


```

{ id:4, title:"OpenClaw", type:"TRADEMARK", status:"FILED" };
];

const DEPLOYMENTS = [
  { version:"1.2.0", env:"PRODUCTION", status:"LIVE", url:"hydra.openclaw.ai", },
  { version:"1.1.3", env:"STAGING", status:"LIVE", url:"staging.openclaw.ai" },
  { version:"1.0.8", env:"PRODUCTION", status:"ARCHIVE", url:"v1.openclaw.ai", },
];
;

const GRANTS = [
  { name:"NSF SBIR Phase I", amount:275000, status:"SUBMITTED", color:C.blue },
  { name:"Argonne CRI", amount:250000, status:"PENDING", color:C.purple },
  { name:"Illinois Innovation", amount:50000, status:"DRAFTING", color:C.green },
  { name:"SBA Microloan", amount:50000, status:"APPROVED", color:C.green },
  { name:"Mozilla Foundation AI", amount:200000, status:"PENDING", color:C.orange },
];
;

const genPnL = (n, base, vol) => Array.from({length:n}, (_,i)=>({
  t:i, v: base + Math.sin(i/3)*vol + (Math.random()-0.4)*vol*0.5
}));

const genActivity = () => Array.from({length:24}, (_,i)=>({
  h:`${i}:00`, calls: Math.floor(Math.random()*800+200), errors: Math.floor(Math.random()*200)
}));

// — Components ——————
;

const Tag = ({label, color, bg}) => (
  <span style={{background:bg||"rgba(0,136,255,0.15)", color:color||C.blue, fontStyle:"italic", padding:"2px 7px", borderRadius:3, letterSpacing:"0.08em", fontFamily:"JetBrains Mono", width:100, height:30, border:1, borderStyle:"solid", borderColor:color, overflow:"hidden", position:"relative", top:0, left:0}}>
    {label}
  </span>
);

const Stat = ({label, value, sub, color=C.text}) => (
  <div style={{textAlign:"center"}}>
    <div style={{fontSize:22, fontWeight:800, color, fontFamily:"JetBrains Mono", width:100, height:30, border:1, borderStyle:"solid", borderColor:color, overflow:"hidden", position:"relative", top:0, left:0}}>
      <div style={{fontSize:11, color:C.textDim, marginTop:2}}>{label}</div>
      {sub && <div style={{fontSize:10, color:C.textFaint, marginTop:1}}>{sub}</div>}
    </div>
  </div>
);

const ProgressBar = ({pct, color=C.blue, h=4}) => (
  <div style={{background:C.textFaint, borderRadius:h, overflow:"hidden", height:30}}>
    <div style={{width:`${pct}%`, background:color, height:"100%", borderRadius:h, transition:"width 1s ease", boxShadow:`0 0 6px ${color}`}} />
  </div>
);
;
```

```

);

const StatusDot = ({status}) => {
  const c = status==="ACTIVE"? "LIVE":status==="WARNING"? "WARN":status==="IDLE"? "I
  const col = {LIVE:C.green, WARN:C.orange, IDLE:C.textDim, OFF:C.red}[c];
  return (
    <span style={{display:"inline-flex",alignItems:"center",gap:5}}>
      <span style={{width:7,height:7,borderRadius:"50%",background:col,boxShadow:
        <span style={{fontSize:10,color:col,fontWeight:700,fontFamily:"monospace"}}>
      </span>
    ) ;
  );
}

const Card = ({children, style={}}) => (
  <div style={{background:C.card, border:`1px solid ${C.border}`, borderRadius:8,
    ...style}}>
    {children}
  </div>
);

const SectionTitle = ({icon, title, sub}) => (
  <div style={{display:"flex", alignItems:"center", gap:10, marginBottom:14}}>
    <span style={{fontSize:18}}>{icon}</span>
    <div>
      <div style={{fontSize:13, fontWeight:700, color:C.text, letterSpacing:"0.05
        {sub && <div style={{fontSize:10, color:C.textDim}}>{sub}</div>}
      </div>
    </div>
  );
}

// — Tab: SWARM ——————
function SwarmTab() {
  const [tick, setTick] = useState(0);
  const pnlData = useRef(genPnL(60, 6800, 300)).current;
  useEffect(()=>{const t=setInterval(()=>setTick(x=>x+1),2000);return()=>clearInt

  const totalPnL = AGENTS.reduce((a,x)=>a+x.pnl,0);
  const active = AGENTS.filter(a=>a.status==="ACTIVE").length;

  return (
    <div style={{display:"grid", gap:14}}>
      /* Top stats */
      <div style={{display:"grid", gridTemplateColumns:"repeat(5,1fr)", gap:10}}>
        [
          {label:"Total Agents", value:AGENTS.length, color:C.blue},
          {label:"Active", value:active, color:C.green},
          {label:"Portfolio P&L", value:`+$${totalPnL.toLocaleString()}`, color:C

