**===**

**Identifying Hardware and Software Factors Affecting the Touchscreen Responsiveness of Android Smartphones**

Project Documentation Submitted

to the Faculty of School of

Computing and Information Technologies

Of

Asia Pacific College

In Fulfillment of the Requirements for the subject

Systems Analysis & Detailed Design for CS-SS

by

CORONEL, Sherine Jane

DELA CRUZ, Joey Bernadette

JIMENEZ, Marc Adrian

LLANTOS, Joneil Thom

Mr. Manuel Sebastian Sanchez

August 30, 2017

Table of Contents

[Abstract 4](#_Toc491868190)

[I. Introduction 5](#_Toc491868196)

[1.1. Project Context 5](#_Toc491868197)

[1.1.1. The Challenges 5](#_Toc491868198)

[1.1.2. The Opportunity 5](#_Toc491868199)

[1.1.3. Purpose and Description 5](#_Toc491868200)

[1.1.4. Background of the Problem 5](#_Toc491868201)

[1.2. Objectives 7](#_Toc491868208)

[1.3. Scope and Limitation 7](#_Toc491868209)

[II. Literature Review 9](#_Toc491868210)

[2.1. Smartphone Usage 9](#_Toc491868212)

[2.2. Hardware and Software Factors Affecting Touchscreen Responsiveness 10](#_Toc491868219)

[2.2.1. User Interface as a Factor 10](#_Toc491868220)

[2.2.2. Mobile Operating System as a Factor 10](#_Toc491868221)

[2.2.3. Random Access Memory (RAM) as a Factor 11](#_Toc491868222)

[2.2.4. Graphics Processing Unit (GPU) as a Factor 12](#_Toc491868225)

[2.2.5. Types of Touchscreen as a Factor 13](#_Toc491868226)

[2.2.6. Noise (Electronics) as a Factor 13](#_Toc491868227)

[III. Technical Background 15](#_Toc491868228)

[3.1. TUIO 15](#_Toc491868232)

[3.2. Java 15](#_Toc491868233)

[3.3. Firebase 15](#_Toc491868234)

[3.4. Definition of Terms 15](#_Toc491868235)

[IV. Design and Methodology 17](#_Toc491868237)

[4.1. Requirement Analysis 17](#_Toc491868239)

[4.2. Requirement Documentation 18](#_Toc491868240)

[4.3. Design of Software, Systems, Products, and Processes 18](#_Toc491868243)

[4.4. Development and Testing 21](#_Toc491868244)

[4.5. Description of Prototype 21](#_Toc491868245)

[V. UML Diagrams 22](#_Toc491868271)

[5.1. Use Case Full Description 22](#_Toc491868273)

[5.1.1. Accept Terms and Conditions 22](#_Toc491868274)

[5.1.2. Calibrate 23](#_Toc491868275)

[5.1.3. Save Configuration 24](#_Toc491868276)

[5.1.4. Send Feedback 25](#_Toc491868277)

[5.1.5. Restore Defaults 26](#_Toc491868278)

[5.1.6. Configure Touch Algorithm 26](#_Toc491868279)

[5.2. Use Case Diagram 27](#_Toc491868280)

[5.3. Activity Diagram 28](#_Toc491868281)

[5.3.1. Touch Calibrator 28](#_Toc491868282)

[5.3.2. Send Feedback 29](#_Toc491868283)

[5.4. Object Diagram 30](#_Toc491868284)

[5.5. Deployment Diagram 31](#_Toc491868285)

[5.6. Sequence Diagram 31](#_Toc491868286)

[5.6.1. Report 31](#_Toc491868287)

[5.6.2. Touch Calibration with History 32](#_Toc491868288)

[5.6.3. Touch Calibration without History 33](#_Toc491868289)

[5.6.4. Recalibrate 34](#_Toc491868290)

[5.7. Communication Diagram 35](#_Toc491868291)

[5.8. State Diagram 36](#_Toc491868292)

[5.9. Timing Diagram 37](#_Toc491868293)

[5.10. Class Diagram 38](#_Toc491868294)

[5.11. Component Diagram 39](#_Toc491868295)

[5.12. Package Diagram 39](#_Toc491868296)

[5.13. Composite Structure 40](#_Toc491868297)

[5.14. Interaction Overview 40](#_Toc491868298)

[V. Appendices 41](#_Toc491868299)

[6.1. Bibliography 41](#_Toc491868301)

Abstract

Touchscreen based smartphones has been constantly increasing, as well as improving, yearly. Its popularity led to an increased effort to enhance its key features, such as its durability, security, as well as touch responsivenes. However, there are still circumstances that inaccuracies and flaws still exists in the system. The researchers will be using a gesture recognizing application in order to gather the touchscreen data (x and y coordinates, time, touch size, and touch amplitude). Also, the researchers performed a scale of study to identify the varying software and hardware factors that affect the touch responsivenes and inaccuracies of the smartphone.

This study aims to study the interaction between touchscreen smartphone and its user(s). The proponents will investigate the main causes of inaccuracies and inefficiencies regarding the touch registers, by developing a gesture recognizing application that will then be used by the proponents to gather the data necessary for the study.

1. Introduction
   1. Project Context
      1. The Challenges

Android smartphone users still experiences touchscreen inaccuracy despite the manufacturers’ continuous efforts to enhance their products. According to the technology strategist of Synaptics, Andrew Hsu, touch-based mobile phones involve a lot of development work and quite a bit of engineering expertise to give consumers the excellent quality.

* + 1. The Opportunity

This research can help to improve future Android smartphones as this research can inform Android smartphone manufacturers on areas to improve on their devices.

* + 1. Purpose and Description

The purpose of this research is to identify the different hardware and software factors affecting the problems in Android smartphone’s touchscreen responsiveness and accurateness. Also, this research will state how it affects the responsiveness and accurateness of Android smartphone’s touchscreens.

* + 1. Background of the Problem

According to M. Nosrati, et. al, smartphones are hand held devices that are built on a mobile operating system that functions like a computer. Like almost any other cellular phone, they can receive and make calls; create and receive messages, except smartphones can do so much more compared to the regular ones (Nosrati, Karimi, & Hasanvand, 2012). Smartphones have the ability to connect to the internet, making them capable of online communication, online gaming, e-commerce, information retrieval, etc. by downloading and running applications from digital distribution platforms such as Google Play and App Store. It has been constantly developing by introducing new features and innovations throughout the years. As per Scott Cromar, a lot of firms have been competing in the smartphone industry and the competition never stops (Cromar, 2010).

One important and main feature of smartphones is the touchscreen technology incorporated in most smartphones today. According to Allison McCann, touchscreen technology allows smartphone users to interact with their phones using their fingers since it works with anything that has electrical charge including human skin (since we are comprised of atoms with positive and negative charges) (McCann, 2017).

The utilization of touchscreen jumps from mobile devices to life size monitors. Smartphones, gaming consoles, Automatic Teller Machines (ATM), Point of Sales (POS), and digital signage that are using touchscreen is rampantly visible in our daily lives. As compared to other human machine interfaces, touchscreens have various advantages as opposed to physical such as the reduction of physical requirements to deploy it because developers can already integrate buttons into the software itself. Also, the maintenance that comes with the use of it is reduced because dirt, dust, and moisture will not be getting into the spaced between buttons(DMC Co., Ltd., 2011). However, disadvantages also exist – a thin layer of oil or specs of debris can sometimes interfere with the touch reading, that will mislead the system in reading multi-touches (Gray, 2014).

All in all, with the constantly increasing market and intense competition in the touchscreen manufacturing industry (Cecere, Corrocher, & Battaglia, 2014), smartphone advancements never cease to halt. These advancements are made to cover up previous mistakes, but as holes are covered, more holes are revealed.

* 1. Objectives

This research aims to identify the different factors affecting the Android smartphone’s touchscreen’s accurateness and responsiveness to analyze and pin-point how these factors affect the touchscreen. Thus, the objectives are…

1. to develop an Android application that will serve as a data gathering tool, extracting data from the user; and
2. to support the different hardware and software factors contributing to the inaccuracies and responsiveness of Android smartphone’s touchscreen; and
3. to analyze the hardware and software factors affecting the Android smartphone’s touchscreen responsiveness.
   1. Scope and Limitation

Touchscreen is widely used in this generation – especially in devices with display outputs. In fact, smartphones are not the only devices that utilize this technology, cars, refrigerators and laptops are starting to implement this technology. However, problems such as inaccuracy and unresponsiveness exists.

