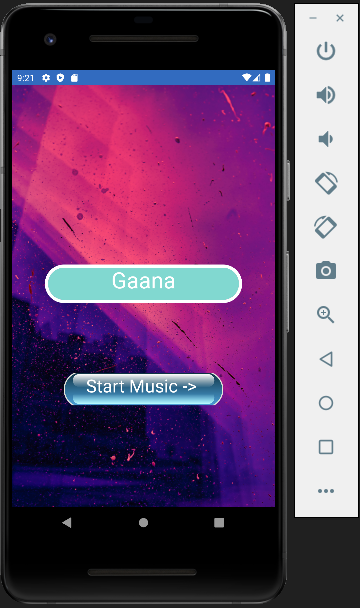
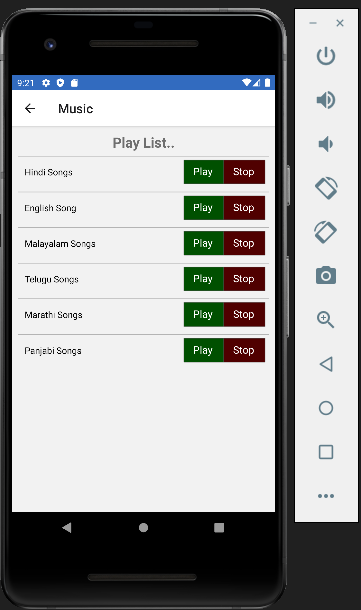
Gaana Music Player App

Abstract

‘Gaana’ is a Music App for streaming Music,the primary purpose of the application is to play music using local project directory and remote url.

UI Screenshots



To Play Sound in React Native

To play Sound / Music in React Native app we are going to use Sound component provided by react-native-sound which supports playing sound clips on iOS, Android, and Windows. react-native-sound does not support streaming.You can run 3 different types of sounds using this library which are listed below:

* aac
* mp3
* wav

### To play the sound from remote URL

var sound1 = new Sound('https://raw.githubusercontent.com/zmxv/react-native-sound-demo/master/pew2.aac', '',

(error, sound) => {

if (error) {

alert('error' + error.message);

return;

}

sound1.play(() => {

sound1.release();

});

});

From the local project directory

var sound1 = new Sound(require('./pew2.aac'),

(error, sound) => {

if (error) {

alert('error' + error.message);

return;

}

sound1.play(() => {

sound1.release();

});

});

## Installation of Dependency

npm **install** react-**native**-sound

## Code to Play Music in React Native App

Home.js

/\* eslint-disable prettier/prettier \*/

import React from 'react';

import {

  ImageBackground,

  StyleSheet,

  View,

  Text,

  TouchableOpacity,

  Image,

  ScrollView,

  StatusBar,

} from 'react-native';

import {useNavigation} from '@react-navigation/native';

import {SafeAreaView} from 'react-native-safe-area-context';

export default function Home() {

  const nav = useNavigation();

  function onPress() {

    console.log('hlo');

    nav.navigate('Music');

  }

  return (

    <>

      <SafeAreaView>

        <StatusBar backgroundColor={'#326bbf'} />

      </SafeAreaView>

      <View style={styles.container}>

        <ImageBackground

          source={require('../assets/Music.jpg')}

          resizeMode="cover"

          style={{flex: 1}}>

          <View style={{marginTop: 400}}>

            <TouchableOpacity>

              <Text style={styles.text}>Gaana</Text>

            </TouchableOpacity>

          </View>

          <TouchableOpacity onPress={onPress} style={[styles.buttonDesign]}>

            <Image

              source={require('../assets/mbutton.png')}

              style={{width: 250, height: 50}}

            />

            <Text style={styles.buttonText}>Start Music -> </Text>

          </TouchableOpacity>

        </ImageBackground>

      </View>

    </>

  );

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    backgroundColor: '#fff',

