PHONE EFFICIENCY AND TROUBLESHOOTING SYSTEM

T. H. FEIROZ KHAN
(Assistant Professor
Computer Science and Engineering Department)
SRM Institute of Science and Technology

ANOUSHKA DUTTA
Computer Science and Engineering
SRM Institute of Science and Technology

CHINMAYA JOSHI
Computer Science and Engineering
SRM Institute of Science and Technology

ANANTHAJITH TCA
Computer Science and Engineering
SRM Institute of Science and Technology

***Abstract* — Whenever a problem arises in our smartphone, the contemporary method of remedy is to contact the corresponding service centre. This is a hectic process as the engineers at the centre would examine the mobile and upon further inspection would conclude the actual cause of the problem. The proposed idea is to eliminate the initial task of the engineer of finding the source of the error by doing an analysis of each component of the device and creating a report which can be handed to the service centre which contains a detailed analysis of every hardware component. This can shorten the time span of correcting the problem exponentially and at the same time give the user .a general idea as to how efficient the system is at the current time. The system comprises of an android application which has the capabilities of checking each and every individual hardware component within the smartphones' chassis. These components range from the CPU and RAM all the way to speakers and more. All these components are checked to find any discrepancies and if found will result in showing an error message to the user prompting them to approach the concerning customer service centre with the useful information in-hand in the form of a e-report. This method works by initially creating a template of the phone's component's working when the phone is brand new i.e. in its peak condition where all the components work at 100% efficiency. The template for the respective smartphones is stored in a centralized database such that on requirement the database can be easily accessed and the data can be retrieved by the application. Then at a later date, whenever a problem arises we can run tests on the system and by comparing this test scores with the original template, we can find out which components are not working at their peak efficiency and by how much the competency of the said component has decreased by. If the decrease is significant then the user will be prompted to go to the service centre.**

Keywords— Phone Efficiency, Troubleshooting System, Component Testing, Template Data, e-Report

I. INTRODUCTION

The main objective of this system is to create an android application which would be able to prompt the user of any defective components and tell the user the efficiency of the said component in an e-report manner. The main objective for the creation of such an application is to minimize the time spent at the service center for the process of reparation of the said smartphone. The proposed android application would have the capabilities of checking each and every individual hardware component within the smartphones' chassis. These components range from the CPU and RAM all the way to speakers and more. All these components are checked to find any discrepancies and if found will result in showing an error message to the user prompting them to approach the concerning customer service centre with the useful information in-hand in the form of a e-report.

In the scenario where a problem on their device is encountered, users are not aware of what the problem is which often results in errors that cause both the user and the system greater problems such as misplacement of parts and unnecessary time and space waste in repair.[1] In such cases a system that checks for the working of the applications in the system can be used such that it reports back when a particular app or device does not respond. In such cases the user gets an output that denotes as to what the problem is . However as an added advantage , the system also allows the user to obtain an efficiency report/data . i.e. the user gets to realize the efficiency rate or difference in efficiency of the device in question from original time and date of use thus allowing the user to monitor the user's device' s peak efficiency.

II. EASE OF USE

We have our smartphones with us so much so that one does not leave their house without it. But what if there arises an issue with the phone. Whenever a problem arises in our smartphone, the contemporary method of remedy is to contact the corresponding service centre. This is a hectic process as the engineers at the centre would examine the mobile and upon further inspection would conclude the actual cause of the problem. The proposed idea is to eliminate the initial task of the engineer of finding the source of the error by doing an analysis of each component of the device and creating a report which can be handed to the service centre which contains a detailed analysis of every hardware component. This can shorten the time span of correcting the problem exponentially.

III. Comparison Between Current and Proposed Systems

If there arises an issue with your smartphone device, the method of solving the issue would be to take the device to the service center of the smartphones' company. There the engineers present at the service center would take the device and run some preliminary tests on your device. At the end of the said tests they would come to the conclusion on which component of the device is the cause of the issue. Upon correctly recognizing the faulty component, thereafter the process is started to rectify the said component. This process could be a total change of the faulty part or it could be just a minor wire which is to be replaced. This process is an extremely long and time consuming process where time is wasted on finding the faulty part of the device.

Our proposed system comprises of an android application which has the capabilities of checking each and every individual hardware component within the smartphones' chassis. These components range from the CPU and RAM all the way to speakers and more. All these components are checked to find any discrepancies and if found will result in showing an error message to the user prompting them to approach the concerning customer service centre with the useful information in-hand in the form of a e-report.This method works by initially creating a template of the phone's component's working when the phone is brand new i.e. in its peak condition where all the components work at 100% efficiency. Then at a later date, whenever a problem arises we can run tests on the system and by comparing this test scores with the original template, we can find out which components are not working at their peak efficiency and by how much the competency of the said component has decreased by. If the decrease is significant then the user will be prompted to go to the service centre.

