

Dokumentasi Navbar dan Autentikasi

Nama : Naufal Azmi Mukhlis

Dokumen ini menjelaskan implementasi komponen navigasi (Navbar dan Navlink) serta konfigurasi autentikasi Google menggunakan **NextAuth** dalam aplikasi.

1. Komponen Navbar dan Navigasi

Komponen ini bertanggung jawab untuk menampilkan bilah navigasi utama aplikasi, termasuk logo, tautan navigasi, dan tombol *Sign In*. Desainnya responsif, menyesuaikan tampilan untuk perangkat *mobile* dan *desktop*.

A. Komponen navbar.tsx

Komponen utama yang menyusun struktur *header* dan menyediakan kerangka *layout* untuk navigasi.

- **Lokasi File:** navbar.tsx
- **Fungsi Utama:**
 - Menampilkan **logo** (`/logo.png`) menggunakan komponen Image dari Next.js, yang merupakan tautan ke halaman utama (`/`).
 - Menggunakan div luar dengan kelas `fixed top-0 w-full bg-white shadow-sm z-20` untuk membuat *navbar* tetap berada di bagian atas layar dan memiliki bayangan halus.
 - Mengimpor dan merender komponen **Navlink** untuk menangani daftar tautan navigasi dan *toggle menu* di perangkat *mobile*.

Elemen	Deskripsi
Image	Menampilkan logo, <i>link</i> ke <code>/</code> . Dimensi: <code>width={128}</code> dan <code>height={49}</code> .
Navlink	Komponen turunan yang berisi semua tautan menu dan logika <i>mobile menu</i> .
Styling	Menggunakan Tailwind CSS untuk <i>styling</i> , termasuk <i>fixed positioning</i> dan <i>shadow</i> .

B. Komponen navlink.tsx

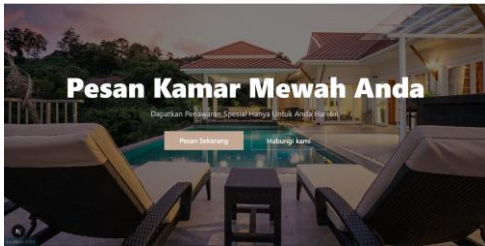
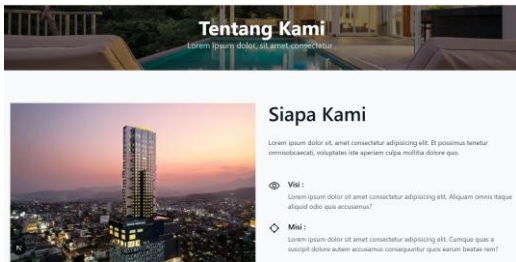
Komponen ini berisi semua item menu, logika responsif (menampilkan/menyembunyikan menu di *mobile*), dan tombol *Sign In*.

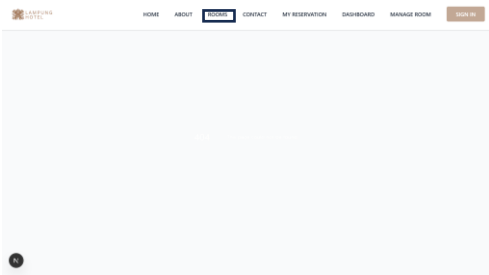
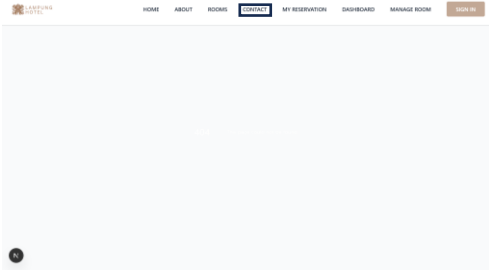
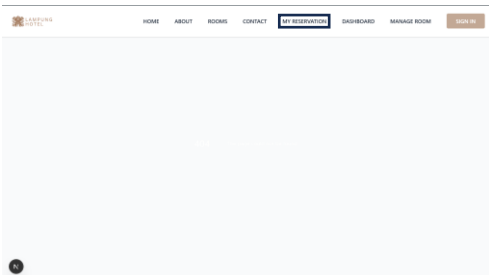
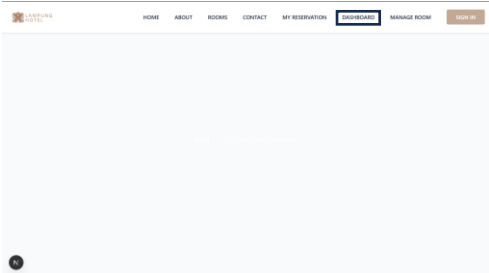
- **Lokasi File:** navlink.tsx
- **Tipe Komponen:** **Client Component** (Ditandai dengan "use client" karena menggunakan *state* `useState` dan *event handler* `onClick`).

