// components/MeditationLibrary.tsx

'use client'

import { useState, useEffect } from 'react'

import { User } from '@supabase/supabase-js'

import { supabase } from '../lib/supabase'

interface MeditationLibraryProps {

user: User

}

interface Meditation {

id: string

created\_at: string

config: {

type: string

duration: number

voice: string

backgroundSound: string

}

}

export default function MeditationLibrary({ user }: MeditationLibraryProps) {

const [meditations, setMeditations] = useState<Meditation[]>([])

const [loading, setLoading] = useState(true)

useEffect(() => {

fetchMeditations()

}, [user.id])

const fetchMeditations = async () => {

try {

const { data, error } = await supabase

.from('meditations')

.select('\*')

.eq('user\_id', user.id)

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

if (error) throw error

setMeditations(data || [])

} catch (error) {

console.error('Error fetching meditations:', error)

} finally {

setLoading(false)

}

}

const getTypeIcon = (type: string) => {

const icons: { [key: string]: string } = {

manifestation: '⭐',

relaxation: '',

sleep: '😴',

visualization: '',

affirmations: '',

}

return icons[type] || '🧘'

}

if (loading) {

return (

<div className="bg-white rounded-xl shadow-sm p-8">

<div className="animate-pulse">

<div className="h-8 bg-gray-200 rounded w-1/4 mb-4"></div>

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

<div className="h-24 bg-gray-200 rounded"></div>

<div className="h-24 bg-gray-200 rounded"></div>

<div className="h-24 bg-gray-200 rounded"></div>

</div>

</div>

</div>

)

}

return (

<div className="bg-white rounded-xl shadow-sm p-6 md:p-8">

<h2 className="text-2xl font-bold text-gray-900 mb-6">My Meditations</h2>

{meditations.length === 0 ? (

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

<div className="w-20 h-20 bg-gray-100 rounded-full flex items-center justify-center mx-auto mb-4">

<span className="text-3xl">🧘</span>

</div>

<h3 className="text-lg font-medium text-gray-900 mb-2">No meditations yet</h3>

<p className="text-gray-600">Create your first personalized meditation to get started!</p>

</div>

) : (

<div className="grid gap-4">

{meditations.map((meditation) => (

<div

key={meditation.id}

className="border border-gray-200 rounded-lg p-4 hover:shadow-md transition-shadow cursor-pointer"

>

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

<div className="flex items-center space-x-3">

<span className="text-2xl">{getTypeIcon(meditation.config.type)}</span>

<div>

<h3 className="font-medium text-gray-900 capitalize">

{meditation.config.type} Meditation

</h3>

<p className="text-sm text-gray-600">

{meditation.config.duration} min • {meditation.config.voice} voice

</p>

</div>

</div>

<div className="text-right">

<p className="text-sm text-gray-500">

{new Date(meditation.created\_at).toLocaleDateString()}

</p>

<button className="mt-1 text-purple-600 hover:text-purple-700 text-sm font-medium">

Play →

</button>

</div>

</div>

</div>

))}

</div>

)}

</div>

)

}