Due to the wide variety of external human factors that affect the touchscreen such as wrist and shoulder posture (Young, Trudeau, Odell, Marinelli, & Dennerlein, 2013); finger diameter (Mobile Usability Testing: Fat Finger Syndrome, 2015) (Yeow & Balakrishnan); melioration (Yechiam, Erev, Yehene, & Gopher, 2003); eye behavior, finger movements (Celik, 2013) etc., the researchers will be focusing on the smartphones’ software more, yet hardware parts of the smartphone will also be checked out. Moreover, due to security constraints in iOS accessibility, this research will only focus on the factors from the Android system only.

The researchers will be using a gesture recognition application that will run on Android mobile phones. For the compatibility of the said app, the handheld requires version 3.0 platform (Honeycomb) or later.

1. Literature Review
2. 1. Smartphone Usage

According to J. Russel, the Philippines is the third largest and fastest growing market for smartphones in South East Asia with the first being Indonesia. Thus, the market of smartphone users in the Philippines is expected to grow more than double its size by 2021, as predicted by Ericsson Mobility Report. Factors affecting the booming industry of smartphones are its increasing affordability, partnered with improvements in technology. Along with the increasing market of Filipinos in the smartphone industry, mobile data that accompanies it is said to grow faster as the percentage of 4G/LTE (long-term evolution) capable smartphones will increase from five percent in 2016 to 70 percent in 2021, a whopping 14-fold increase (Jaio, 2016). In the Philippines, as of 2016, the quantity of internet users already reached around 60 percent of the total population, which is higher than the global internet penetration rate by 10 percent (Subido, 2017).

The existence of smartphones indeed made daily lives easier, with the dynamic functionalities that it can give by enhancing connectivity, efficiency, functionality, entertainment, and much more. Text messages, phone calls, and messaging applications such as Viber, Messenger, Line, etc. and video call applications like Skype reduced the time end effort needed to communicate (talk and see) with people in far places; a feature that revolutionized communication. Smartphones are basically compact personal computers (PC), as it can provide basic functionalities that a PC can offer, just much more small and portable. Smartphones today facilitates everything a user needs in his daily life, starting from e-mails, calculators, and high-resolution cameras, and the advancement never stops, smartphones nowadays are also capable of functioning as virtual reality devices, partnered with their ultra-high definition (UHD) displays and resonating Dolby audio.

The most common text-input devices in a smartphone is a virtual keyboard. Commercial models such as SwiftKey, Fleksy, and Gboard achieves millions of downloads in the AppStore and Google Play Store. These virtual keyboards support functionalities such as glide typing, auto-correction, word prediction, and voice typing which helped different users increases their typing accuracy.

In addition, SMS (Short Message Service) which refers to the exchanging of composed short character based messages (160 characters) to another mobile phone subscriber, is the most widely used form of mobile data service up to date. According to Yeow and Balakrishnan, problems on further studies concluded that the thumb size of the user has a significant effect on mobile phone texting satisfaction (Yeow & Balakrishnan).

2. 2. Hardware and Software Factors Affecting Touchscreen Responsiveness
      1. User Interface as a Factor

Per Vipin Tyagi, iOS is significantly better than Android when it comes to touchscreen “smoothness” because of their indifferences in prioritizing touches. When the users’ finger interacts with the screen, iOS schedules it as real-time priority wherein the system stops other processes and will focus more on rendering its UI, on the other hand, Android treats them as *normal* priority, resulting in a choppy UI (Tyagi, 2015).

* + 1. Mobile Operating System as a Factor

A mobile operating system is a software platform designed to run on mobile devices that runs on top of other application programs. Applications such as Facebook, Uber, or Spotify require a mobile operating system in order to run. (Beal, n.d.) Typically, most operating systems are only applicable to specific hardware specs. Others, on the other hand, root or jailbreak their phones in order to gain more flexibility and access with their mobile operating systems because doing so allows them to use any Mobile Operating System on a rooted or jailbroken phone which also allows them to get access to restricted apps. (Rouse, Mobile Operating System, 2011)

Sensing lines or also known as impulse lines can be found in the touchscreen sensor. It is the one that determines the location where the gesture happened by detecting the current for the electric node layer.(Lorex Technology, 2012)These sensing lines use volt power, the higher the volts it uses, the more sensitive the touchscreen is and the touch experience would be better. According to some researchers at Moto, one factor why Apple iOS’ touchscreen technology is more accurate because it uses 12-volt power source for the sensing lines in touchscreen sensor compared to 3 to 5-volt power source that most other component manufacturers have. (Ganapati, 2010)

Touch response time is the delay time for the user to receive a response or a visual update from the device that the user is expecting, starting from the moment the user performed a gesture. It is measured through small time samples, often ranging from 100-500 ms, and is identified to achieve the best User Experience. (Pantels, Guo, & Chabukswar, 2014) With the use of Agawi's TouchMarks benchmark, he was able to measure the Response time of the device running on Windows, iOS, and Android. And based on his measurements an iOS smartphone is two times faster than an Android or Windows base phone. (Anthony, 2013)

* + 1. Random Access Memory (RAM) as a Factor

Random Access Memory is a type of memory that is volatile and fast compared to other types of memory such as hard drives or solid states drives. Usually, RAM stores the operating system and running applications for them to be quickly accessed by the CPU or processor. (Rouse & Brown, Random Access Memory, 2016) RAM is one of the critical component of a smartphone because critical files that the processor needs, waiting to be accessed, are stored in the RAM. These files, such as OS, Applications, or Game graphics, require speed faster than any other storage. (Schiesser, 2012)

In smartphones, RAM enables you to run multiple applications at the same time, such as Facebook, Twitter, Viber, etc. These applications remain in the RAM which is limited so when it reaches its capacity it will have a difficult time opening any other application causing a degrade in performance, slowing down the phone, and affecting the touchscreen responsiveness. (Tomshardware, 2014) For the touchscreen to be more responsive, you can clear the RAM by killing apps not in use or remove bloatware. (Sinha, 2016)

* + 1. Graphics Processing Unit (GPU) as a Factor

Graphics Processing Unit is a type of circuit that improves the image output by lessening the work of the CPU and increases the performance of graphics and videos. Smartphones today use chipsets in which part of these chipsets include GPUs to stay on parwith the increasing demand of hardware technology. (technopedia , n.d.)

Smartphones have two options in running software. The first one is run the application directly using the CPU offload the software in the GPU then process it to the CPU causing the UI to process it smoothly. The second option is it will put less load on the CPU which is more optimal. Per Simon Kinahan, the CPU frees up, when an application offloads all rendering to the GPU, allowing the CPU to work on other tasks (Kinahan, 2015). Applications, especially games, and user interface of smartphones rely on Graphical Processing Unit since this is the hardware that defines the quality of the smartphone's graphics. Furthermore, if the GPU of the phone cannot handle a certain application or game, this would cause the GPU to not function properly and overheat that might affect the touchscreen’s responsiveness in the application. (Campos, Williams, & Tiganus, 2011)

* + 1. Types of Touchscreen as a Factor

According to Tru-VuMonitors, there are four types of touchscreen. Resistive, Capacitive, Surface acoustical wave, and infrared. (Tru-vumonitors, n.d.) But the study will only discuss the two types of touchscreen that are predominantly used by phones, the Resistive touchscreen and Capacitive touchscreen. (Williams, 2010)

Resistive touchscreen is the most used touch technology. It is composed of two layers which are the flexible top layer and the rigid bottom later. When you touch the screen, the flexible top layer will push onto the rigid bottom layer and determined based on the pressure of the touch. (Tru-vumonitors, n.d.) Since it depends on the pressure applied to the touchscreen, you can use anything to make the touch interface work just as long as it creates enough pressure, such as a gloved finger or a fingernail. Resistive touchscreens contain multiple layers which in return contributes a negative outcome such as the unresponsiveness of resistive touchscreens due to the multiple layers. (Lancet, 2012)

Capacitive touchscreen uses a material that stores electrical charges. It does not rely on pressure, but on the conductive properties of our fingers which releases electrical charges. Since it relies on the electrical charges, it will not respond if not touched by a human skin. (Tru-vumonitors, n.d.) This is made of only one layer, called the insulting layer which is coated by a transparent conductive material on the inside, allowing it to respond from the conductive human body. Because it is made of one layer, it is more sensitive and accurate. It also allows user to make-use of multi-touch gestures. (Lancet, 2012)

* + 1. Noise (Electronics) as a Factor

In order for a touchscreen controller to perform well, it must get the useful signal and suppress the noise. Noise can alter recorded touch data, degrading the responsiveness, performance and reliability of the touchscreen device. (Erickson, 2013) A touchscreen device can be vulnerable to various noise sources, both internal or external and while there are many sources of noise that can interfere with capacitive touch sensing, noises from displays and battery chargers are the outstanding ones. LCD noise is an inherent and fairly predictable thus it can be compromised during the touch solution development process while battery chargers degrades touch performance by emitting noise that has high amplitude, high frequency, or both. (Klein, 2013)

1. Technical Background
2. 1. TUIO

TUIO is a simple transmission protocol designed for table-top tangible user interfaces, such as touchscreens with size ranging from a smartphone to city-wide art installation. It uses User Datagram Protocol as transport and Open Sound Control for media. TUIO has the functionality to send touch events and its position, rotation and shape of objects. TUIO is versatile as it is available for many programming languages such as: C++ and Java (TUIO, n.d.).

