User Manual

February 28, 2021

Tabs2XML

Alp Sirek
Andrew Ngov
Arjit Johar
Daniel Santorelli
Muhammad Azizi

Submitted in Partial Fulfillment of The Midterm of EECS 2311 Software Development Project

Table of Contents (Click to go to section)

1.0 Introduction

1.1 Purpose

Tabs2XML is being developed for the purpose of converting guitar tablature files and drums tablature files into MusicXML files. Due to the relatively new format, there aren't many music pieces written in MusicXML. While tablature for guitars and drums are easy to understand, they offer a low degree of readability and modification. The MusicXML format builds on these shortcomings to allow readers to better understand the music piece and easily play it. Tab2xml is being developed for those who want to play songs in the format of MusicXML, but find that they can only find the tablature of those pieces (common occurrence as there aren't as many pieces of song in the MusicXML format). Tabs2XML is also being developed for those that would like to play their songs in different keys. Tabs2XML will have implemented features to allow the user to change the key of their songs (given that the song being played is locally stored and that it is a file of type tablature or MusicXML).

1.2 Disclaimers for Midterm Version

There are some limitations of the current Midterm Submission version of Tabs2XML. These include:

- Tabs2XML currently only accepts conversions for guitar tab files.
- Tabs2XML currently only work with tablature in the format given under the project section of the course wiki page (look at acceptedFormat.txt).

2.0 Technical Specifications

To use Tabs2XML, the following criteria must be met:

- The device must be running the most recent version of Windows 10.
- The most recent version of Python must be installed.
- While any editor that supports Python will be sufficient, we highly recommend installing VS Code. We also recommend that the user install the Python extensions pack for VS Code, which can be found here: https://marketplace.visualstudio.com/items?itemName=donjayamanne.python-extension-pack

3.0 Installation Instructions

Once you have ensured your device meets the requirements specified in the "Technical Specifications" section, install Tabs2XML by following these instructions (these instructions assume you are using the recommended software, VS Code).

The user of Tabs2XML is required to be operating on a system that has the following programs/packages/extensions:

- Python (latest).
- VS Code (latest). While many editer/IDEs will work, we suggest VS Code as the the setup process has been tested with it and everything works.

- pip (Package Installer for Python).
- PySimpleGUI.
- lxml.
- numpy.

To get Python:

- Go to: https://www.python.org/downloads/.
- Download the latest version for the platform the current operating system is utilizing.
- Run the setup and configure necessary settings for the current system. *Ensure that the "Add Python to Path" option is selected on the initial Python installation screen*.
- In the event that there are any troubles along the way, please refer to: https://www.python.org/community-landing/.

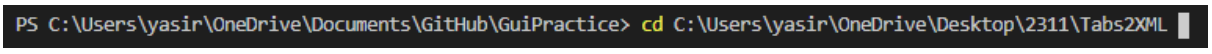
to get VS Code:

- Go to: https://code.visualstudio.com/download.
- Download the latest version for the platform the current operating system is utilizing.
- Run the setup and configure necessary settings for the current system.
- In the event that there are any troubles along the way, please refer to: https://www.python.org/community-landing/.
- To best optimize the system that will be running the program, please install the following VS Code Python extension pack: https://marketplace.visualstudio.com/items?itemName=donjayamanne.python-extension-pack.

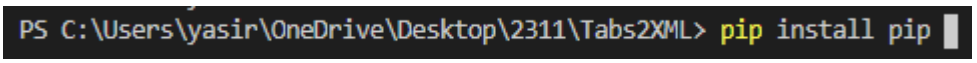
To get the packages for Python:

For the following commands, ensure that Windows PowerShell (Windows) or Terminal (MacOS/Linux) has the directory change so that it is in the same folder as the *non-zipped project folder*.

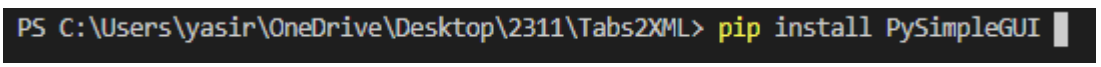
- To do so, Open Windows PowerShell (Windows) or Terminal (MacOS/Linux).
- Run the command: cd <THE PATH TO THE PROJECT FOLDER>
  - Refer to the image below for help. In the example below, the non-zipped project file was placed in folder “2311” on the Desktop.



- First, pip (Package Installer for Python) will have to be installed
  - Open Windows PowerShell (Windows) or Terminal (MacOS/Linux).
  - run the command: pip install pip.



  - Note the working directory in the above example!
  - In the even there are any issues, please refer to: https://pypi.org/project/pip/#files
- To get PySimpleGUI:
  - Open Windows PowerShell (Windows) or Terminal (MacOS/Linux).
  - run the command: pip install PySimpleGUI.



  - Note the working directory in the above example!

- In the even there are any issues, please refer to: https://pypi.org/project/PySimpleGUI/
- To get lxml:
    - Open Windows PowerShell (Windows) or Terminal (MacOS/Linux).
    - run the command: pip install lxml.

```
PS C:\Users\yasir\OneDrive\Desktop\2311\Tabs2XML> pip install lxml
```

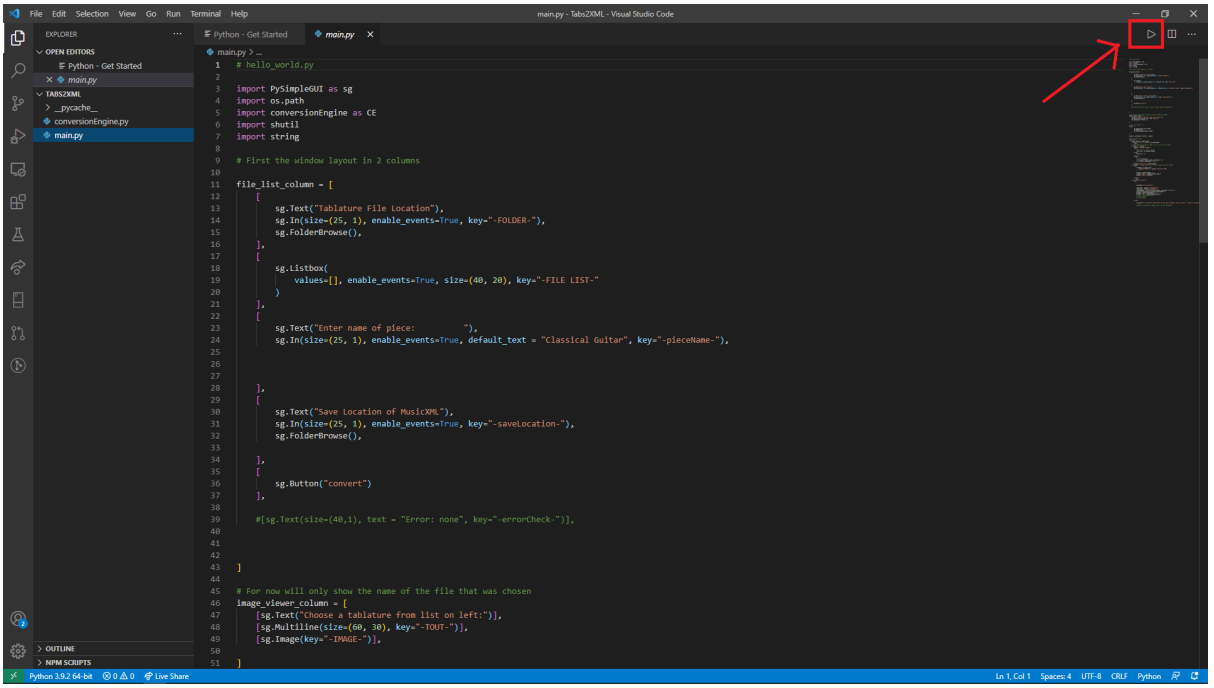
    - Note the working directory in the above example!
    - In the even there are any issues, please refer to: https://lxml.de/installation.html
- To get numpy:
    - Open Windows PowerShell (Windows) or Terminal (MacOS/Linux).
    - run the command: pip install numpy.

```
PS C:\Users\yasir\OneDrive\Desktop\2311\Tabs2XML> pip install numpy
```

