelola Carousel</h2>
5
6     <div className="mb-4 flex gap-2">
7       <input
8         type="text"
9         value={newImage}
10        onChange={(e) => setNewImage(e.target.value)}
11        placeholder="Image URL"
12        className="border px-3 py-1 rounded flex-grow"
13      />
14       <button>
```

```

14     onClick={addImage}
15     className="bg-[#033414] text-white px-4 py-1 rounded
16     hover:bg-green-800"
17     >
18       Tambah
19     </button>
20   </div>
21
22   <div className="grid grid-cols-2 md:grid-cols-3 gap-4">
23     {images.map((item) => (
24       <div key={item.id} className="relative group">
25         <img
26           src={item.image_url}
27           alt="Carousel"
28           className="rounded shadow w-full h-40 object-cover"
29         />
30         <button
31           onClick={() => confirmDelete(item.id)}
32           className="absolute top-2 right-2 bg-red-600 text-
33           white p-1 rounded-full hover:bg-red-700"
34           >
35             <Trash2 size={16} />
36           </button>
37         </div>
38     )) }
39   </div>
40
41   {confirmDeleteId && (
42     <div className="fixed inset-0 bg-black/40 flex items-
43     center justify-center z-50">
44       <div className="bg-white rounded-lg shadow-lg max-w-sm
45       w-full p-6">
46         <h3 className="text-lg font-semibold mb-3 text-
47         -[ #033414 ]">Konfirmasi Hapus</h3>
48         <p className="text-sm mb-5 text-gray-700">
49           Apakah kamu yakin ingin menghapus gambar ini?
50         </p>
51         <div className="flex justify-end gap-2">
52           <button
53             onClick={cancelDelete}
54             className="bg-gray-300 hover:bg-gray-400 text-
55             gray-800 px-4 py-2 rounded"
56           >

```

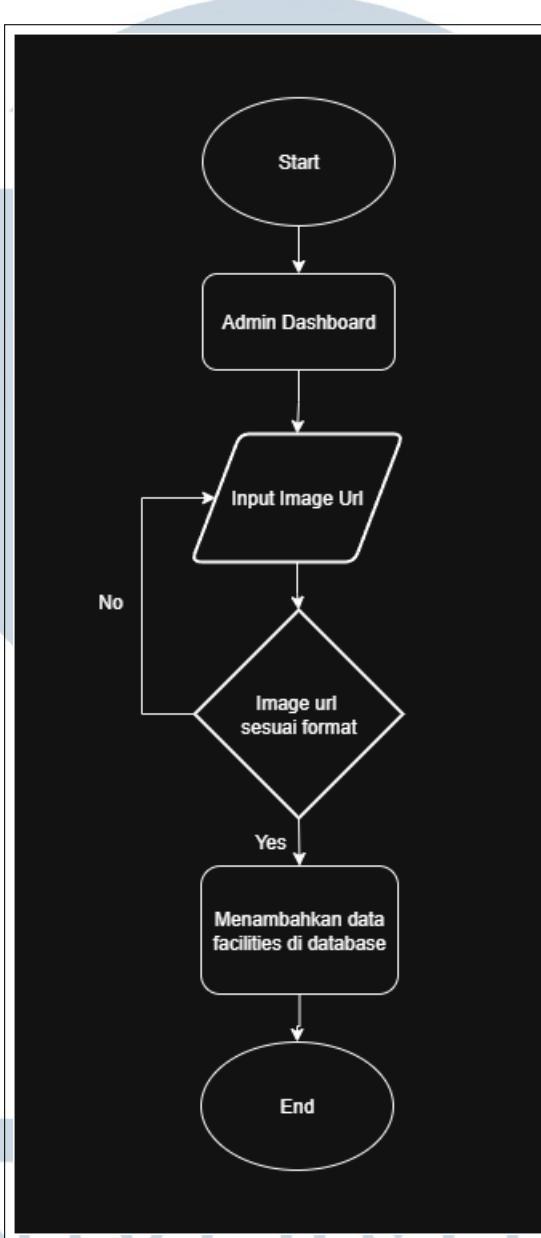
```
51         Batal
52     </button>
53     <button
54         onClick={deleteImage}
55         className="bg-red-600 hover:bg-red-700 text-white
px-4 py-2 rounded"
56     >
57         Hapus
58     </button>
59     </div>
60   </div>
61 )
62 </div>
63 )
64 );
```

Kode 3.18: Halaman Admin Dashboard Ubah Carousel

Kode 3.18 merupakan bagian dari komponen React yang berfungsi untuk mengelola gambar-gambar carousel di halaman dashboard admin sebuah website properti. Komponen ini mencakup fitur menambahkan, menampilkan, dan menghapus gambar, serta menyediakan modal konfirmasi saat akan menghapus.



L Halaman Admin Dashboard Ubah Fasilitas



Gambar 3.20. Diagram Alur Sistem Ubah Fasilitas

Gambar 3.20 menjelaskan alur proses penambahan gambar fasilitas ke dalam sistem oleh admin melalui *dashboard admin*. Proses dimulai dari halaman dashboard, di mana admin diberikan form untuk menginput URL gambar fasilitas yang ingin ditampilkan di website.

Setelah URL dimasukkan, sistem akan melakukan validasi apakah URL tersebut sesuai dengan format yang diperbolehkan. Jika tidak sesuai, admin akan

diminta untuk memasukkan ulang URL gambar. Jika URL valid, maka sistem akan menyimpan URL tersebut ke dalam database (tabel facilities) yang selanjutnya akan digunakan untuk menampilkan gambar fasilitas di halaman pengguna.



Gambar 3.21. Halaman Admin Dashboard Ubah Fasilitas

Gambar 3.21 menunjukkan tampilan halaman pengelolaan fasilitas dalam *dashboard admin*. Pada halaman ini, *admin* dapat menambahkan atau menghapus gambar fasilitas yang ditampilkan di halaman utama *website*. Gambar-gambar fasilitas ditampilkan dalam bentuk galeri, dan terdapat ikon tempat sampah berwarna merah pada setiap gambar untuk menghapus data yang tidak diperlukan.