* 1. Java

Java is a simple, object-oriented, distributed, interpreted, robust, secure, architecture neutral, portable, high-performance, multithreaded, and dynamic language (The Java Language: A White Paper, 1995).

* 1. Firebase

Firebase is a Google provided application program interface (API) that is deployable on an Android, iOS, or web application, serving its back-end with features such as: analytics, authentication, messaging, real-time database, storage, etc. (Singh, 2016)

* 1. Definition of Terms
* Gestures – how the user interacts with the touchscreen (touch mechanics).
* Flip – Time between press and detachment is less than 351ms & slide of the finger is bigger than 100 pixels vertically (Cichoń, Sobecki, & Szymański, 2013).
* Dragging – Finger moved at least 50 pixels in either direction (Cichoń, Sobecki, & Szymański, 2013).
* Double Tapping – first finger was pressed in less than 501ms & slide of the finger is not more than 50 pixels (Cichoń, Sobecki, & Szymański, 2013).
* Tap – finger touches the screen for less than 1 second, and did not move for more than 50 pixels (Cichoń, Sobecki, & Szymański, 2013).
* Press – finger touches the screen for more than 1 second, and did not move for more than 50 pixels (Cichoń, Sobecki, & Szymański, 2013).
* Noise – Fluctuation in electric signal.

1. Design and Methodology
2. 1. Requirement Analysis

The developers are going to make an application specifically a game which contains multiple gestures such as: Tap, Double Tap, Flip, Drag, and Press. These gestures intend to obtain data such as coordinates, time, touch size, and touch amplitude. The user will conduct the gesture named Tap firstly by tapping five randomly displayed objects that shows up at a time. Second, almost the same as the first one but the only difference is that the user must double tap the randomly placed objects which appears also appears at a time. Third one is where the user must perform the gesture named Flip. This is where the user must flip five specified locations in the canvas. The fourth one is where the participants must select and move the target object by dragging it to another location of the canvas. The user will be given five objects to drag and five locations to move the target object. Lastly, the gesture named Press. The user will be instructed to press his thumb to randomly displayed objects. Each object mentioned will have a limited appearance in terms of time.

* 1. Requirement Documentation
  2. Design of Software, Systems, Products, and Processes

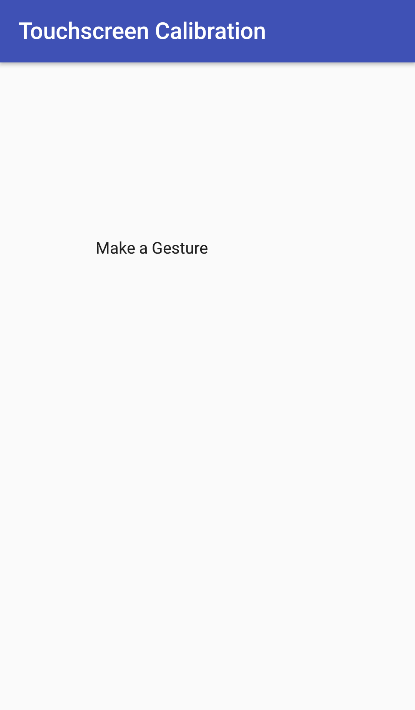
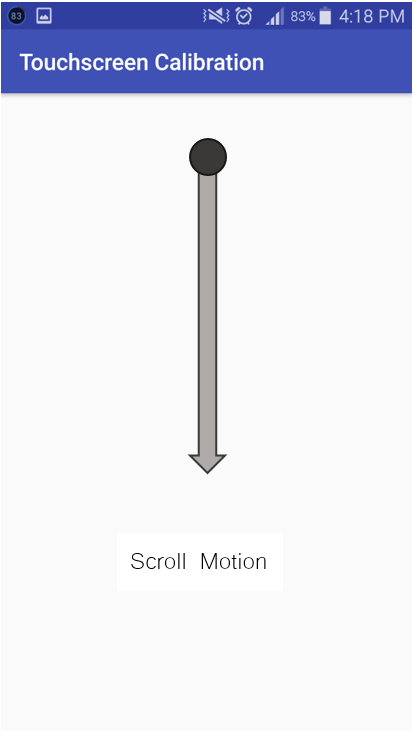
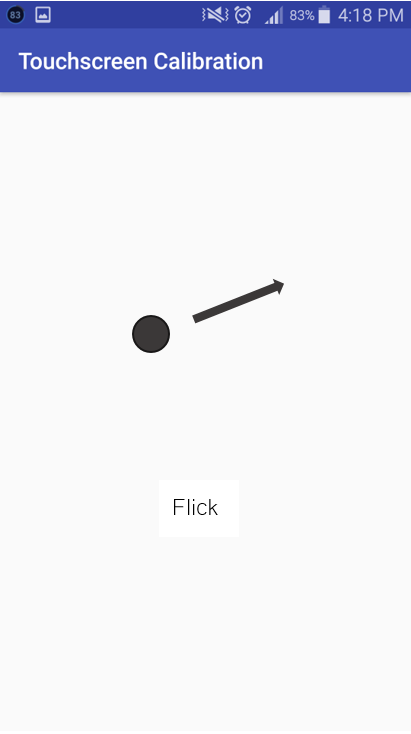
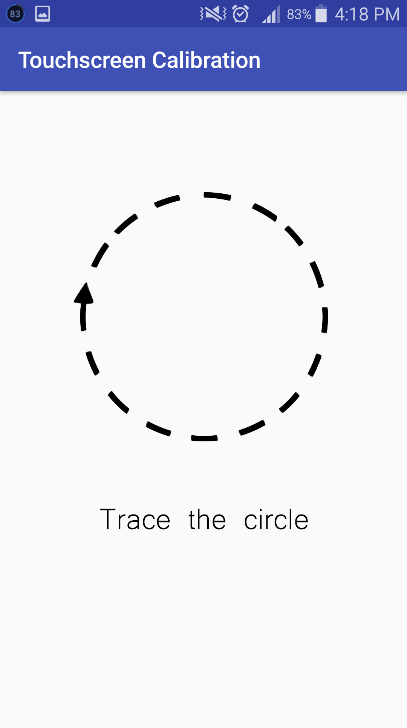
     

Figure 1: Gestures

* 1. Development and Testing

The prototype will be created by using Android Studio mainly because it provides the fastest tools for building applications on every type of Android device. The prototype would enable various users that downloaded the application to perform the gestures in their specific hardware and software components which in return, the researchers would get the data from the results of the user’s performance based on their device components to solidify the assumption that the factors stated affects the touchscreen system.

* 1. Description of Prototype

The prototype was developed from Android Studio and works on Honeycomb version of Android operating system. The prototype operates like a game where the users are given multiple objectives. Its main function is to gather the performance of the users from their phone in order to provide insights that may help the researchers to prove that the factors stated affects the touchscreen system.

