TRANSLATION OF STANDARD MUSICAL NOTATION TO NUMBERED MUSICAL NOTATION MOBILE APPS

BY

CHIA TZE WEI

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology

(Kampar Campus)

FEBRUARY 2026

COPYRIGHT STATEMENT

This Final Year Project proposal is submitted in partial fulfillment of the requirements for the degree of Bachelor of Computer Science (Honours) at Universiti Tunku Abdul Rahman (UTAR). This Final Year Project proposal represents the work of the author, except where due acknowledgment has been made in the text. No part of this Final Year Project proposal may be reproduced, stored, or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the author or UTAR, in accordance with UTAR's Intellectual Property Policy.

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Mr. Soong Hoong Cheng who has given me this bright opportunity to engage in this Music translation project. It is my first step to establish a career in IT music field. A million thanks to you.

Finally, I must say thanks to my parents and my family for their love, support, and continuous encouragement throughout the course.

ABSTRACT

This project focuses on the development of a music translation system designed to assist musicians and students in converting standard musical notation into numbered musical notation (Jianpu). The field of study covers music technology and human-computer interaction. The main problem addressed is the difficulty for beginner learners to understand and interact with standard music notation. The proposed solution is a cross-platform system that enables users to import and export input file, translate them into numbered musical notation, edit music note, and play back the audio for learning purposes. The system integrates various technologies: HTML and CSS for frontend design and user interface, Dart for user interactions, and JavaScript for back-end logic. Additional functionality includes editing both notations, importing music sheet in various image file type and exporting files in MusicXML, PNG and PDF formats, and playback music. The methodology follows the Phased Development approach from the Software Development Life Cycle (SDLC), where each module is built and refined incrementally. The research and development process includes requirement planning, analysis, system design, implementation, and maintenance phases.

Area of Study (Maximum 2): Music Technology, Human-Computer Interaction

Keywords (Maximum 5): Music Translation, Standard Musical Notation, Numbered Musical Notation, MIDI, Sheet Music, Jianpu

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

SDLC	Software Development Life Cycle
MIDI	Musical Instrument Digital Interface
PDF	Portable Document Format
XML	Extensible Markup Language
API	Application Programming Interface
HTML	HyperText Markup Language
PNG	Portable Network Graphics

Chapter 1

Introduction

This project will be developing a music translation system where users can use it to translate standard music sheet into numbered music sheet, which also call Jianpu. It also can do playback, edit note and export it into PDF or image form. Below will show the problem statements and motivation, project objectives, project scope, impact, significance and contribution and background information.

1.1 Problem Statement and Motivation

This project is motivated by the need to develop a music translation system. The aim is to simplify the process of translating music sheet and help users overcome several existing challenges, such as lack of availability of free complete translating tools, complex interface, and the concern of security problem. The problem statements are stated as follows:

I. **There are currently no free and reliable applications available in the market that can fully translate standard musical notation to numbered notation.**
Even though there are many application and websites available in the market for translating the notation, many of them need to pay to use it. It is not worth and expensive for the people who only need to translate for a few times only. Moreover, even the few applications that offer limited free access, it often come with significant restrictions, such as watermarked output or a cap on the number of notations that can be translated. These limitations reduce their practicality and usability, and especially for casual users, students, or amateur musicians. The lack of flexibility in these tools makes it hard for more people to use them. It becomes a barrier for those who want to learn or try music notation but don't want to spend money on expensive software.

Furthermore, there are also some software or applications that are cost-free to do the translation. However, these software have some bugs in doing translation such as missing symbols after translation, or incorrect recognition of complex or unclear notes. For example, some systems mistakenly interpret a note with a tremolo bow

symbol (typically represented by three slashes) as a thirty-second note, leading to inaccurate translation results. These errors reduce the effectiveness of such tools and user may need to correct them manually, which make them trouble and inconvenient.

II. Most translation tools are not user friendly for music beginners

Many existing music notation translation tools are designed with advanced users or professionals in mind, often requiring prior knowledge of music theory. This makes it difficult to use for beginners, students, or casual musicians who may not be familiar with technical aspects of music. The complexity of these tools from navigates multiple settings and options to understanding output formats will make the users get trouble and confuse to implement it.

For example, some software applications may need user to input MIDI number, which are a parameter that is necessary to determine the key signature before translation. MIDI numbers are a specialized concept that most casual users or beginners are unlikely to be familiar with, creating an extra obstacle for those who just want to translate music without dealing with complicated technical details.

III. Existing online converters may be an untrusted websites or applications that expose users to security risk such as viruses or malware.

Although there are many online platforms that offer free musical notation translation services, not all of them are safe or reliable. While free access may seem beneficial to users, especially casual learners and musicians, these websites often come with hidden dangers. In addition to providing inaccurate translation results, some sites pose significant cybersecurity risks.

For example, the website will free to give the translation service and after click “generate Jianpu” button, the virus or malware will spread into user’s devices. Then, all the information about the user will be retrieved to the scammer such as IC number, IP address and so on. Thus, it is unsafe to rely on unverified websites for notation translation, especially when users may unknowingly compromise their devices and personal information.

1.2 Objectives

The objective of this project is to develop an system that can translate standard musical notation into numbered musical notation and playback music. Below is the object for this project:

I. **To create a simple and user-friendly system interface design and function for users.**

This project will design an intuitive and accessible user interface for a wide range of users, especially beginners, students, and casual musicians who may not have extensive knowledge of music theory or technical terms. The user input will be designed with easier concepts and understandable words to not make user confusing. Features such as step-by-step instructions, file upload, label description for each tool components, and keyboard interaction will be implemented to enhance user experience. The layout should be clean and not messy to ensure the first-time users are easier to use.

II. **To develop a cost-free and secure system that able to translate the musical note without restrictions.**

This project will mainly aim to develop an system that provides a complete and unrestricted translation service from standard musical notation to numbered musical notation without requiring any payment, subscription, or usage limits. The tools in this system will be made freely accessible to all users without imposing watermarks, export limitations, or function limitations. In addition, this system will be able to translate and export fully the standard musical notation without missing any symbol such as the beat line of the note.

III. **To integrate existing numbered music rules and functions into an system which are more secure and convenient for users**

This project will integrate the existing function and translation rules to form a complete feature within an system. There are some existing systems doing well in translate each note into numbered musical notation, but it does not translate along the beat of line at the same time, while there are also some existing systems able to translate the note along with the beat line but the accuracy of the translation is low. Thus, this project will integrate the advantage for each existing system to create a more reliable and accurate translation tool. By combining the strengths of various

systems, it will reduce errors during translation and minimize the need for manual corrections.

Problems	**Objectives**
There are currently no free and reliable applications available in the market that can fully translate standard musical notation to numbered notation	To develop a cost-free and secure system that able to fully translate the musical note without restrictions
Most translation tools not user friendly for music beginners	To create a simple and user-friendly system interface design and function for users.
Existing online converters may be a untrusted websites or applications that expose users to security risk such as viruses or malware.	To integrate all the existing numbered music rules and functions into an system which are more secure and convenient for users.

Table 1.1 Summary of Problem and Objectives

1.3 Project Scope

The project scope will state the important feature that is included in this system to translate the standard musical notation based on the user's input in specific type of file format, edit the note by user interaction and playback music based on musical note. Through these sufficient functionalities, it will overcome the limitation of the reviewed methods.

I. Translate standard musical notes to numbered musical notes

The system shall be able to translate the standard musical note based on the user's uploaded input file. Using Optical Music Recognition (OMR) technology, the system will scan and detect musical symbols from supported file formats. Once recognized, each note will be systematically converted into its corresponding numbered form using existing coding language with predefined translation rules and the translated output will then be displayed to the user in a clear and structured format.

II. Edit musical note

The system will be able to let the user edit the musical notes after they have been translated. Users can make adjustments such as changing the pitch, duration, or position of specific notes, as well as inserting or deleting notes. The system will provide a user-friendly interface, allowing users to interact with the numbered musical notation directly using visual tools. This gives flexibility for user to correct the unsatisfied note or customize their note.

III. Playback

The system shall be able to playback the numbered musical notation using Web Audio API functionality. After translation and editing, the user can press a "Play" button to hear how the music sounds. This helps users verify the accuracy of the translated notes and supports auditory learning. The playback will reflect changes made during editing, including tempo and note adjustments.

IV. Export and Import Sheet Music

The system will accept input in any image files containing standard musical notation. The image file will then be converted into MusicXML to make the translation of note. In addition to importing, the system also allows users to save the translated numbered musical notation in multiple formats. Supported export formats include MusixXML, PDF, and PNG file formats.

1.4 Impact, Significance and Contribution

The novelty of this project is to produce a music translation system using Dart, HTML, CSS, and JavaScript. The expected outcome is a user-friendly system that simplifies the process of translating musical notation. This system will enable the user to translate the standard musical note into numbered musical note based on the uploaded music sheet file. Additionally, this system provides playback functionality and offers editing capabilities for numbered notations. Furthermore, this system enables user to import and export in form of MusicXML, image format and PDF format. This feature allows users to seamlessly integrate their work with other platforms used for music editing, learning, or publishing

1.5 Background Information

Music plays an important role in every human's life. The essence of music lies in its ability to evoke emotions, convey narratives, and connect individuals across cultures and time (Dmitriy, 2025). It serves many purposes and brings numerous benefits. For example, music can create soothing sounds that help to relax the mind and relieve stress. In today's world, people are not just content with listening to music through speakers, they are more preferring to create their own music by learning and playing musical instruments. This not only allows for personal expression but also deepens one's connection to the art of music. To learn a music instrument, people may need to spend at least 1 months or more to familiar with the instrument. To be expert, it may take about 1 year or more so that people can be very fluent and comfortable to play their instrument with nice music sound.

In the beginner's learning phase, people need to learn to recognize the musical notation, including the basic elements such as note values (whole, half, quarter, eighth), rests, clefs (like treble and bass), and key signatures. They also need to understand how pitch is represented on the staff, how time signatures work to organize rhythm, and how accidentals like sharps, flats, and naturals alter notes. As learners advance, they also need to understand dynamics (such as forte and piano), articulation marks (like staccato or legato), and other expressive notations. In addition to these foundational elements, beginners should also become familiar with the different types of musical intervals and how they are used to build melodies and harmonies. Recognizing intervals, both visually and aurally, allows students to better understand musical structure and improve their ability to sight-read and perform music. This foundational knowledge of musical symbols and intervals serves as a stepping stone toward understanding how music is communicated and shared globally.

Musical notation is a universal language for communicating music. There are 2 types of notations which are traditional standard notation and numbered musical notation. Traditional standard notation, which is five-line staff, has been widely adopted in Western music education and performance. For the pitch, it uses note heads, stems, flags, and other symbols to represent pitches on a five-line staff. For the rhythm, it uses note durations (whole notes, half notes, quarter notes, etc.) and rests to represent rhythm. For the key signature, it uses clefs and key signatures to indicate the key to a piece. For example, the symbol ☐ is the treble clef, which is used for higher-pitched instruments such as the violin, flute, and the right hand music of the piano. In the treble clef, the notes in the spaces of the staff spell F-A-C-E from bottom to top. The two sharp symbols ♯♯ indicate the key signature. In this case, the

sharps are placed on the F and C lines, meaning that all F notes become F♯ and all C notes become C♯.

However, in Chinese music publications, the numbered musical notation, which also called ‘jianpu’ is widely used. Numbered musical notation offers a simple method for notating music using numbers to represent pitch and some additional symbols for rhythm. It uses numbers (1, 2, 3, 4, 5, 6, 7) to represent the scale degrees, with dots above or below the numbers indicating the octave. For the key signatures, it uses a movable "do" system, making it easier to transpose music to different keys. For example, if it is a C key, the 1 will represent C; if it is a D key, the 1 will represent D, and so on. In numbered musical notation, both the intervals from 3 to 4 and from 7 to 1 are half steps. This differs from other intervals, such as from 2 to 3 or from 5 to 6, which are whole steps. For the rhythm, it uses dashes and other symbols to represent note durations. It also easier for the people who are not expert or experienced in music to understand and performing music without requiring memorizing the complex symbol or notation rules found in traditional musical notation system. This will increase the sense of musical accessibility, enabling people with little or no formal training in music area to engage with music more confidently.

Nowadays, people live in an era of advanced technology and rapid development. Modern technology offers great convenience, including access to various online music platforms and tools. These online resources allow individuals to listen to music anytime and anywhere, learn to play instruments through virtual lessons, and even compose and share their own music digitally. This has making it easier for people of all ages and skill levels to engage with music in meaningful ways. Whether through system, video tutorials, or digital sheet music, technology has transformed how we experience and interact with music. In this digital age, technology has also greatly supported the learning and dissemination of both traditional standard notation and numbered musical notation. With the help of interactive system, online notation software, and video tutorials, learners can explore and understand the structures of both systems more intuitively. For example, music education platforms can visually demonstrate how notes are placed on a staff or how numbers correspond to pitch and rhythm. Some software even allows users to convert between traditional notation and numbered notation, making it easier for learners from different musical backgrounds to access and share music.

In recent years, many applications have been developed to support the translation between traditional musical notation and numbered musical notation. For instance, software

such as MuseScore3 allows users to input music using the traditional five-line staff, and through plugins or extensions, users can generate numbered notation outputs. Another example is EOP NMN Master, which is specifically designed to convert sheet music into numbered notation, helping users who are more familiar with the "jianpu" system. These tools not only assist in cross-notation understanding but also promote greater musical inclusivity by accommodating learners from diverse musical traditions. With such advancements, technology continues to bridge the gap between Western and Eastern music education methods, offering users the flexibility to learn and share music in the format they are most comfortable with.

Music remains a powerful and universal form of expression that transcends boundaries of language, culture, and time. As people increasingly seek to connect with music not only as listeners but also as performers and creators, understanding musical notation becomes a crucial part of their journey. While traditional standard notation offers a detailed and structured approach, numbered musical notation simplifies learning for many, especially in Eastern music contexts. By embracing both traditional and numbered systems, and leveraging the tools available in the digital age, learners can gain a more comprehensive understanding of music, fostering greater inclusivity and creativity in the musical world.

1.6 Report Organization

This report is organized into 6 chapters: Chapter 1 Introduction, Chapter 2 Literature Review, Chapter 3 System Methodology/Approach, Chapter 4 System Design, Chapter 5 System Implementation, Chapter 6 System Evaluation and Discussion, and Chapter 7 Conclusion and Recommendation. The first chapter is the introduction of this project which includes problem statement, project background and motivation, project scope, project objectives, project contribution, and report organization. The second chapter is the literature review carried out on several existing IoT applications in the market to evaluate the strengths and weaknesses of each product. The third chapter discusses the system architecture for overall system, including use flowchart, case diagram, and activity diagram. The fourth chapter describes the overall system design, which includes system flow, integration components, user requirements, system specifications, and user interface design. Chapter five is about the system methodology used to develop this system, and also specify the details of project workflow, timeline, technology and tool involved, and implementation outcome. Chapter six is a chapter that discusses the testing setup and result, project challenges, and objectives

evaluation. The last chapter is conclusion, which conclude the report and some recommendations for this system.

Chapter 2

Literature Review

2.1 Technique Used in Translation

The review of technique used in translation is using MusicXML file to do translation. Before investigate the content of MusicXML file, it needed to extract the file extension with .mxl into .xml file. The reasons is when open mxl file in notepad, the content is not readable. After convert to xml file, the content is change to readable data. First, rename the file extension from .mxl to .zip. Then only extract zip file to .xml file. Example of a song showed in standard musical notation is shown below:

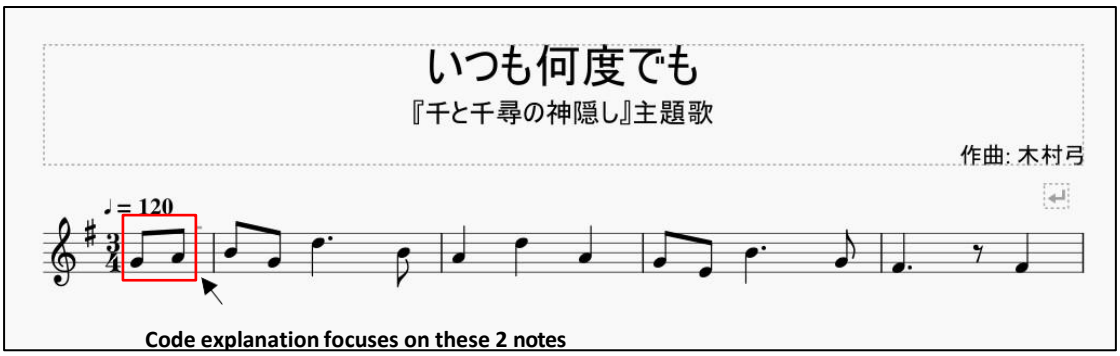


Figure 2.1 Standard musical notation with the song name いつも何度でも

Below is the coding explanation for .xml file with the song name いつも何度でも:

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE score-partwise PUBLIC "-//Recordare//DTD MusicXML 3.1 Partwise//EN"
   "http://www.musicxml.org/dtds/partwise.dtd">
3  <score-partwise version="3.1">
```

Figure 2.2 Coding Part 1

The file is XML and use UTF-8 letters. It follows the MusicXML 3.1 rules. The music score is in partwise format, which the note is organized by instrument.

```
4    <work>
5      <work-title>いつも何度でも</work-title>
6      </work>
7    <identification>
8      <creator type="arranger">Che-Huai Lin</creator>
9      <encoding>
10       <software>MuseScore 2.3.2</software>
11       <encoding-date>2018-09-24</encoding-date>
12       <supports element="accidental" type="yes"/>
13       <supports element="beam" type="yes"/>
14       <supports element="print" attribute="new-page" type="yes" value="yes"/>
15       <supports element="print" attribute="new-system" type="yes" value="yes"/>
16       <supports element="stem" type="yes"/>
17       </encoding>
```

Figure 2.3 Coding Part 2

<work-title> - It show the title of the piece is 「いつも何度でも」.

<creator> - The person who arranges the song is Che-Huai Lin.

<encoding> - It shows the information about this song. The song is made with MuseScore 2.3.2, the file is created on 24/9/2018, and the supported things are accidentals, beams, printing new pages/systems, stems.

```
<defaults>
  <scaling>
    <millimeters>7.05556</millimeters>
    <tenths>40</tenths>
    </scaling>
  <page-layout>
    <page-height>1584</page-height>
    <page-width>1224</page-width>
    <page-margins type="even">
      <left-margin>56.6929</left-margin>
      <right-margin>56.6929</right-margin>
      <top-margin>56.6929</top-margin>
      <bottom-margin>113.386</bottom-margin>
      </page-margins>
    <page-margins type="odd">
      <left-margin>56.6929</left-margin>
      <right-margin>56.6929</right-margin>
      <top-margin>56.6929</top-margin>
      <bottom-margin>113.386</bottom-margin>
      </page-margins>
    </page-layout>
  <word-font font-family="FreeSerif" font-size="10"/>
  <lyric-font font-family="FreeSerif" font-size="11"/>
  </defaults>
```

Figure 2.4 Coding Part 3

<defaults> - It show the basic page settings

<scaling> - It set 7.05556 mm as basic size for 40 tenths. (A tenth is a standard unit in MusicXML, 10 tenths = 1 staff space, which is the distance between 2 lines). The example of staff space as shown in Figure 2.5.

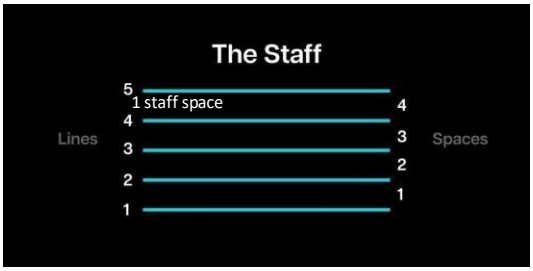


Figure 2.5 Staff space

<page-layout> - The height of page is 1584 units; the width of page is 1224 units. Type="even" and type="odd" is means that set margins differently for left and right pages.

<word-font> - The font using is FreeSerif font at size 10.

<lyric-font> - The font using in lyrics is FreeSerif font at size 11.

