IIT INDORE

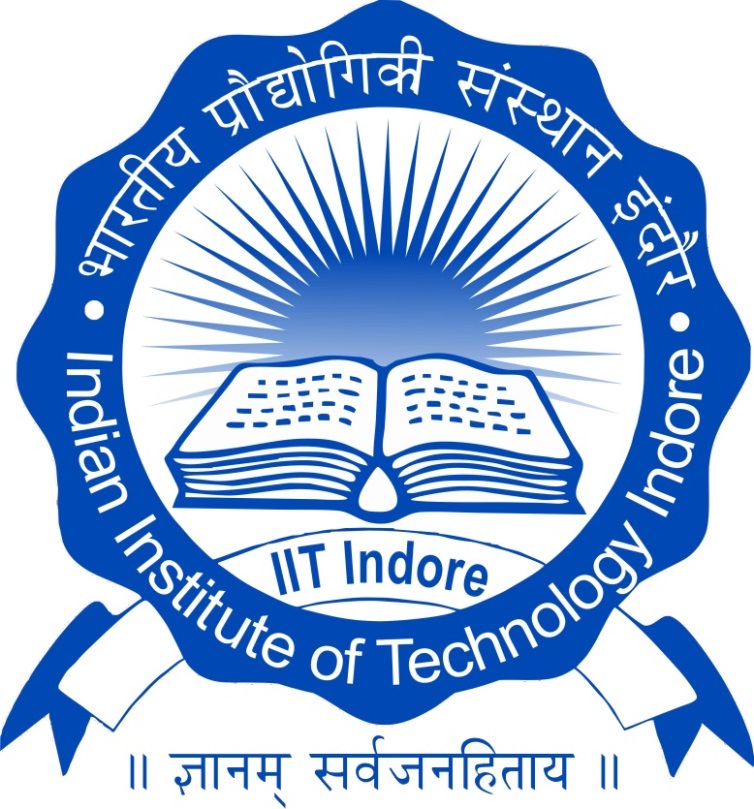
S

R

S

**Department of Computer Science and Engineering**

**CS 258: Software Engineering**



**Project 10**

Mobile Application for User Identification based on Gestures and Pressure Applied on Touch Screen.

Software Requirements Specification

Version 1.0

**January 31, 2014**

**Group 5:**

* Aneela Bellamkonda (1200107)
* Amey Nagapurkar (1200118)
* Ravi Prakash Verma (1200127)
* Vinay Bagade (1200138)

**Prepared For**

IIT INDORE

**Instructor:** Dr. Abhishek Srivastava

Table of Contents

[1. Introduction 3](#_Toc378958118)

[1.1 Purpose 3](#_Toc378958119)

[1.2 Scope 3](#_Toc378958120)

[1.3 Definitions, Acronyms and Abbreviations 3](#_Toc378958121)

[1.4 References 4](#_Toc378958122)

[1.5 Overview 5](#_Toc378958123)

[2. General Description 5](#_Toc378958124)

[2.1 Product Perspective 5](#_Toc378958125)

[2.2 Product Functions 5](#_Toc378958126)

[2.3 User Characteristics 6](#_Toc378958127)

[2.4 General Constraints 6](#_Toc378958128)

[2.5 Assumptions and Dependencies 6](#_Toc378958129)

[3. Specific Requirements 7](#_Toc378958130)

[3.1 External Interface Requirements 7](#_Toc378958131)

[3.1.1 User Interfaces 7](#_Toc378958132)

[3.1.2 Hardware Interface 7](#_Toc378958133)

[3.1.3 Software Interfaces 7](#_Toc378958134)

[3.1.4 Communication Interfaces 7](#_Toc378958135)

[3.2 Functional Requirements 8](#_Toc378958136)

[3.2.1 Collection of User’s Characteristics 8](#_Toc378958137)

[3.2.2 Identification of User 8](#_Toc378958138)

[3.3 Use Cases 9](#_Toc378958139)

[3.3.1 Actors 10](#_Toc378958140)

[3.3.2 Functions 10](#_Toc378958141)

[3.3.3 Use Case Index 11](#_Toc378958142)

[3.4 Non-Functional Requirements 11](#_Toc378958143)

[3.4.1 Performance 11](#_Toc378958144)

[3.4.2 Reliability 11](#_Toc378958145)

[3.4.3 Availability 11](#_Toc378958146)

[3.4.4 Security 11](#_Toc378958147)

[3.4.5 Portability 12](#_Toc378958148)

[3.5 Design Constraints 12](#_Toc378958149)

[3.6 Logical Database Requirements 12](#_Toc378958150)

# 1. Introduction

This document is about the required specifications for an Android™ Application for user identification, which can store analysed data for various gestures and pressure applied on screen for different users, and can identify the user based on stored analysed data.