1. UML Diagrams\
2. 1. Use Case Full Description
      1. Accept Terms and Conditions

|  |  |  |
| --- | --- | --- |
| Use Case | Accept terms and Conditions | |
| Scenario | User accept terms and conditions of the application | |
| Triggering Event | N/A | |
| Brief Description | User accept or reject to terms and condition of the application | |
| Actor | User | |
| Related Use Cases | N/A | |
| Stakeholder | User | |
| Pre-Condition | N/A | |
| Post-Condition | Agreed or Disagreed to terms and condition | |
| Flow of Activities | Actor | System |
| 1. User accept or reject to terms and conditions | 1.1 System will proceed to homepage if the user accepted |
|  | 1.2 System will end if the user rejected |
|  |  |
| Exception Conditions | The application will not proceed | |

* + 1. Calibrate

|  |  |  |
| --- | --- | --- |
| **Use Case** | Calibrate | |
| **Scenario** | User performs series of gestures | |
| **Triggering Event** |  | |
| **Brief Description** | User performs series of gestures to be computed for the change of calibration | |
| **Actor** | User | |
| **Related Use Cases** | N/A | |
| **Stakeholder** | User | |
| **Pre-Condition** | Accept Permission | |
| **Post-Condition** | Configure touch algorithms | |
| **Flow of Activities** | Actor | System |
| 1. User perform Gesture: Tap | 1.1 System will calculate results |
| 2. User perform Gesture: Double Tap | 2.1 System will calculate results |
| 3. User perform Gesture: Press | 3.1 System will calculate results |
| 4. User perform Gesture: Press and tap | 4.1 System will calculate results |
| 5. User perform Gesture: Pinch | 5.1 System will calculate results |
| **Exception Conditions** | The algorithm of the calibration application will not change. | |

* + 1. Save Configuration

|  |  |  |
| --- | --- | --- |
| **Use Case** | Save Configuration | |
| **Scenario** | User saves his configuration | |
| **Triggering Event** | Calibrate | |
| **Brief Description** | User saves his calibration and apply to phone | |
| **Actor** | User | |
| **Related Use Cases** | Calibrate | |
| **Stakeholder** | User | |
| **Pre-Condition** | Calibrate | |
| **Post-Condition** |  | |
| **Flow of Activities** | Actor | System |
| User clicks on save settings | 1.1 System will commit the saved configuration |
|  |  |
| **Exception Conditions** | Configuration will not be saved | |

* + 1. Send Feedback

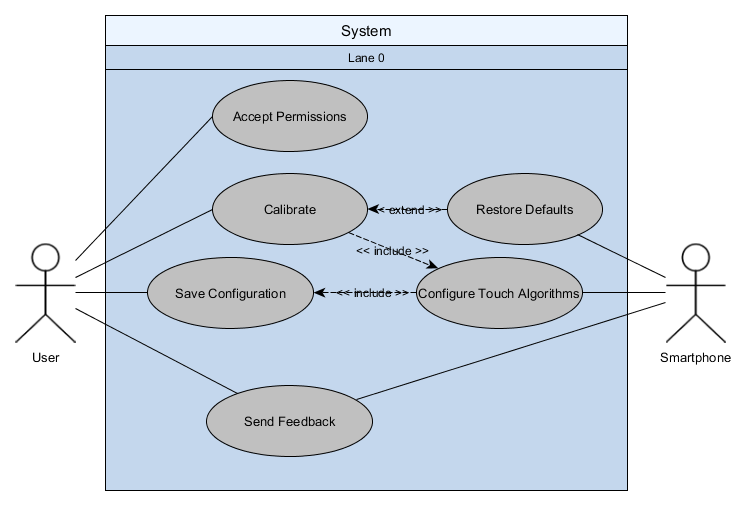
|  |  |  |
| --- | --- | --- |
| **Use Case** | Send Feedback | |
| **Scenario** | User will send feedback or send a report | |
| **Triggering Event** | User want to comment on a feature; User finds an application error | |
| **Brief Description** | User sends feedback or sends a report for comments or problems respectively | |
| **Actor** | User | |
| **Related Use Cases** | N/A | |
| **Stakeholder** | User | |
| **Pre-Condition** | Accept Permissions | |
| **Post-Condition** | Feedback Sent | |
| **Flow of Activities** | Actor | System |
| 1. User clicks on Send Feedback tab | 1.1 System displays the Feedback Form |
| 2. User fills up the Feedback Form |  |
| 3. User clicks on the submit button | 3.1 System saves the Response |
|  | 3.2 System sends the Response to the server/database |
| **Exception Conditions** |  | |

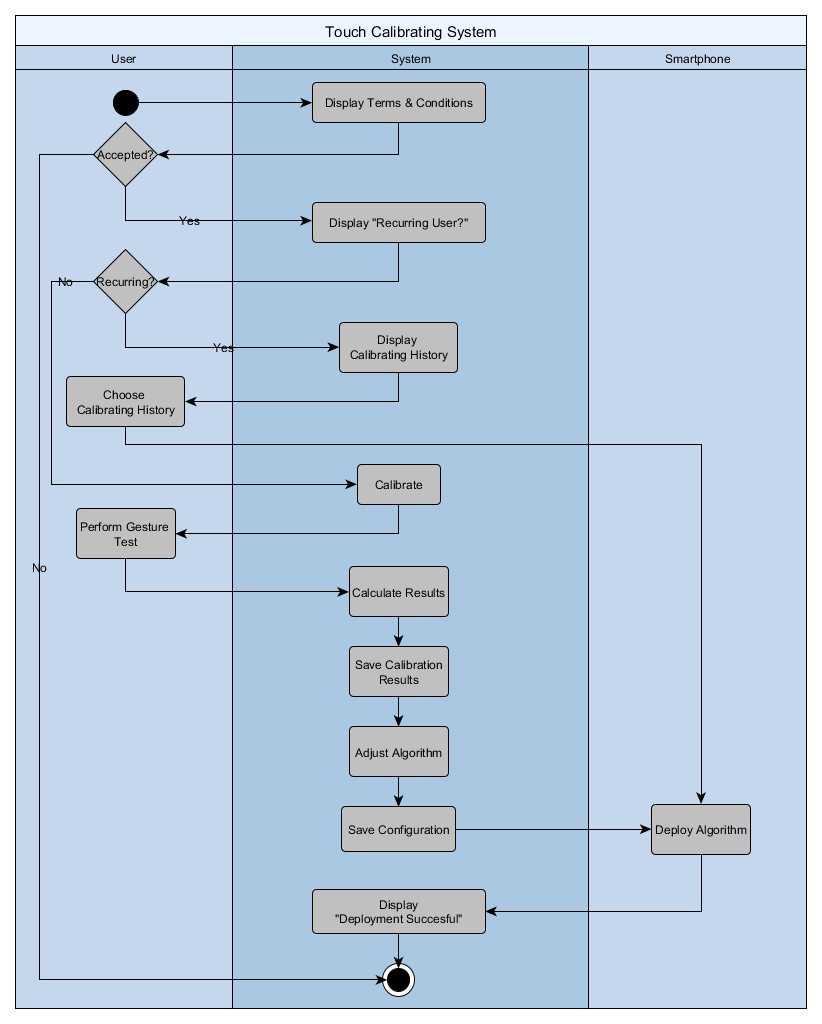
* + 1. Restore Defaults

|  |  |  |
| --- | --- | --- |
| **Use Case** | Restore Defaults | |
| **Scenario** | User will restore default settings | |
| **Triggering Event** | User wants to restore original calibration of the smartphone | |
| **Brief Description** | User will restore default settings of the smartphone application calibration | |
| **Actor** | User | |
| **Related Use Cases** | Calibrate | |
| **Stakeholder** | User | |
| **Pre-Condition** | Calibrate | |
| **Post-Condition** | Default settings restored | |
| **Flow of Activities** | Actor | System |
| 1. User clicks restore defaults | 1.1 System will deploy default settings on the smartphone |
| **Exception Conditions** |  | |

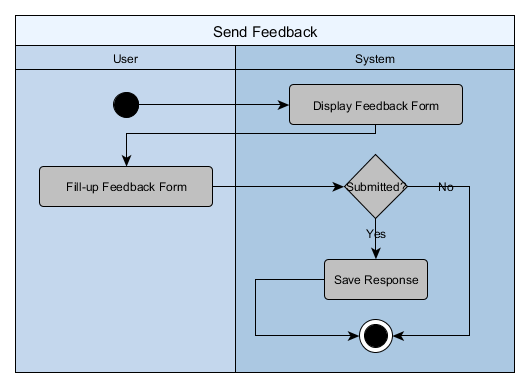
* + 1. Configure Touch Algorithm

|  |  |  |
| --- | --- | --- |
| **Use Case** | Configure touch algorithms | |
| **Scenario** | System configure the touch algorithms | |
| **Triggering Event** | System adjust algorithms based on the calculated results | |
| **Brief Description** | System will configure the touch algorithms based on the calibration results | |
| **Actor** | Smartphone | |
| **Related Use Cases** | Calibrate | |
| **Stakeholder** | Smartphone | |
| **Pre-Condition** | Calibrate | |
| **Post-Condition** |  | |
| **Flow of Activities** | Actor | System |
| 1. Smartphone calculates total results from the calibration. | 1.1 System will adjust algorithms based on the calculation. |
| **Exception Conditions** | Configuration failed, the algorithms will restore the default settings. | |