```

```

        {label:"Avg Reputation", value:`${(AGENTS.reduce((a,x)=>a+x.rep,0)/AGEN
        {label:"Total Tasks", value:AGENTS.reduce((a,x)=>a+x.tasks,0).toLocaleS
    ].map((s,i)=>(
        <Card key={i} style={{padding:14, textAlign:"center"}}>
            <Stat {...s}/>
        </Card>
    )))
</div>

<div style={{display:"grid", gridTemplateColumns:"1fr 340px", gap:14}}>
    /* Agent Table */
    <Card>
        <SectionTitle icon="🧠" title="Agent Registry" sub="MACNN-RLS Hierarchy
        <div style={{overflowX:"auto"}}>
            <table style={{width:"100%", borderCollapse:"collapse", fontSize:12}}>
                <thead>
                    <tr style={{borderBottom:`1px solid ${C.border}`}}>
                        {"ID","Name","Tier","Status","Rep","P&L","Tasks","Load","Syste
                            <th key={h} style={{textAlign:"left", padding:"6px 8px", colo
                        )))
                    </tr>
                </thead>
                <tbody>
                    {AGENTS.map(a=>(
                        <tr key={a.id} style={{borderBottom:`1px solid ${C.textFaint}`,
                            onMouseEnter={e=>e.currentTarget.style.background=C.bg3}
                            onMouseLeave={e=>e.currentTarget.style.background="transparen
                            <td style={{padding:"7px 8px", fontFamily:"monospace", fontSi
                            <td style={{padding:"7px 8px", color:a.color, fontWeight:600}
                            <td style={{padding:"7px 8px"}}><Tag label={a.tier} color={a.
                            <td style={{padding:"7px 8px"}}><StatusDot status={a.status}/
                            <td style={{padding:"7px 8px", fontFamily:"monospace", color:
                            <td style={{padding:"7px 8px", fontFamily:"monospace", color:
                                {a.pnl>=0?"+"":""}{a.pnl==0?"-":"$"+a.pnl.toLocaleString()
                            </td>
                            <td style={{padding:"7px 8px", fontFamily:"monospace", color:
                            <td style={{padding:"7px 8px", minWidth:80}}>
                                <div style={{display:"flex", alignItems:"center", gap:6}}>
                                    <ProgressBar pct={a.load} color={a.load>85?C.orange:C.blu
                                        <span style={{fontSize:10, color:C.textDim, minWidth:28}}
                                    </div>
                            </td>
                            <td style={{padding:"7px 8px"}}><Tag label={a.sys} color={a.s
                        </tr>
                    )))
                </tbody>
            </table>
        </Card>
    </div>

```

```

        </div>
    </Card>

    /* P&L chart + OMEGA ROOT */
    <div style={{display:"grid", gap:14}}>
        <Card>
            <SectionTitle icon="📈" title="Portfolio P&L" sub="Live - 60 ticks"/>
            <ResponsiveContainer width="100%" height={130}>
                <AreaChart data={pnlData}>
                    <defs>
                        <linearGradient id="pnlGrad" x1="0" y1="0" x2="0" y2="1">
                            <stop offset="5%" stopColor={C.green} stopOpacity={0.3}/>
                            <stop offset="95%" stopColor={C.green} stopOpacity={0}/>
                        </linearGradient>
                    </defs>
                    <XAxis hide/>
                    <YAxis hide domain={[ "auto", "auto" ]}/>
                    <Tooltip contentStyle={{background:C.card,border:`1px solid ${C.b`>
                        <Area type="monotone" dataKey="v" stroke={C.green} fill="url(#pnl
                    </AreaChart>
                </ResponsiveContainer>
            </Card>
        <Card>
            <SectionTitle icon="🔒" title="OMEGA ROOT" sub="Governance Kernel - C
        {[

            {label:"Max Position", value:"2.0%", ok:true},
            {label:"Risk/Reward Min", value:"2.0:1", ok:true},
            {label:"Daily Loss Ceiling", value:"10.0%", ok:true},
            {label:"Leverage", value:"DISABLED", ok:true},
            {label:"Human Approval", value:"REQUIRED", ok:true},
            {label:"Wolfpack Threshold", value:"95%", ok:true},
        ].map((r,i)=>(
            <div key={i} style={{display:"flex", justifyContent:"space-between"
                padding:"5px 0", borderBottom:i<5?`1px solid ${C.textFaint}`:"none`>
                <span style={{fontSize:11, color:C.textDim}}>{r.label}</span>
                <span style={{fontSize:11, fontFamily:"monospace", color:r.ok?C.g
            </div>
        )));
    </Card>
</div>
</div>
</div>
);

}

// — Tab: RESEARCH —————