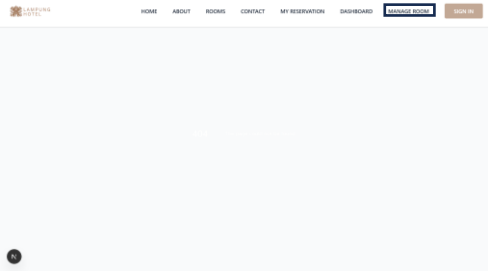
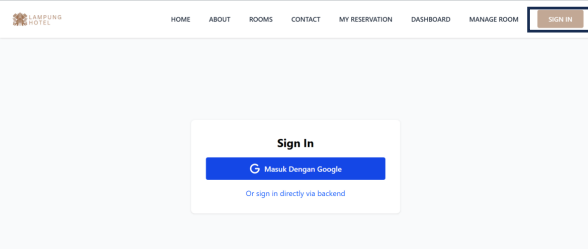
- **Fitur Responsif (Mobile Toggle):**

- Menggunakan *state open* untuk melacak status buka/tutup menu navigasi *mobile*.
- Tombol *toggle* (<button>) hanya terlihat pada ukuran layar **kecil** (md:hidden).
- Ikon menu menggunakan IoMenu (saat tertutup) dan IoClose (saat terbuka) dari react-icons/io5.
- Daftar tautan (<div> yang membungkus) disembunyikan secara *default* pada *mobile* dan ditampilkan ketika *state open* adalah true, dikontrol oleh fungsi **clsx**:

- **Daftar Tautan Navigasi ():**

Teks Tautan	href (Tujuan)	Catatan
Home	/	Halaman utama.
<div>  <p>Ketika pengguna/user menekan tombol <i>Home</i> pada bagian atas (navbar). Maka pengguna akan di arahkan ke halaman pada samping ini.</p> </div>		
About	/about	Halaman tentang.
<div>  <p>Ketika pengguna/user menekan tombol <i>About</i> pada bagian atas (navbar). Maka pengguna akan di arahkan ke halaman pada samping ini.</p> </div>		
Rooms	/room	Halaman daftar kamar.

Teks Tautan	href (Tujuan)	Catatan
		<p>Ketika pengguna/user menekan tombol <i>Rooms</i> pada bagian atas (navbar). Maka pengguna akan di arahkan ke halaman pada samping ini.</p> <p>(masih dalam tahap pengembangan)</p>
Contact	/contact	Halaman kontak.
		<p>Ketika pengguna/user menekan tombol <i>Contact</i> pada bagian atas (navbar). Maka pengguna akan di arahkan ke halaman pada samping ini.</p> <p>(masih dalam tahap pengembangan)</p>
My Reservation	/myreservation	Reservasi pengguna.
		<p>Ketika pengguna/user menekan tombol <i>My Reservation</i> pada bagian atas (navbar). Maka pengguna akan di arahkan ke halaman pada samping ini.</p> <p>(masih dalam tahap pengembangan)</p>
Dashboard	admin/dashboard	Area admin (perlu otorisasi).
		<p>Ketika admin menekan tombol <i>Dashboard</i> pada bagian atas (navbar). Maka admin akan di arahkan ke halaman pada samping ini.</p> <p>(masih dalam tahap pengembangan)</p>

Teks Tautan	href (Tujuan)	Catatan
Manage Room	admin/room	Area admin (perlu otorisasi).
<div>  <p>Ketika admin menekan tombol <i>Manage Room</i> pada bagian atas (navbar). Maka admin akan di arahkan ke halaman pada samping ini. (masih dalam tahap pengembangan)</p> </div>		
Sign In	/signin	Tombol login, dengan <i>styling</i> warna latar belakang cokelat/krem (bg-[#C2A895]).
<div>  <p>Ketika admin dan user menekan tombol <i>Sign In</i> pada bagian atas (navbar). Maka admin dan user akan di arahkan ke halaman pada samping ini. (masih dalam tahap pengembangan)</p> </div>		

2. Konfigurasi Autentikasi (NextAuth)

Autentikasi diimplementasikan menggunakan **NextAuth.js** dan dikonfigurasi untuk menggunakan **Google Provider**. Hal ini memungkinkan pengguna untuk masuk (*Sign In*) menggunakan akun Google mereka, seperti yang terlihat pada tangkapan layar.