The proposed system plans on making the check for the faulty component at the service center completely redundant by already creating an e-report which could be presented to the engineers at the service center. They could examine the report and could come to the conclusion as to which component of the device may or may-not be causing the problem the device is experiencing.

IV. System development

A. System Architecture

The proposed system consists of a database that contains a template of values corresponding to different phones and the respective components such that the template values can be used to compare and allow the app to measure and thus obtain a difference that can be used to measure the efficiency of the components. It works by initially creating a template of the phone's component's working when the phone is brand new i.e. in its peak condition where all the components work at full efficiency. Then at a later date, whenever a problem arises we can run tests on the system and by comparing this test scores with the original template, we can find out which components are not working at their peak efficiency and by how much the competency of the said component has decreased by. If the decrease is significant then the user will be prompted to go to the service centre.

The application would have the capabilities of checking each and every individual hardware component within the smartphones' chassis. These components range from the CPU and RAM all the way to speakers and more. All these components are checked to find any discrepancies and if found will result in showing an error message to the user prompting them to approach the concerning customer service centre with the useful information in-hand in the form of a e-report. Depending on that the tests, the components are divided into direct and indirect.

1) *Directly Tested Components*

2) *Indirectly Tested Components*

1) Directly Tested Components

These components are those which can directly be tested programmatic-ally. Such components have predefined libraries present within the Android ecosystem which could be used to get the values related to the component being checked. One example of such a component would be the battery. For checking the battery component, we use the BatteryManager class.[2]

The BatteryManager class contains strings and constants used for values in the Intent.ACTION_BATTERY_CHANGED intent, and provides a method for querying battery and charging properties. Instances of this class must be obtained using Context.getSystemService(Class) with the argument BatteryManager.class or Context.getSystemService(String) with the argument Context.BATTERY_SERVICE.

2) Indirectly Tested Components

Those components are categorized under this which do not have any predefined libraries for the measurement of the efficiency of them. Such components need to be found for them a method by which we can calculate their efficiency. The way we plan on handling these type of components is assigning the said component a small task for it to complete and we would measure the time the component takes to finish the task. Then we will compare the time and result of the said task with the template values of the time and result for that component of that specific smartphones' model.

If the result of the same task for the template and the current value differs then the conclusion is made that the component being checked is not performing correctly. If the result is indeed same for the template and current values then the time is compared. If the time taken in both cases are same then the result is shown that the component is working at 100% efficiency. Otherwise the report indicates that the component is working slower than usual and depending upon how slow the user might be prompted to go to the service center for a professional analysis of the said component. One example component which comes under this category would be the CPU of the device.

B. Use Case

The Fig. 1. shows us how the user(s) will interact within the application. It showcases the start of the application where the user is asked to select the component that they wish to run a test on. Once the selection is made by the user, the application would grab the template test data for that specific device's model's component which will be used to compare the test results of the currently occurring test.

