**Functional Specification Contents**

## **0. Table of contents**

A table of contents with pages numbers indicated for all sections / headings should be included.

## **1. Introduction**

**1.1** **Overview**

Keeping track of attendance can be very useful in a number of scenarios. Some college lecturers like to award students who have high attendance records. Secondary schools have mandatory attendance and need to be aware of truancy. Keeping record of meetings and the numbers of attendees is important in any educational or work environment.

The system would be used to keep attendance at particular events, such as lectures or project supervisor meetings, with zero configuration other than the user’s details. It is made up an NFC chip, an android mobile application and a web interface. For example, a lecturer would own a personal NFC tag, be that a keyring or card, and students would use the mobile app to scan the tag and record their attendance with that lecturer at that particular time slot.

It can be assumed that if 10 people scan the tag at the same time period, they were all at the same event. This info would be recorded and stored in a database. This information can then be viewed by the lecturer on a timetable web interface. Time slots where the tag had been scanned will be highlighted on the lecturer own timetable, showing how many students were in attendance at that event. Clicking into the event will show more detail such as individual student names.

This application stores information about the attendance at an event and removes the need of individual times, timetables, locations, lists of names and numbers, and any other configuration that may had needed to be set up or known previously .

**1.2** **Glossary**

Define and technical terms used in this document. *Only include those with which the reader may not be familiar.*

* NFC (Near Field Communication) is a set of communication protocols that enable two electronic devices, one of which is usually a portable device such as a mobile phone, to pass data to one another by touching them or putting them very close together.

## 

## **2. General Description**

**2.1 Product / System Functions**

Describes the general functionality of the system / product.

The product is aimed at those who want an improved, simpler way of taking attendance at an event.

**2.2 User Characteristics and Objectives**

Describes the features of the user community, including their expected expertise with software systems and the application domain. Explain the objectives and requirements for the system from the user's perspective. It may include a "wish list" of desirable characteristics, along with more feasible solutions that are in line with the business objectives.

**Target demographic expectations:** Basic IT knowledge for initial setup of mobile application and use of mobile NFC technology. Ability to install mobile application and set up user details such as student name and student number.

**2.3 Operational Scenarios (use cases)**

This section should describe a set of scenarios that illustrate, from the user's perspective, what will be experienced when utilizing the system under various situations.

In the article Inquiry-Based Requirements Analysis (IEEE Software, March 1994), scenarios are defined as follows:

In the broad sense, a scenario is simply a proposed specific use of the system. More specifically, a scenario is a description of one or more end-to-end transactions involving the required system and its environment. Scenarios can be documented in different ways, depending up on the level of detail needed. The simplest form is a use case, which consists merely of a short description with a number attached. More detailed forms are called scripts.

1. **Scan NFC tag**

**Action:** Student opens mobile app and scans lecturers nfc tag by holding tag close or against phone’s nfc antenna.

**Success:** Students details (name, student num, phone number/IMEI num) are saved to the database and the time and date is recorded. Database is checked that the student name/IMEI num hasn’t been scanned already.App indicates the nfc tag was read.

**Fail:** Details are already in database. The student has already signed in, or this phone has already been used to sign in another student, App indicates that scan had failed.

**2. Sign up (**Name, email, student ID, imei, phone no.**)**

**Action:** Sign up to application for first time. **(Database with all registered users is checked?)**

**3. Lecturer accesses web interface**

**Action:** Lecturer uses details to sign into web interface

**Success:** A timetable is displayed, slots where the tag had been scanned are highlighted on the timetable. **(Will timetable be personalised/edit-able?)**

**Fail:** Interface alerts user sign in was unsuccessful

**Check imei isnt being used more than once**

**Check student isn’t registered twice**

**How does the lecturers interface work?**

**2.4 Constraints**

Lists general constraints placed upon the design team, including speed requirements, industry protocols, hardware platforms, and so forth.

1. Mobile device must have NFC technology. **(secondary version with QR code?)**
2. Application needs internet access

## **3. Functional Requirements**

This section lists the functional requirements in ranked order. Functional requirements describes the possible effects of a software system, in other words, *what* the system must accomplish. Other kinds of requirements (such as interface requirements, performance requirements, or reliability requirements) describe *how* the system accomplishes its functional requirements.

As an example, each functional requirement could be specified in a format similar to the following:

* **Description -** A full description of the requirement.
* **Criticality -** Describes how essential this requirement is to the overall system.
* **Technical issues -** Describes any design or implementation issues involved in satisfying this requirement.
* **Dependencies with other requirements -** Describes interactions with other requirements.
* **Others as appropriate**

## **4. System Architecture**

This section describes a high-level overview of the anticipated system architecture showing the distribution functions across (potential) system modules. Architectural components that are reused or 3rd party should be highlighted.

## **5. High-Level Design**

This section should set out the high-level design of the system. It should include one or more system models showing the relationship between system components and the systems and its environment. These might be object-models, DFD, etc.

## **6. Preliminary Schedule**

This section provides an initial version of the project plan, including the major tasks to be accomplished, their interdependencies, and their tentative start/stop dates. The plan also includes information on hardware, software, and wetware resource requirements. The project plan should be accompanied by one or more PERT or GANTT charts.

## **7. Appendices**

Specifies other useful information for understanding the requirements.