**Gesture Based Calibration in Android Touchscreen Devices**

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 17, 2017

Table of Contents

[Abstract 4](#_Toc490593521)

[I. Introduction 5](#_Toc490593523)

[1.1. Project Context 5](#_Toc490593524)

[1.1.1. The Challenges 5](#_Toc490593525)

[1.1.2. The Opportunity 5](#_Toc490593526)

[1.1.3. Purpose and Description 5](#_Toc490593527)

[1.1.4. Background of the Problem 5](#_Toc490593528)

[1.2. Objectives 7](#_Toc490593529)

[1.3. Scope and Limitation 7](#_Toc490593530)

[II. Literature Review 8](#_Toc490593531)

[2.1. Smartphone Usage 8](#_Toc490593533)

[2.2. Usability Testing 8](#_Toc490593537)

[III. Technical Background 9](#_Toc490593538)

[3.1. Theoretical Framework 9](#_Toc490593540)

[3.1.1. Usability Testing 9](#_Toc490593541)

[3.1.2. Touch Calibration 9](#_Toc490593542)

[3.1.3. Gesture Recognition 10](#_Toc490593543)

[3.2. Conceptual Framework 11](#_Toc490593544)

[3.3. Definition of Terms 11](#_Toc490593545)

[IV. Design and Methodology 12](#_Toc490593546)

[4.1. Nature of Study 12](#_Toc490593548)

[4.2. Sampling Technique 12](#_Toc490593549)

[4.3. Population and Sample 12](#_Toc490593550)

[4.4. Research Instruments 12](#_Toc490593551)

[4.5. Gathering Procedures 13](#_Toc490593552)

[4.6. Data Analysis 13](#_Toc490593553)

[V. UML Diagrams 15](#_Toc490593554)

[4.1. Use Case Full Description 15](#_Toc490593556)

[4.1.1. Accept Terms and Conditions 15](#_Toc490593557)

[4.1.2. Calibrate 16](#_Toc490593558)

[4.1.3. Save Configuration 17](#_Toc490593559)

[4.1.4. Send Feedback 18](#_Toc490593560)

[4.1.5. Restore Defaults 19](#_Toc490593561)

[4.1.6. Configure Touch Algorithm 19](#_Toc490593562)

[4.2. Use Case Diagram 20](#_Toc490593563)

[4.3. Activity Diagram 21](#_Toc490593564)

[4.3.1. Touch Calibrator 21](#_Toc490593565)

[4.3.2. Send Feedback 22](#_Toc490593566)

[4.4. Object Diagram 23](#_Toc490593567)

[4.5. Deployment Diagram 24](#_Toc490593568)

[4.6. Sequence Diagram 24](#_Toc490593569)

[4.6.1. Report 24](#_Toc490593570)

[4.6.2. Touch Calibration with History 25](#_Toc490593571)

[4.6.3. Touch Calibration without History 26](#_Toc490593572)

[4.7. Communication Diagram 27](#_Toc490593573)

[4.8. State Diagram 28](#_Toc490593574)

[4.9. Timing Diagram 29](#_Toc490593575)

[4.10. Class Diagram 30](#_Toc490593576)

[4.11. Component Diagram 31](#_Toc490593577)

[4.12. Package Diagram 31](#_Toc490593578)

[4.13. Composite Structure 32](#_Toc490593579)

[4.14. Interaction Overview 32](#_Toc490593580)

[V. Appendices 33](#_Toc490593581)

[5.1. Bibliography 33](#_Toc490593583)

Abstract

The versatillity and functionality of touch screen is highly commended by researchers, as it is an absolute coordinate system wherein a point is selected directly as opposed to relative positioning systems such as a mouse. However, due to software and hardware incompatibilities, inaccuracies and flaws exists in the system. This study aims to study the human computer interaction between touchscreen smartphone and its user(s). The proponents will investigate the main causes of inaccuracies and inefficiencies regarding the touch registers, and will propose a calibration app that can also serve as a solution for smartphone manufacturers with regards to the said issue.

1. Introduction
   1. Project Context
      1. The Challenges

The touch registers of touchscreen smartphones in the market is highly dependable on the user as the device should scale to the users’ hand size to achieve more accurate touch input**.**Varying factors also exists not only in the hardware part, but also in the software as the touch responsiveness also differ depending on the current running operating system. In addition, according to the Agawi Touchmarks, smartphones running iOS is 2.5 times faster at responding to touch than Google Android. **The Touchmarks benchmark measured the minimum app response time scores on the lightest possible apps, it measures how immediately they respond on a given device.** Also, virtual keyboards, which are software components play a huge role in the typing accuracy of a smartphone.

* + 1. The Opportunity

This research can help smartphone manufacturers on improving their devices by increasing the satisfaction experienced by their customers through the custom-calibrated touch registers.

* + 1. Purpose and Description

**This research serves as a possible solution to the touchscreen issue of Android devices to further increase the consumer satisfaction rate with regards to the manufacturers’ smartphones.** This research aims to develop an android application that will calibrate the touch screen of their mobile phone/s depending on the user’s usage to further increase accuracy and precision of future touch registers after performing a series of evaluating tests. Also, this research aims to know the causes of the malfunctions and inaccuracies of android touch registers.

* + 1. Background of the Problem

Smartphones are hand held devices that allows the user to make calls, it has a special feature that can only be found on a personal computer before. Smartphones have the capability to connect to the internet, send and receive e-mails, and edit office documents by downloading and running apps from digital distribution platforms such as Google Play and App Store.

Smartphones are hand held devices that functions like a computer. They can receive and make calls, create and receive messages. This technology is a type of visual display that enables the users to interact with an electronic device by touch.

Also, these smartphones have the capability to connect to the internet, making them capable of online communication, e-commerce, information retrieval, etc.. It has been constantly developing by introducing new features and innovations throughout the years. In fact, a lot of firms have been competing in the smartphone industry and the competition never stops.

Touch screen added dynamics and enabled development of many applications. The utilization of touch screen jump from mobile devices to life size monitors. Smartphones, gaming consoles, Automatic Teller Machines (ATM), Point of Sales (POS), and digital signage’s using touch screen is rampantly visible in our daily lives. As compared to other human machine interfaces, touch screens have various advantages as opposed to physical such as the reduction of physical requirements to deploy it, also the maintenance that comes with the use of it.

The same condition also affects touch registers in touch screens; a thin layer of oil or other debris sitting on the screen can sometimes be registered as touch, and can also mislead the system in reading multi-touches. The developers of Android didn’t overlook this, they made an OS touch diagnostic measure, by typing “**\*#0\*#**”, a compilation of system diagnostic measures is listed, and one of them is the touch diagnosis wherein there are boxes and the user should go through and touch the boxes and the box will be filled with a color signifying a recorded touch. Also, developers made the information about the touch registers developer friendly, by having the option to enable a screen overlay that can display and track the current touch data.

