Software Requirements Specification

Group 6

Project 10

Archit Jugran , 160101087
Shubham Goel, 160101083
Yagyansh Bhatia, 160101079

INDEX

1. Introduction

1.1 Purpose

The purpose of the project is to allow the teacher to know the names of students who may not be attentive in class, for a better class environment and for the benefit of the students themselves. In this document, we provide a detailed description of the requirements for our yet unnamed application. It explains user and hardware interfaces ,system requirements, specifications and other constraints as well as the functional and nonfunctional requirements.

1.2 Document Conventions

API : Application Programming Interface
API Level : An indication of the diversity of the features of an Android version.
As Android version increases, API level increases and functionality increases but support to devices decreases.
User: Someone who interacts with the application.
NFC: Near Field Communication.
Wi-Fi : Wireless Fidelity
GHz : Gigahertz

1.3 Intended Audience

The intended audience of this SRS document is our overseeing project professor, Dr. Samit Bhattacharya and our teaching assistants , Sir Ujjwal Biswas, Sir Md Shakeel Iqbal Saikia and Sir Subrata Tikadar.

1.4 Product Scope

The product aims at making the classroom environment better by sending warnings in the form of notifications to un-attentive students as well as sending notifications to the teacher about the students, hence motivating the students to stay attentive in class for their own benefit. The system is based on a database on a server which accepts, processes and pushes data from/to Android devices. A comfortable user-experience will also be built.

1.5 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer Society, 1998.
Android Studio: https://developer.android.com/training/index.html

1.6 Overview

This SRS also includes -

a. Functional Analysis : Explains the modular structure of the software

b. Softwares/Resources : Softwares and other things that constitute the final product.

c. Interface Description : The user interface as seen by end user.

2. Overall Description

2.1 Product Perspective

The product is meant to serve as an Android Application which lets the teacher know through alerts or notifications the change in behavior or status of students during the class. Alerts are sent to multiple devices in which the same user, whose state interpretation changes, is logged in. This product will act in collaboration in the parent system which generates the state of a student, but for now we are generating random states.

2.2 Product Functions

The functions included in the final product are as follows:

- ➔ User authentication
- ➔ WLAN setup
- ➔ State Table generation
- ➔ Message generation
- ➔ Notify user

2.3 User Classes and Characteristics

There are two types of users and the interface is slightly different for both classes:-
1.Teachers
2.Students studying in a school or university .

The teacher will receive notifications about all students whose state interpretation changes either from low to high or high to low on all devices that he is registered on.
Similarly, the students will receive notifications about their own change in state interpretation on the multiple devices on which they are registered on. The user interface is explained in more detail in Section 3 of this document.

The users of this product must also have the following characteristics :

1. Have an Android device with Android version 4.0 or above (API Level 15 or above.)
2. Have familiarity with the functioning and use of Android operating system on a basic level.
3. Can understand simple English to use this product.
4. The user should have multiple Android devices if he wants to register himself on more than one device. For example, a smartphone and a smart watch.

2.4 Assumptions and Dependencies

We are assuming that the application is being used on an Android device meeting the requirements

1. Operating system used : Android v4.0 / API Level 15 or higher
2. Either Wi-Fi with internet connection OR Wi-Fi direct working properly.
3. Google Firebase server properly set up.

If any of the requirements or hardware resources are not met, then the application may not work properly as intended or even may not work at all. Hence the mobile phones must have sufficient resources not occupied i.e. not allocated to other processes.

2.5 Apportioning Requirements

Constructing a function or module for implementation of support of non Android devices may not be included in the first version of the design.

3. External Interface Requirements

3.1 User Interfaces

The application will have a user friendly touch-interface (GUI - Graphical User Interface) , for example, a “Help” section which appears on every screen and gives information about the application and instructions on how to use it.

a) User interface for student: For user registration, a simple form-based interface is used for allowing the user to enter their information and in case of invalidation of details, a red colored error message will be displayed. The next screen contains a button, which on clicking allows the student to connect to the network.

b) User interface for teacher: The same button, when clicked by the teacher enables the functionality of the application, i.e. now the students can connect to the server .

After connecting , both classes of users are presented with a screen to view all the notifications sent to their device by the server : the student receives notifications pertaining to him whereas the teacher receives notifications pertaining to change of state of all students.

3.2 Hardware Interfaces

The application can be used for any Android device(smartphone/ smartwatch) having the following features :

1. Wi-Fi 802.11ac or 802.11n.
2. Wi-Fi direct feature or NFC (Near Field Communication).
3. Aspect ratio should be between 1.333 (4:3) and 1.779 (16:9)
4. At least 1.0 GHz Krait Processor

3.3. Software Interfaces

1. Android version 4.0 (Ice Cream Sandwich) ,i.e. API Level 15 or above
2. Internet availability in the local area network.

