

Project Name - Smart warning System

CS223

Team No. - 18

Project No. - 9

Software Requirements Specification Document

Table of Contents

1. Introduction

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms, and Abbreviations

1.4 References

1.5 Overview

2. The Overall Description

2.1 Product Perspective

2.1.1 System Interfaces

2.1.2 Interfaces

2.1.3 Hardware Interfaces

2.1.4 Communications Interfaces

2.1.5 Memory Constraints

2.2 Product Functions

2.3 User Characteristics

2.4 Assumptions and Dependencies

2.5 Apportioning of Requirements

3. Specific Requirements

3.1 External interfaces

3.2 Functions

3.3 Software System Attributes

3.3.1 Reliability

3.3.2 Availability

3.3.3 Security

3.3.4 Maintainability

3.3.5 Portability

1. Introduction

This SRS aims at developing an Android App for the general use of students and teachers. This app alerts students and teachers during a lecture in a classroom. It generates alerts in form of warning messages.

1.1 Purpose

This app is being developed as a software project for software engineering course of IIT Guwahati. The intended audience would be students and teachers.

1.2 Scope

Project: Smart Warning System

Features:

1. Number of the students studying or not studying.
2. State change alert(student is engaged with the lecture or doing something else).
3. Ensure that alert is received and acted upon.

This Software when completed will be usable by all users who can read and understand English language and can operate an Android Application efficiently. Though the intended audience is supposed to be the students and teachers.

1.3 Definitions, Acronyms, and Abbreviations.

SDK : Software Development Kit

NDK : Native Development Kit

API : Application Programming Interface

API level : A measure of the version of Android device being used.

API levels 8 ~ Android 2.2 (Ice Cream Sandwich)

23 ~ Android 6.0 (Marshmallow)

1.4 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

<https://developer.android.com/training/index.html>

<http://www.androidauthority.com/android-studio-tutorial-beginners-637572/>

https://www.tutorialspoint.com/android/android_studio.htm

<https://unity3d.com/learn>

1.5 Overview

Other parts of the SRS :-

1. Functional Analysis : contains the modular structure of the whole software
2. Softwares/Resources Requirements: Softwares required to construct the product
3. Interface Description

2. The Overall Description

This section provides a background for the requirements, which are defined in section three and thus presents an overall view of the specific requirements.

2.1 Product Perspective

This product is a part of the teaching app which uses Android Application to let the teachers know the current engagement of the students whether they are studying or engaged somewhere else. This feature will be activated only when chosen. This product will act in collaboration with the parent system.

2.1.1 System Interfaces

The parent system also developed for Android Platform will be directly Interfaced with this product and thus won't need any external added API for communication.

2.1.2 User Interfaces

This Product uses the following software - user interfaces --

1. The application will be used with help of any Android device supported by the final product.
2. The software will be GUI accessible and will use the touch screen interface of the android device to communicate with the users.
3. The choice of feature being used will use Menu-Based interface for letting the users select the desired features(such as get a text alert or a pop up window or an alarm when the state of a student changes).
4. The User Registration page will use form-based interface for letting the user enter the required information. This will be shown when the user

enters the application for the very first time and in subsequent times it will ask to login if the user is logged out else it will run directly.

2.1.3 Hardware Interfaces

The final developed product will be usable by any Android with :-

1. API level 8 or Android 2.2 (Ice Cream Sandwich).
2. Proximity sensor.
3. Gyroscope sensor
4. Accelerometer sensor
5. Beacon
6. Wifi allowance.

2.1.4 Communications Interfaces

Local server access to be able to connect to the network of devices connected while the lecture is progressing. All students' Android in turn will be connected to a main interface on the professor's Android.

2.1.5 Memory Constraints

Minimum memory storage space of 100Mb.

2.2 Product Functions

Functions included in the final product will be as follows :-

1. State change alert.
2. No. of students studying and not studying.
3. Unignorable alert.

2.3 User Characteristics

The intended users for the product will have the following characteristics.

1. Able to afford and use an Android device above the specified API level.
2. Able to understand the functioning and operation of the software on a basic level.
3. Able to understand the English Language to operate the application.
4. Belonging to the age group of 20 - 40 years (preferably students).

2.4 Assumptions and Dependencies

The underlying assumption while creating this application are as follows-

1. OS used on the device :- Android 2.2 or higher / API level 8 or higher.
2. Device must have a minimum of 150 Mb storage space for storing the application and also some minimal data generated whilst operation.
3. The device must have a working wifi connection.
4. The sensors work in same fashion in all devices.

The dependencies are follows-

1. Must have ARM architecture based hardware design.

2.5 Apportioning of Requirements.

In the case that the project is delayed, there are some requirements that could be transferred to the next version of the application. The subtle details may be omitted such as small changes in orientation or position. Since these requirements are secondary they can be avoided without affecting overall functioning of the application.

3. Specific Requirements

The specific requirements for the software to work in the desired manner and provide all required features are described in the following section.

3.1 External Interfaces

The inputs and outputs from the application related to the external environment --

1. Inputs-

- (a) Orientation and current state and position of the device.
- (b) Access to local server.

2. Outputs-

Alphanumeric data corresponding to engagements of students.

- A). Student inside the class and studying attentively.
- B). Student inside the class but not attentive.
- C). Student outside the class.
- D). Alarm - on/off.

3.2 Functions

The description of the functions used in the application is given below.

3.2.1 State Change Alert.

(1) Inputs-

Current state and previous state of user.

(2) Output-

Alarm - on/off.

Description - If previous state change and current state are different then alarm beeps otherwise it doesn't.

3.2.1.1 State Change Rate

(1) Inputs-

Current state and previous state of user.

(2) Output-

No. of times state changes.

Description- Feature to detect the rate at which the state of the user changes.

3.2.2 No. of students studying

(1) Inputs-

Current state of all the users.

(2) Outputs-

No. of students studying in class.

Description-

If the current state of a student is in the range 1 to 4 then the student is studying attentively.

3.2.3 Unignorable Alert

(1) Inputs-

Previous and current state of user.

(2) Output-

Continuous alert in student's and prof's device.

Description-

Alarm beeps until the prof. Manually dismisses it so that we confirm that the delivered report is acted upon.

3.3 Software System Attributes

This section includes all the non functional requirements for the software --

3.3.1 Reliability

MTBF (Mean Time Between Failures) : Minimum value - 30
Seconds

The software is supposed to work properly while running any applications while providing user with probable suggestions to change their state.

The software might fail when user tries to run multiple features in succession as the API takes a lot of processor memory.

3.3.2 Availability

The system will be available for use whenever the user deems necessary 24/7.

The system shall allow users to restart the application after failure with the loss of at most the last image matrix captured while operation.

3.3.3 Security

The system will use a login system for authentication and thus will be highly secure and will prevent any type of unauthorized access to private content.

3.3.4 Maintainability

The system will be updatable from software patches available through the Google Play Store. Any discrepancies will be addressable by any developer as the coding will be done according to the coding standards of IEEE.

3.3.5 Portability

The software will be easily transferrable to any Android device satisfying the minimum software dependency requirements as specified in this SRS Document. The software can be installed on an Android using the same method as any other Android App via the Android App Manager.