CSE 2311 Software Development Project

TAB2PDF

System Requirements

York University

Professor Vassilios Bil Tzerpos

bil@cse.yorku.ca

Team 3

Mohamed Nasar - nasarmo@hotmail.com

Calvin Tran - calvin.tran0@gmail.com

Jason Kuffaur - icefullbuster@gmail.com

Muhammad Shah - share.truth@yahoo.ca

Siraj Rauf - sirajrauff@gmail.com

Waleed Azhar - waleedghuman@gmail.com

Umer Zahoor - umerzahoor93@gmail.com

Jan – Apr 2014

Table of contents

Background

Ideal Music School is a medium size organization, of North America, that has grown from a single branch to six branches, in his first five years, since his inauguration in 2007. Recently they found that they have a lot of Guitar tablature files in ASCII TEXT format, and every day more files are coming. These music files are written in plain text format. That format does not support a lot of new and interesting features, those are available now in PDF format. But to write the files again, word by word, in PDF format, will be a big task, that will also have lot of chances for mistakes. Here comes the idea of our project. The chief executive of the school, came to York University, as he knew that his requirement is not so straight forward that he can buy a program off the shelf. He explained his idea to the Head of the Computer Department, that he wants to develop a Software System that can translate guitar tablature from ASCII to PDF format. The head of the department, referred him to Mr. Vassilios Bil Tzerpos, who is professor in the department and teaches several courses of Computer Science and electrical engineering. He assigned this project to a class of the under graduate students.

Overview

This document is prepared by the Ideal Software Developers group three, and will be presented to the client, upon completion. If the client will agree with the presentations in this document, he will sign and return the copy for the confirmation. Otherwise he can write notes on the document and, return it for the review. In several iterations this document will reach to its final shape.

The project has been named TAB2PDF. This document will describe the requirements for the Project. The project is to develop a software system that will translate guitar tablature from ASCII to PDF format.

The PDF format is a graphical format. So the client wants to utilize the powers of the graphical presentation and wants to have different line spacing and character spacing, just a few improvements to mention.

Title and Subtitle and spacing will be taken from the input text file, but client also wants to have the freedom to specify different Title, Subtitle and the spacing in the graphical user interface.

The client wants to improve the presentation of the files in a lot of ways. We will explain more requirements shorty later in this document.

Basic Requirements

Develop a system to convert guitar tablature from a text file in ASCII format to a PDF format.

The system should fulfil the following minimum requirements.

1. The system should work on at least the three following platforms:
    a. Windows.
    b. Unix (and similar operating systems).
    c. Macintosh.
2. System should present a Graphical User Interface.
3. User should be able to select the input file in GUI.
4. User should be able to select the output folder, where the output file will be created.
5. System should generate an output PDF file, according to the guidelines given in the Conversion Requirements.
6. The output file name should be extracted from the title with in the text file.
7. If the title is missing, the output file name should be set to untitled.pdf.
8. Before saving the output file the system should present the file name to the user, so user can change it if needed.
9. If the output file is present already in the same folder, the system should ask permission for overwriting.
10. Output PDF file should be previewed on the screen, before saving.

Basic Requirements (Continued)

11. The user should be able to change the following parameters in GUI. Then the display should reflect new parameters.
    1. Title
    2. Subtitle
    3. Font Name
    4. Font Size
    5. Number Spacing
    6. Measures Spacing
    7. Line Spacing
    8. Left Margin
    9. Right Margin
    10. Percentage of Zoom
12. Help File, User manual and “How to Get Started” files should be provided, in digital media.
13. Output PDF File Printing facility should be available.
14. System and all its functions should be easy to use.
15. All the functions should be available in the GUI.
16. The input file will be in the format, that is explained in The Conversion Requirements section, but the system should be able to overcome minor errors, those are explained in the error tolerance section.
17. Default page for the output file should be A4 size.
18. If output file is not possible to create, for example if the disk is full or the user has no right permissions for the folder, the system should display an error message.

Additional Requirements

Following requirements are nice to have requirements. Those were captured during the requirements analysis phase. If the system can incorporate it will be nice otherwise those are not in the condition for the final system acceptance.

1. Incorporate user preferences so user can opt out, to give the output file name.
2. User can opt that system warns if the file already exists. Means if the user preference says, the system should overwrite the file without asking.
3. System may incorporate version facility also by having an option to check the input file name and the output file will have a matching output file name. If the output file name is already present system will append an integer to the file name that will be incremented if the user converts same input file again and again. This way user can have many versions of a file.
4. System will show the progress of the file conversion
5. System will keep a list of the recently used files
6. System Can Display the Input File
7. System can perform a Demo.
8. User can select the back ground colour of the output file.
9. User can select the colour of the output writing.
10. User can give text to append at the top of the page.
11. User can give text to append at the bottom of the page.
12. User can give picture(s) to add in to the file.
13. User can select to print the date and time in the output file. This can be printed on the top or on the bottom. This can be system date and time at the time of printing, or user provided date and time. This can be the creation Date and Time of the input file.

