import { useEffect, useState } from "react";

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

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

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

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

import { Plus, MessageCircle, LogOut, Users, BookOpen, Code } from "lucide-react";

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

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

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

import { CreateProjectDialog } from "@/components/projects/CreateProjectDialog";

import { ProjectCard } from "@/components/projects/ProjectCard";

const Dashboard = () => {

const [user, setUser] = useState<any>(null);

const [projects, setProjects] = useState<any[]>([]);

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

const [showCreateProject, setShowCreateProject] = useState(false);

const navigate = useNavigate();

const { toast } = useToast();

useEffect(() => {

getUser();

getProjects();

}, []);

const getUser = async () => {

const { data: { user } } = await supabase.auth.getUser();

if (!user) {

navigate("/login");

return;

}

setUser(user);

};

const getProjects = async () => {

try {

const { data, error } = await supabase

.from('projects')

.select('\*')

.order('created\_at', { ascending: false });

if (error) throw error;

setProjects(data || []);

} catch (error) {

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

} finally {

setLoading(false);

}

};

const handleLogout = async () => {

await supabase.auth.signOut();

navigate("/");

};

const getCategoryIcon = (category: string) => {

switch (category) {

case 'writing':

return <BookOpen className="w-4 h-4" />;

case 'design':

return <Users className="w-4 h-4" />;

case 'coding':

return <Code className="w-4 h-4" />;

default:

return <Users className="w-4 h-4" />;

}

};

const getCategoryColor = (category: string) => {

switch (category) {

case 'writing':

return 'creative';

case 'design':

return 'collaboration';

case 'coding':

return 'warm';

default:

return 'secondary';

}

};

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 flex justify-between items-center">

<h1 className="text-2xl font-bold text-foreground">CreativeMinds</h1>

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

<Avatar>

<AvatarImage src={user?.user\_metadata?.avatar\_url} />

<AvatarFallback>

{user?.user\_metadata?.full\_name?.charAt(0) || user?.email?.charAt(0)}

</AvatarFallback>

</Avatar>

<Button variant="ghost" size="sm" onClick={handleLogout}>

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

Logout

</Button>

</div>

</div>

</header>

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

{/\* Welcome Section \*/}

<div className="mb-8">

<h2 className="text-3xl font-bold text-foreground mb-2">

Welcome back, {user?.user\_metadata?.full\_name || 'Creative'}!

</h2>

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

Ready to create something amazing today?

</p>

</div>

{/\* Quick Actions \*/}

<div className="grid md:grid-cols-3 gap-6 mb-8">

<Card className="group hover:shadow-warm transition-all duration-300 cursor-pointer"

onClick={() => setShowCreateProject(true)}>

<CardHeader className="text-center pb-4">

<div className="w-16 h-16 bg-gradient-creative rounded-full flex items-center justify-center mx-auto mb-4 group-hover:shadow-glow transition-all duration-300">

<Plus className="w-8 h-8 text-primary-foreground" />

</div>

<CardTitle className="text-xl">Start New Project</CardTitle>

</CardHeader>

<CardContent>

<CardDescription className="text-center">

Begin a creative collaboration with other talented minds

</CardDescription>

</CardContent>

</Card>

<Card className="group hover:shadow-warm transition-all duration-300 cursor-pointer">

<CardHeader className="text-center pb-4">

<div className="w-16 h-16 bg-gradient-collaboration rounded-full flex items-center justify-center mx-auto mb-4 group-hover:shadow-glow transition-all duration-300">

<MessageCircle className="w-8 h-8 text-foreground" />

</div>

<CardTitle className="text-xl">Browse Projects</CardTitle>

</CardHeader>

<CardContent>

<CardDescription className="text-center">

Find exciting projects to join and collaborate on

</CardDescription>

</CardContent>

</Card>

<Card className="group hover:shadow-warm transition-all duration-300 cursor-pointer">

<CardHeader className="text-center pb-4">

<div className="w-16 h-16 bg-secondary rounded-full flex items-center justify-center mx-auto mb-4 group-hover:shadow-glow transition-all duration-300">

<Users className="w-8 h-8 text-secondary-foreground" />

</div>

<CardTitle className="text-xl">Connect & Chat</CardTitle>

</CardHeader>

<CardContent>

<CardDescription className="text-center">

Message other creatives and build your network

</CardDescription>

</CardContent>

</Card>

</div>

{/\* Recent Projects \*/}

<div>

<div className="flex justify-between items-center mb-6">

<h3 className="text-2xl font-bold text-foreground">Recent Projects</h3>

<Button variant="outline" onClick={() => setShowCreateProject(true)}>

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

New Project

</Button>

</div>

{loading ? (

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

{[1, 2, 3].map((i) => (

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

<CardHeader>

<div className="h-4 bg-muted rounded w-3/4"></div>

</CardHeader>

<CardContent>

<div className="h-3 bg-muted rounded mb-2"></div>

<div className="h-3 bg-muted rounded w-1/2"></div>

</CardContent>

</Card>

))}

</div>

) : projects.length > 0 ? (

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

{projects.map((project) => (

<ProjectCard

key={project.id}

project={project}

onClick={() => navigate(`/project/${project.id}`)}

/>

))}

</div>

) : (

<Card className="text-center py-12">

<CardContent>

<div className="w-16 h-16 bg-muted rounded-full flex items-center justify-center mx-auto mb-4">

<Plus className="w-8 h-8 text-muted-foreground" />

</div>

<h3 className="text-lg font-semibold mb-2">No projects yet</h3>

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

Start your first creative project and connect with other makers

</p>

<Button variant="creative" onClick={() => setShowCreateProject(true)}>

Create Your First Project

</Button>

</CardContent>

</Card>

)}

</div>

</div>

<CreateProjectDialog

open={showCreateProject}

onOpenChange={setShowCreateProject}

onProjectCreated={getProjects}

/>

</div>

);

};

export default Dashboard;