3.4 Communication interfaces

There must be a local area connection / server to which all the Android devices are connected while the lecture is being delivered by the teacher. The state data will be transmitted from the student’s device to the server and from the server to the teacher’s and student’s Android device.

4 Functional Requirements

4.1 User authentication

Input: User credentials (username, password) to login or register

Output: User logged in/not logged in.

Description: User credentials are verified/registered and he is logged in.

4.1.1 Login

Input: User credentials to login (username, password)

Output: User is logged in or error displayed.

Description: If user credentials are authenticated, then log in the user otherwise display an error, asking the user to try again.

4.1.1.1 Validate login details

Input : User’s login details - Username, Password

Output: Result of validation – valid/invalid

Description : Validate the login details and return true if the login credential are correct otherwise false.

4.1.1.2 Login the user

Input : Result of validation - valid / invalid

Output : Welcome user or error message displayed.

Description : If input is true, then login the user otherwise display a message asking the user to re-enter the details.

4.1.2. Register

Input: User details

Output : User logged in.

Description : User details are stored and User is logged in.

4.2 WLAN setup

Input: LAN turned on and User logged in - True
Output:List of Users Connected.
Description: A local network is created and Users connect to it.

4.2.1 Create Server

Input: Local Area Connection turned on
Output: Server Created
Description: Server is created to which teacher and students will connect.

4.2.2 Join Server

Input:User logged in as Student
Output: User connected - Boolean
Description: Return True if user is connected to server,else false.

4.2.3 List of Students Generator

Input: Users connected – Boolean
Output: List of Users connected
Description: The boolean values for each student is checked and list of the connected users is generated.

4.3 State table generation

Input: List of Users Connected
Output: State table of Users
Description: Random States of Users are generated and returned as a table.

Eg.

S. No.	Name	State
1	Archit Chaddha	9
2	Shubham Bhatia	4
3	Yagyansh Goel	6

4.4 Message Generation

Input: State table of the students.

Output: Message generated on the server

Description: If interpretation of state changes , generate a message on the server which needs to be later sent to the device of the student and the teacher.

4.4.1 State Change Detector

Input: State table of the students.

Output: Result of change of interpretation of state change.

Description: Whenever there is a change in the interpretation of the state of a student, this function will return 0 or 1 corresponding to low to high and high to low respectively, otherwise 2.

Eg.

Interpretation	State
Low	1 to 4
Medium	5 to 7
High	8 to 10

4. 4.2 Message generator

Input: Result of change of interpretation of state.

Output: Message to be sent or not to be sent.

Description : Depending upon the result of the previous function, a message will be generated accordingly.

4.4.2.1 High-to-low message generator

Input: Result of change of interpretation of state change

Output: Message for the user

Description: If the interpretation of the state of the user is changed from high to low, then the message generated is ”Please pay more attention”

4.4.2.2 Low-to-high message generator

Input: Result of change of interpretation of state change

Output: Message for the user

Description: If the interpretation of the state of the user is changed from low to high, then the message generated is “Keep up the good work!”

4.4.2.3 No State change

Input: Result of interpretation of state change

Output: No message to be sent

Description: If interpretation of states of every user is same as before, then no message is to be generated.

4.5. Notify user

Input: Message generated on server

Output: Notification on the user's device

Description: Depending on the input, a notification is generated if needed and the message is displayed.

4.5.1 Teacher Notifier

Input : Message and its Corresponding User
Output: A log of messages with their Username
Description: Whenever interpretation of state of user changes ,the corresponding message along with their username will be displayed in the log of the application in the teacher's device(s).

4.5.1.1 Alert Decider

Input: Number of students paying attention from state table.

Output: More than 50% of the class status goes down- True or False

Description: Generates the percentage of students whose attention is going down.

4.5.1.2 Vibrate Notifier

Input: More than 50% of the class status goes down - True

Output: Notification through vibration

Description: Notification will be sent to teacher’s device with vibration, as it is an important notification and should be notified to teacher immediately.

4.5.1.3 Silent Notification

Input: More than 50% of the class status goes down - False

Output: Silent Notification

Description: Notification will be generated but it will get stored in teacher’s log and he will not get disturbed . He can access the log whenever he want.

4.5.2 Student Notifier

Input: Message
Output: Notification and confirmation
Description: The corresponding message will be sent to user device(s) as a notification which waits for the user to confirm and sent the confirmation.

4.5.2.1 Notification Sender

Input:Message to be sent.

Output: Notification Sent

Description: Whenever interpretation of state of user changes ,the corresponding message will be sent to user device(s) which waits for the user to confirm and send the confirmation. All the devices on which user is logged in will show the notification, be it a smartphone/

smartwatch or any other smart device.