```
<credit page="1">
  <credit-words default-x="612" default-y="1527.31" justify="center" valign="top" font-size="24">いつも何度でも
  </credit-words>
  </credit>
<credit page="1">
  <credit-words default-x="1167.31" default-y="1427.31" justify="right" valign="bottom" font-size="12">作曲：木村
  弓</credit-words>
  </credit>
<credit page="1">
  <credit-words default-x="612" default-y="1470.61" justify="center" valign="top" font-size="14">『千と千尋の神隠
  し』主題歌</credit-words>
  </credit>
```

Figure 2.6 Coding Part 4

<credit> - It describe the things that appear as text on the first page: Title「いつも何度でも」, Composer 木村弓, and the subtitle 『千と千尋の神隠し』主題歌.

```
<part-list>
  <score-part id="P1">
    <part-name>Piano</part-name>
    <part-abbreviation>Pno.</part-abbreviation>
    <score-instrument id="P1-I1">
      <instrument-name>Piano</instrument-name>
      </score-instrument>
    <midi-device id="P1-I1" port="1"></midi-device>
    <midi-instrument id="P1-I1">
      <midi-channel>1</midi-channel>
      <midi-program>1</midi-program>
      <volume>78.7402</volume>
      <pan>0</pan>
      </midi-instrument>
    </score-part>
  </part-list>
```

Figure 2.7 Coding Part 5

<part-list> - It tells what instruments are used in the music and how they are set up.

<score-part id= “P1”> - It define one part of instruments which the part-name is piano

<part-abbreviation> - It is a short form of instrument’s name, which is Pno. In Piano.

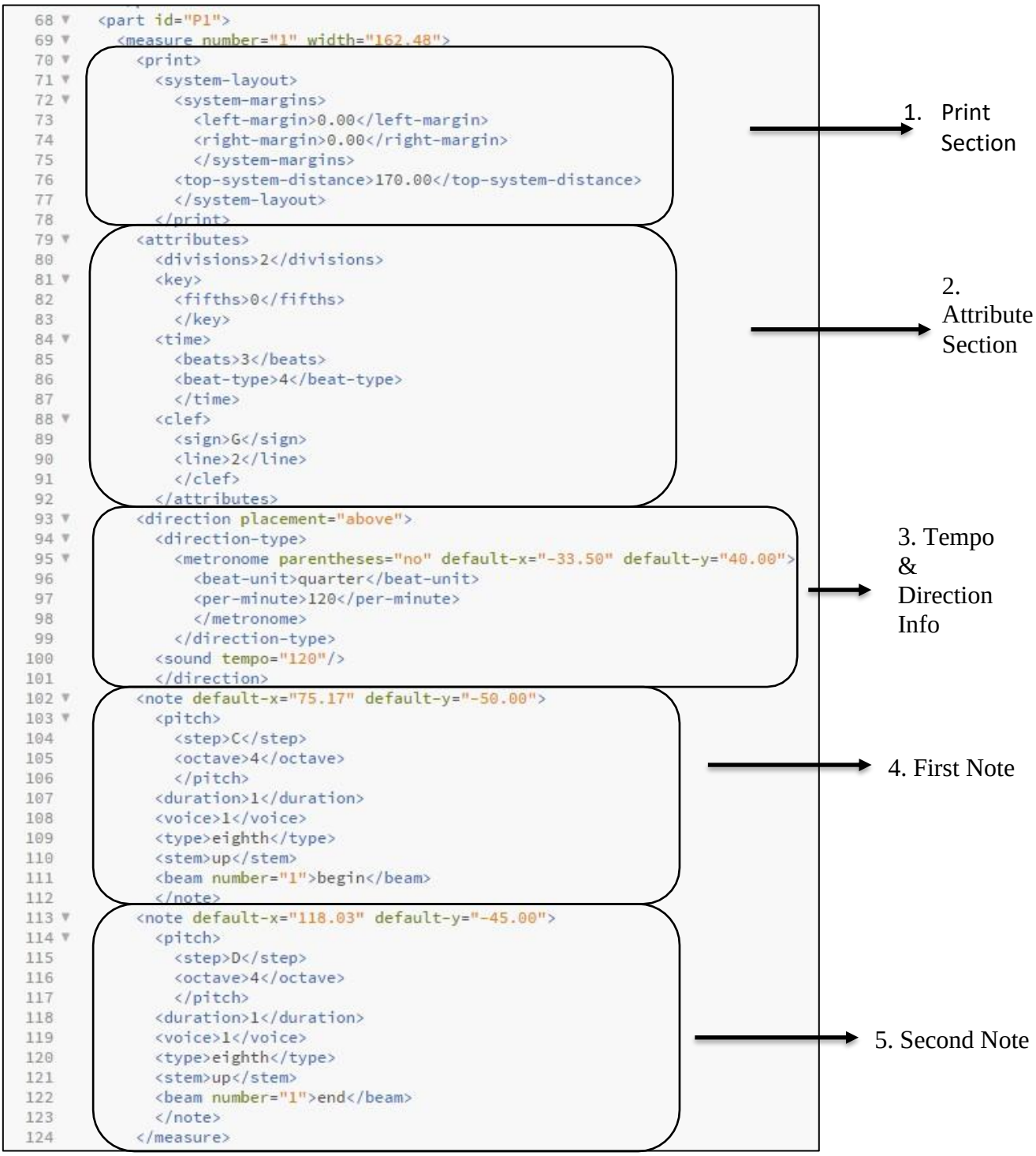
<score-instrument id= “P1-I1”> - It define the instruments used which is Piano.

<midi-device id= “P1-I1”> - It set the MIDI sounds, which direct the P1-I1 to use port 1 to play the MIDI sound.

<midi-channel>1</midi-channel> - means that it uses Channel 1 in MIDI which have a total of 16 channels.

<midi-program>1</midi-program> - means that it plays the piano sound. (program 1 = Acoustic Grand Piano in General MIDI).

<volume>78.7402</volume> - means how loud the piano will sound.