```
1 return (
2   <div className="p-4">
3     <h2 className="text-xl font-bold mb-4 text-[#033414]">
4       Kelola Fasilitas </h2>
5
6     <div className="mb-4 flex gap-2">
7       <input
8         type="text"
9         value={newImage}
10        onChange={(e) => setNewImage(e.target.value)}
11        placeholder="Image URL"
12        className="border px-3 py-1 rounded flex-grow"
13      />
14      <button
15        onClick={addImage}
16        className="bg-[#033414] text-white px-4 py-1 rounded
17        hover:bg-green-700">
```

```

17         Tambah
18     </button>
19 </div>
20
21 <div className="grid grid-cols-2 md:grid-cols-3 lg:grid-
cols-4 gap-4">
22     {images.map((item) => (
23         <div key={item.id} className="relative group">
24             <img
25                 src={item.image_url}
26                 alt="Facility"
27                 className="rounded shadow w-full h-40 object-cover"
28             />
29             <button
30                 onClick={() => confirmDelete(item.id)}
31                 className="absolute top-2 right-2 bg-red-600 text-
white p-1 rounded-full shadow hover:bg-red-700"
32             >
33                 <Trash2 size={16} />
34             </button>
35         </div>
36     )) }
37 </div>
38
39 {confirmDeleteId && (
40     <div className="fixed inset-0 bg-black/40 flex items-
center justify-center z-50">
41         <div className="bg-[#f3f2ee] rounded-lg shadow-lg max-w-
sm w-full p-6">
42             <h3 className="text-lg font-semibold mb-3 text-
-[#033414]">Konfirmasi Hapus</h3>
43             <p className="text-sm mb-5 text-gray-700">
44                 Apakah kamu yakin ingin menghapus gambar ini?
45             </p>
46             <div className="flex justify-end gap-2">
47                 <button
48                     onClick={cancelDelete}
49                     className="bg-white hover:bg-gray-400 text-gray-
800 px-4 py-2 rounded"
50                 >