  },

  text: {

    color: 'white',

    backgroundColor: '#81d8d0',

    borderWidth: 5,

    borderColor: 'white',

    borderRadius: 30,

    fontSize: 35,

    overflow: 'hidden',

    textAlign: 'center',

    width: '75%',

    height: 60,

    marginTop: -120,

    alignSelf: 'center',

    fontFamily: 'Papyrus',

  },

  buttonDesign: {

    justifyContent: 'center',

    alignItems: 'center',

    alignSelf: 'center',

    marginTop: 50,

  },

  buttonText: {

    color: 'white',

    fontSize: 28,

    textAlign: 'center',

    fontFamily: 'Papyrus',

    marginTop: -50,

  },

});

Music-player.js

/\* eslint-disable prettier/prettier \*/

import React, {useEffect} from 'react';

import {

  SafeAreaView,

  StyleSheet,

  Text,

  TouchableOpacity,

  View,

  ScrollView,

  ImageBackground,

} from 'react-native';

import Sound from 'react-native-sound';

const Music = () => {

  let sound1, sound2, sound3, sound4, sound5, sound6;

  useEffect(() => {

    Sound.setCategory('Playback', true);

    return () => {

      if (sound1) sound1.release();

      if (sound2) sound2.release();

      if (sound3) sound3.release();

      if (sound4) sound4.release();

      if (sound5) sound5.release();

      if (sound6) sound6.release();

    };

  }, []);

  const audioList = [

    {

      title: 'Hindi Songs',

      url: 'https://cdn.glitch.global/9f5e8ad4-172b-42a4-b444-a6f2ced0e078/hindi.mpeg?v=1663077389341',

    },

    {

      title: 'English Song ',

      url: 'https://cdn.glitch.global/9f5e8ad4-172b-42a4-b444-a6f2ced0e078/english.mpeg?v=1663077171615',

    },

    {

      title: 'Malayalam Songs',

      url: 'https://cdn.glitch.global/9f5e8ad4-172b-42a4-b444-a6f2ced0e078/ms.mp3?v=1663074537719',

    },

    {

      title: 'Telugu Songs',

      url: 'https://cdn.glitch.global/9f5e8ad4-172b-42a4-b444-a6f2ced0e078/hindi.mpeg?v=1663077389341',

    },

    {

      title: 'Marathi Songs ',

      url: 'https://cdn.glitch.global/9f5e8ad4-172b-42a4-b444-a6f2ced0e078/english.mpeg?v=1663077171615',

    },

    {

      title: 'Panjabi Songs',

      url: 'https://cdn.glitch.global/9f5e8ad4-172b-42a4-b444-a6f2ced0e078/ms.mp3?v=1663074537719',

    },

  ];

  const playSound = (item, index) => {

    if (index == 0) {

      sound1 = new Sound(item.url, '', (error, \_sound) => {

        if (error) {

          alert('error' + error.message);

          return;

        }

        sound1.play(() => {

          sound1.release();

        });

      });

    } else if (index == 1) {

      sound2 = new Sound(item.url, '', (error, \_sound) => {

        if (error) {

          alert('error' + error.message);

          return;

        }

        sound2.play(() => {

          sound2.release();

        });

      });

    } else if (index == 2) {

      sound3 = new Sound(item.url, '', (error, \_sound) => {

        if (error) {

          alert('error' + error.message);

          return;

        }

        sound3.play(() => {

          sound3.release();

        });

      });

    } else if (index == 3) {

      sound4 = new Sound(item.url, '', (error, \_sound) => {

        if (error) {

          alert('error' + error.message);

          return;

        }

        sound4.play(() => {

          sound4.release();

        });

      });

    } else if (index == 4) {

      sound5 = new Sound(item.url, '', (error, \_sound) => {

        if (error) {

          alert('error' + error.message);

          return;

        }

        sound5.play(() => {

          sound5.release();

        });

      });

    } else if (index == 5) {

      sound5 = new Sound(item.url, '', (error, \_sound) => {

        if (error) {

          alert('error' + error.message);

          return;

        }

        sound5.play(() => {

          sound5.release();

        });

      });