Additional Requirements (Continued)

14. User can specify the height and width of the output page.
15. User can specify top and bottom margins of the output page.

Conversion Requirements

Numbers should be converted to corresponding numbers. Dashes should be converted to lines.

Title should be grabbed from one of the two, following format line examples, and it should be displayed in Bigger Font, with the center alignment.

TITLE=Remembering Rain
title=Remembering Rain

Subtitle should be taken from the following format line examples. It will be displayed in a font size smaller than the Title, but still bigger than the details of the page. It should be also in the center alignment.

SUBTITLE=Jim Matheos
subtitle= Jim Matheos

Spacing will be captured from the following line example. It will define the spacing between the numbers.

SPACING=4.4
spacing=4.4

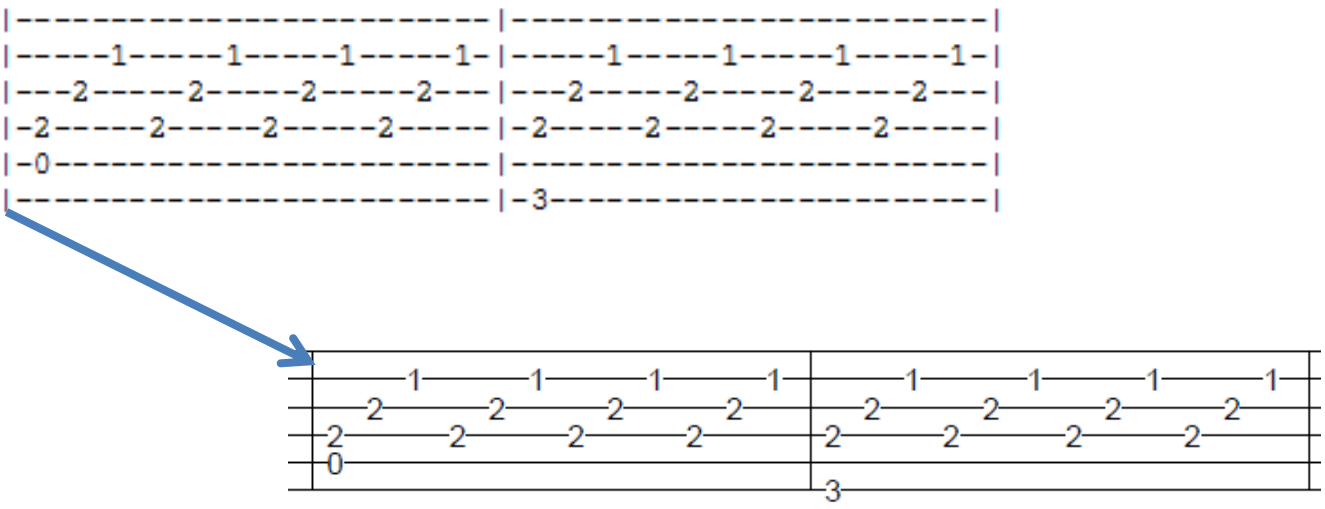
These parameters once captured, should be displayed in the GUI. Then the user should be able to make changes in them, and the result should be updated in the display area.

If the input file has additional white spaces between the groups of the six lines, it should not crash the system, these lines should be simply ignored and the output should not have that extra white space.

Similarly if there is no space between the two sets of 6 lines, system should not crash. The system should put proper space in between such these two sets, in the output.

