

**Université de Sousse**

**Ecole Nationale d’Ingénieurs de Sousse**

****

***Rapport du Projet module***

**Traitement numérique d’un signal musical**

**Réalisé par :**

AMMARI Rafik

BOURKHIS Iheb

**Filière :**

**Génie électronique industrielle : Ei2.2**

**AU : 2023-2024.**

# **Remerciements**

Nous tenons à exprimer nos sincères remerciements à notre encadrante, Mme AKKARI Hanen, pour son engagement et son soutien inestimable tout au long du projet du module de traitement du signal.

Sa confiance dès le début de notre collaboration a été un moteur essentiel dans la réalisation de ce travail. Sa disponibilité constante pour comprendre nos problématiques, échanger des idées et discuter des différentes approches a grandement contribué à l'avancement du projet.

Les conseils éclairés de Mme AKKARI Hanen, sa capacité à guider nos réflexions et son dévouement à la réussite du projet ont été des atouts précieux.



**Rafik AMMARI**

**Iheb BOURKHIS**

**Table de matières**

[**Remerciements** 2](#_Toc151670059)

[**Introduction générale** 6](#_Toc151670060)

[**Chapitre 1 :**  [**Cahier de charges** 8](#_Toc151670062)](#_Toc151670061)

**I-Introduction**…………………………………………..…………………………………………………………………………………………...6

[**II-Elaboration du cahier de charges** 8](#_Toc151670063)

[**1.Cadre du travail** 8](#_Toc151670064)

[**2.Contexte musical** 8](#_Toc151670065)

[**3.Problématique** 8](#_Toc151670066)

[**4.Objectif** 9](#_Toc151670067)

**5.Etapes de travail**…………………………………………………………………………………………………………………………8

#### L’acquisition du signal…………………………………………………………………………………………………..9

#### Stopper l’acquisition………………………………………………………………………………………………………9

#### Fenêtrage du signal………………………………………………………………………………………………………….9

#### Affichage du spectre……………………………………………………………………………………………………….9

#### L’Ajout de bruit………………………………………………………………………………………………………………9

#### Utilisation du spectrogramme pour la représentation visuelle………………………………10

#### Egaliser…………………………………………………………………………………………………………………………….10

#### La compression du signal .wav…………………………………………………………………………………….10

#### Filtrage par filtre Wiener……………………………………………………………………………………………..10

[**6.Base de Données** 11](#_Toc151670068)

[**IV-Conclusion** 12](#_Toc151670069)

[**Chapitre 2 :** [**Réalisation du projet** 14](#_Toc151670071)](#_Toc151670070)

[**I-Introduction** 14](#_Toc151670072)

[**II-IDE utilisé** 14](#_Toc151670073)

[**III- Les fonctionnalités réalisés** 15](#_Toc151670074)

##### **Importation des bibliothèques**……………………………………………………………………………………………………15

##### **Création de l'interface graphique**………………………………………………………………………………………………15

##### **Chargement du fichier audio initial**…………………………………………………………………………………………..15

##### **Définition des fonctions de traitement du signal**……………………………………………………………………….15

##### **Définition des fonctions de l'interface graphique**………………………………………………………………………15

##### **Création des boutons et gestion des événements**……………………………………………………………………….16

[**IV-Réalisation de l’interface** 17](#_Toc151670075)

[**V-Perspectives** 18](#_Toc151670076)

[**VI-Conclusion** 19](#_Toc151670077)

[**Conclusion générale** 20](#_Toc151670078)

**Listes de figures**

[**Figure1. 1 : Logo plateforme freesound** 10](#_Toc151667406)

[**Figure2. 1 : Logo pycharm** 14](#_Toc151667413)

[**Figure2. 2 : Acquisition du signal musical** 17](#_Toc151667414)

[**Figure2. 3 : Fenêtrage** 17](#_Toc151667415)

[**Figure2. 4 : Ajout de bruit** 18](#_Toc151667416)

# **Introduction générale**

Le traitement du signal numérique du son est une discipline incontournable dans le contexte technologique contemporain, jouant un rôle crucial dans divers domaines allant de la communication audio à la production musicale, en passant par la reconnaissance vocale et les systèmes audio embarqués. Ce champ d'étude s'attache à l'analyse, à la modification et à l'interprétation des signaux sonores capturés par des dispositifs tels que les microphones.

