import { useEffect, useState } from "react";

import { useParams, useNavigate } from "react-router-dom";

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

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

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

import { Avatar, AvatarFallback, AvatarImage } from "@/components/ui/avatar";

import { Separator } from "@/components/ui/separator";

import { ArrowLeft, Users, Send } from "lucide-react";

import { supabase } from "@/lib/supabase";

import { useToast } from "@/hooks/use-toast";

import { ChatSection } from "@/components/chat/ChatSection";

const Project = () => {

const { id } = useParams();

const navigate = useNavigate();

const [project, setProject] = useState<any>(null);

const [loading, setLoading] = useState(true);

const { toast } = useToast();

useEffect(() => {

if (id) {

getProject(id);

}

}, [id]);

const getProject = async (projectId: string) => {

try {

const { data, error } = await supabase

.from('projects')

.select('\*')

.eq('id', projectId)

.single();

if (error) throw error;

setProject(data);

} catch (error) {

console.error('Error fetching project:', error);

toast({

title: "Error",

description: "Could not load project",

variant: "destructive",

});

} finally {

setLoading(false);

}

};

const getCategoryLabel = (category: string) => {

switch (category) {

case 'writing':

return 'Writing & Storytelling';

case 'design':

return 'Art & Design';

case 'coding':

return 'Coding & Development';

case 'music':

return 'Music & Audio';

default:

return 'Creative Project';

}

};

if (loading) {

return (

<div className="min-h-screen bg-gradient-warm flex items-center justify-center">

<div className="text-center">

<div className="animate-spin rounded-full h-12 w-12 border-b-2 border-primary mx-auto mb-4"></div>

<p className="text-muted-foreground">Loading project...</p>

</div>

</div>

);

}

if (!project) {

return (

<div className="min-h-screen bg-gradient-warm flex items-center justify-center">

<Card className="text-center p-8">

<CardContent>

<h2 className="text-xl font-semibold mb-2">Project not found</h2>

<p className="text-muted-foreground mb-4">

The project you're looking for doesn't exist or has been removed.

</p>

<Button onClick={() => navigate("/dashboard")}>

Back to Dashboard

</Button>

</CardContent>

</Card>

</div>

);

}

return (

<div className="min-h-screen bg-gradient-warm">

{/\* Header \*/}

<header className="border-b bg-card/50 backdrop-blur-sm">

<div className="container mx-auto px-6 py-4">

<div className="flex items-center gap-4">

<Button variant="ghost" size="sm" onClick={() => navigate("/dashboard")}>

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

Back

</Button>

<div className="flex-1">

<div className="flex items-center gap-3 mb-1">

<h1 className="text-2xl font-bold text-foreground">{project.title}</h1>

<Badge variant="secondary">

{getCategoryLabel(project.category)}

</Badge>

<Badge className={

project.status === 'open' ? 'bg-green-100 text-green-800' :

project.status === 'in-progress' ? 'bg-blue-100 text-blue-800' :

'bg-gray-100 text-gray-800'

}>

{project.status}

</Badge>

</div>

<p className="text-muted-foreground">

Created {new Date(project.created\_at).toLocaleDateString()}

</p>

</div>

</div>

</div>

</header>

<div className="container mx-auto px-6 py-8">

<div className="grid lg:grid-cols-3 gap-8">

{/\* Project Info \*/}

<div className="lg:col-span-1">

<Card className="mb-6">

<CardHeader>

<CardTitle className="flex items-center gap-2">

<Users className="w-5 h-5" />

Project Details

</CardTitle>

</CardHeader>

<CardContent className="space-y-4">

<div>

<h4 className="font-medium mb-2">Description</h4>

<p className="text-muted-foreground">

{project.description || "No description provided"}

</p>

</div>

<Separator />

<div>

<h4 className="font-medium mb-2">Collaboration</h4>

<p className="text-sm text-muted-foreground">

Max {project.max\_collaborators} collaborators

</p>

</div>

<Separator />

<div>

<h4 className="font-medium mb-2">Category</h4>

<p className="text-sm text-muted-foreground">

{getCategoryLabel(project.category)}

</p>

</div>

</CardContent>

</Card>

<Card>

<CardHeader>

<CardTitle>Collaborators</CardTitle>

<CardDescription>

People working on this project

</CardDescription>

</CardHeader>

<CardContent>

<div className="flex items-center gap-3 p-3 rounded-lg bg-muted/50">

<Avatar>

<AvatarFallback>C</AvatarFallback>

</Avatar>

<div>

<p className="font-medium">Creator</p>

<p className="text-sm text-muted-foreground">Project owner</p>

</div>

</div>

<div className="mt-4">

<Button variant="outline" className="w-full">

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

Join Project

</Button>

</div>

</CardContent>

</Card>

</div>

{/\* Chat Section \*/}

<div className="lg:col-span-2">

<ChatSection projectId={project.id} />

</div>

</div>

</div>

</div>

);

};

export default Project;