```jsx
import React, { useState, useEffect } from 'react';

// --- INLINE SVG ICONS (replaces lucide-react) ---
const Wand2 = ({ size = 24, className = "" }) => (
    <svg xmlns="http://www.w3.org/2000/svg" width={size} height={size}
viewBox="0 0 24 24" fill="none" stroke="currentColor" strokeWidth="2"
strokeLinecap="round" strokeLinejoin="round" className={className}>
        <path d="M11 5a2 2 0 0 1 2-2h4a2 2 0 0 1 2 2v4a2 2 0 0 1-2 2h-4a2 2
0 0 1-2-2z"></path>
        <path d="m21 21-9.5-9.5"></path>
        <path d="M9.5 14.5 3 21"></path>
        <path d="M3 3 9 9"></path>
    </svg>
);

const ImageIcon = ({ size = 24, className = "" }) => (
    <svg xmlns="http://www.w3.org/2000/svg" width={size} height={size}
viewBox="0 0 24 24" fill="none" stroke="currentColor" strokeWidth="2"
strokeLinecap="round" strokeLinejoin="round" className={className}>
        <rect width="18" height="18" x="3" y="3" rx="2" ry="2"></rect>
        <circle cx="9" cy="9" r="2"></circle>
        <path d="m21 15-3.086-3.086a2 2 0 0 0-2.828 0L6 21"></path>
    </svg>
);

const Upload = ({ size = 24, className = "" }) => (
    <svg xmlns="http://www.w3.org/2000/svg" width={size} height={size}
viewBox="0 0 24 24" fill="none" stroke="currentColor" strokeWidth="2"
strokeLinecap="round" strokeLinejoin="round" className={className}>
        <path d="M21 15v4a2 2 0 0 1-2 2H5a2 2 0 0 1-2-2v-4"></path>
        <polyline points="17 8 12 3 7 8"></polyline>
        <line x1="12" x2="12" y1="3" y2="15"></line>
    </svg>
);

// --- MOCK DATA & STYLING (for standalone demonstration) ---
```

```jsx
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' };
const mockCurrentUser = { name: "Alexina B." };
const mockProject = { title: "Community Garden Build" };



// --- HELPER & NEW COMPONENTS (required for the form) ---
const Card = ({ children, className = "" }) => ( <div
className={`${colors.card} rounded-lg ${className}`}> {children} </div> );
const Textarea = ({ label, placeholder, rows = 3, required = false, value,
onChange, name }) => ( <div> <label className="block text-sm font-medium
text-gray-300 mb-1">{label}</label> <textarea name={name} value={value}
onChange={onChange} placeholder={placeholder} required={required}
rows={rows} className="w-full bg-gray-900 border border-gray-700
rounded-lg py-2 px-3 focus:outline-none focus:ring-2 focus:ring-lime-400"
/> </div> );
const Button = ({ children, onClick, variant = 'primary', className = ""
}) => {
   const baseStyle = 'px-4 py-2 rounded-md font-semibold transition-all
duration-300 flex items-center justify-center gap-2';
   const styles = { primary: `${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 MessageModal = ({ message, show, onClose }) => {
   if (!show) return null;
   return (
      <div className="fixed inset-0 bg-black bg-opacity-70 flex
items-center justify-center z-50">
         <div className="bg-[#1e1e1e] border border-gray-700 rounded-lg
p-8 max-w-sm text-center shadow-2xl">
```

```jsx
                <p className="text-white mb-6">{message}</p>
                <Button onClick={onClose}>Close</Button>
            </div>
        </div>
    );
};


// --- CORE COMPONENT: Global Good Project Impact Form ---
const CompleteProject = ({ setPage, showModal }) => {
    const [step, setStep] = useState('form'); // 'form', 'overviewReview',
'postReview'
    const [formData, setFormData] = useState({
        projectInYourOwnWords: '',
        impactData: '',
        methodology: '',
        futureNeeds: '',
        projectOverview: ''
    });

    const handleFormChange = (e) => {
        const { name, value } = e.target;
        setFormData(prev => ({ ...prev, [name]: value }));
    };

    const handleGenerateOverview = () => {
        const { projectInYourOwnWords, impactData, methodology, futureNeeds
} = formData;
        if (!projectInYourOwnWords || !impactData || !methodology ||
!futureNeeds) {
            showModal("Please fill out all fields before proceeding.");
            return;
        }
        // --- ENHANCED AI GENERATION SIMULATION ---
        // Creates a more structured, narrative overview with paragraphs.