A. File Konfigurasi route.ts

File ini mendefinisikan *endpoint* API untuk proses otentikasi.

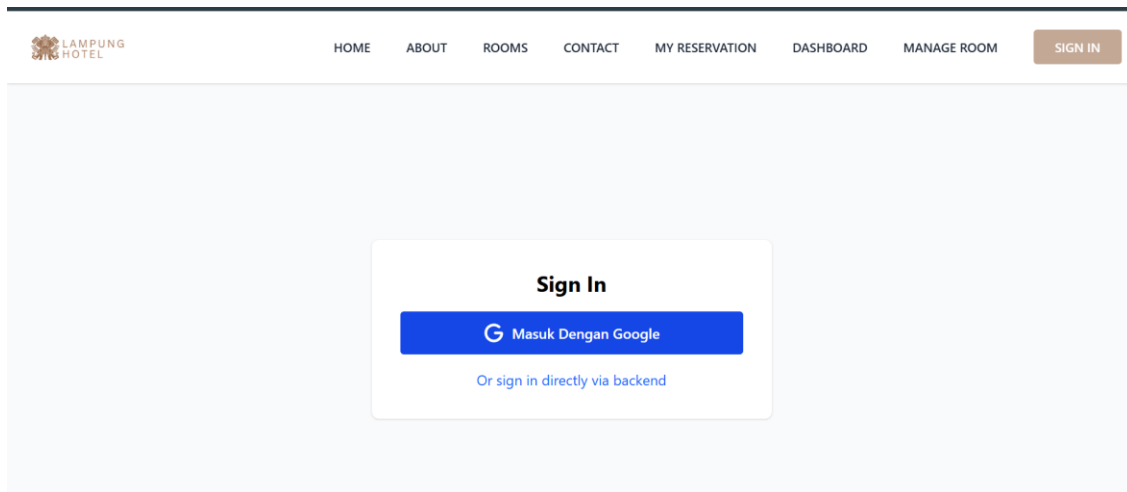
- **Lokasi File:** route.ts (di jalur app/api/auth/[...nextauth]/route.ts)
- **Tipe File: Server-Side** (Ditandai dengan "use server").
- **Provider yang Digunakan: GoogleProvider**
 - Konfigurasi membutuhkan dua variabel lingkungan (environment variables):
 - **GOOGLE_CLIENT_ID:** ID Klien yang didapatkan dari Google Developer Console.
 - **GOOGLE_CLIENT_SECRET:** Rahasia Klien yang didapatkan dari Google Developer Console.

(Catatan: Karakter ! pada akhir variabel menunjukkan bahwa nilainya dijamin ada).

- **Export Handler:**

- **export { handler as GET, handler as POST };**
- Ini mengkonfigurasi NextAuth untuk merespons permintaan **GET** (untuk menampilkan halaman *Sign In* dan memulai proses *handshake*) dan permintaan **POST** (untuk menangani *callback* dan *sign out*) ke *endpoint* autentikasi.

B. Alur Kerja Autentikasi



menunjukkan antarmuka *Sign In* yang dihasilkan oleh NextAuth/aplikasi:

1. **Akses:** Pengguna mengklik tombol **Sign In** di *Navbar*, yang mengarah ke halaman */signin*.
2. **Tampilan:** Halaman *Sign In* menampilkan opsi utama:
 - **"Masuk Dengan Google":** Tombol ini akan memicu alur autentikasi Google OAuth. NextAuth akan mengarahkan pengguna ke halaman *login* Google untuk memasukkan kredensial mereka.
 - **"Or sign in directly via backend":** Opsi cadangan atau alternatif untuk *login* melalui *backend* aplikasi secara langsung (jika ada).
3. **Proses Login:** Setelah pengguna berhasil masuk ke Google, Google akan mengarahkan balik (*callback*) ke *endpoint* yang dikonfigurasi di *route.ts*, di mana NextAuth akan memproses sesi pengguna.
4. **Status Login:** Setelah berhasil, pengguna akan dianggap terautentikasi dan biasanya diarahkan kembali ke halaman utama atau halaman yang diminta sebelumnya. Tautan **Dashboard** dan **Manage Room** (dan mungkin tautan lain) hanya boleh dapat diakses oleh pengguna dengan peran tertentu (misalnya, **Admin**), yang harus diimplementasikan melalui pemeriksaan sesi NextAuth. **(Namun Proses ini masih dalam tahap pengembangan lebih lanjut.)**