Calibration is the action or process of calibrating an instrument or experimental readings. Calibration

* 1. Objectives

This research aims improve the user experience in touch screen smartphones; increasing accuracy and precision of touch registers made by the user in his smartphone by prompting an initial calibration (and also if the user deemed necessary; re-calibration). The results of the initial calibration will alter and prompt the system to adapt depending on the said results gathered from the user. Thus, the objectives are…

1. to increase user satisfaction upon product usage to help the system satisfy the varying users when it comes to the touch screen display;
2. to produce a touch calibrating software wherein the system will adapt on the users’ usage, increasing touch accuracy and precision;
3. to determine the causes of the inaccuracy of touch registers on a touch screen smartphone.
   1. Scope and Limitation

This study focuses on human computer interaction, identifying the causes of inefficiencies and inaccuracies of smartphone touch registers. Due to the extremities of resources, this research will only be focusing on Android OS devices.

1. Literature Review
2. 1. Smartphone Usage

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 AR devices, partnered with their ultra-high definition (UHD) displays and resonating Dolby audio.

The Philippines is the third largest and fastest growing market for smartphones in South East Asia with the first being Indonesia (Russel, 2016). 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 5 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 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 PlayStore. These virtual keyboards support functionalities such as glide typing, auto-correction, word prediction, and voice typing which helped different users increases their typing accuracy.

Text messages, also known as SMS (Short Message Service) refers to the exchanging of composed short character based messages (160 characters) to another mobile phone subscriber. Texting is the most widely used form of mobile data service up to date. Problems Further studies concluded that the thumb size of the user has a significant effect on his mobile phone texting satisfaction. (Yeow & Balakrishnan).

In the analysis of different (Vertanen, Memmi, Emge, Reyal, & Kristensson, 2015)

2. 2. Usability Testing

The Touchscreen Usability Meta-Test, proposed by Glavinic, Ljubic, & Kukec focuses on measuring the u**s**er performance in numerical figures of errors and time completion of a certain action. It consists of 5 test scenarios, namely *1 – Target Size*, *2 – Object Search*, *3 -Boundary Objects*, *4 – Drag&Drop*, and lastly, *5 – Soft Keystroking*. While performing these test scenarios, the user interaction with the software is logged, that the system will process in order to calculate performance time, identify errors, and determine the users’ behavior. (Glavinic, Ljubic, & Kukec)

1. Technical Background
2. 1. Theoretical Framework
      1. Usability Testing

Usability in the sense of software engineering refers to the degree of difficulty in which a software can be used to perform prescribed tasks as well as measuring its effectiveness, efficiency, and satisfaction. Upon the birth of a new smartphone software, there are two types of testing environments used for the testing and evaluation of its touchscreen responsiveness, using the mobile device itself and in a virtual testing environment, with the help of emulators and touch sensitive monitors (Glavinic, Ljubic, & Kukec). Usability issues in certain activities such as the virtual keyboard usage, dragging gestures, visual search and selection arises in touch screen devices.

* + 1. Touch Calibration

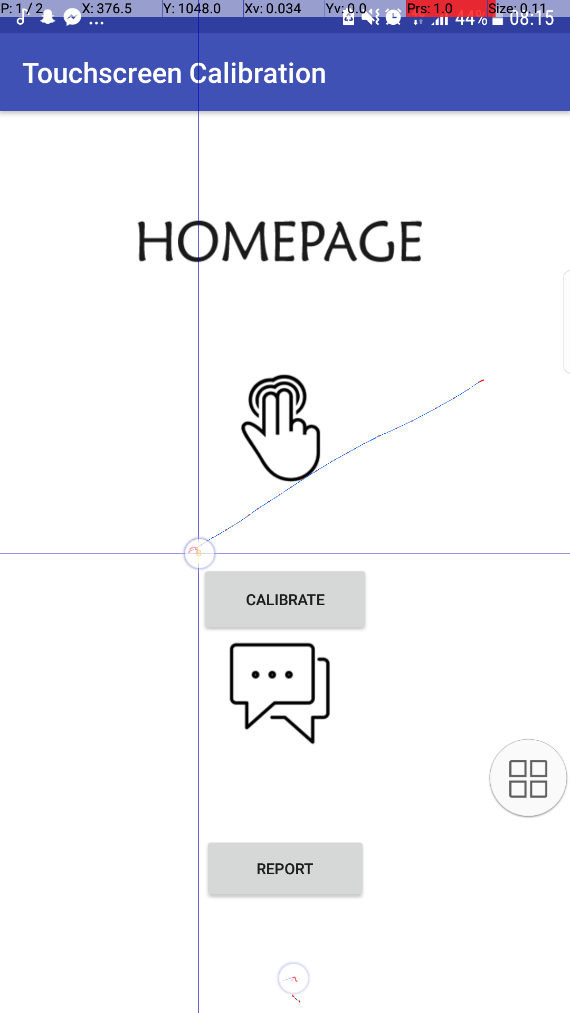
The higher number of AC/DC converter bits, the higher touch screen accuracy. However, Liquid Crystal Displays (LCD) doesn’t necessarily need high levels of A/D bits. A/D resolution is computed as N = 2M where M = number of bits; therefore, for a 1920x1080 display, a 11-bit A/D converter is already sufficient. Resistive touch screen requires frequent calibration as parameters constantly change. The two main factors affecting the touch screen and LCD as stated by Ping-yong and Zhong-wen are the errors in rotation and translation that was introduced upon the installation of the touch screen to the LCD, and the fact that the touch screen and LCD are two different physical aspects. Mechanical errors in touchscreen are generated upon installation of touch screen on the LCD, this includes displacement errors and rotation errors (Ping-yong & Zhong-wen, 2014). They proposed a five-point/multi-point calibration algorithm, which is divided into six parts: Initialization, Screen Display, Access of Coordinates, Calculation of Calibration Coefficient, Deviation Correction, Storage of Calibration Coefficient (Ping-yong & Zhong-wen, 2014).

* + 1. Gesture Recognition

Touch gestures refers to the movement in a touchscreen using a finger, or an object (stylus), and with the rapid innovations in computing devices, most human to computer interactions utilizes these touch gestures. Touch gestures are then categorized into the number of fingers used: Single Touch, Double Touch, and Multi Touch (Villamor, Willis, & Wroblewiski, 2010). Certain standards were set by the proponents to recognize the different categories of touch gestures. When time between pressing a finger and detachment is less than 351ms, and the slide of the finger is greater than 100 pixels vertically, it is recognized as *Flip*. When the finger moved at least 50 pixels in either direction from the starting point, it is recognized as *Dragging*. *Double Tapping* is recognized when the first finger was pressed not more than 501ms, and the next touch occurs in not more than 50 pixels in any direction. To gather data from the touchscreen, the proponents used the application is specially designed to work with OpenSoundControl/TUIO (a protocol for table based tangible user interfaces) touch events. Data such as coordinates, touch type, acceleration, supports touch ID are recorded (Cichoń, Sobecki, & Szymański, 2013).

