**Functional Spec**

## **0. Table of contents**

* [1. Introduction](#_giidg5q7zuy1)
  + [1.1 Overview](#_3063bgbpb6ix)
  + [1.2 Business Context](#_ks4prz79p34a)
  + [1.3 Glossary](#_s8p560h8aeq8)
* [2. General Description](#_o9t2xwxx5ljt)
  + [2.1 Product / System Functions](#_rxtsnn9h66u1)
  + [2.2 User Characteristics and Objectives](#_3694w0epak8m)
  + [2.3 Operational Scenarios](#_199dkdtrzn89)
  + [2.4 Constraints](#_65rbl4x2q8l7)
* [3. Functional Requirements](#_ws8tgt65kh5x)
* [4. System Architecture](#_ighw4ak7etiv)
* [5. High-Level Design](#_l974eufzdr5j)
* [6. Preliminary Schedule](#_62v80q4c9wuj)
* [7. Appendices](#_typzlkcdesm6)

## **1. Introduction**

### 1.1 Overview

Our product is a chord learning android app to help users to learn and practise chords without a guitar. This is especially useful for those who travel a lot, use public transport or for beginners who can’t practice on a real guitar for long periods of time without their fingers hurting.

The idea is that the phone’s screen would be transformed to look like a guitar’s fretboard. When the user makes a strumming motion near the base of the phone’s screen, notes will be played that will correspond to the where the user’s fingers are placed . If a user presses down more than one note the app will combine them to play a chord. If your fingers are all in the right spots for the chord being practiced, the app will give you feedback by making the phone vibrate.

Since the app is aimed at beginners learning the guitar, we will make it focused on learning basic chords. Our main focus is for the app to to allow the user to learn chords by practicing in a freestyle mode and we hope to implement a feature to teach all the major and minor open chords in a series of lessons.

### 1.2 Business Context

Not applicable to our project since we aren’t working with a business and we don’t intend to put ads on or charge for our app.

### 1.3 Glossary

Fretboard - Also called the neck or fingerboard of a guitar. A long thin wooden part of a guitar divided up by frets where the player pushes the strings to make notes.

Midi files - Musical Instrument digital interface – A language that represents sound files designed to standardise communication between computers and electronic musical instruments.

Multi-touch - A technology allowing a touchscreen to detect and track 2 or more points of contact (usually the user’s fingers) at once.

Firebase - A cloud development and hosting platform for android apps

## **2. General Description**

### 2.1 Product / System Functions

The user will download our app from the Android Play Store onto their phone. The app will essentially turn the phone’s screen into a fretboard, divided into 4 frets with the 6 strings displayed above. The user will hold the phone like the neck of a guitar and as they push down on the virtual strings the note associated with the string in that position will be played.

The main feature for our app is the freestyle mode. The user simply opens the app and selects this mode. Here, the user can place their fingers anywhere on the strings and combine whichever notes they like in order to build up their own chords. The user then strums the base of the phone screen representing the strum of a guitar. The user receives feedback based on the sound of the chord being played: a correct sounding chord, or and error sounding vibration.

The secondary feature will be the chord learning mode. This mode will show the user the correct positions to place their fingers for the selected chord. When all the fingers are in the correct position, the user strums the phone similarly to the main feature. This feature is especially important for beginners as they can use this feature to learn chords to use in the freestyle mode or better still, to learn songs on their real guitar.

Using our app, the user is able to practise the muscle memory for chords but also be able to hear the correct sound of the chord further increasing their learning experience.

### 2.2 User Characteristics and Objectives

We are creating this app for guitarists. It will be important to make sure that using the app replicates the experience of playing the guitar as closely as possible so practicing with String Theory translates to improved guitar skills. With this in mind ensuring that the fretboard appears on screen as roughly the correct width is important. Making sure that the app produces the correct sounds in response to where your fingers are placed is also a requirement, because developing a connection between sound and where your fingers are is fundamental to playing any instrument.

Our idea specifically has newer guitarists in mind, those who are still learning the instrument. Once we have the main functionality of our app down we would like to add a learning mode for these users. This mode would teach the users the main “open chords”standard chords beginners learn when they first pick up the guitar. It would be important for this mode to give feedback to the users when they get a chord correct (for example, the phone vibrates).

The users we have in mind will be familiar with the android environment and should be able to download and install the app. While most of our users will have some prior experience with using android apps but they won’t need to be particularly “good” with technology to use String Theory. To make sure our app is as accessible to as many users as possible we will follow good user interface guidelines. A requirement is that our app is intuitive to use and consistent with other apps.

### 2.3 Operational Scenarios

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 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.

### 2.4 Constraints

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

## **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.