import React, { useState, useEffect } from "react";

import { LegalCase, Document, ChatMessage } from "@/entities/all";

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

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

import { Badge } from "@/components/ui/badge";

import { Plus, MessageSquare, FileText, Calendar, AlertCircle } from "lucide-react";

import { format } from "date-fns";

import { Link, useLocation } from "react-router-dom"; // Added useLocation

import { createPageUrl } from "@/utils";

import CaseCard from "../components/cases/CaseCard";

import CaseStats from "../components/cases/CaseStats";

export default function CasesPage() {

const [cases, setCases] = useState([]);

const [documents, setDocuments] = useState([]);

const [messages, setMessages] = useState([]);

const [isLoading, setIsLoading] = useState(true);

const location = useLocation(); // Initialize useLocation

useEffect(() => {

loadData();

}, [location]); // Added location to dependency array for data refresh on navigation

const loadData = async () => {

setIsLoading(true);

try {

const [casesData, documentsData, messagesData] = await Promise.all([

LegalCase.list("-created\_date"),

Document.list("-created\_date"),

ChatMessage.list("-created\_date")

]);

setCases(casesData);

setDocuments(documentsData);

setMessages(messagesData);

} catch (error) {

console.error("Error loading data:", error);

}

setIsLoading(false);

};

const getCaseDocuments = (caseId) => {

return documents.filter(doc => doc.case\_id === caseId);

};

const getCaseMessages = (caseId) => {

return messages.filter(msg => msg.case\_id === caseId);

};

const handleDeleteCase = async (caseId) => {

try {

// Delete associated documents

const caseDocuments = documents.filter(doc => doc.case\_id === caseId);

await Promise.all(caseDocuments.map(doc => Document.delete(doc.id)));

// Delete associated messages

const caseMessages = messages.filter(msg => msg.case\_id === caseId);

await Promise.all(caseMessages.map(msg => ChatMessage.delete(msg.id)));

// Delete the case

await LegalCase.delete(caseId);

// Reload data to refresh the UI

loadData();

} catch (error) {

console.error("Error deleting case:", error);

}

};

return (

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

<div className="max-w-7xl mx-auto">

<div className="flex flex-col md:flex-row justify-between items-start md:items-center mb-8 gap-4">

<div>

<h1 className="text-3xl font-bold text-[var(--navy)] mb-2">Legal Cases</h1>

<p className="text-slate-600">Manage your cases, documents, and legal research</p>

</div>

<Link to={createPageUrl("CreateCase")}>

<Button

className="bg-gradient-to-r from-[var(--navy)] to-[var(--navy-light)] hover:shadow-lg transition-all duration-300"

size="lg"

>

<Plus className="w-5 h-5 mr-2" />

New Case

</Button>

</Link>

</div>

<CaseStats

cases={cases}

documents={documents}

messages={messages}

isLoading={isLoading}

/>

<div className="grid gap-6">

{isLoading ? (

<div className="grid md:grid-cols-2 lg:grid-cols-3 gap-6">

{Array(6).fill(0).map((\_, i) => (

<Card key={i} className="animate-pulse">

<CardHeader>

<div className="h-6 bg-slate-200 rounded w-3/4"></div>

<div className="h-4 bg-slate-200 rounded w-1/2"></div>

</CardHeader>

<CardContent>

<div className="space-y-3">

<div className="h-4 bg-slate-200 rounded"></div>

<div className="h-4 bg-slate-200 rounded w-2/3"></div>

</div>

</CardContent>

</Card>

))}

</div>

) : cases.length === 0 ? (

<Card className="text-center py-16 bg-white/80 backdrop-blur-sm border-0 shadow-lg">

<CardContent>

<div className="w-16 h-16 mx-auto mb-6 bg-gradient-to-br from-[var(--navy)] to-[var(--navy-light)] rounded-2xl flex items-center justify-center">

<FileText className="w-8 h-8 text-[var(--gold)]" />

</div>

<h3 className="text-xl font-semibold text-[var(--navy)] mb-2">No Cases Yet</h3>

<p className="text-slate-600 mb-6 max-w-md mx-auto">

Start by creating your first legal case. Upload documents, chat with AI, and manage your legal work efficiently.

</p>

<Link to={createPageUrl("CreateCase")}>

<Button

className="bg-gradient-to-r from-[var(--navy)] to-[var(--navy-light)] hover:shadow-lg transition-all duration-300"

>

<Plus className="w-5 h-5 mr-2" />

Create Your First Case

</Button>

</Link>

</CardContent>

</Card>

) : (

<div className="grid md:grid-cols-2 lg:grid-cols-3 gap-6">

{cases.map((legalCase) => (

<CaseCard

key={legalCase.id}

case={legalCase}

documents={getCaseDocuments(legalCase.id)}

messages={getCaseMessages(legalCase.id)}

onDelete={handleDeleteCase}

/>

))}

</div>

)}

</div>

</div>

</div>

);

}