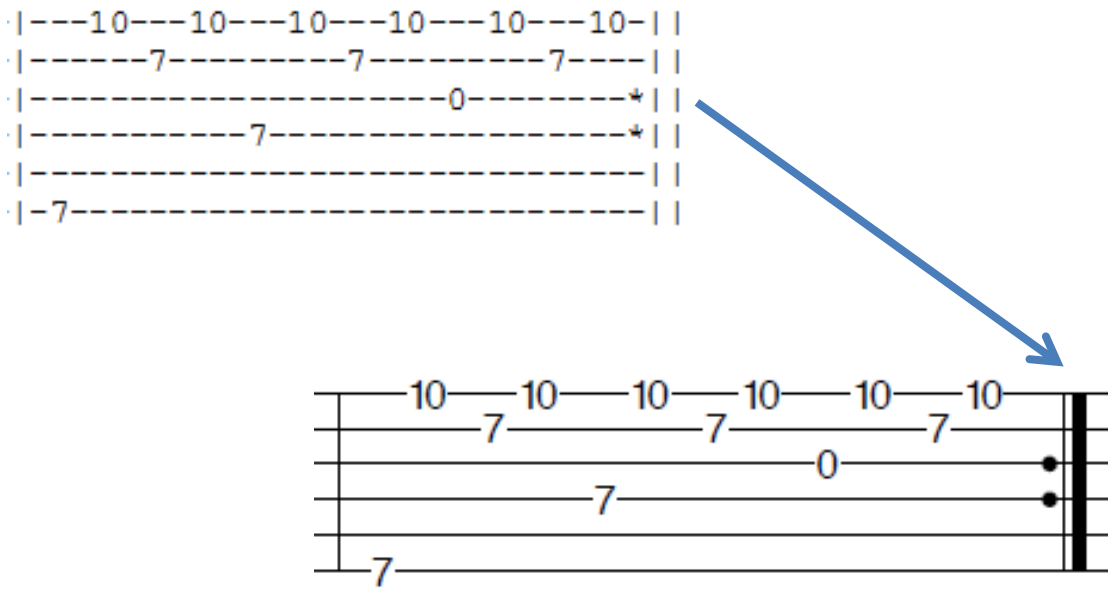
Conversion Requirements (Continued)

Single bars appear in the text file like the following example.



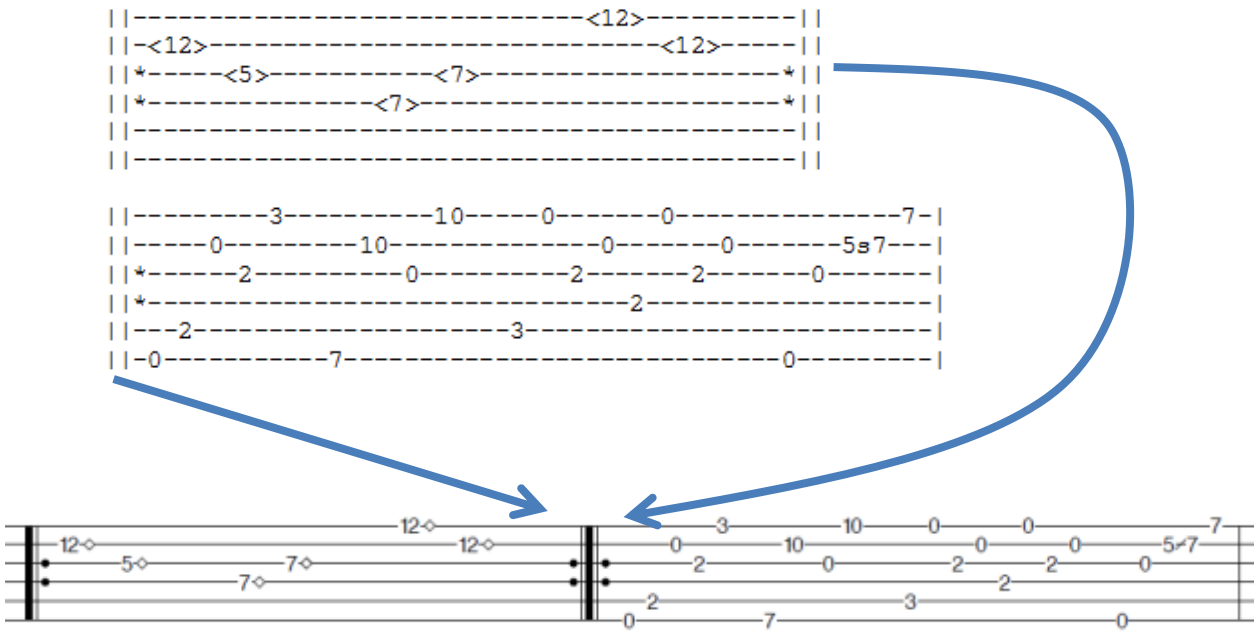
Those should be replaced by a single vertical bar, just one continuous bar for the six horizontal lines, shown as above.

If there appears double bar and * in the input text file, it should be converted to a circle in the output PDF file, like the following example.