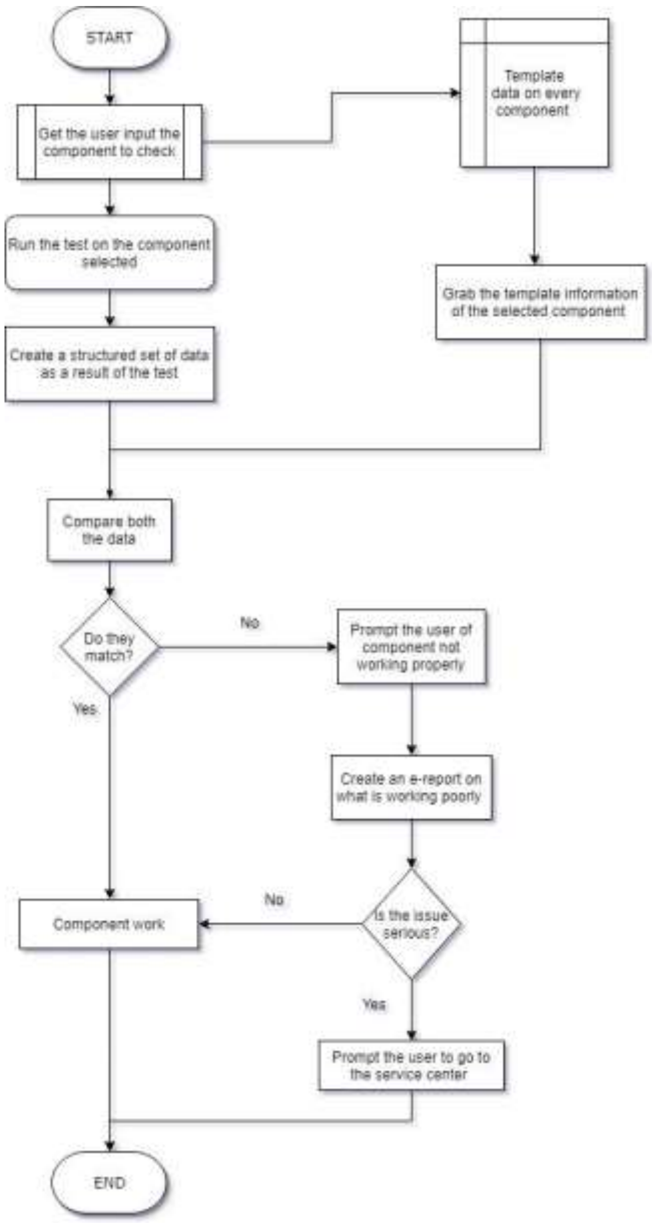

Fig. 1. System Workflow

Thereafter, the app will perform the necessary tests on that component and note the time taken by the component to complete the test.

C. System Modules

The main portion of the application would consist of the further elaborated upon modules, which as a whole would comprise of the working of the proposed system.

The android application would consist of the following 5-modules, namely:

1) Component Selection Module

In this module, the user is asked to make a selection of the component they wish to run the test on. There also is an option of selecting the "All Components" where the tests would be run on each and every components one after the other and a combined e-report would be created at the end. Once the selection is made of the component which the user choose, thereafter the next module begins.

2) Template Data Module

Once the component is selected, the template test values are grabbed from the database. These values will be useful after completion of the test to compare the current test scores with the best-case-scenario test scores.

3) Test Run Module

Once the selection process is finished, the tests are run on the selected component. The type of tests vary with each different component. Depending upon what type of component has been chosen, the test are done directly or indirectly (see Section IV.A.1 and IV.A.2). Related data is collected via the results of the test.[3] The time taken by the component being tested is also calculated in this module.

4) Result Analysis Module

After the completion of the test, the test value scored by the component is compared with the template value for that component by taking the difference between both the values.

$$D_{test\_score} = T_{test\_score} - C_{test\_score}$$

If the test score difference value is negative, then the user is prompted that this component is not working properly and they need to go to the service center.

If the test score difference value is positive or zero, then the time taken to complete the task is compared with the template value in a similar manner.

$$D_{time\_taken} = T_{time\_taken} - C_{time\_taken}$$

If the time taken difference value is negative, then the component is not working at 100% efficiency. Depending on how small the difference value is, the user may or may-not be prompted to go to the service center.

If the time taken difference value is positive, then the component is working at it's peak efficiency.

5) e-Report Creation Module

In this module, the results from the previous modules are organized in a structured and meaningful manner for the user to understand. The final e-report is generated wherein the user can check the whether the component is working or not and if they need to visit the service center or not.

V. INDIRECT TEST EXAMPLE FOR CPU

A. *Testing the CPU*

The CPU (or the Central Processing Unit) is a very essential part of a mobile. To check whether it is working properly, we would have to create a small task for the CPU to complete. This task needs to be computationally heavy so that we can measure the CPU's work at its maximum capability. We would calculate the time required by the CPU to complete that specific task. We would then compare this time to the template time for the same task and measure the difference.[4] If the difference is substantial then we can conclude that the CPU is not working at its peak efficiency and accordingly the user could be prompted if the problem is severe or not.

Now we just need to find a computationally taxing task for the CPU. One such task could be to solve a complex math problem or run some similarly complex algorithm. One such algorithm would be the SHA-1 which stands for ‘Secure Hash Algorithm 1’. In this algorithm, we solve for a ‘cryptographic hash function’. It is a popular algorithm which is used to encrypt the data and store it in the form of an array of 40 digit hexadecimal values (known as the 'hash value') which basically represent the address where the actual data is stored.

Though the SHA-1 algorithm is not as secure as many other similar encryption algorithms, our primary purpose here is just to give the CPU a complex enough algorithm for us to measure it’s performance over.

