

## **OOM Mini Project # 2: Creation of simple Media Player Software**

### **Abstract**

*JMF offers several mechanisms for playing media. The simplest is using objects that implement interface `Player` declared in package `javax.media`. Package `javax.media` and its subpackages contain the classes that compose the JavaMedia Framework. To play a media clip, you must first create a `URL` object that refers to it. Then pass the `URL` as an argument to static method `createRealizedPlayer` of class `Manager` to obtain a `Player` for the media clip. Class `Manager` declares utility methods for accessing system resources to play and to manipulate media. Declaration of a `JPanel` demonstrates some of these methods.*

*The constructor sets up the `JPanel` to play the media file specified by the constructor's `URL` parameter. `MediaPanel` uses a `BorderLayout`. Inside the `try` block we can invoke static method `createRealizedPlayer` of class `Manager` to create and realize a `Player` that plays the media file. When a `Player` realizes, it identifies the system resources it needs to play the media. Depending on the file, realizing can be a resource-consuming and time-consuming process. Method `createRealizedPlayer` throws three checked exceptions, `NoPlayerException`, `CannotRealizeException` and `IOException`. A `NoPlayerException` indicates that the system could not find a player that can play the file format. A `CannotRealizeException` indicates that the system could not properly identify the resources a media file needs. An `IOException` indicates that there was an error while reading the file. These exceptions are handled in the `catch` block.*

*A method `getVisualComponent` of `Player` can be invoked to get a `Component` that displays the visual (generally video) aspect of the media file. A method `getControlPanelComponent` of `Player` to get a `Component` that provides playback and media controls can be invoked. These components are assigned to local variables `video` and `controls`, respectively.*

### **Specific Technology**

Java, Swing, JSF, XML.

### **Project Tasks**

Use case analysis, Design: Class Diagram, CRC and Implementation, User Interface Functional components of the project

- A positioning slider to jump to certain points in the media clip.
- A pause button.
- A volume button that provide volume control by right clicking and a mute function by left clicking.
- A media properties button that provides detailed media information by left clicking and frame rate control by right clicking.

### **Submission**

The project presentation has to be done by each member to show periodically the progress and the complete project submission should contain the following:

-> UML diagrams:-Use Case diagram, Class diagram, CRC diagram(s), illustrating the design of your program.

-> All the Java source code should be necessary to compile and execute.

#### Video links:-

**Format:- [link][channel name][purpose of video]**

- 1) <https://www.youtube.com/watch?v=7GdxI2045I8> (Awais Mirza)(basic model)
- 2) <https://www.youtube.com/watch?v=LWM1G7gLTXY> (Ali Boukhachem)(using swing)
- 3) <https://www.youtube.com/watch?v=lg5zWJTQWx8> (Mahmoud Hamwi)(using swing)
- 4) <https://www.youtube.com/watch?v=TErboGLHZGA> (Max o'Didili)(how to play,pause,stop)
- 5) <https://youtu.be/ceXHH9sYlkk> (Assembly)(basic model)
- 6) <https://www.youtube.com/watch?v=GKiHB5AzhE> (SIMPLECODE)(play YT videos in JFrame)
- 7) <https://www.youtube.com/watch?v=sjiS4mhb0gQ&t=49s> (programming knowledge)(basic model)

#### Source codes:-

**Format:- [link][creator]**

- 1)

#### Other links:-

**Format:- [link][purpose of the link]**

- 1) <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/> (UML Use Case Diagram)
- 2) <https://stackoverflow.com/questions/35958661/how-to-get-all-files-from-a-directory-using-javafx> (getting all the files from the folder)

**Features:-**

- |                        |        |
|------------------------|--------|
| 1) Basic Layout        | --Done |
| 2) Play/pause          | --Done |
| 3) Choose file         | --Done |
| 4) Fast_forward/slowMo | --Done |
| 5) Main Slider         | --Done |
| 6) Volume/Mute         | --Done |
| 7) Info                |        |

**TO DO:-**

- 1) get the information of the media

## Use Case Diagram :-