```
<part id="P1">
  <measure number="1" width="162.48">
    <print>
      <system-layout>
        <system-margins>
          <left-margin>0.00</left-margin>
          <right-margin>0.00</right-margin>
          </system-margins>
        <top-system-distance>170.00</top-system-distance>
        </system-layout>
      </print>
    <attributes>
      <divisions>2</divisions>
      <key>
        <fifths>0</fifths>
        </key>
      <time>
        <beats>3</beats>
        <beat-type>4</beat-type>
        </time>
      <clef>
        <sign>G</sign>
        <line>2</line>
        </clef>
      </attributes>
    <direction placement="above">
      <direction-type>
        <metronome parentheses="no" default-x="-33.50" default-y="40.00">
          <beat-unit>quarter</beat-unit>
          <per-minute>120</per-minute>
          </metronome>
        </direction-type>
      <sound tempo="120"/>
      </direction>
    <note default-x="75.17" default-y="-50.00">
      <pitch>
        <step>C</step>
        <octave>4</octave>
        </pitch>
      <duration>1</duration>
      <voice>1</voice>
      <type>eighth</type>
      <stem>up</stem>
      <beam number="1">begin</beam>
      </note>
    <note default-x="118.03" default-y="-45.00">
      <pitch>
        <step>D</step>
        <octave>4</octave>
        </pitch>
      <duration>1</duration>
      <voice>1</voice>
      <type>eighth</type>
      <stem>up</stem>
      <beam number="1">end</beam>
      </note>
    </measure>
```

Figure 2.8 Coding Part 6

Print Section

<measure number=”1” width=”162.48”> - This is the first bar of the song. The bar is 162.48 units wide. The example of bar as shown in Figure 2.9.

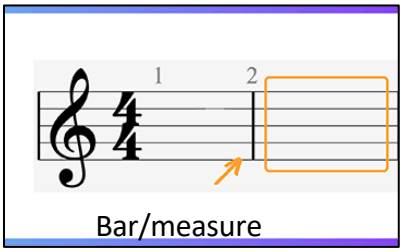


Figure 2.9 Bar/measure

<print> - it sets the printing or layout setting:

<top-system distance>170</top-system distance> - means that the space from the top of page to the first staff line is 170 units.

1. **Attribute Section**

<attributes> - it sets the important musical setting:

<divisions>2</divisions> - 1 quarter note (1 beat) is equal to 2 divisions.

<key><fifths>0</fifths></key> - The key signature has 0 sharps( ) and 0 flats(♭). So, the key of this song is C major or A minor. The key signature list as shown in Figure 2.10.

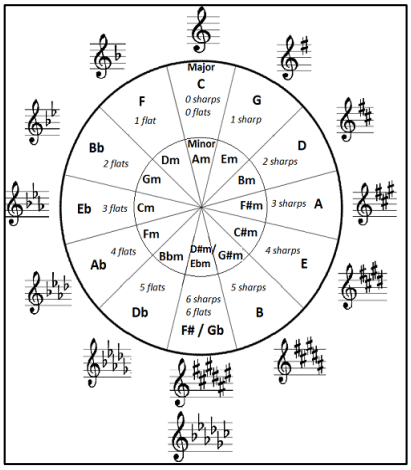
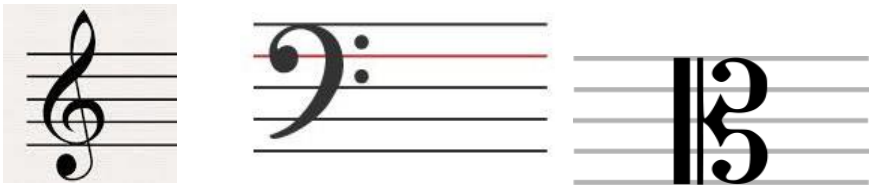


Figure 2.10 Key signature list

<time><beats>3</beats><beat-type>4</beat-type></time> - Time signature is ¾, which means that 3 beats per bar.

<clef><sign>G</sign><line>2</line></clef> - The clef is G clef (treble clef) that place the line 2 is G. The symbol represented for each clef is shown in Figure 2.11.



G F C

Figure 2.11 Symbol for each clef

2. **Tempo and Direction Info**

   <per-minute>120</per-minute> - The tempo is 120 beats per minutes.

   <beat-unit>quarter</beat-unit> - Each beat is a quarter note.

   <sound tempo="120"/> - This tell the playback speed for software.

3. **First Note**

   <step>C</step> and <octave>4</octave> - This is C4 note (middle C).

   <type>eighth</type> - This is an eighth note .

   <duration>1</duration> - This refer to the length of note in terms of division.

   <beam number="1">begin</beam> - It identifies the beam level and start the bean group.

4. **Second Note**

   This format is the same as first note. It has D4 note and an eighth note to end the beam group.

2.2 Review Existing System in Market

2.2.1 MuseScore3

MuseScore3 is a free and open-source music notation software that allows users to compose, arrange, edit, and print sheet music with ease. There are many functions in MuseScore3:

1. **Create sheet music using standard musical notation**

   MuseScore3 included many types of instruments such as piano, guitar, orchestra, voice and so on. It enables users to write music with variable type of instrument. Users can use a mouse and keyboard to modify the musical notation. It supports key

signatures, time signatures, accidentals, dynamics, articulations, and more. The list of instruments as shown in Figure 2.12.

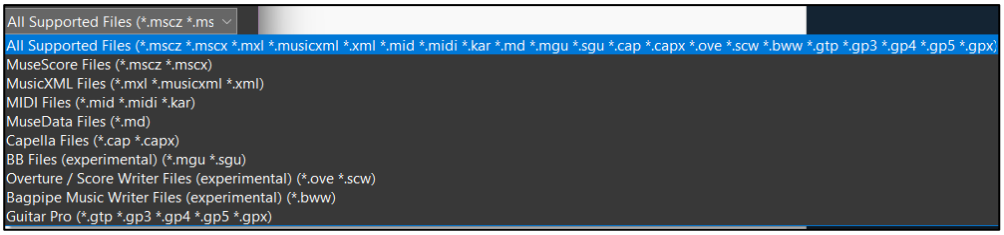
Figure 2.12 List of Instruments

2. **Able to playback the music**

MuseScore3 has playback music function with realistic instrument sounds. After done to write music, users able to listen the music result immediately without the needed to export to other application to play back. Users also can import the file to play back the music. It is useful for checking arrangements, ear training and practice. The button of playback music is shown in Figure 2.13.

3. **Import and export music**

It is a useful function where MuseScore3 has a lot of choices to import many type of files such as MIDI, MusicXML, MuseScore files (.mscz). This enable user to modify the musical notation or play back the music easily and convenient. MuseScore3 also enable user to export to PDF (music sheet), MIDI, MusicXML,



MP3, WAV, PNG and so on. The lists of supported import files as shown in Figure 2.14 and supported export file shown in Figure 2.15.

Figure 2.13 Supported import files

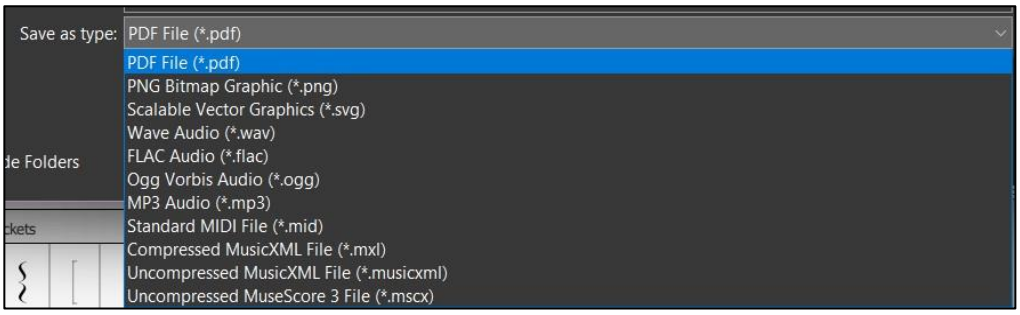


Figure 2.14 Supported export files

4. Modify existing music score

MuseScore3 allow users to edit, rearrange, or improve an existing music score. Users can change the pitch, duration and rhythm if there have any error in music sheet. Other than this condition, users can rearrange the music note to produce a new type of music style. Figure 2.15 is showing the note before editing, and Figure 2.16 is showing the note after editing.

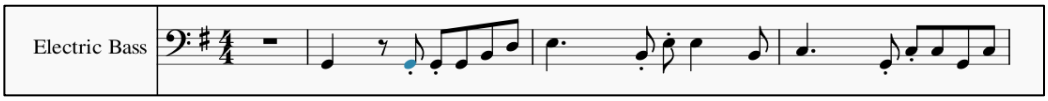


Figure 2.15 Before note editing

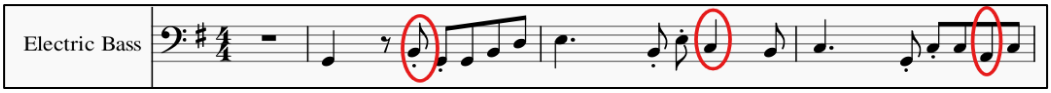


Figure 2.16 After note editing

5. Plugins and Extensions

MuseScore3 able to let the users use the plugins such as Jianpu (numbered notation) or ABC import/export. Plugins is a small script that able to add new features or automate tasks in MuseScore3. For example, Jianpu Plugin is used to convert standard musical notation into numbered musical notation. It will convert each note one by one and show the translated number under the musical note. These plugins and extensions are mostly found in GitHub or other code- sharing websites.

Limitations

1. **Not able to write music in the form of numbered musical notation.**

   MuseScore3 only supports standard musical notation. It is not able to create music in form of numbered musical notation.

2. **Not user friendly for beginners**

   The function of Jianpu plugins is complex to use for the user who doesn't know about technical things such as MIDI numbers. Although nowadays it has AI tools to ask them about what the MIDI number for each key signature is, but it is making it trouble that user need to ask all the time when they need to translate the music sheet. The user interface of Jianpu plugins as shown in Figure 2.17.

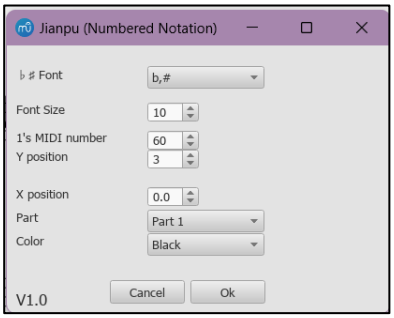
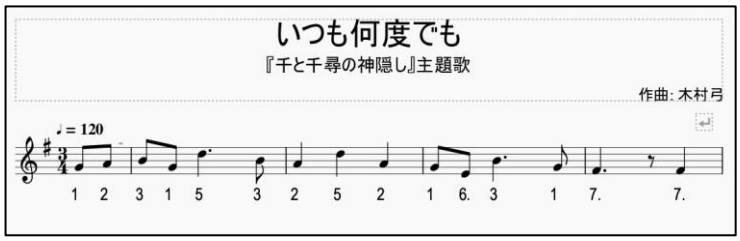


Figure 2.17 User interface of Jianpu Plugins

3. **Symbol showed after translating to numbered musical notation are not complete.**

   Numbered musical notation can translate using plugins from GitHub. However, the function of the plugins is not completed. For example, after translating music sheets using Jianpu plugin, it shows the numbered musical notation under each note but it is not able to show the line of the time signature. Figure 2.18 shows the result after



   using Jianpu plugins to translate the note.

Figure 2.18 Result after note translation using Jianpu plugins

2.2.2 Fanqie Jianpu by using command prompt

Fanqie Jianpu is an open-source toolset and platform designed to convert MusicXML files into Jianpu using command prompt. This software is found in GitHub, which are contribute by wanderor and lzh9102. It is using Python to implement the conversion, which has provided a Python-based conversion script (converter.py) to parses the MusicXML input and generate jianpu style of command in command prompt. Users can then copy the jianpu command into Fanqie Jianpu website, and it will automatically convert the command into jianpu.

Way to use

1. Download the open-source code from GitHub. (https://github.com/lzh9102/musicxml_to_jianpu/blob/master/converter.py)

2. Run Python interpreter, converter.py, and MusicXML file with jianpu99 grammar format in command prompt.

   Location of python interpreter:

   "C:\Users\tzewei\AppData\Local\Programs\Python\Python311\python.exe"

   Location of converter.py:

   "C:\Users\tzewei\OneDrive\Desktop\converter.py"

   Location of xml file:

   "C:\Users\tzewei\Downloads\musicxml_to_jianpumaster\musicxml_to_jianpu -master\tests\case4.mxl"

3. After running this command, it will show the result of jianpu style of command in command prompt. Below is the command after translating MusicXML into numbered musical notation.

   V: 1.0

   B: いつも何度でも

   D: G P: 3/4 J: 120

   Q: 1'/ 2'/ | 3'/ 1'/ 5'. 3'/ | 2' 5' 2' | 1'/ 6/ 3'. 1'/ |

   Q: 7. 0/ 7 | 6 7 1'/ 2'/ | 5 1' 2'/ 3'/ | 4' 4'/ 3'/ 2'/ 1'/ | Q: 2'. 0/ 1'/ 2'/ | 3'/ 1'/ 5'. 3'/ | 2' 5' 2' | 1'/ 6/ 6 7/ 1'/ | Q: 5 - 5 | 6 7 1'/ 2'/ | 5 1' 2'/ 3'/ | 4' 4'/ 3'/ 2'/ 1'/ |

   Q: 1' - - | 0 - 3'/ 4'/ | 5' 5' 5' | 5' 5'/ 6'/ 5'/ 4'/ |

   Q: 3' 3' 3' | 3' 3'/ 4'/ 3'/ 2'/ | 1' 1' 1'/ 7/ | 6 7 7/ 1'/ |

   Q: 2' 2'/ 3'/ 2'/ 3'/ | 2' - 3'/ 4'/ | 5' 5' 5' | 5' 5'/ 6'/ 5'/ 4'/ |