    - Note the working directory in the above example!
    - In the even there are any issues, please refer to: https://pypi.org/project/numpy/

To launch Tabs2XML

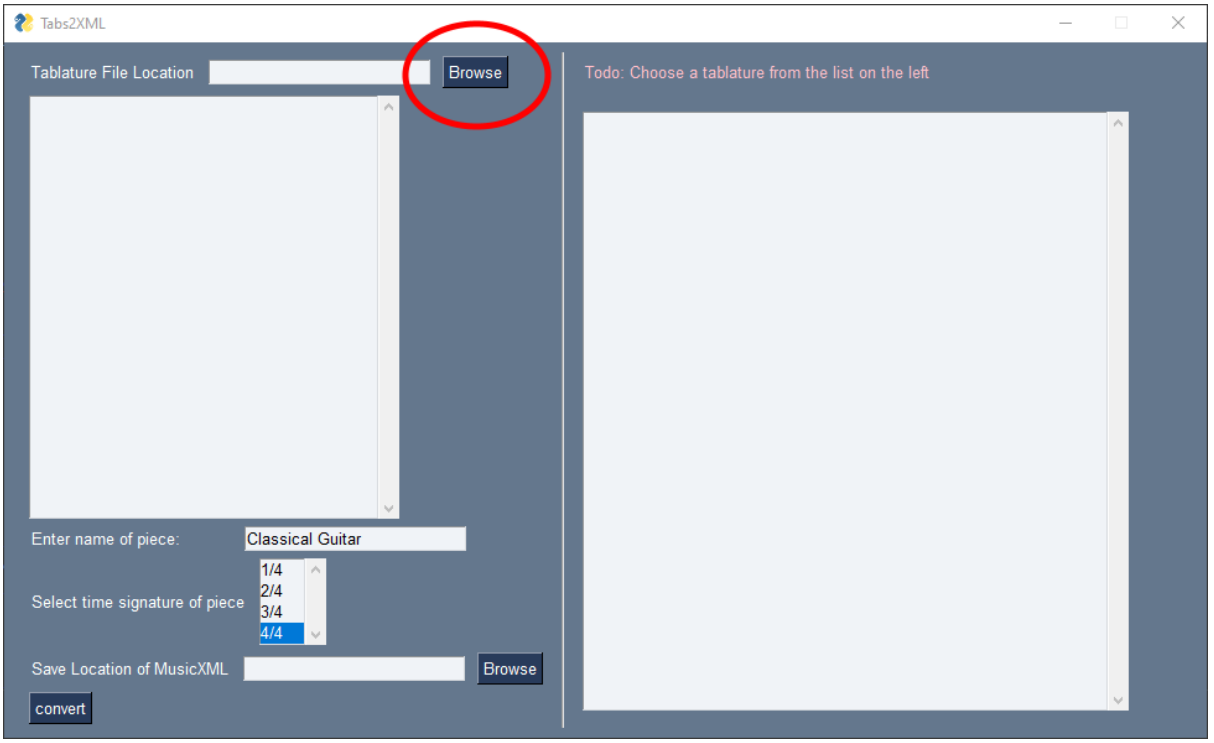
- Download the project as a zipped file from the github. (https://github.com/arjitjohar/Group10Project)
- Unzip the download and store it on your system.
- Launch VS Code.
- Click File > Open Folder > *Select the directory where the non-zipped project file is located*.
- Open main.py.
- Hit “Run Code” (play button at the top right) or hit Ctrl + ALT + N.



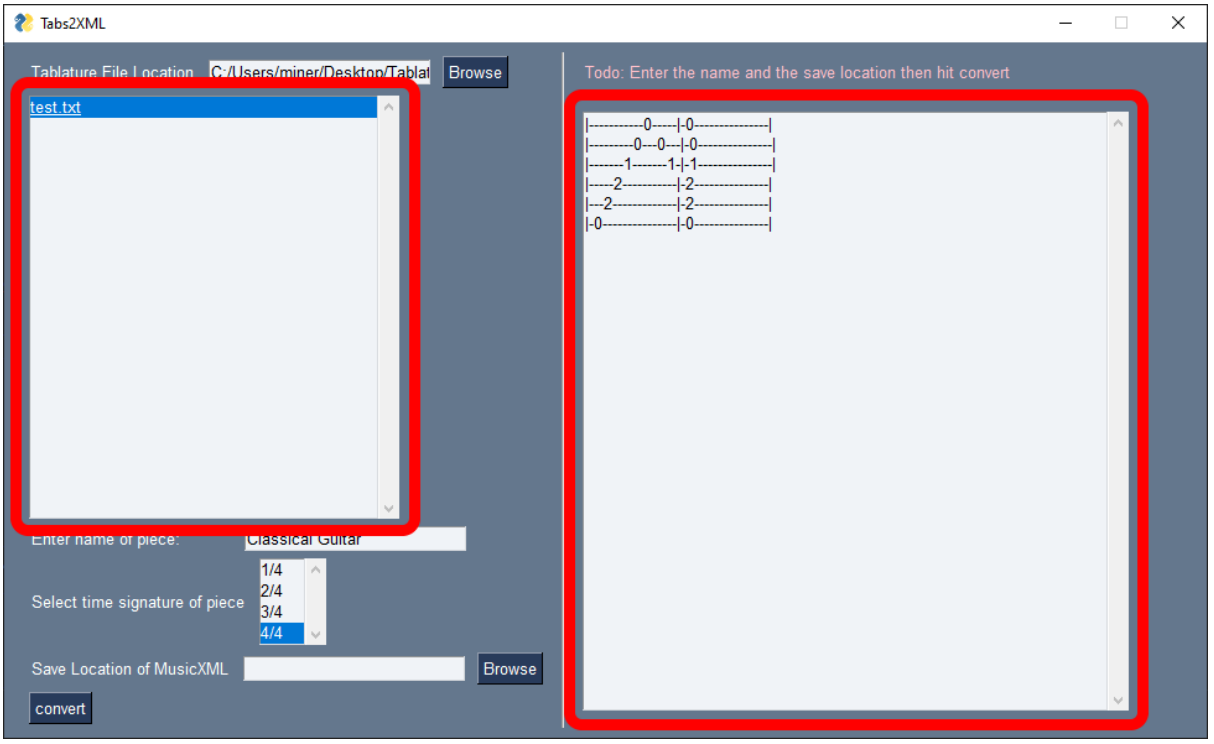
In the event that they require further assistance, they can react out to our support staff at hiangel@my.yorku.ca.