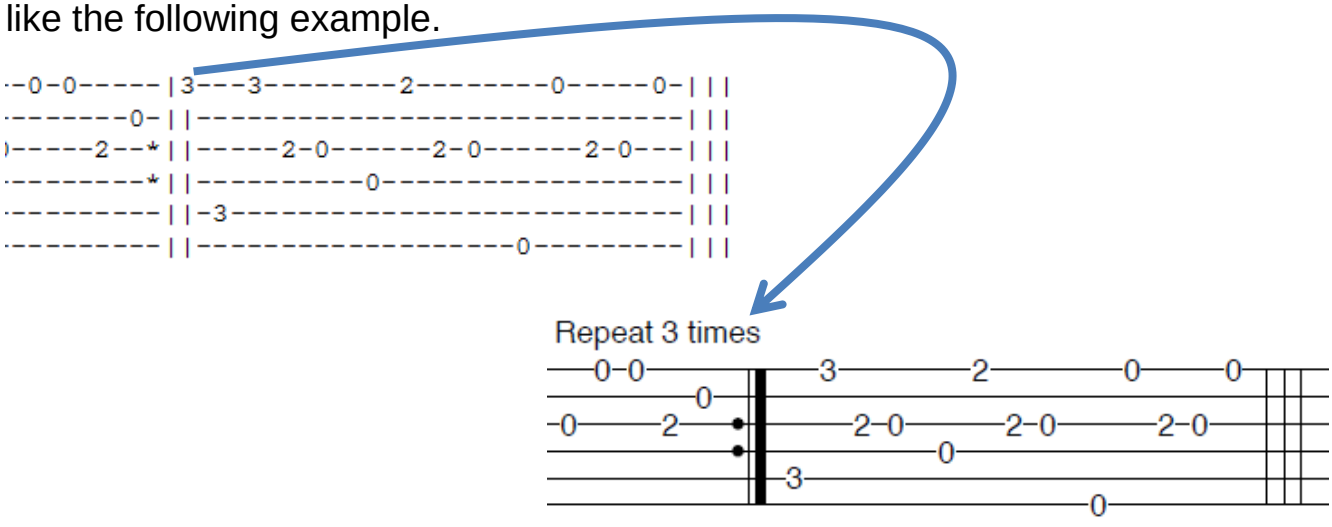


Conversion Requirements (Continued)

If double bar appears at the end, and also appears at the next beginning, and the output is coming on one line, it will be replaced only by the three lines not by the four lines. See the following example.



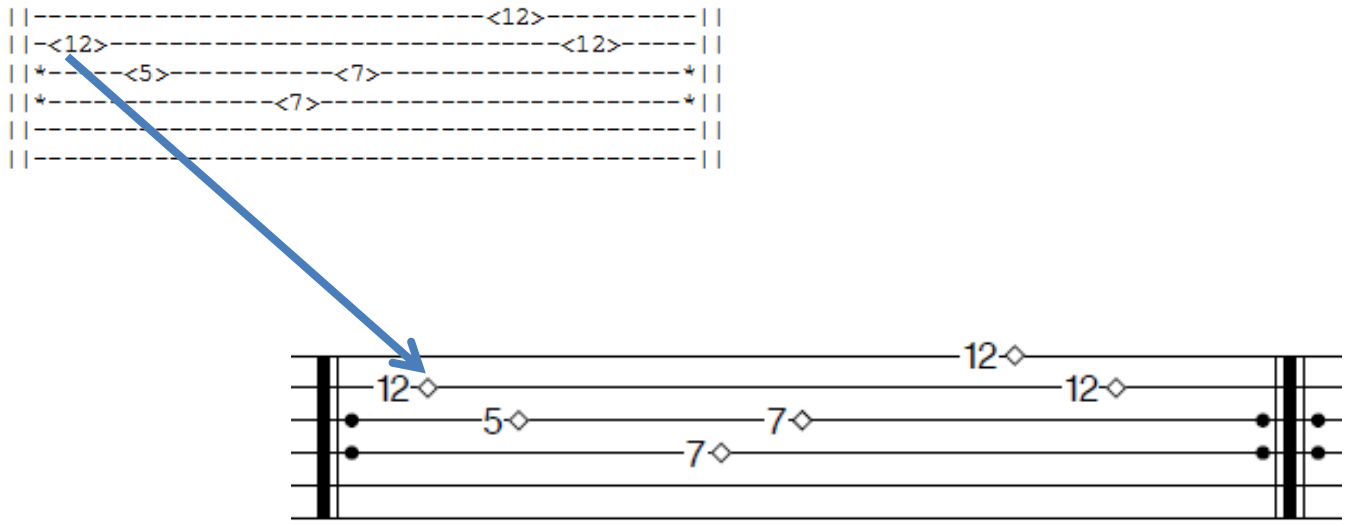
If a bar is followed by a number it is signal for repetition. It should be converted like the following example.



In the above example we also see that how the three vertical lines will be converted in to corresponding three lines.

