

# Software Requirements Specifications (SRS) v1.0

Changes to <http://musicalgorithms.org/3.5/>

**Authors: Chase Myers**

**Customer: Dr. Middleton, Jonathan**

**Instructor: Chris Peterson**

## 1 Introduction

This is a living document and intended for Dr. Middleton, Chris Peterson, and the group working on the project.

This section covers the document's purpose, scope, the definitions, acronyms, and abbreviations of the terms used in this document, and the overall structure of this document.

### 1.1 Purpose

The purpose of this project is working on a real world project to expand each team member's experience. To facilitate this, the team will be helping to develop a website for Dr. Middleton while balancing time and effort to make sure deadlines are met even when certain maintenance hurdles come up.

### 1.2 Scope

Over the course of this project, the team will develop and upload a website for Dr. Middleton based off the input provided by Dr. Middleton. Deliverables will include this document, a documentation outlining the changes made to the existing code as well as a full upload of the new software to the site(<http://musicalgorithms.org/>). This all will be done and delivered by no later than June 11th, 2016.

### 1.3 Definitions, Acronyms, and Abbreviations

This section contains all definitions, acronyms, and abbreviations used in the course of this document.

Term	Definition
------	------------

User	Person who is using the provided interface for its intended purpose.
Map, Mapping, Mapped	An operation that associates each element of a given set with one or more elements of the second set.
Pitch	The quality of a sound governed by the rate of vibrations producing it
Tool Tip	A popup that explains what a particular field or button does and expects.
The website	<a href="http://musicalgorithms.org/">http://musicalgorithms.org/</a>

## 1.4 Organization

In section 2 of this document is the Overall Description. It contains in order, the product perspective, product functions, user characteristics, constraints, assumptions and dependencies.

Section 3 of this document is the specific requirements. This includes in the following order, the external interfaces, functions, performance requirements.

## 2 Overall Description

The produced site will guide the user through an aesthetically pleasing and responsive series of steps to turn entered data into music. The user will easily be able to pick a

variety of options to turn the entered data into melodies to be played and saved to their system for current or later use. Should the user ever need help in understanding any particulars of the website, an easy to find button will be able to explain what each part does, as well as the expected values using descriptions and pictures.

## **2.1 Product Perspective**

In many types of data there are patterns that people look for to try and understand the data. There are several reason the patterns in the data might not be inherently clear or easy to identify. There are several different techniques used to help identify these patterns, one of which is to turn the data into music, for which the one can listen to the data and detect any discernible patterns. The site allows for this system of turning data into music, offering options to change and customize the mapping of the music to fit the user needs.

## **2.2 Product Functions**

The product allows users to choose to either generate input using various algorithms or to entire their own data for the pitch and duration. During these step the user has access to a utilities function that offers additional ways to manipulate their data.

After the user is done with the input, they can move onto mapping, which allows them to pick from a variety of functions and ranges for the previously entered data to map to.

When the user gets to scale options, they can choose from a variety of scales and keys in which their data will be played.

The last step is playing the generated music which offers to save the music to the user's personal device.

## **2.3 User Characteristics**

The user can be from any country, gender, age, and occupation. The typical user will tend to have at least a basic knowledge of music though may not necessarily be familiar with computer navigation. Typical users will have between a high school and college level English reading skill.

Users will be accessing the Website using a variety of browsers and hardware ranging from powerful desktops, to older laptops, and potentially mobile browsers.

## **2.4 Constraints**

The site development will be limited to non-mobile development. Software must work with current technologies and hardware on the current web server. Memory will be limited and must not leak.

## 2.5 Assumptions and Dependencies

In order for the software to work, the site needs to be up and running. The user must be using a compatible browser, have active internet, and be capable of internet navigation.

## 3 Specific Requirements

- Changing versions displayed on the website from version 3.5 to 3.1 until the release of version 3.2.
- Data to note duration mapping must play correctly..
- Morph option must be moved from list of musical scales and placed nearby.
- Site must be aesthetically pleasing and user friendly.
  - Input boxes must match input size and have tooltips.
  - Replace loading carousel with loading bar.
  - For range inputs, must be able to view the piano key mapping.
- Utilities interface must be clear about how a user moves data manipulation results to the data input box.
  - ASCII conversion needs to work without commas between each letter.
  - ASCII conversion needs to be placed in a less conspicuous area of utilities.
  - Utilities needs to be more user friendly with undo and save buttons.
- Sound fonts for the cello and flute replaced with more pleasant and realistic sounding fonts.
- [time permitting] Notation feature will be added.
- [time permitting] Play from any slider position.

### 3.1 External interfaces

For the purposes of this project the only external interface of concern is the website as viewed by the current versions of Chrome, Safari, and Internet Explorer. In each of these browsers the site must be responsive, user friendly, and aesthetically pleasing.

### 3.2 Functions

The user must be able to generate or enter data for both the pitch and duration. The data may be generated by using the functions sine, Fibonacci, integers, pascal, phi, pi, powers, and Euler's number constants.

This data may be modified using the utilities functions reverse, add commas, invert, parse by range, sample rate, replace, decimals, ASCII conversion which prompts for add commas, then converts. A record of these changes should be kept and reversed with an undo button. Once done, the user will be able to save the changes, or cancel.

After input the user can map pitch and duration using division, logarithmic, or modulo functions in the range of 1 to 88. During this time the user can use a replace function to replace any value in the displayed output with a specified value. The user can also modify the displayed output using the same utility functions as before.

The second to the last step will be the scale options in which the user can specify a scale. If the user wants, the data may be morphed in a utilities like function. At this point the user can then save everything to their local device.

The last step, the user may choose an instrument to hear the data using sliders to change the progress and tempo as desired. From this page the song can be played, paused, or stopped instantly, as well as save the music in a MIDI format.

### **3.3 Performance requirements**

While loading, the site must display a loading bar and must be responsive letting the user know at all times when it is progressing and not frozen.