

# Music Sheet Maker

CaaM

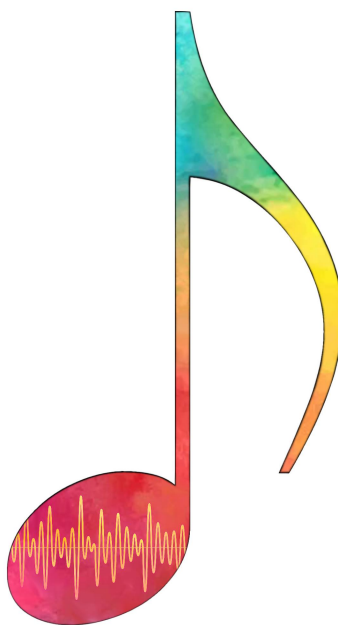
Armand FALABREGUES

Mickael HAWAT

Camille LOUIS

Amin SDOUGA

22 février 2018



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Présentation du Groupe</b>	<b>4</b>
2.1	Mickael EL HAWAT, el_haw_m (chef de groupe)	4
2.2	Amin Sdouga, sdouga_a	4
2.3	Armand Falabrègues, falabr_a	4
2.4	Camille LOUIS, louis_e	4
<b>3</b>	<b>Technologies Existantes</b>	<b>5</b>
3.1	Transformée de Fourier	5
3.2	Transformée de Fourier Rapide (FFT)	5
3.3	GTK+	5
3.4	Glade	5
3.5	SDL	5
3.6	SDL mixer	5
3.7	FMOD	5
<b>4</b>	<b>Technologies Utilisées</b>	<b>6</b>
4.1	Transformée de Fourier Rapide (FFT)	6
4.2	Le maximum d'une fonction	6
4.3	Glade	6
4.4	SDL	6
4.5	FMOD	6
4.6	Limite	6
<b>5</b>	<b>Travail accompli</b>	<b>7</b>
5.1	Interface Graphique	7
5.2	La transformée rapide de Fourier	8
5.3	Repérage des notes	9
5.4	La création de la partition	10
5.5	Le site web	12
<b>6</b>	<b>Travail à accomplir pour les prochaines soutenances</b>	<b>13</b>
6.1	Interface Graphique	13
6.2	Transformée rapide de Fourier	13
6.3	Repérage des notes	13
6.4	La création de la partition	13
6.5	Site Web	14
<b>7</b>	<b>Difficultés rencontrées</b>	<b>15</b>
7.1	Camille	15
7.2	Amin	15
7.3	Armand	15
7.4	Mickael	16
<b>8</b>	<b>Conclusion</b>	<b>17</b>

# 1 Introduction

Vous êtes musicien et il y a une musique qui vous plait beaucoup et que vous voudriez jouer ? Nous sommes heureux de vous présenter notre groupe : CaaM, et notre projet : Music Sheet Maker, un programme qui permettra de sortir la mélodie principale d'une musique (fichier audio) et de l'écrire sur une partition.

Music Sheet Maker est un programme qui permettra, à partir d'une piste audio, de trouver une mélodie et d'écrire une partition. Le but de ce projet est d'aider tout amateur de musique à trouver une partition simple pour ses musiques préférées. L'utilisateur donnera au programme une piste audio et le programme créera, en tant qu'image, la partition correspondante. L'utilisateur pourra alors enregistrer la partition et s'entraîner à jouer le morceau qu'il a souhaité sur son instrument préféré. Le but étant d'avoir un produit utilisable par un humain sans connaissance particulière.

L'intérêt de ce projet n'est pas seulement algorithmique ou mathématique mais aussi ancré dans la réalité. Nous pourrions l'utiliser une fois créé. Cela rendrait ce projet un peu plus qu'un simple projet scolaire, il deviendrait un outil.

## 2 Présentation du Groupe

Le groupe CaaM est formé d'Amin SDOUGA (sdouga\_a), Camille LOUIS (louis\_e), Armand FALABREGUES (falabr\_a) et Mickael EL HAWAT (el\_haw\_m) qui est le chef de groupe. Ce groupe est formé exclusivement d'ancien élève de la promo 2020 et le nom du groupe a été créer avec la première lettre de nos prénoms.