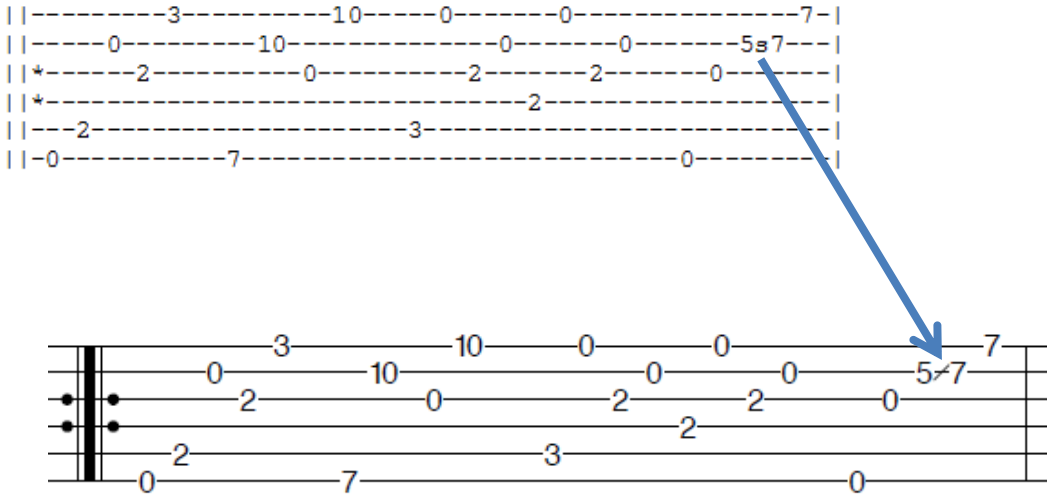
Conversion Requirements (Continued)

Harmonics should be converted like the following example.



They appear in text as < and >, embracing the number.

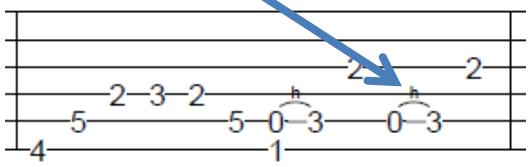
“Slide” should be converted as the following example. It appears as the small ‘s’ in between the two numbers in the input text file.



Conversion Requirements (Continued)

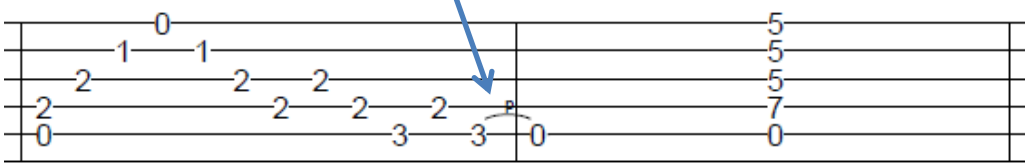
“Hammer” should be converted as shown in the following example.

```
|------------------------|
|------------------------|
|----------------2-----2-|
|-----2-3-2--------------|
|---5-------5-0h3---0h3--|
|-4----------1-----------|
```



It appears as small ‘h’ in between the two numbers.

```
|-------0----------------|-------------5----------|
|-----1---1--------------|-------------5----------|
|---2-------2---2--------|-------------5----------|
|-2----------2---2---2---|-------------7----------|
|-0----------------3---3-|p0----------0----------|
|------------------------|------------------------|
```



Small p should be converted as shown in the above example.

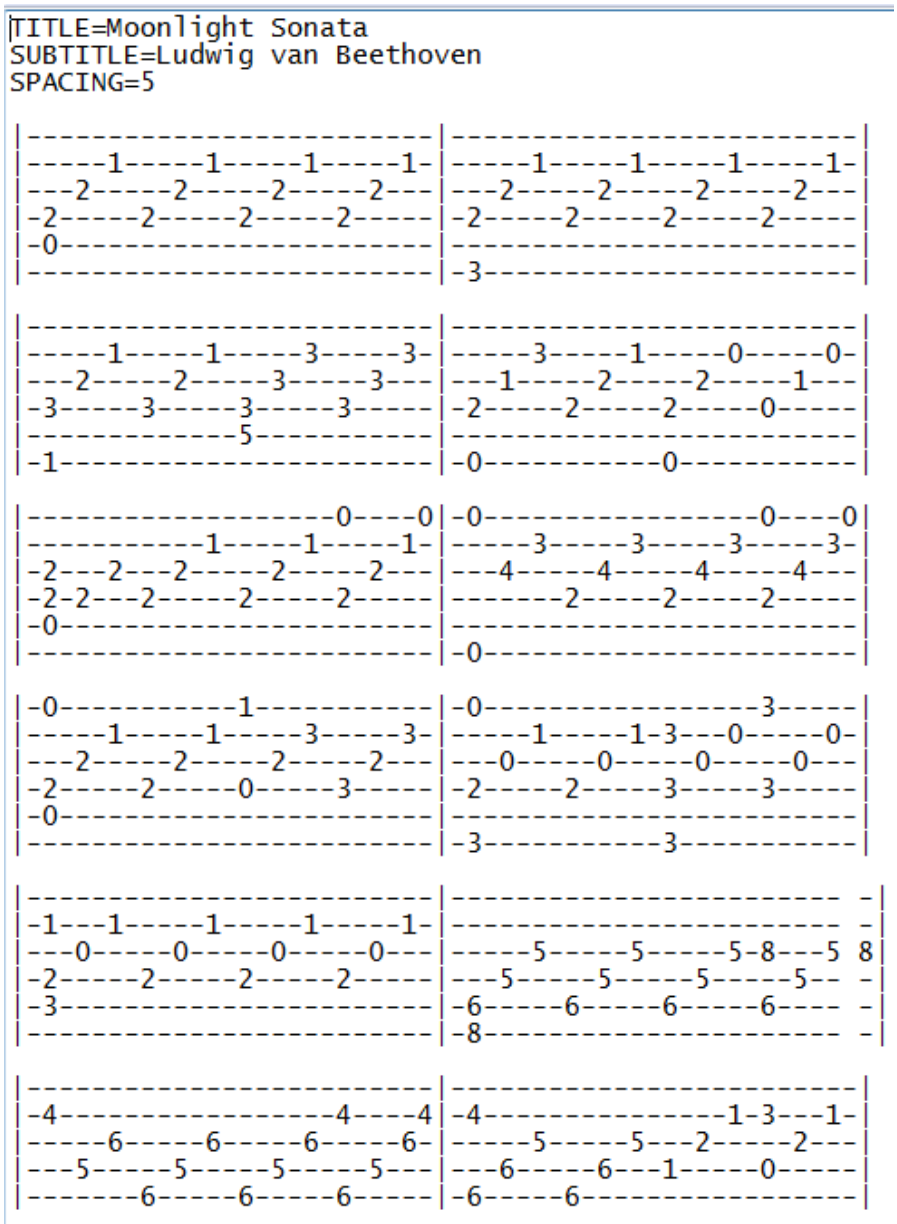
List of the accepted symbols is given below.
\ | - s , * + < > 0-9 ^ ( ) h p = g S s % e x /