Dans ce contexte, notre projet se focalise sur le traitement du signal numérique du son, visant à explorer et à mettre en œuvre des méthodes avancées pour extraire des informations pertinentes, réduire le bruit indésirable et améliorer la qualité globale du signal sonore. Le son étant présent partout dans notre quotidien, des applications allant de la suppression de bruit indésirable à la reconnaissance vocale, en passant par la compression audio et la synthèse sonore, justifient pleinement l'importance de ce projet.

L'évolution rapide des technologies numériques offre de nouvelles perspectives pour la manipulation du son, que ce soit dans le cadre de la recherche académique, de l'industrie audiovisuelle ou de l'intégration de systèmes sonores dans des dispositifs électroniques grand public. Notre projet s'inscrit dans cette dynamique, visant à développer des compétences techniques et analytiques dans le domaine du traitement du signal numérique du son, tout en explorant des applications concrètes et pertinentes.

**Chapitre1 :**

**Cahier de charges**

## **Chapitre 1**

## **Cahier de charges**

**I-Introduction**Dans ce premier chapitre, notre analyse approfondie se concentre sur le cahier des charges, un élément essentiel qui précise de manière stricte les besoins fondamentaux guidant le développement du projet de traitement de signal.

## **II-Elaboration du cahier de charges**

### **1.Cadre du travail**

Module-Traitement du signal numérique.

### **2.Contexte musical**

Notre contribution pourrait être appliquée dans deux contextes distincts :

D'une part, les producteurs de musique pourraient tirer avantage de notre travail en effectuant un prétraitement des enregistrements audio, en ajustant les amplitudes des différentes fréquences pour obtenir le son désiré. Ainsi, notre script peut être intégré à l'analyse d'enregistrements audio personnels, tels que la voix, la musique, etc.

D'autre part, les créateurs d'effets sonores pour le cinéma, les jeux vidéo et d'autres médias interactifs pourraient exploiter notre travail pour manipuler et améliorer les enregistrements audios utilisés dans la création d'ambiances sonores.

Nous avons choisi de privilégier le premier contexte.

### **3.Problématique**

Comment notre application peut-elle répondre aux besoins spécifiques des producteurs de musique, en offrant des fonctionnalités de prétraitement pour ajuster les amplitudes des différentes fréquences et parvenir au son désiré ? D'autre part, comment notre application peut-elle être adaptée pour répondre aux exigences des créateurs d'effets sonores, en offrant des outils de manipulation et d'amélioration des enregistrements audio utilisés dans la création d'ambiances sonores pour le cinéma, les jeux vidéo et d'autres médias interactifs ?

### **4.Objectif**

Le but de l’application est de fournir une application graphique en utilisant Tkinter et Matplotlib en Python pour visualiser, manipuler et analyser des signaux audios. L'application permet le chargement d'un fichier audio (musicale), l'affichage du signal temporel correspondant, le fenêtrage du signal, l'ajout de bruit, l'égalisation, la compression MP3, la réduction du bruit avec le filtre de Wiener, et l'affichage du spectre d'amplitude. L'interface propose également des boutons pour démarrer/arrêter l'acquisition en temps réel du signal

**5.Etapes de travail**  
Comme guide, nous avons suivi cette démarche :

#### L’acquisition du signal :

* + Démarrer l'acquisition du signal et commence à mettre à jour le graphique à intervalles réguliers.

#### Stopper l’acquisition :

* + Arrêter l'acquisition en cours.

#### Fenêtrage du signal :

* + En utilisant la fenêtre de Hamming, on obtient une représentation fréquentielle plus précise et moins sujette aux distorsions. Cela permet de mieux mettre évidence les composantes fréquentielles du signal dans la plage temporelle définie, facilitant ainsi l'interprétation des caractéristiques du signal.

#### Affichage du spectre :

* + L’application de la transformée de Fourier au signal et affichage du spectre d'amplitude. En fait la transformé de fourrier. Lorsqu'elle est appliquée à un signal, elle permet d'analyser ce signal dans le domaine fréquentiel, offrant ainsi une représentation alternative du signal en termes de ses composantes fréquentielles.