```

```

function ResearchTab() {
  const [query, setQuery] = useState("");
  const [sources, setSources] = useState("ALL_WEB");
  const [domains, setDomains] = useState("arxiv.org, ssrn.com, sec.gov, pubmed.ncbi");
  const [running, setRunning] = useState(false);
  const [log, setLog] = useState([]);
  const [files, setFiles] = useState([
    {name:"MACNN-RLS-v3.0.pdf", type:"PDF", size:"2.4MB", status:"INDEXED"},
    {name:"OpenClaw-Architecture.pptx", type:"PPT", size:"8.1MB", status:"INDEXED"},
    {name:"HydraPerformance-Q1-2025.xlsx", type:"XLS", size:"1.2MB", status:"INDEXED"}
  ]);
  const actData = useRef(genActivity()).current;

  const STEPS = [
    "🔍 Decomposing query into sub-tasks...",
    "🌐 Dispatching parallel web agents...",
    "📄 Retrieving documents from trusted domains...",
    "🔗 Cross-referencing sources (Neural Context Protocol)...",
    "🧠 System 2 deep analysis initiated...",
    "⚖️ OMEGA ROOT compliance check...",
    "📊 Synthesizing multi-source output...",
    "✅ Synthesized report ready."
  ];

  const runResearch = () => {
    if (!query.trim()) return;
    setRunning(true);
    setLog([]);
    STEPS.forEach((s,i)=>setTimeout(()=>{
      setLog(l=>[...l, s]);
      if(i==STEPS.length-1) setRunning(false);
    }, (i+1)*700));
  };

  return (
    <div style={{display:"grid", gap:14}}>
      <div style={{display:"grid", gridTemplateColumns:"1fr 340px", gap:14}}>
        {/* Research Agent Panel */}
        <Card>
          <SectionTitle icon="🤖" title="Autonomous Research Agent" sub="Multi-science"
            <div style={{display:"grid", gridTemplateColumns:"1fr 1fr", gap:10, margin:10}}>
              <div>
                <div style={{fontSize:10, color:C.textDim, marginBottom:5, textTransform:'uppercase'}}>
                  {"ALL_WEB", "SPECIFIC_DOMAINS", "PRIORITIZED_SITES"].map(m=>
                    <label key={m} style={{display:"flex", alignItems:"center", gap:8}}>
                      <div onClick={()=>setSources(m)} style={{width:14, height:14, border:1, border-radius:50%, padding:2px}}>
                        {m[0]} {m[1]}

```

```

        border: `2px solid ${sources==m?C.blue:C.border}`,
        background: sources==m?C.blue:"transparent", cursor:"pointer"
        boxShadow: sources==m?`0 0 8px ${C.blue}`:"none"})}/>
      <span style={{fontSize:11, color:sources==m?C.blue:C.textDim}}>
    </label>
  )})
</div>
{sources!="ALL_WEB" && (
  <div>
    <div style={{fontSize:10, color:C.textDim, marginBottom:5, textTr
      {sources=="SPECIFIC_DOMAINS"?`Trusted Domains`:"Prioritized Si
    </div>
    <textarea value={domains} onChange={e=>setDomains(e.target.value)
      style={{width:"100%", background:C.bg3, border:`1px solid ${C.b
        color:C.text, fontSize:11, fontFamily:"monospace", padding:8,
        outline:"none", boxSizing:"border-box"}}}/>
    </div>
  )}
</div>

<div style={{display:"flex", gap:8, marginBottom:12}}>
  <input value={query} onChange={e=>setQuery(e.target.value)}>
    onKeyDown={e=>e.key==="Enter"&&runResearch()}
    placeholder="Research NSF SBIR Phase I acceptance rates for AI gove
    style={{flex:1, background:C.bg3, border:`1px solid ${C.border}`, b
      color:C.text, fontSize:12, padding:"9px 12px", outline:"none",
      fontFamily:"inherit"}}/>
  <button onClick={runResearch} disabled={running}>
    style={{padding:"9px 18px", background:running?C.bg3:C.blue, color:
      border:"none", borderRadius:4, fontWeight:700, fontSize:12, curso
      opacity:running?0.6:1}}>
      {running?"Running...":"Research"}
    </button>
</div>

/* Agent Log */
<div style={{background:C.bg, border:`1px solid ${C.border}`}, borderRad
  {log.length==0 && (
    <div style={{color:C.textFaint, paddingTop:60, textAlign:"center"}}>
      Agent idle. Enter a query to begin multi-source synthesis.
    </div>
  )}
  {log.map((l,i)=>(
    <div key={i} style={{color:i==log.length-1?C.green:C.textDim, padd
      display:"flex", gap:8, alignItems:"flex-start"}}>
      <span style={{color:C.textFaint, minWidth:60}}>`[${String(i*0.7)
        <span>{l}</span>

