import React from 'react';

import { Card, CardContent } from "@/components/ui/card";

import { Button } from "@/components/ui/button";

import { Loader2, CheckCircle, AlertTriangle, RefreshCw } from 'lucide-react';

const statusIcons = {

processing: <Loader2 className="w-5 h-5 animate-spin text-[var(--navy)]" />,

complete: <CheckCircle className="w-5 h-5 text-green-500" />,

error: <AlertTriangle className="w-5 h-5 text-red-500" />,

};

export default function AIAnalysisProgress({ steps, error, onRetry }) {

return (

<div className="min-h-screen bg-gradient-to-br from-slate-50 to-blue-50 p-6 flex items-center justify-center">

<Card className="w-full max-w-md bg-white/80 backdrop-blur-sm border-0 shadow-lg">

<CardContent className="p-8">

<div className="text-center mb-6">

<h2 className="text-2xl font-bold text-[var(--navy)]">AI Case Analysis in Progress</h2>

<p className="text-slate-600 mt-2">Please wait while our AI structures your case...</p>

</div>

<div className="space-y-4 mb-6">

{steps.map((step, index) => (

<div key={index} className="flex items-center gap-4">

<div className="flex-shrink-0">

{statusIcons[step.status]}

</div>

<div className="flex-1">

<p className={`font-medium ${step.status === 'processing' ? 'text-[var(--navy)]' : 'text-slate-600'}`}>

{step.title}

</p>

{step.error && (

<p className="text-sm text-red-600 mt-1">{step.error}</p>

)}

</div>

</div>

))}

</div>

{error && (

<div className="bg-red-50 border border-red-200 rounded-lg p-4 mb-4">

<div className="flex items-start gap-3">

<AlertTriangle className="w-5 h-5 text-red-500 mt-0.5 flex-shrink-0" />

<div>

<h4 className="font-medium text-red-800 mb-1">Analysis Failed</h4>

<p className="text-sm text-red-700">{error}</p>

</div>

</div>

</div>

)}

{error && onRetry && (

<div className="flex gap-2">

<Button

onClick={onRetry}

variant="outline"

className="flex-1"

>

<RefreshCw className="w-4 h-4 mr-2" />

Try Again

</Button>

</div>

)}

{!error && steps.length > 0 && (

<div className="text-center">

<div className="inline-flex items-center gap-2 text-sm text-slate-500">

<Loader2 className="w-4 h-4 animate-spin" />

Processing...

</div>

</div>

)}

</CardContent>

</Card>

</div>

);

}