#### L’Ajout de bruit :

* + L'ajout de bruit permet de simuler des conditions réelles auxquelles le signal audio pourrait être exposé dans des environnements du monde réel. Cela rend les analyses et les traitements du signal plus robustes, car le système est capable de gérer des signaux perturbés, tout comme il pourrait l'être dans des situations réelles.il constitue une approche stratégique pour renforcer la robustesse de votre système, évaluer ses performances dans des conditions variées et assurer son adaptabilité à un large éventail de scénarios réels.

#### Utilisation du spectrogramme pour la représentation visuelle :

* + En utilisant un spectrogramme, on peut visualiser les caractéristiques fréquentielles d'un signal et identifier les zones où l'égalisation pourrait être souhaitable**. Par exemple**, si le spectrogramme révèle qu'une plage de fréquences spécifique est particulièrement forte ou faible, cela pourrait indiquer la nécessité d'une égalisation.

#### Egaliser :

* + Égalisation su signal **en normalisant les amplitudes** par fréquence et mise à jour du graphique principal.
  + La normalisation est couramment utilisée dans le traitement du signal audio pour garantir que les signaux aient des amplitudes comparables et pour éviter tout problème lié à des valeurs d'amplitude trop élevées ou trop basses.
  + L'égalisation peut impliquer des ajustements sur l'ensemble de la bande passante agit directement sur la répartition de fréquence

#### La compression du signal .wav :

* + La compression audio du format WAV vise à réduire la taille des fichiers audio sans perte de qualité perceptible. Contrairement à la compression avec perte, qui élimine des données, la compression en domaine temporel pour le format WAV exploite les redondances temporelles dans le signal audio. L'objectif est d'optimiser l'utilisation des ressources de stockage et de transmission tout en préservant une qualité audio élevée, ce qui est particulièrement important dans des contextes où l'espace disque est limité.

#### Filtrage par filtre Wiener :

* + Le filtre de Wiener est généralement classé comme un filtre adaptatif, et sa conception est souvent basée sur l'estimation du signal et la minimisation de l'erreur quadratique moyenne entre le signal d'origine et le signal filtré. Il n'est pas classé spécifiquement comme un filtre passe-bas ou passe-haut dans le sens traditionnel de ces termes.
  + Du point de vue de son type, le filtre de Wiener est généralement classé comme un filtre adaptatif. Cela signifie qu'il est capable de s'ajuster automatiquement en fonction des caractéristiques du signal et du bruit présents. Il est conçu pour minimiser la différence entre le signal d'origine et le signal filtré en utilisant des paramètres adaptatifs.
  + En fin, le filtre de Wiener est un filtre adaptatif utilisé pour l'estimation et la réduction du bruit dans les signaux, et il appartient à la catégorie des filtres adaptatifs.

### **6.Base de Données**

La constitution de la base de données s'est réalisée grâce à l'utilisation de Freesound **[https://freesound.org/]**, une plateforme en ligne offrant une abondance de contenus audio. Nous avons organisé un répertoire spécifique dans lequel nous avons enregistré une variété d'échantillons sonores musicaux provenant de cette source. Cette démarche a abouti à la création d'une collection d'audios diversifiée et représentative, englobant une large palette de genres musicaux, d'instruments et de styles.



**Figure1. 1 : Logo plateforme freesound**

En adoptant cette approche, nous avons garanti une source de données variée pour alimenter nos analyses et nos expérimentations en traitement du signal. Chaque fichier audio dans le dossier constitue un élément essentiel de la base de données, offrant des opportunités riches pour les opérations telles que le filtrage, la compression, l'égalisation, et autres traitements du signal numérique que nous avons implémentés dans notre application.

## **IV-Conclusion**

En conclusion de ce premier chapitre dédié à l'exploration du cahier des charges et du contexte du projet de traitement de signal, il ressort clairement que la définition rigoureuse des besoins fondamentaux constitue un socle solide pour orienter le développement à venir. Dans le chapitre suivant, nous approfondirons ces exigences et nous plongerons dans la conception et l'implémentation du système de traitement de signal, cherchant à concrétiser les objectifs définis dans ce cahier des charges.

**Chapitre2 : Réalisation du projet**

# **Chapitre 2**

# **Réalisation du projet**

**I-Introduction**Le deuxième chapitre de notre travail offre une analyse détaillée des aspects techniques et pratiques du projet. Structuré en sections clés, il cherche à approfondir la compréhension des choix fondamentaux, des fonctionnalités intégrées et de la mise en œuvre concrète de l'interface utilisateur. L'objectif principal est de fournir une vision complète du processus de développement, mettant en évidence les outils et les techniques utilisés pour la concrétisation du projet.

