## Music Band Management System / Music Label

A Music Labeling company is interested in a program that allows them to better manage their data, ranging from their Artists and Bands to their Albums and Live Concerts.

The Music Label has an official name, an address, a declared annual revenue and a list of its Artists and the Albums they have produced. The label can sign new contracts with recently discovered talented Artists, create an Album with one or more Artists, modify the Album Artist composition until its finalization and release Artists from their contracts if these are not respected. The company also has the possibility create a new Band from its contracted Artists. Finally, the music label is keeping a close eye on its investments and the revenue these generate, thus it manages large amount of money.

Every employed Artist has a given salary, his real name and a stage name, the letter being able to correspond to the real name, but it is not necessary. As Artists grow in popularity or start their career with some friends at their side, they may form a Band. A Music Band has a unique name, an origin, and a formation date. As it is composed of Artists, these can easily be added or removed from the Band, especially when they intent on starting a new solo career, continuing it or going to a job transition.

The key to any Artist's or Band's success is a great Album. This represents the collective effort of a group of Artists or individual creation example, which has an inspiring title, a list of Plays, and an official language in which it is sung. Every Album release has a well calculated strategy to compensate financially the Artist(s) and the producing label, so it comes with a total production cost, a base price per disc and the number of copies sold.

Every Play is part of Music Genre. Every Genre has a name, a recent rating based on listener count and an (approximate) apparition date. Some of the most distinct Music Genres are: Classical Music, Pop Music, Rock Music, and Rap Music, each differentiating itself from the other by certain unique characteristics.

Finally, an important time of an Artist's career is represented by the Concerts he is playing at, thus the Music Label made sure to organize a series of Concerts for its hardworking Artists. Locations have been found, dates determined and advertised in order to have an as large as possible audience, without exceeding the locations capacity. After the Concert, the profit will be calculated, by multiplying the number of tickets sold by one ticket's price and subtracting the costs the Music Label covered by renting the location and, possibly, the equipment.

## Song -name: String -rating: String -apparitionDate: Date +Song(name: String, rating: String, apparition: Date)

Classical	Pop	Rock	Rap
-relatedSongs:	-relatedSongs:	-relatedSongs:	-relatedSongs:
List <song></song>	List <song></song>	List <song></song>	List <song></song>
+Classical(name: String,	+Pop(name: String,	+Rock(name: String,	+Rap(name: String,
rating: String, apparition:	rating: String, apparition:	rating: String, apparition:	rating: String, apparition:
Date)	Date)	Date)	Date)

<b>Album</b> – implements Profitable		
-title: String		
-artistList: List <artists></artists>		
-playsList: List <song></song>		
-language: String		
-productionCost: Float		
-copiesSold: Integer		
-copyPrice: Float		
+calculateProfit()		

Band		
-name: String		
-formationDate: Date		
-origin: String		
-artistList: List <artists></artists>		
+add(artist: Artists)		
+remove(artist: Artists)		

<b>Concert</b> – implements Profitable		
-artistsList: List <artists></artists>		
-location: String		
-date: Date		
-capacity: Integer		
-ticketPrice: Float		
-ticketsSold: Integer		
-rentCosts: Float		
+calculateProfit()		

Artists	
-name: String	
-stageName: Stri	ng
-salary: Float	

Music Label
-name: String
-address: String
-revenue: Float
-artistList: List <artists></artists>
-albumList: List <album></album>
+addArtist(artist: Artists)
+createAlbum(artistList:
List <artists>)</artists>
+modifyAlbum(album:
Album)

Controller		
+authenticate(username: String,		
password: String)		
+showArtists()		
+showAlbums()		
+showAlbumsForArtist(artist: Artists)		
+sortAlbumsByRevenue()		
+showUpcomingConcerts()		

Admin	User
	-favourites: List <song></song>
+addArtist(artist: Artists)	+addFavourite(melody: Song)
+createAlbum(artistList: List <artists>)</artists>	+removeFavourite(melody: Song)
+modifyAlbum(album: Album)	+showFavourites()
+showConcerts()	+buyTicket(concert: Concert)
	+showRecommended()