4.0 Description of How to Use/Operate the Product

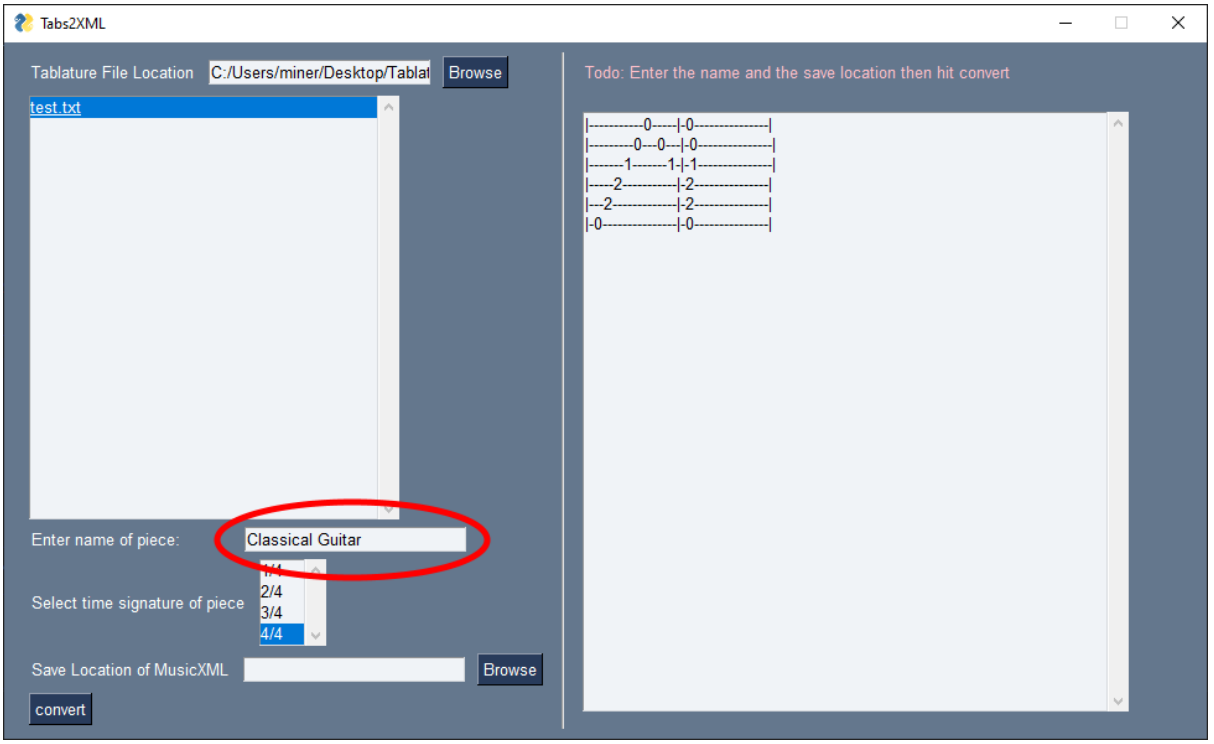
1. The user selects the directory where the file is stored by clicking on the “Browse” button.



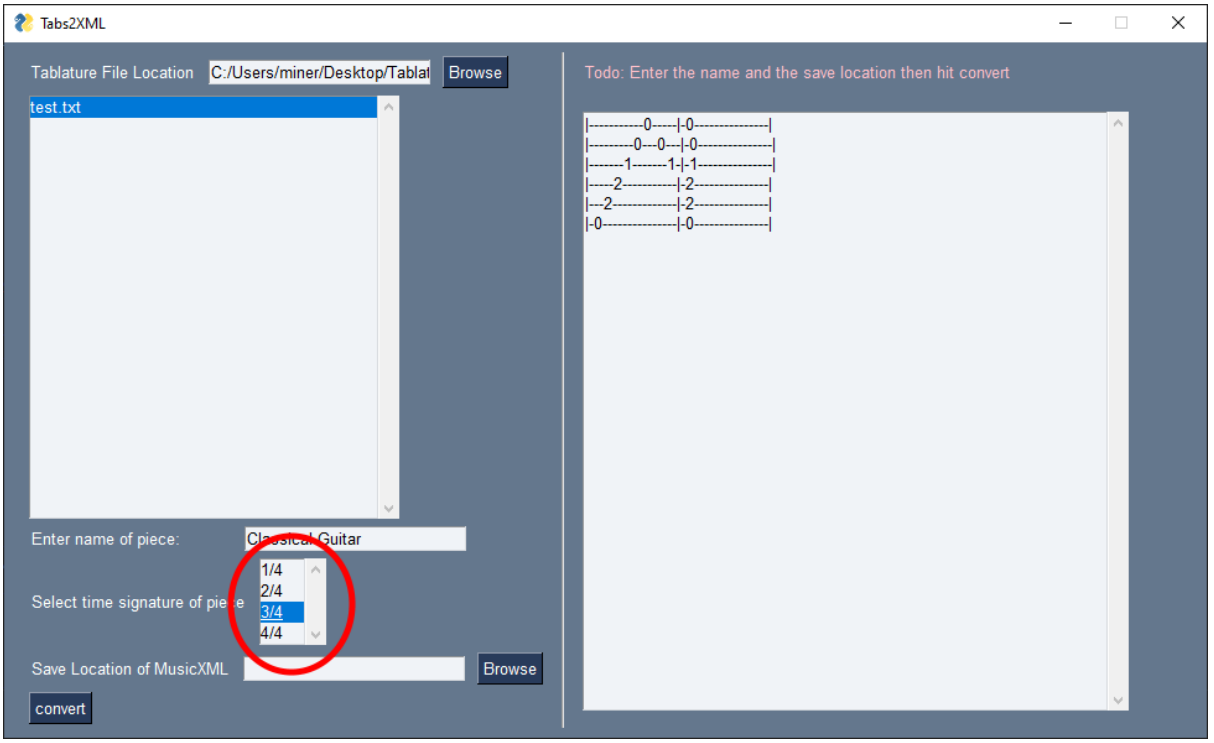
2. The user selects the tablature from the file list (left) that they wish to convert. A preview is then shown as to what the MusicXML file looks like (right).