# **II-IDE utilisé**

Nous avons choisi d'utiliser l'environnement d'exécution PyCharm IDE en conjonction avec le langage de programmation Python. Cette décision découle de la polyvalence de Python en tant que langage de programmation, offrant une multitude de fonctionnalités et de bibliothèques qui ont grandement facilité le développement de notre projet.

Vous trouverez ci-joint une capture d'écran de l'IDE PyCharm que nous avons employé pour ce projet :

Top of Form



**Figure2. 1 : Logo pycharm**

# **III- Les fonctionnalités réalisés**

La conception de l'interface que nous avons sélectionnée s'appuie sur l'intégration des fonctionnalités suivantes :Top of Form

##### **Importation des bibliothèques**

* Importation de toutes les bibliothèques nécessaires, notamment tkinter, Matplotlib, NumPy, soundfile, scipy, numpy, os, pydub, etc.

##### **Création de l'interface graphique**

* La création de la fenêtre principale de l'interface graphique en utilisant Tkinter.
* Initialisation de la figure Matplotlib qui sera utilisée dans toute l'application.

##### **Chargement du fichier audio initial**

* Chargement du fichier audio initial depuis l'emplacement spécifié.

##### **Définition des fonctions de traitement du signal**

* **apply\_window** : Application d’une fenêtre de Hamming à un segment de signal.
* **ajouter\_bruit** : Intègration du bruit gaussien au signal.
* **compress\_signal\_mp3** : compression du signal audio au format MP3.
* **equalize\_signal** : Normalisation des amplitudes du signal par fréquence.
* **reduire\_bruit\_et\_afficher\_resultats** : Réduction de bruit du signal fenêtré en utilisant le filtre de Wiener et affiche les résultats.
* **update\_equalized\_graph** : Mise à jour le graphique avec le signal égalisé.
* **plot\_spectrogram** : Affichage du spectrogramme du signal.

##### **Définition des fonctions de l'interface graphique**

* **update\_after\_compression** : Mise à jour du graphique après la compression du signal.
* **update\_graph** : Actualisation de graphique en fonction de l'état de l'acquisition.
* **start\_acquisition** : Démarrage de l'acquisition en temps réel du signal.
* **stop\_acquisition** : Arrêter l'acquisition en temps réel du signal.
* **on\_enter**, **on\_leave** : Gèrent l'effet de surbrillance des boutons.
* **fenetrer\_signal** : Applique une fenêtre de Hamming au signal audio initial et affiche le signal fenêtré dans une nouvelle figure.
* **ajout\_bruit** : Ajoute du bruit au signal audio initial, met à jour le signal fenêtré et affiche les résultats.
* **apply\_equalization** : Égalise le signal initial entre 0 et 5 secondes et affiche le spectrogramme égalisé.
* **afficher\_spectre\_sur\_signal\_complet** : Affiche le spectre d'amplitude du signal complet.
* **afficher\_spectre\_apres\_fenetrage** : Affiche le spectre d'amplitude du signal fenêtré.

##### **Création des boutons et gestion des événements**

* Création des boutons pour les différentes actions telles que démarrer/arrêter l'acquisition, fenêtrer le signal, afficher le spectre, égaliser le signal, compresser le signal MP3, ajouter du bruit, filtrer le signal, etc.
* L'association des fonctions adéquates à chaque bouton.

# **IV-Réalisation de l’interface**

Après la phase de codage, nous passerons à la création de l'interface graphique globale.

La figure ci-jointe offre une illustration de l'interface ainsi créée :

A screen shot of a graph

Description automatically generated

**Figure2. 2 : Acquisition du signal musical**

La figure ci-dessous représente l'étape de fenêtrage :

A screenshot of a computer

Description automatically generated

**Figure2. 3 : Fenêtrage**

Par la suite, l'étape suivante dévoile le processus d'ajout de bruit de manière opérationnelle.