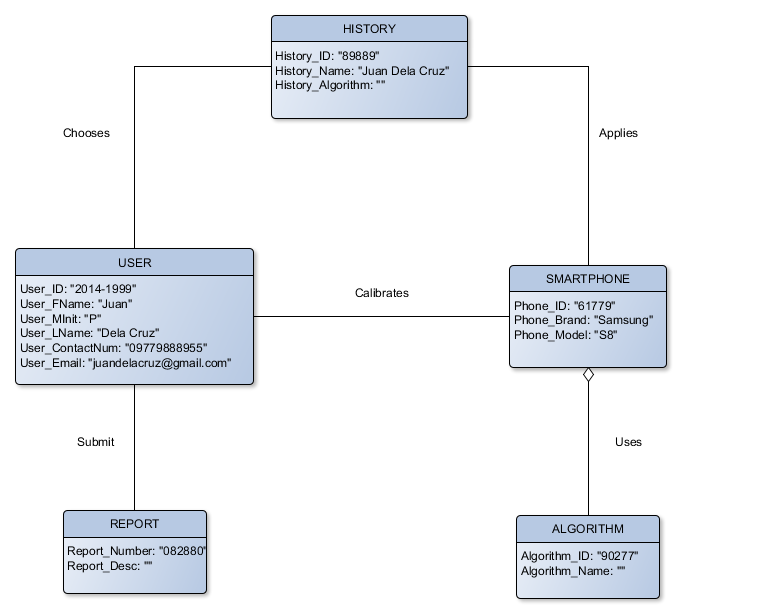
* 1. Use Case Diagram
  2. Activity Diagram
     1. Touch Calibrator