51                     Batal
52                 </button>
53                 <button

```

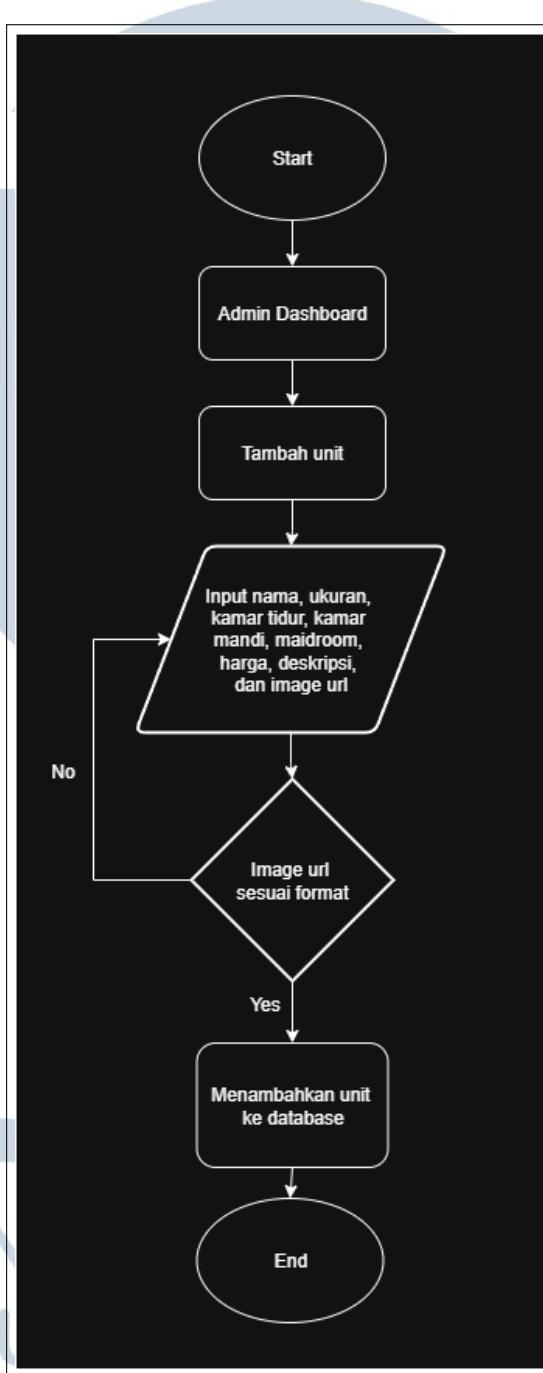
```
54         onClick={deleteImage}
55         className="bg-red-600 hover:bg-red-700 text-white
56         px-4 py-2 rounded"
57         >
58             Hapus
59         </button>
60     </div>
61 </div>
62 ) }
63 </div>
64 );
```

Kode 3.19: Halaman Admin Dashboard Ubah Fasilitas

Kode 3.19 merupakan bagian dari komponen yang digunakan untuk mengelola gambar-gambar pada bagian Fasilitas dalam halaman *dashboard admin*. Komponen ini mencakup fitur untuk menambahkan, menampilkan, dan menghapus gambar, lengkap dengan modal konfirmasi penghapusan.



M Halaman Admin Dashboard Ubah Produk

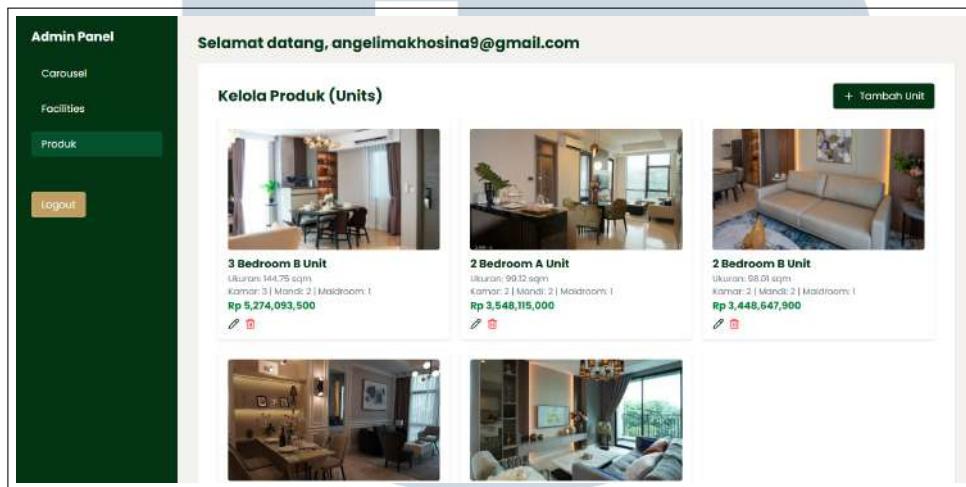


Gambar 3.22. Diagram Alur Sistem Ubah Produk

Gambar 3.22 menggambarkan proses alur kerja *admin* saat menambahkan data produk atau unit apartemen baru melalui *dashboard admin*. Proses dimulai ketika admin mengakses halaman produk, kemudian mengisi seluruh informasi

unit seperti nama, ukuran, jumlah kamar tidur, kamar mandi, maidroom, harga, deskripsi, dan gambar unit.

Setelah data diinput, sistem akan melakukan validasi terhadap format data. Jika terdapat data yang tidak sesuai (misalnya kosong atau format salah), maka admin akan diminta untuk memperbaiki. Namun, apabila semua data telah valid, maka data unit tersebut akan ditambahkan ke dalam tabel `apart_images` pada database Supabase.



Gambar 3.23. Halaman Admin Dashboard Ubah Produk

Gambar 3.23 menampilkan halaman pengelolaan produk di dalam *dashboard admin*. Halaman ini memungkinkan *admin* untuk melihat, menambah, mengubah, atau menghapus data unit apartemen yang tersedia. Setiap unit ditampilkan dalam bentuk kartu yang memuat informasi lengkap seperti nama unit, ukuran, jumlah kamar, harga, serta foto properti. Tindakan mengubah dan hapus tersedia di bagian bawah setiap kartu produk.