```

```

        </div>
    ) )
    {running && <span style={{color:C.blue, animation:"blink 1s infinite"
</div>

<div style={{marginTop:10, display:"flex", gap:6, flexWrap:"wrap"}}>
  {[ "NSF SBIR eligibility AI", "Argonne CRI 2025 cohort", "Illinois DCEO
    <button key={s} onClick={()=>{setQuery(s);}}
      style={{fontSize:10, padding:"4px 8px", background:C.bg3, border:
        color:C.textDim, borderRadius:3, cursor:"pointer"}}>
        {s}
      </button>
    ) )
  </div>
</Card>

/* Context Grounding */
<div style={{display:"grid", gap:14, alignContent:"start"}}>
  <Card>
    <SectionTitle icon="📎" title="First-Class Context" sub="Grounded dat
    {files.map((f,i)=>(
      <div key={i} style={{display:"flex", justifyContent:"space-between"
        padding:"7px 0", borderBottom:i<files.length-1?`1px solid ${C.tex
        <div style={{display:"flex", gap:8, alignItems:"center"}}>
          <span style={{fontSize:16}}>
            {f.type==="PDF"?"📄":f.type==="PPT"?"📊":"
          </span>
          <div>
            <div style={{fontSize:11, color:C.text}}>{f.name}</div>
            <div style={{fontSize:10, color:C.textDim}}>{f.size}</div>
          </div>
        </div>
        <Tag label={f.status} color={C.green} bg={C.greenBg}/>
      </div>
    ) )
    <div style={{marginTop:10}}>
      <label style={{display:"flex", alignItems:"center", justifyContent:
        padding:"8px 12px", background:C.bg3, border:`1px dashed ${C.bord
        borderRadius:4, cursor:"pointer", color:C.textDim, fontSize:11}}>
        <span>+ Add context file (PDF, PPT, XLS)</span>
      </label>
    </div>
  </Card>

<Card>
  <SectionTitle icon="📊" title="Research Activity" sub="24h API calls"
  <ResponsiveContainer width="100%" height={120}>

```

```

        <BarChart data={actData}>
            <XAxis dataKey="h" hide/>
            <YAxis hide/>
            <Tooltip contentStyle={{background:C.card,border:'1px solid ${C.b
                <Bar dataKey="calls" fill={C.blue} opacity={0.7} radius={[2,2,0,0
            </BarChart>
        </ResponsiveContainer>
    </Card>

    <Card>
        <SectionTitle icon="🎤" title="Voice Interface" sub="Siri → SmythOS W
        <div style={{fontSize:11, color:C.textDim, lineHeight:1.6, marginBott
            Trigger research tasks with Siri via webhook integration. "Hey Siri
        </div>
        <div style={{background:C.bg, borderRadius:4, padding:8, fontFamily:"
            POST https://openclaw.ai/api/v1/research<br/>
            {'query": "{{SiriInput}}", "sources": "SPECIFIC_DOMAINS" }`}
        </div>
        <Tag label="WEBHOOK ACTIVE" color={C.green} bg={C.greenBg}/>
    </Card>
    </div>
    </div>
    </div>
)
}

// — Tab: CLAWHUB ——————
function ClawHubTab() {
    const [filter, setFilter] = useState("ALL");
    const TYPES = ["ALL","SKILL","SOUL","MEMORY"];
    const filtered = filter==="ALL"?SKILLS:SKILLS.filter(s=>s.type==filter);

    return (
        <div style={{display:"grid", gap:14}}>
            <div style={{display:"grid", gridTemplateColumns:"repeat(4,1fr)", gap:10}}>
                [
                    {label:"Published Packages", value:SKILLS.length, color:C.blue},
                    {label:"Total Installs", value:SKILLS.reduce((a,x)=>a+x.installs,0).toL
                    {label:"Revenue (CLAW)", value:"12,847", color:C.gold},
                    {label:"Verified Packages", value:SKILLS.filter(s=>s.verified).length,
                ].map((s,i)=><Card key={i} style={{padding:14, textAlign:"center"}}><Stat
            </div>

            <Card>
                <div style={{display:"flex", justifyContent:"space-between", alignItems:"
                    <SectionTitle icon="👾" title="ClawHub Marketplace" sub="SKILL · SOUL ·
                    <div style={{display:"flex", gap:6}}>