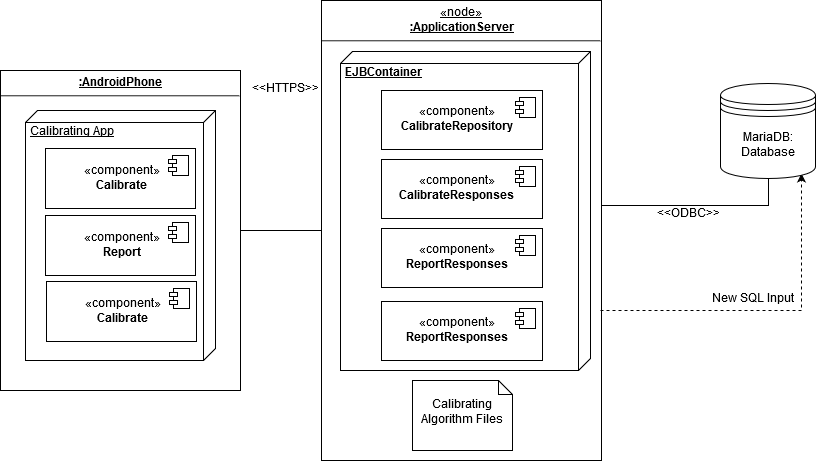
* + 1. Send Feedback



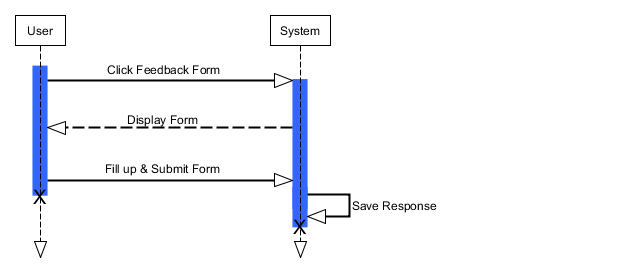
* 1. Object Diagram



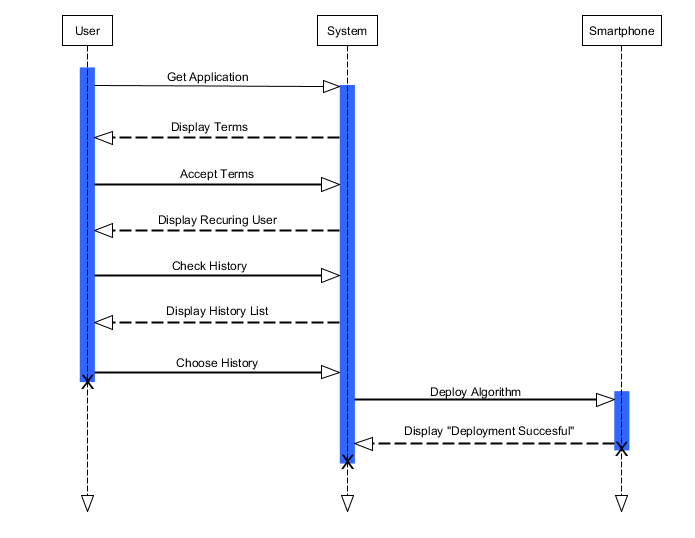
* 1. Deployment Diagram



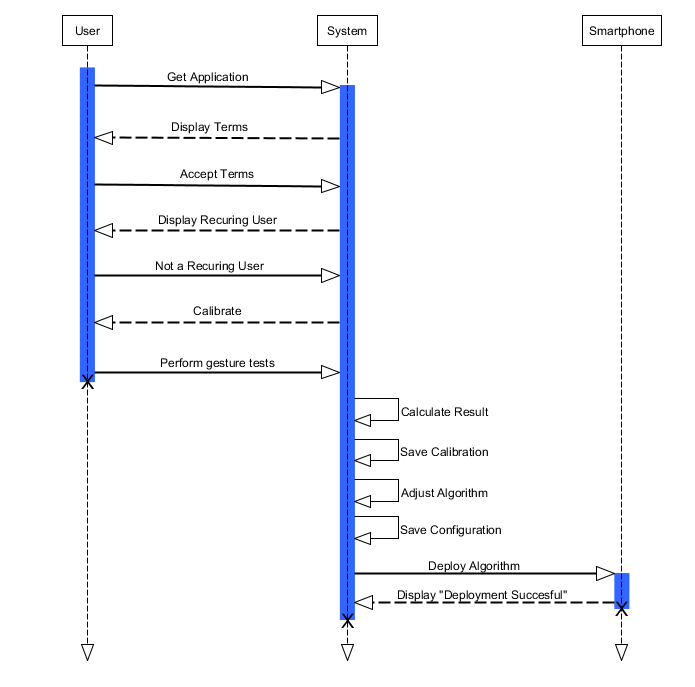
* 1. Sequence Diagram
     1. Report



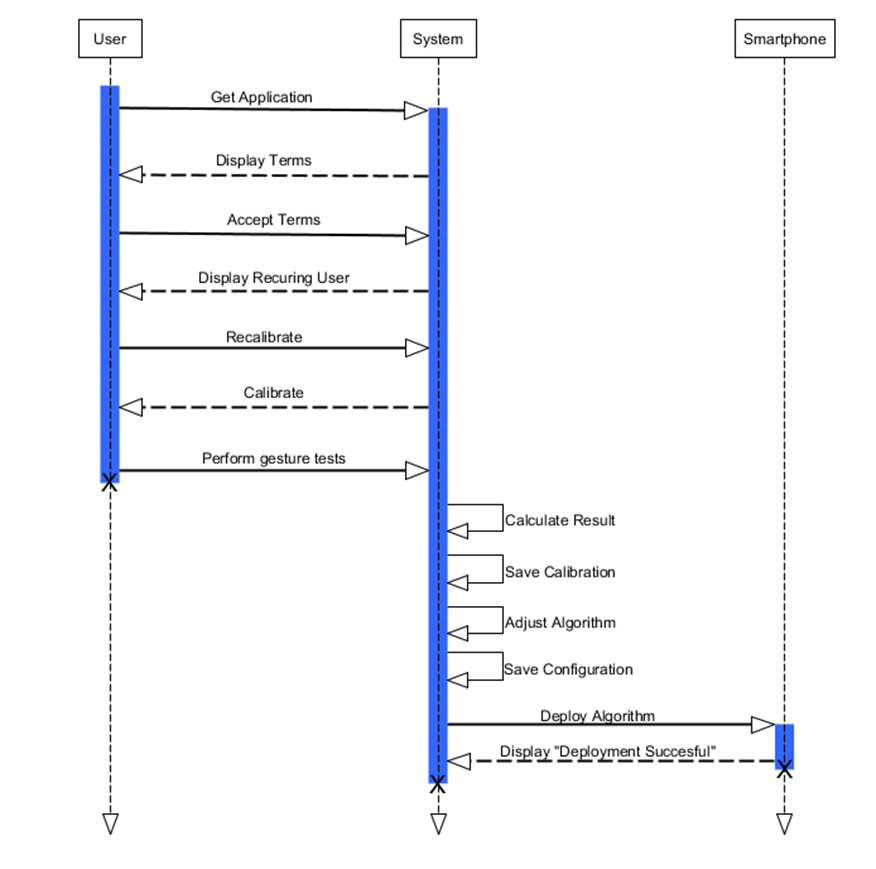
* + 1. Touch Calibration with History



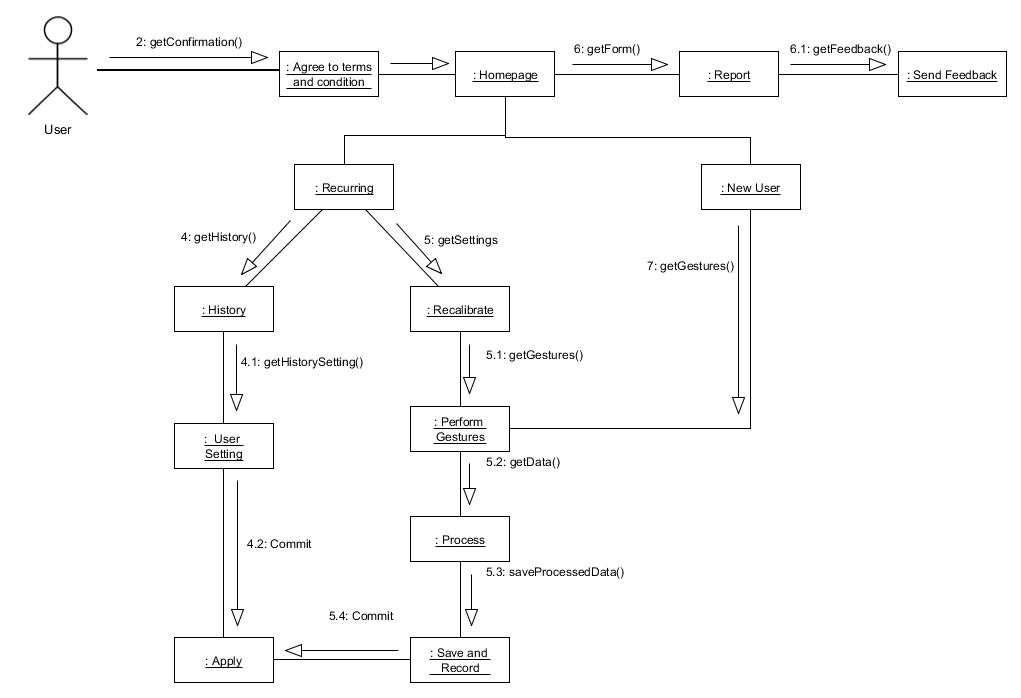
* + 1. Touch Calibration without History



