

```
import React, { useState, useEffect, useRef } from 'react';
import { Search, User, Compass, BookOpen, MessageSquare, PlusCircle, Star,
Users, Briefcase, Heart, Wind, Share2, Filter, MapPin, MoreVertical,
Trash2, Bookmark, Flag, X, Sparkles, CheckCircle, Edit3, Send,
ChevronDown, ShieldAlert, Zap, ArrowRight, Sun, Users2, Target, Bell,
Video, ThumbsUp, ThumbsDown, Wand2, Image as ImageIcon, Upload, LogOut,
Check, UserPlus, Globe, MessageCircle, Navigation, ChevronLeft,
ChevronRight, UsersRound } from 'lucide-react';

// --- MOCK DATA ---
const mockUsers = {
  101: {
    id: 101,
    name: 'Alexina B.',
    avatar: 'https://placeholder.co/100x100/a3e636/101010?text=AB',
    followers: 124,
    following: 56,
    followedTags: ['Community Gardens', 'Reforestation'],
    ggCredits: { leader: 6, collaborator: 17, mobilised: 53 },
    activeProject: { id: 7, title: "Leiderdorp Community Cleanup",
location: "Leiderdorp Center" },
    contributionHistory: { environment: 60, social: 30, animal: 10 },
    vision: "A world with flourishing ecosystems, teeming with
beautiful nature and wildlife, where humans nurture our planet as one
human family.",
    mission: { name: "Urban Greening Initiative", description: "To
design and implement sustainable green spaces within urban environments,
fostering community engagement and increasing local biodiversity." },
  },
  102: { // New user with only a vision
    id: 102,
    name: 'Charlie D.',
    avatar: 'https://placeholder.co/100x100/38bdf8/FFFFFF?text=CD',
    followers: 0,
    following: 0,
    followedTags: [],
```

```

    ggCredits: { leader: 0, collaborator: 0, mobilised: 0 },
    activeProject: null,
    contributionHistory: { environment: 0, social: 0, animal: 0 },
    vision: "A community where everyone has access to healthy food and
no one goes hungry.",
    mission: null,
  },
};

let currentUser = null; // Default to logged out

const mockFeed = [
  { id: 4, type: 'Project Invite', authorId: 101, title: 'Lend Your
Design Skills for a Local Animal Shelter', description: "Our local animal
shelter needs a new, heartwarming logo and branding guide...", imageUrls:
['https://placeholder.co/800x600/1a1a1a/a3e636?text=Design+for+Paws'], tags:
['Design', 'Animals', 'Community'], location: null, remote: true },
];

const colors = { background: 'bg-black', card: 'bg-[#1a1a1a]', text:
'text-white', textMuted: 'text-gray-400', primary: 'text-lime-400',
primaryBg: 'bg-lime-400', primaryBgHover: 'hover:bg-lime-500', border:
'border-gray-700' };

// --- HELPER COMPONENTS ---
const Card = ({ children, className = "" }) => ( <div
className={` ${colors.card} rounded-lg ${className}`} > {children} </div> );
const Button = ({ children, onClick, variant = 'primary', className = ""
}) => {
  const baseStyle = 'px-4 py-2 rounded-md font-semibold transition-all
duration-300 flex items-center justify-center gap-2';
  const styles = { primary: ` ${colors.primaryBg} text-black
${colors.primaryBgHover}`, secondary: `bg-transparent border-2
${colors.border} ${colors.text} hover:bg-gray-800` };
  return <button onClick={onClick} className={` ${baseStyle}
${styles[variant]} ${className}`} >{children}</button>;

```

```

};

const Textarea = ({ label, placeholder, rows = 3, value, onChange }) => (
  <div>
    <label className="block text-sm font-medium text-gray-300
mb-1">{label}</label>
    <textarea placeholder={placeholder} rows={rows} value={value}
onChange={onChange} className="w-full bg-gray-900 border border-gray-700
rounded-lg py-2 px-3 focus:outline-none focus:ring-2 focus:ring-lime-400"
/>
  </div>
);

```

```
// --- CORE COMPONENTS ---
```

```

const GlobalGoodDashboard = ({ setPage, isLoggedIn, setIsLoggedIn }) => {
  const user = isLoggedIn ? currentUser : null;
  const isActive = !!user?.activeProject;
  const { environment = 0, social = 0, animal = 0 } =
user?.contributionHistory || {};
  const totalContributions = environment + social + animal;

  const [leader, setLeader] = useState(0);
  const [collaborator, setCollaborator] = useState(0);
  const [mobilised, setMobilised] = useState(0);
  const [isAnimated, setIsAnimated] = useState(false);

  const targetCredits = { leader: 5, collaborator: 12, mobilised: 105 };

  useEffect(() => {
    if (!isLoggedIn) {
      const duration = 2000;
      const startTime = Date.now();

      const animate = () => {
        const elapsed = Date.now() - startTime;