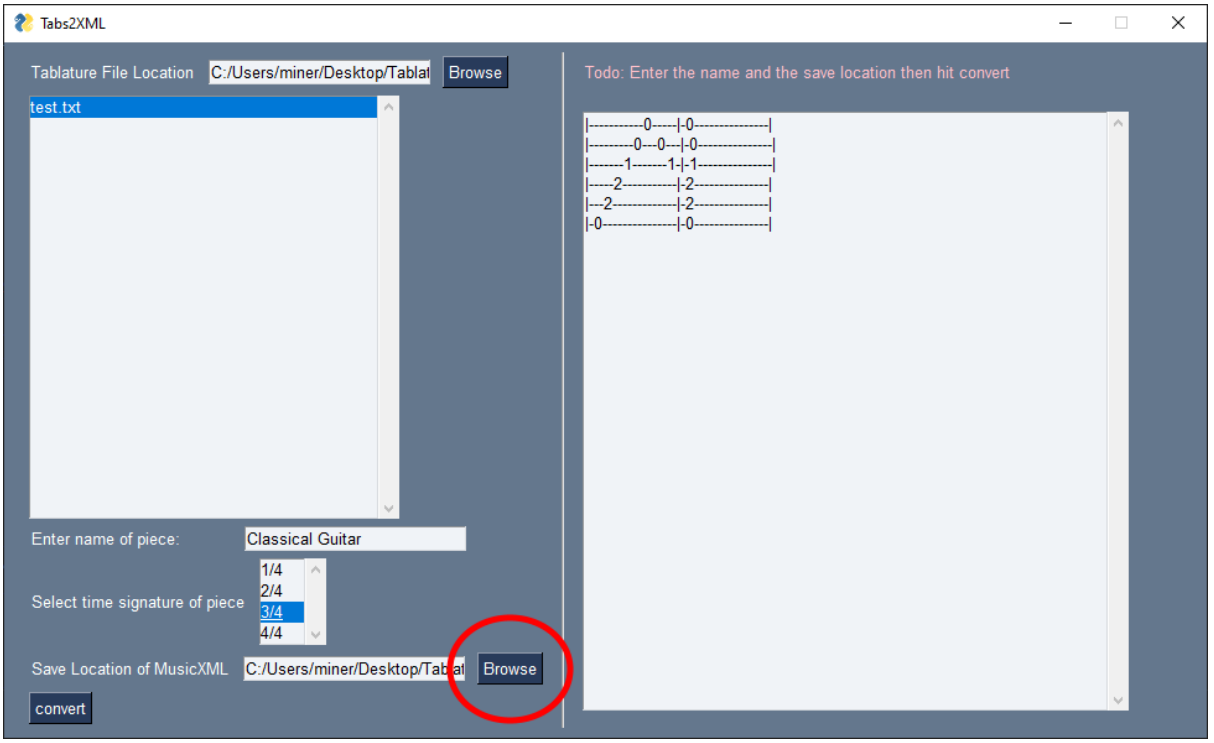
3. The user renames the name of the piece into what they want.



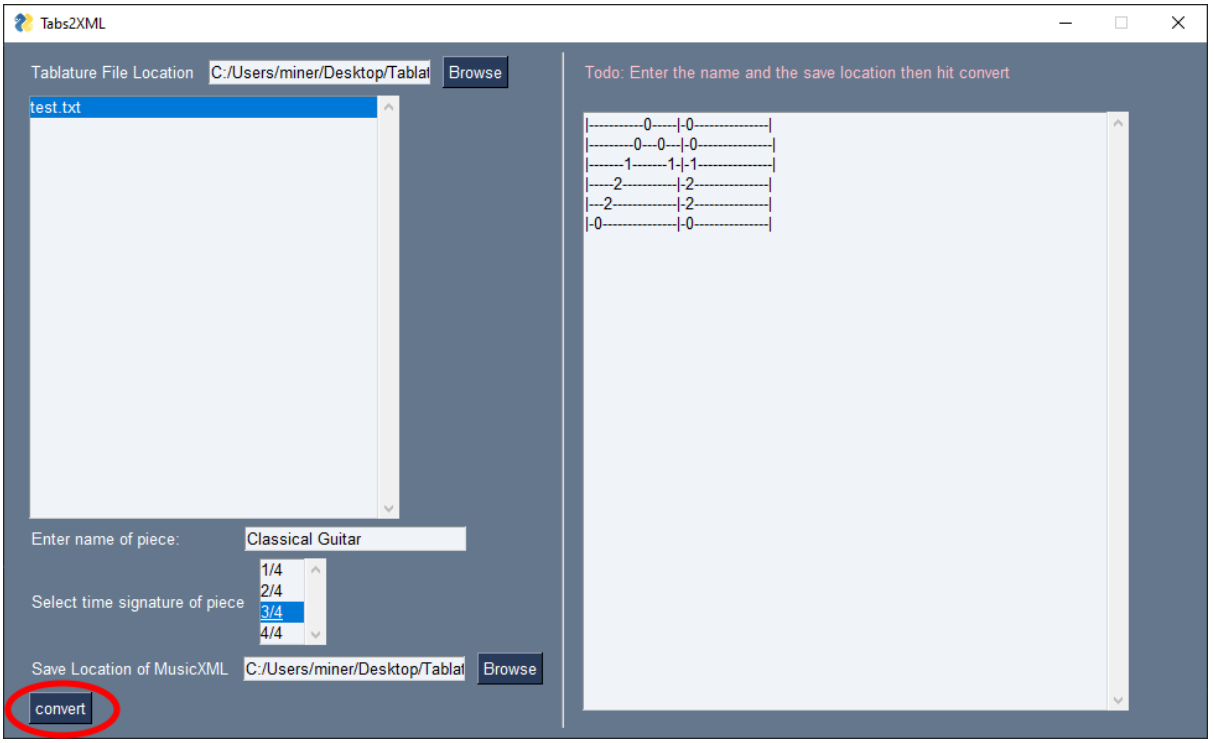
4. The user chooses which time signature they want to use if they want to change it(by default, this is set to 4/4).



5. The user then selects the directory where they want to save the converted MusicXML file.



6. The user can then press the convert button to change the tablature into a MusicXML file and save it onto the chosen directory.