### 2.1 Mickael EL HAWAT, el\_haw\_m (chef de groupe)

Toujours de bonne humeur il saura motiver ses camarades pour créer le projet ensemble. Ses talents en solfège et sa grande motivation seront utiles au groupe.

### 2.2 Amin Sdouga, sdouga\_a

Son envie de rigueur et son besoin d'optimiser et sécuriser ne pourra être que bénéfique au projet Music Sheet Maker, sans évoquer ses connaissances en musique, en effet il joue de l'accordéon depuis maintenant 11 ans.

### 2.3 Armand Falabrègues, falabr\_a

A la recherche d'un projet intéressant et instructif, je m'engage entièrement dans le projet Music Sheet Maker même si mes connaissances en musique se limitent à la batterie.

### 2.4 Camille LOUIS, louis\_e

Très motivée, mon envie de produire un projet abouti et fonctionnel permettra de donner à mon équipe l'énergie nécessaire à la réalisation de celui-ci. De plus, j'ai également fait quelques années de piano et de guitare qui m'ont données des bases en solfège.

## 3 Technologies Existantes

Ayant effectué des recherches sur Internet, nous avons trouvé plusieurs technologies qui pourraient nous aider à réaliser notre projet. Les voici :

### 3.1 Transformée de Fourier

La transformée de fourier vient de la somme de fourier qui permet de décomposer tout signal sonore périodique en somme de signaux periodiques. Grâce à la transformée de fourier nous pouvons analyser un signal sonore, en obtenir son spectre d'amplitudes et trouver l'harmonique principal pour en déduire la note musicale correspondante. Nous n'avons pas besoin d'utiliser les autres capacités des théorèmes de fourier.

### 3.2 Transformée de Fourier Rapide (FFT)

La transformée de fourier rapide est un procédé qui permet de calculer la transformée de fourier discrète (DFT). Nous pouvons la mettre en oeuvre dans un algorithme de complexité  $O(n \log(n))$ . Nous pourrions alors obtenir l'harmonique principal et en déduire la note de musique comme avec la transformée de fourier.

### 3.3 GTK+

GTK+ est une bibliothèque permettant de créer des interfaces graphiques (GUI) très facilement. GTK+ existe sous Linux et est programmable en C, ce qui le rend très adapté à notre projet. Cette bibliothèque nous permettra de créer une interface graphique facilement et elle sera plus élaborée que si nous utilisions une autre bibliothèque comme SDL.

### 3.4 Glade

Glade est une application qui permet la construction d'une interface graphique à la souris sans avoir à écrire du code. Glade utilise la bibliothèque GTK+ et écrit donc le code nécessaire pour créer l'interface graphique. Glade permet donc de créer des interfaces graphique encore plus facilement que sous GTK+ et existe sous Linux.

### 3.5 SDL

SDL est une bibliothèque très ouverte qui regroupe un grand nombre d'utilisations différentes (graphique, audio, clavier, souris, etc.) Elle permet notamment de créer, analyser et modifier une image. Cette bibliothèque est disponible sur Linux et est utilisable en C.

### 3.6 SDL mixer

La SDL ne peut gérer que les fichiers WAV, avec la SDL mixer nous pourrions lire des fichiers audio MP3 et d'autres (mp3 étant le format qui nous intéresse le plus).

### 3.7 FMOD

FMOD est une bibliothèque très avancée dans le traitement de son et assez légère, elle peut nous permettre d'obtenir un spectre d'amplitude assez facilement, malheureusement elle n'est pas installée sur les machines de l'école.

## 4 Technologies Utilisées

### 4.1 Transformée de Fourier Rapide (FFT)

Après avoir ouvert le fichier avec SDL mixer, nous pourrions analyser le son et en sortir le spectre d'amplitude grâce à ce procédé.

### 4.2 Le maximum d'une fonction

Une fois une fonction ressorti de la transformée de Fourier, nous obtenons le spectre du son original. A partir de ce spectre, nous pouvons ressortir les notes sous forme de fonctions. Ces fonctions nous donneront l'information sur la note grâce à leurs maximums. Pour cette étape, il faut limiter le nombre de pics (qui représente les fréquences), donc limiter le nombre d'harmonique, voire isoler la note de ces harmoniques.