Only the symbols explained in the above examples are required to be converted in this phase of the development. Rest of the symbols are nice to have conversions, may be dealt with in some next development. If those are also dealt with, in this development, that will be very nice.

Acceptance Test Cases

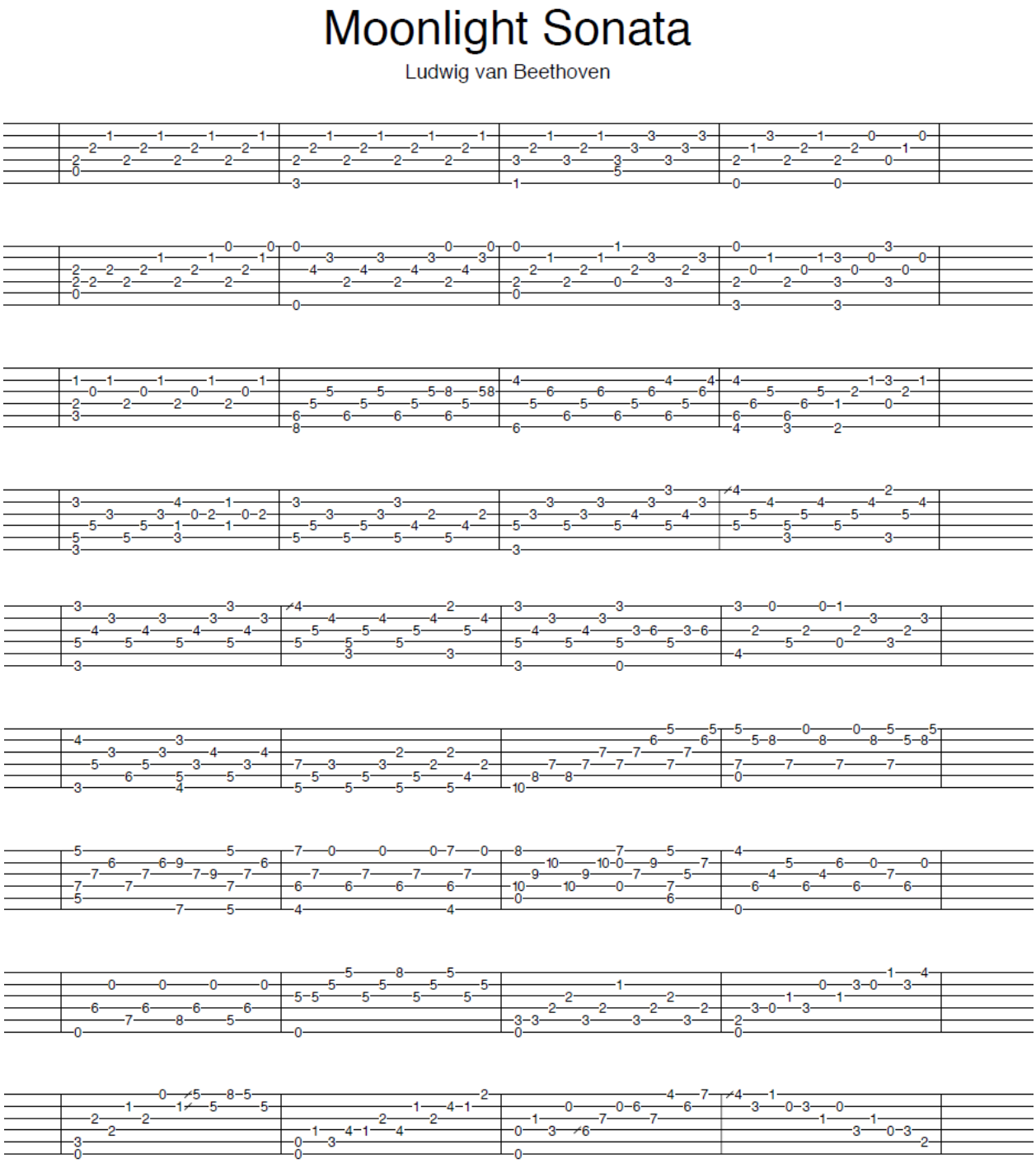
System will have only one type of input that will be in text format in a text file. Input file will be actually a guitar musical instruction file. Following are two input samples. Here are only the images of the first pages of the files, Client has provided the input and expected output files in the digital format. The system will be accepted only if it can produce the required output from the given input files.

