CS 503: Final Project: Version 1.0

DISCLAIMER:
The Sample provided with the spec is merely a sample and does not reflect all elements of the spec. Follow the spec and use the sample to clarify any ambiguity the spec may have.

CS 503: Final Project: Version 1.0

In this project, you are to develop a wave manager application. The following is a list of specifications and requirements that your project should support.

1. Your project should be an MDI Application that supports:
   a. A Main Frame MDI Parent window that contains a user control (call it FileViewControl).
   b. An MDI child window containing a user control (Call it GraphViewControl).
   c. The FileViewControl has a splitter Window (not a split container) to separate it from all MDI Children.

2. The main menu should support a similar set of menus as demonstrated in the provided sample .

3. The tool bar should support a similar set of buttons as demonstrated in the Sample executable.

4. The Status bar should have 5 panels:
   a. A progress bar that shows memory usage in MB. Note that the scale of the progress bar is 0-100 MB. Also note that a tool tip should show the correct memory usage when the mouse hovers over that control.

   b. A DropDownList-style Combo Box. This should hold any errors that occur during the application's lifetime. In real-life this should be errors that have been logged to some logging medium, but for this project, it simply logs to the combo box Status panel.

   c. The number of currently open files.

   d. The number of samples for the currently active MDI child window.

   e. A Track bar that allows volume control.

5. When the application starts up it displays a splash screen that contains some picture. Please do not put in a large image (must be < 2 MB).

Startup

6. When the application starts, it should display the last set of wave files that were open when the application was last closed.

File Menu

7. A File > New menu to allow a new/blank MDI child window to be launched.

8. A File > Open… menu.

9. The application should be capable of loading wave (.wav) files either by
   a. Invoking the File > Open… menu. This should support selecting one or more files (Multiple files Must be allowed) in the Open File Dialog .

   b. By dragging file(s) from Windows explorer and dropping it onto the application.

   c. Once a file is successfully open, the directory in which that file resides should be added to the FileView and all files that have the same extension are added as subnodes. Using distinct icons for File and Folder nodes is required.

10. The user additionally should be able to:

    a. Drag a file from the file view and drop it into the main frame window which also results in the dragged file being open.

    b. Double Clicking on the file path within the FileView would open the document as well.

    c. When a file is open the full path name is displayed as the title of the MDI Child window.

    The application is not required to open all types of wave files. It is required to be able to open the wave files I supplied, and any similar wave files that are mono uncompressed wave files.

    I have supplied some wave files that are corrupt files. Your application should be able to gracefully inform the user of the corrupt file instead of crashing.

    The mono uncompressed wave file format on disk is as follows:
    Header [40 bytes]
    Represents the wave header information. You do not need to interpret this header except as explicitly indicated in this spec.

    N [4 bytes]
    N is an integer that represents the number of samples to follow.

    Data Samples [N bytes]
    Each byte represents a sample point.

    The first 4 bytes out of the wave header contains “RIFF”. Make sure you test this header cookie for validity of the wave file. Keep in mind that some wave files provided are corrupt, while others are fake wave files.

11. File > Save automatically saves the file if the file is modified and exists.

12. If the file does not exist, the application should provide saving the file to a different filename. Invoking (File > Save As…) should provide the same behavior.

13. The File > Save menu item should be grayed out if the document contains no data or if the data it contains has not been modified.

14. The "File" > "Save As…" menu should be grayed out only if the document contains no data.

15. The “File > Save As…” should allow saving the file to a .wav or a .png format. The "Save As…" Dialog should contain two filters: .wav or .png, if the user selects .wav, then the file is saved as a valid wav file, otherwise it is saved as an image using the extension of png.

16. A File > Close as well as a File > Close All menu items should be available to allow closing the active MDI child or all MDI Child windows respectively. If a document has been modified the user should be prompted before the corresponding MDI child window(s) is closed.

17. A File > Page Setup… menu that allows the user to setup page properties: Margins, orientation, etc. These settings should be sticky each time a print job is initiated. Sticky is the highly regarded technical term :) for persistent as long as the application has NOT restarted.

18. A File > Print… and Print Preview… menus that allow printing and print previewing graphically of the currently active wave file. This Printing/Previewing should utilize the properties of the

Page Setup dialog. Printing should be performed according to the Viewing mode and should support both Full/Standard view modes. In the Full mode, the whole wave as seen should be printed to one page. In Standard mode, multiple pages should be printed to accommodate the whole wave spectrum.

