

## Sujet

Il s’agit de créer un logiciel de gestion d’un ensemble de morceaux de musique au format “MP3<sup>1</sup>”. On trouvera ci-dessous les contraintes à respecter, en particulier les fonctionnalités minimales à implémenter.

### Fonctionnalités attendues

- Votre programme doit posséder une interface graphique (essayez de la rendre la plus intuitive possible).
- Cette interface doit être séparée du reste du programme (classe spécifique). Une version “textuelle” - à base de menus simples - doit être disponible pour tester l’application indépendamment de l’IHM.
- L’application doit permettre de trouver un/des morceau(x) de musique satisfaisant un ensemble de critères donnés (genre, durée, interprète etc.). L’IHM doit faciliter la formulation des critères de recherche.
- Elle doit également permettre la création, la sauvegarde et l’utilisation de playlists (fichiers texte d’extension “.m3u”).
- Lorsqu’un morceau est sélectionné à la souris, l’IHM doit afficher ses caractéristiques (tag ID3, durée, date, bitrate, fréquence d’échantillonnage etc.).
- Les fichiers audio sont censés se trouver dans une arborescence de répertoires. L’application doit être capable d’explorer cette arborescence en ne connaissant au départ que l’emplacement de la racine (par exemple `/home/moi/music`). Cet emplacement fait partie des paramètres de configuration de l’application.
- Tous les paramètres de configuration de l’application (*look-and-feel*, racine, limites diverses...) doivent avoir une valeur par défaut. Si la valeur choisie diffère de la valeur par défaut, un fichier de configuration devra pouvoir être généré sur demande de l’utilisateur.

Bien entendu, il vous est possible d’implémenter davantage de fonctionnalités (système de notation, compteurs de nombre d’écoute, gestion de choix aléatoire de morceaux parmi un ensemble quelconque - album, genre, interprète...). Celles-ci devront être décrites dans le document de synthèse (voir plus loin) et pourront donner lieu à un bonus. En particulier, il serait judicieux de coupler l’application avec un player audio!

## Travail à rendre

Le travail est à réaliser en binôme (groupes de TD). Vous devez fournir

- Les sources commentés de votre application.
- La javadoc générée par les commentaires de votre code source (ensemble de fichiers HTML).
- Un document expliquant comment installer et utiliser (2 parties donc, manuel d’utilisation + guide de l’utilisateur) votre application. Ce document devra être au format PDF et livré avec son document “source”, rédigé avec  $\text{\LaTeX}$  ou  $\text{\LyX}$  (une sur-couche facilitant l’utilisation de  $\text{\LaTeX}$ ). Si le document contient des images, elles devront être fournies, car on doit pouvoir reconstituer le PDF à l’identique à partir des éléments fournis.
- Un *résumé* du projet en anglais, d’environ une page, au format PDF également.

L’ensemble des sources du programme et les documents devront être placés dans une archive<sup>2</sup> (format `tar` (compressé ou pas) ou `zip et pas autre chose!`) envoyée par mail à votre responsable de TD (lui demander une adresse mail valide à laquelle il souhaite recevoir les archives, par ex. `marc.lemaire@u-cergy.fr`), avec copie à l’adresse `laroque@u-cergy.fr`. Pensez à demander un accusé de réception. Le nom de l’archive doit contenir le nom des deux étudiants formant le binôme. Par exemple on pourrait avoir une archive nommée `MLemaire-PLaroque.tgz`

<sup>1</sup>MPEG-1 audio layer 3. Description sur wikipedia - entre autres.

<sup>2</sup>On rappelle qu’une archive bien faite crée un sous-répertoire pour extraire son contenu et ne doit pas mélanger les fichiers extraits avec ceux du répertoire courant!

## Echéances

- Les archives doivent parvenir à votre responsable de TD au plus tard le *jeudi 18 décembre 2008*. Tout retard dans l'envoi sera pénalisé d'1/2 point par jour, sans excéder 5 points.
- Le travail fera également l'objet d'une soutenance avec diaporama de 10 minutes (suivi de questions pendant environ 5 minutes), la date dépendant de votre responsable de TD (voir avec lui, probablement à la rentrée de janvier).

## Barème

- Document : 4 points (attention à la présentation, au niveau de langue et à l'orthographe)
- Respect des formats et du timing : 2 points
- Soutenance : 3 points
- Fonctionnalités : 4 points
- Qualité du code et des commentaires : 3 points

Le reste, donc 4 points, est attribué par votre responsable de TD sur la base de votre niveau de participation lors des séances précédant le projet.