### 4.3 Glade

Pour notre projet nous avons décidé d'utiliser Glade au lieu de tout coder avec GTK+. Cela nous permettra de passer plus de temps sur les parties les plus importantes de notre projet et de rendre un produit fonctionnel et abouti.

### 4.4 SDL

Nous avons déjà utilisé la bibliothèque SDL dans d'anciens TP d'informatique. Puisque cette bibliothèque est toujours disponible pour ce projet et que nous savons déjà l'utiliser, nous avons décidé de garder cette bibliothèque. En effet nous nous en servirons pour la création des partitions de musique. En effet la partition de musique ne sera qu'une suite d'image en noir et blanc.

### 4.5 FMOD

Cette bibliothèque est compliquée mais très documentée, au final de notre projet ce serait un atout d'utiliser FMOD. Elle est capable de bien des choses dans le traitement de son et c'est plus qu'il nous en faut.

### 4.6 Limite

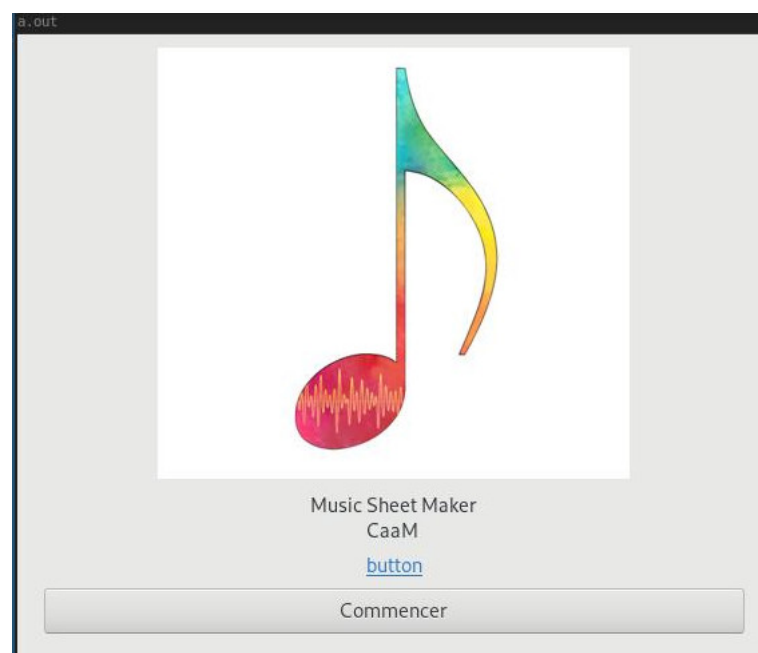
Il est actuellement impossible de différencier de manière algorithmique plusieurs instruments à la fois. c'est à dire dans le cas où un piano joue avec une guitare pendant qu'une voix chante, il sera impossible d'avoir la partition séparée du piano, de la guitare et de la voix. La partition qui résultera de l'algorithme sera la mélodie avec qui ressort le plus du morceau (avec le volume sonore le plus élevé). En effet l'ordinateur ne recevra un signal sonore qui sera l'addition de tous les instruments. C'est pour cela que notre programme se limitera à trouver en premier temps la partition d'une musique avec qu'un seul instrument.