```

```

{TYPES.map(t=>(
  <button key={t} onClick={()=>setFilter(t)}
    style={{fontSize:10, padding:"4px 10px", borderRadius:3, fontWeig
      background: filter==t?C.blue:C.bg3, color:filter==t?C.white:C
      border:`1px solid ${filter==t?C.blue:C.border}`, cursor:"point
      {t}
    </button>
  ))}
</div>
</div>
<div style={{display:"grid", gridTemplateColumns:"repeat(3,1fr)", gap:10}
  {filtered.map(s=>(
    <div key={s.id} style={{background:C.bg3, border:`1px solid ${C.borde
      cursor:"pointer", transition:"border-color 0.2s"}}
      onMouseEnter={e=>e.currentTarget.style.borderColor=C.blue}
      onMouseLeave={e=>e.currentTarget.style.borderColor=C.border}>
      <div style={{display:"flex", justifyContent:"space-between", margin
        <Tag label={s.type} color={s.type==="SKILL"?C.blue:s.type==="SOUL
          bg={s.type==="SKILL"?C.blueBg:s.type==="SOUL"?C.goldBg:C.purple
          {s.verified && <Tag label="✓ VERIFIED" color={C.green} bg={C.gree
        </div>
        <div style={{fontSize:13, fontWeight:700, color:C.text, marginBotto
        <div style={{fontSize:10, color:C.textDim, fontFamily:"monospace",
        <div style={{display:"flex", justifyContent:"space-between", alignI
          <span style={{fontSize:12, fontWeight:700, color:s.price==0?C.gr
            {s.price==0?"FREE":`${s.price} CLAW`}
          </span>
          <span style={{fontSize:10, color:C.textDim}}>★ {s.rating} · {s.in
        </div>
        <div style={{marginTop:10}}>
          <ProgressBar pct={Math.min(100, s.installs/25)} color={s.price==0?C
        </div>
      </div>
    ))}
  </div>
</Card>

/* CLAW token stats */
<div style={{display:"grid", gridTemplateColumns:"1fr 1fr", gap:14}}>
  <Card>
    <SectionTitle icon="🌐" title="CLAW Token Allocation" sub="1,000,000,00
    {[{"label":"Ecosystem Fund", "pct:30, color:C.blue}, {"label":"Community/ClawHub", "pct:25, color:C.green}, {"label":"R&D Treasury", "pct:20, color:C.purple}, {"label":"Team (4yr vest)", "pct:15, color:C.gold}, {"label":"Public Sale", "pct:10, color:C.orange}, {"label":"Reserves", "pct:10, color:C.purple}, {"label":"Marketing", "pct:5, color:C.purple}, {"label":"Operational Costs", "pct:5, color:C.purple}, {"label":"Strategic Partnerships", "pct:5, color:C.purple}, {"label":"Future Expansion", "pct:5, color:C.purple}], [{"label": "Total Allocation", "value": 100, "color": "#007bff"}]}

```

```

    ].map((r,i)=>(
      <div key={i} style={{marginBottom:10}}>
        <div style={{display:"flex",justifyContent:"space-between",marginBo
          <span style={{fontSize:11,color:C.textDim}}>r.label</span>
          <span style={{fontSize:11,color:r.color,fontFamily:"monospace",fo
        </div>
        <ProgressBar pct={r.pct} color={r.color}/>
      </div>
    ))}
  </Card>
<Card>
  <SectionTitle icon="📋" title="Smart Contracts" sub="Verified on Ethereum"
  {[

    {name:"CLAWToken.sol",           addr:"0xa1b2...c3d4", status:"VERIFI
    {name:"OMEGARootKernel.sol",     addr:"0xe5f6...g7h8", status:"VERIFI
    {name:"BlackBoxRecorder.sol",    addr:"0xi9j0...k1l2", status:"VERIFI
    {name:"ClawHubMarketplace.sol",  addr:"0xm3n4...o5p6", status:"PENDING"
    {name:"OpenClawIPRegistry.sol",   addr:"0xq7r8...s9t0", status:"PENDING"
    {name:"OpenClawAgentRegistry.sol",addr:"0xu1v2...w3x4",status:"DEPLOY
  ].map((c,i)=>(
    <div key={i} style={{display:"flex", justifyContent:"space-between",
      padding:"6px 0", borderBottom:i<5?`1px solid ${C.textFaint}`:"none"
      <div>
        <div style={{fontSize:11, color:C.text, fontFamily:"monospace"}}>
          <div style={{fontSize:10, color:C.textDim, fontFamily:"monospace"
            </div>
            <Tag label={c.status} color={c.color}/>
          </div>
        ))}
      </div>
    </div>
  </div>
)
);
}

// — Tab: DEPLOY ——————
function DeployTab() {
  const [version, setVersion] = useState("1.3.0");
  const [notes, setNotes] = useState("");
  const [env, setEnv] = useState("STAGING");
  const [deploying, setDeploying] = useState(false);
  const [deployLog, setDeployLog] = useState([]);

  const DEPLOY_STEPS = [
    "📦 Building OpenClaw runtime bundle...",
    "🔒 Signing release with OMEGA ROOT kernel hash...",
    "🌐 Provisioning subdomain + SSL certificate...",
  ]