Under the developer tools in the settings of an android phone, there exists an option called “Pointer Location” that trails and displays information about the users’ latest, previous touch register. The information it displays consists of the following:

1. The maximum number of touch points detected since detachment. (P)
2. The coordinates (horizontally and vertically) of the first touch point registered. (X & Y)
3. The velocity of the first registered touch point. (Xv & Yv)
4. The pressure applied to the screen. (Prs)
5. The size of the touch point. (Size)
6. The change in coordinates (horizontally and vertically) from the initial touch point coordinate up until the detachment point. (dX & dY)



Statistics

Crosshair

Touch Trail

Figure 1. Pointer location in a phone running Android v7.0

These information resets, along with the visual trail of your latest, previous touch register when the user touches the screen. (Ruffer, 2013-2016)

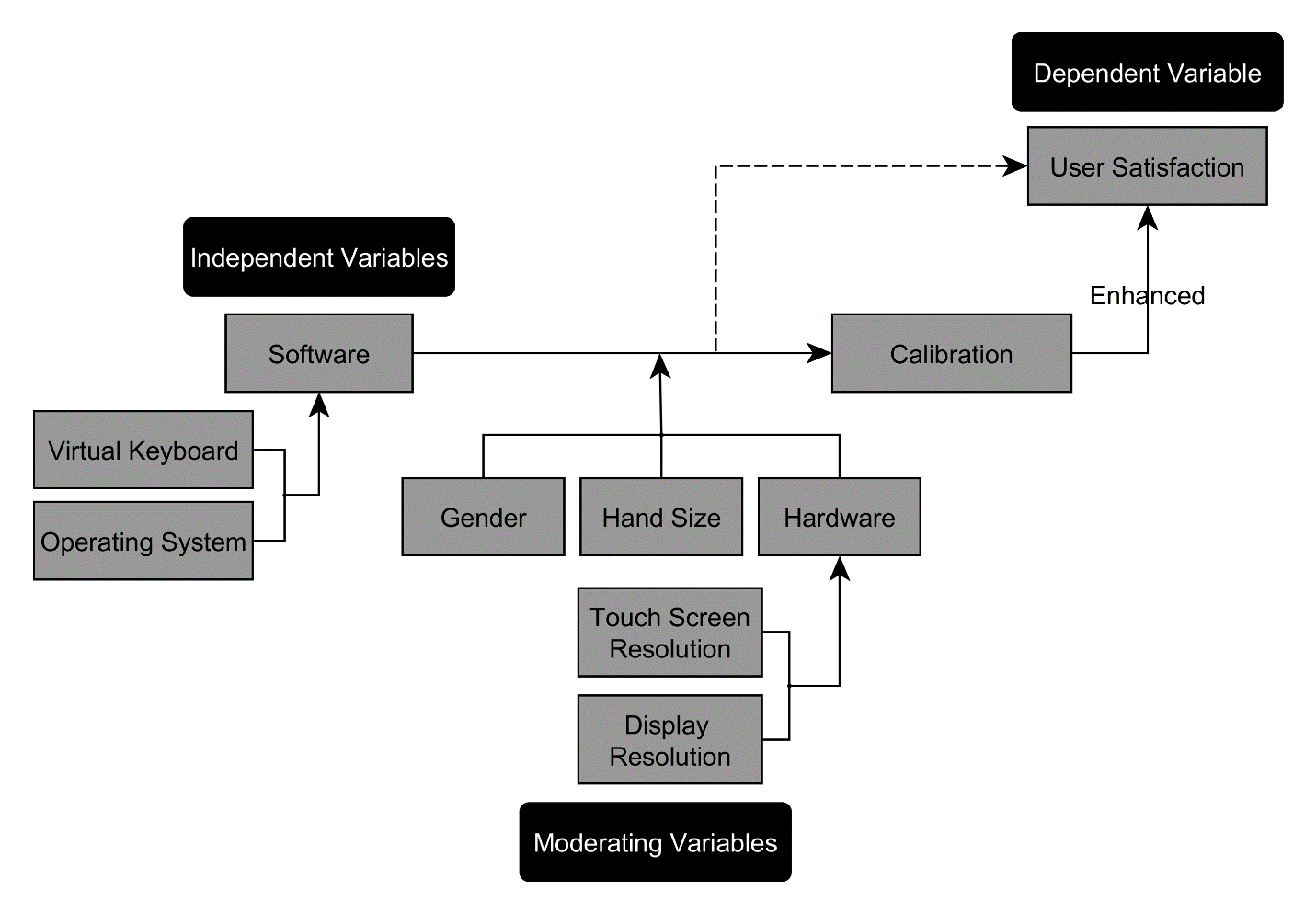
* 1.  Conceptual Framework

Figure 2. Conceptual Framework

The researchers theorize that factors which directly affect the user satisfaction when using a smartphone are the software factors, which include virtual keyboards, and the operating system itself, along with the specific touch registering algorithm. Additional moderating variables affect the user satisfaction of a smartphone user are its biological attributes such as the users’ hand size and gender, along with hardware factors such as the smartphones RAM, touch screen resolution, and display resolution.

* 1. Definition of Terms
* Virtual Keyboard – Software component that allows its user to enter text into the system.

1. Design and Methodology
2. 1. Requirement Analysis

The researchers are going to conduct a series of tests to gather data from the respondents. The first test contains an application where it displays an object randomly at a time, while the user's objective is to hit the objects as accurately, and efficiently as possible. The second test is somehow similar to the first one but the object that the user must target is surrounded by multiple distractions. The third test is where the participants must repetitively hit the given targets which are randomly placed near the edges of 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 fifth test is where the participants must be as fast and accurate at using the keypad. The last or sixth test contains a circle on the canvas which the participants must trace. These tests intend to obtain data such as the accuracy, speed, error ratio, and response time which are essential in defining the factors relevant to the History entity in the paper’s diagrams.

* 1. Requirement Documentation
  2. Gap Analysis

The researchers’ current prototype working features are Perform Calibration and Report. Although these features are working, there are still missing parts of it. One of the missing parts that is supposed to be included in the system is a database for all the data to put into. Currently, the calibration tests being performed are not stored in a database, the calibration results are just shown on the screen.