```
1 return (
2   <div className="p-2">
3     <div className="flex justify-between items-center mb-4">
4       <h2 className="text-2xl text-[#033414] font-bold">Kelola
5         Produk (Units)</h2>
6       <button
7         onClick={() => setIsCreateOpen(true)}
8         className="flex items-center gap-2 px-4 py-2 bg
9           -[#033414] text-white rounded hover:bg-green-800"
10        >
11          <Plus size={18} /> Tambah Unit
12        </button>
```

```

11     </div>
12
13     <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-
14       cols-3 gap-4">
15       {units.map((u) => (
16         <div key={u.id} className="bg-white rounded shadow p-4">
17
18           <img
19             src={u.image_url[0]}
20             alt={u.name}
21             className="w-full h-48 object-cover rounded"
22           />
23           <h3 className="text-lg text-[#033414] font-semibold
24             mt-2">{u.name}</h3>
25           <p className="text-sm text-gray-500">Ukuran: {u.
26             ukuran} sqm</p>
27           <p className="text-sm text-gray-500">
28             Kamar: {u.bedroom} | Mandi: {u.bathroom} | Maidroom
29             : {u.maidroom}
30           </p>
31           <p className="text-green-700 font-bold">
32             Rp {Number(u.harga).toLocaleString()}
33           </p>
34           <div className="flex gap-2 mt-2">
35             <button
36               onClick={() => handleEdit(u)}
37               className="text-[#033414] hover:text-blue-700"
38             >
39               <Pencil size={18} />
40             </button>
41             <button
42               onClick={() => {
43                 setSelectedToDelete(u);
44                 setDeleteConfirmOpen(true);
45               }
46               className="text-red-500 hover:text-red-700"
47             >
48               <Trash2 size={18} />
49             </button>
50           </div>
51         </div>
52       )) }
53     </div>

```

