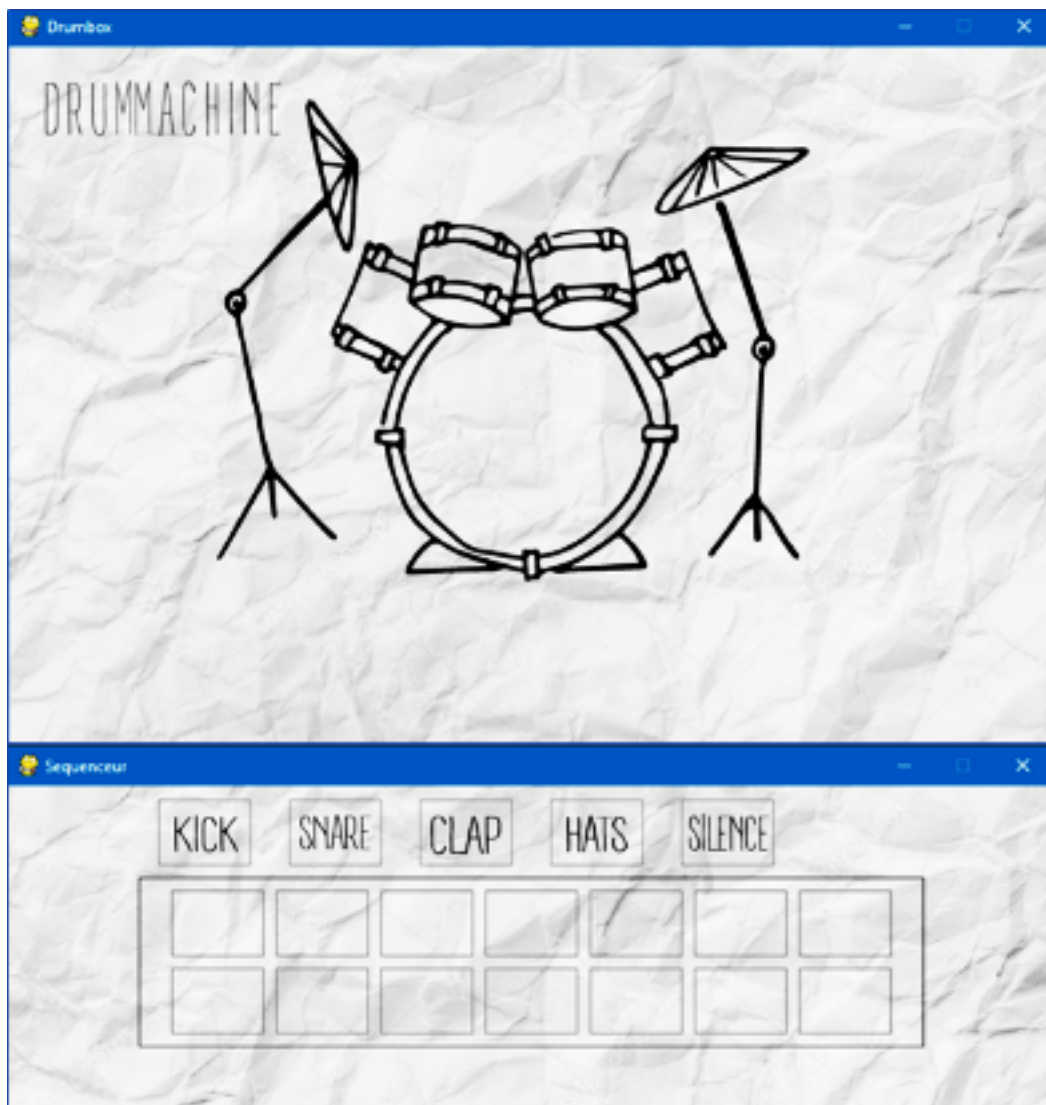


PROJET ISN : DRUMBOX



ROSINA Léo, **CALLEWAERT Jules**, MARTIN-LOUMEAU Adèle
Terminales S

SOMMAIRE

I/ Présentation du projet

- A) Pourquoi ce projet ?*
- B) Cahier des charges : objectifs du projet, structure globale, positionnement par rapport à d'autres solutions existantes.*
- C) Répartition des tâches : carnet de bord*

II/ Réalisation technique personnelle

- A) Réalisation personnelle*
- B) Résultat final*

III/ Réflexion post-projet

- A) Perspectives d'améliorations*
- B) Bilan personnel sur les apports du projet*

Annexes

I / PRÉSENTATION DU PROJET :

La DrumBox est le projet que nous avons réalisé durant cette année de Terminale dans le cadre de notre enseignement de spécialité ISN. Nous y avons rencontré de nombreuses difficultés et consacré une grande partie de notre temps, et nous sommes fiers du résultat. Notre DrumBox est une batterie virtuelle à laquelle on peut jouer en pressant les touches du clavier de notre ordinateur, et qui peut lire une partition (liste d'instruments) que l'on écrit nous-même en utilisant la souris.