        const overviewText = `**Project
Overview**\n${projectInYourOwnWords}\n\n` +
```

```
                            `**Key Impacts**\nThe tangible results of this
initiative were significant. We successfully achieved the following key
milestones: ${impactData.split(',').map(item => `\n-
${item.trim()}`).join('')}.\n\n` +
                            `**Reflections on Methodology**\nReflecting on
the project's execution, we found that ${methodology}. This experience
provided valuable insights that will inform our approach to future
activities.\n\n` +
                            `**Next Actions & Future Needs**\nLooking
ahead, this project highlighted a clear need for follow-on activities.
Specifically, ${futureNeeds}. Our next steps will involve planning for
these future needs to build upon the momentum we've created.`;

        setFormData(prev => ({ ...prev, projectOverview: overviewText }));
        setStep('overviewReview');
    };


    // STEP 3: Final Review of the simplified "Impact Post"
    if (step === 'postReview') {
        return (
            <div className="max-w-3xl mx-auto py--8 px-4">
                <h2 className="text-4xl font--bold text-center mb-4">Review
Your Impact Post</h2>
                <p className="text-center text-gray-400 mb-8">This is the
simplified post that will appear on the public feed. The full details will
be archived in the Living Library.</p>

                <Card className="p-6">
                    <h3 className="text-xl font-bold text-lime-400
mb-4">Impact Post Preview</h3>
                    <div className="border border-gray-700 rounded-lg p-4
space-y-4 bg-black">
                        <div className="flex justify-between items-start">
                            <div>
                                <h4 className="font-bold
text-lg">{mockProject.title}</h4>
```

```
                                <p className="text-sm text-gray-500">by
{mockCurrentUser.name}</p>
                        </div>
                         <span className="text-xs font-semibold
text-lime-400 bg-lime-900/50 px-2 py-1 rounded-full">IMPACT</span>
                    </div>
                    <div className="w-full h-48 bg-gray-800 rounded-md
flex items-center justify-center">
                            <p className="text-gray-500">[Curated Media
Preview]</p>
                    </div>
                    <div>
                        <h5 className="font-bold mb-2">Narrative</h5>
                        <p className="text-sm
text-gray-400">{formData.projectInYourOwnWords || "No narrative
provided."}</p>
                    </div>
                    <div>
                        <h5 className="font-bold mb-2">Key Impact</h5>
                        <p className="text-sm
text-gray-400">{formData.impactData || "No impact data provided."}</p>
                    </div>
                </div>
            </Card>

            <div className="mt-8 pt-6 border-t border-gray-700 flex
justify-between items-center">
                <Button variant="secondary" onClick={() =>
setStep('overviewReview')}>Back to Edit Overview</Button>
                <Button onClick={() => {
                    showModal("Impact Published! Your post is live on
the feed and archived in the library.");
                    setTimeout(() => setPage('Feed'), 3000);
                }}>
                    Publish Impact
                </Button>
```

```
              </div>
          </div>
      );
  }


  // STEP 2: Review the auto-generated "Project Overview" for the Library
  if (step === 'overviewReview') {
      return (
          <div className="max-w-3xl mx-auto py-8 px-4">
              <h2 className="text-4xl font-bold text-center mb-4">Review
Project Card</h2>
              <p className="text-center text-gray-400 mb-8">This
AI-generated overview will form your Project Card in the Living Library.