```

49
50     /* Modal Edit */
51     <Dialog open={isEditOpen} onClose={() => setIsEditOpen(
52       false)} className="relative z-50">
53       <div className="fixed inset-0 bg-black/30" aria-hidden="true" />
54       <div className="fixed inset-0 flex items-center justify-center p-4">
55         <Dialog.Panel className="bg-white text-[#033414] rounded-lg shadow-xl max-w-xl w-full p-6 space-y-4">
56           <Dialog.Title className="text-xl font-bold mb-2">Edit
57             Unit</Dialog.Title>
58           {editData && (
59             <form className="space-y-3">
60               <div>
61                 <label className="block font-medium">Nama:</
62                   label>
63                   <input
64                     type="text"
65                     value={editData.name}
66                     onChange={(e) => setEditData({ ...editData,
67                       name: e.target.value })}
68                     className="w-full p-2 border rounded"
69                     />
70               </div>
71               <div>
72                 <label className="block font-medium">Ukuran (sqm):</label>
73                 <input
74                   type="text"
75                     value={editData.ukuran}
76                     onChange={(e) => setEditData({ ...editData,
77                       ukuran: e.target.value })}
78                     className="w-full p-2 border rounded"
79                     />
80               </div>
81               <div className="grid grid-cols-3 gap-4">
82                 <div>
83                   <label className="block font-medium">Kamar
84                     Tidur:</label>
85                   <input

```

```

82         type="number"
83         value={editData.bedroom}
84         onChange={(e) => setEditData({ ...editData,
bedroom: parseInt(e.target.value) })}
85             className="w-full p-2 border rounded"
86         />
87     </div>
88     <div>
89         <label className="block font-medium">Kamar
Mandi:</label>
90         <input
91             type="number"
92             value={editData.bathroom}
93             onChange={(e) => setEditData({ ...editData,
bathroom: parseInt(e.target.value) })}
94             className="w-full p-2 border rounded"
95         />
96     </div>
97     <div>
98         <label className="block font-medium">Maidroom
:</label>
99         <input
100            type="number"
101            value={editData.maidroom}
102            onChange={(e) => setEditData({ ...editData,
maidroom: parseInt(e.target.value) })}
103            className="w-full p-2 border rounded"
104        />
105    </div>
106 </div>
107
108 <div>
109         <label className="block font-medium">Harga:</
label>
110         <input
111             type="number"
112             value={editData.harga}
113             onChange={(e) => setEditData({ ...editData,
harga: parseInt(e.target.value) })}
114             className="w-full p-2 border rounded"
115         />
116     </div>
117

```

```

118         <div>
119             <label className="block font-medium">Deskripsi
120             :</label>
121             <textarea
122                 value={editData.desc}
123                 onChange={(e) => setEditData({ ...editData,
124                 desc: e.target.value })}
125                     className="w-full p-2 border rounded h-24"
126                     />
127             </div>
128
129         <div>
130             <label className="block font-medium">Image URL
131             :</label>
132             <input
133                 type="text"
134                 value={editData.image_url}
135                 onChange={(e) => setEditData({ ...editData,
136                 image_url: e.target.value })}
137                     className="w-full p-2 border rounded"
138                     />
139             </div>
140
141         <div className="flex justify-end gap-2">
142             <button
143                 type="button"
144                 onClick={() => setIsEditOpen(false)}
145                 className="px-4 py-2 bg-[#f3f2ee] rounded
146                 hover:bg-gray-400">
147                 Batal
148             </button>
149             <button
150                 type="button"
151                 onClick={handleUpdate}
152                 className="px-4 py-2 bg-[#033414] text-white
153                 rounded hover:bg-green-800">
154                 Simpan
155             </button>
156         </div>
157     </form>
158   ) }

```