Q: 3' 3' 3' | 3'/ 4'/ 3'/ 2'/ 1'/ 7/ | 6 6/ 7/ 1'/ 2'/ | 5 1' 2'/ 3'/ | Q: 2'. 2'/ 2'/ 1'/ | 1' - - ||

4. These commands are then copy and paste to Fanqie Jianpu website to generate a formal Jianpu. The result is shown in Figure 2.19.

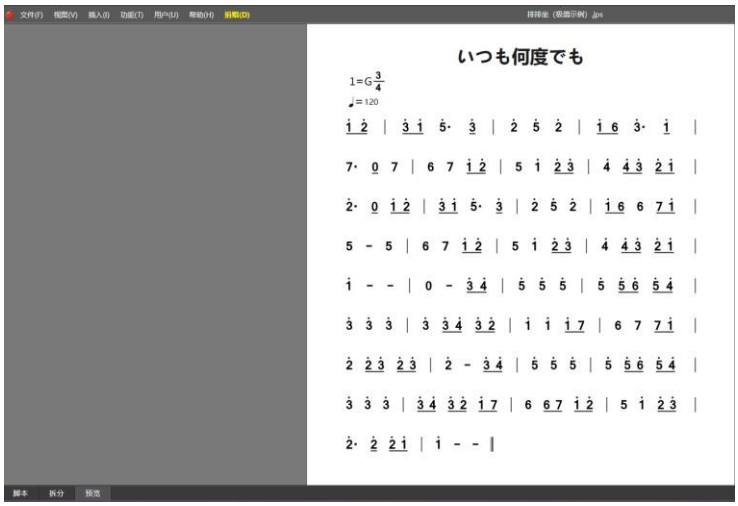


Figure 2.19 Result of note translation using Fanqie Jianpu

5. It able to export in form of PDF, JPG and SVG. Figure 2.20 has shown an expanding tab that includes the option of file type for exporting the music sheet.

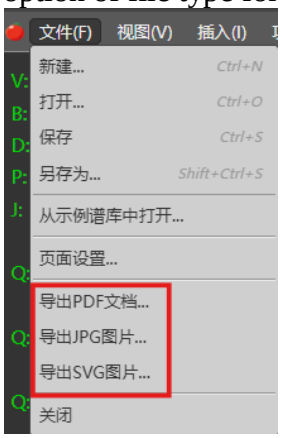


Figure 2.20 Option of file export types

Limitations

1. **Limited functions**

   If the numbered musical notation has exceeded 1 page, it would not move the remaining note to the next page, and it totally cannot add another new page to place the note. The extra note that has not enough space to place then disappears from the music sheet. Figure 2.21 shows the result when there is not enough space for the note.

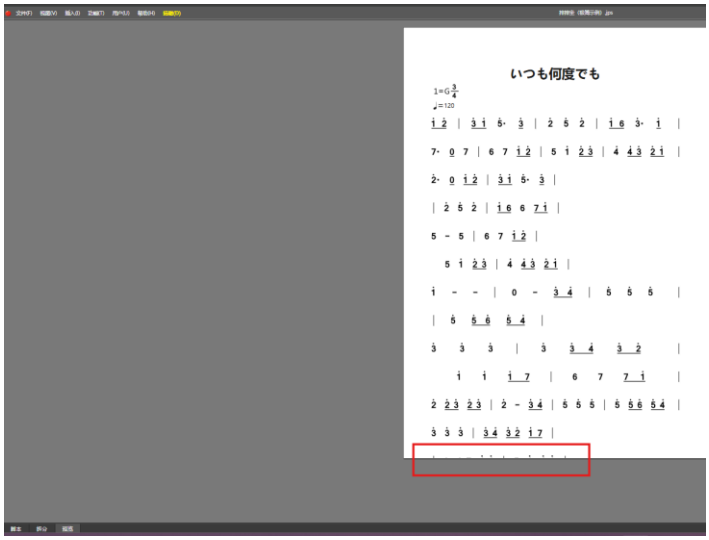


Figure 2.21 Result of the note placing when not enough space

2. **Lack of intelligence**

   The bar needs to be modified manually when the space does not look satisfying and clear. It is not able to automatically estimate each bar needs how much space and change their spacing according to the estimation. This will make users waste more time in modifying the music bar.

3. **It is not a friendly application for international user**

   Since the application is founder by the person in China, thus the language used is Chinese. It became a trouble for user who does not know Chinese.

4. **Complex when using command prompt to translate**

   Users may not know how to implement using command in command prompt Users who are not experts in using computers may find it a problem. For example, they may not know where the locations of that mxl file.

2.2.3 EOP NMN Master

EOP NMN Master is a software tool developed by Everyone Piano (EOP) for editing, creating, or playing numbered musical notation (NMN). There are many functions in EOP NMN Master:

1. **Jianpu Editing**

Users can create and edit numbered musical notation by using keyboard and mouse. It supports input of pitch, duration, lyrics and so on. It is easy to use because the user only need to type the number on keyboard to create a numbered musical note.

2. **Score Playback**

Users able to play back their music in piano sound and string sound. This will make it easier for the users to check the feeling of music that they has created. The playback music button is shown as Figure 2.22.

Figure 2.22 Playback Music Button in EOP NMN Master

3. **File Support**

Users able to import in 2 types of file format which is .eopn and .mid. It also enables users to export in form of midi file. Figure 2.23 showed the supported import file and figure 2.24 showed the supported export file.

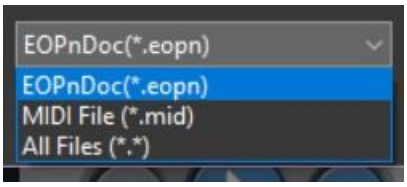


Figure 2.23 Supported import file

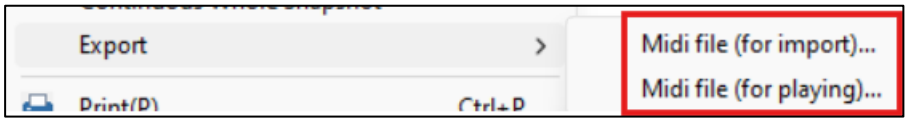


Figure 2.24 Supported export file

<u>Limitations</u>

1. **The beat of the music sheet may get wrong.**

The beat of music may change although the note is correct.

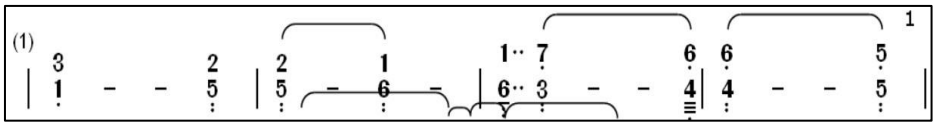


Figure 2.25 Wrong beat

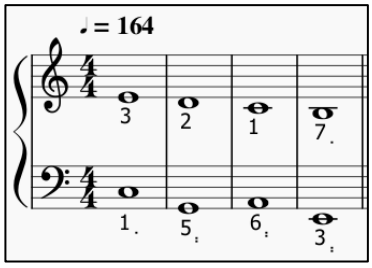


Figure 2.26 Correct beat

2. **The arrangement of the music sheet is messy**

   After translation, the spacing from top and bottom of each bar is not enough when there is multiple note place together. Users need to adjust manually to make music sheet clearer and tinier.

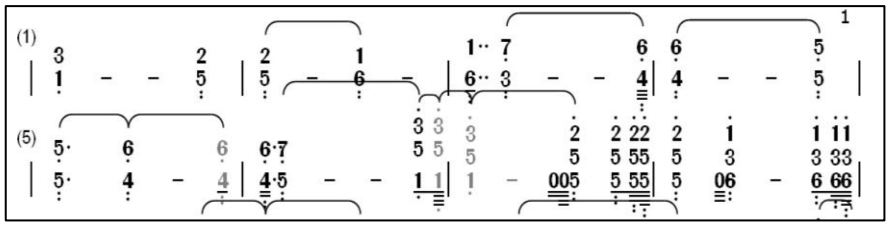


Figure 2.27 Result when there is no space up and down of the bar

3. **Not able to translate multiple music sheets together**

   It only supports one line of music sheet. If there are multiple music sheets needed to translate together, the second line of music notes will be placed below the first line music note. It will make the music sheet messier and confuse to view while playing the note especially when playing piano (piano have left hand note and right hand note which needed to separate each section up and down).

2.3 Summary

Functionality	MuseScore3	Fanqie Jianpu	EOP NMN Master	Proposed system
Edit standard note	✓	×	×	×
Edit numbered note	✓	✓	✓	✓
Playback music	✓	✓	✓	✓
Adjust playback speed	×	×	✓	✓
Adjust playback sound according to respective key signature	×	×	×	✓
Translate to numbered note based on key signature	×	×	×	✓
Import MIDI	✓	×	✓	×
Import MusicXML	✓	✓	×	✓
Import PNG	×	×	×	✓
Export PDF	✓	✓	×	✓
Export PNG	×	×	×	✓
Show duration in numbered note	×	✓	✓	✓
Show pitch in numbered note	✓	✓	✓	✓

Table 2.1 Comparison between 3 existing systems with proposed system

Chapter 3

System Methodology/Approach

This chapter describes the methodology used in the system development. It includes Unified Modeling Language (UML) diagrams, such as use case diagrams, to illustrate the functionalities available to users within the system. Activity diagrams are also provided to represent the workflow, including the control flow and decision-making processes involved in each activity.

3.1 System Design Diagram/Equation

3.1.1 System Architecture Diagram

Figure 3.1 shows an architecture diagram of how user interacts with system. Users provide a music sheet in image file format which is PNG, JPG, JPEG, WEBP, or BMP, and convert into PNG format so that HOMR able to do conversion and extraction. The PNG is then sent to HOMR to do image processing and music note extraction and then convert it into MusicXML file. Furthermore, users can use converted MusicXML file and upload them to JianPu Converter to convert MusicXML into JianPu. The converted PNG and MusicXML file are automatically saved in user's device.

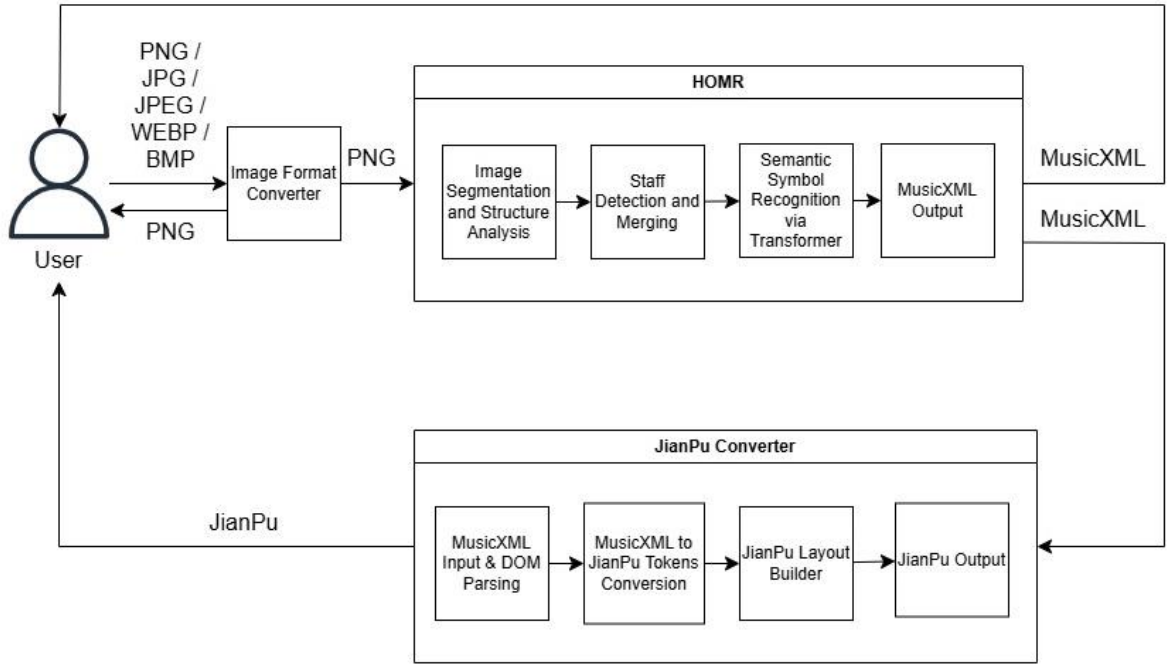


Figure 3.1 System architecture design

3.1.2 User Case Diagram

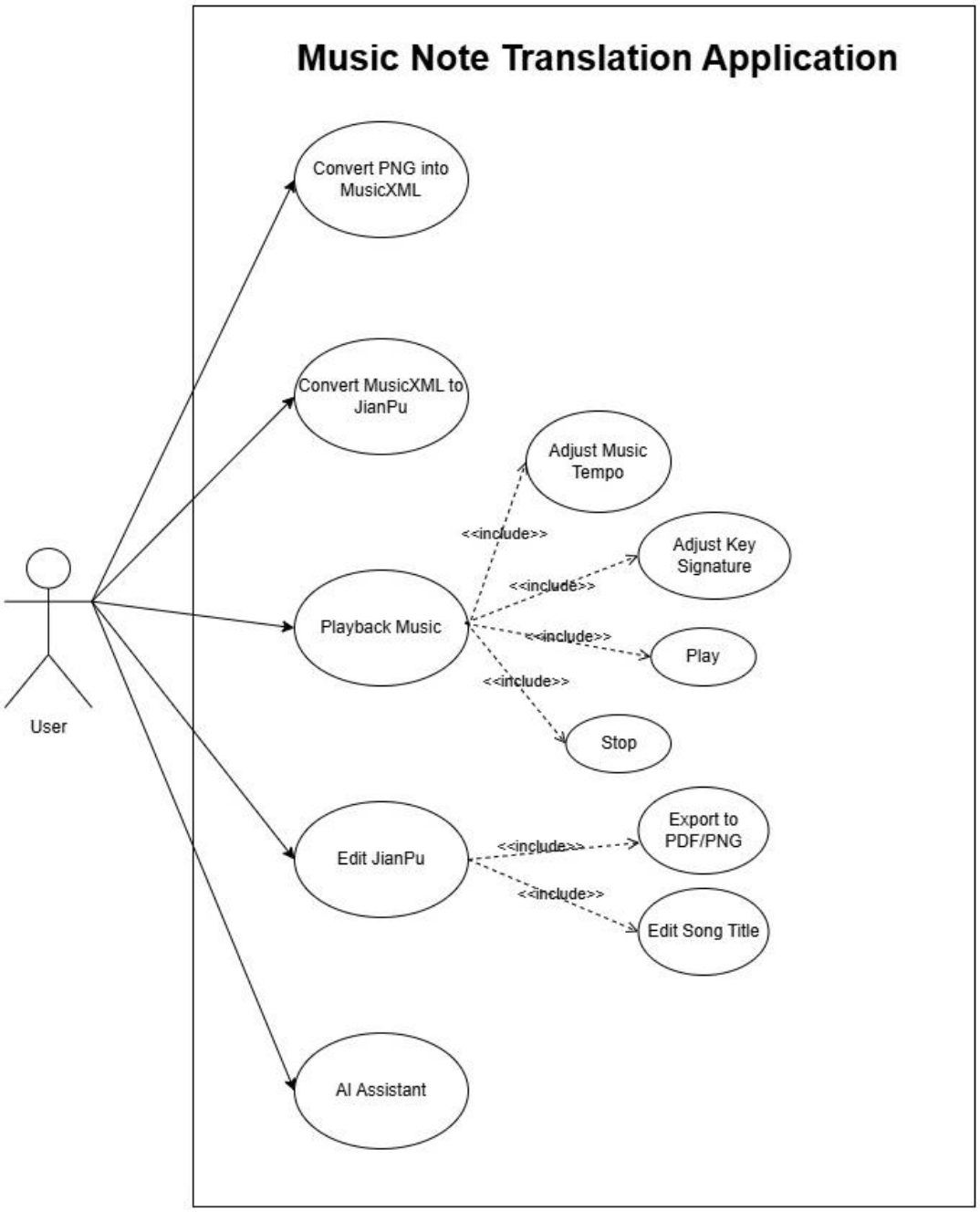


Figure 3.2 Use Case Diagram

3.1.3 Activity Diagram

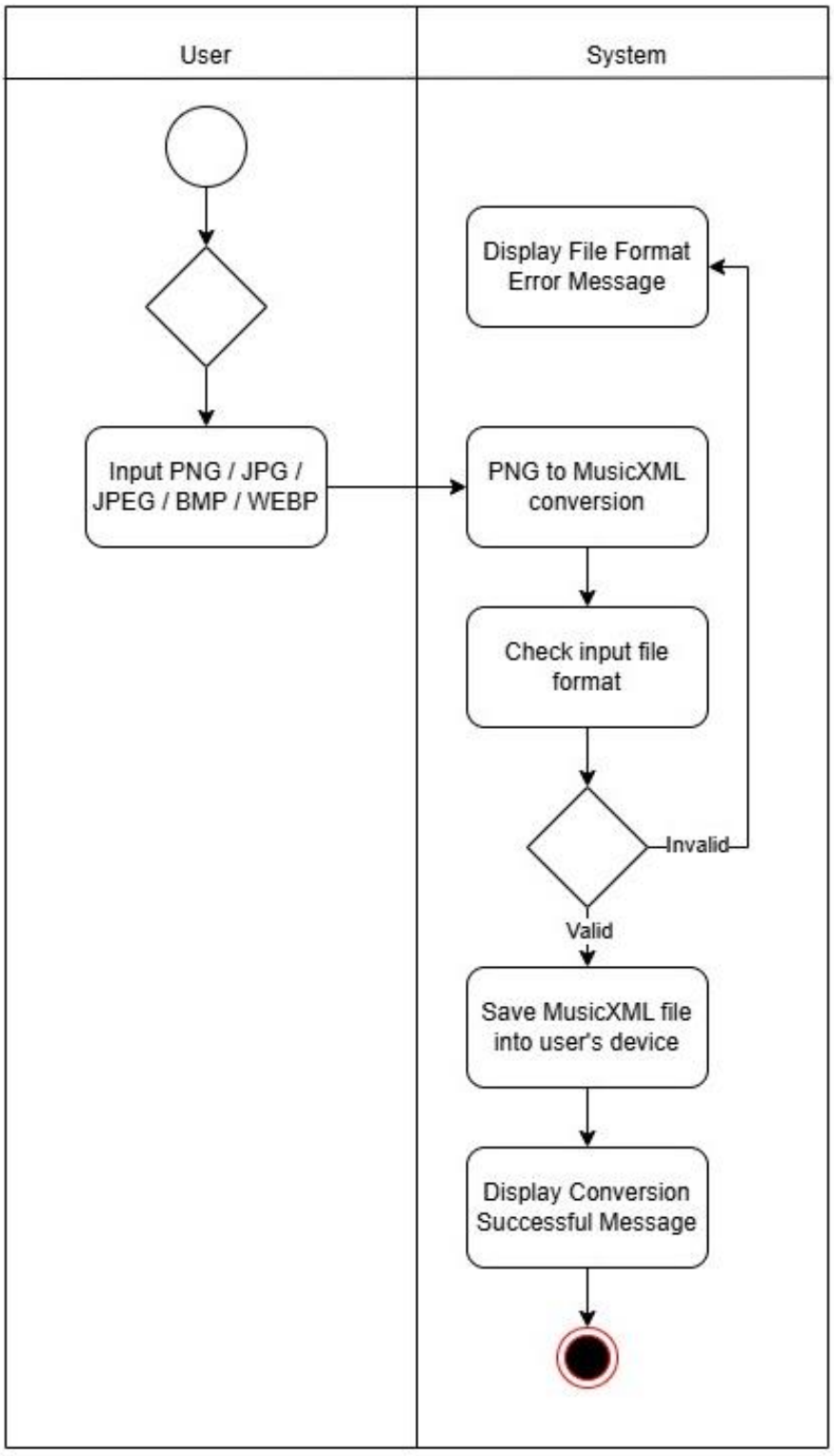


Figure 3.3 Activity Diagram for PNG to MusicXML Conversion

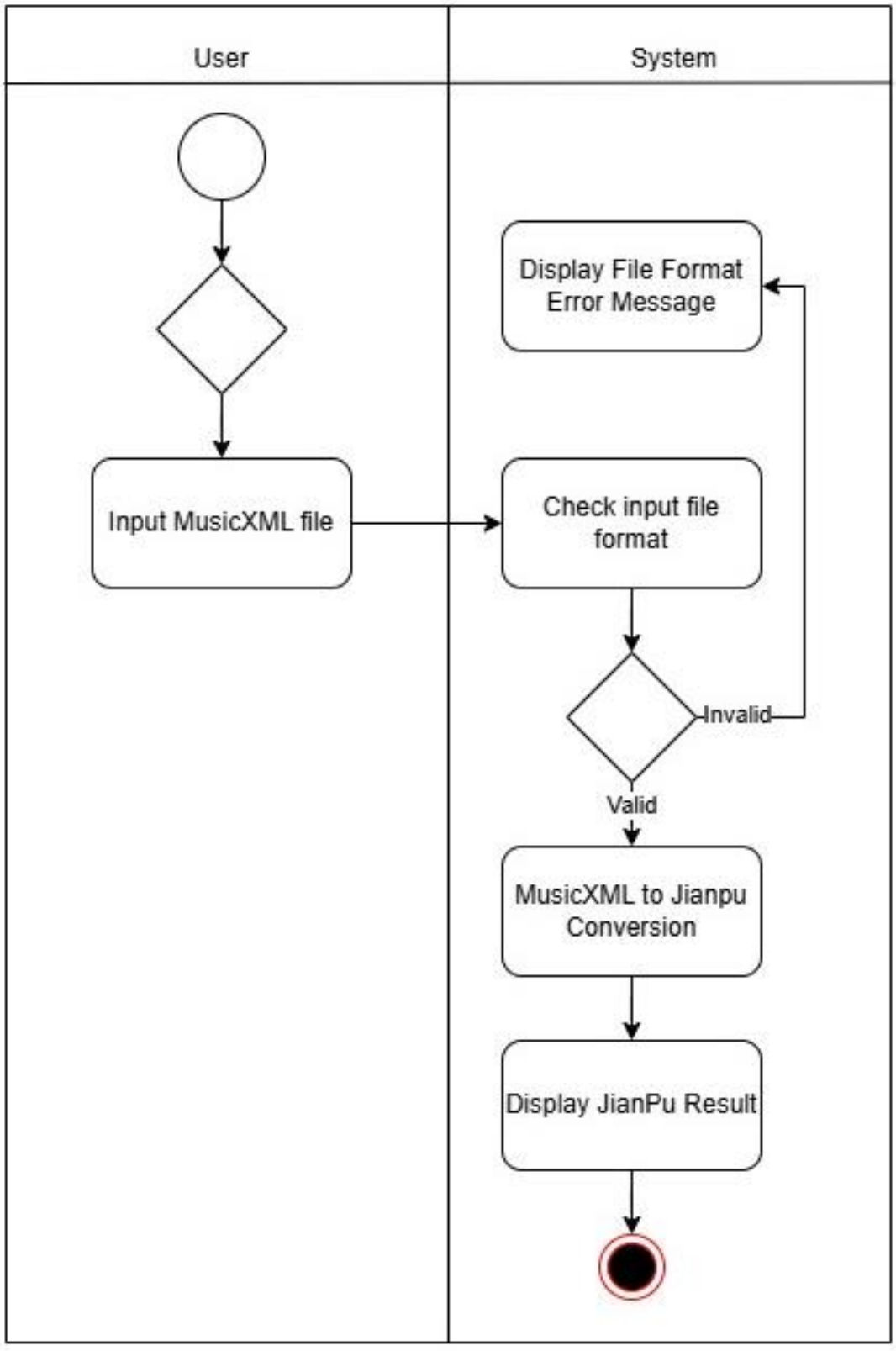


Figure 3.4 Activity Diagram for MusicXML to Jianpu Conversion

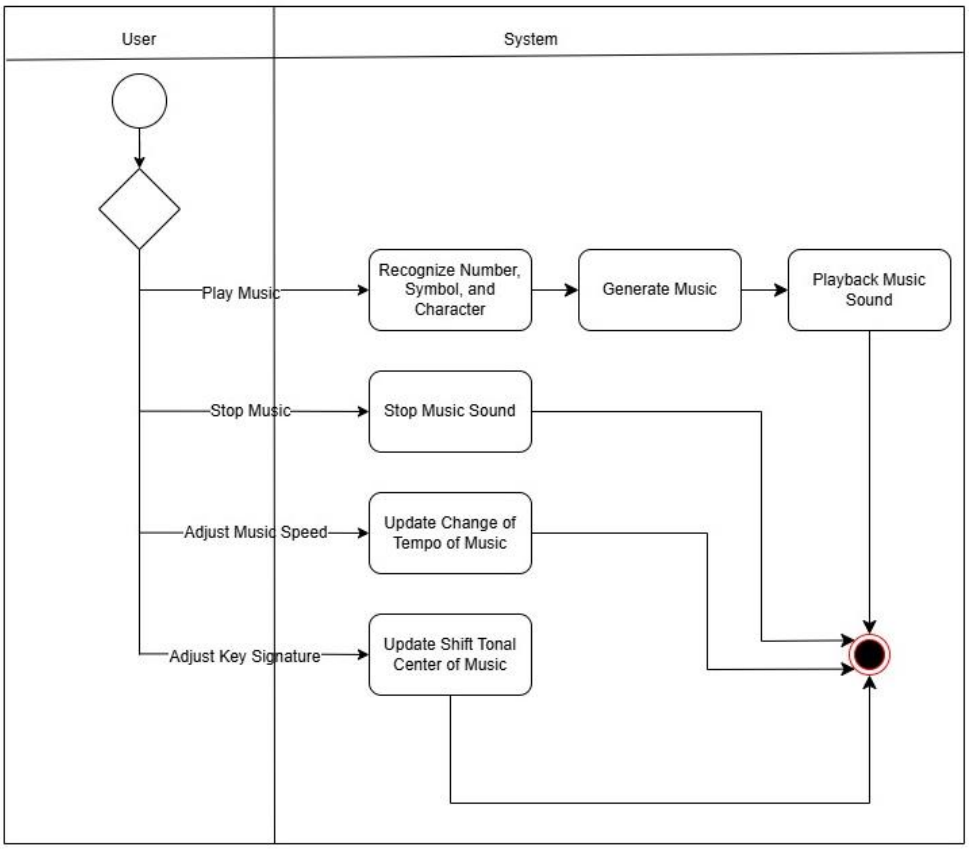


Figure 3.5 Activity Diagram for Playback

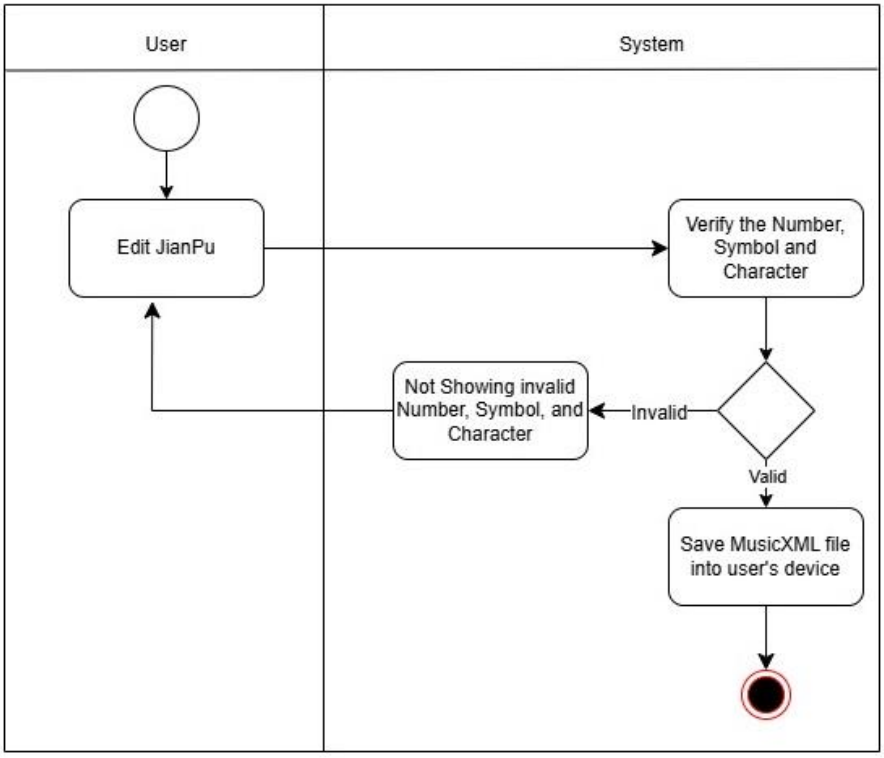


Figure 3.6 Activity Diagram for Edit Jianpu

Chapter 4

System Design

This chapter discusses the system design of the project. The developed system utilizes **HOMR**, an Optical Music Recognition (OMR) software from GitHub, which is designed to transform images of sheet music into machine-readable MusicXML format. In addition, the system integrates the **Gemini API** to develop an AI-powered assistant that supports user input interpretation and intelligent processing. Besides the Gemini API, several **Web APIs** are also utilized in the system. Furthermore, this system integrates **JianPu Ascii Font** that get from GitHub, which aim to generate JianPu symbol and note to form a JianPu music sheet.

4.1 System Flow

4.1.1 Image to MusicXML Conversion

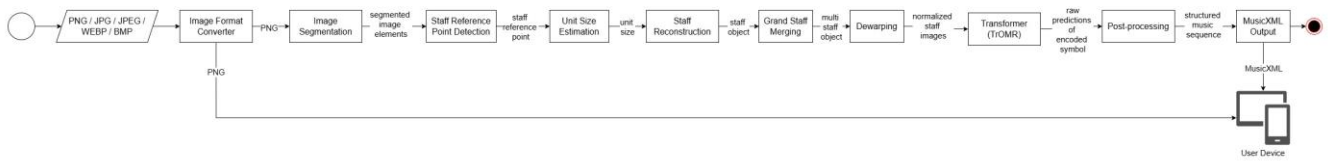


Figure 4.1 System Flow for PNG to MusicXML Conversion

The flow of PNG to MusicXML conversion is shown as above. Firstly, system will receive a music sheet in image form (PNG/JPG/JPEG/WEBP/BMP) that uploaded by users. Then, system convert these image form into PNG format, which convenient in image processing and staff detection process. After system converted image into PNG format, it automatically saved into user's device and proceed to next process.

From the image segmentation process to the final output, the system uses HOMR, a type of Optical Music Recognition (OMR), designed to convert images of sheet music into a machine-readable MusicXML format.

In the image segmentation stage, HOMR breaks the sheet music image into smaller fragments. These fragments are then used to detect staff reference points, such as clefs and bar lines, which act as anchors for accurately locating each staff on the page.

Next, the system uses these reference points to estimate the spacing between the five staff lines by calculating the pixel distance between them. Using this estimated unit spacing, it reconstructs complete five-line staff from partial or fragmented detections.

After that, the system analyzes nearby staffs and groups them into musical systems when appropriate. For example, a grand staff combines treble and bass staffs, while instrument groupings may include multiple staffs that belong to different instruments but are performed together.

Once the staff are reconstructed and grouped, the image is normalized and dewarped. Each normalized staff image is then processed by a transformer model, which generates a sequence of encoded symbol predictions representing musical elements.

These encoded symbols are then refined through post-processing steps to form structured music sequences. Finally, these structured sequences are converted into MusicXML format and saved to the user's device.

4.1.2 MusicXML to JianPu Conversion

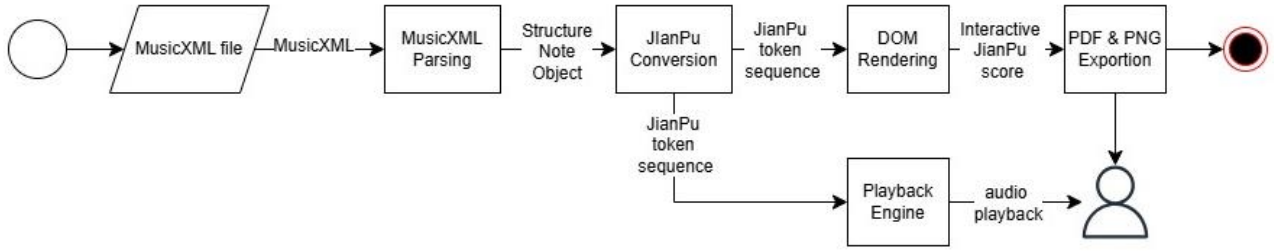


Figure 4.2 System Flow for MusicXML to PNG Conversion

The flow of MusicXML to PNG conversion is illustrated above. First, the user uploads a MusicXML file to the system. The system reads the file using the FileReader API and loads its content. Next, MusicXML is parsed into structured note objects such as {step, octave, alter, duration, measure}) in JavaScript using the DOMParser API.

