```java
//author(s): Alex Berliner
package com.cyf.challengeyourfriends;

import java.io.BufferedWriter;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.Scanner;

import android.content.Context;
import android.util.Log;

public class FriendManager {
    //friend local database manager
    private File fileFriendsList;
    private String friendsListName = "friendsList.txt";
    private Context context;

    public FriendManager(Context context) {
        this.context = context;
        // check if friends file exists
        fileFriendsList = new File(context.getFilesDir(), friendsListName);
        if (!fileFriendsList.exists()) {
            log("Friend file not found");
            try {
                fileFriendsList.createNewFile();
            } catch (IOException e) {
                log("Friends list create failed");
                e.printStackTrace();
            }
        }

    }
    //returns a list of all the user's friends
    public ArrayList<String> getFriendsList() {
        ArrayList<String> friendsList = new ArrayList<String>();

        Scanner sc = new Scanner("error");
        try {
            sc = new Scanner(fileFriendsList);
        } catch (FileNotFoundException e1) {
            log("Error opening friends list");
            e1.printStackTrace();
        }
        while (sc.hasNextLine()) {
            String temp = sc.nextLine();
            friendsList.add(temp);
        }
        sc.close();
        return friendsList;
    }
    //adds a friend to the user's db
    public void addFriend(String friendName) {
        // open file to check for friend
        Scanner sc = new Scanner("error");
        try {
            sc = new Scanner(fileFriendsList);
        } catch (FileNotFoundException e1) {
            log("Error opening friends list");
            e1.printStackTrace();
        }

        // return without writing if the friend has been added
        while (sc.hasNextLine()) {
            if (sc.nextLine().trim().equals(friendName)) {
                log("found dupe: " + friendName);
                return;
            }
        }
        sc.close();
        writeLine(friendName);
    }

    public void writeLine(String str) {
        str += "\n";
        try {
            FileOutputStream fos = context.openFileOutput(friendsListName,
                    Context.MODE_APPEND);
            fos.write(str.getBytes());
            fos.close();
        } catch (IOException e) {
            log("File write not successful");
            log("error: " + e);
        }
        log("wrote line " + str);
    }
    //delete all friends :(
    public void wipeFriends() {
        try {
            fileFriendsList.delete();
            fileFriendsList.createNewFile();
        } catch (IOException e) {
            log("Friends list create failed");
            e.printStackTrace();
        }
    }
    //remove one friend based on name
    public void removeFriend(String friendName) {
        // open file to check for friend
        Scanner sc = new Scanner("error");
        try {
            sc = new Scanner(fileFriendsList);
        } catch (FileNotFoundException e1) {
            log("Error opening friends list");
            e1.printStackTrace();
        }
        ArrayList<String> fileTemp = new ArrayList<String>();
        while (sc.hasNextLine()) {
            String temp = sc.nextLine().trim();
            if (!friendName.trim().equals(temp)) {
                fileTemp.add(temp);
            }
        }

        sc.close();
        // delete friends file and then write AL contents sans rm friend
        wipeFriends();

        for (String str : fileTemp) {
            writeLine(str);
        }

    }

    void log(String str) {
        //Log.w("Friends", str);
    }
}
```