

## Séquenceur novembre

Je m'étais fixé comme objectifs pour ce mois de novembre de mettre en place un premier prototype initial. Celui-ci devait permettre la lecture d'un sample audio et disposer d'une interface graphique simple avec les commandes de base, conformément au planning prévisionnel.

Cependant, grâce au travail réalisé en octobre (choix des bibliothèques, acquisition des assets et début du développement d'un prototype fonctionnel en avance), j'ai abordé ce mois avec une base déjà fonctionnelle.

Cette avance m'a permis de ne pas me limiter à un « simple test de fonctionnement ». J'ai pu utiliser l'avance prise en octobre pour créer une version déjà bien organisée, avec plus d'options et bien plus complète que ce que j'avais prévu au départ pour ce mois de novembre.

## Développement du prototype

Le prototype initié en octobre (lecture d'un son simple via un script) a servi de fondation. L'objectif principal pour novembre était d'embellir ce code en y ajoutant l'architecture modulaire définie dans le cahier des charges.

J'ai procédé à une refonte complète en POO (initialement il n'y avait qu'un fichier prototype à la racine du projet) séparant clairement les responsabilités :

- MoteurAudio : Gestion optimisée de la lecture et du volume (sounddevice).
- SequenceurCore : Cœur logique gérant les objets « piste ».
- FenetrePrincipale : Interface utilisateur réactive (PySide).

## Avance prise sur les objectifs de décembre

Le prototype existant déjà en début de mois, le temps initialement alloué à sa création a été réinvesti pour anticiper les tâches de décembre (« Implémentation de plusieurs pistes »).

Le logiciel gère désormais une liste dynamique de piste. Celle-ci sont distinctes (Kick, Snare, Hi-Hat) grâce à la base de son constituée en octobre et entièrement testables et fonctionnelles indépendamment. Cette gestion multipistes est opérationnelle un mois avant l'échéance prévue.

## Amélioration de l'expérience utilisateur

L'avance prise a également permis d'intégrer immédiatement des fonctionnalités de confort qui n'étaient attendues que plus tard :

- Contrôles par piste : Ajout d'une fonction « Mute » individuelle sur chacune des pistes.
- Contrôle global : Implémentation d'un volume général minimal (50% ou 100%).
- Développement d'un thème sombre via CSS, offrant un confort visuel et une esthétique plus en adéquation avec le monde de la musique électronique.

## Conclusion

Le mois de novembre marque une étape pivot. En capitalisant sur le prototype existant dès octobre, j'ai pu transformer une simple ébauche en une application structurée et modulaire. Avec l'architecture POO en place et la gestion des pistes déjà active, je suis idéalement positionné pour me concentrer sur le défi majeur de décembre, la logique algorithmique des patterns, du tempo ainsi qu'une première gestion de la lecture en boucle et continuer sur cette bonne lancée.