These structured note objects are then converted into a Jianpu token sequence. The tokens are rendered into an interactive Jianpu score by constructing a staff-like layout using HTML. The score is displayed using Jianpu ASCII font style.

In addition to rendering, the Jianpu tokens are also used to generate audio playback through the Web Audio API. Finally, the interactive Jianpu score can be exported as PDF or PNG using the Canvas API (html2canvas) and PDF API (jsPDF), depending on the user's selection.

This system has integrated 6 **Web APIs** that list as below:

1. **Web Audio API**

- AudioContext: Manages audio processing and controls playback timing
- OscillatorNode: Generates sound waves with specific frequencies (pitch)
- GainNode: Controls the volume (amplitude) of the audio signa

2. **File API**

- FileReader: Used to read and load the MusicXML file uploaded by the user into the system.

3. **Canvas API**

- html2canvas: Used to capture HTML elements and convert them into an image format (canvas rendering), enabling export as PNG.

4. **PDF API**

- jsPDF: Used to generate PDF from image.

5. **Web Storage API**

- localStorage: Used to store AI chat history in the user's browser, allowing the conversation to persist even after page refresh.

6. **DOM Parsing and Serialization API**

- DOMParser: Used to convert XML or HTML string into DOM document objects.

4.2 User Requirements

Functional Requirements

1. User shall be able to convert image into MusicXML file.
2. User shall be able to convert MusicXML file into JianPu.
3. User shall be able to edit in JianPu.
4. User shall be able to playback the sound from the music sheet.
5. User shall be able to adjust the speed of sound.
6. User shall be able to adjust the key signature.
7. User shall be able to export JianPu into PDF and PNG file.
8. User shall be able to ask AI Assistant about music theory.

Non-Functional Requirements

1. The system shall be run in Desktop.
2. The system shall be able to playback the music according to user's selected speed and key signature.
3. The system shall be able to automatically fill in the music sheet information such as title, speed and key signature into JianPu converter website.
4. The system shall provide a user-friendly interface for beginner.
5. The system shall be able to answer questions about music theory.

4.3 System Specification

The system should be able to run with the minimum requirements as shown below:

Description	Specification
Processor	Intel(R) Core (TM) i3-10110U or higher
Operating System	Windows 11 Home Single Language
Memory	4GB of system memory
Storage	Minimum 3GB

Table 4.1 System Specifications

4.4 System User Interface Design

User interface for this system is shown below. The button, progress and result will show at the centre of the user interface, while the upper part will display the title of each page. Besides that, there will also has another interface which direct user to website. The control panel is mainly placed at the upper left of the screen, while the result will be displayed at the centre.

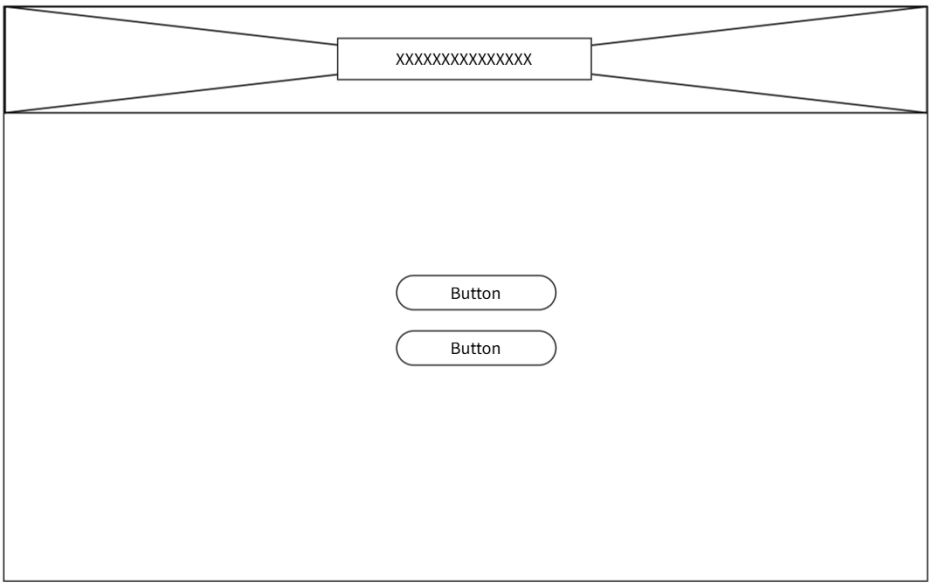


Figure 4.3 UI for main menu

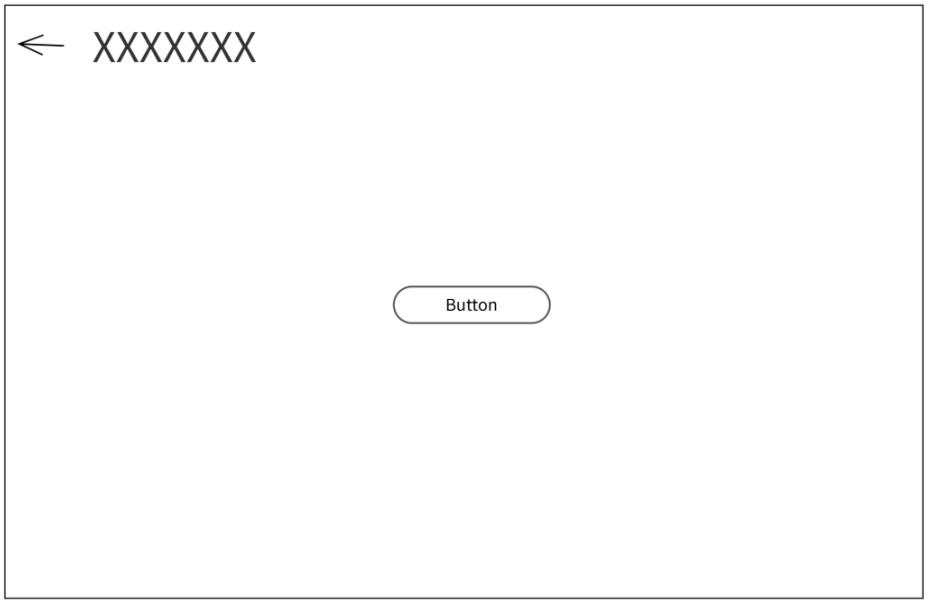


Figure 4.4 UI for PNG to MusicXML button

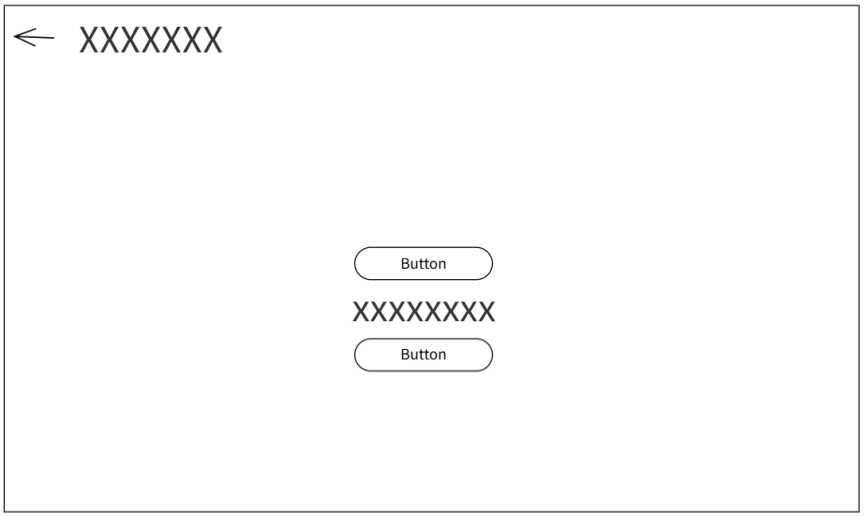


Figure 4.5 UI for uploaded result after uploaded PNG file

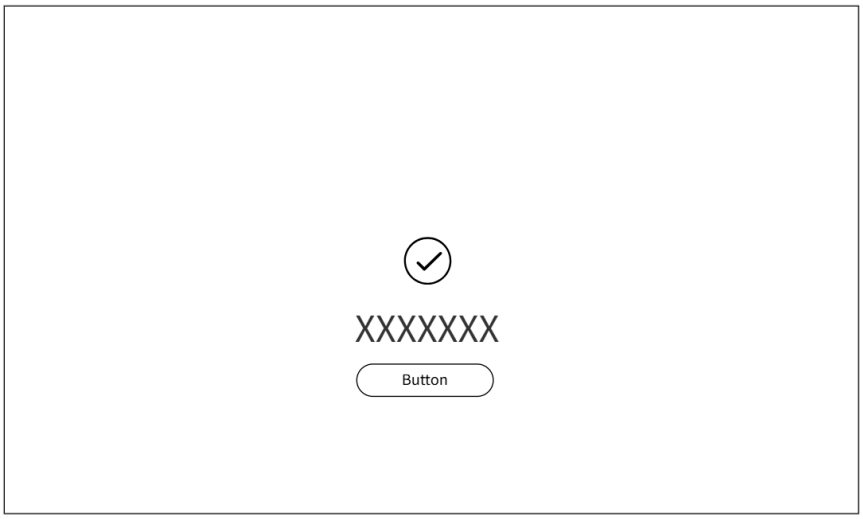


Figure 4.6 UI for successful page after PNG to MusicXML conversion

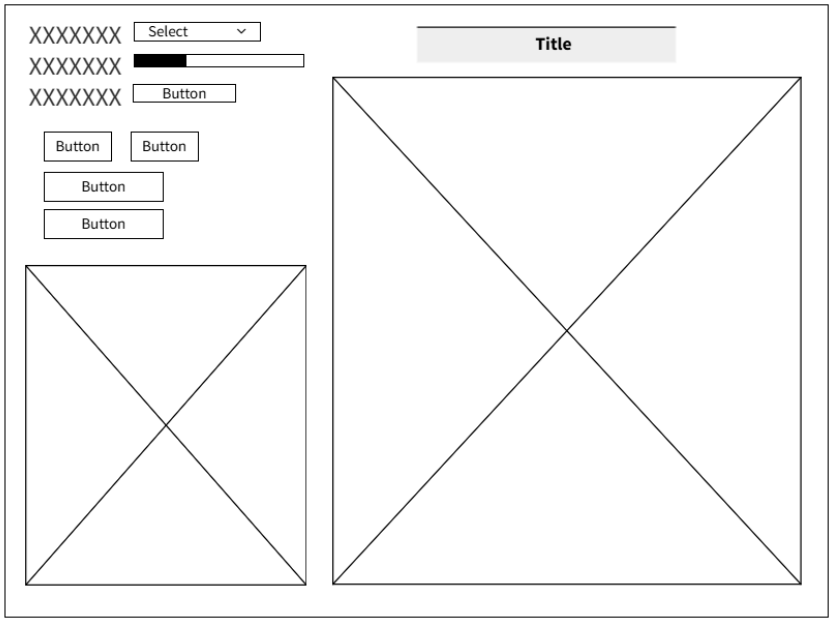


Figure 4.7 UI for MuiscXML to JianPu conversion

Chapter 5

System Implementation

The methodology for this project is phased development to develop the music note translation system, which is RAD based methodology. The system will be built using Dart, HTML, and CSS programming language and interact using Javascript. In this chapter, each of the methods and technologies stated will be further explained in detail.

5.1 System Methodology

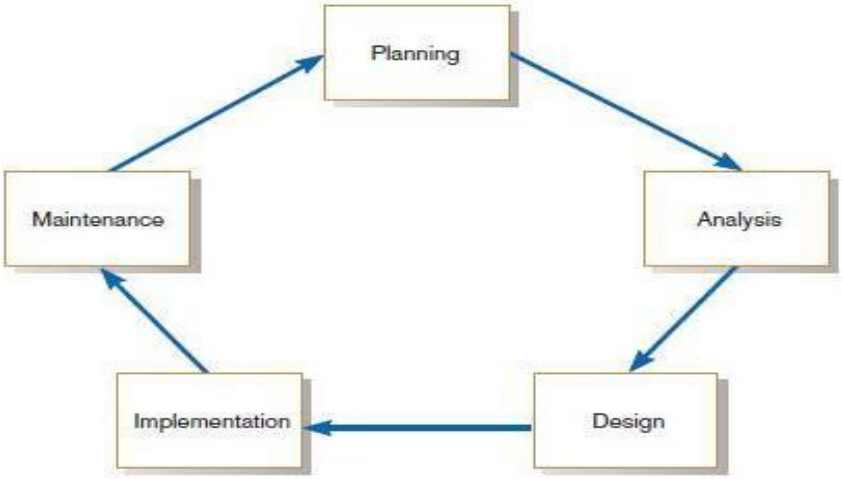


Figure 5.1 SDLC diagram

The Software Development Life Cycle (SDLC) is an organized methodology used to plan, create, test, and deliver reliable software effectively. It divides the entire software development process into 5 clear, manageable stages, which include planning, analysis, design, implementation, and maintenance. In this project, the phased development methodology is selected to build this music translation system. Phased development is a SDLC approach where the system is developed and delivered in incremental stages. This methodology allows us to build, test, and refine one part of the system before moving to the next. There are several reasons for choosing phased development as the methodology for this project. Firstly, the whole system can break down into small and manageable phases, where each phase is completed and continues to improve and add the functionality of system in next phases. Secondly, the system will be easier to test and fix errors since each phase has less functionality to be tested. Lastly, phased development is easier to showcase the incremental progress. Each completed phase will serve as a milestone, showcasing tangible outcomes and improvements over time.

5.2 Project Workflow in Phased Development Model

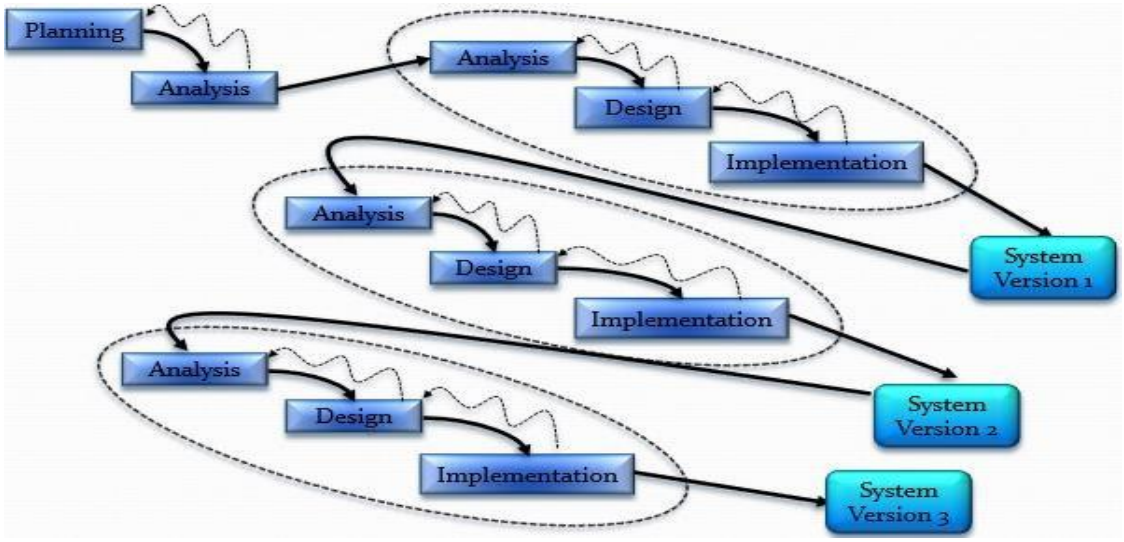


Figure 5.2 Phase Development Flow

1. Planning Phase

In planning phase, the overall project goals are defined, and the full system is divided into manageable phases. The critical features and functions, which is importing music sheet image file in this system, is developed first. Furthermore, the resources, timelines, and responsibilities for each phase are determined to ensure that the clear goals to be implemented within the timelines, and do not miss any functionality.

2. Analysis Phase

In analysis phase, the primary goal is to understand and define what the system must do in the current development stage. By achieve these primary goals, the analysis models is created to represent system processes and data relationship. For example, UML, use cases, and data flow diagram (DFD). The functional and non-functional requirements are also specified.

3. Design Phase

Once the requirements have been identified and documented in analysis phase, design phase begins. Design phase focuses on converting the gathered requirements into a clear blueprint that guides the development process for the current phase of the system. The design phase included logical, physical, high-level, architectural, interface, database and file, and program design.

4. Implementation Phase

In implementation phase, the functionalities defined will be implemented in coding based on the design that has defined in implementation phase. Since the system is divided into manageable parts, the coding can concentrate on delivering one set of features at a time. In this phase, the system will undergo testing to verify its correctness and dependability before moving on to the next stage.

5. Maintenance Phase

Finally, in the maintenance phase, the developed phase is reviewed, tested further for any bugs, and refined based on internal feedback. There are 4 types of maintenance which is corrective, adaptive, perfective and preventive maintenance. Adjustments may be made to enhance performance or fix minor issues, ensuring a stable foundation before the next functionality is added. This phase also allows to document outcomes and prepare for the next cycle of development.

5.3 Project Timeline

In Project 1, the tasks are arranged according to the report chapter by chapter, which included Introduction, Literature Review, Proposed Methodology, Implementation, Preliminary work, and Conclusion. While in Project 2, the tasks are arranged phase by phase, which is Planning, Analysis, Design, Implementation, and Testing. Both timelines for Project 1 and Project 2 are shown below:

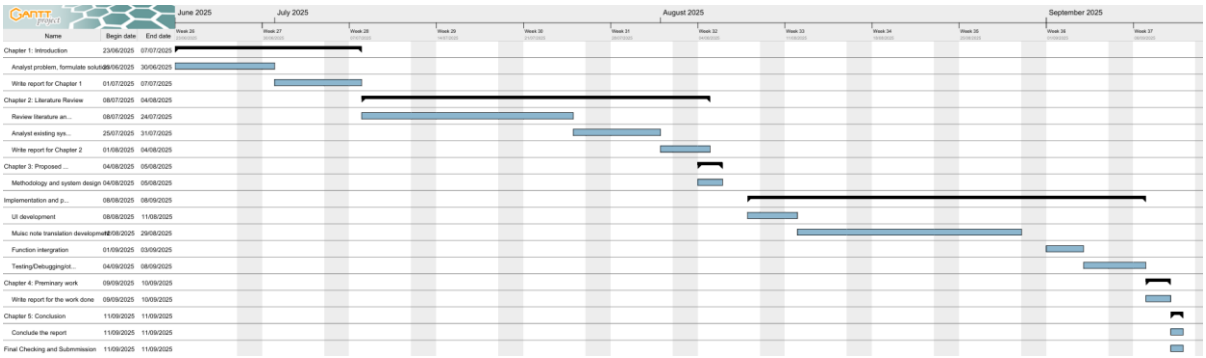
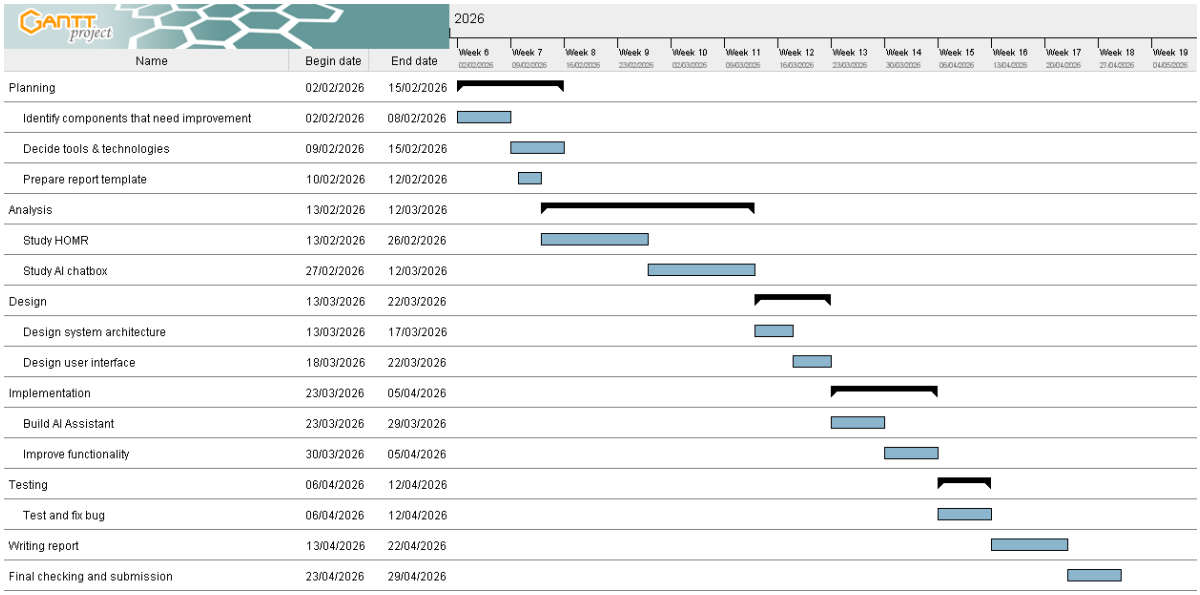


Figure 5.3 Timeline for Project 1



Name	Begin date	End date
Planning	02/02/2026	15/02/2026
Identify components that need improvement	02/02/2026	08/02/2026
Decide tools & technologies	09/02/2026	15/02/2026
Prepare report template	10/02/2026	12/02/2026
Analysis	13/02/2026	12/03/2026
Study HOMR	13/02/2026	26/02/2026
Study AI chatbox	27/02/2026	12/03/2026
Design	13/03/2026	22/03/2026
Design system architecture	13/03/2026	17/03/2026
Design user interface	18/03/2026	22/03/2026
Implementation	23/03/2026	05/04/2026
Build AI Assistant	23/03/2026	29/03/2026
Improve functionality	30/03/2026	05/04/2026
Testing	06/04/2026	12/04/2026
Test and fix bug	06/04/2026	12/04/2026
Writing report	13/04/2026	22/04/2026
Final checking and submission	23/04/2026	29/04/2026

Figure 5.4 Timeline for Project 2

5.4 Technologies and Tools Involved

This system will use Dart and JavaScript programming languages, along with HTML markup language and CSS stylesheet language. Dart is used to develop an executable file and handle the overall application logic. It is also responsible for converting images into MusicXML format. JavaScript is used to support the system's logic and operations, and to add interactivity and functionality to HTML components such as buttons. HTML is used to structure the web content and is also used to display Jianpu notation, as the Jianpu font style needs to be embedded within HTML. Other than JianPu font style, another reason is HTML having a lot of API that can be utilized to perform special function such as audio playback, export PDF, DOM parsing, and so on. CSS is used to style and enhance the HTML structure, making the website more user-friendly and visually comfortable to use.