Acceptance Test Case 1. First input file looks like this.



Acceptance Test Cases (Continued)

System has only one output and that will be in PDF format. The system should create a PDF file like this, from the first input file.



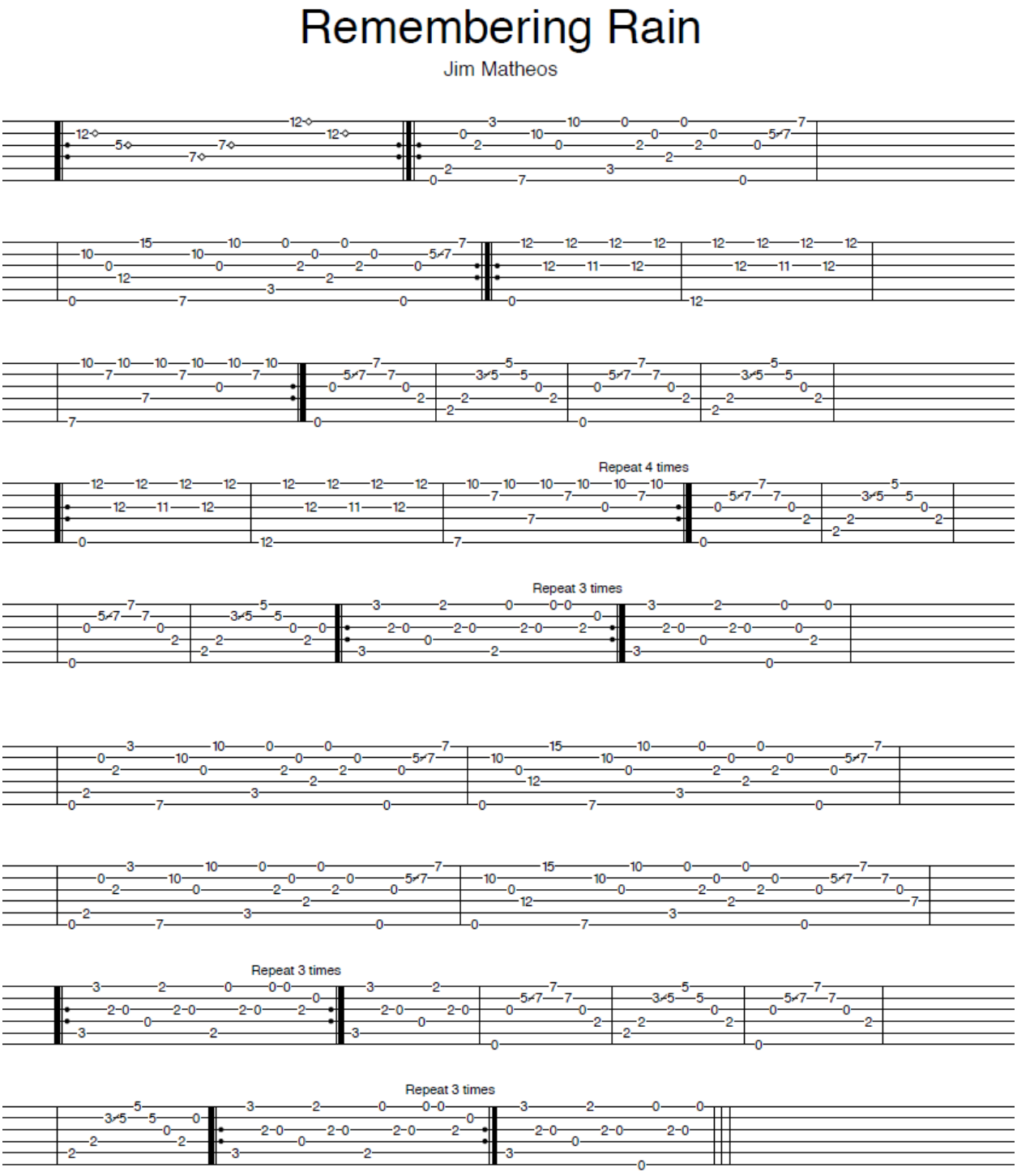
Acceptance Test Cases (Continued)