7. The user can then view the MusicXML file on third party software or websites.

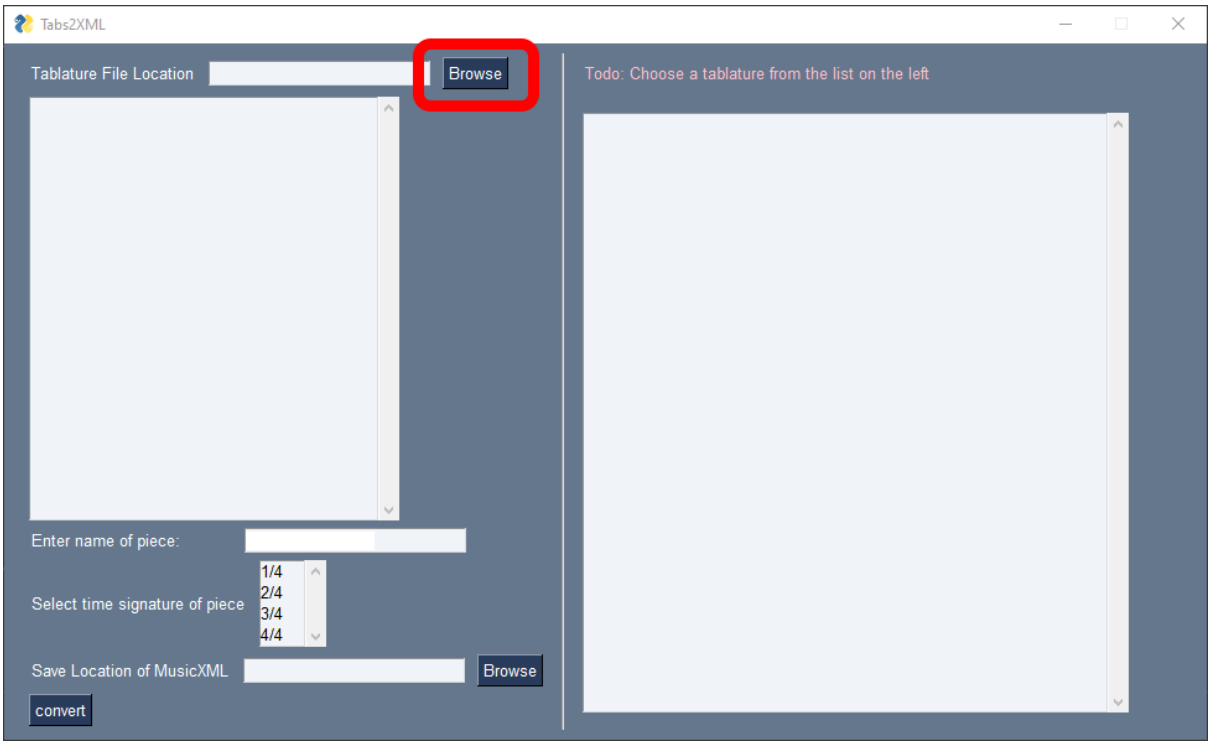
5.0 Troubleshooting & Solving Problems

In the case there is a problem with the conversion:

1. Close and re-open the program (sometimes, waiting roughly 30 seconds before running the program again is required).
2. If there is an update available, update your current version of Tabs2XML.
3. Ensure your VS Code is up to date and functioning properly with Tabs2XML.
4. In the event that they require further assistance, they can react out to our support staff at hiangel@my.yorku.ca.