## 1.1 Purpose

The purpose of this document is to provide a brief functionality and requirements of the application. This document is intended for all users having a basic idea about using a touch device and with no prerequisite knowledge of biometrics.

## 

## 1.2 Scope

1. For touch devices privacy of user is one of most important concern because any one can easily see and access one’s device data. Normal password protection does not help so much. So for its solution verification of user can be done by analysing user’s activities (i.e. Gestures, Pressure applied on screen etc.). When users put their fingers on the touch screens to control the devices, the related touching information from users is collected, analysed and compared with owner’s samples which are stored to verify the identity of users. In this way, touch devices will be able to recognize their owners when their screens are touched by users.
2. The application requires a Smartphone based on Android™ platform, it will not work on any platform other than Android™ (i.e. Windows phone or ios).

## 1.3 Definitions, Acronyms and Abbreviations

* **Definitions**

**What is an Android™?**

Android™ is open source Operating System based on Linux Kernel developed and owned by Google Inc. designed for touchscreen mobile devices like smartphones and tablet PCs. It was firstly used on mobile devices and then expanded to the tablet PCs. The first publicly available smartphone running Android, the HTC Dream, was released on October 22, 2008. As of today Android™ is the most popular mobile operating system with about billions of users.

**Integrated Development Environment (IDE) for the Application**

Eclipse is a popular cross-platform Integrated Development Environment (IDE). This project used Eclipse android SDK to develop android application to collect touch characteristics from individual users. After the programming environment was set up, a virtual mobile phone named Android Virtual Device (AVD) was used for running Android projects.

**Biometrics**

Biometrics (or biometric authentication) refers to the identification of humans by their characteristics or traits. Biometrics is used in computer science as a form of identification and access controls. It is also used to identify individuals in groups that are under surveillance.

**Android™ Emulator**

Android Emulator is a virtual mobile device that runs on your computer and allows you to test developed applications without using a physical device.

**Application Programming Interface (****API)**

An interface used by applications to communicate with each other. API is a library that may include specification for routines, data structures, object classes, and variables.  APIs are particularly useful in extending a platform.

**APK File**

APK file stands for Android application package file. Each Android application is compiled and packaged in a single file that includes all of the application's code resources. To run an application and android device its .apk file need to be installed.

* **Acronyms and Abbreviations**

**ADT:** Android™ Development Tools, the process by which new applications are created for Android™ Operating System.

**SDK:** Software Development Kit, which includes a comprehensive set of development tools like debugger, libraries and emulators etc.

**IDE:** Integrated Development Environment, a platform for developing the Application.

**AVD:** Android™ Virtual Device, a utility provided by Google as part of the Android™ SDK which allows creating Emulated Android Devices.

## 1.4 References

* Android™ Developers: [developer.android.com](http://www.developer.android.com)
* Wikipedia: [www.wikipedia.org](http://www.wikipedia.org)
* Tutorials Points: [www.tutorialspoint.com](http://www.tutorialspoint.com)

## 1.5 Overview

This document is structured as follows

* Section 1 Consists of the preliminary Introduction regarding the Project
* Section 2 contains the general description of Project.
* Section 3 talks about the specific requirements of the Project.

# 2. General Description

This Section contains the basic and optimum description, functionalities, dependencies and perspective regarding the Application.

## 2.1 Product Perspective

* This Application is platform (Operating System) dependent and runs on Android™ Smartphones or Tablet PCs with Android™ version higher than 2.2 (Froyo).
* Most of the currently available user authentication systems and firewalls do not provide any authentication functionalities to identify the users who access the computer system. So for data protection and confidentiality, authentication of a user becomes increasingly important. One efficient way is validating users with legitimate accounts and giving the data access control to the users.
* Since this application requires users’ personal characteristics (touch measures), so validation of user becomes precise. For this application issues like forgetting passwords do not cause any problem.
* From any other user authentication application, in this application issue of identity theft can never arrive.