```

155         </Dialog.Panel>
156     </div>
157   </Dialog>
158
159   /* Modal Konfirmasi Delete */
160   <Dialog open={deleteConfirmOpen} onClose={() =>
161     setDeleteConfirmOpen(false) } className="relative z-50">
162     <div className="fixed inset-0 bg-black/30" aria-hidden="true" />
163     <div className="fixed inset-0 flex items-center justify-center p-4">
164       <Dialog.Panel className="bg-[#f3f2ee] rounded-lg shadow-xl max-w-sm w-full p-6 space-y-4">
165         <Dialog.Title className="text-lg font-bold text-left text-[#033414]">Konfirmasi Hapus</Dialog.Title>
166         <p className="text-left text-gray-700">
167           Apakah kamu yakin ingin menghapus unit <strong>{selectedToDelete?.name}</strong>?
168         </p>
169         <div className="flex justify-end gap-2 mt-4">
170           <button
171             onClick={() => {
172               setDeleteConfirmOpen(false);
173               setSelectedToDelete(null);
174             }
175             className="px-4 py-2 bg-white rounded hover:bg-gray-400"
176             >
177               Batal
178             </button>
179           <button
180             onClick={handleDelete}
181             className="px-4 py-2 bg-red-600 text-white rounded hover:bg-red-700"
182             >
183               Hapus
184             </button>
185           </div>
186         </Dialog.Panel>
187       </div>
188     </Dialog>
189
190   /* Modal Create */

```

```

190     <Dialog open={isCreateOpen} onClose={() => setIsCreateOpen(
191       false)} className="relative z-50">
192       <div className="fixed inset-0 bg-black/30" aria-hidden="true" />
193       <div className="fixed inset-0 flex items-center justify-center p-4">
194         <Dialog.Panel className="bg-white rounded-lg shadow-xl max-w-xl w-full p-6 space-y-4">
195           <Dialog.Title className="text-xl font-bold mb-2">
196             Tambah Unit Baru</Dialog.Title>
197             <form className="space-y-3">
198               <div>
199                 <label className="block font-medium">Nama:</label>
200               >
201                 <input
202                   type="text"
203                   value={createData.name}
204                   onChange={(e) => setCreateData({ ...createData,
205                     name: e.target.value })}>
206                   className="w-full p-2 border rounded"
207                   />
208               </div>
209
210               <div>
211                 <label className="block font-medium">Ukuran (sqm)</label>
212               <input
213                   type="text"
214                   value={createData.ukuran}
215                   onChange={(e) => setCreateData({ ...createData,
216                     ukuran: e.target.value })}>
217                   className="w-full p-2 border rounded"
218                   />
219               </div>
220
221               <div className="grid grid-cols-3 gap-4">
222                 <div>
223                   <label className="block font-medium">Kamar Tidur:</label>
224                   <input
225                       type="number"
226                       value={createData.bedroom}
227                       onChange={(e) => setCreateData({ ...

```

```

223     createData, bedroom: parseInt(e.target.value) })
224             className="w-full p-2 border rounded"
225         />
226     </div>
227     <div>
228         <label className="block font-medium">Kamar
229         Mandi:</label>
230         <input
231             type="number"
232             value={createData.bathroom}
233             onChange={(e) => setCreateData({ ...
234             createData, bathroom: parseInt(e.target.value) })
235             className="w-full p-2 border rounded"
236             />
237         </div>
238         <div>
239             <label className="block font-medium">Maidroom
240             :</label>
241             <input
242                 type="number"
243                 value={createData.maidroom}
244                 onChange={(e) => setCreateData({ ...
245             createData, maidroom: parseInt(e.target.value) })
246             className="w-full p-2 border rounded"
247             />
248         </div>
249     </div>
250
251     <div>
252         <label className="block font-medium">Harga:</
253         label>
254         <input
255             type="number"
256             value={createData.harga}
257             onChange={(e) => setCreateData({ ...createData,
258             harga: parseInt(e.target.value) }) }
259             className="w-full p-2 border rounded"
260             />
261         </div>
262
263     <div>
264         <label className="block font-medium">Deskripsi:</
265         label>

```