Edit Menu

19. Undo: This is a one-level undo. It should be capable of undoing any operations that modify the data. e.g. Cut, Paste, Delete, Modulate and Rotate. The Undo menu should be grayed out if there is nothing to Undo. This can happen under 2 conditions:

    a. The document has not been modified, so there is nothing to Undo.

    b. The document has been modified more than once followed by one Undo. Since the Undo is a one-level operation, it means that once you Undo the last operation, all prior operations cannot be further undone.

    c. Cut: Allows cutting the entire wave into the clipboard using a custom clipboard format. This allows applications that receive that custom format to paste it into them. At this point, the only application that would understand this format would be another instance of the same application. This should be grayed out if there is no data in the currently active document.

    d. Copy: Allows copying into the clipboard of the entire wave using a custom clipboard format. This allows applications that receive that custom format to paste it into them. This should be grayed out if there is no data in the currently active document.

    e. Copy As Bitmap: Allows copying the entire wave into the clipboard using the Bitmap format. This allows applications like Word/Visio to paste the bitmap into their documents.

    f. Paste: Allows pasting of any wave data (Your previous custom clipboard wave format) present on the clipboard **in place** of the current wave document. This menu item should be grayed out if no Wave data is present on the clipboard.

    g. Delete: Allows clearing of the entire wave. This item should be disabled if the document is empty.

View Menu

20. Toggle menus for viewing/hiding Status strip and Tool strip.

21. A Toggle menu item for viewing the wave either in standard mode or Full Mode.

    a. Standard Mode: means each sample point maps to a single pixel. In all examples given the number of samples exceed the window size, so a scrollbar should be provided to scroll through the data.

    b. Full/Zoomed mode, which allows viewing of the entire wave within the window. No scrollbars should be present.

Format Menu

22. Menu items that allows changing the Wave foreground and background colors and pen thickness.

Tools Menu

23. A menu item as well as a toolbar button should be available to play the currently loaded wave file, if not modified. If the document has been modified, the user should be prompted to save the file first using a Message box with OK🡐🡒 Cancel. If the user chooses OK, then the file will be saved and the sound will be played. If the user chooses Cancel, the file should neither be saved nor a sound should be played.

24. Two menus as well as toolbar buttons should be available to allow the use to do the following operations:

a. Modulate the wave by modulation function you desire. This menu item should be grayed out if there is no data in the currently active document. Use any computation that visibly changes the wave. I have used a weighted hyperbola for simple modulation:
   - Sample[i] = Sample[i] * i / NumSamples;
   - or
   - Sample[i] = (byte) (Math.Sin(i+3.2f)*20 + Sample[i];

b. Rotate the wave with respect to the y-axis; e.g. ---^^^--vv--- 🡒 ---vv--^^^---. This menu item should be grayed out if there is no data in the currently active document.

Windows Menu

25. Tile Horizontally
26. Tile Vertically
27. Cascade
28. List of all MDI windows Open.

Help Menu

29. The Help About menu should launch a centered dialog box that contains the following controls:
    a. A picture control that shows some arbitrary bitmap.

    b. A Link Label that displays your URL of choice. When the user clicks on the label, internet explorer will be launched with the set URL.

    c. A Close button that closes the dialog.

Toolbar

30. Mimic the Buttons in the Sample. This makes everyone's project consistent and simplifies my Grading.

Context Menus

31. Mimic the context menus in the Sample. This makes everyone's project consistent and simplifies my Grading.

The File View Control

32. Two menu items 1- Font…, 2-Background Color (under Options menu) and toolbar buttons should be available to allow the user to change the Tree's text font/color using the Font common dialog, and to change the background color using the color common dialog.

33. When a file is deleted, renamed or added within the displayed directory (using windows explorer or any other mechanism), the corresponding file name should automatically be removed, renamed, added from/to the FileView. This required that you set a system file watch notifications.

User Settings

34. All configurable settings should be persistent in a .config/.xml file
    a. The Background, Wave color and thickness.
    b. The Font and Color of the Text and background of the FileView TreeView.
    c. The last set of files that were open when the application was closed.

Extreme Caution

Make sure you test your application on a separate computer or at the very least unzip to a completely different folder and test. A lot of the common problems are due to using hardcoded absolute or relative paths or the assumption that a .config/.xml file exists.

Additional Optional features:

35. Any additional features that you desire to add should be listed in a form under the Help menu: call it Help > Extra. I will give extra credit for any additional features. I will use my judgment in how many points the extra features deserve.