



---

# Générateur de musique interactif

---

Mon magasin d'instruments de musique *Musicalau* veut toucher un plus large éventail d'acheteurs. Les instruments sont chers et prennent souvent de la place. J'aimerais pouvoir proposer une application, accessible pour des budgets plus modestes, qui permette de faire de la musique.

Cette application (en console) doit donner la possibilité de choisir l'instrument que l'on souhaite utiliser et l'afficher.

Il faudra avoir au moins le choix entre 3 instruments : un piano, un xylophone et une guitare.

Le programme devra permettre :

- De jouer des **notes**.
- De pouvoir choisir le **rythme** de ces notes (lent, normal ou rapide).
- De lire un fichier de partition (format donné).

## Innovation :

Les éléments demandés dans ce projet ne constituent pas l'intégralité de la note. Une partie sera notée sur les apports que vous ferez (idées : ajout d'instruments, enregistrement, séquence aléatoire dans une gamme donnée, etc...).

A vous d'être force de proposition !

## Conseils :

Dans un premier temps, ne vous occupez pas de générer réellement du son.

Réfléchissez à l'architecture de votre code, puis codez le squelette.

Vous pouvez très bien avoir dans un premier temps des touches qui affichent « do », « la », etc... en console plutôt que d'avoir le son réel.

Le plus important est que vous ayez une structure orientée objet !

### ★ Attendus techniques :

Le but étant de faire un projet de POO, votre code doit respecter ces contraintes :

- Il faut qu'il y ait des **classes**
- Il doit y avoir de l'**héritage**
- Il doit y avoir du **polymorphisme**
- Pour ne pas avoir une fonction *main()* qui soit géante il faudra essayer au maximum de structurer votre code à l'aide de fonctions. Dans l'idéal, il ne doit pas y avoir de code identique à différents endroits.
- Il faut que votre code soit découpé correctement entre fichiers d'en-tête et fichiers source.



★ Délivrables attendus :

- ★ Code (sous Git à me partager dès le début du projet !!!!).
- ★ Présentation PowerPoint (soutenance)

★ Grille d'évaluation

Présentation	Qualité de la présentation : clair, compréhensible, timing respecté, support de présentation.
	Justification des choix d'architecture qui ont été faits.
	Répartition de la parole dans le groupe.
Démonstration	Démonstration préparée.
	Démonstration qui fonctionne.
Technique	Le code est fourni tout au long du projet : accessible via Git.
	Les fonctionnalités demandées sont présentes.
	Le code est de qualité.
	Le code est documenté.
Soft Skills	Respecter les conditions et modalités de rendu.
	Répartition/organisation du travail dans le groupe.
	Prise de recul sur le projet : esprit critique sur son travail, propositions pour la suite, analyse des décisions, etc.