A) Pourquoi ce projet ?

Lors de la création de notre équipe en janvier, l'idée d'un projet autour de la musique nous trottait déjà en tête. En effet, nous sommes tous trois de grands passionnés de musique, et en présence de ce fort point commun, le projet d'une drumbox nous est rapidement apparu comme étant un choix judicieux.

Aujourd'hui, la musique est très liée au numérique et l'industrie musicale utilise beaucoup l'outil informatique, et sa maîtrise peut être un atout dans ce secteur.

De plus, chacun de nous trois ayant déjà observé ou même pratiqué de la MAO (Musique Assistée par Ordinateur), il était très enrichissant pour nous de comprendre les rouages derrière les logiciels de MAO. Nous voulions découvrir comment la musique peut être codée avec Python, et nous voulions explorer, non seulement la façon dont on associe une action sur l'ordinateur (clavier ou souris) à un son, mais également celle dont on crée un son à part entière, à l'aide de principes de physique comme les fréquences.

Ainsi, la problématique de notre projet est : de quelles différentes manières peut-on générer et contrôler un son grâce à l'outil informatique et au langage python?

C) Cahier des charges :

OBJECTIFS DU PROJET:

Même si nous avons appris depuis le début de l'année à utiliser Python, nous savions que nous aurions beaucoup à apprendre sur la gestion du son dans ce langage. Mais il est vrai que nous nous lançons quelque peu dans l'inconnu, c'est pourquoi nous avons beaucoup changé d'avis au cours du projet (suppression et ajout de fonctionnalités).

Nous avons été légèrement ambitieux au début : nous voulions pouvoir enregistrer les morceaux que l'on créait. Il est apparu très vite que cela demandait des connaissances trop poussées et que nous manquions d'informations, et qu'il était plus judicieux de se concentrer sur d'autres fonctionnalités. Finalement nos concertations dans le groupe ainsi qu'avec notre professeur nous ont tournés vers deux nouvelles fonctionnalités très intéressantes : le clavier virtuel et le séquenceur. Le séquenceur est une fonctionnalité qui nous a poussé à mieux comprendre le fonctionnement du son, mais surtout nous a permis de découvrir la gestion des événements de souris. Le clavier virtuel nous a appris à coder des sons à partir de données numériques.

Notre programme consiste en une drumbox, un clavier virtuel ainsi qu'un séquenceur, et nous accédons à ces fonctions par un menu principal.

DrumBox :

- Quatre sons de batterie (grosse caisse, caisse claire, charleston, cymbale crash)
- Image d'une batterie dont chaque instrument change de couleur lorsqu'on appuie sur la touche qui lui est attribué
- Utilisation des touches A, Z, Q, S.

Clavier virtuel :

- Notes de clavier qui dépendent d'une fréquence (de DO à LA)
- Affectation de chaque touche à une note, de F à L
- Image de clavier dont chaque touche change de couleur lorsqu'on appuie sur la touche qui lui est attribuée.
- Matériel : Adobe Illustrator (design), modules : numpy et sounddevice

Séquenceur :

- 5 cases avec sur chacune un instrument de la batterie + une case "silence"
- 14 "boîtes" où placer les cases
- affectation de la touche "p" au séquenceur qui joue la suite de cases en fonction de l'ordre dans lequel on les a placées dans les boîtes.