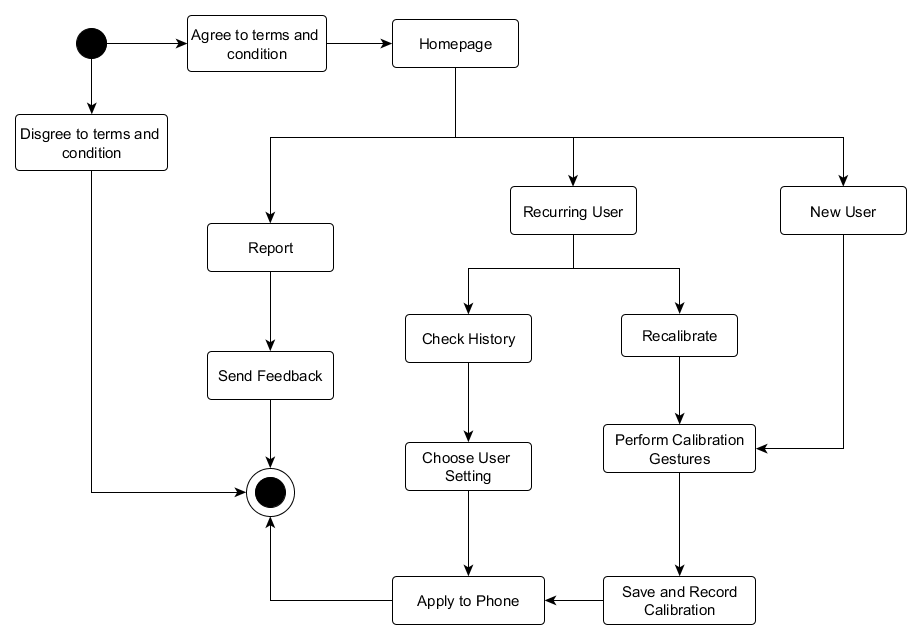
* + 1. Recalibrate



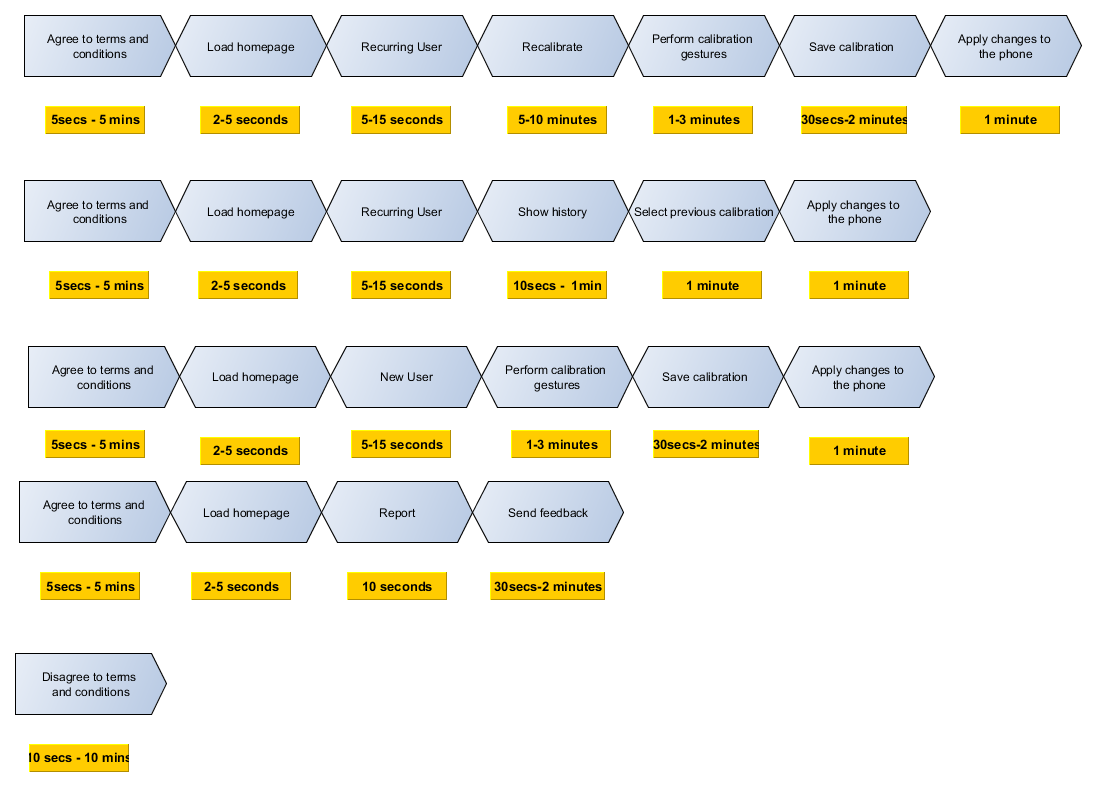
* 1. Communication Diagram



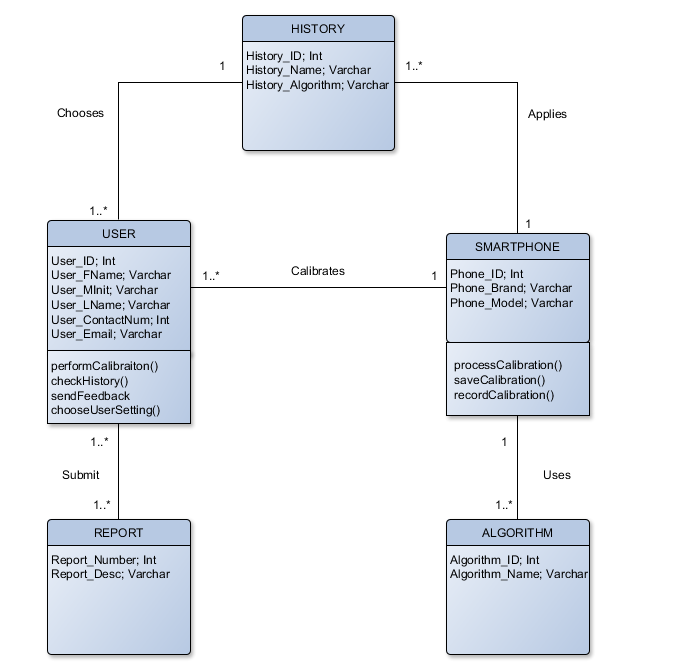
* 1. State Diagram



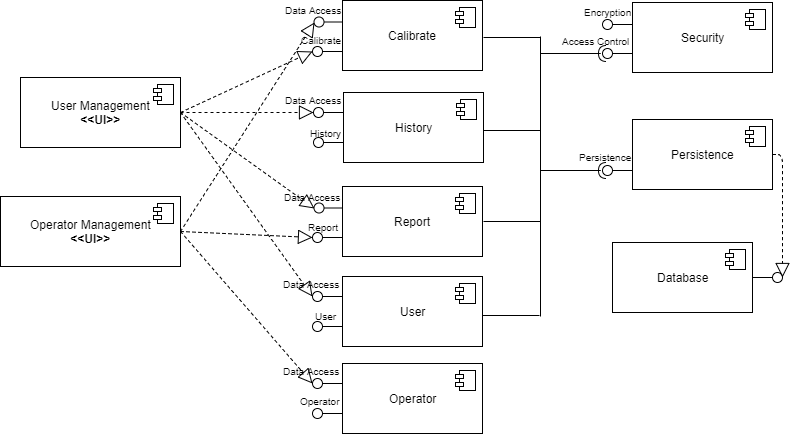
* 1. Timing Diagram



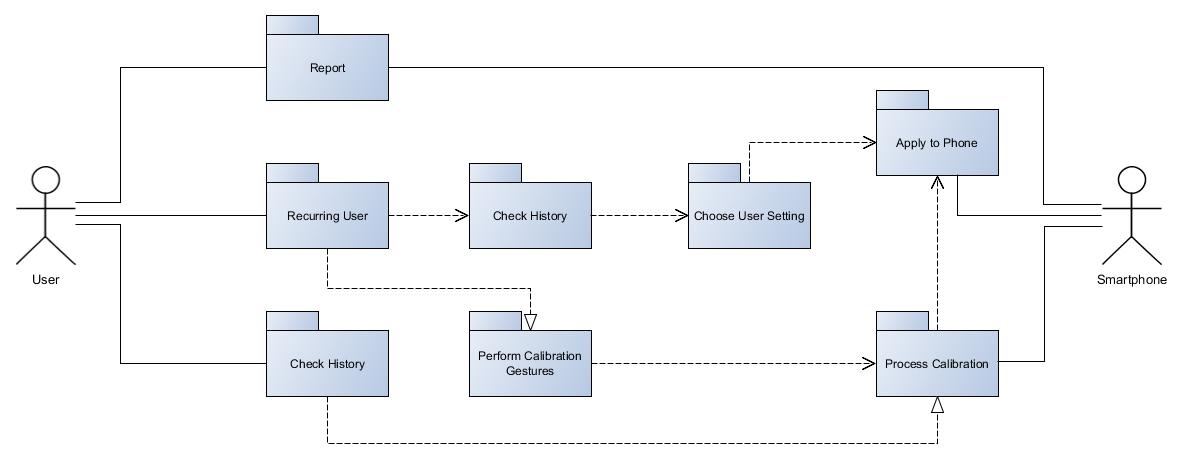
* 1. Class Diagram



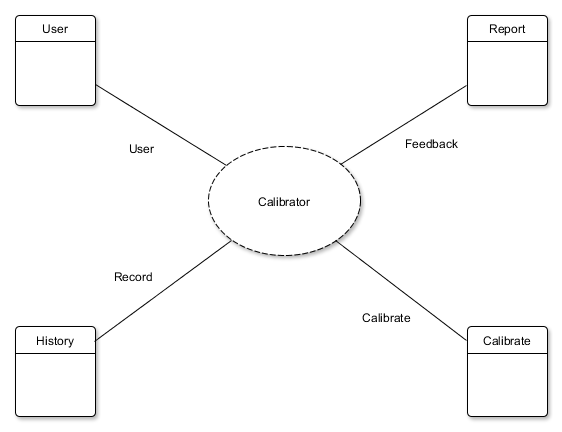
* 1. Component Diagram



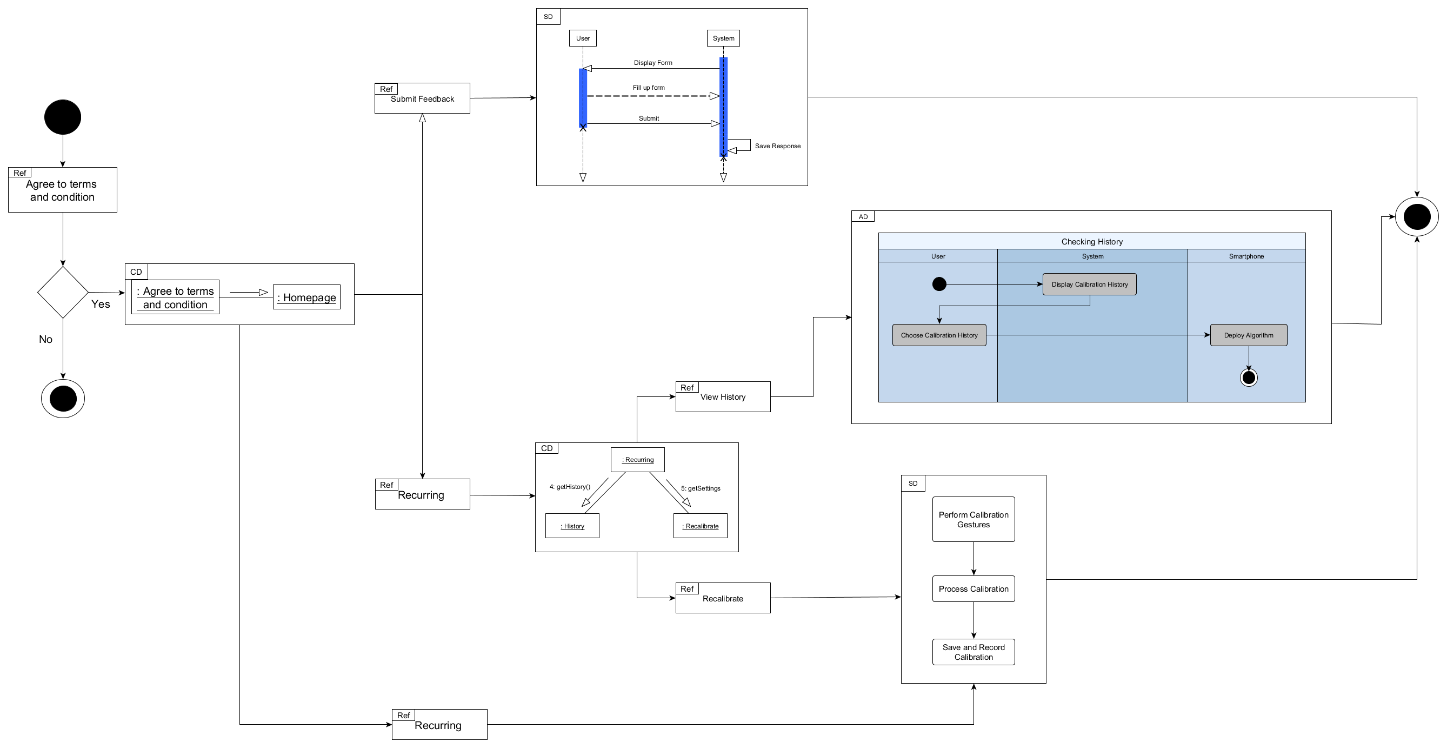
* 1. Package Diagram



* 1. Composite Structure



* 1. Interaction Overview



1. Appendices

* 1. Bibliography

Anthony, S. (2013, September 20). *iPhone 5 vs. Galaxy S4: Apple smartphones trounce Android in overall responsiveness*. Retrieved from Extremetech: https://www.google.com.ph/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwiI0crV9PzVAhXGTLwKHVKtAvkQFggmMAA&url=https%3A%2F%2Fwww.extremetech.com%2Fcomputing%2F167040-iphone-5-vs-galaxy-s4-apple-phones-trounce-android-in-touchscreen-res

Beal, V. (n.d.). *Mobile Operating System (OS)*. Retrieved from webopedia: http://www.webopedia.com/TERM/M/mobile\_operating\_system.html

Campos, D., Williams, G., & Tiganus, A. (2011, November 18). *Why are iOS touch and swipe interfaces so much more responsive than Android?* Retrieved from Quora: https://www.quora.com/Why-are-iOS-touch-and-swipe-interfaces-so-much-more-responsive-than-Android

Cecere, G., Corrocher, N., & Battaglia, R. D. (2014). Innovation and Competition in the Smartphone Industry: Is there a dominant design? *Telecommunications Policy*, 2-13.