Therefore, based on the advantages mentioned, by using the technologies mentioned, it would allow us to save time and develop a interactive system.

Below are the hardware and software used to develop this project:

1. Hardware:

- Laptop

Description	Specifications
Model	HUAWEI BOHB-WAX9
Processor	Intel Core i3-10110U
Operating System	Windows 11
Graphic	Intel UHD Graphics
Memory	8GB
Storage	238GB

Table 5.1 Laptop Specifications

2. Software:

- Visual Studio Code
- Flutter

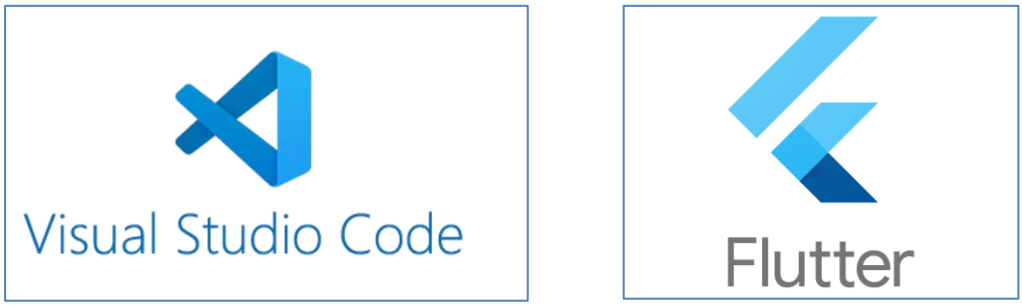


Figure 5.5 Software used

5.5 Implementation Outcome

This section will show the results of this system.

When user first come in the system, he will see the interface with 2 buttons, which is “Image to MusicXML” and “MusicXML to JianPu”. Users need to implement the first button first then only can implement the second button. The interface is shown below:

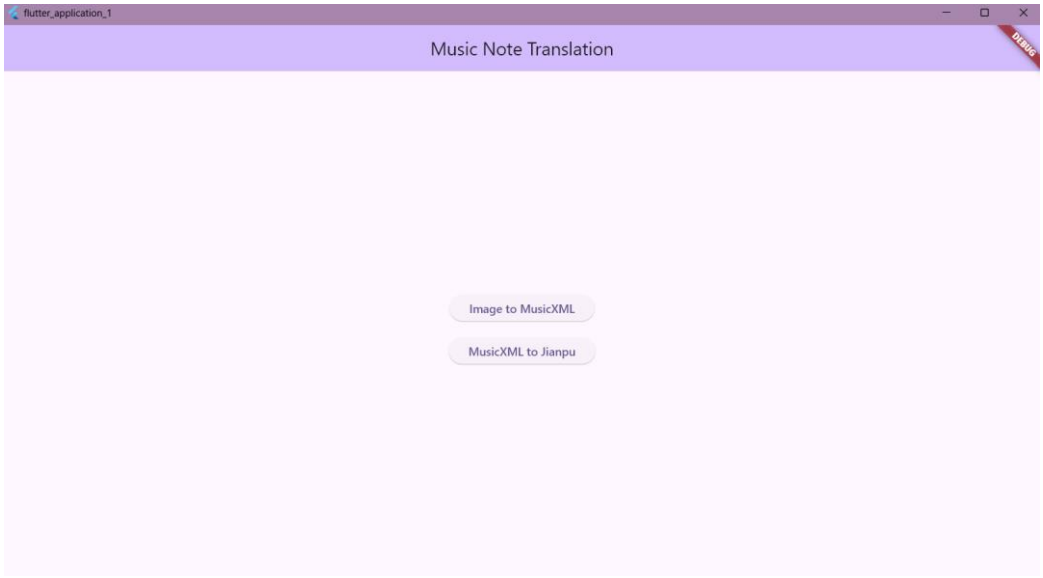


Figure 5.6 Interface of main menu

When users click the first button, which is Image to MusicXML, system will direct users to another page to let user click “Select Image File” button (Figure 5.7), and a file explorer will pop up to let user to choose music sheet (Figure 5.8).

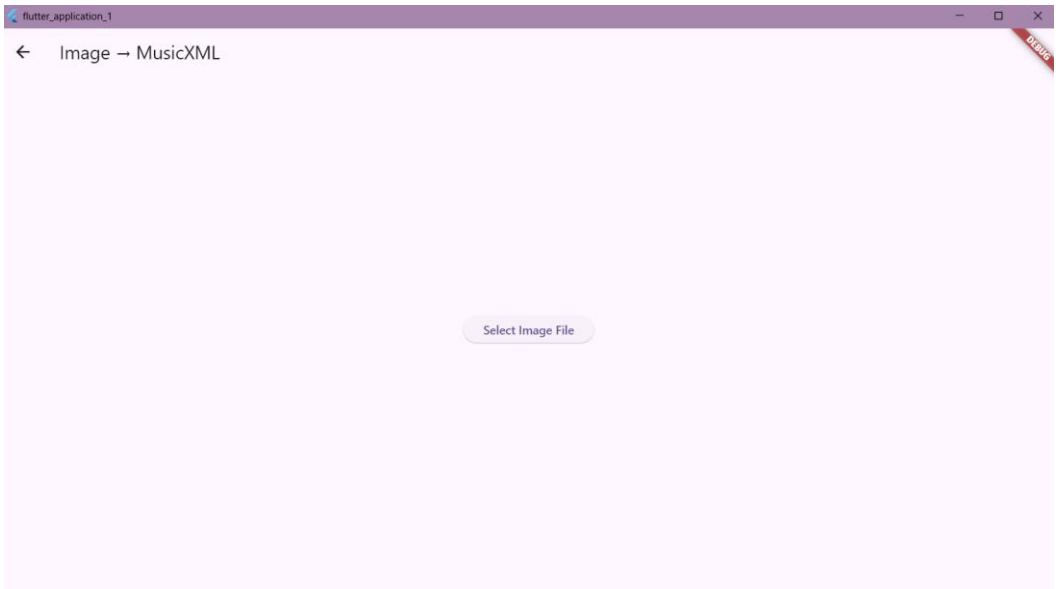


Figure 5.7 Interface of Image to MusicXML

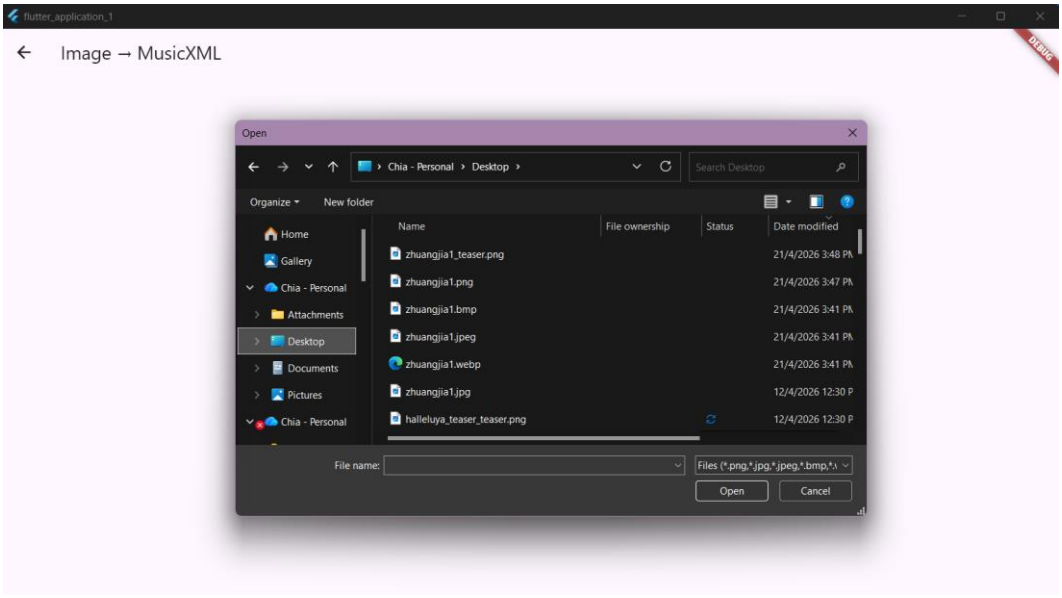


Figure 5.8 Interface of pop up file explorer

After users selected music sheet, it will show the selected file path and file name, at the same time, “Convert” button appears to let user click to convert music sheet into MusicXML file. (Figure 5.9)

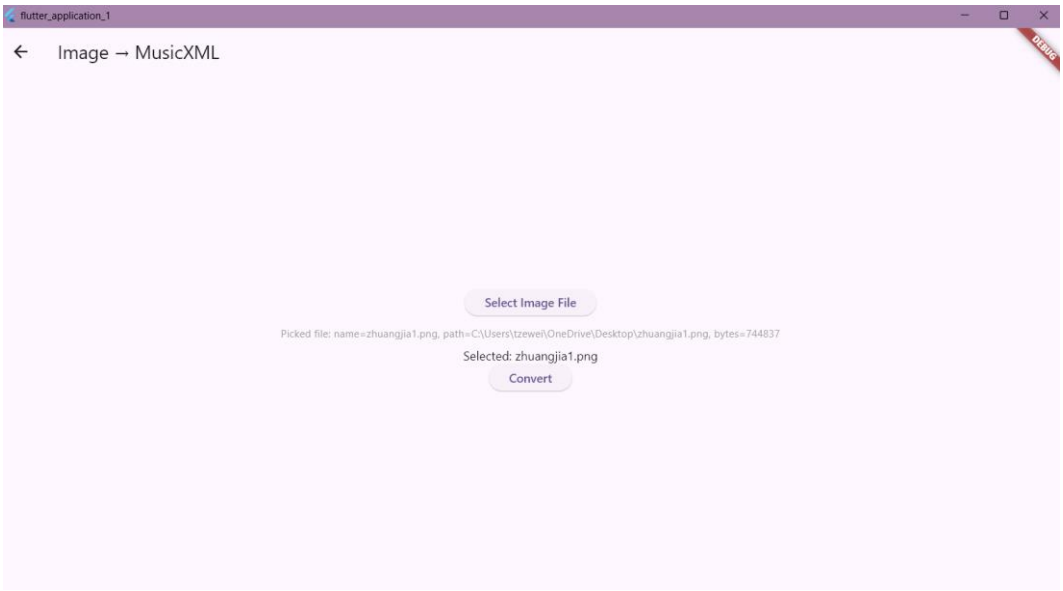


Figure 5.9 Interface of selected file

After click “Convert”, system will show a animation loading bar shown below:

Figure 5.10 Interface of loading page

When the conversion is successful, it will show user a successful message and state the location that the MusicXML file have saved, and users can click back to go back to main menu. Noted that the file path is clickable and it will direct to the file path.

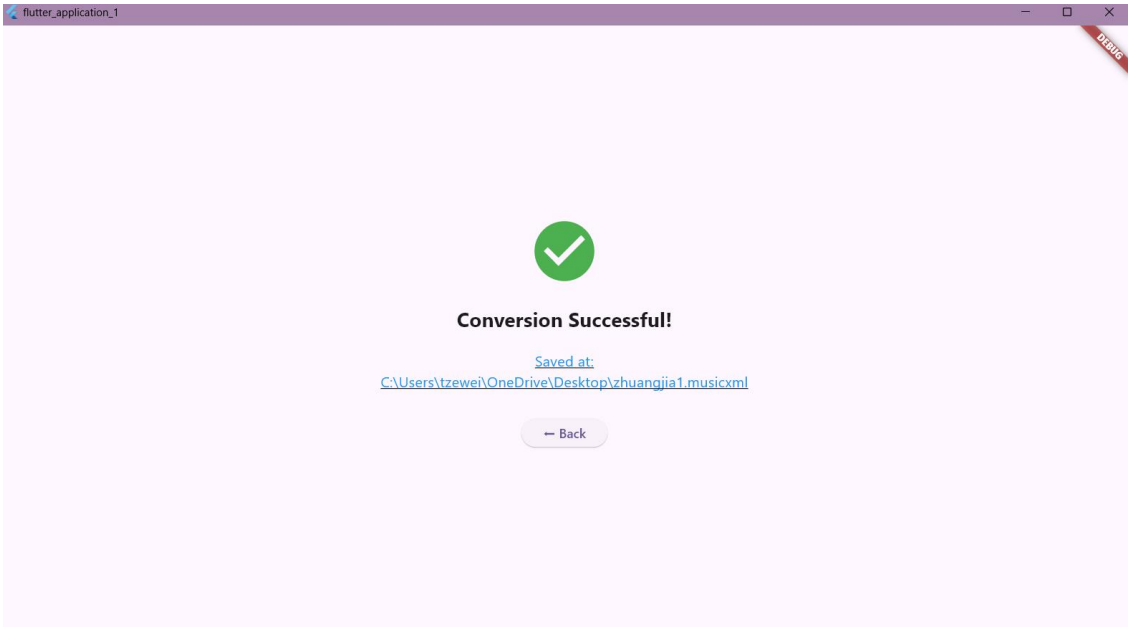


Figure 5.11 Interface of successful page

Go back to main menu, users can click on “MusicXML to JianPu” button and system will direct users to a website. (Figure 5.12)

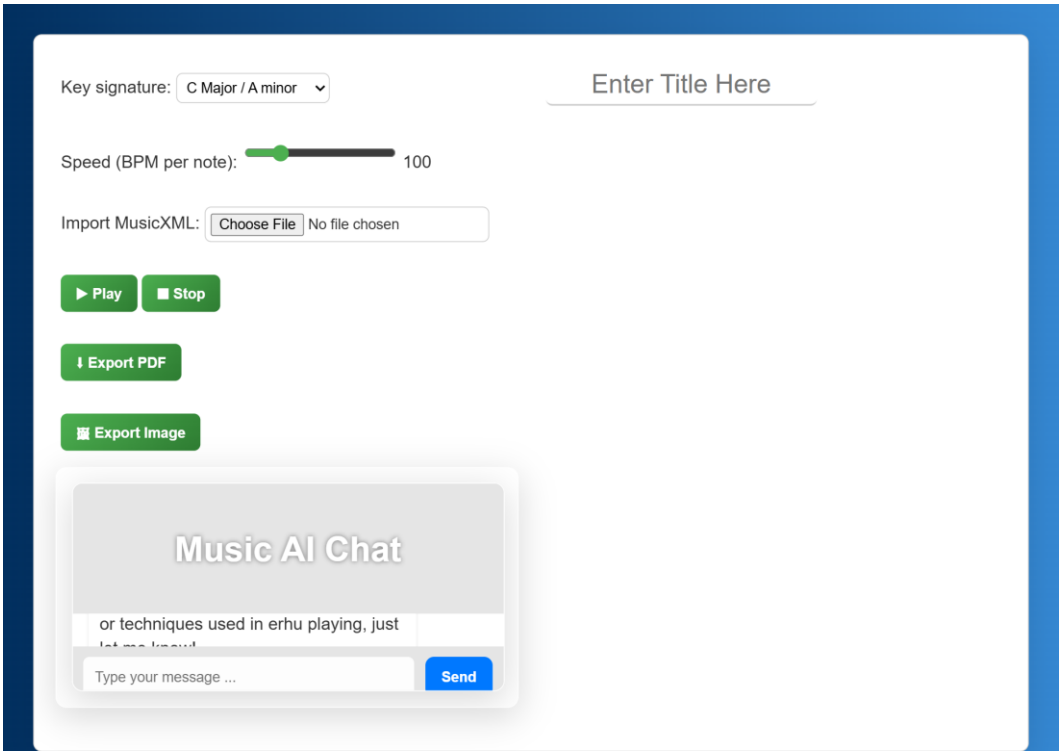


Figure 5.12 Interface of MusicXML to JianPu

Users need to choose MusicXML and the JianPu result will be shown on the screen.

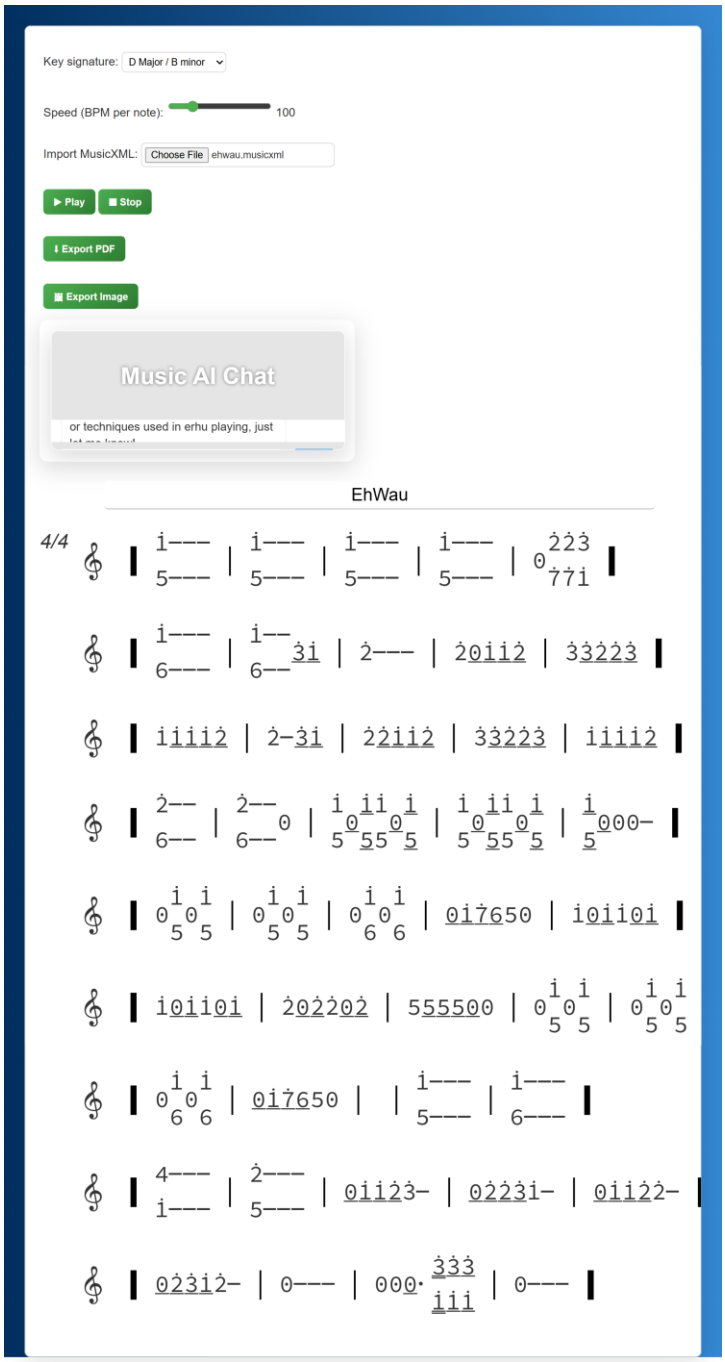


Figure 5.13 Interface of JianPu result

When user click on a note, it will be able to edit the note by showing the typing box. Below will list down the symbol with respective function:

1. note = 1234567
2. rest = 0
3. bar line = |
4. higher octave = '
5. lower octave = ,
6. sixteenth note = s
7. eighth note = q
8. eighth note = -
9. dotted note = .
10. big sharp = #
11. big flat = b
12. small sharp = ^
13. small flat = _

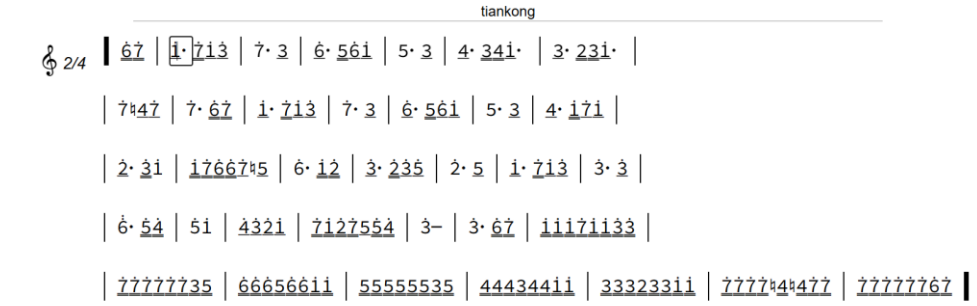


Figure 5.14 Before edit

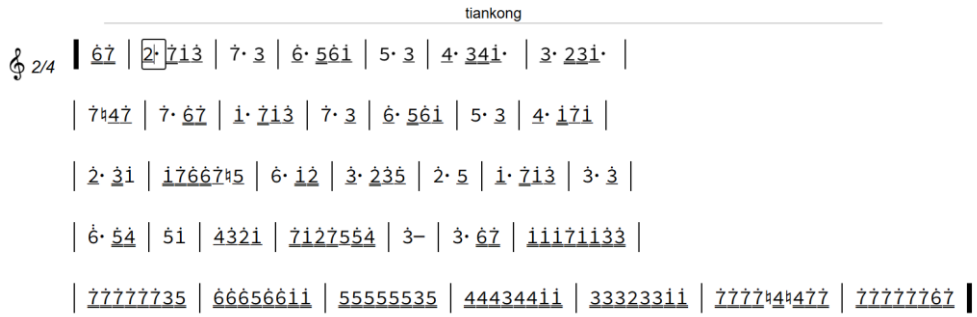


Figure 5.15 After edit

When users want to export to PDF or PNG, after click on button, the file will automatically download to user.

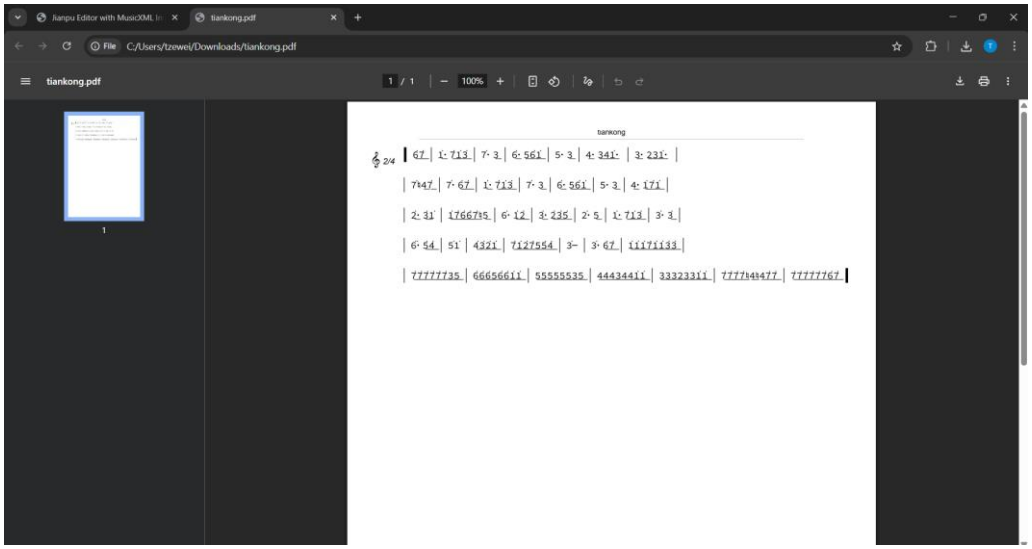


Figure 5.16 PDF form

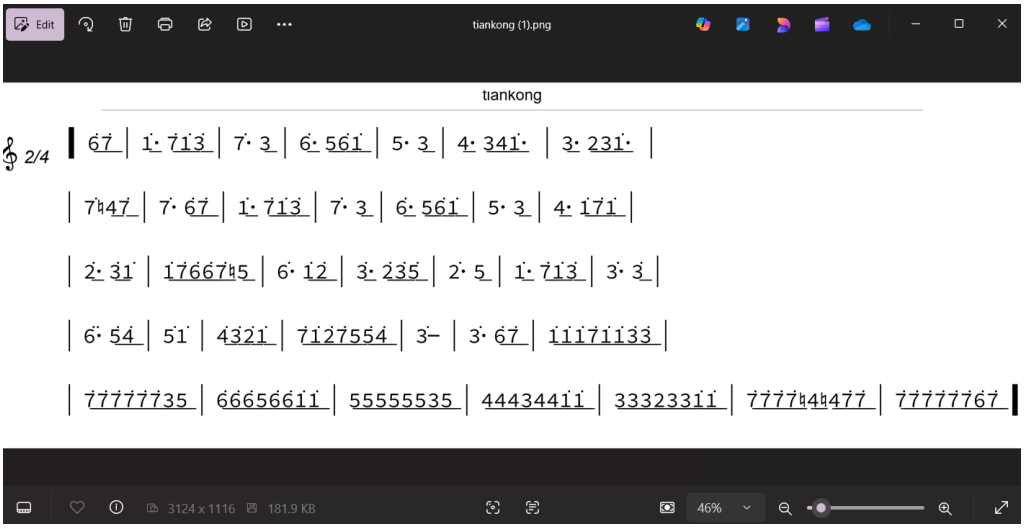


Figure 5.17 PNG form

Users also can ask AI Assistant about current uploaded music sheets. (key signature, tempo, title)

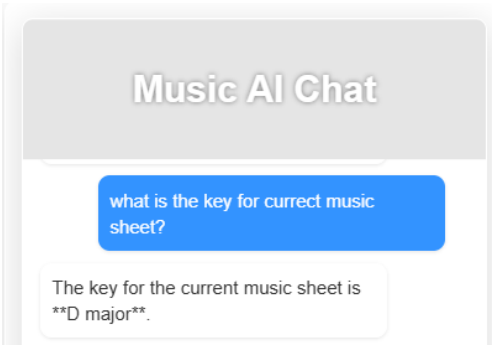


Figure 5.18 AI Assistant answer key signature

Also, can ask anything about music.

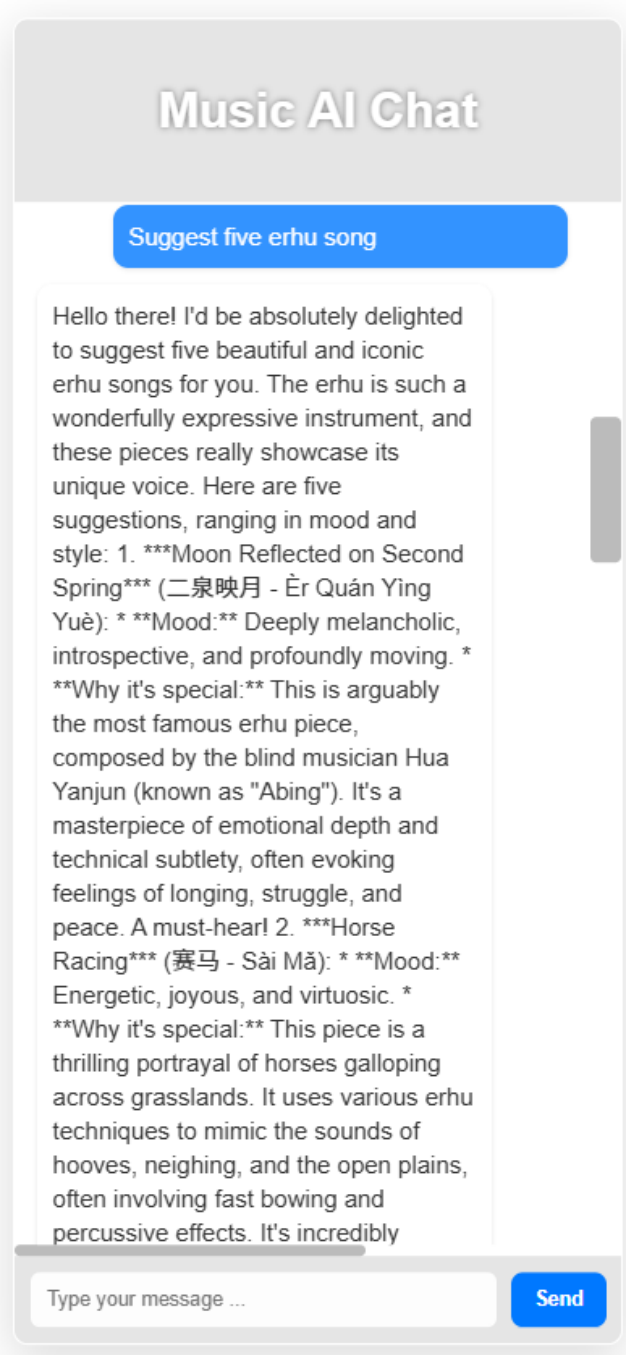


Figure 5.19 AI Assistant suggest erhu song

Chapter 6

System Evaluation and Discussion

This chapter will show the result from evaluation testing and discuss about the challenge while developing this system.

6.1 Testing Setup and Result

The system's performance is evaluated based on its functionality and accuracy. The system should be capable of converting an image of a music sheet into MusicXML format. In addition, it should accurately convert MusicXML into JianPu notation. Core functionalities such as music playback, tempo and key signature adjustment, file upload (image or MusicXML), and exporting files in PDF or PNG format should operate efficiently and reliably.

Furthermore, the JianPu editor should support proper symbol usage and provide guidance to assist users during editing. The AI assistant should be able to answer questions related to the uploaded file, including details such as the sheet title, tempo, and key signature. It should also be capable of responding to general music-related queries.

Below shows the verification plans for this system, which is import and export music sheet, MusicXML conversion, JianPu conversion, edit JianPu, music playback and AI Assistant.

1. Import Music Sheet

Procedure Number	P1
Method	Testing
Purpose/Scope	To make sure whether the import file format is able to loaded and recognize by system.
Special Conditions/Limitations	User may import the file format other than PNG.
Acceptance Criteria	The system is able to loaded and recognized the PNG and MusicXML file.
Prerequisites	None
Procedures	1. Pop out the file window to let user choose their file. 2. Click open button.
Troubleshooting	Refine the code, ensure the file format is limited correctly.

Table 6.1 Verification Plan P1 (Import Music Sheet)

2. MusicXML Conversion

Procedure Number	P2
Method	Testing
Purpose/Scope	To ensure the user can successfully translate PNG music sheet into MusicXML file.
Special Conditions/Limitations	User may import PNG file that does not contain music sheet.
Acceptance Criteria	The system recognize the music sheet and translate each music components into MusicXML format correctly.
Prerequisites	The imported PNG file needed to be able to loaded by the system.
Procedures	1. Open the system. 2. Click PNG to MusicXML button. 3. Select PNG file. a. If system are successfully recognize MusicXML file, proceed to step 4. b. Else if the file name is contain non English character or system cannot recognize music components in PNG file, show error message. 4. Observe the conversion of system is converted the correct MusicXML format file.
Troubleshooting	Refine the code, make sure that show the error message if the file unable to recognize music component and unable convert to MusicXML file.

Table 6.2 Verification Plan P2 (MusicXML Conversion)

3. JianPu Conversion

Procedure Number	P3
Method	Testing
Purpose/Scope	To make sure the user able to translate standard notes into numbered notes based on MusicXML file.
Special Conditions/Limitations	User may import MusicXML file that contain invalid structure of MusicXML.
Acceptance Criteria	The system translate each standard note into numbered note correctly.
Prerequisites	The imported PNG file needed to be recognize and converted correctly by the system.
Procedures	1. Open the system. 2. Click MusicXML to Jianpu button. 3. Select MusicXML file. a. If system are successfully recognize MusicXML file, proceed to step 4. b. Else pop-up error messages appear to alert users to provide a valid file format. 4. Observe the translation of system to translate the correct numbered note.
Troubleshooting	Refine the code, make sure that show the error message if the file unable to recognize music component and unable convert to MusicXML file.

Table 6.3 Verification Plan P3 (JianPu Conversion)

4. Edit JianPu

Procedure Number	P4
Method	Testing
Purpose/Scope	To make sure the user able to modify the numbered note by using keyboard to change the number.
Special Conditions/Limitations	User may type in the number out of range which outside range of 1 to 7, 7 symbols which is comma(,), dot(.), apostrophe(‘), hashtag(#), vertical bar(|), dash(-) and slash(/), and 2 character “b” and “n”.
Acceptance Criteria	The system enable user type in number from 1 to 7, 7 symbols and 2 character only.
Prerequisites	The imported MusicXML file needed to be translated successfully by the system.
Procedures	1. Click on the number that needed to modify. 2. Edit the number between 1 to 7 or symbol, symbols (, . ’ / | - #) and character (b, n). a. If the user’s input is not in range, do not showing the input on screen. b. Else if number is in range, replace new number type by user. 3. Observe whether the value is in the range.
Troubleshooting	Refine the code, make sure that the number, symbol or character is limited correctly.

Table 6.4 Verification Plan P3 (Edit JianPu)

5. Music Playback

Procedure Number	P5
Method	Testing
Purpose/Scope	To make sure the music sheet is able to playback according to jianpu.
Special Conditions/Limitations	User may type number, symbol and character in wrong sequences.
Acceptance Criteria	The system able to play the music according to jianpu symbol, character and number.
Prerequisites	The system needed to ensure the jianpu is editable and is translated correctly from MusicXML.
Procedures	1. Click playback music button 2. Observe whether music is being played.
Troubleshooting	Refine written code, make sure the music playback is according to jianpu.

Table 6.5 Verification Plan P4 (Music Playback)

6. Export Music Sheet

Procedure Number	P6
Method	Testing