```

```

        const progress = Math.min(elapsed / duration, 1);

        setLeader(Math.floor(progress * targetCredits.leader));
        setCollaborator(Math.floor(progress *
targetCredits.collaborator));
        setMobilised(Math.floor(progress *
targetCredits.mobilised));

        if (progress === 1) setIsAnimated(true);
        if (progress < 1) requestAnimationFrame(animate);
    };
    requestAnimationFrame(animate);
}
}, [isLoggedIn]);

const conicGradient = user ? `conic-gradient(#22c55e 0%
${user.contributionHistory.environment}%, #eab308
${user.contributionHistory.environment}%
${user.contributionHistory.environment +
user.contributionHistory.social}%, #f97316
${user.contributionHistory.environment + user.contributionHistory.social}%
100%)` : (isAnimated ? `conic-gradient(#22c55e 0% 60%, #eab308 60% 90%,
#f97316 90% 100%)` : '#1a1a1a');

if (!isLoggedIn) {
    return (
        <div className="max-w-4xl mx-auto py-12 px-4 text-center
animate-fade-in-up">
            <h1 className="text-4xl font-bold">Your Global Good Starts
Here!</h1>
            <p className="text-gray-400 mt-4 max-w-2xl mx-auto">On OHM,
your identity is a reflection of the positive impact you create. Every act
of goodwill contributes to the collective wellbeing of life on earth.</p>
            <div className="relative w-72 h-72 sm:w-80 sm:h-80 mx-auto
my-12">

```

```

        <div className={`absolute inset-0 rounded-full
transition-all duration-1000 p-1 ${isAnimated ? 'bg-lime-500/10' :
'bg-gray-800'}}`>
            <div className="w-full h-full rounded-full p-1
bg-black">
                <div className="w-full h-full rounded-full p-8
transition-all duration-1000" style={{ background: conicGradient }}>
                    <div className="w-full h-full bg-black
rounded-full flex flex-col items-center justify-center p-4">
                        <span className="text-lime-400 text-xs
font-bold uppercase tracking-wider">Global Good Credits</span>
                        <div className="text-center mt-2">
                            <p className="text-sm"><span
className="font-bold text-xl">{leader}</span> Leader</p>
                            <p className="text-sm"><span
className="font-bold text-xl">{collaborator}</span> Collaborator</p>
                            <p className="text-sm"><span
className="font-bold text-xl">{mobilised}</span> Mobilised</p>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div className="mt-8 flex flex-col items-center gap-2">
        <Button onClick={() => setPage('Subscribe')}
className="!py-3 !text-lg min-w-[300px]">Subscribe</Button>
        <p className="text-xs text-gray-500 mt-1">Join OHM to
start your own Global Good journey.</p>
        <Button onClick={() => setPage('Feed')}
variant="secondary" className="!py-2 !text-md min-w-[300px] mt-4">Continue
to Feed</Button>
    </div>
</div>
);
}

```

```

    return (
      <div className="max-w-4xl mx-auto py-12 px-4 text-center
animate-fade-in-up">
        <h1 className="text-3xl font-bold">Welcome back,
{user.name.split(' ')[0]}</h1>
        <p className="text-gray-400 mt-2">{isActive ? "Your
contribution is flowing." : "Your current is stationary. Let's start a new
flow."}</p>
        <div className="relative w-72 h-72 sm:w-80 sm:h-80 mx-auto
my-12">
          <div className={`absolute inset-0 rounded-full
transition-all duration-500 ${isActive ? 'p-2 bg-lime-500/10' : 'p-1
bg-gray-800'}`}>
            <div className="w-full h-full rounded-full p-1
bg-black">
              <div className="w-full h-full rounded-full p-8"
style={{ background: conicGradient }}>
                <div className="w-full h-full bg-black
rounded-full flex flex-col items-center justify-center p-4 cursor-pointer"
onClick={() => setPage('MyGlobalGood')}>
                  <span className="text-lime-400 text-xs
font-bold uppercase tracking-wider">Global Good Credits</span>
                  <div className="text-center mt-2">
                    <p className="text-sm"><span
className="font-bold">{user.ggCredits.leader}</span> Leader</p>
                    <p className="text-sm"><span
className="font-bold">{user.ggCredits.collaborator}</span>
Collaborator</p>
                    <p className="text-sm"><span
className="font-bold">{user.ggCredits.mobilised}</span> Mobilised</p>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    )
  }
}

