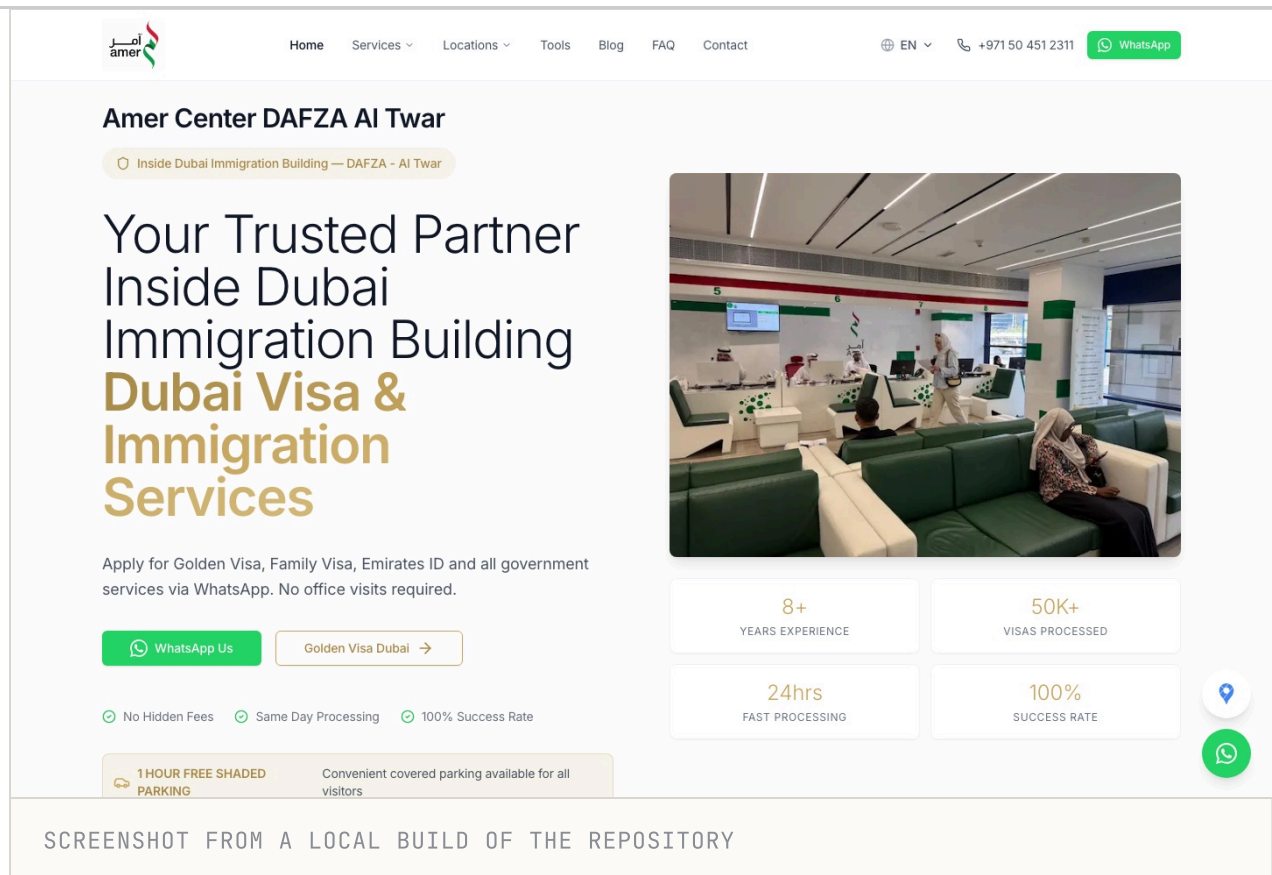


Amer Center Dubai: a multi-branch government-services portal built for local search

Amer centres process UAE government services: Emirates ID, visas, family sponsorship, business setup. This client operates several physical branches across Dubai and Sharjah, and the business problem was discoverability: a...

Role: Architect and developer

Stack: Next.js frontend (Radix UI, TanStack Query), Node and TypeScript backend, plus a separate e-services application-management portal



Context

Amer centres process UAE government services: Emirates ID, visas, family sponsorship, business setup. This client operates several physical branches across Dubai and Sharjah, and the business problem was discoverability: a resident searching for "Emirates ID renewal Karama" needs to find the right branch and the right service, then enquire, with as little friction as possible. The site is built around that search behaviour.

What I built

Location and service architecture

A dedicated page for each branch (Deira, Al Nahda, Karama, Sharjah, Mirdif, Al Twar and central Dubai) and a dedicated page for each service line (Emirates ID, Golden Visa, Family Visa, Business Setup and general visa services), so every realistic local-search query has a page that answers it. Supporting pages for FAQs, tools, a blog and contact, and a client portal.

A deliberately light backend

Three models and four route groups: services (the catalogue), inquiries (lead capture), guest blog posts (content marketing, including externally contributed articles) and authentication. The complexity of this product is in its content and location structure, not in transactions, and the backend is sized to match rather than padded.

The e-services portal

A separate application-management portal for the centre's e-services operations, with its own frontend and backend repositories, handling the internal side of service applications.

Decisions worth noting

- ◆ **Structure follows search intent.** The URL and page structure mirror how residents actually look for these services, which is the whole SEO strategy for a multi-branch local business, and it costs nothing in engineering compared to one generic services page.
- ◆ **Right-sized backend.** It would have been easy to build a heavy booking engine. The client needed leads and content, so that is what the backend does well, and nothing else.

Why it is in the portfolio

It is an example of matching the architecture to the business rather than the other way round: a content-and-location-heavy frontend for a local-search-driven business, with a backend that is exactly as complex as the problem. It also touches a regulated domain (government-services intermediation) where careful handling of enquiry data matters.