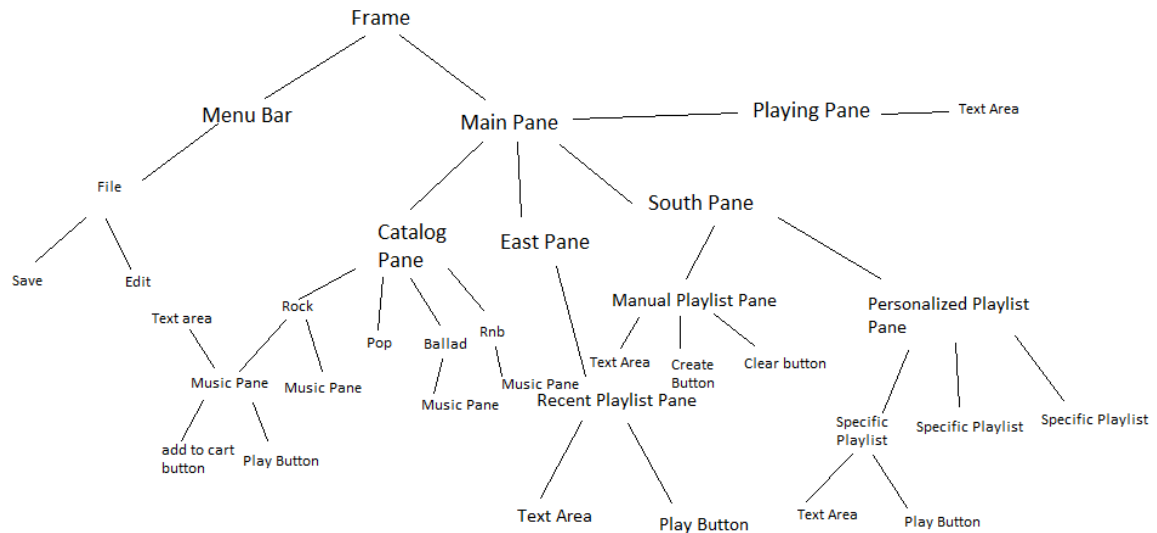# Milestone three:

GUI component tree structure :

To start, I have a Frame that will come to contain everything. This frame is divided between a Menu Bar that will contain a menu "File" and two menu Items "Save" and "Exit" that when clicked would either make us able to save the data that we have or exit the interface all together. The frame also contains a Main Pane that will come to contain all the other components. In fact, it is divided between a Catalog Pane in the middle, a Playing Pane at the top, an East Pane on the right side and a South Pane on the south. To achieve this, we will make use of the Border Layout. The Playing Pane has a Text Area that gets updated automatically whenever a song is played. The Catalog Pane is a tabbed pane compozed of the different music style and each one contains a Music Pane. A music pane has an add to cart button and a play button that plays the tracks in the recent playlist as well as a text area. The east pane is made out of a recent Playlist Pane that has a text Area and a play Button. A south Pane has a manual playlist pane with a text area, a clear button to delete the text area and a create button. Each time the user clicks on the create button a new Specific Playlist is created. The south Pane also has a personalized Playlist and is tabbed with the different Specific playlists creates which each contain a text area to display and a play button to play the tracks in the playlist created.

Updated class diagram :

Displayable

+abstract display()

Group

-members: List<Member>

+Group()
+addMember(Member member):void

AlbumConcert

- title:String
- picturePath:String
- description:String
- date:String
- numberOfTracks:int
- list:MusicList
- i:int

+AlbumConcert()
+update(String title, String picturePath, String description, String Date, int numberOfTracks):void
+addTrack(Music music):void
+getName():String
+getDescription():String
+getNumOfTracks():int
+getDate():String
+display():abstract String
+play():void

Artist

+ Artist()

Author

-name:String
-imagePath:String
-bio:String
-singles:MusicList
-albums:List<Album>
-concerts:List<Concert>
-i:int=0
-MAX: final int= 100

+Author()
+update(String name, String path, String bio):void
+addAlbum(Album album):void
+addSingles(Music music):void
+addConcert(Concert concert):void
+getNameAuthor():String
+display():String

