EECS2311: SOFTWARE DEVELOPMENT PROJECT

TAB2XML

System Requirement Specifications

March 6, 2022

PREPARED FOR

Vassilios Tzerpos
Lassonde School of Engineering

PREPARED BY

Hiba Jaleel - 215735020
Kuimou Yi - 216704819
Kamsi Idimogu - 216880288
Maaz Siddiqui - 216402927

Table of Contents

1. Introduction

1.1 Purpose

The purpose of the Software Requirement Specifications document is to describe the functions our software, TAB2MXL, the MusicXML previewing system will be able to perform and how exactly our software will be able to achieve the desired results.

1.2 Intended Audience

This document is intended for:

Customer/Client: The customer or client will be expressing what features they would like the software to have, and through this document ensure that the software will cover the aspects they would like.

Project Manager: The project manager will be a point of contact for any conveyance between the developers and the customer/client and communicate any new demands or changes in the customer/client's requirements. The project manager will also be setting deadlines for the developers to follow and assist in maximising developer efficiency.

Developers: The developers will be working on implementing the requirements of this software as laid out by the customer/client through their development of the software.

Musicians (also referred to as user/users): Musicians are defined as people who enjoy listening to music and partake in playing text-based tablature music. They will be the end-users of this software.

1.3 Intended Use

The software development timeline of the TAB2XML is from January 10, 2022 to April 11, 2022. The main costs involved in this project are time, energy, and electricity. The software's ability to perform the required tasks will be the measure of success. Our team has determined the core features/functionality, important features, and nonfunctional requirements to be the deliverables of this project. Risk will be evaluated by grading it's exposure, probability of occurrence, and loss size (in days). Based on the context and severity of the risk, effective mitigation measures will be implemented.

2. Overall Description

2.1 Product Perspective

TAB2XML can be accessed through GitHub, under the following repository: https://github.com/CCSCovenant/TAB2XML/releases

2.2 Product Overview

TAB2XML will allow users to convert their text-based tablatures into music sheets. Additionally, it will allow users to play those music tablatures after the conversion.

2.3 User Class Characteristics

2.3.1 Use Case #1

Title*:* Preview Music Sheet
Primary Actor: Musician
Preconditions*:* The musician has started the application and input the text-based tablature music.
Success Scenario*:* The system displays sheet music. The musician previews the music sheet.

2.3.2 Use Case #2

Title: Saving Music Sheet
Primary Actor: Musician
Preconditions: The musician has started the application, input the text-based tablature and previewed the music sheet.
Success Scenario: The musician saves the music sheet.

2.3.3 Use Case #3

Title: Play Music
Primary Actor: Musician
Preconditions: The musician has started the application, input the text-based tablature and previewed the music sheet.
Success Scenario: The system identifies the type of instrument that the inputted text-based tablature represents. The system plays the music.

2.4 User Stories

Although the scope of TAB2XML is vast, here are some of the user stories where it would be well implemented.

2.4.1 User Story #1

As Max, a music student, I want to hear the music sheet I am practising so I can ensure that I am playing the music sheet correctly.

2.4.2 User Story #2

As Bob, a music composer, I enjoy hearing how my music sounds when played with different instruments.

2.4.3 User Story #3

As Ben, a music writer, I like to be able to write my music easily and hear how it sounds.

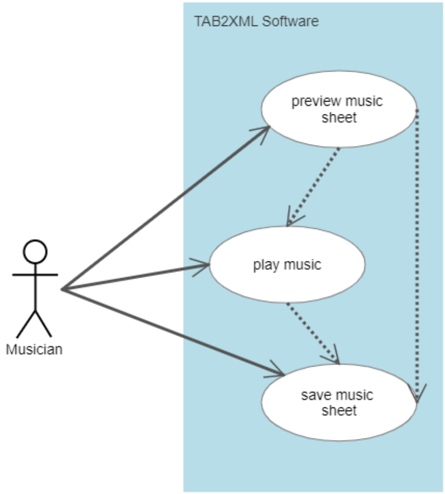
2.4.4 User Story #4

As Bob, a music composer, I like to be able to write my music easily and save my music sheet for later use.

2.4.5 User Story #5

As Sam, a music teacher, I like to be able to print music sheets out for my students.

2.5 Use Case Diagram



3. System Features and Requirements

3.1 Functional Requirements

3.1.1 Tablature to MusicXML conversion

3.1.1.1 Description

This feature allows the users to convert their text-based tablature into a MusicXML file.

3.1.1.2 Stimulus/Response Sequences

- The user inputs their text-based tablature into the designated text input area in the system.
- The system identifies the type of instrument that the inputted text-based tablature represents.
- The system outputs it into a MusicXML file

3.1.2 MusicXML Visualisation

3.1.2.1 Description

- This feature allows the user to visualise the MusicXML file. The visualisation feature previews the text-based tablature in the form of a music sheet and allows users to save the file.

3.1.2.2 Stimulus/Response Sequences

- The user will select desired MusicXML-based tablature.
- The user will preview the music sheet.
- The user can export it as a PDF file in order to print or share.

3.1.3 MusicXML playing

3.1.3.1 Description

This feature allows the customer to play the MusicXML file.

3.1.3.2 Stimulus/Response Sequences

- The user will select desired MusicXML-based tablature.
- The user will preview the music sheet.
- The user can play the music sheet.

3.2 Nonfunctional Requirements

3.2.1 User Interface

The user interface will be designed to ensure simplicity and ease of use. The user interface will prioritise efficiency in both speed and use.

3.2.2 Software Abilities

The software should:

- Be accurate in the results
- Be easy to use and understand
- Be reusable, and handle requests back to back
- Be robust, and handle large text-based tablatures
- Be efficient, fast and responsive