```

```

    "🐳 Initializing Docker Compose swarm containers...",
    "🩺 Running health checks across all agents...",
    "📝 Writing immutable deployment record to chain...",
    "✅ Version "+version+" deployed successfully.",
];

const deploy = () => {
  setDeploying(true);
  setDeployLog([]);
  DEPLOY_STEPS.forEach((s,i)=>setTimeout(()=>{
    setDeployLog(l=>[...l,s]);
    if(i===DEPLOY_STEPS.length-1) setDeploying(false);
  },(i+1)*800));
};

return (
  <div style={{display:"grid", gap:14}}>
    <div style={{display:"grid", gridTemplateColumns:"1fr 1fr", gap:14}}>
      {/* Deploy config */}
      <Card>
        <SectionTitle icon="📝" title="Deploy OpenClaw" sub="SmythOS Agent Clou
        <div style={{display:"grid", gap:12}}>
          <div>
            <div style={{fontSize:10,color:C.textDim,marginBottom:5,textTransfo
            <input value={version} onChange={e=>setVersion(e.target.value)}
              style={{width:"100%", background:C.bg3, border:`1px solid ${C.bor
                color:C.text, fontSize:13, padding:"8px 12px", outline:"none",
                <div style={{fontSize:10, color:C.textFaint, marginTop:4}}>Once pub
              </div>
              <div>
                <div style={{fontSize:10,color:C.textDim,marginBottom:5,textTransfo
                <div style={{display:"flex", gap:8}}>
                  {"STAGING","PRODUCTION","LOCAL"].map(e=>(
                    <button key={e} onClick={()=>setEnv(e)}
                      style={{flex:1, padding:"7px 0", borderRadius:4, fontWeight:7
                        background: env==e?(e=="PRODUCTION"?C.orange:C.blue):C.bg
                        color: env==e?C.white:C.textDim,
                        border:`1px solid ${env==e?(e=="PRODUCTION"?C.orange:C.bl
                      {e}
                    </button>
                  )));
                </div>
              </div>
              <div>
                <div style={{fontSize:10,color:C.textDim,marginBottom:5,textTransfo
                <div style={{background:C.bg, border:`1px solid ${C.border}`, borde
                  fontFamily:"monospace", fontSize:12, color:C.blue}}>

```

```

        {env==="PRODUCTION"? "hydra.openclaw.ai": env==="STAGING"? "staging.
      </div>
    </div>
    <div>
      <div style={{fontSize:10,color:C.textDim,marginBottom:5,textTransfo
      <textarea value={notes} onChange={e=>setNotes(e.target.value)}
      placeholder="e.g., Adds ClawHub ERC-1155 marketplace integration.
      style={{width:"100%", background:C.bg3, border:`1px solid ${C.bor
      color:C.text, fontSize:12, padding:"8px 12px", resize:"none", h
      outline:"none", fontFamily:"inherit", boxSizing:"border-box"}}/
    </div>
    <div>
      <div style={{fontSize:10,color:C.textDim,marginBottom:5,textTransfo
      <div style={{background:C.bg, border:`1px solid ${C.border}`}, borde
      fontFamily:"monospace", fontSize:10, color:C.textDim}}>
      {env==="PRODUCTION"
        ?"https://openclaw.agent.psatge.smyth.ai/mcp/sse"
        :"https://openclaw.agent.stage.smyth.ai/mcp/sse"}
      </div>
    </div>
    <button onClick={deploy} disabled={deploying}
      style={{padding:"12px 0", background:deploying?C.bg3:env==="PRODUCT
      color:C.white, border:"none", borderRadius:4, fontWeight:800, fon
      cursor:deploying?"not-allowed":"pointer", opacity:deploying?0.6:1
      boxShadow:deploying?"none":`0 0 20px ${env==="PRODUCTION"?C.orang
      {deploying?"Deploying...":"🚀 Deploy Version "+version}
      </button>
    </div>
  </Card>

  /* Deploy log + history */
  <div style={{display:"grid", gap:14, alignContent:"start"}}>
    <Card>
      <SectionTitle icon="💻" title="Deploy Log" sub="Real-time output"/>
      <div style={{background:C.bg, border:`1px solid ${C.border}`}, borderR
      {deployLog.length==0 && (
        <div style={{color:C.textFaint, paddingTop:60, textAlign:"center"
      ) }
      {deployLog.map((l,i)=>(
        <div key={i} style={{color:i==deployLog.length-1&&!deploying?C.g
          <span style={{color:C.textFaint}}>{(i*0.8).toFixed(1)}s </spa
          </div>
        )))
      {deploying && <span style={{color:C.blue}}>■</span>}
      </div>
    </Card>