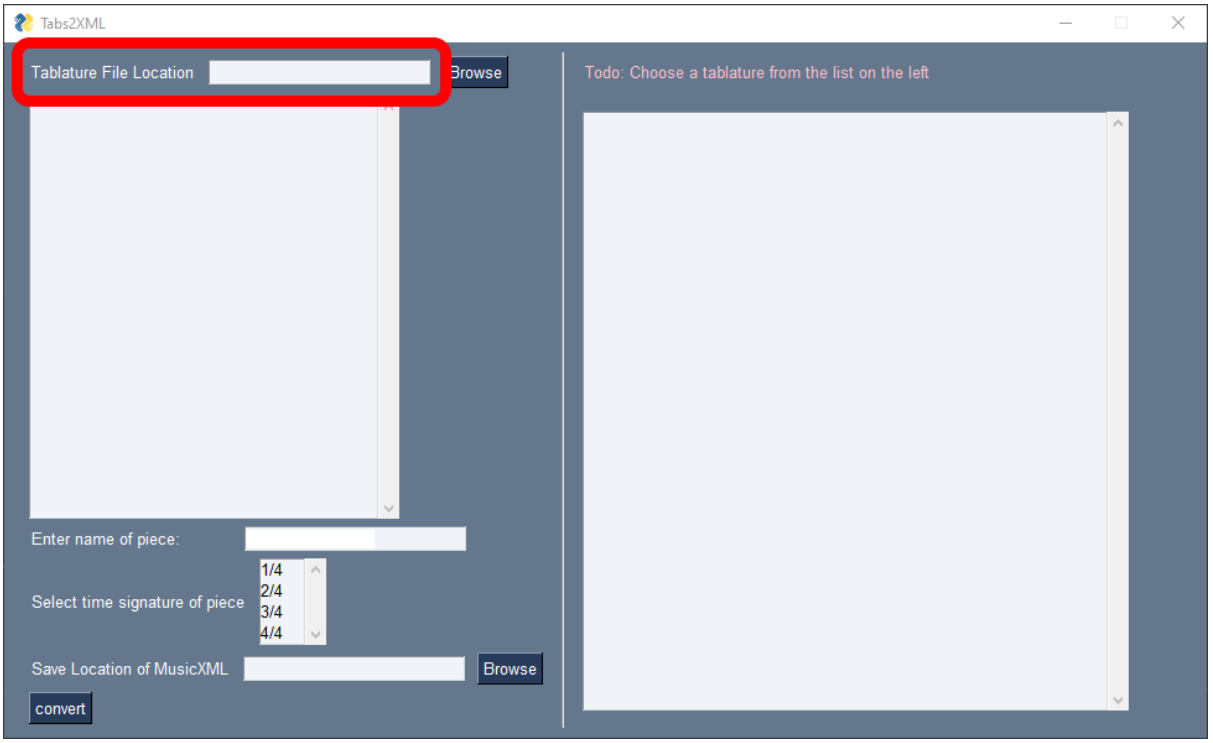
6.0 Description of the UI

6.1 Browse Button (Input)



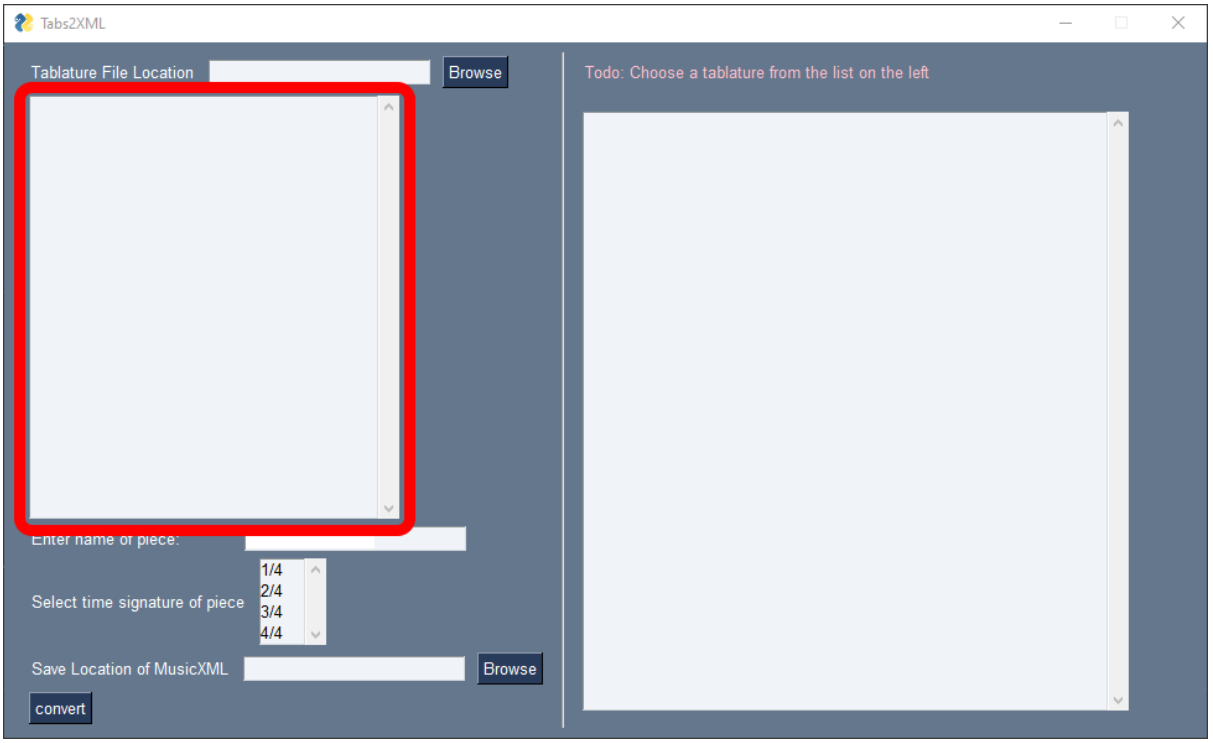
When clicked on, this button sends the user to their file management application, so they can choose the Tablature text file to convert.

6.2 Tablature File Location



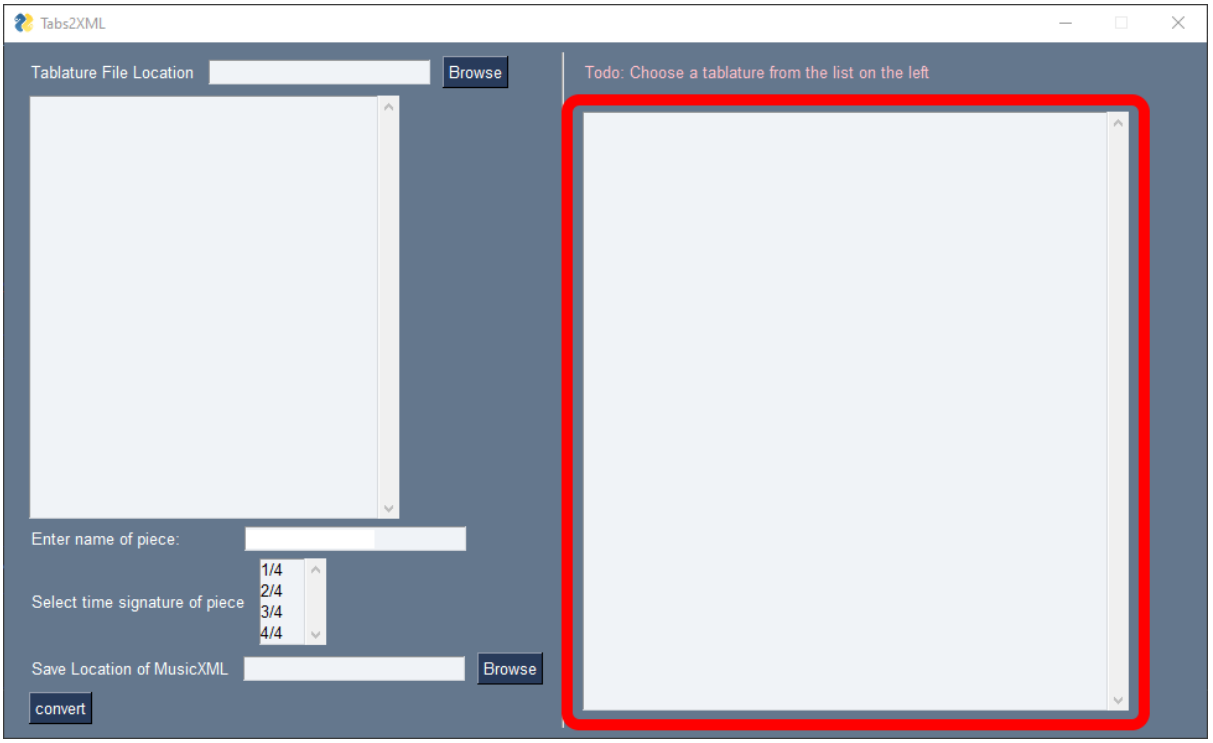
Displays the directory of the Tablature File being inputted. This is determined by the directory of the file selected when using the "Browse" button.

6.3 List of Files



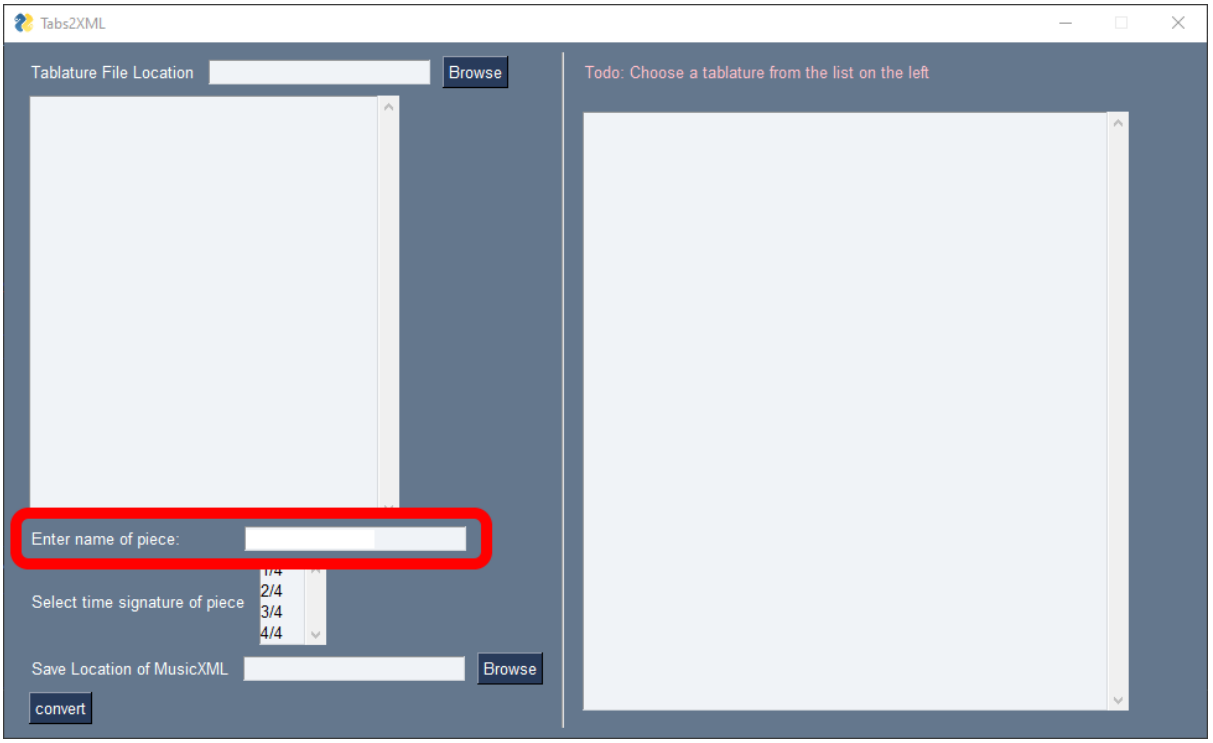
This is the list of files which have been inputted by the user using the "Browse" button (see above). Here, a file is chosen to be converted.