Edit as you wish. After publishing, this information cannot be
changed.</p>
              <Card className="p-6">
                  <Textarea
                      name="projectOverview"
                      value={formData.projectOverview}
                      onChange={handleFormChange}
                      rows={18}
                  />
              </Card>
              <div className="mt-8 pt-6 border-t border-gray-700 flex
justify-between items-center">
                  <Button variant="secondary" onClick={() =>
setStep('form')}>Back to Form</Button>
                  <Button onClick={() => setStep('postReview')}>Approve
& Continue</Button>
              </div>
          </div>
      );
  }


  // STEP 1: Main form for data entry
  return (
```

```jsx
        <div className="max-w-3xl mx-auto py-8 px-4">
            <h2 className="text-4xl font-bold text-center mb-4">Project
Impact Form</h2>
            <p className="text-center text-gray-400 mb-12">This is your
opportunity to present the impact of your project to the wider community
and inspire others in their Global Good!</p>
            <Card className="p-6">
                <div className="space-y-8">
                    <div>
                        <h3 className="text-xl font-bold text-lime-400
mb-2">1. Project in Your Own Words</h3>
                        <p className="text-gray-400 text-sm mb-4">Describe
the story of your project and its outcomes in your own words. Feel free to
provide rich detail, as it provides the basis for your Impact Post.</p>
                        <Textarea
                            name="projectInYourOwnWords"
                            value={formData.projectInYourOwnWords}
                            onChange={handleFormChange}
                            rows={6}
                        />
                    </div>
                     <div>
                        <h3 className="text-xl font-bold text-lime-400
mb-2">2. Project Impact Data</h3>
                        <p className="text-sm text-gray-500 mb-4">At least
one result. Separate results by comma (,).</p>
                        <Textarea name="impactData"
value={formData.impactData} onChange={handleFormChange} placeholder="e.g.,
50 trees planted, 80 food rations provided to labourers, 200kg waste
collected of which 120kg plastic" rows={4}/>
                    </div>
                    <div>
                        <h3 className="text-xl font-bold text-lime-400
mb-4">3. Media Uploads</h3>
```

```
                        <p className="text-gray-400 text-sm mb-4">Upload up
to 20 photos or short video clips that capture the spirit of your project.
Curate the top 1-5 to feature in the impact post.</p>
                        <div className="p-6 border-2 border-dashed
border-gray-700 rounded-lg text-center">
                            <ImageIcon className="mx-auto h-12 w-12
text-gray-500" />
                            <p className="mt-2 text-sm text-gray-400">Drag
& drop files or</p>
                            <Button variant="secondary" className="mt-2
text-sm !px-3 !py-1"> <Upload size={16}/> Select Files </Button>
                        </div>
                    </div>
                    <div>
                        <h3 className="text-xl font-bold text-lime-400
mb-2">4. Methodology</h3>
                        <p className="text-gray-400 text-sm mb-4">Reflect
on the execution of your project.</p>
                        <Textarea name="methodology"
value={formData.methodology} onChange={handleFormChange}
placeholder="Looking back on your project was there anything you found
challenging or that you would do differently next time? Would you add,
remove or change some of the activities?" rows={4}/>
                    </div>
                     <div>
                        <h3 className="text-xl font-bold text-lime-400
mb-2">5. Future Needs</h3>
                        <p className="text-gray-400 text-sm mb-4">Describe
any potential future developments following this project.</p>
                        <Textarea name="futureNeeds"
value={formData.futureNeeds} onChange={handleFormChange} placeholder="Eg,
you discovered the next door labour camp suffered financial difficulty and
also needs food rations. Or the community garden you just built needs a
regular watering schedule and ongoing maintenance." rows={4}/>
                    </div>
                </div>
```

```jsx
            <div className="mt-8 pt-6 border-t border-gray-700 flex
justify-end">
                <Button onClick={handleGenerateOverview}> Generate &
Review Overview </Button>
            </div>
          </Card>
      </div>
    );
};


// --- Main App Shell (for demonstration) ---
export default function App() {
    const [page, setPage] = useState('CompleteProject');
    const [modalState, setModalState] = useState({ show: false, message: ''
});

    const showModal = (message) => setModalState({ show: true, message:
message });
    const closeModal = () => setModalState({ show: false, message: '' });

    useEffect(() => {
        if(modalState.show) {
            const timer = setTimeout(() => {
                closeModal();
            }, 3000);
            return () => clearTimeout(timer);
        }
    }, [modalState]);


    const renderPage = () => {
        switch (page) {
            case 'CompleteProject':
```

```jsx
            return <CompleteProject setPage={setPage}
showModal={showModal} />;
        case 'Feed':
            return (
                <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">Your impact post has
been successfully published.</p>
                    <Button onClick={() => setPage('CompleteProject')}
className="mt-8"> Submit Another Impact Report </Button>
                </div>
            );
        default:
            return <CompleteProject setPage={setPage}
showModal={showModal} />;
    }
  };

  return (
    <div className={`${colors.background} ${colors.text} min-h-screen
font-sans`}>
        <MessageModal message={modalState.message}
show={modalState.show} onClose={closeModal} />
        <main>{renderPage()}</main>
    </div>
  );
}
```