1) Curate + “tell the site where to place imagery”

Create a tiny imagery manifest that maps each file to its job, focal point, and tags. Put your Lenovo images in public/images/ with the same names as in your screenshot, then add:

/lib/imagery-manifest.ts

export type ImgRole = "hero" | "section" | "badge" | "avatar" | "sprite" | "gallery";

export type ImgMeta = {

id: string;

src: string; // public path

alt: string;

role: ImgRole;

aspect?: string; // e.g. "16/9", "4/3", "1/1"

focal?: [number, number]; // 0..1, x/y focal point for objectPosition

tags?: string[];

};

export const IMAGES: ImgMeta[] = [

{ id:"reach\_for\_the\_stars", src:"/images/reach\_for\_the\_stars.png",

alt:"Silhouette reaching toward a luminous starfield.",

role:"hero", aspect:"16/9", focal:[0.45,0.35], tags:["space","cta"] },

{ id:"globe\_3d\_with\_ravens", src:"/images/globe\_3d\_with\_ravens.png",

alt:"3D earth with stylized ravens in orbit.", role:"section", aspect:"16/9",

focal:[0.56,0.42], tags:["globe","brand","motion"] },

{ id:"ai\_construction\_bridge\_banner", src:"/images/ai\_construction\_bridge\_banner.png",

alt:"Bridge construction scene augmented by AI overlays.",

role:"section", aspect:"21/9", focal:[0.6,0.5], tags:["industry","engineering"] },

{ id:"AI\_DNA", src:"/images/AI\_DNA.png", alt:"Abstract double helix made of circuitry.",

role:"section", aspect:"4/3", focal:[0.5,0.4], tags:["biology","innovation"] },

{ id:"human\_AI\_space", src:"/images/huma\_AI-space.jpg",

alt:"Human hand meeting a cosmic energy trail.", role:"section",

aspect:"16/9", focal:[0.35,0.5], tags:["vision"] },

{ id:"data\_sovereignty\_badge", src:"/images/data\_sovereignty\_badge.png",

alt:"Data sovereignty badge.", role:"badge", aspect:"1/1", tags:["trust"] },

{ id:"compliance\_shield", src:"/images/compliance\_shield\_premium.png",

alt:"Premium compliance shield emblem.", role:"badge", aspect:"1/1", tags:["compliance"] },

{ id:"michael\_headshot", src:"/images/michael\_howardbio.jpeg",

alt:"Michael Howard, founder portrait.", role:"avatar", aspect:"1/1", tags:["about"] },

{ id:"raven\_huginn", src:"/images/raven\_huginn.png",

alt:"Raven crest—Huginn.", role:"sprite", aspect:"1/1", tags:["brand","sprite"] },

{ id:"raven\_muninn", src:"/images/raven\_muninn.png",

alt:"Raven crest—Muninn.", role:"sprite", aspect:"1/1", tags:["brand","sprite"] },

// Gallery defaults: anything not explicitly placed goes in the gallery

{ id:"ai\_technology", src:"/images/ai\_technology.jpeg",

alt:"Futuristic circuit board and AI interface.", role:"gallery", aspect:"3/2" },

{ id:"ireland\_landscape", src:"/images/ireland\_landscape.jpg",

alt:"Green Irish landscape with bright skies.", role:"gallery", aspect:"3/2" },

{ id:"earth\_globe\_realistic", src:"/images/earth\_globe\_realistic.png",

alt:"Realistic Earth globe on black.", role:"gallery", aspect:"1/1" },

{ id:"ELevation", src:"/images/ELevation.jpg",

alt:"Climber ascending toward sunlight.", role:"gallery", aspect:"3/2" },

];

This single file answers “what goes where” and lets components pull exactly what they need (role: "hero", "section", etc.). It also encodes focal points so text never overlaps the wrong area.