6.4 MusicXML Preview



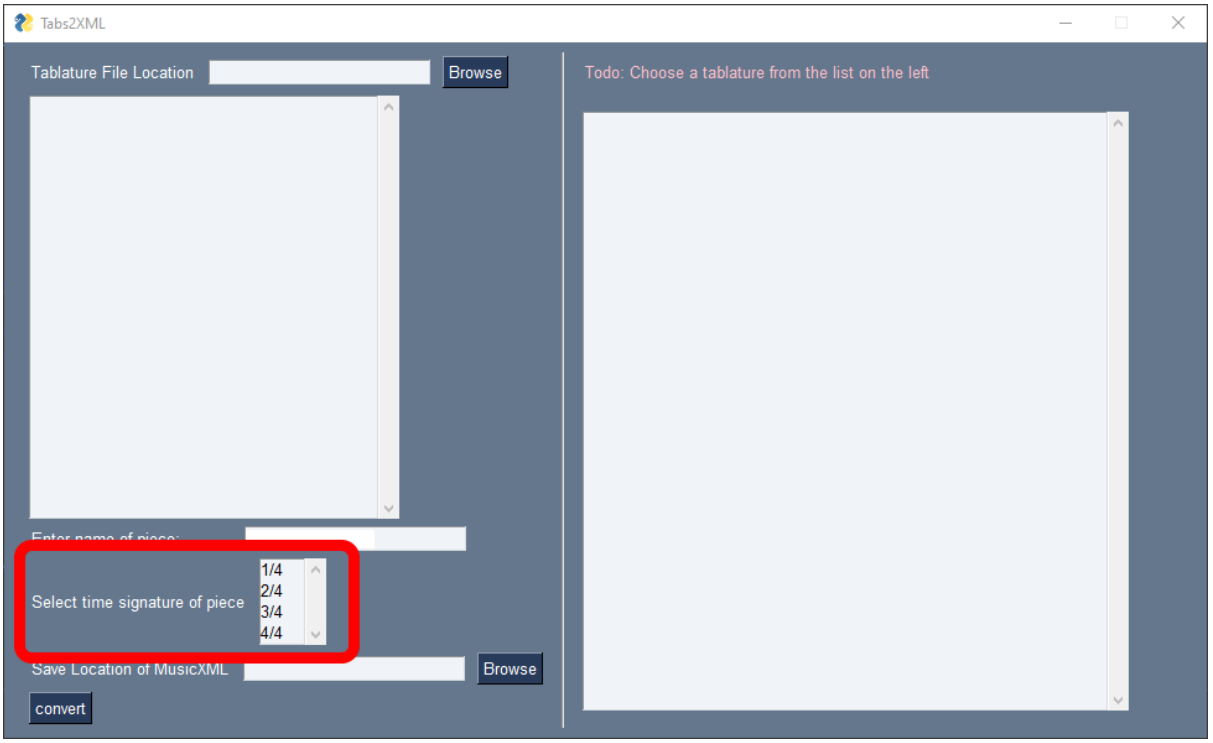
This text box displays what the final MusicXML file will look like once the conversion is complete.

6.5 Name of Piece



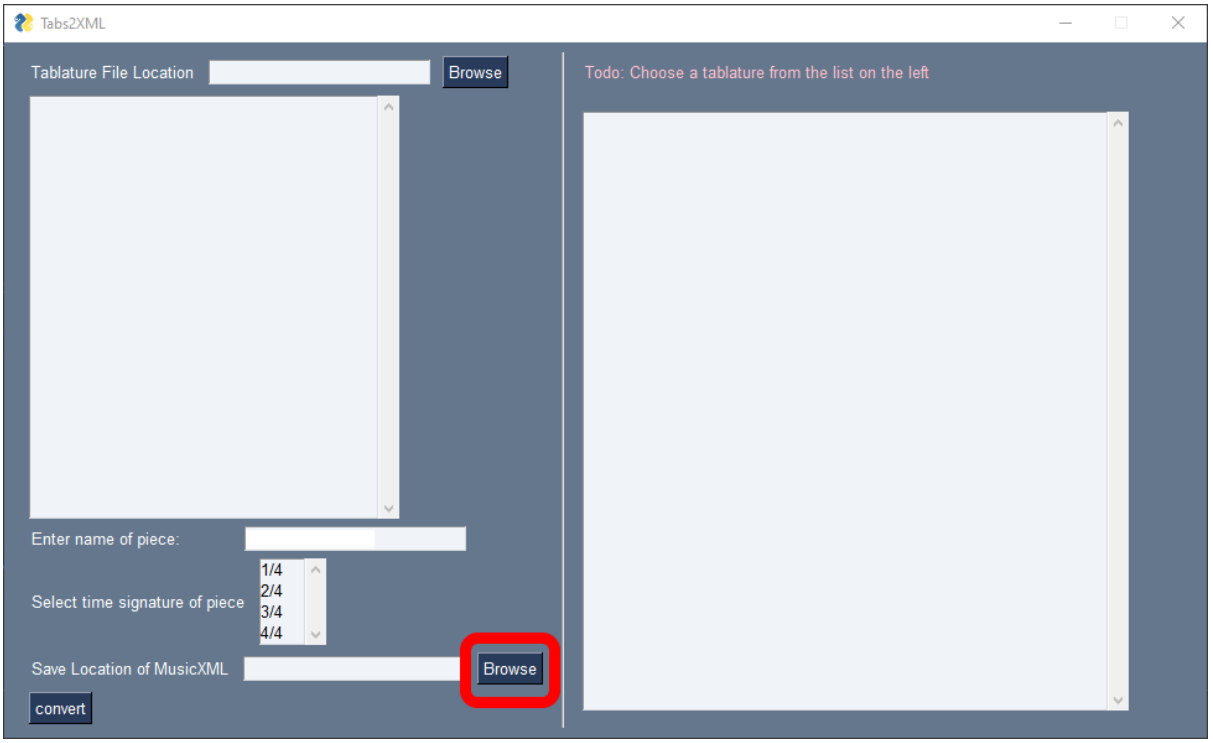
This text box is where the user changes the name of the song, and resultingly the name of the outputted MusicXML file.