* 1. Design of software, Systems, Products, and Processes

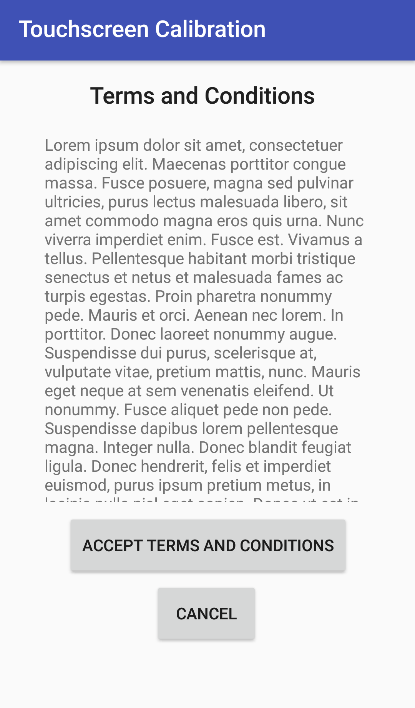


Figure 3: Terms and Conditions

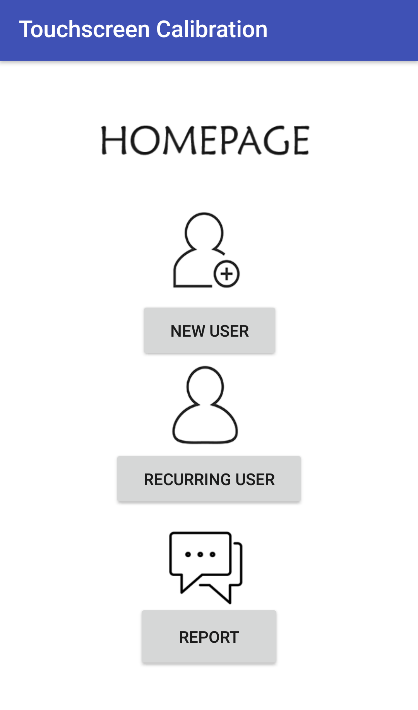


Figure 4: Homepage

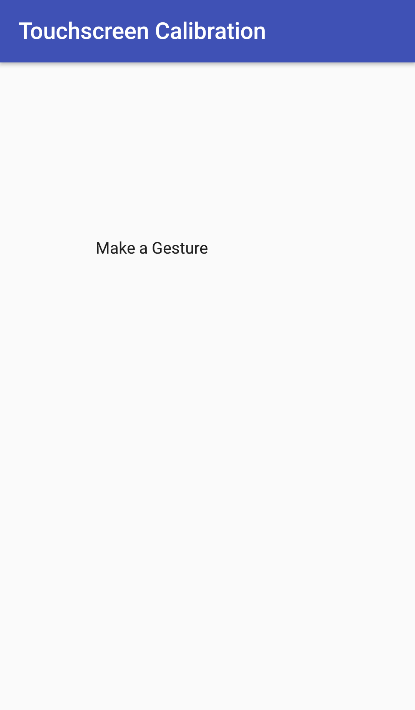
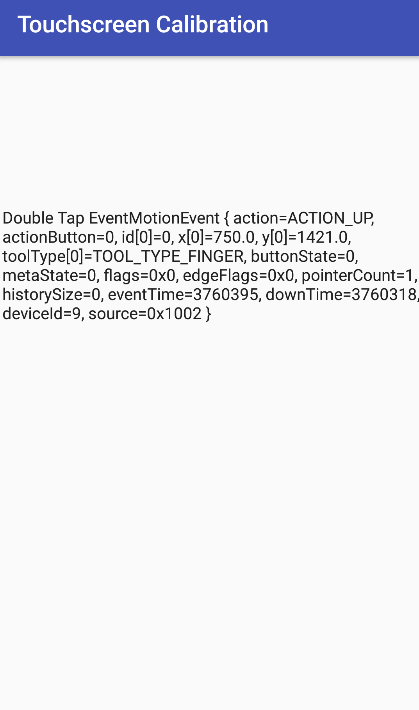
    

Figure 5: Gestures

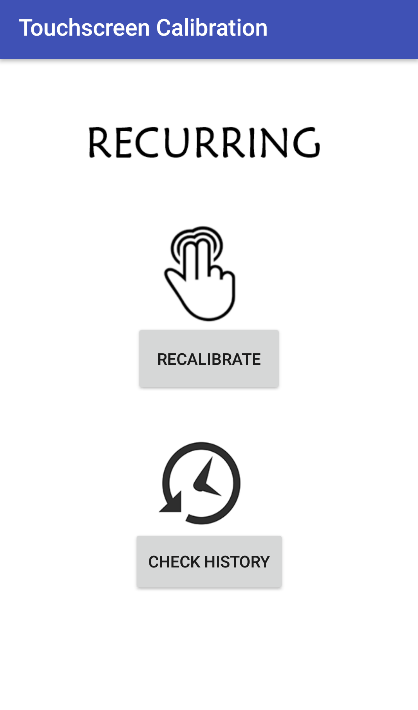


Figure 6: Recurring

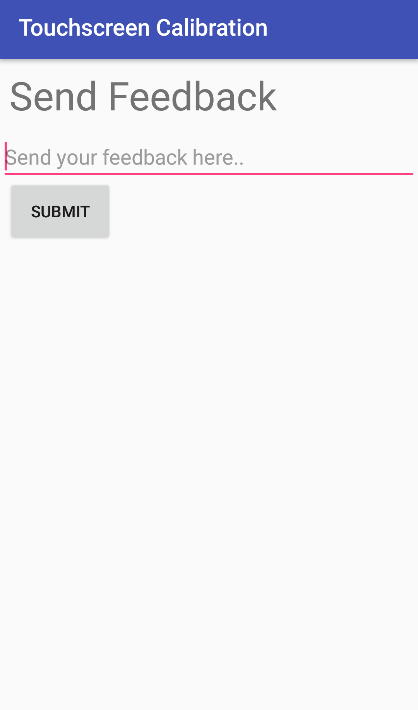
 

Figure 7: Send Feedback

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 |  | |
| 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

