Multitrack Waveform Viewer implemented in JavaFX

The goal oft the project is to create multitrack waveform viewer in JaxaFX

The implementation shall be done in Java with JavaFX. If possible be compatible with JavaFX 11.

Part of the implemtation are

- One panel showing all tracks
- Tracks being grouped in unlimited deep
- On the right side, waveform view (or other time related data graphical view upon provider class)

The main functionalities are

- Multiple Tracks based on data source <time, float> or <time, long>
- Timeformat 64 bit or 96 bit seconds.nanoseconds
- Datasource reloadable by interface callback
- Group tracks into Groups, subgroups etc.
- Zoom in and out
- Scroll (both directions)
- Adjust track height
- Select by mouse
- Multiselect
- Callback for all actions, mainly reload data, selection change
- MVC approach with Model, Controller and view

Datasource

Define a good approach to provide data with time and content. The Waveform viewer checks if the data is available (with limits). If not the data is reloaded via a Callback.
The Datasource shall be float or long.

Layout and View

As to attached sheet the layout consist of

- General
  - All elements shall be configurable freely regarding colors and fonts (also font size etc). Can be done in several ways, with CSS classes or by grabbing element properties.

    - One class to hold all information. Groups, Subgroups and Track with corresponding manipulation methods (model)
    - Implement this as node. Shall be usable on a pane
- Tracks
    - Each track has a (not showed) identifier
    - Use a theoretically free layouted pane on the left for all track information and properties
    - Forsee name, a collapse button, a color chooser, an enable / disable marking checkbox and a value indication label
    - Button to hide track view and enlarge waveformview
- Groups
    - Can be created with a right click context menu
    - Subgroups can be similar generated by clicking into a group
    - Tracks can be moved with drag and drop out and into groups and subgroups. Find a good way where to let drop the track and how to inticate to position
- Waveform view
    - Possibility for different implementations. Obviously now only one will be done. In future may other data also is shown. Define in the track a viewprovider or similar.
    - Color for the waveform settable in the track
    - Adjust the view regarding to the zoom
    - Bonus – implement a spline interpolator when zoom under one sample view!
    - Dickness of line adjustable
    - Neon or shadow effect (optional)
    - Drag to adjust height of track
    - Zoom in and out via seperate bar
    - Zoom in / out via Alt+Mouse wheel
    - Scroll left and right with Ctrl+Mouse wheel
    - Scroll up and down with mouse wheel
    - Context menu on track with
        - List of selections
        - Add / clear selection
        - May other functionalities in future
    - Selection by mouse
        - The the user scroll left and right while selectring
        - Two modes
            - Either select over all tracks
            - Allow totaly individial selections on each track
            - Provide a list with track identifier and start / end time
    - Multiple selection via list
    - Allow clearing selection via a method call (done by a button then in reality)
    - Time Base Line where the mouse is. Leads to value indication in the track view
    - Popup window for one track with time and data on specific time / track mouse position
    - Callback for all events. Redraw after having called manipulated tracks
    - Method to update data continously (when recording)
- Time scale on top and on bottom
    - Indicate the time with lines

Store configuration / arrangement into XML and JSON

- Serializer and Deserializer functionalty to store the whole arrangement configuration (tracks with config, in what group, subgroups, track heigth etc.) Data is stored SEPERATELY!
- Do it generic to allow later on add more parameters

Example project

To demonstrate and verify the functionality implement a sample application

- Having multiple tracks
- Reading for these tracks multiple wavefiles (can be fix coded a ArrayList with the files)
- Code the callback reload for the wavefile to demonstrate the instant reload of data
- Implement two label to demonstrate the callback functionality
  - Showing the mean value of the current selection of one track
  - Showing the mean value of the current selection of all tracks
- Implement the possibility for individual selection (each track individually)

Key aspects of the project

- Proper waveform drawing
- Full resize / movement / selection capabilities by mouse and by list (multi selections)

Use standard components for buttons, scrollbars etc. Graphical fine adjustment will be done via CSS.

The project must not rely on any open source software / other licenses! Implement the functionaly from top to bottom by your own. (With exception Spline interpolator algorithm)

Specification checklist

The application is written in Java and uses JavaFX for all GUI implementation	
The application should be compatible with JavaFX 11	
No open source code usage is allowed!	
Datasource structure holding time and samples. Methods for all needed manipulations. Reload data, manipulate data etc. Possibility for float and integer values Time format must be seconds.nanoseconds to fit 64 bit and 96 bit time format for PTP and similar applications	
Track class or similar to hold all the track information, including also the Datasource object	
A data structure to hold all tracks. Methods to add and remove tracks to.	
One panel implementation having on the left all tracks, on the right the corresponding waveform view	
The waveform shall implement an interface to also have substitution capabilities. Other approach than Interface also welcomed.	
Wavefrom scroll and zoom abilities. Zoom in / out, scroll left / right and up /	

down.	
Adjust zoom when parent is resized (to still draw the same time range).	
The track list also can have groups and subgroups (infinit order theoretically). They are added via methods. A context menu shall be implemented already doing this (add / remove group)	
On top and may also on the bottom there is a timeline indicating the base times for the corresponding position.	
Time Base Line where the mouse pointer is. In each track view the corresponding value ist shown (of that time)	
Track popup window. On mouse position, indicating the time and data value (as to drawing). Optionally enable / disable	
Resizings Vertically between the track view and the waveform view Each track in height Track collapsing. Only showing the name but no more data. Also no data reload etc. is processed on «closed» / collapsed tracks.	
Choosable selection mode. All tracks / only selected tracks (selected with a checkbox in the track view) One start / end, multiple start end (4 combinations though) In this case all tracks / the selected ones have the SAME start and end positions Or Totally individual selection on each track with multiple start end. In this case all tracks may have INDIVIDUAL start and end positions	
Provide a list of tracks and their selection (start and end sample indexes or specific timestamp). Homogenous for all selection cases.	
De- and Serialising of all configuration data, including the layout data and track heights etc. Data Source although is held seperately!	
Use expressive Method and Class names. For example use insted of UpdateData UpdateDynamicPropertyListData Thus keeping the code selfexplaining	
Keep methods short and strait forward	
Comment all methods with input / output and action	
Where important logic hold place (view.x-screen.x+option.maxsize-….) explain in short what is done. Again expressive Names for methods and variables already do a lot!	

Proposed project steps

Setup the datasource structure. Mainly also with functions to check if the needed data in the defined time is available. This shall be part of the model.	8

Preproject • Implement waveview to assure proper math to show waveform • Basic Layout container to show concept of arranging the elements (track list with groups and waveforms) • Sample Drag and Drop actions on nodes to assure elegant coding for this	10
Implement the track and group view	7
Implement the waveform view (yet without mouse handling). Selection, zoom in, track heigth change can be demonstrated with manual inputs	25
Bonus spline view for full zoom in. Real samples must be distinguished from interpolation. Form and color wise	3
Implement Group and Subgroup creation. Track regrouping via drag and drop	6
Implement track height change, zoom in/out (mouse wheel) selection by mouse Scroll by mouse (left/right and up/down) Time Base moving line Track Popup window with timestamp and value Track context menu	15
Implement the demonstration software as described above (Do this in parallel to provide test abilities for each milestone)	10
Create a de-/serializer for the current layout (group and track arrangement, selections, zoom factor, start position) to XML and JSON	7
Sum	91 hours

Based on $40 per hour the project costs are assumed to $3640.00