```

258         <textarea
259             value={createData.desc}
260             onChange={(e) => setCreateData({ ...createData,
261             desc: e.target.value })}
262             className="w-full p-2 border rounded h-24"
263             />
264         </div>
265
266         <div>
267             <label className="block font-medium">Image URL:</
268             label>
269             <input
270                 type="text"
271                 value={createData.image_url}
272                 onChange={(e) => setCreateData({ ...createData,
273                 image_url: e.target.value })}
274                 className="w-full p-2 border rounded"
275                 />
276         </div>
277
278         <div className="flex justify-end gap-2">
279             <button
280                 type="button"
281                 onClick={() => setIsCreateOpen(false)}
282                 className="px-4 py-2 bg-gray-300 rounded hover:
283                 bg-gray-400"
284                 >
285                 Batal
286             </button>
287             <button
288                 type="button"
289                 onClick={handleCreate}
290                 className="px-4 py-2 bg-[#033414] text-white
291                 rounded hover:bg-green-800"
292                 >
293                 Simpan
294             </button>
295         </div>
296     </form>
297 </Dialog.Panel>
298 </div>
299 </Dialog>
300 </div>

```

Kode 3.20: Halaman Admin Dashboard Ubah Produk

Kode 3.20 adalah komponen React yang digunakan untuk mengelola data produk atau unit properti di halaman dashboard admin. Komponen ini menyajikan fitur utama berupa tampilan daftar unit, serta modal untuk menambah, mengedit, dan menghapus unit.

Unit-unit ditampilkan dalam bentuk kartu dengan grid responsif. Setiap kartu menampilkan gambar properti, nama unit, ukuran, jumlah kamar tidur, kamar mandi, maidroom, serta harga yang diformat sebagai rupiah. Dua tombol aksi tersedia di bagian bawah kartu, yaitu tombol edit (ikon pensil) yang membuka modal pengeditan, dan tombol hapus (ikon tempat sampah) yang memicu modal konfirmasi penghapusan.

Modal edit akan muncul saat admin mengklik tombol edit. Modal ini menampilkan form dengan kolom-kolom yang sudah terisi nilai unit yang dipilih (nama, ukuran, kamar, harga, deskripsi, dan URL gambar). *Admin* dapat mengubah data tersebut dan menekan tombol Simpan untuk memperbarui informasi. Sementara itu, modal hapus menampilkan pesan konfirmasi yang menyebutkan nama unit yang akan dihapus dan menyediakan tombol Batal dan Hapus.

Modal tambah unit baru serupa dengan modal *edit*, namun semua *field* kosong. *Admin* dapat mengisi detail unit baru dan mengklik tombol Simpan untuk menambahkan unit ke dalam sistem.

3.5 Pengujian dan Evaluasi

Setelah seluruh fitur dan halaman *website* Apartemen Aerium Residence selesai dikembangkan, dilakukan proses pengujian terhadap sistem secara menyeluruh oleh supervisor. Pengujian ini mencakup fungsionalitas halaman utama, sistem autentikasi *admin*, serta fitur *dashboard admin* seperti manajemen gambar carousel, fasilitas, dan produk (unit apartemen), serta tampilan detail unit bagi calon klien.

Supervisor melakukan evaluasi terhadap aspek fungsionalitas, kemudahan penggunaan, dan konsistensi tampilan antarmuka. Setelah pengujian tahap awal, supervisor kemudian mendistribusikan tautan website kepada beberapa agen properti yang tergabung dalam tim pemasaran.

Para agen properti ini diminta untuk mencoba sistem secara langsung, termasuk:

- Mengakses halaman publik untuk menampilkan *listing* apartemen
- Menggunakan halaman detail unit saat berdiskusi dengan calon pembeli
- Memberikan umpan balik terkait kecepatan akses, kenyamanan antarmuka, serta kejelasan informasi properti yang ditampilkan

Dari hasil pengujian ini, diperoleh berbagai masukan dan umpan balik yang menjadi dasar dalam proses perbaikan dan penyempurnaan akhir website. Evaluasi yang dilakukan oleh supervisor dan agen properti memastikan bahwa sistem yang dibangun telah memenuhi kebutuhan operasional dan siap digunakan dalam kegiatan pemasaran properti secara digital.

3.6 Kendala dan Solusi

Selama proses magang, ditemukan sejumlah kendala teknis yang memengaruhi kelancaran pengembangan *website property listing*, khususnya dalam implementasi dashboard admin untuk pengelolaan konten. Salah satu kendala utama adalah tidak berfungsi fitur unggah gambar langsung dari perangkat lokal ke Supabase Storage melalui dashboard admin. Sistem pada awalnya hanya mendukung input berupa URL gambar yang sudah tersedia, karena fungsi `upload()` dari `supabase-js` belum diimplementasikan dan kebijakan keamanan (*policy*) pada bucket penyimpanan belum mengizinkan operasi *insert*.

Permasalahan ini diatasi dengan mengintegrasikan komponen input file pada antarmuka admin dan menerapkan fungsi `upload()` agar file dapat dikirim langsung dari browser ke Supabase Storage. Setelah berhasil diunggah, gambar diproses menggunakan `getPublicUrl()` untuk mendapatkan tautan publik yang kemudian disimpan ke dalam tabel database. Selain itu, kebijakan akses (*access policy*) pada bucket dikonfigurasi ulang agar pengguna yang terautentifikasi memiliki izin untuk mengunggah file.

Kendala lain yang dihadapi adalah keterbatasan asset visual dan data properti saat proses awal pengembangan. Informasi seperti gambar unit, ikon fasilitas, ukuran unit, dan status ketersediaan belum tersedia secara lengkap, sehingga menyulitkan penyusunan konten halaman. Untuk menjaga kelangsungan pengembangan, digunakan sementara gambar *placeholder* dan data dummy, sambil

dilakukan koordinasi dengan supervisor untuk memperoleh aset resmi secara bertahap.

Tantangan teknis juga ditemukan pada integrasi antara sisi *frontend* (Next.js) dan *backend* (Supabase), khususnya dalam pengaturan autentikasi dan kontrol akses. Beberapa permintaan data ditolak karena token autentikasi tidak dikenali atau karena aturan *row-level security* (RLS) yang belum sesuai. Solusi yang diterapkan mencakup konfigurasi ulang Supabase client dengan konteks autentikasi pengguna aktif, serta penyusunan ulang kebijakan RLS agar selaras dengan logika pengguna yang telah login.

Dari sisi manajemen proyek, keterbatasan komunikasi akibat sistem kerja jarak jauh menjadi tantangan tersendiri. Proses klarifikasi teknis, validasi desain, dan pengambilan keputusan terkadang mengalami keterlambatan karena tidak adanya komunikasi real-time. Untuk mengatasinya, dilakukan pertemuan mingguan (*weekly meeting*) secara terjadwal dan pelaporan progres dilakukan secara rutin setiap hari melalui platform internal, sehingga proses pengembangan tetap terpantau meskipun tanpa tatap muka langsung.

Dengan pendekatan iteratif dan pemecahan masalah secara teknis yang sistematis, fitur utama seperti unggah gambar, autentikasi pengguna, dan pengelolaan konten melalui dashboard berhasil direalisasikan dan dapat berfungsi dengan baik untuk kebutuhan internal perusahaan.

