

Requisiti. L'applicazione da progettare riguarda la gestione di playlist di brani musicali in formato elettronico. Una playlist è caratterizzata dal nome (una stringa che rappresenta il nome del file nel quale la playlist viene memorizzata), da un insieme non vuoto e ordinato di brani musicali, e da un'immagine (che viene usata come sfondo durante l'esecuzione della playlist). Un brano musicale è caratterizzato da un nome (una stringa che rappresenta il nome del file nel quale è memorizzato il brano) e dal genere musicale del brano stesso (una stringa, per esempio "rock", "country", "classica", ecc.). Una immagine è caratterizzata da un nome (una stringa che rappresenta il nome del file nel quale è memorizzata l'immagine). Alcune playlist sono "speciali" in quanto possono associare ad alcuni brani degli effetti di equalizzazione speciali. Gli effetti stessi sono identificati da un codice numerico (un intero). Le playlist speciali hanno un titolo (una stringa) che viene visualizzato quando vanno in esecuzione.

Una playlist può essere in tre stati "non-attiva", "in-esecuzione", "in-pausa". Quando è non attiva può essere messa in esecuzione con il comando "play"; quando è in esecuzione può essere messa in pausa con il comando "pause" o può essere messa nello stato non-attiva con il comando "stop"; quando è in pausa può essere rimessa in esecuzione con il comando "play". La playlist può essere modificata (aggiungendo o eliminando brani musicali, ecc.) solo quando è nello stato "non-attiva".

L'utente del sistema è interessato ad effettuare diversi controlli. In particolare:

- data una playlist p ed un genere musicale g , generare in modo automatico una nuova playlist p_0 con la stessa immagine di sfondo di p e solo brani musicali del genere g presenti in p (in un ordine qualsiasi);
- dato un brano musicale b restituire il numero di playlist speciali in cui b è presente.

Domanda 1. Basandosi sui requisiti riportati sopra, svolgere la fase di analisi producendo lo schema concettuale in UML per l'applicazione e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

Domanda 2. Svolgere la fase di progetto, illustrando i prodotti rilevanti di tale fase e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

È obbligatorio solo progettare gli algoritmi e definire le responsabilità sulle associazioni.

Domanda 3. Svolgere la fase di realizzazione, producendo un programma JAVA e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

È obbligatorio realizzare in JAVA solo i seguenti aspetti dello schema concettuale:

- le classi `Playlist` e `BranoMusicale`, e tutte le associazioni che le legano;
 - il primo use case.
-