## 5 Travail accompli

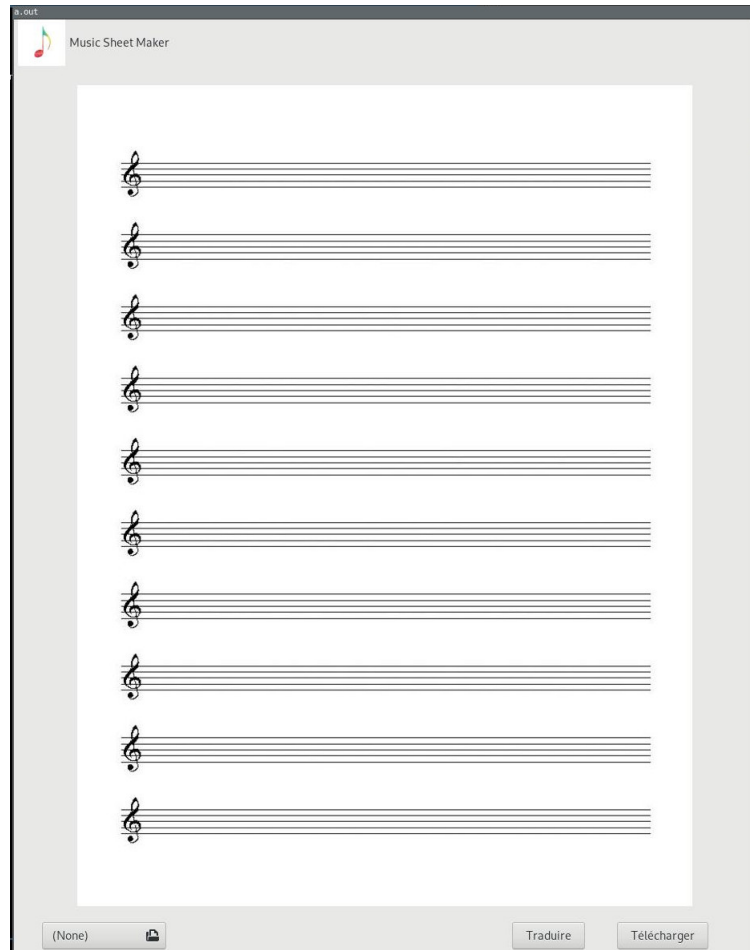
### 5.1 Interface Graphique

Comme nous l'avons expliqué, nous avons décidé d'utiliser Glade pour la création de notre interface graphique. Très simple d'utilisation et intuitif, nous n'avons pas eu beaucoup de mal à comprendre comment le logiciel marchait, mais nous avons néanmoins dû nous documenter un peu sur internet. En effet, nous nous sommes rendu compte qu'il y avait quand même une partie de code à écrire, et avons donc dû faire un peu de GTK+.

À ce stade du projet, l'interface graphique comporte une première fenêtre comportant notre logo, le nom de notre groupe, et un bouton commencer pour lancer l'application. Il y a aussi un lien qui renverra vers notre site web quand celui-ci existera. Voici ci-dessous à quoi ressemble cette première fenêtre.



Pour démarrer l'application il faut donc appuyer sur le bouton commencer qui ouvre une deuxième fenêtre. Sur cette fenêtre on a rajouté le logo et le nom de groupe, mais aussi une image d'une partition vierge. Le but de l'application est de remplacer cette photo par une photo de la partition avec les notes correspondantes à la musique qu'on veut traduire. On a donc aussi un bouton qui permet de choisir un fichier dans l'ordinateur (ce fichier sera une version audio de la musique que l'on veut traduire), un bouton traduire qui lancera la fonction de traduction, et un bouton télécharger qui permettra de télécharger l'image de la partition. Voici ci-dessous à quoi ressemble cette fenêtre.



## 5.2 La transformée rapide de Fourier

Il fallait tout d'abord comprendre le principe de la transformée de Fourier puis comment fonctionne la transformée discrète de Fourier et enfin comprendre et implémenter la transformée rapide de Fourier. Si on n'utilise pas la transformée rapide de Fourier un traitement de donnée qui prendrait trente longue secondes avec celle ci prendrait alors plus d'un an avec la transformée discrète. Ensuite étant donné que nous avons choisi d'utiliser la SDL mixer dans un premier temps, car elle est installée sur les machines de l'école, nous avons pu constaté que nous étions limité et qu'il nous fallait développer cette algorithme de transformée rapide de Fourier.

Notre première objectif est de donner un fichier audio qui contient le son d'un diapason, un La parfait à 440Hz, en entrée et de trouver qu'en effet c'est bien un La parfait grâce à la transformée rapide de Fourier.