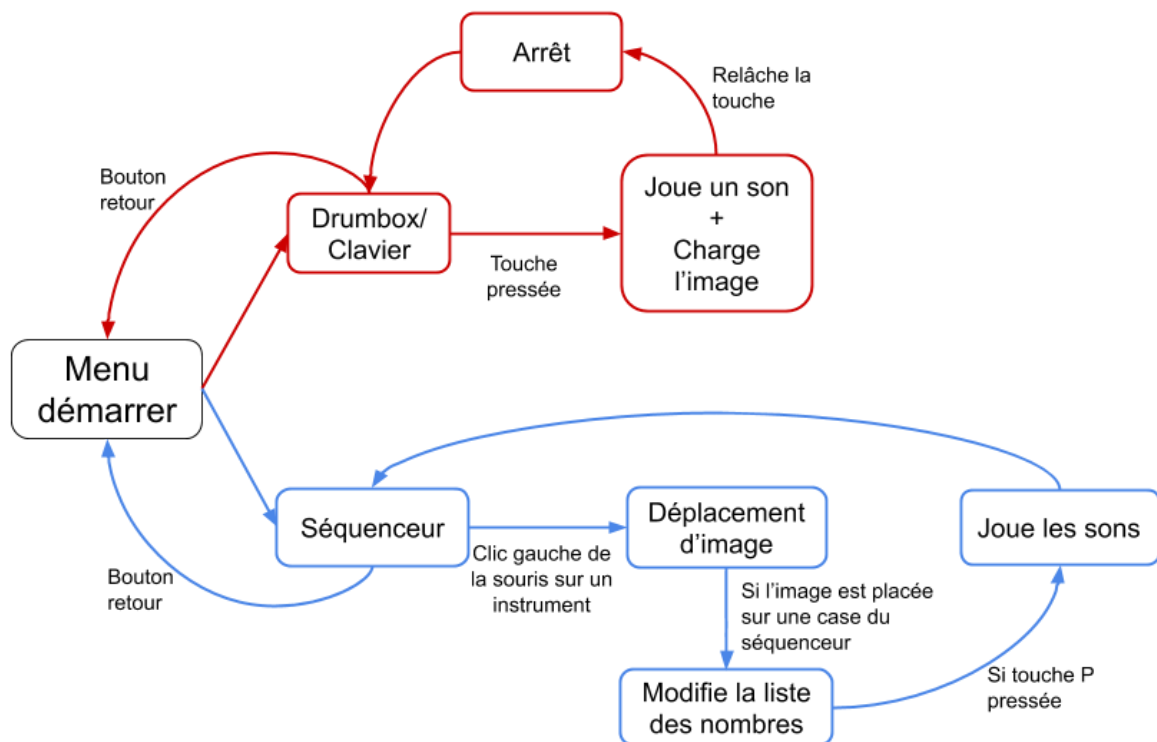
Voici le matériel et les sources que nous avons utilisé:

Plateforme de partage : *github* (facilite la communication grâce au tableau de bord et le partage de documents avec les “branches”)

Logiciel : *Thonny* (simple d'utilisation et assez performant pour des projets d'une longueur comme le notre, possède beaucoup de bibliothèques python)

Bibliothèques python : *Pygame, Sounddevice, Numpy, Time*

STRUCTURE DU PROGRAMME :



POSITIONNEMENT PAR RAPPORT À D'AUTRES SOLUTIONS DÉJÀ EXISTANTES :

Il existe déjà de nombreux logiciels qui permettent de jouer de la musique (le plus connu est “garageband”, mais il y a aussi des logiciels de MAO comme Fl Studio, Logic Pro). Ils sont très développés et performants, et possèdent une multitude d'instruments, d'effets sonores, et des fonctionnalités comme l'enregistrement. Sans oser prétendre faire face à ces logiciels, notre projet s'en démarque tout de même premièrement car il est plus spécialisé (batterie et clavier) et peut servir à des utilisateurs qui ne recherchent que ces fonctions. De plus, les logiciels de MAO sont assez compliqués à utiliser, alors que notre projet est très simple à manier ! Enfin, notre démarche de création du son par différents moyens est originale et constitue un atout par rapport

aux autres logiciels qui n'utilisent souvent que des sons enregistrés ou que des données numériques.

B) Répartition des tâches :

JOURNAL DE BORD DU PROJET ET RÉPARTITION DES TÂCHES :