List<T>

-head:Node<T>
-last:Node<T>
-count:int

+MemberList()
+addMember(Member member)

Node<T>

+value : T
+next:Node<T>

+MemberNode(member Member)

Member

-name:String
-bio:String

+Member()

Album

+Album()
+display():String

Concert

-location:String

+Concert()
+updateLocation(String location):void
+display():String

IllegalDurationException

+IllegalDurationException()
+getMessage()

Music

-title:String
-style:Style
-duration:float
-author:Author
-path:String
-concert:Concert
-album:Album
-type:String
-recent:Recenplaylist

+Music(String title, float duration, String path, String type, RecentPlaylist recent, Style style)
+update(String name, float duration, String path, String type, Style style):void
+display()
+setAuthor(Author author):void
+setAlbum(Allbine album):void
+setConcert(Concert concert):void
+type():void
+play():void

MusicList

-head:MusicNode
-last:MusicNode
-size:int
-previous:MusicNode
-walker:MusicNode

+MusicList()
+add(Music music):void
+toArray():Music[]
+delete(Music music):void
+empty():void
+search(Music music):boolean
+display():void

MusicNode

+node:MusicNode
+next:MusicNode

+MusicNode(Music music)
+getMusic():Music

CatalogStyle

-styles:Style[]
-MAX:static final int
-i:int

+CatalogStyle()
+addStyle(Style style)

Style

-name:String
-list:MusicList

+Style(String name)
+addToStyle(Music music)
+getName():void

RecentPlaylist

+RecentPlaylist()
+play():void

Playlist

#list:MusicList

+Playlist()
+add(Music music) abstract void
+play(): abstract void

ManualList

+name:String

+ManualPlaylist()
+add():void
+deleteMusic(Music music):void
+deletePlaylist():void
+play()

ManualPlaylistPane

+ manual : ManualPlaylist
- deleteButton : JButton()
-recent: RecentPlaylistPane
-createButton : JButton
-personal :
PersonalizedPlaylistPane
-name : string

+ManualPlaylistPane(RecentPlaylistPane recent, ManualPlaylist manual, PersonalizedPlaylistPane personal)
+update():void
+delete():void

AbstractMusicPane

pane : Jpanel
textArea : JTextArea

+AbstactMusicPane(String Title)
+update(): abstract void

CatalogPane

- recent : RecentPlaylistPane

+CatalogPane(CatalogStyle catalog, ManualPlaylisPane manual, RecentPlaylistPane recent, PlayingPane playing)

RecentPlaylistPane

-music:Music
-recent : RecentPlaylist
- playButton :JButton

+RecentPPlaylistPane(RecentPlaylist recent)
+update(): void
+play():void

PlayingPane

-current : Music

+PlayingPane()
+getMusic():void
+update():void

MusicPane

-music:Music
-musicTextArea:JTextArea
-addToPlaylist : JButton
-playButton:JButton
flow : FlowLayout
-manualP : ManualMusicPane
-recentP: RecentMusicPane
-manual : ManualPlaylist
-recent : RecentPlaylist
-play:PlayingPane

+ operation1(params):returnType
- operation2(params)
- operation3()

PersonalizedPlaylistPane

-name:String
-elementsPane:JPanel

+PersonalizedPlaylistPane()
+add(ManualPlaylist manual, String name, RecentPlaylistPane recent):void
+update():void

MainFrame

-mainPanel : JPanel
-eastPanel : JPanel
-southPanel : JPanel
-catalogPane : CatalogPane
-recentPane:RecentPlaylistPane
-manualPane :
ManualPlaylistPane
-playingPane : PlayingPane
-personal :
PersonalizedPlaylistPane

+MainFrame(CatalogStyle catalog, RecentPlaylistPane recent, ManualPlaylistPane manual, PlayingPane play)

SpecificPlaylistPane

-manual:ManualPlaylist
-name:String
-recent:RecentPlaylistPane
-playButton : JButton
- flow : FlowLayout
-textArea : JTextArea

+SpecificPlaylistPane(ManualPlaylist manual, String name, RecentPlaylistPane recent)

MenuBar

-menufile: JmenuFile()
-menuFileSave: JMenuItem
-menuFileExist : JMenuItem

#MenuBar()