6.6 Time Signature



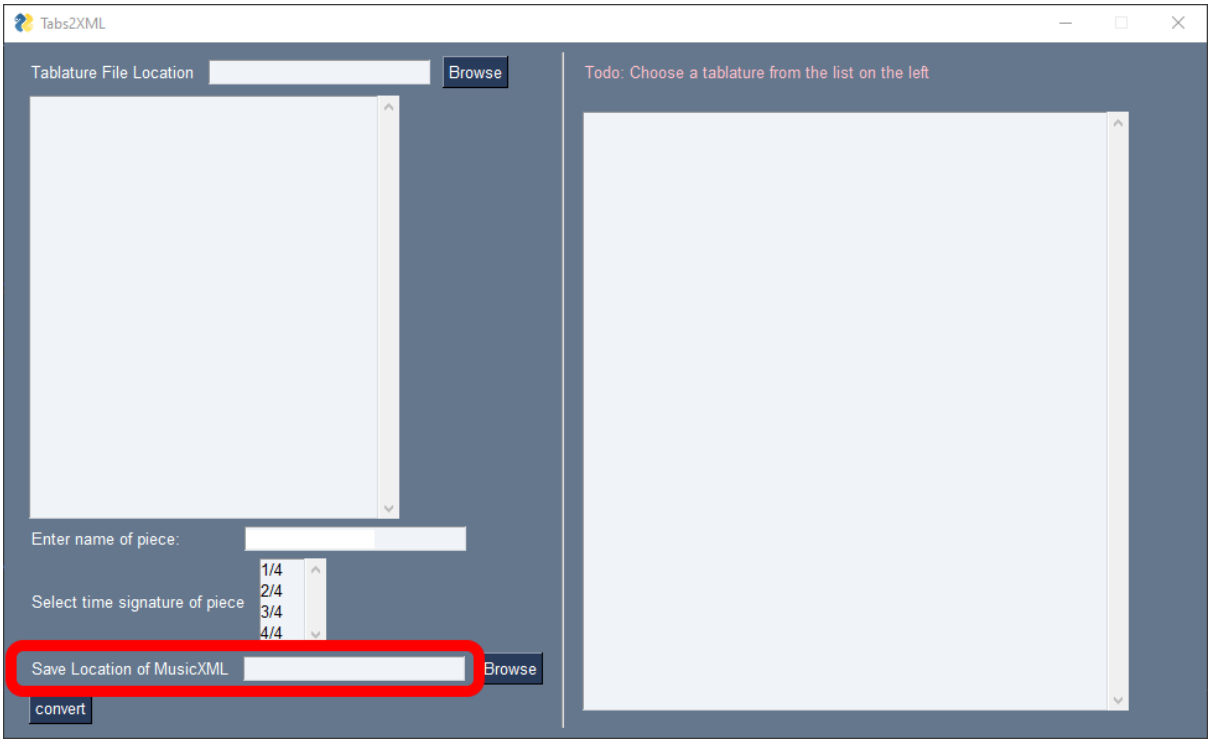
The user may select one of a set number of available time signatures for their converted piece.

6.7 Browse Button (Output)



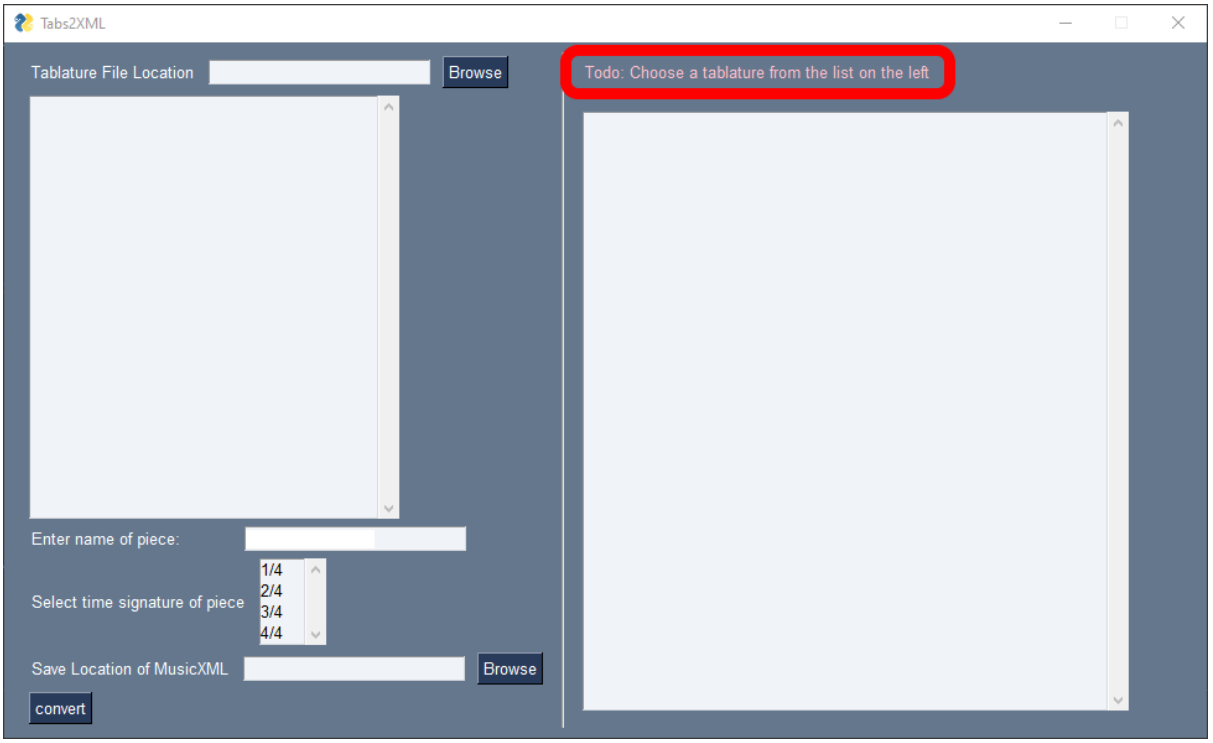
When clicked on, this button sends the user to their file management application, so they can choose the location where the outputted MusicXML file will save to.

6.8 Save Location of MusicXML



Displays the directory of the MusicXML file being outputted. This is determined by the directory of the file selected when using the "Browse" button (see above).

6.9 "To-Do" Instruction



The red text displayed on the top-right of the UI explains to the user what the next step of the conversion process is. For example, until a Tablature file is chosen from the list of files (see above), this text will read: “Todo: Choose a tablature from the list on the left.”