CSC 561 : Assignment 2

Name: Jyoti Sheoran

Student id : V00801666

Project Design Document:

High level Overview:

JMediaPlayer Class:

- Implements User Interface (buttons, panel, and container).
- Contains classes to listen to events (button clicked and slider move).
- Implements controllerListener interface's controllerUpdate() method.

Details:

1. Main() method :
   - Creates an instance of JMediaPlayer class.
   - Initialize the UI and sets default size and location.
2. Control_Start class:
   - Initialize the player.
   - Set slider to far left end. (i.e value = min)
   - If Player is started for the first time, then start media from time 0:0, else start media from last paused time.
   - Start the timer.
3. Control_Stop class:
   - Player is stopped. (i.e. player go back to prefetching state).
   - Set isStopped to true.
   - Stop the timer and set slider to far right end. (i.e. value = max)
   - Deallocate the player. (i.e. Player go back to unrealized state).
4. Control_Pause class:
   - Player is stopped. (i.e. player go back to prefetching state).
   - Stops the timer.
   - Retrieve media duration and current mediaTime (i.e. how much of media is played).
   - Set isPaused Boolean variable to true.
5. Control_Forward:
   - Increases the rate at which media is played (upto 3x) with 0.5 increment steps. If rate is below 1, then it sets the rate to 1.
   - Also, prints the current rate (after setting the rate) to console.

6. Control_Backward:
    - Decreases the rate at which media is played (upto 0.1) with -0.2 increments if current rate is 1 or lower, else it sets it to 1 (if rate is above 1).
7. Control_Slider:
    - Implements Slider actionListener.
    - If slider value is changed, then update the MediaTime of player and also update the timer to reflect new MediaTime.
    - It calls updateTimer() method to update timer and player mediaTime.
8. Control_Timer:
    - Implements timer actionListener.
    - It calls updateTimer() to update the timer and slider value.
9. Control_Open:
    - Calls FileOpen() method to open a file (by opening a file explorer).
    - Resets the slider value to zero.
10. ControllerUpdate:
    - This method is called whenever there is a state change in player.
    - If player has visual component, then it adds it to the content and resize the window accordingly.
11. updateTimer:
    - Gets current media time, media duration and slider_value. It then updates slider position and timer accordingly.
    - It stops the timer and resets slider to max value, if media is completed.
12. Init:
    - Creates a player by calling Manager.createPlayer() method using the given file_location.
    - It then initialize various Boolean values and sets slider & timer to zero.
13. FileOpen:
    - Opens the File explorer to choose a file.

Player Controls:

1. Start:

Player is started by calling player.start() method. This method automatically tranisiton the player to a Started state (by calling realize -> prefetch -> clock.syncStart()).

2. Stop:
    Player is stopped by calling player.stop() which changes the state of player to prefetched. After this, player.dealocate() is called to change Player's state to realized.

3. Pause:

   Player is stoped by calling player.stop(). This changes the player's state to prefetched. Player's current mediaTime is stored in a variable to be used later when start button is clicked.
4. Forward:

   First we retrieve the current rate of player by calling player.getRate(). We then increment the rate by 0.5 if current rate is less than 3.0
5. Backward:

   We first retrieve the player's current rate. If current rate is greater than 0.2, then we decrement it by 0.1 or 0.2 or 0.5 depending on its current value. The rate is set by calling player.setRate() method.
6. Slider:

   When slider is moved, we retrieve its current value. Player is stopped from playing the media (by calling player.stop()). Then we set player's mediaTime equivalent to slider-value and restart the player (by calling player.start()).