Progress Report

System Requirements Document for the Music XML Converter

LE/EECS 2311 - Software Development Project

Prepared for: Vassilios Tzerpos

Prepared by: Group 16

Maisha Rahman 215083876

Nabaa Gaziy 215820095

Wenxuan Li 216374324

Ali Sheikhi 216777120

Lars Jaylen Palalon 217280041

February 28, 2021

LIST OF CONTENTS

1.0 INTRODUCTION

When designing a software system, it is important to understand the users' needs so that an effective software system can be designed. This document describes the user's needs that must be met in order to create a successful program that can translate text tablatures to MusicXML files.

1.1 Customers' Needs

After talking with our customer, we learned that the customer needs a computer program that:

- ❏ Can accept a guitar, bass or drums tablature then convert them into MusicXML.
- ❏ Able to copy a tablature from anywhere and paste it into the app as well as loading a text tablature file from the local system.

1.2 Purpose:

The purpose of this project is to create software that can precisely denote a guitar, bass, and drums tablature that is in a text file format. As such, our software will allow the user to input these musical tablatures and produce a MusicXML file. More detailed requirements information for our project will be developed as this term progresses.

1.3 Scope:

- ❏ Accepts guitar, bass, and drums tablature.
- ❏ Transform musical tablature to MusicXML file.
- ❏ Imports tablatures in .txt format and exports to .xml format.
- ❏ Has an easy-to-use and user-friendly GUI.
- ❏ Is accessible to most people.

1.4 Intended Audience:

This application is intended to be used by music artists and music enthusiasts, ranging from amateur to professional.

2.0 FUNCTIONALITY

2.1 Functional Requirements:

The application must

- ❑ Accepts guitar, bass, or drums tablature text file or a text copy-pasted in the application's interface
- ❑ Be able to identify which instrument of the tablature.
- ❑ Be able to interact with the user regarding the number of the strings, additional drum pieces or any other missing information from the tablature file.
- ❑ Notify the user if they wish to change settings
- ❑ Notify the user when the file has been converted
- ❑ Continue to function without an active network connection
- ❑ Would output a .xml file.

2.2 Detailed Non-Functional Requirements:

The system should be

- ❑ Compatible with any software(Windows, Mac, Linux)
- ❑ Open-source and free to use software
- ❑ Usable without a network connection

3.0 USABILITY

3.1 Use Cases:

Who will be the main users of this system?

- Musicians, music teachers, students, and music enthusiasts.

What are the Use Cases?:

1. Scenario 1: Load text file
    a. User uploads a text file into the application, ensuring that the tablature is either guitar, drum, or bass. The user can copy the text from a website then paste it into the application window. Or, the user can load a text file from the file system by clicking on File→ Open and then choosing the file from his computer system. The user can also load the file by clicking on *Open* button, then choosing a file from his computer system.

2. Scenario 2: Convert text file to MusicXML
    a. The program can convert tablatures into MusicXML files, so that the file can be used in other applications. The user will provide the guitar, bass, or drum tablature and load it into the application. Then the user will confirm that he wants this text to be converted. The system will respond by converting the tab to MusicXML file and notifying the user that the conversion happened. Then this file will be provided to the user.
    b. The user does not need to worry about the guitar or base having additional strings or the drum set having extra instruments. The system will accommodate these changes from the tablature and let the user know about them.
3. Scenario 3: Modify the song
    a. The user can change the scale of the song or modify it easily using our application. The user will provide the tablature, and load it into the application. Then edit the tablature within the application window. The system will respond by creating the MusicXML file with the new changes.

3.2 Required Accessibility

- ❑ Ability to change GUI language
- ❑ Consistency within the user interface
- ❑ Easy to learn with reasonable learning curve
- ❑ Having keyboard shortcuts for different functions

3.3 Required Interface Aesthetics

- ❑ Dark mode and light mode
- ❑ Easy to use and minimalistic design
- ❑ The program's GUI provides menus, buttons, and different theme modes to allow the user to have the best experience

4.0 RELIABILITY

4.1 Availability

Available for free.

4.2 Required Recoverability

This program handles errors and failures by

- ❑ Allowing the user to change settings to their preference
- ❑ Informing the user if excessive strings or other instruments are used
- ❑ Informing the user with any occurring problems while importing the tablature or outputting the MusicXML file

5.0 SUPPORTABILITY

5.1 Maintainability

Software would be open-source.

6.0 CONSTRAINTS

6.1 Design Constraints

- ❑ The program accepts drum, guitar, and bass tablature only
- ❑ The program creates .xml files only
- ❑ The program only accepts .txt files

6.2 Implementation Constraints

- ❑ Application has to decide the instrument if it is a 6 or 7 string guitar, 4 or 5 string bass and different types of drums.
- ❑ Application needs to work with a variety of tablature styles, also able to denote the extra information in the tablature properly.

7.0 CONCLUSION

As of right now, we created system requirements for our software application based on the customer's needs. More detailed requirements information for our project will be developed as this term progresses.