B. *The Testing Algorithm (SHA-1)*

The Secure Hash Algorithm 1 (SHA-1) is a cryptographic computer security algorithm. It was created by the US National Security Agency in 1995, after the SHA-0 algorithm in 1993, and it is part of the Digital Signature Algorithm or the Digital Signature Standard (DSS).

SHA-1 produces a 160-bit hash value or message digests from the inputted data (data that requires encryption), which resembles the hash value of the MD5 algorithm. It uses 80 rounds of cryptographic operations to encrypt and secure a data object. Some of the protocols that use SHA-1 include:

- Transport Layer Security (TLS)
- Secure Sockets Layer (SSL)
- Pretty Good Privacy (PGP)
- Secure Shell (SSH)
- Secure/Multipurpose Internet Mail Extensions (S/MIME)
- Internet Protocol Security (IPSec)

SHA-1 is commonly used in cryptographic applications and environments where the need for data integrity is high. It is also used to index hash functions and identify data corruption and checksum errors.

C. *SHA-1 Code Snippets*

The codes for indirect testing vary for every components. For the CPU, the algorithm used is SHA-1 (Secure Hash Algorithm-1). The JAVA codes for the algorithm is given below:[5]

```
import java.io.UnsupportedEncodingException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

public class AeSimpleSHA1 {
  private static String convertToHex(byte[] data) {
    StringBuffer buf = new StringBuffer();
    for (int i = 0; i < data.length; i++) {
      int halfbyte = (data[i] >>> 4) & 0x0F;
      int two_halfs = 0;
      do {
        if ((0 <= halfbyte) && (halfbyte <= 9))
          buf.append((char) ('0' + halfbyte));
        else
          buf.append((char) ('a' + (halfbyte - 10)));
        halfbyte = data[i] & 0x0F;
      } while(two_halfs++ < 1);
    }
    return buf.toString();
  }
  public static String SHA1(String text) throws NoSuchAlgorithmException,
   UnsupportedEncodingException {
  MessageDigest md;
  md = MessageDigest.getInstance("SHA-1");
  byte[] sha1hash = new byte[40];
  md.update(text.getBytes("iso-8859-1"), 0, text.length());
  sha1hash = md.digest();
  return convertToHex(sha1hash);
  }
}
```

You can use this class directly calling this method:

```
String sha1_ad1 = AeSimpleSHA1.SHA1("An example string which can contain anything you want to encrypt etc");
String sha1_ad2 = AeSimpleSHA1.SHA1("Some more data to check!");
```

One more example, allows user to input string from console and prints SHA1 hash:

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.security.NoSuchAlgorithmException;

public class Ex01 {

  public static void main(String[] args) throws IOException {
    BufferedReader userInput = new BufferedReader (new InputStreamReader(System.in));
```

```
        System.out.println("Enter string:");
        String rawString = userInput.readLine();

        try {
            System.out.println("SHA1 hash of string: " +
AeSimpleSHA1.SHA1(rawString));
        } catch (NoSuchAlgorithmException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (UnsupportedEncodingException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

VI. Conclusion

The project named ”PHONE EFFICIENCY AND TROUBLESHOOTING SYSTEM” can be designed with the basic functions of Android Studio which utilizes JAVA and XML (eXtensive Markup Language). This project helps decrease the time wasted by going to the service center for finding the faulty component of the device. The application gives the user an e-report which could be shown directly to the service center engineers via which they could determine which component is not working properly. This would be extremely time saving for the user(s).

REFERENCES

[1] Gino Alvarado , Andres Reyes , Maylee Ordoñez; "Desing of system for electrical faults diagnostic in multibranch motorcycle, using free hadware and android APP"; International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT); pp. 27-31; December 2017

[2]

[3] "Android Studio Developers Documentation: BatteryManager", 2018 [Online]. Available: https://developer.android.com/reference/android/os/BatteryManager

[4] Waku, G. M., Bollis, E. R., Rubira, C. M. F., & Torres, R. da S. ; “A Robust Software Product Line Architecture for Data Collection in Android Platform”; 2015 IX Brazilian Symposium on Components, Architectures and Reuse Software. pp. 31-39; September 2015

[5] “How to write an Android CPU benchmark tool (part1)”, March 15, 2016 [Online]. Available: https://www.androidauthority.com/write-an-android-cpu-benchmark-part-1-679929/

[6]

[7] “Java simple class to compute SHA-1 hash", 2010 [Online] http://www.anyexample.com/programming/java/java_simple_class_to_compute_sha_1_hash.xml

[8]

[9]