A screenshot of a computer

Description automatically generated

**Figure2. 4 : Ajout de bruit**

# **V-Perspectives**

En ce qui concerne les perspectives de notre projet, nous envisageons d'intégrer deux modèles d'intelligence artificielle distincts pour enrichir davantage l'expérience utilisateur. Tout d'abord, nous envisageons de développer un modèle basé sur la recommandation. Ce modèle utilisera des techniques avancées d'apprentissage automatique pour suggérer des morceaux de musique similaires au signal d'origine. En exploitant les caractéristiques et les motifs du signal audio, le modèle recommandera des pistes musicales qui partagent des similitudes tonales, rythmiques et mélodiques, offrant ainsi à l'utilisateur une expérience d'écoute personnalisée et diversifiée.

De plus, nous explorons la possibilité d'implémenter un modèle de classification du genre musical. Ce modèle d'intelligence artificielle sera conçu pour identifier et classer automatiquement le genre de la musique à partir du signal audio. En utilisant des techniques avancées de traitement du signal et d'apprentissage automatique, le modèle attribuera avec précision des étiquettes de genre à chaque morceau, permettant aux utilisateurs de découvrir et d'explorer des genres musicaux spécifiques selon leurs préférences.

Ces deux modèles promettent d'ajouter une dimension supplémentaire à notre projet, offrant aux utilisateurs une expérience plus riche et plus personnalisée. En intégrant ces fonctionnalités, nous aspirons à transformer notre application en une plateforme musicale intelligente et innovante, alliant technologie de pointe et plaisir auditif.**Top of Form**

En synthèse, deux modèles d'intelligence artificielle seront implémentés : l'un axé sur la recommandation de musique similaire au signal d'origine et l'autre dédié à la classification automatique du genre musical à écouter.

**VI-Conclusion**   
En résumé, la réussite de notre projet de traitement de signal repose sur une intégration harmonieuse d'éléments essentiels. En les combinant de manière synergique, nous avons pu développer une interface fonctionnelle de haute qualité, répondant pleinement aux exigences du projet. Cette approche globale garantit la fiabilité et les performances optimales du système de prétraitement des enregistrements audio, contribuant ainsi grandement à la réussite globale de notre initiative en matière de traitement de signal. L'utilisation des fonctionnalités telles que la réduction de bruit, le filtrage temporel, l’ajout de bruit, la compression pour diminuer la taille volumineuse du fichier, l'égalisation, entre autres, joue un rôle crucial dans notre travail. Leur inclusion renforce significativement la qualité et le succès global de notre projet. En somme, notre approche globale a été essentielle pour atteindre nos objectifs et assurer le succès de notre projet de traitement de signal.

# **Conclusion générale**

En conclusion, notre projet consacré au traitement du signal numérique du son a été une exploration captivante des multiples facettes de ce domaine dynamique. À travers ce parcours, nous avons abordé divers concepts théoriques allant de l'acquisition du signal à la compression audio, en passant par la fenêtre du signal et la réduction du bruit. Ces éléments constituent les fondements du traitement du signal numérique du son, un domaine dont l'importance s'étend à des applications aussi variées que cruciales dans notre quotidien moderne.

L'élaboration de notre code, avec une interface graphique conviviale, a permis de mettre en pratique ces concepts théoriques de manière tangible. La fonctionnalité d'acquisition du signal en temps réel a illustré la dynamique changeante des signaux sonores, tandis que l'application de fenêtrage a mis en lumière l'impact des choix de fenêtres sur la représentation spectrale des signaux. L'ajout de bruit, la compression audio MP3, et d'autres fonctionnalités ont ajouté des dimensions pratiques en montrant comment ces techniques sont appliquées dans des contextes concrets.

Au-delà de l'aspect technique, notre projet a également souligné l'importance du traitement du signal numérique du son dans des domaines tels que la communication audio, la production musicale, la reconnaissance vocale, et bien d'autres. Les compétences acquises au cours de ce projet sont non seulement pertinentes d'un point de vue académique, mais également dans le contexte professionnel, où la maîtrise du traitement du signal ouvre des portes vers des opportunités variées.

En somme, ce projet nous a offert une perspective approfondie sur les principes fondamentaux et les applications du traitement du signal numérique du son. En tant que domaine en constante évolution, il continue de susciter l'intérêt par ses implications innovantes et son potentiel pour façonner notre expérience quotidienne du son. Cette exploration nous laisse avec une appréciation accrue pour la complexité et la richesse de ce domaine, tout en soulignant son rôle central dans la technologie contemporaine.