```

```

<Card>
  <SectionTitle icon="📅" title="Deployment History" sub="All versions
  {DEPLOYMENTS.map((d,i)=>(
    <div key={i} style={{display:"flex", justifyContent:"space-between"
      padding:"8px 0", borderBottom:i<DEPLOYMENTS.length-1?`1px solid ${C.borderColor}`:"none"
      borderTop:"1px solid ${C.borderColor}"}}>
      <div style={{display:"flex", gap:8, alignItems:"center", margin:"0 auto", width:"fit-content", padding:"0 8px", border:"1px solid ${C.borderColor}"}}>
        <span style={{fontFamily:"monospace", fontSize:12, color:C.textColor}}>${d.env}</span>
        <Tag label={d.env} color={d.env==="PRODUCTION"?C.orange:C.blue}>${d.env}</Tag>
        <Tag label={d.status} color={d.status==="LIVE"?C.green:C.textFaint}>${d.status}</Tag>
      </div>
      <div style={{fontSize:10, fontFamily:"monospace", color:C.blue}}>${d.ts}</div>
    </div>
    {d.health && (
      <div style={{textAlign:"right", padding:"0 8px 0 0", border:"1px solid ${C.borderColor}"}}>
        <div style={{fontSize:14, fontWeight:700, color:C.green, font:"normal 14px sans-serif", margin:"0 auto", width:"fit-content", padding:"0 8px", border:"1px solid ${C.borderColor}"}}>${d.uptime}</div>
      </div>
    )}>
  </div>
))}>
</Card>
</div>
</div>
</div>
);
}

// — Tab: FUNDING ——————
function FundingTab() {
  const total = GRANTS.reduce((a,g)=>a+(g.status==="APPROVED"?g.amount:0),0);
  const pipeline = GRANTS.reduce((a,g)=>a+(g.status!="APPROVED"?g.amount:0),0);

  return (
    <div style={{display:"grid", gap:14}}>
      <div style={{display:"grid", gridTemplateColumns:"repeat(4,1fr)", gap:10}}>
        [
          {label:"Secured Capital", value:`$${(total/1000).toFixed(0)}K`, color:C.purple},
          {label:"Pipeline Value", value:`$${(pipeline/1000).toFixed(0)}K`, color:C.purple},
          {label:"Active Apps", value:GRANTS.filter(g=>g.status!="APPROVED").length},
          {label:"Target (Year 1)", value:"$1.37M", color:C.purple},
        ].map((s,i)=><Card key={i} style={{padding:14, textAlign:"center"}}><Stat>
      </div>

      <div style={{display:"grid", gridTemplateColumns:"1fr 1fr", gap:14}}>
        <Card>

```

```

<SectionTitle icon="💰" title="Grant Pipeline" sub="Federal · State · F
{GRANTS.map((g,i)=>
    <div key={i} style={{padding:"10px 0", borderBottom:i<GRANTS.length-1
        <div style={{display:"flex", justifyContent:"space-between", margin
            <div>
                <div style={{fontSize:12, fontWeight:700, color:C.text}}>{g.nam
                    <Tag label={g.status} color={g.color}/>
                </div>
                <div style={{textAlign:"right"}}>
                    <div style={{fontSize:13, fontWeight:700, color:g.color, fontFa
                        ${ (g.amount/1000).toFixed(0) }K
                    </div>
                </div>
            </div>
            <ProgressBar pct={g.pct} color={g.color}/>
            <div style={{fontSize:9, color:C.textFaint, marginTop:3, textAlign:
                </div>
            ) ) }
        </Card>

        <div style={{display:"grid", gap:14, alignContent:"start"}}>
            <Card>
                <SectionTitle icon="🏛️" title="IP Registry" sub="On-chain disclosure
{IP_RECORDS.map((r,i)=>
    <div key={i} style={{padding:"8px 0", borderBottom:i<IP_RECORDS.len
        <div style={{display:"flex", justifyContent:"space-between", align
            <div style={{fontSize:11, fontWeight:700, color:C.text, flex:1,
                <Tag label={r.status} color={r.status==="PROVISIONAL"||r.status
            </div>
            <div style={{display:"flex", gap:8, alignItems:"center"}}>
                <Tag label={r.type} color={r.type==="PATENT"?C.gold:r.type==="T
                    <span style={{fontSize:9, fontFamily:"monospace", color:C.textF
                    <span style={{fontSize:9, color:C.textFaint}}>{r.date}</span>
                </div>
            </div>
        ) ) }
    </Card>

    <Card>
        <SectionTitle icon="📅" title="Next 30-Day Actions" sub="Priority ran
        [
            {a:"Register SAM.gov", due:"TODAY", c:C.red},
            {a:"Book Illinois SBDC appointment", due:"TODAY", c:C.red},
            {a:"Apply NSF I-Corps ($50K)", due:"This week", c:C.orange},
            {a:"Apply Argonne CRI", due:"This week", c:C.orange},
            {a:"File USPTO provisional ($320)", due:"Month 1", c:C.gold},
            {a:"Apply NAIRR compute access", due:"Month 1", c:C.blue},