## 2.2 Product Functions

To verify whether every user has special touching characteristics, personal touch information needs to be collected and analysed. A data collecting and classification method was designed and implemented to collect the touch parameters and properties from individual users. An Android application is developed as the tool to be installed into a touch device and collect touch information such as finger location, finger pressure and finger movements from individuals. Then this data collecting process is tested through a quantitative method. Multiple users perform the same touch testing experiment under the same testing conditions. Observation and analysis from the data of full test can show the different touching characteristics from individual users. Statistical methods are applied to measure and analyse observations in order to improve the accuracy of the results. Based on the data of the experimental testing, the various comparisons of several touch parameters among different users will be discussed and evaluated, with respect to achieve a good accuracy of individual identification.

## 2.3 User Characteristics

The specific requirements of this Application will vary as the characteristics of user will change. This variation can be categorised for 4 types of users.

* The users who have never used any touch devices.
* The users who have not owned any touch devices but used some public machines with touch screens a few times.
* The users who have touch devices but use them with a low frequency such as once a week or several times per month.
* Most important, the users who have at least one touch device and use it every day.

## 2.4 General Constraints

* Every time when the users touched the screen, only one finger could be used, 2 or more finger touching points makes subsequent analysis and classification more difficult.
* Sufficient Memory should be available to store the collected data, as for analysing a user more and data is required.

## 2.5 Assumptions and Dependencies

* As long as phone supports the required Android™ constraints, this application so designed will work fine without any problems.
* The working and performance of the application will be good with sufficient availability of RAM, storage capacity and good processor.

# 3. Specific Requirements

## 3.1 External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

### 3.1.1 User Interfaces

The application starts with a splash screen, followed by home page which contains options to add a new user for data analysis and to identify a user. Adding a new user requires a username by which one will be identified. In both cases application redirects user to next page on which all data inputs will be taken for logging and analysing the user. In case of logging all data will be stored in a log file, while in case of analysing the user identification of matched user will be shown by the related username.

### 3.1.2 Hardware Interface

As being an Android™ phone application, it is designed to interface with the hardware present on the Android™ phone. Since at time of designing applications can be tested on emulator, but in this application as we require touch interface, so for testing use of physical Android™ device is mandatory. Some minimum requirements are

* Multi-Touch Screen
* 256 MB RAM
* 2 GB External Memory
* 512 MHz Processor

### 3.1.3 Software Interfaces

For software interface, it is essential that the platform (Operating System) of Android™ device should be higher than Android™ 2.2 (Froyo).

### 3.1.4 Communication Interfaces

The communication between the different parts of the system is not a concern for user, as it is managed by Android™ system itself. Apart from that no specific communication interface with external environment (i.e. internet connection, GPS, Wi-Fi etc.) is required for working of this application.

## 3.2 Functional Requirements

This section talks about the specific functionalities and their working that will be done by this Application.

### 3.2.1 Collection of User’s Characteristics

**3.2.1.1 Introduction**

This application does the data collection of users’ characteristics. And based on stored logs does the analysis on every user.

**3.2.1.2 Inputs**

Each user need to create a separate username in application to allow it to collect the data (in form of touch responses and gestures) and process it to perform the required the functionality.

**3.2.1.3 Processing**

In processing application does the statistical calculation to create different databases for every user.

**3.2.1.4 Outputs**

No specific output is obtained in collection of user characteristics, as all collected data will be managed by application itself internally.

**3.2.1.5 Error Handling**

Error is a part of the Statistical Calculations. So the mean values obtained from the applications from a user may match the values of some another user, thus there is always a chance a crept in error.

### 3.2.2 Identification of User

**3.2.2.1 Introduction**

This application can also identify the user based on previously collected data stored in log files.

**3.2.2.2 Inputs**

As inputs user needs to allow application to collect similar data again (in form of touch responses and gestures) for verification.

**3.2.2.3 Processing**

In processing application will cross match and verify the analysed input with previously stored analysed data.