Second input file looks like this.

```
TITLE=Remembering Rain
SUBTITLE=Jim Matheos
SPACING=4.4

||-------------------------------<12>----------||
||-<12>-------------------------------<12>-----||
||*-----<5>-----------<7>--------------------*||
||*---------------<7>-------------------------*||
||-----------------------------------------------||
||-----------------------------------------------||

||---------3----------10-----0-------0---------------7-|
||-----0---------10---------------0-------0-------5s7---|
||*------2----------0----------2-------2-------0-------|
||*-----------------------------------2-----------------|
||---2----------------------3-------------------------|
||-0-----------7-------------------------------0---------|

|-----------15----------10-----0-------0---------------7-||
|---10-------------10--------------0-------0-------5s7---||
|------0--------------0----------2-------2-------0------*||
|--------12---------------------------2-----------------*||
|-------------------------3-------------------------||
|-0--------------7-------------------------------0---------||

||---12----12----12----12-|----12----12----12----12-|---10---10---10---10---10---10-||
||-----------------------|-------------------------|------7---------7---------7----||
||*-----12----11----12----|-------12----11----12----|---------------------0--------*||
||*-----------------------|-------------------------|-----------7------------------*||
||-----------------------|-------------------------|-------------------------------||
||-0---------------------|-12---------------------|-7-----------------------------||

|---------7-------|---------5-------|---------7-------|---------5-------|
|-----5s7---7-----|-----3s5---5-----|-----5s7---7-----|-----3s5---5-----|
|---0---------0---|-------------0---|---0---------0---|-------------0---|
|---------------2-|---2-----------2-|-----------------2-|---2-----------2-|
|-----------------|-2---------------|-----------------|-2---------------|
|-0---------------|-----------------|-0---------------|-----------------|

||---12----12----12----12-|----12----12----12----12-|---10---10---10---10---10---10-|4
||-----------------------|-------------------------|------7---------7---------7----||
||*-----12----11----12----|-------12----11----12----|---------------------0--------*||
||*-----------------------|-------------------------|-----------7------------------*||
||-----------------------|-------------------------|-------------------------------||
||-0---------------------|-12---------------------|-7-----------------------------||

|---------7-------|---------5-------|---------7-------|---------5---------|
|-----5s7---7-----|-----3s5---5-----|-----5s7---7-----|-----3s5---5-------|
|---0---------0---|-------------0---|---0---------0---|-------------0---0-|
|---------------2-|---2-----------2-|-----------------2-|---2-----------2---|
|-----------------|-2---------------|-----------------|-2-----------------|
|-0---------------|-----------------|-0---------------|-------------------|


||---3--------2--------0-----0-0-----|3---3--------2--------0-----0--|
||-----------------------------------0-||-------------------------------|
||*----2-0------2-0------2-0-----2--*||-----2-0------2-0------0------|
||*---------0-------------------------*||----------0--------------2----|
||-3-----------------2----------------||-3-----------------------------|
||-----------------------------------||--------------------0----------|

|---------3----------10-----0-------0---------------7--|
|-----0----------10--------------0-------0-------5s7----|
|-------2----------0----------2-------2-------0--------|
|-------------------------------2---------------------|
|---2----------------------3-------------------------|
|-0----------7-------------------------------0----------|
```

Acceptance Test Cases (Continued)

system should create an output like this from the 2nd input file.



Error Tolerance

The input file is assumed to be as, it is explained in the Conversion Requirements and the Acceptance Test Cases sections, but the system should be able to work even the input file has following error situations.

1. Title is missing.
2. Subtitle is missing.
3. Spacing is missing.

If these items are not mentioned in the input file, system should be able to get those from the user and able to include those in the output file.

4. If the page size of the input file is not an A4 size, the system should not complain and should set the output file page size to A4.

Budget

The project should be completed within the budget limits, discussed and agreed upon between the client and the department management.

Time Frame

The project should be completed within the time frame limits, discussed and agreed upon between the client and the department management.