Celik, S. (2013, February). Factors Which Influence Key Entry Speed on Hard and Soft Keyboards: Experience, Eye Behaviors, and Finger Movements. Massachusets, United States of Americac.

Cichoń, K., Sobecki, J., & Szymański, J. (2013, June 24-25). Gesture Tracking and Recognition in Touchscreens Usability Testing. Warsaw, Poland.

Cromar, S. (2010, November 29). Smartphones in the U.S.: Market Analysis.

DMC Co., Ltd. (2011). *The Touch Screen Museum.* Retrieved from DMC Co., Ltd.: https://www.dmccoltd.com/english/museum/touchscreens/ad.html

Erickson, J. (2013, October 15). *Improving performance, accuracy, and reliability in touch-screen based applications*. Retrieved from Embedded-computing: http://embedded-computing.com/articles/improving-accuracy-reliability-touch-screen-based-applications/

Ganapati, P. (2010, March 4). *Finger Fail: Why Most Touchscreens Miss the Point*. Retrieved from wired: ttps://www.wired.com/2010/03/touchscreens-smartphones/

Gao, Y., Luo, Y., Chen, D., Huang, H., Dong, W., Xia, M., . . . Bu, L. (2017, May 1-4). Every Pixel Counts: Fine-Grained UI Rendering Analysis for Mobile Applications. Atlanta, Georgia, USA.

Glavinic, V., Ljubic, S., & Kukec, M. (n.d.). Supporting Universal Usability of Mobile Software: Touchscreen Usability Meta-Test. Croatia.

Gray, L. (2014). *How Does A Touch Screen Work?* New York: Gareth Stevens Publishin.

Jaio, C. (2016, July 12). *Smartphone users to double to 90 million in five years*. Retrieved from CNN Philippines: http://cnnphilippines.com/business/2016/07/12/smartphone-users-90-million-in-five-years.html

Kinahan, S. (2015, February 10). Why is the CPU of the iPhone 6 and iPhone 6 Plus not more than Dual-Core 1.4 GHz?

Klein, H. (2013, February ). *Noise Immunity of Touchscreen Devices*. Retrieved from Cypress: http://www.cypress.com/file/120641/download

Lancet, Y. (2012, July 19). *What Are the Differences Between Capacitive & Resistive Touchscreens?* Retrieved from Makeuseof: http://www.makeuseof.com/tag/differences-capacitive-resistive-touchscreens-si/

Lorex Technology. (2012, April 19). *How Touch Screen Monitors Work*. Retrieved from lorextechnology: https://www.lorextechnology.com/self-serve/how-touch-screen-monitors-work/R-sc3100030

McCann, A. T. (2017, January 17). *Okay, but how do touch screens actually work?* Retrieved from ScienceLine: http://scienceline.org/2012/01/okay-but-how-do-touch-screens-actually-work/

*Mobile Usability Testing: Fat Finger Syndrome*. (2015, April 20). Retrieved from Usability 24/7: https://www.usability247.com/blog/mobile-usability-testing-fat-finger-syndrome/

Nosrati, M., Karimi, R., & Hasanvand, H. A. (2012). Mobile Computing: Principles, Devices and Operating Systems. *World Applied Programming*, 399-408.

Pantels, T., Guo, S., & Chabukswar, R. (2014). *Touch Response Measurement, Analysis, and Optimization for Windows\**. Retrieved from Intel : https://software.intel.com/sites/default/files/managed/85/c5/Touch\_Response\_Optimization\_for\_Windows\_Apps.pdf

Ping-yong, L., & Zhong-wen, L. (2014). Study on Calibration Algorithm of Embedded Touch Screen. Sichuan, China.

Rouse, M. (2011, December). *Mobile Operating System*. Retrieved from searchmobilecomputing: http://searchmobilecomputing.techtarget.com/definition/mobile-operating-system

Rouse, M., & Brown, R. (2016, August). *Random Access Memory*. Retrieved from TechTarget - Search Storage: http://searchstorage.techtarget.com/definition/RAM-random-access-memory

Russel, J. (2016, September 1). *Smartphone sales jump 6.5% in Southeast Asia, but sub-$150 devices rule*. Retrieved from TechCrunch: https://techcrunch.com/2016/09/01/smartphone-sales-jump-6-5-in-southeast-asia-but-sub-150-devices-rule/

Schiesser, T. (2012, March 12). *Guide to smartphone hardware (3/7): Memory and Storage*. Retrieved from Neowin: https://www.neowin.net/news/guide-to-smartphone-hardware-37-memory-and-storage

Singh, N. (2016). Study of Google Firebase API for Android. *International Journal of Innovative Research in Computer and Communication Engineering*, 16738-16743.

Sinha, R. (2016, September 29). *How to Fix Touchscreen Problems on Android*. Retrieved from Beebom: https://beebom.com/how-fix-touchscreen-problems-android/

Subido, L. K. (2017, January 24). *Growing 27% in 2016, PH Now Has 60 Million Internet Users*. Retrieved from Entrepreneur Philippines: http://www.entrepreneur.com.ph/news-and-events/ph-now-has-60-million-internet-users-growing-27-in-2016-a36-20170124

technopedia . (n.d.). *Graphics Processing Unit (GPU)*. Retrieved from Technopedia: https://www.techopedia.com/definition/24862/graphics-processing-unit-gpu

*The Java Language: A White Paper.* (1995). Retrieved from Tech-Insider: http://tech-insider.org/java/research/acrobat/9503.pdf

Tomshardware. (2014, November 2). *What determines a smartphone's speed?* Retrieved from Tomshardware : http://www.tomshardware.com/answers/id-2357518/determines-smartphone-speed.html

Tru-vumonitors. (n.d.). *The 5 Types Of Touch Screen Technology.* Retrieved from Tru-vumonitors: http://www.tru-vumonitors.com/images/Touch\_Screen\_Basics.Comparisons.pdf

*TUIO*. (n.d.). Retrieved from Connecting Cities: https://www.aec.at/connectingcities/?p=746

Tyagi, V. (2015, December 3). *Why apple iphone touchscreen responds better than android ones?* Retrieved from LinkedIn: https://www.linkedin.com/pulse/why-apple-iphone-touchscreen-responds-better-than-android-vipin-tyagi

Vertanen, K., Memmi, H., Emge, J., Reyal, S., & Kristensson, P. O. (2015, April). VelociTap: Investigating Fast Mobile Text Entry using Sentence-Based Decoding of Touchscreen Keyboard Input. Seoul, Korea.

Villamor, C., Willis, D., & Wroblewiski, L. (2010, April 15). Touch Gesture Reference Guide.

Williams, A. (2010, January 5). *Touchscreen lowdown -- Capacitive vs Resistive*. Retrieved from Knowyourmobile: http://www.knowyourmobile.com/products/7401/touchscreen-lowdown-capacitive-vs-resistive

Yechiam, E., Erev, I., Yehene, V., & Gopher, D. (2003, December 1). Melioration and the Transition from Touch-Typing Training to Everyday Use. Haifa, Israel.

Yeow, P., & Balakrishnan, V. (n.d.). A Study of the Effect of Thumb Sizes on Mabile Phone Texting Satisfaction. *Journal of Usability Studies*, 118-128.

Young, J. G., Trudeau, M. B., Odell, D., Marinelli, K., & Dennerlein, J. T. (2013). Wrist and shoulder posture and muscle activity during touch-screen tablet use: Effects of usage configuration, tablet type, and interacting hand. *Work* , 59-71.