```

```

        </div>
        <Button onClick={() => setPage('Feed')} className="!py-3
!text-lg min-w-[300px]">Continue</Button>
    </div>
    );
};

const SubscribeModal = ({ setPage, setIsLoggedIn }) => (
    <div className="fixed inset-0 bg-black bg-opacity-80 flex items-center
justify-center z-50 p-4">
        <Card className="max-w-md w-full text-center relative
animate-fade-in-up p-6">
            <button onClick={() => setPage('Dashboard')}
className="absolute top-4 right-4 text-gray-500 hover:text-white"> <X
size={24} /> </button>
            <h2 className="text-3xl font-bold mb-4">Join the Movement</h2>
            <p className="text-gray-400 mb-6">Become a contributing member
of our ecosystem.</p>
            <div className="bg-gray-900 p-6 rounded-lg border
border-gray-700 mb-6">
                <p className="text-4xl font-bold mb-2">$9.99 <span
className="text-lg text-gray-400 font-normal">/ month</span></p>
                <p className="text-lime-400 font-semibold">Complete access
to all platform features.</p>
            </div>
            <Button onClick={() => setPage('Onboarding')} className="w-full
!py-3 text-lg"> Start Your Journey </Button>
        </Card>
    </div>
);

const OnboardingVision = ({ setPage, onVisionCreated }) => {
    const [vision, setVision] = useState('');
    return (
        <div className="max-w-3xl mx-auto py-16 px-4 text-center
animate-fade-in-up">

```

```

        <h1 className="text-4xl md:text-5xl font-bold mb-4">Welcome to
OHM</h1>

        <p className="text-lg text-gray-400 mb-12">Your journey begins
with intention.</p>

        <Card className="text-left p-6">
            <h2 className="text-2xl font-bold mb-6">Craft Your Personal
Vision</h2>

            <Textarea
                label="What is the positive impact you wish to create
in the world? This will become your north star."
                placeholder="e.g., A world with flourishing ecosystems,
where humans nurture our planet as one human family."
                rows={5}
                value={vision}
                onChange={(e) => setVision(e.target.value)}
            />

            <p className="text-sm text-gray-500 mt-2">Your Vision is a
living statement which you can edit at any time in your profile.</p>

            <div className="mt-8 flex justify-end">
                <Button onClick={() => onVisionCreated(vision)}> Save
Vision & Go to Feed </Button>
            </div>
        </Card>
    </div>
);
};

const Feed = ({ setPage }) => (
    <div className="text-center p-16">
        <h1 className="text-4xl font-bold text-lime-400 mb-4">Welcome to
the Feed!</h1>

        <p className="text-gray-400">This is where the river of good
flows.</p>

        <Button onClick={() => setPage('Dashboard')} className="mt-8">Back
to Dashboard</Button>
    </div>

```



```

);

// --- MAIN APP ---
export default function App() {
  const [page, setPage] = useState('Dashboard');
  const [isLoggedIn, setIsLoggedIn] = useState(false);

  const handleVisionCreated = (vision) => {
    currentUser = mockUsers[102]; // Assign a new user object
    currentUser.vision = vision;
    setIsLoggedIn(true);
    setPage('Feed');
  };

  const renderPage = () => {
    switch (page) {
      case 'Dashboard': return <GlobalGoodDashboard setPage={setPage}
isLoggedIn={isLoggedIn} />;
      case 'Subscribe': return ( <> <GlobalGoodDashboard
setPage={setPage} isLoggedIn={false} /> <SubscribeModal setPage={setPage}
setIsLoggedIn={setIsLoggedIn} /> </> );
      case 'Onboarding': return <OnboardingVision setPage={setPage}
onVisionCreated={handleVisionCreated} />;
      case 'Feed': return <Feed setPage={setPage} />;
      default: return <GlobalGoodDashboard setPage={setPage}
isLoggedIn={isLoggedIn} />;
    }
  };

  return (
    <div className={` ${colors.background} ${colors.text} min-h-screen
font-sans flex items-center justify-center`} >
      <div className="absolute top-4 right-4">
        <Button onClick={() => { setIsLoggedIn(!isLoggedIn);
currentUser = isLoggedIn ? null : mockUsers[101]; setPage('Dashboard'); }}
variant="secondary">

```

```
        {isLoggedIn ? `Log Out (${currentUser?.name})` : 'Log  
In (Demo User)'}  
        </Button>  
    </div>  
    <main className="w-full">  
        {renderPage()}  
    </main>  
</div>  
);  
}
```