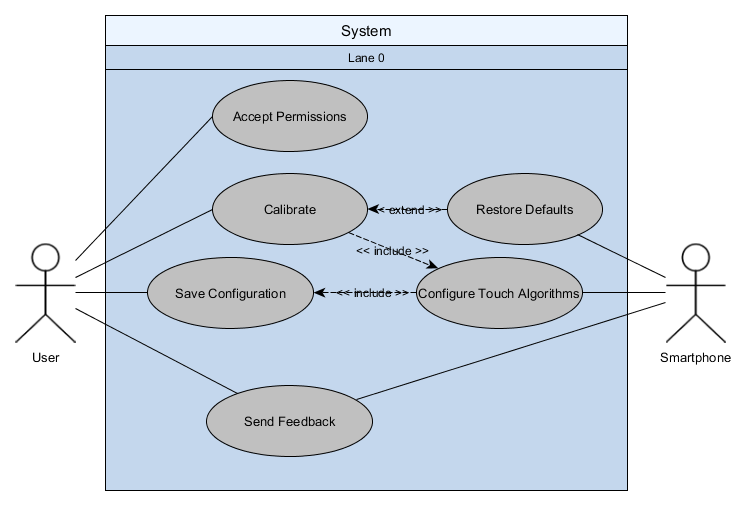
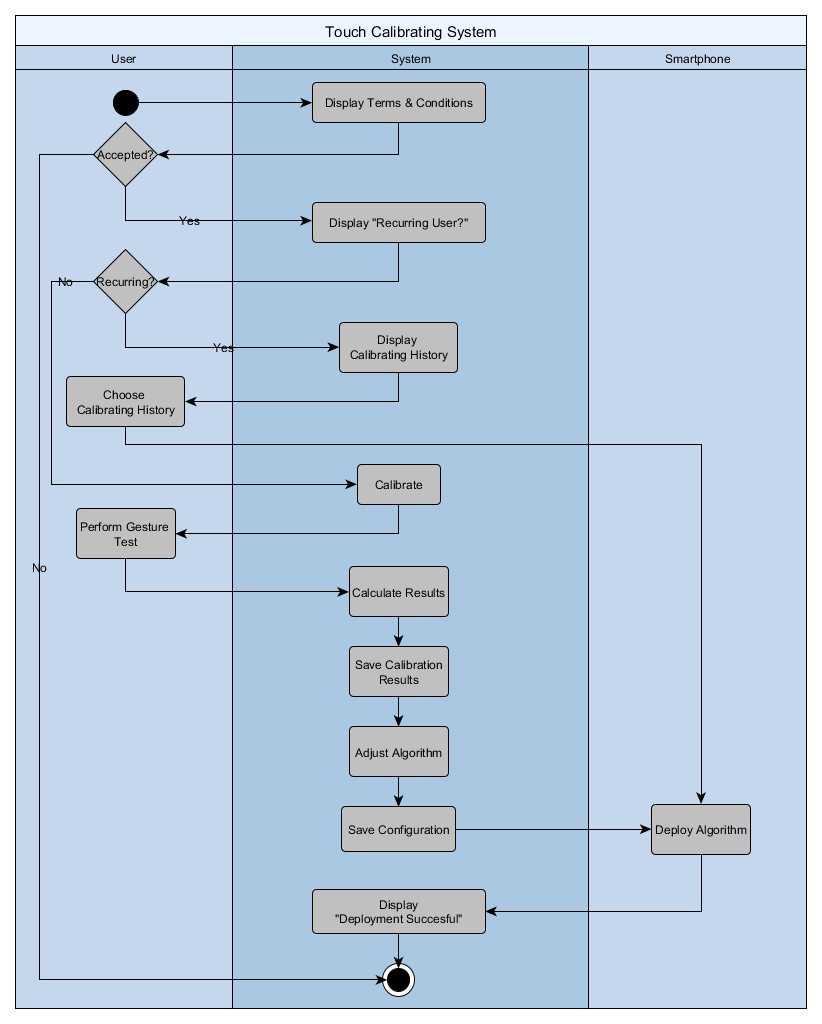
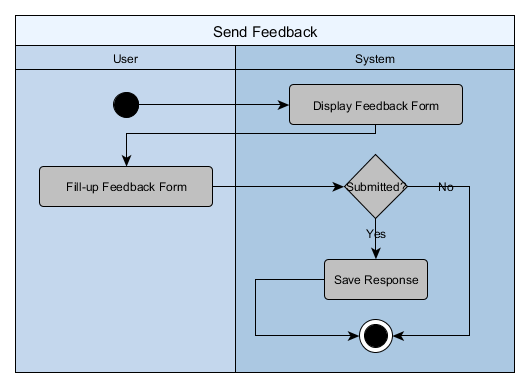
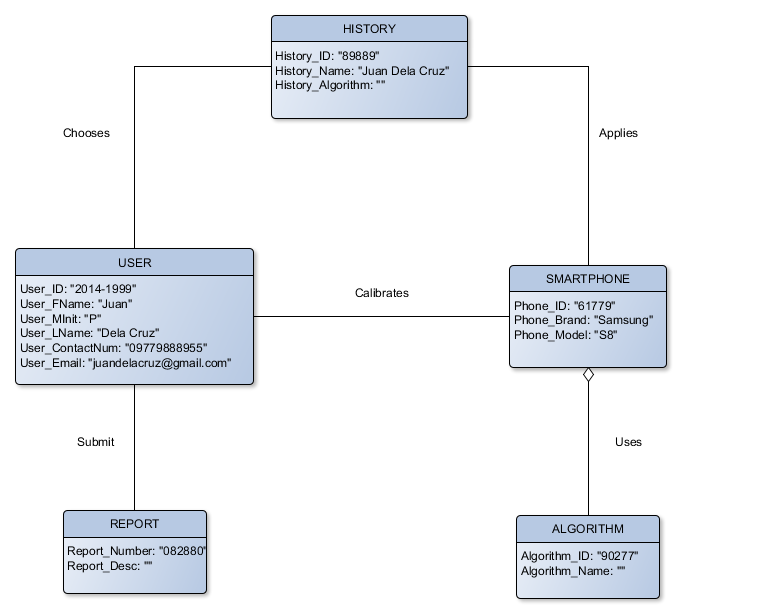
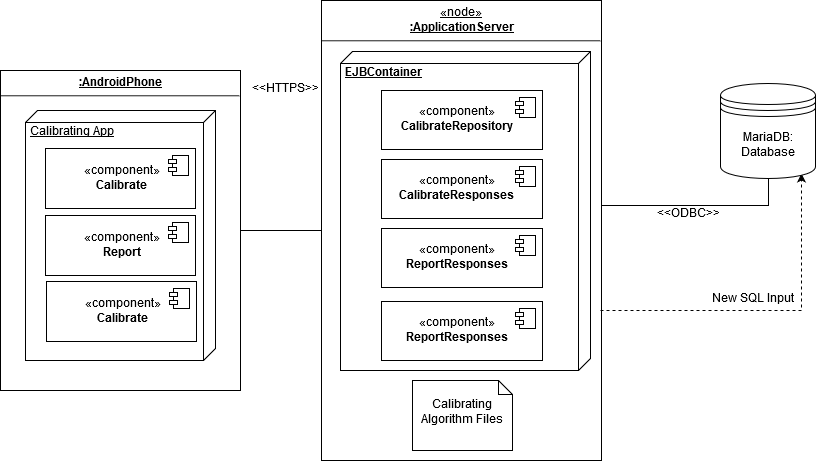
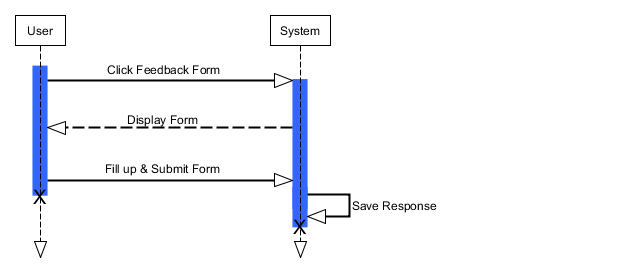
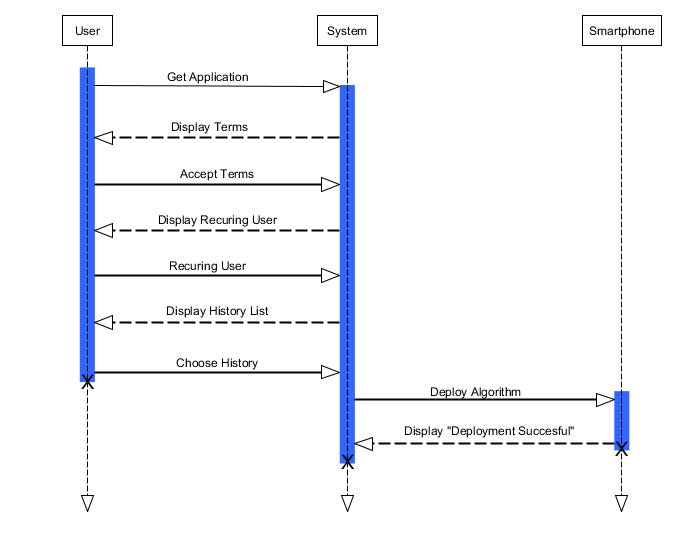
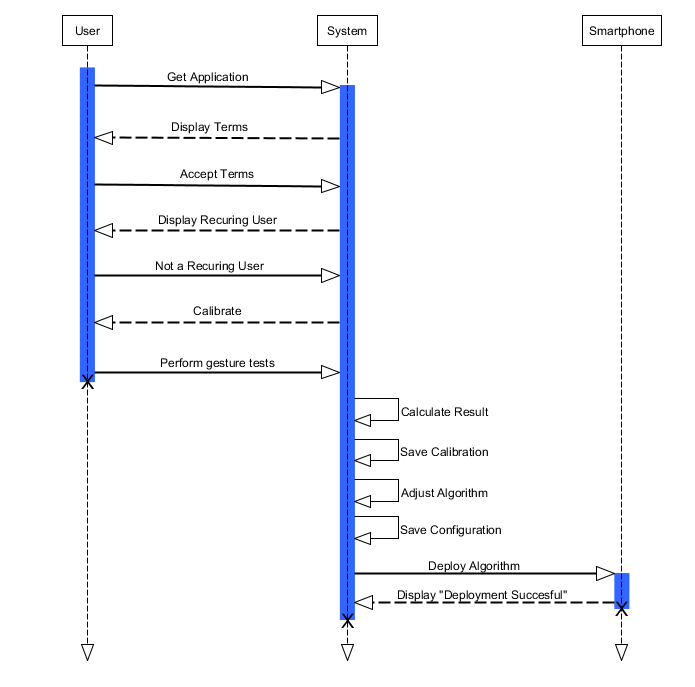
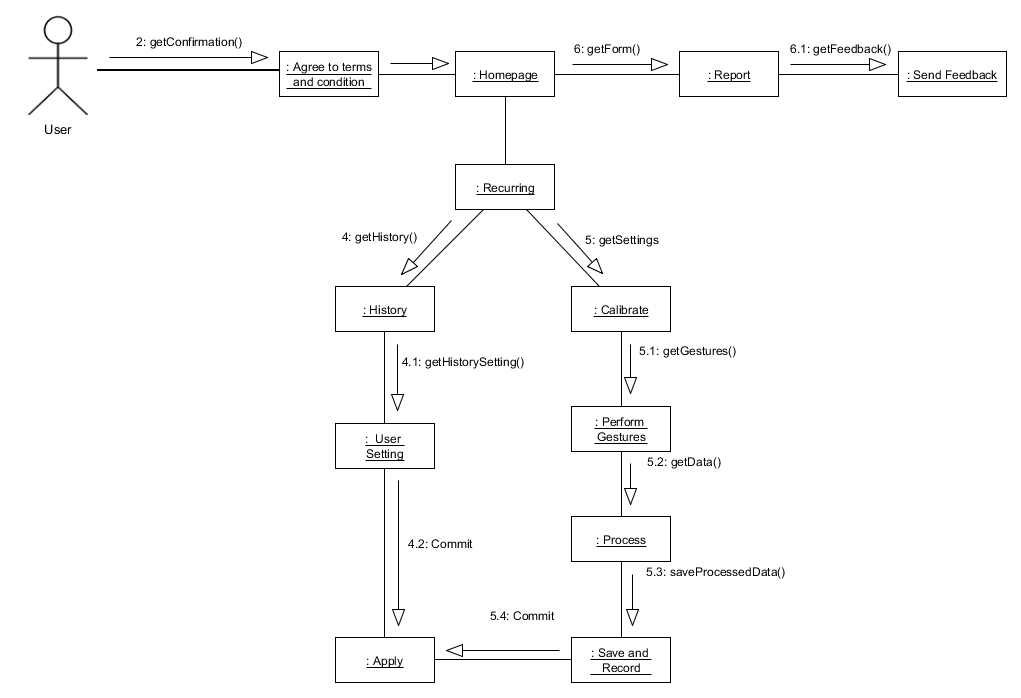
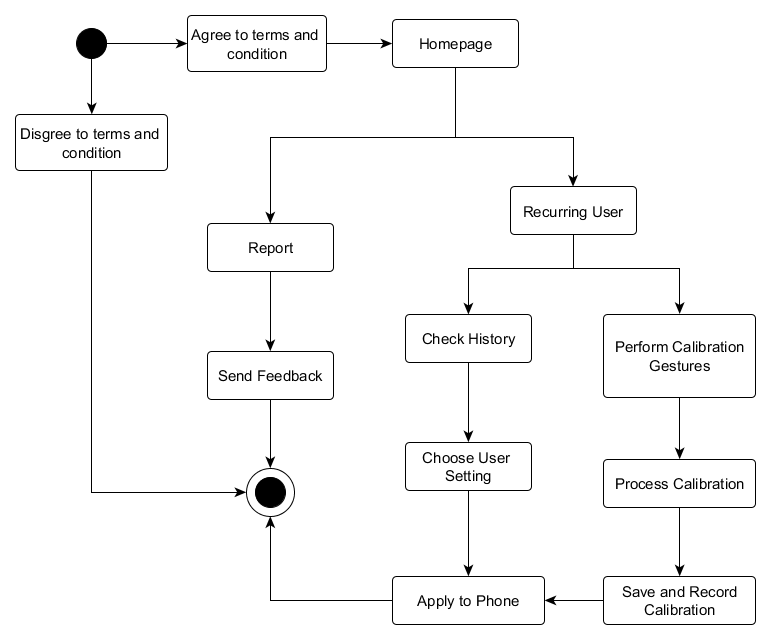
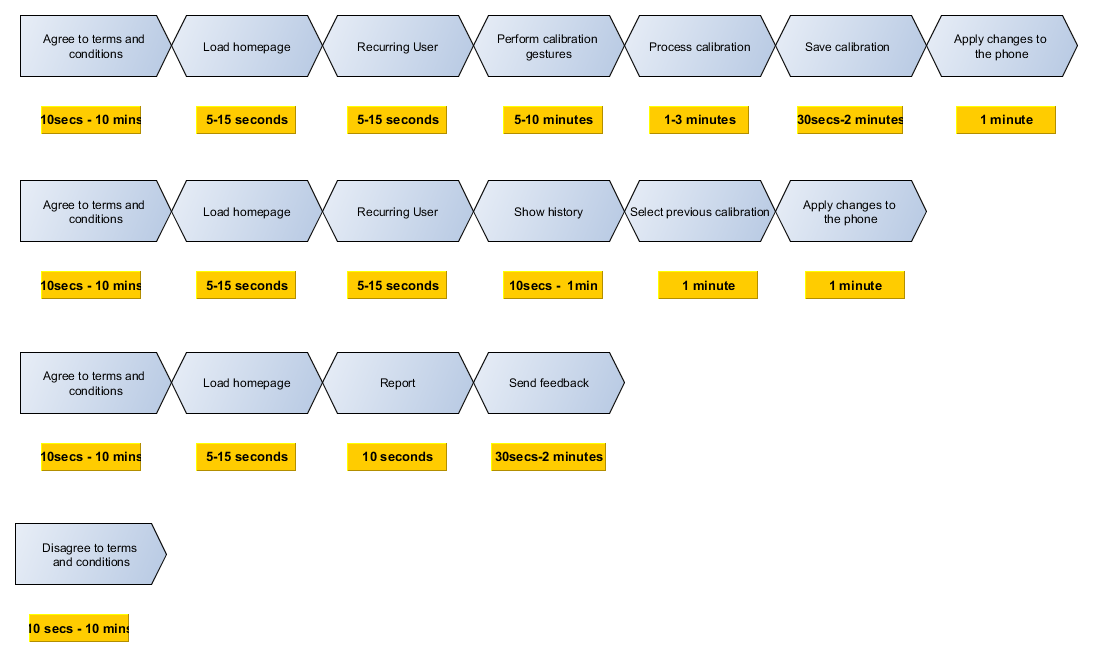
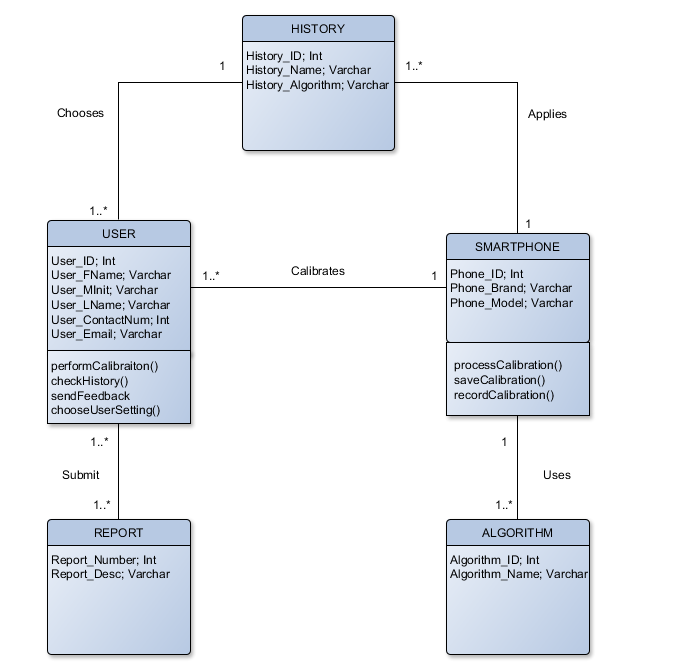