4.5.2.2 Confirmation

Input: Touch
Output:Confirmation-Boolean

Description: User confirms that he has received the notification and the confirmation is sent back.

5 Other Nonfunctional Requirements

5.1 Reliability and Performance Requirements

1. Performance should not be an issue since all of our server queries involve small pieces of data.
2. Server updates will take only a few seconds of time as long as the phone maintains a steady signal.
3. The performance of the product may depend to a small extent on the hardware components of User's devices.
4. An error may occur if while downloading information from the server, the data on the server changes which would lead to inconsistency.

5.2 Usability

The app should not be closed i.e. the user can switch to some other process in the smartphone but the application should still be present in the list of background processes. It must be ensured that the server (local area network) does not shut down at any moment of time during the lecture.

5.2.1 Contextual Inquiry

Contextual Inquiry is a technique of studying users in their own natural environment, to get under their skin, be it in their place of work or home life, understanding what makes them tick.In our development project we are following the passive type of contextual Inquiry.

1 Interview

We provided Users the general idea of the application and then asked the same set of question to all of them.

1.1 **Question-** How would you like to receive the notification(to teachers) Audio/vibration/anything else?

Teacher1 :- I don’t want to get distracted in my class. So I would like it to be absolutely silent.

Teacher2 :- I think vibration will do just fine.

Teacher3:- I think it will disturb me if it buzzes too much, but i need to know if majority of class isn’t paying attention.

1.2 **Question:-** How would you like to receive and confirm notifications(to students)?

Student1:- I would like that my device should vibrate so that only I will know when the notifications comes and my peers won’t get disturbed .It should get confirmed when I click on it.

Student2:- Device should vibrate when it receives a notification and confirm it when I see it.

Student3:- Clicking on it will be convenient.

1.3 **Question:-** Do you want to get a notification everytime your attention level changes or only when it falls ?

Student1:- No, I want to get a notification only when my attention level falls , as It will interrupt me when I am not paying attention, and won’t do anything if I am studying.

Student2:- Yes, It should notify me every time as when my attention level increases it will serve as a motivation and when it falls It will remind me to pay more attention.

Student3:- only when it falls, then notify me. Otherwise i don’t want to get disturbed

1.4 **Question** : What are your preferred devices which you would like to use with this application?

Student1:- I would like to use both my Android smartphone and my smart watch .

Student2:- I only have one Android smartphone and would like to use that only.

Student3:- I think phone will be better

Teacher1:- I usually carry both my smartphone and my laptop to the class so any of them will work, but i don’t want the application to notify me on both.

Teacher2:- I don’t want the application to be on my laptop as I might be using slides on it, and displaying it to the class.

Teacher3:- I usually carry phone in my pocket, and when it will vibrate, it will notify me. That would be very convenient.

1.5 **Question:-** Would you like this application to take attendance of the class as well?

Teacher1: Yes, it will save me a lot of time which i can use to teach more.

Teacher2: No, students may give fake attendance as they they might bring their friend’s phone and then his/her attendance will be counted.

Teacher3:- Nah, if a student runs out of battery, his/her attendance won’t be counted

Student1 : Yes, i always have problems with the biometric as it never accepts my fingerprint easily.

Student2 : anything would be fine for me.

Student3:- No, fake attendance will take place then.

Observation

Question 1:- Teachers will be notified silently unless there is downfall in the state of more than 50% of the class

Question 2:- Device will vibrate (when state falls) on receiving notification and will send the confirmation once clicked.

Question 3:- Students will be notified only when their interpretation of state falls, otherwise notifications will be displayed silently

Question 4:- Most of the students primarily use android smartphones and other android based products only, hence we will make an android application

Question 5:- No, attendance will not be taken be application , because of its inability to check for fake attendance.

5.3 Security Requirements

The system will allow students connected to the local area network the ability to register if they are using the application for the first time and a login feature for further use. Hence, security is provided from unwanted use.

5.4 Availability

The application will be available for download from the Google Play Store to students and all notifications will be viewable at all times. However, the application's interaction with the server will function only during lecture hours.

5.5 Portability

Since the application will be made on Android Studio , it will function properly in any Android device(smartphone/ smartwatch) satisfying the minimum requirements mentioned earlier in this document.

5.6 Safety

1. Information must be securely transmitted to the server without any changes in information.
2. The application will not affect data stored outside of the server or data related to other applications.

5.7 Software Quality attributes

1. The application will take into account situations in which a user loses connection to the server. Any changes in state detected or notifications to be sent to that device will be cached until the connection is restored.
2. The app will be made user friendly (easy to navigate) and visually appealing with a good graphical user interface . For example, The "Help " section will tell the user how to use the application properly.