Purpose/Scope	To make sure whether the system able to export PDF and PNG file format.
Special Conditions/Limitations	Users are able to export PDF and PNG files.
Acceptance Criteria	The system is able to successfully export JianPu music sheets in PDF and PNG formats.
Prerequisites	The system must ensure that MusicXML is loaded and displayed on the screen.
Procedures	1. Click “Export PDF” or “Export PNG” button. 2. The PDF or PNG file will downloaded into user’s device.
Troubleshooting	Refine the code and ensure the system is able to export files to the user’s device

Table 6.6 Verification Plan P6 (Export Music Sheet)

7. AI Assistant

Procedure Number	P7
Method	Testing
Purpose/Scope	To make sure AI Assistant able to answer user's questions.
Special Conditions/Limitations	For questions related to uploaded music sheets, users only can ask title, tempo and key signature. Otherwise, users can ask other questions related to music topic.
Acceptance Criteria	The system is able to answer user's question.
Prerequisites	System needs to ensure MusicXML is loaded into the system.
Procedures	1. Users type their questions in AI Assistant. 2. AI Assistant will return the result to users.
Troubleshooting	Refine the code, ensure user's question is able to answer.

Table 6.7 Verification Plan P7 (AI Assistant)

6.2 Project Challenges

The issues and challenges faced in developing this project included layout and positioning complexity in converting MusicXML to PNG. It is because MusicXML contain musical data in text form but not fixed visual position. Thus, we need to carefully decide the note spacing, staff alignment and page layout to make sure the JianPu music sheet able to show visually on user's screen. Next is complexity of musical expressions. Some advanced notation such as tremolo and triplet is not supported in JianpuASCII font style, the system may interpret it wrongly instead of ignoring it. It may show the wrong result to the users.

Another challenge includes technical limitations and time constraints. Developing an accurate translation system requires integration of multiple technologies such as optical music recognition (OMR), data parsing, and rendering into another musical representation. This demands both programming expertise and musical knowledge, and due to unfamiliarity with specialized music theory concepts in programming form, frequent external resources are required. This process can be time consuming and requires careful scheduling. Thus, effective time management and organized planning are crucial to achieving the project's goals on schedule.

6.3 Objectives Evaluation

The objective that list in Chapter 1 has been achieved by this developed system. From the result returned from system, it can be proved that the objective is reached successfully. Below is a table that lists out the objectives and their respective results:

Objective	Result
To create a simple and user-friendly system interface design and function for users.	This system shows only 2 buttons for MusicXML conversion and clean look and interactive interface for JianPu conversion.
To develop a cost-free and secure system that able to translate musical notes without restrictions.	It is an open-source system uploaded to GitHub to ensure code transparency. GitHub also automatically scans project dependencies for known vulnerabilities.
To integrate existing numbered music rules and functions into a system.	This system integrates two components, which are HOMR and JianpuASCII.ttf obtained from GitHub. The system combines these components into a single system for user use.

Chapter 7
Conclusion and Recommendation

The report will conclude in this chapter. Recommendations for future work and improvement will be mentioned in this chapter.

7.1 Conclusion

Music note translation system is used to convert standard musical notes into numbered musical note, which also called JianPu. The problem domain for this project is that there is no free and reliable applications available in the market that can fully translate standard musical notation to numbered notation, most translation tools not user friendly for music beginners, and existing online converters may be a untrusted websites or applications that expose users to security risk such as viruses or malware. The suggested approach to address these issues is to develop a system interface that is simple and easy to use and function for users, developing a cost-free and secure system that able to fully translate the musical note without restrictions, and integrate existing numbered music rules and functions into an system which are more secure and convenient for users. This project has 3 objectives, which is: (1) To create a simple and user-friendly system interface design and function for users. (2) To develop a cost-free and secure system that able to fully translate the musical notes without restrictions. (3) To integrate existing numbered music rules and functions into an system which are more secure and convenient for users.

In chapter two, a review of technique used in translation has been done, which technique is using MusicXML file to do translation. The coding within the MusixXML file has been explained. Besides that, a literature review has been done on three existing systems for music note translation, which are Musescore3, Fanqie Jianpu by using command prompt, and EOP NMN Master. Each system has its own functionalities and limitations, which could be resolved by integrating the functionalities from this existing system.

In chapter three, system architecture diagram shows the system flow between user and the system. Additionally, UML diagrams, such as the use case diagram and the activity diagrams for each use case are also presented. The sole actor of the system is user, and the main

function include MusicXML conversion, JianPu conversion, playback music, edit JianPu, and AI assistant.

In chapter four, the system flow diagram shows the detailed interaction of MusicXML conversion and JianPu conversion within the system. For MusicXML conversion, it use HOMR algorithm to convert image into MusicXML. For JianPu conversion, system use JianPu token and organize them into a structure layout. The font style is using JianpuASCII. The system design is also shown in this chapter to show the user interface framework for this system. System specification and user requirements also list in this chapter.

In chapter five, methodology is specified, which is RAD based methodology of phased development, which include planning, analysis, design, implementation, and maintenance phase. Hardware used for this system is a Huawei laptop, and software used to implement coding is Visual Studio Code and Flutter. Programming language used is Dart and JavaScript, HTML as markup language, and CSS as stylesheet language. The step by step implementation outcome has also been screenshot and shown in this chapter.

In chapter six, testing setup and result is done to make sure the system performs well without bug. Verification plan is developed to verify the functionality of the system. The testing is done in import and export, edit JianPu, MusicXML conversion, JianPu conversion, music playback, and AI assistant. The challenge for this project is include layout and positioning issues, technical limitations and time constraints.

In conclusion, the novelty of this project is to produce a music translation system that integrates the function from existing systems using Dart, HTML, CSS, and JavaScript. The expected outcome is a user-friendly system that simplifies the process of translating musical notation. This system will enable the user to translate the standard musical note into numbered musical notes based on the uploaded music sheet file. Additionally, this system provides playback functionality, enabling users to listen to the music directly from the sheet, and offers editing capabilities for both standard and numbered notations. Furthermore, this system enables users to import and export in form of text PDF, PNG and MusicXML.

7.2 Recommendation

The recommendation that could be made to make the whole system better is added the advanced music symbol to JianPu. Currently the system is not able to convert advanced music symbols like slur line or tremble symbols.

REFERENCES

[1]
"Welcome To Zscaler Directory Authentication," *Freedownloadmanager.org*, 2025. https://download.freedownloadmanager.org/Windows-PC/EOP-NMN

[2]
lzh9102, "GitHub - lzh9102/musicxml_to_jianpu: Create JianPu notation from MusicXML," *GitHub*, 2018. https://github.com/lzh9102/musicxml_to_jianpu

[3]
ssb22, "GitHub - ssb22/jianpu-ly: Jianpu in Lilypond," *GitHub*, 2017. https://github.com/ssb22/jianpu-ly/tree/master

[4]
nybbs2003, "GitHub - nybbs2003/lilypond-Jianpu: Display plugin of Jianpu notation for Lilypond," *GitHub*, 2017. https://github.com/nybbs2003/lilypond-Jianpu/tree/master

[5]
"Audiveris - Open-source Optical Music Recognition," *GitHub*, Aug. 19, 2023. https://github.com/Audiveris/audiveris

[6]
Trumanchho, "GitHub - Trumanchho/omr-jianpu-melody-generator: Training a TensorFlow OMR model to process Jianpu musical notation and produce a MIDI output.," *GitHub*, 2024. https://github.com/Trumanchho/omr-jianpu-melody-generator

[7]
R. Fong, "Numbered Notation App - Rex Fong - Medium," *Medium*, Apr. 17, 2020. https://rexfng.medium.com/numbered-notation-app-bd98c7e099b