2) The <Img> helper (art-direction + perf)

/components/Img.tsx

"use client";

import Image from "next/image";

import { IMAGES } from "@/lib/imagery-manifest";

type Props = {

assetId: string;

className?: string;

priority?: boolean;

sizes?: string; // e.g. "(max-width: 768px) 100vw, 1200px"

rounded?: boolean;

};

export default function Img({ assetId, className="", priority=false, sizes="100vw", rounded=true }: Props) {

const meta = IMAGES.find(i => i.id === assetId);

if (!meta) return null;

const [fx, fy] = meta.focal ?? [0.5, 0.5];

const objectPosition = `${Math.round(fx\*100)}% ${Math.round(fy\*100)}%`;

const radius = rounded ? "rounded-2xl" : "";

return (

<Image

src={meta.src}

alt={meta.alt}

fill

priority={priority}

sizes={sizes}

className={`object-cover ${radius} ${className}`}

style={{ objectPosition }}

/>

);

}

3) Cinematic hero with (optional) ravens

This gives you layered depth, safe performance, and a clean fallback when “Reduce motion” is enabled.

/components/CinematicHero.tsx

"use client";

import { useEffect, useState } from "react";

import { motion, useReducedMotion } from "framer-motion";

import Img from "@/components/Img";

import { flags } from "@/feature-flags"; // globe/news flags model; add raven if you like

export default function CinematicHero() {

const prefersReduced = useReducedMotion();

const [mounted, setMounted] = useState(false);

useEffect(()=>setMounted(true), []);

return (

<section className="relative h-[70vh] min-h-[540px] w-full overflow-hidden">

{/\* Background image with gradient for text contrast \*/}

<div className="absolute inset-0">

<div className="absolute inset-0 bg-gradient-to-t from-black/60 via-black/30 to-transparent z-10" />

<div className="absolute inset-0">

<Img assetId="reach\_for\_the\_stars" priority sizes="(max-width: 1024px) 100vw, 1400px" />

</div>

</div>

{/\* Headline \*/}

<div className="relative z-20 max-w-4xl px-6 md:px-10 top-[18%]">

<h1 className="text-white text-4xl md:text-6xl font-semibold leading-tight drop-shadow">

Intelligence that builds, protects, and scales your world.

</h1>

<p className="mt-4 text-white/90 md:text-lg">

IntegAI • data sovereignty • compliance by design.

</p>

</div>

{/\* Subtle floating particles (perf-safe) \*/}

{!prefersReduced && mounted && (

<motion.div

aria-hidden

className="pointer-events-none absolute inset-0 z-10"

initial={{ opacity: 0 }} animate={{ opacity: 1 }} transition={{ duration: 1.2 }}

>

<div className="absolute left-10 top-16 w-1 h-1 rounded-full bg-white/60 blur-[2px]" />

<div className="absolute right-24 top-40 w-1 h-1 rounded-full bg-white/50 blur-[2px]" />

</motion.div>

)}

{/\* Optional raven sprites (very light). Gate behind a flag if you want. \*/}

{!prefersReduced && mounted && (

<div aria-hidden className="absolute inset-0 z-20">

<motion.img

src="/images/raven\_huginn.png" alt="" className="absolute w-24 md:w-32 opacity-80"

initial={{ x: -80, y: 40, rotate: -8 }} animate={{ x: ["-10%", "110%"], y: [30, -10, 50], rotate: [-6, 4, -3] }}

transition={{ duration: 16, repeat: Infinity, ease: "easeInOut" }}

/>

<motion.img

src="/images/raven\_muninn.png" alt="" className="absolute w-20 md:w-28 opacity-70"

initial={{ right: -80, top: 120, rotate: 6 }}

animate={{ right: ["-10%", "110%"], top: [120, 80, 160], rotate: [4, -5, 3] }}

transition={{ duration: 20, repeat: Infinity, ease: "easeInOut", delay: 3 }}

/>

</div>

)}

</section>

);

}

Where to use:

// app/page.tsx

import CinematicHero from "@/components/CinematicHero";

export default function Home() {

return (

<main className="relative">

<CinematicHero />

{/\* Sections below pull “section” images \*/}

{/\* Example: \*/}

<Section

title="Data Sovereignty"

imageId="data\_sovereignty\_badge"

copy="Your data. Your borders. Your rules."

/>

</main>

);

}

Notes on the ravens

Keep them subtle and transform-only (no heavy filters) to protect LCP & battery.

If you want more realism later, swap the PNGs for a Lottie export or a lightweight sprite sheet—still cheaper than a 3D bird.

If performance dips, gate with NEXT\_PUBLIC\_FLAG\_RAVENS=0 and you’re back to static in one env var.