Dates	Léo	Adèle	Jules
10/01/2019	Invention du nom de l'équipe; Début d'idées sur le projet: appuyer sur les touches de clavier afin de produire un son de batterie/percussion; Donner à l'utilisateur la possibilité de s'enregistrer et de rejouer la piste précédemment enregistrée.		
17/01/2019	Recherche des sons de batteries; Imagination du profil qu'aura la fenêtre; Début d'apprentissage d'outils sur Python afin de jouer des sons		
24/01/2019	Design de la batterie avec des images.png	Recherches sur la possibilité d'enregistrer des sons que l'on produit nous-même	Apprentissage de la commande liée à la pression et au relâchement d'une touche
31/01/2019	Apprentissage de des tkinter puis abandon de cette bibliothèque car la gestion des événements est compliquée	Apprentissage sur Pygame (avec Pyaudio et le package Numpy) de comment enregistrer un son à partir d'un fichier	Mini programme de test afin de jouer des sons lorsque l'on appuie sur une touche avec des problèmes
7/02/2019	Début du code de la Drumbox à l'aide du petit programme de Jules	Abandon de l'enregistrement. Exploration du package Numpy. Début de la création artificielle de sons sur des fréquences différentes.	Réussite du mini programme, celui-ci joue des sons enregistrés dans le fichier lorsqu'on appuie sur une touche. Début du codage de la Drumbox.
Du 7/02 au 14/03	Durant les vacances nous avons chacun avancé sur notre partie du programme		
14/03/2019	Finition du code de la drumbox	Association de touches à une fréquence pour jouer des notes de clavier.	Finition du code de la drumbox
4/04/2019	Entretien avec Mr Jules, il nous fait constater que notre programme ne présente pas tellement de difficultés, et nous propose l'idée d'une fonctionnalité de		

	séquenceur.		
18/04/2019	Début de la réflexion autour du séquenceur		
22/04/2019	Début de la programmation du séquenceur. Création de l'interface graphique à l'aide d'Adobe Illustrator. Découverte de la gestion des événements d'une souris.	Début de la programmation du séquenceur. Découverte de la gestion des événements d'une souris.	Début de la programmation du séquenceur. Découverte de la gestion des événements d'une souris.
23/04/2019	Fonctionnalité de glisser/déposer codée, finitions de l'interface graphique du séquenceur.	Elaboration du menu principal (design des touches + élaboration des événements de la souris correspondant aux touches du menu).	Suite de la programmation du séquenceur avec Léo.
26/04/2019	Changement du design de la drumbox. Fin de la programmation du séquenceur avec la fonction d'analyse de la liste de nombres jouant. Insertion du code du séquenceur dans le menu principal avec la drumbox	Fusion des codes du menu principal et de celui de la drumbox.	Rangement et organisation du dossier github.

DÉMARCHE COLLABORATIVE:

Nous avons bien su collaborer, et une vraie dimension de confiance et d'entraide s'est installée. Léo étant très doué en design, Jules qui s'occupait des sons enregistrés et Adèle de la partie mathématique et physique du projet, nous avons pu échanger nos connaissances et partager nos compétences. Nous avons utilisé Github mais aussi Google Drive et Messenger pour faire des points réguliers sur l'avancée du projet.

II/ PARTIE PERSONNELLE:

A) Réalisation personnelle

Dans le cadre du Projet Drumbox, j'ai été chargé de programmer les sons et participé à la programmation du séquenceur avec mon camarade Léo. Je me suis occupé de la partie "Sons" ainsi que de la partie "Touches pressées". Les problèmes que j'ai eu se trouvaient principalement sur la fonction "Sons"



Le séquenceur de 14 cases.

B) Résultat final

J'ai pu réaliser la fonction Keydown (Bouton Pressé), qui permet de lancer la lecture des sons disposés dans les cases du séquenceur.

```
def SequenceurBoutonPressé():  
    #joue le son quand on appuie sur P  
    if event.type == KEYDOWN and event.key == K_p:  
        for loop in range(14):  
            if sequenceur[loop] == 1:  
                kicksound.play()  
                time.sleep(0.3)  
            if sequenceur[loop] == 2:
```

```

        snaresound.play()
        time.sleep(0.3)
    if sequenceur[loop] == 3:
        clapsound.play()
        time.sleep(0.3)
    if sequenceur[loop] == 4:
        hatssound.play()
        time.sleep(0.3)
    if sequenceur[loop] == 5:
        time.sleep(0.5)

```

Fonction créée pour les sons du séquenceur lorsque l'on appuie sur "P".

En appuyant sur la touche "P" après avoir rempli les 14 cases du séquenceur, la pression de la touche lancerait une boucle itérative qui lit les valeurs affectées aux sons et qui les joue.