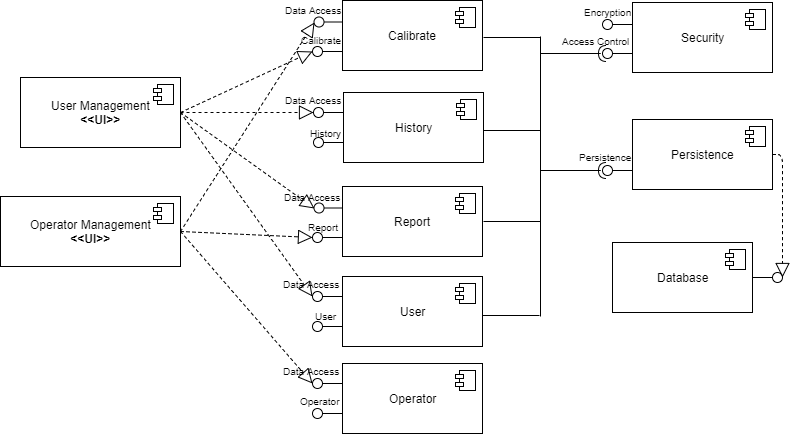
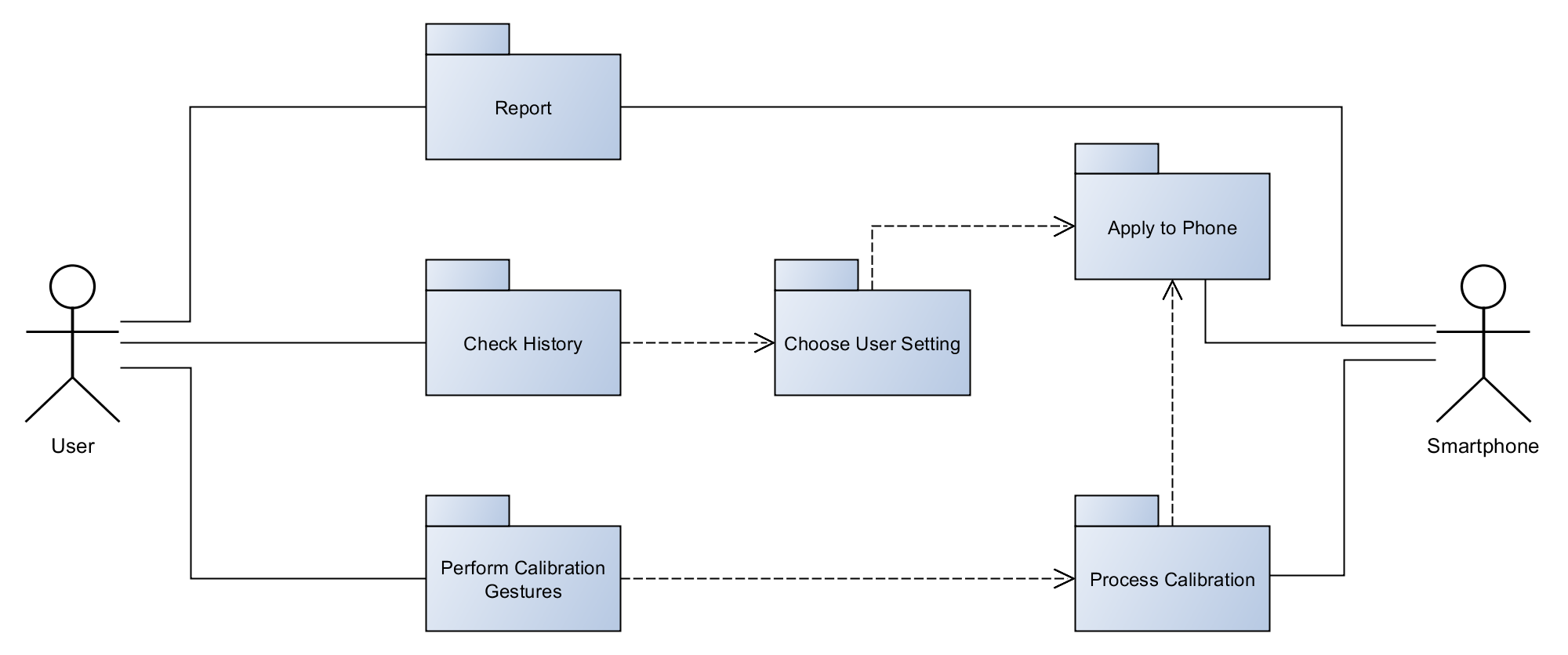
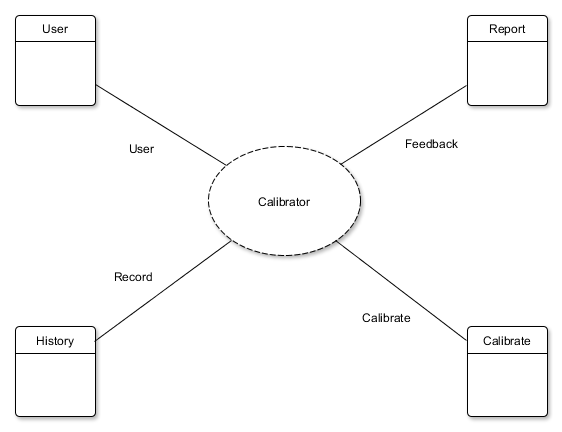
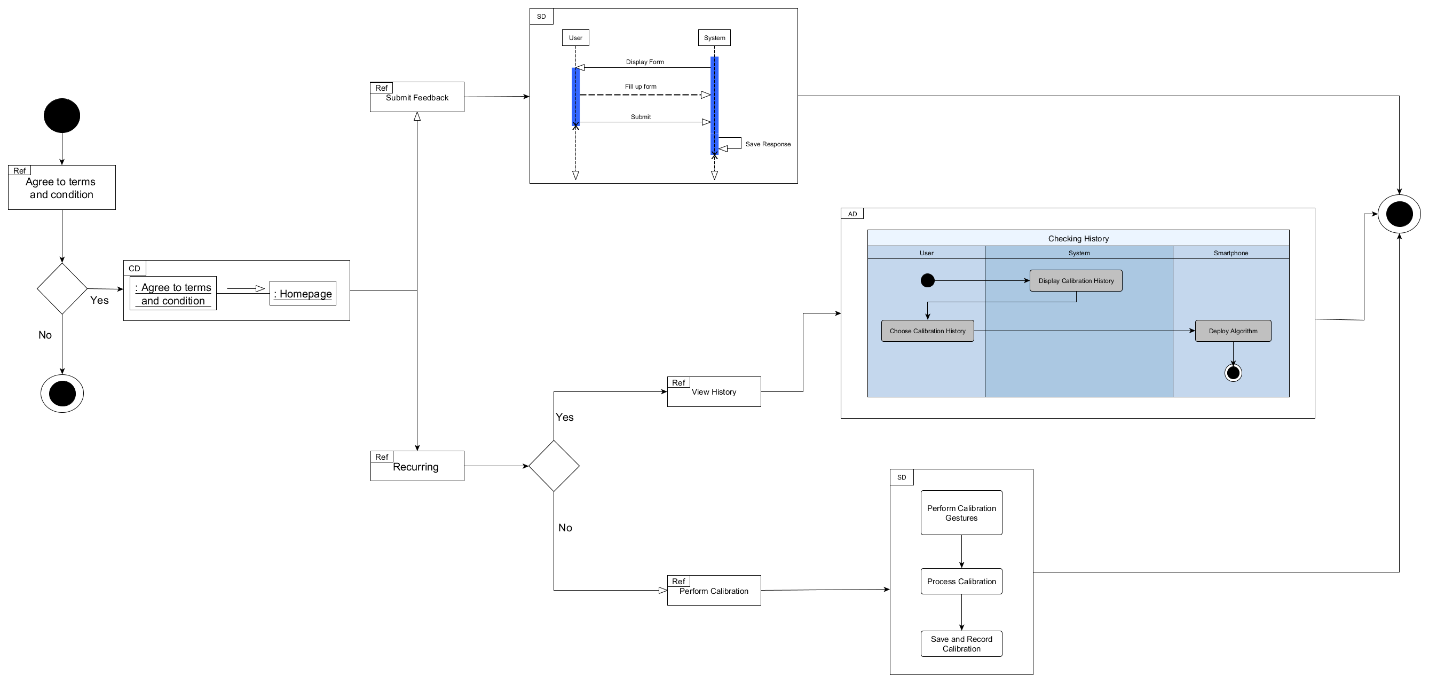
|  |  |  |
| --- | --- | --- |
| **Use Case** | Send Feedback | |
| **Scenario** | User will send feedback or send a report | |
| **Triggering Event** | User want to comment on a feature; User finds an application error | |
| **Brief Description** | User sends feedback or sends a report for comments 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
     2. Send Feedback
  3. Object Diagram
  4. Deployment Diagram
  5. Sequence Diagram
     1. Report
     2. Touch Calibration with History
     3. Touch Calibration without History
  6. Communication Diagram
  7. State Diagram
  8. Timing Diagram
  9. Class Diagram
  10. Component Diagram
  11. Package Diagram
  12. Composite Structure
  13. Interaction Overview

1. Appendices

* 1. Bibliography

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

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

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

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

Ruffer, T. (2013-2016). *Developer Options: Pointer Location*. Retrieved from Taming the Droid: https://tamingthedroid.com/pointer-location

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/

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

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.

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.