Après une pléthore de tentatives et le temps qui commençait à nous manquer nous nous n'arrivions pas à calculer le spectre d'amplitude, implémenter la transformée rapide de Fourier et son fameux système de papillons était beaucoup plus compliqué que prévu. C'est alors que nous apprenons que nous pouvons utiliser nos propre pc linux pour la soutenance, ni une ni deux nous ajoutons FMOD au projet et après de la lecture de documentation nous avons déjà un résultat. Il reste cela dit c'est problème pour calculer la fréquence et en déduire la note de musique mais je ne doute pas que nous y arriverons très prochainement.



### 5.3 Repérage des notes

À partir d'une liste d'amplitudes classés par fréquence, il nous faut ressortir une note. Pour ce faire nous cherchons la plus grande amplitude car nous partons du principe que le reste des fréquences sont des nuisances sonores. La liste est composée de 20 000 entiers pour les 20 000 fréquences qu'un humain peut entendre (de 20 Hz à 20 000 Hz en moyenne). Ces 20 000 cases sont remplies de l'amplitude en dB à la fréquence de l'indice. Il nous faut donc une fonction qui renvoie l'indice de la liste où le maximum est atteint. Cette fonction boucle dans la liste et garde en mémoire le maximum et son indice et retourne l'indice à la fin de la boucle.

Maintenant il faut savoir de quelle note il s'agit à partir du maximum. Il faut non seulement trouver la note mais aussi l'octave. Nous avons donc utilisé un switch pour trouver les bonnes fréquences. Si les fréquences sont entre 32 et 62, il s'agit de la première octave, etc. De plus, le double des fréquences fait partie de l'octave supérieure.

## 5.4 La création de la partition

Grâce à une liste de structure représentant les notes, avec pour information le nombre de temps, la tonalité et l'octave, une partition peut être créée. On peut diviser la création des partitions en 2 parties : Le rythme et la tonalité. Pour cette soutenance, nous nous sommes plus concentrer sur l'implémentation du rythme pour cette partie.

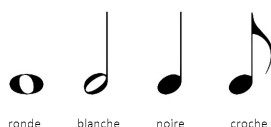
Les notes sont dessinées de différentes manières selon le temps qu'elles doivent durer. Les types de notes disponibles pour cette création de partition sont :

Les croches qui valent un demis temps

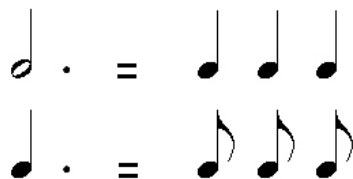
Les noires qui valent un temps (deux demis temps)

Les blanches qui valent deux temps (quatre demis temps)

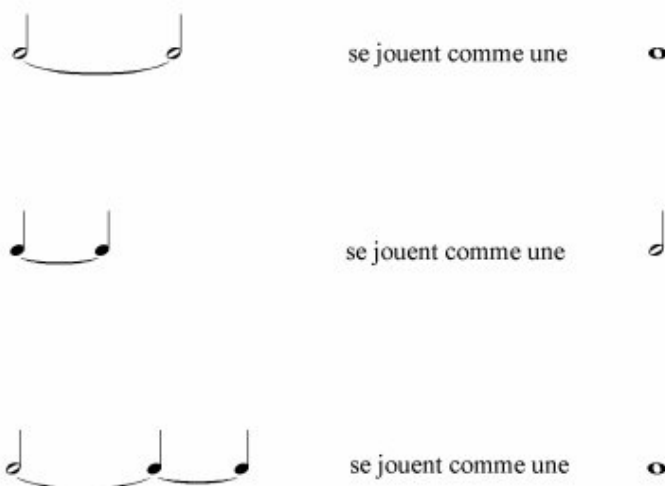
Les rondes qui valent quatre temps (huit demis temps)



Il faut savoir que pour les noires et les blanches il peut y avoir une règle spéciale. Il peut y avoir un point à droite de la note. Ce point permet de multiplier par 150% le temps de la note. C'est-à-dire que les noires pointées valent un temps et demi (trois demis temps) et les blanches pointées valent trois temps (six demis temps).