[8]
flans39, “JianpuPlugin/JianpuPlugin.qml at main · flans39/JianpuPlugin,” *GitHub*, 2021. https://github.com/flans39/JianpuPlugin/blob/main/JianpuPlugin.qml

[9]
“免费高效的制谱软件_番茄简谱,” *Jianpu99.net*, 2015. http://www.jianpu99.net/

[10]
Sepehr Keyhani, “Sheet music | Musescore,” *Musescore.com*, May 15, 2012. https://musescore.com/

[11]
“LilyPond 中如何输出简谱_lilypond 简谱-CSDN 博客,” *Csdn.net*, 2023. https://blog.csdn.net/qq_36822973/article/details/130724932

[12]
“LilyPond – Music notation for everyone: Manuals,” *Lilypond.org*, 2025. https://lilypond.org/doc/v2.23/Documentation/web/manuals.html

[13]
“PDFtoMusic,” *Myriad-online.com*, 2025. https://myriad-online.com/en/products/pdftomusic.htm

[14]
“曲谱转换 - 在线免费一键将 MIDI 转为五线谱与简谱谱子导出下载应用,” *Serocs.cn*, 2022. https://www.serocs.cn/conversion/

[15]
“Popular MIDIs,” BitMidi. https://bitmidi.com/

[16]
“How do you turn musical notes into numbers?,” *Quora*, 2019. https://www.quora.com/How-do-you-turn-musical-notes-into-numbers

[17]
"Add Jianpu numbers as text (lyrics): 五线谱->简谱," *MuseScore.org*, 2025. https://musescore.org/en/project/add-jianpu-numbers-text-lyrics-wuxianpu-jianpu

[18]
"The element | MusicXML 4.0," W3.org, 2025. https://www.w3.org/2021/06/musicxml40/musicxml-reference/elements/score partwise/ (accessed May 05, 2025).

[19]
Yoyo, "BreezeWhite/oemer," *GitHub*, Apr. 19, 2024. https://github.com/BreezeWhite/oemer

[20]
NetEase, "GitHub - NetEase/Polyphonic-TrOMR: TrOMR:Transformer-based Polyphonic Optical Music Recognition," *GitHub*, 2022. https://github.com/NetEase/Polyphonic-TrOMR (accessed Sep. 11, 2025).

[21]
Liebharc. (n.d.). GitHub - liebharc/homr: homr is an Optical Music Recognition (OMR) software designed to transform camera pictures of sheet music into machine-readable MusicXML format. GitHub. https://github.com/liebharc/homr

[22]
RobertWinslow. (n.d.). *GitHub - RobertWinslow/jianpu-ascii-font: A jianpu 简谱 font which uses ascii ligatures to display simplified numerical musical notation. Works basically everywhere.* GitHub. https://github.com/RobertWinslow/jianpu-ascii-font

[23]
AudioContext - Web APIs | MDN. (2025, October 30). https://developer.mozilla.org/en-US/docs/Web/API/AudioContext

[24]
OscillatorNode - Web APIs | MDN. (2025, August 27). https://developer.mozilla.org/en-US/docs/Web/API/OscillatorNode

[25]
GainNode - Web APIs | MDN. (2023, April 6). https://developer.mozilla.org/en-US/docs/Web/API/GainNode

[26]
FileReader - Web APIs | MDN. (2025, June 23). https://developer.mozilla.org/en-US/docs/Web/API/FileReader

[27]
Turner, N. (2026, February 19). *html2canvas – Advanced Guide for Modern Web Developers*. https://html2canvas.org/html2canvas-advanced-guide-for-modern-web-developers/

[28]
Turner, N. (2026b, February 19). *html2canvas – Advanced Guide for Modern Web Developers*. https://html2canvas.org/html2canvas-advanced-guide-for-modern-web-developers/

[29]
Web Storage API - Web APIs | MDN. (2025, February 22). https://developer.mozilla.org/en-US/docs/Web/API/Web_Storage_API

POSTER

UTAR
UNIVERSITI TUNKU ABDUL RAHMAN

Faculty of Information and Communication Technology

Translation of Standard Musical Notation to Numbered Musical Notation Mobile Apps

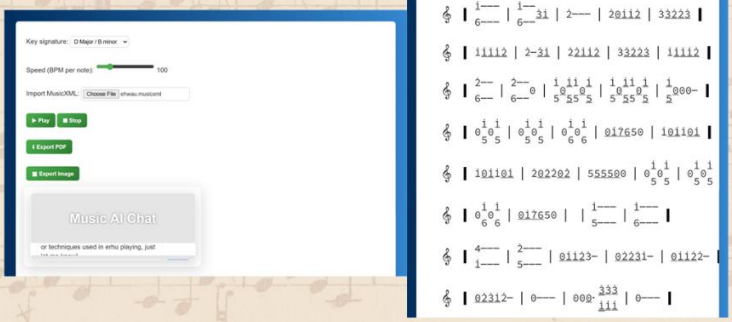
Introduction

This project is about to develop an system that able to translate from standard music note into numbered music note, which also called Jianpu, and also able to playback music according to Jianpu's number and symbol. Jianpu translation is using JavaScript to implement the operation in translating each note.

Methodology

In this project, Homer's Optical Music Recognition (HOMR) from GitHub was utilized to process images, recognize musical components, and convert them into MusicXML format. For Jianpu translation, JavaScript has used to parse MusicXML notes, map pitches and durations into Jianpu notation. To enable it, an external browser was launched to provide the necessary functionality. AI assistant is also developed to answer questions about music topic.

Result



The result will automatically filled for the title of song, clef, time signature, key signature, and speed.

Discussion

The operation in translating music sheet to Jianpu is very simple, user friendly for people who starting learning music. The deliverable for this project is a system that used to translate music sheet into Jianpu.

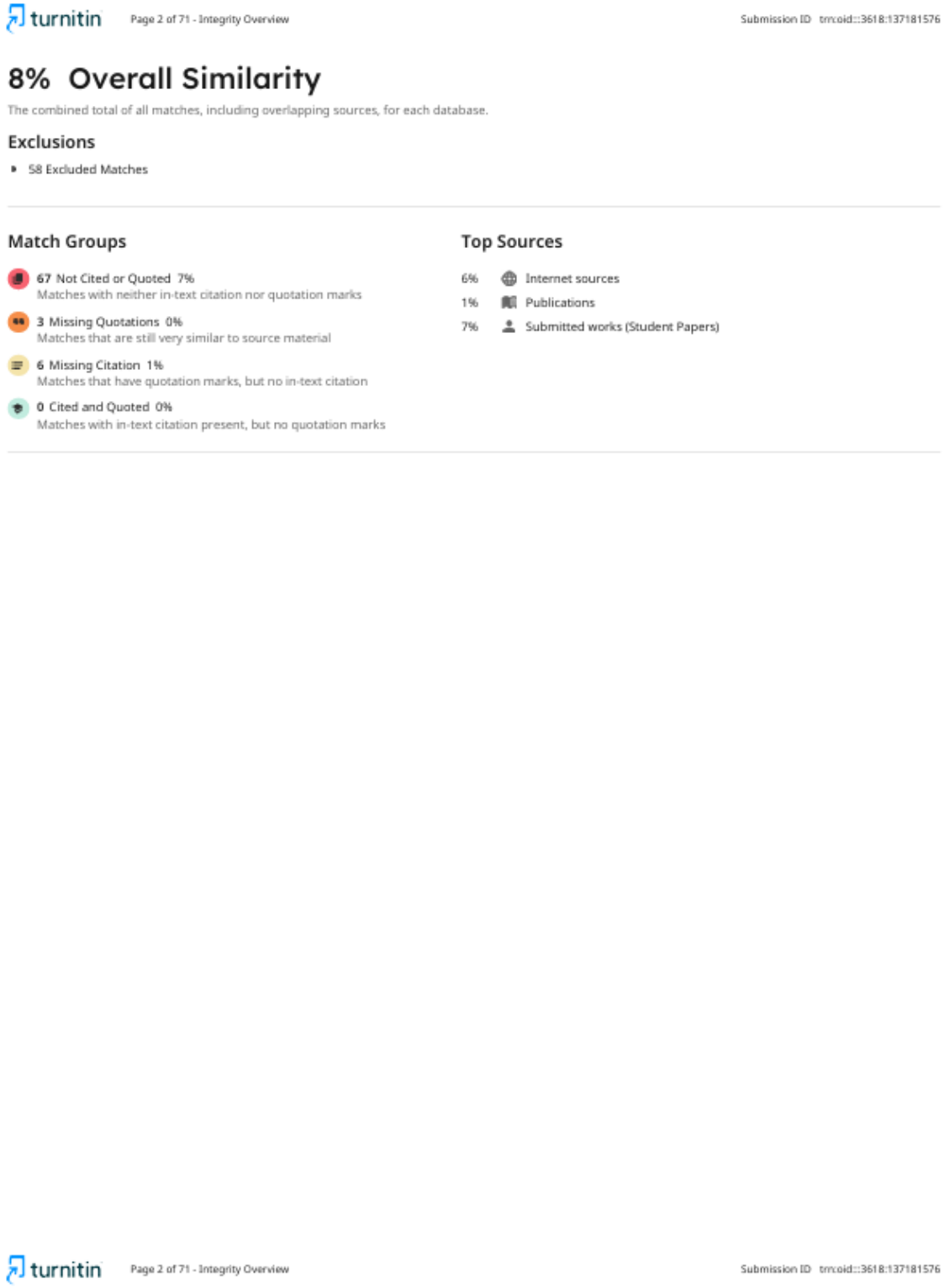
Conclusion

The system is able to translate from music sheet in image file to Jianpu and able to export it in PNG or PDF form. The Jianpu is editable, playback music speed, tempo, and key signature is able to adjust according to user's preferences. AI assistance can help users to gain knowledge about music topic.

Bachelor of Computer Science (Hons)
Final Year Project

Project by: Chia Tze Wei
Project Supervisor: Ts Soong Hoong Cheng

PLAGIARISM CHECK RESULT



turnitin Page 2 of 71 - Integrity Overview Submission ID trn:oid:::3618:137181576

8% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Exclusions

- 58 Excluded Matches

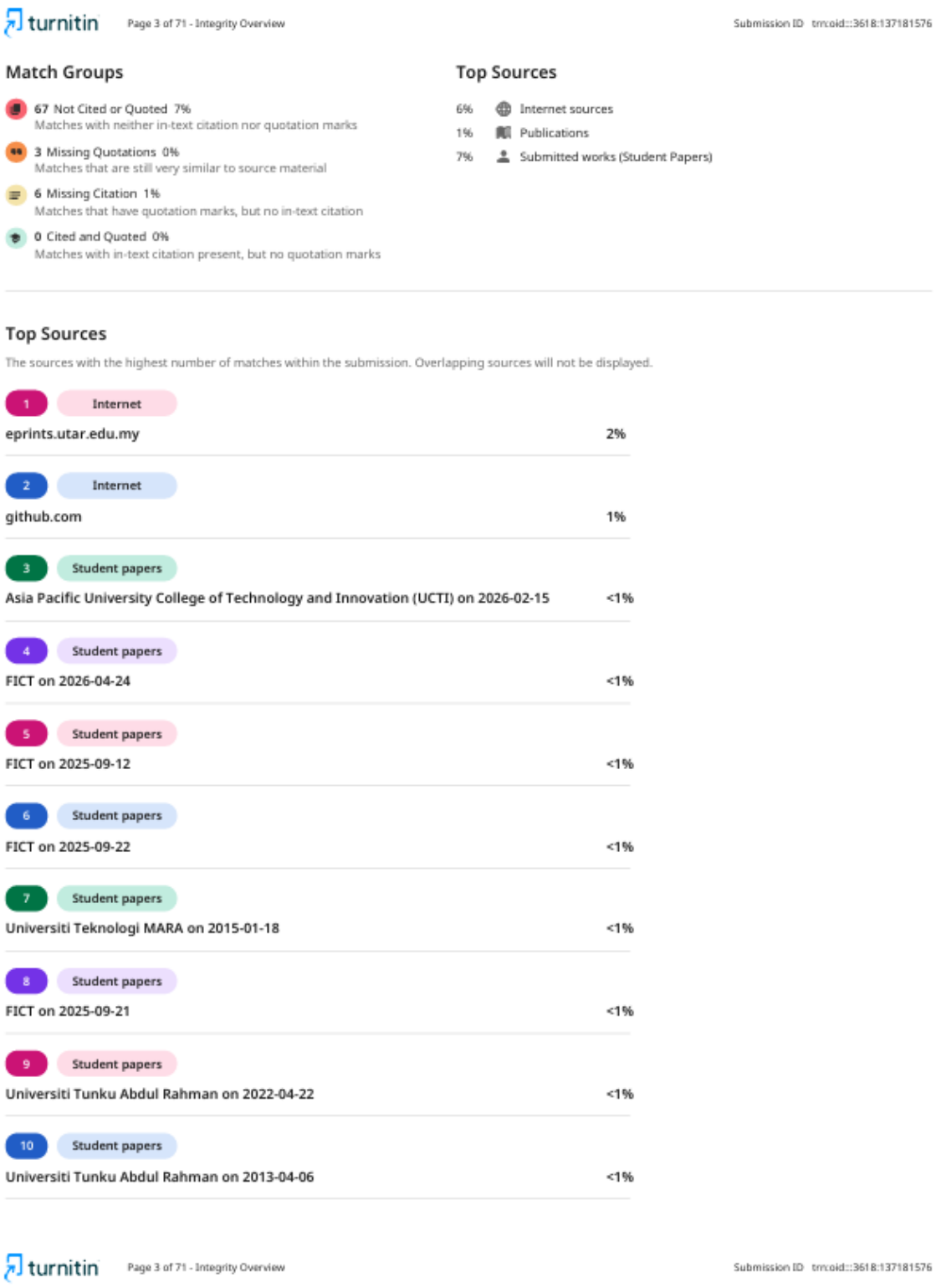
Match Groups

- 67 Not Cited or Quoted 7%
  Matches with neither in-text citation nor quotation marks
- 3 Missing Quotations 0%
  Matches that are still very similar to source material
- 6 Missing Citation 1%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

6% Internet sources
1% Publications
7% Submitted works (Student Papers)

turnitin Page 2 of 71 - Integrity Overview Submission ID trn:oid:::3618:137181576


turnitin
Page 3 of 71 - Integrity Overview
Submission ID trn:oid:::3618:137181576
Match Groups
67 Not Cited or Quoted 7%
Matches with neither in-text citation nor quotation marks
3 Missing Quotations 0%
Matches that are still very similar to source material
6 Missing Citation 1%
Matches that have quotation marks, but no in-text citation
0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks
Top Sources
6% Internet sources
1% Publications
7% Submitted works (Student Papers)
Top Sources
The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.
1 Internet
eprints.utar.edu.my 2%
2 Internet
github.com 1%
3 Student papers
Asia Pacific University College of Technology and Innovation (UCTI) on 2026-02-15 <1%
4 Student papers
FICT on 2026-04-24 <1%
5 Student papers
FICT on 2025-09-12 <1%
6 Student papers
FICT on 2025-09-22 <1%
7 Student papers
Universiti Teknologi MARA on 2015-01-18 <1%
8 Student papers
FICT on 2025-09-21 <1%
9 Student papers
Universiti Tunku Abdul Rahman on 2022-04-22 <1%
10 Student papers
Universiti Tunku Abdul Rahman on 2013-04-06 <1%
turnitin
Page 3 of 71 - Integrity Overview
Submission ID trn:oid:::3618:137181576

11	Student papers	
Universiti Tunku Abdul Rahman on 2022-09-09		<1%
12	Student papers	
University of York on 2022-08-31		<1%
13	Publication	
de Sousa, Leonor Martins. "Natural Interaction in Smartphone Virtual Reality Exp...		<1%
14	Student papers	
FICT on 2025-09-21		<1%
15	Student papers	
Universiti Tunku Abdul Rahman on 2022-12-06		<1%
16	Student papers	
Columbia University on 2024-12-10		<1%
17	Student papers	
University of Cape Town on 2024-04-05		<1%
18	Student papers	
FICT on 2025-09-20		<1%
19	Student papers	
University of Queensland on 2024-10-31		<1%
20	Internet	
musescore.org		<1%
21	Student papers	
Liberty University on 2025-06-04		<1%
22	Student papers	
Universiti Malaysia Pahang on 2012-05-20		<1%
23	Student papers	
Universiti Tunku Abdul Rahman on 2023-04-27		<1%
24	Student papers	
FICT on 2025-08-14		<1%

25 Student papers
Queen Mary and Westfield College on 2022-11-05 <1%

26 Student papers
Arab Open University on 2025-05-01 <1%

27 Student papers
Islington College,Nepal on 2026-04-21 <1%

28 Student papers
Oxford Brookes University on 2014-04-25 <1%

29 Student papers
Universiti Tunku Abdul Rahman on 2013-03-11 <1%

30 Student papers
Universiti Tunku Abdul Rahman on 2022-04-18 <1%

31 Student papers
University of Leeds on 2011-12-14 <1%

32 Student papers
University of Technology, Sydney on 2006-05-05 <1%

33 Student papers
Higher Education Commission Pakistan on 2012-04-24 <1%

34 Student papers
FICT on 2025-09-21 <1%

35 Student papers
FICT on 2025-09-22 <1%

36 Student papers
Universiti Putra Malaysia on 2011-05-13 <1%

37 Student papers
Universiti Tunku Abdul Rahman on 2022-09-08 <1%

38 Student papers
University of Sydney on 2024-08-30 <1%

39 Publication

Yixuan Li, Huaping Liu, Qiang Jin, Miaomiao Cai, Peng Li. "TrOMR:Transformer-Ba... <1%

40 Student papers

FICT on 2025-09-19 <1%

41 Student papers

Universiti Teknologi Malaysia on 2022-01-21 <1%

turnitin Page 6 of 71 - Integrity Overview Submission ID trn:oid:::3618:137181576

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 REPORT SUBMISSION

Student ID	22ACB02694
Student Name	CHIA TZE WEI
Supervisor Name	Ts. Soong Hoong Cheng

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Copyright Statement
√	Acknowledgement
√	Abstract (with Area of Study and Keywords)
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
x	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	References
√	All references are cited in the report, especially in the chapter of literature review
√	Appendices (if applicable)
√	Poster
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

I, the author, have checked and confirmed all the items listed in the above table are included in my report.

_____

(Signature of Student)

Date: 3 May 2026