**3.2.2.4 Outputs**

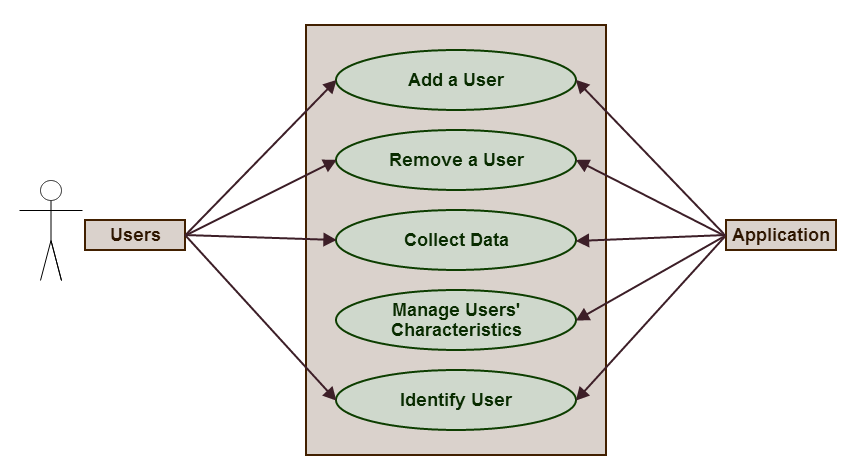
Bases on processing done by application it will identify the user and will display the corresponding username. Thus it will serve the purpose of User Identification.

**3.2.2.5 Error Handling**

In user identification there is a possibility that 2 user can have similarities in their statistical data, which may lead to cross matching of two identities. In such cases user should try again and behavioural user input should be provided and user self-consciousness should be avoided.

## 3.3 Use Cases

This section describes different Use Cases and their connectivity to each other.



**Use Case Diagram**

### 3.3.1 Actors

* **Users**

Users for this application will act as an operator for its functioning.

* **Application**

Application will also act as an operator for itself, as all functioning running in background will be controlled by application.

### 3.3.2 Functions

* **Add a User**

Addition of new users in application will be done by user itself by just logging in with a unique username, while as a new user makes an account application will create a different log file for storing user’s characteristics

* **Remove a User**

Removing of existing user will be done with user’s choice, and while removing a user its existing data files and logs will be removed by application.

* **Collect Data**

In data collection both user and application have the same importance, as for data input user need to provide touch responses and to store all data application will create different log files and store it in external memory.

* **Manage Users’ Characteristics**

In data management application does all the statistical calculations to make an average database for every user as log files and while identification of user retrieve data from these files.

* **Identify User**

In user identification users to provide input in form of touch responses. And application will verify these inputs with existing databases, and identify the user.

### 3.3.3 Use Case Index

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID | **Use Case Name** | **Primary Actor** | **Complexity** | **Priority** |
| 1. | Add a User | User | Low | 1 |
| 2. | Remove a User | User | Low | 2 |
| 3. | Collect Data | User and Application | Medium | 1 |
| 4. | Manage Users’ Characteristics | Application | Medium | 1 |
| 5. | Identify User | User and Application | High | 2 |

## 3.4 Non-Functional Requirements

### 3.4.1 Performance

As seeing the current configuration and functionalities of Android™ devices, performance of application will not be a point of issue. However, some old devices with weaker hardware may incur some problems and relatively run slower.

### 3.4.2 Reliability

This Application is sufficiently reliable, although due to involvement of the Statistical calculations in processing the data, there is a chance of data collision and cross matching of two users. However the possibility is quite rare.

### 3.4.3 Availability

This Application will be available on Google Play Store for every Android™ device which fulfils the minimum requirements of it.

### 3.4.4 Security

The application will not ask for any personal information from the user. Users just require installing this application on Android™ device and after starting it have to create a username to start using it. Every user need to create a different username, so that application can create different logs for different users.

### 3.4.5 Portability

This application is easily portable as it can be shared from one Android™ device to another Android™ device via Bluetooth or through any other communicative resources. However the sharing of analysed log files (which store data) is not suggested because of privacy measures.

## 3.5 Design Constraints

* Slow processor speed can affect how users perceive the responsiveness of the application.
* Since a shorter battery life can impact the user experience, be aware that the more efficiently that a device handles data, the less often it needs to turn on the radio and the longer the battery life.
* A variety of different screen sizes, resolutions and orientations (i.e. portrait, landscape, switchable) increase complexity for developer.
* Only the methods in Java™ API and Android™ ADT can be used, so the look and feel of the application is limited to Java™ frameworks only.

## 3.6 Logical Database Requirements

Logical database is required for this application. All the databases of user’s characteristics will be stored in log files in XML format. And thus sufficient external memory is required for storage of logs for different users.