Cette boucle effectuée 14 fois permet de lire pour chaque case du séquenceur la valeur qui lui est affectée. En effet pour chaque case lorsque nous déplaçons une vignette et que nous l'y intégrons, cela lui affecte une valeur de 1 à 5. Le kick sera égal à 1, le snare à 2, le clap à 3 etc... Par exemple si le nombre est 4, le programme jouera le hat. La lecture du son est programmée par un module pygame: .play. Celui-ci ouvre le fichier du son et le joue.

À la suite de la lecture du chiffre, on peut voir une ligne avec l'indication "time.sleep". Cette instruction fait en sorte qu'après le son joué, il y ait un temps de silence afin de ne pas enchaîner tous les sons et de donner une rythmique au morceau. Les valeurs sont affichées en secondes. Elle est fixée à 0,3 secondes pour le silence après avoir joué un rythme, et à 0,5 secondes pour le silence.

Avant d'obtenir cette fonction, je suis passé par beaucoup d'étapes depuis janvier. J'ai en effet dû apprendre beaucoup sur la bibliothèque en relation avec les sons sur Python, la bibliothèque "Pygame". Et dans cet apprentissage viennent aussi les problèmes. Par exemple, je n'arrivais pas à faire jouer les sons dans mon tout premier programme. Je me suis remis en question, j'ai cherché sur plusieurs plateformes de ressources Python sans succès. Jusqu'au moment où j'ai trouvé solution à ce problème en changeant la taille des sons. Ceux-ci étaient en 32 bits au lieu de 16, j'ai donc dû procéder à un transfert sur Audacity.

III/ RÉFLEXION POST-PROJET

A) Perspectives d'amélioration:

Afin d'améliorer notre programme, j'avais pour idée de donner la possibilité à l'utilisateur de choisir le nombre de cases qu'occupera le séquenceur. L'idée serait de créer un input au début de la fonction.

En plus de cela, on pourrait envisager l'ajout d'un autre input dans lequel on pourrait fixer la durée du time.sleep pour chaque instrument. Cela induirait une modification du tempo des 14 mesures jouées par le séquenceur, un but important pour un logiciel musical.

Perspectives d'amélioration impactant le programme entier:

- Séquenceur : le majeur problème que nous devrions régler est le fait que lorsque l'on relâche la souris, les cases des instruments ne restent pas exactement dans leur boîte. Nous pourrions également améliorer le séquenceur en faisant en sorte de pouvoir placer nos instruments plusieurs*

fois dans les boîtes car pour l'instant, il ne sont utilisables qu'une seule fois. Nous pourrions ensuite ajouter le clavier au séquenceur.

- Drumbox : Le grand abandon de notre projet a été la possibilité d'enregistrement. La principale amélioration que nous pourrions faire sur notre programme serait de s'enregistrer. Cela serait rendu possible grâce à l'utilisation de la bibliothèque "sounddevice".

B) Bilan personnel sur les apports du projet

Ce projet m'a été fortement bénéfique en plusieurs points:

- Tout d'abord dans l'apprentissage du langage Python, dans la maîtrise des bibliothèques et de tous les outils dont il dispose. Certes je sais que je n'ai découvert qu'une infime partie de ce grand monde qu'est Python mais je pense avoir des connaissances de bases grâce à mon année passée sur France IOI puis sur le projet. Le projet m'a permis d'approfondir ces connaissances de base, notamment dans la découverte et l'utilisation des bibliothèques telles que "time" ou encore "pygame". Je me suis familiarisé avec les fonctions, qui représentaient avant un vrai défi pour moi.
- Ensuite, du point de vue collectif et de gestion de tâches au sein d'un groupe, ce projet m'a été très bénéfique car j'ai appris le respect des délais, la répartition des tâches et l'apport personnel que chacun a sur le projet. C'est sur cette complémentarité que nous avons misé dès la création de notre projet et lorsque quelqu'un avait un problème, il y avait toujours un camarade du groupe pour aider celui en difficulté.
- Sur le point de vue musical, j'ai pu approfondir ma connaissance dans le domaine de programmation de la musique, cela appartenant au domaine de la Musique Assistée par Ordinateur. Cela ajoute une corde de plus à mon arc de connaissances et de techniques musicales.
- Enfin, sur le point de vue personnel, cela m'a permis de découvrir mes capacités de recherche, de démarches personnelles. J'ai pu mettre mes défauts de côté et mettre à l'oeuvre mes qualités pour moi et mes camarades. J'ai donc pu développer mon optimisme dans les moments durs. J'ai aussi été entraîné vers l'avant par mon groupe qui fournissait un très grand travail, et cela m'a vraiment plu.