    }

  };

  const stopSound = (\_item, index) => {

    if (index == 0 && sound1) {

      sound1.stop(() => {

        console.log('Stop');

      });

    } else if (index == 1 && sound2) {

      sound2.stop(() => {

        console.log('Stop');

      });

    } else if (index == 2 && sound3) {

      sound3.stop(() => {

        console.log('Stop');

      });

    } else if (index == 3 && sound4) {

      sound4.stop(() => {

        console.log('Stop');

      });

    } else if (index == 4 && sound5) {

      sound5.stop(() => {

        console.log('Stop');

      });

    } else if (index == 5 && sound6) {

      sound6.stop(() => {

        console.log('Stop');

      });

    }

  };

  const ItemView = (item, index) => {

    return (

      <View style={styles.feature} key={index}>

        <Text style={styles.textStyle}>{item.title}</Text>

        <TouchableOpacity onPress={() => playSound(item, index)}>

          <Text style={styles.buttonPlay}>Play</Text>

        </TouchableOpacity>

        <TouchableOpacity onPress={() => stopSound(item, index)}>

          <Text style={styles.buttonStop}>Stop</Text>

        </TouchableOpacity>

      </View>

    );

  };

  return (

    <SafeAreaView style={{flex: 1}}>

      <View style={styles.container}>

        <Text style={styles.titleText}>Play List..</Text>

        <ScrollView style={{flex: 1}}>{audioList.map(ItemView)}</ScrollView>

      </View>

    </SafeAreaView>

  );

};

export default Music;

const styles = StyleSheet.create({

  container: {

    flex: 1,

    padding: 10,

  },

  titleText: {

    fontSize: 22,

    textAlign: 'center',

    fontWeight: 'bold',

  },

  textStyle: {

    flex: 1,

    padding: 5,

    fontFamily: 'Roboto',

    color: 'black',

  },

  buttonPlay: {

    fontSize: 16,

    color: 'white',

    backgroundColor: 'rgba(00,80,00,1)',

    borderWidth: 1,

    borderColor: 'rgba(80,80,80,0.5)',

    overflow: 'hidden',

    paddingHorizontal: 15,

    paddingVertical: 7,

  },

  buttonStop: {

    fontSize: 16,

    color: 'white',

    backgroundColor: 'rgba(80,00,00,1)',

    borderWidth: 1,

    borderColor: 'rgba(80,80,80,0.5)',

    overflow: 'hidden',

    paddingHorizontal: 15,

    paddingVertical: 7,

  },

  feature: {

    flexDirection: 'row',

    padding: 5,

    marginTop: 7,

    alignSelf: 'stretch',

    alignItems: 'center',

    borderTopWidth: 1,

    borderTopColor: 'rgb(180,180,180)',

  },

});

Navigation.js

/\* eslint-disable prettier/prettier \*/

import \* as React from 'react';

import {NavigationContainer} from '@react-navigation/native';

import {createNativeStackNavigator} from '@react-navigation/native-stack';

import Home from '../screens/home';

import Music from '../screens/music-player';

const Stack = createNativeStackNavigator();

export default function Navigation() {

  return (

    <NavigationContainer>

      <Stack.Navigator initialRouteName="Home">

        <Stack.Screen name="Music" component={Music} />

        <Stack.Screen

          name="Home"

          component={Home}

          options={{

            headerShown: false,

          }}

        />

      </Stack.Navigator>

    </NavigationContainer>

  );

}

App.js

import \* as React from 'react';

import Navigation from './src/navigation/navigation';

export default function App() {

  return <Navigation />;

}

Reference

* https://aboutreact.com/react-native-play-music-sound/
* https://glitch.com/

Benefits

* I was able to learn more about React native sound
* Helps to explore Glitch .