```

```

        {a:"Submit NSF SBIR Phase I", due:"Month 2", c:C.blue},
    ].map((item,i)=>(
        <div key={i} style={{display:"flex", justifyContent:"space-between"
            padding:"5px 0", borderBottom:i<6?'1px solid ${C.textFaint}`:'none"
            <span style={{fontSize:11, color:C.text}}>{item.a}</span>
            <Tag label={item.due} color={item.c}/>
        </div>
    ))})
</Card>
</div>
</div>
</div>
);

}

// — Main App ——————
const TABS = [
    {id:"swarm", icon:"🧠", label:"Swarm"}, 
    {id:"research", icon:"🔬", label:"Research"}, 
    {id:"clawhub", icon:"🧩", label:"ClawHub"}, 
    {id:"deploy", icon:"🚀", label:"Deploy"}, 
    {id:"funding", icon:"💰", label:"Funding"}, 
];

```

```

export default function App() {
    const [tab, setTab] = useState("swarm");
    const [time, setTime] = useState(new Date());
    useEffect(()=>{const t=setInterval(()=>setTime(new Date()),1000);return()=>clearInterval(t)}

```

```

    return (
        <div style={{minHeight:"100vh", background:C.bg, color:C.text, fontFamily:'I
            /* Header */
        <div style={{background:C.bg2, borderBottom:`1px solid ${C.border}`, padding: "10px 0", border-radius: "10px", width: "100%", height: "100%", display:"flex", align-items:"center", justify-content:"space-between", height: "100%", width: "100%"}}
            <div style={{display:"flex", align-items:"center", gap:14}}>
                <div style={{display:"flex", align-items:"center", gap:8}}>
                    <div style={{width:28, height:28, borderRadius:6, background:`linear-gradient(45deg, transparent, transparent 5px, ${C.bg} 5px, ${C.bg} 10px, transparent 10px)`}}
                        <img alt="Logo" style={{width:100%, height:100%}}/>
                    </div>
                <div>
                    <div style={{fontSize:14, fontWeight:800, color:C.text, letterSpacing: "0.1em", margin: "0 10px 0 0", width: "100% - 20px", height: "100% - 20px", display: "flex", align-items: "center", justify-content: "center", fontStyle: "italic", fontVariant: "small-caps", fontWeight: "normal", color: "white", background: "black", border: "1px solid black", border-radius: "50%", width: "20px", height: "20px", position: "relative", overflow: "hidden", transition: "all 0.3s ease-in-out", transform: "scale(1.1) rotate(-15deg)"}}>
                        L
                    </div>
                </div>
            <div style={{width:1, height:28, background:C.border}}/>
            <div style={{display:"flex", gap:2}}>

```

```

{TABS.map(t=>(
  <button key={t.id} onClick={()=>setTab(t.id)}
    style={{padding:"6px 14px", borderRadius:4, fontWeight:600, fontS
      background: tab==t.id?C.blue:"transparent",
      color: tab==t.id?C.white:C.textDim,
      border: tab==t.id?'1px solid ${C.blue}`:"1px solid transparent
      display:"flex", alignItems:"center", gap:5,
      boxShadow: tab==t.id?'0 0 12px ${C.blue}44`:"none"})>
    <span>{t.icon}</span><span>{t.label}</span>
  </button>
))}

</div>
</div>

<div style={{display:"flex", alignItems:"center", gap:16}}>
  <div style={{display:"flex", gap:12}}>
    {[

      {label:"OMEGA ROOT", ok:true},
      {label:"CHAIN", ok:true},
      {label:"MCP", ok:true},
    ].map(s=>(
      <div key={s.label} style={{display:"flex", alignItems:"center", gap
        <div style={{width:6, height:6, borderRadius:"50%", background:s.
          boxShadow:`0 0 5px ${s.ok?C.green:C.red}`}}/>
        <span style={{fontSize:9, color:C.textDim, letterSpacing:"0.08em"
        </div>
      ))}
    </div>
    <div style={{fontSize:11, fontFamily:"monospace", color:C.textDim}}>
      {time.toLocaleTimeString()}
    </div>
    <div style={{background:C.greenBg, border:`1px solid ${C.greenDim}`, bo
      padding:"4px 10px", fontSize:10, fontWeight:700, color:C.green}}>
      MAINNET LIVE
    </div>
  </div>
</div>

/* Content */

<div style={{padding:"18px 20px", maxWidth:1400, margin:"0 auto"}}>
  {tab=="swarm" && <SwarmTab/>}
  {tab=="research" && <ResearchTab/>}
  {tab=="clawhub" && <ClawHubTab/>}
  {tab=="deploy" && <DeployTab/>}
  {tab=="funding" && <FundingTab/>}
</div>

```

```
<style>{`  
  * { box-sizing: border-box; margin: 0; padding: 0; }  
  ::-webkit-scrollbar { width: 6px; height: 6px; }  
  ::-webkit-scrollbar-track { background: ${C.bg}; }  
  ::-webkit-scrollbar-thumb { background: ${C.border}; border-radius: 3px;  
    input::placeholder, textarea::placeholder { color: ${C.textFaint}; }  
  @keyframes blink { 0%,100%{opacity:1} 50%{opacity:0} }  
 `}</style>  
</div>  
) ;  
}
```