4) “Maximum effect” placement guide (using your current files)

Hero: reach\_for\_the\_stars.png (already focal’d), headline left/top; gradient overlay for legibility.

Trust row: data\_sovereignty\_badge.png, compliance\_shield\_premium.png, integai\_logo.png as badges on dark card backgrounds.

Industry/Use-case bands:

“Engineering & Capital Projects” → ai\_construction\_bridge\_banner.png

“Healthcare & Life Sciences” → huma\_AI-space.jpg or AI\_DNA.png

About: michael\_howardbio.jpeg + ireland\_landscape.jpg (use landscape as a soft background with 20–30% overlay).

Tech stack / Build with us: next.svg, vercel.svg, globe.svg as crisp, monochrome icons.

General rules:

Add focal for any image with subject off-center; set headline on the opposite side of the focal.

Always place a black → transparent gradient over busy images behind text (from-black/60).

Keep section imagery wide (16/9 or 21/9) to avoid layout jumps; square for badges/avatars.

Prefer one dominant visual per fold.

5) Gallery page (fast, accessible, keyboard-friendly)

/app/gallery/page.tsx

import { IMAGES } from "@/lib/imagery-manifest";

import Link from "next/link";

export default function Gallery() {

const items = IMAGES.filter(i => i.role === "gallery" || i.role === "section");

return (

<div className="mx-auto max-w-6xl px-6 py-10">

<h1 className="text-3xl md:text-4xl font-semibold mb-6">Image Gallery</h1>

<ul className="grid grid-cols-2 md:grid-cols-3 gap-4" role="list">

{items.map(img => (

<li key={img.id} className="group relative aspect-[4/3] overflow-hidden rounded-xl">

<Link href={`/gallery/${img.id}`} className="focus:outline-none focus:ring-2 focus:ring-white/80">

{/\* Using native <img> for faster grids; detail page can use <Image> \*/}

<img src={img.src} alt={img.alt} className="h-full w-full object-cover" />

<div className="absolute inset-0 bg-black/0 group-hover:bg-black/20 transition-colors" />

<span className="sr-only">{img.alt}</span>

</Link>

</li>

))}

</ul>

</div>

);

}

/app/gallery/[id]/page.tsx

import Image from "next/image";

import { notFound } from "next/navigation";

import { IMAGES } from "@/lib/imagery-manifest";

export default function GalleryItem({ params }: { params: { id: string } }) {

const img = IMAGES.find(i => i.id === params.id);

if (!img) return notFound();

return (

<article className="mx-auto max-w-5xl px-6 py-10">

<div className="relative w-full aspect-[16/9] overflow-hidden rounded-2xl">

<Image src={img.src} alt={img.alt} fill sizes="(max-width: 1024px) 100vw, 1200px" className="object-contain bg-black" />

</div>

<h1 className="mt-6 text-2xl font-semibold">{img.alt}</h1>

<p className="text-sm text-neutral-500 mt-2">Tags: {(img.tags ?? []).join(", ") || "untagged"}</p>

</article>

);

}

6) Quick alt-text pass for your filenames (copy these into the manifest)

globe\_3d\_with\_ravens.png: “3D earth with stylized ravens in orbit.”

earth\_globe\_realistic.png: “Realistic Earth globe on a black background.”

ai\_construction\_bridge\_banner.png: “Bridge construction scene with AI overlays.”

AI\_DNA.png: “Abstract double helix made of circuitry.”

ai\_technology.jpeg: “Futuristic circuit board and AI interface.”

huma\_AI-space.jpg: “Human hand meeting a stream of cosmic light.”

ireland\_landscape.jpg: “Green Irish landscape with bright skies.”

reach\_for\_the\_stars.png: “Silhouette reaching toward a luminous starfield.”

raven\_huginn.png: “Raven crest—Huginn.”

raven\_muninn.png: “Raven crest—Muninn.”

7) Importing your Lenovo folder fast

From repo root (Linux/WSL), assuming your files are at /home/michael/Public/images/:

mkdir -p public/images

rsync -av --progress /home/michael/Public/images/ public/images/

# (optional) run your Sharp pipeline if you’ve set it up; otherwise Next/Image will still optimize