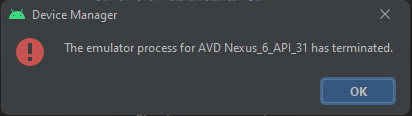
Additional Features

* Drawer
* Play the music from local project directory
* Add a activity Indicator

Issues

* Call multiple functions in a single button click is not working
* I try to run an emulator, it gets killed with this error:

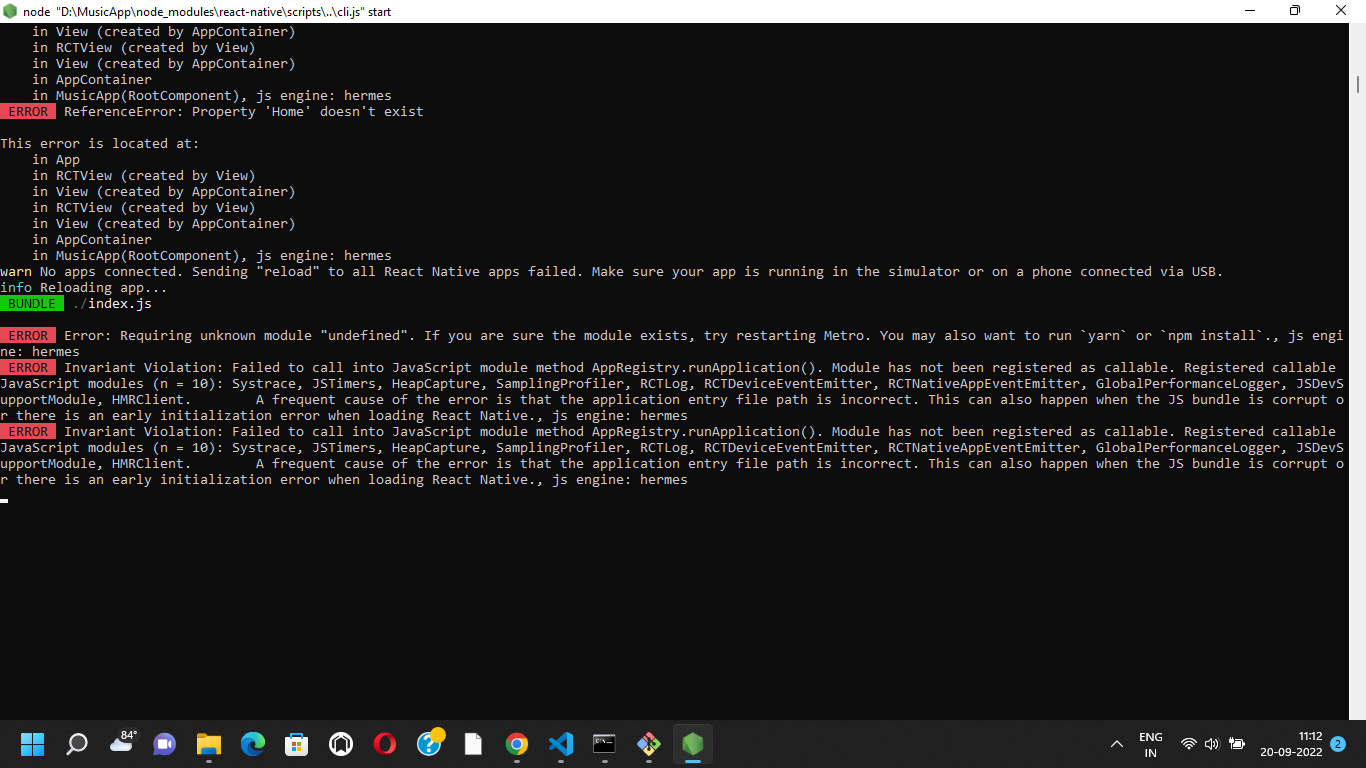
The emulator process for AVD Pixel\_4\_API\_30 has terminated.



I've tried some solutions had suggested in almost similar cases; however, It still doesn't work.

1. I deleted my emulator (also I cleared inside of directory: /.android/avd/) and created a new one again
2. I updated the version
3. I tried to run an emulator with lower APIs but..

* Whenever i am adding StackNavigator code in my App.js then showing below issue..



Solution

This worked for me:

npm start -- --reset-cache

npx react-native start --reset-cache