Afm vedio imbed

import React, { useState, useEffect } from 'react';  
import { View, Text, StyleSheet, FlatList, TextInput } from 'react-native';  
import { WebView } from 'react-native-webview';  
  
const SermonsScreen = () => {  
 const [sermons, setSermons] = useState<any[]>([]);  
 const [searchQuery, setSearchQuery] = useState('');  
  
 const fetchSermons = async () => {  
 // Fetch sermon data from your API endpoint  
 try {  
 const response = await fetch('https://192.168.100.5:7000/sermons');  
 const data = await response.json();  
 setSermons(data);  
 } catch (error) {  
 console.error('Error fetching sermons:', error);  
 }  
 };  
  
 useEffect(() => {  
 fetchSermons();  
 }, []);  
  
 return (  
 <View style={styles.container}>  
 <TextInput  
 style={styles.searchBar}  
 placeholder="Search for sermons..."  
 value={searchQuery}  
 onChangeText={setSearchQuery}  
 />  
 <FlatList  
 data={sermons.filter((sermon) =>  
 sermon.title.toLowerCase().includes(searchQuery.toLowerCase())  
 )}  
 renderItem={({ item }) => (  
 <View style={styles.sermonCard}>  
 {/\* Use WebView to render the iframe \*/}  
 <WebView  
 originWhitelist={['\*']}  
 source={{ html: item.video\_link }} // Dynamically inject iframe HTML  
 style={styles.webview}  
 />  
 <Text style={styles.sermonTitle}>{item.title}</Text>  
 <Text style={styles.sermonSpeaker}>{item.speaker}</Text>  
 <Text style={styles.sermonDate}>{new Date(item.date).toLocaleDateString()}</Text>  
 </View>  
 )}  
 keyExtractor={(item) => item.id.toString()}  
 />  
 </View>  
 );  
};  
  
const styles = StyleSheet.create({  
 container: {  
 flex: *1*,  
 padding: *16*,  
 },  
 searchBar: {  
 height: *40*,  
 borderColor: '#ccc',  
 borderWidth: *1*,  
 borderRadius: *8*,  
 marginBottom: *16*,  
 paddingLeft: *8*,  
 },  
 sermonCard: {  
 marginBottom: *16*,  
 },  
 webview: {  
 width: '100%',  
 height: *200*, // Adjust height as needed  
 borderRadius: *15*,  
 overflow: 'hidden',  
 },  
 sermonTitle: {  
 fontSize: *18*,  
 fontWeight: 'bold',  
 color: '#000',  
 },  
 sermonSpeaker: {  
 fontSize: *16*,  
 color: '#666',  
 },  
 sermonDate: {  
 fontSize: *14*,  
 color: '#999',  
 },  
});  
  
export default SermonsScreen;

import React, { useState, useEffect } from 'react';  
import { View, Text, StyleSheet, FlatList, TextInput, Button } from 'react-native';  
import { WebView } from 'react-native-webview';  
import { LinearGradient } from 'expo-linear-gradient';  
  
const SermonsScreen = () => {  
 const [sermons, setSermons] = useState<any[]>([]);  
 const [searchQuery, setSearchQuery] = useState('');  
  
 const fetchSermons = async () => {  
 try {  
 const response = await fetch('https://192.168.100.5:7000/sermons');  
 const data = await response.json();  
 setSermons(data);  
 } catch (error) {  
 console.error('Error fetching sermons:', error);  
 }  
 };  
  
 useEffect(() => {  
 fetchSermons();  
 }, []);  
  
 return (  
 <View style={styles.container}>  
 {/\* Search Bar \*/}  
 <TextInput  
 style={styles.searchBar}  
 placeholder="Search for sermons..."  
 placeholderTextColor="#999"  
 value={searchQuery}  
 onChangeText={setSearchQuery}  
 />  
  
 {/\* Sermons List \*/}  
 <FlatList  
 data={sermons.filter((sermon) =>  
 sermon.title.toLowerCase().includes(searchQuery.toLowerCase())  
 )}  
 renderItem={({ item }) => (  
 <LinearGradient colors={['#0A2647', '#273752']} style={styles.sermonCard}>  
 <WebView  
 originWhitelist={['\*']}  
 source={{ html: item.video\_link }}  
 domStorageEnabled={*true*}  
 allowsFullscreenVideo={*true*}  
 style={styles.webview}  
 />  
 <Text style={styles.sermonTitle}>{item.title}</Text>  
 <Text style={styles.sermonSpeaker}>{item.speaker}</Text>  
 <Text style={styles.sermonDescription}>{item.description}</Text>  
 <Button  
 title="Watch Full Sermon"  
 onPress={() => {/\* handle navigation to full sermon video or page \*/}}  
 color="#FFA500"  
 />  
 </LinearGradient>  
 )}  
 keyExtractor={(item) => item.id.toString()}  
 />  
 </View>  
 );  
};  
  
const styles = StyleSheet.create({  
 container: {  
 flex: *1*,  
 backgroundColor: '#0A2647',  
 padding: *16*,  
 },  
 searchBar: {  
 backgroundColor: '#1B263B',  
 color: '#fff',  
 padding: *10*,  
 borderRadius: *8*,  
 marginBottom: *20*,  
 fontSize: *16*,  
 },  
 sermonCard: {  
 marginBottom: *20*,  
 borderRadius: *12*,  
 overflow: 'hidden',  
 padding: *16*,  
 width: '100%',  
 shadowColor: '#000',  
 shadowOffset: { width: *0*, height: *2* },  
 shadowOpacity: *0.3*,  
 shadowRadius: *6*,  
 elevation: *5*,  
 },  
 webview: {  
 width: '100%',  
 height: *200*,  
 borderRadius: *8*,  
 marginBottom: *10*,  
 },  
 sermonTitle: {  
 fontSize: *18*,  
 fontWeight: 'bold',  
 color: '#FFA500',  
 marginBottom: *5*,  
 },  
 sermonSpeaker: {  
 fontSize: *16*,  
 color: '#fff',  
 marginBottom: *5*,  
 },  
 sermonDescription: {  
 fontSize: *14*,  
 color: '#fff',  
 marginBottom: *10*,  
 },  
});  
  
export default SermonsScreen;