Sur cette partition nous avons pris le choix de prendre des mesures à quatre temps. Cela signifie que la somme des temps des notes de chaque mesure ne doit pas dépasser quatre temps (huit demis temps). Un problème vient donc s'ajouter. Par exemple si pour une musique il y a sept croches puis une blanche. Cela signifie que sur la première mesure il y aura sept demis temps et il reste donc plus qu'un temps disponible pour finir la mesure ( $8-7 = 1$ ). Cependant une blanche vaut quatre demis temps et il n'y en a plus qu'un valable pour cette première mesure. La solution est la liaison. La liaison permet de « lier » deux notes entre deux mesures pour respecter la règle des quatre temps par mesure et ne pas modifier le rythme de la mélodie.



Pour savoir qu'elle sera le type de note que l'on doit lier, le programme va calculer le rythme de la première note et la deuxième note. Sur l'exemple que l'on suivait, la première notes sera une croche car il reste un demi temps pour compléter la mesure. Puisqu'on a « volé » un demi temps à la blanche qui est la note de base qu'on aurait dû placer, la deuxième note sera une noire pointée, pour former en se liant avec la croche, une blanche. Dans le cas où la prochaine mesure est à la ligne d'après la liaison est coupé en deux

Un autre problème apparait après plusieurs tests : les notes à cinq et sept demis temps. En effet, il n'y a aucun symbole pour représenter des notes avec des temps de ces durées. Cela pose donc un problème mais qui sera également régler grâce aux liaisons. Car les liaisons peuvent être également utilisées dans les mesures. En liant une noire avec une noire pointée on « crée » une note avec une durée de cinq demis temps et en liant une blanche avec une noire pointée on crée une note avec une durée de sept demis temps.

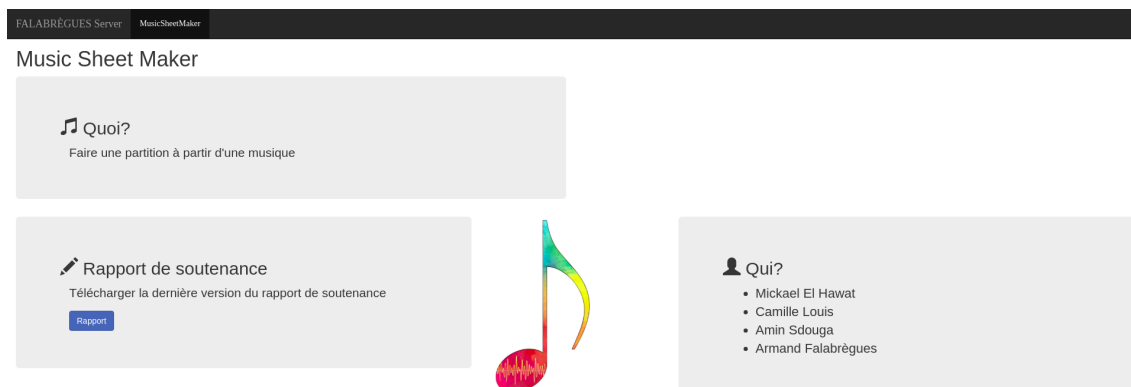
Pour des soucis d'esthétique et de compréhension pour le lecteur mais également par convention, les distances entre les notes doivent être proportionnelle selon les temps. Pour régler ce problème il y a deux listes. Une où il y a les ordonnées par rapport au pixel de chaque ligne et une autre où il y a les abscisses par rapport au pixel à chaque demi temps. Il y a 3 mesures par lignes donc vingt-quatre demis temps. Entre chaque demi temps il y a environ 25 pixels donc la note qui suivra une blanche (quatre demis temps) sera à 100 pixels de distance.

La tonalité se dessine en variant l'ordonnée de la note de quelque pixel pour bien la placer sur la partition grâce au ton et à l'octave de la structure de la note. La bonne ordonné et facile à calculer car chaque lignes est séparés par la même distance et la tonalité dépend du placement de la note selon les lignes.



## 5.5 Le site web

Pour cette première soutenance, nous n'avons qu'une version basique du site web. Il est uniquement possible de télécharger ce rapport de soutenance. Mais pour faire les choses jusqu'au bout, nous avons installé un serveur. Ce serveur tourne sur Raspberry Pi en utilisant Node.js, ce qui permet d'avoir le client et le serveur en JavaScript. L'URL de ce site est [falabregues.ml/MSM](http://falabregues.ml/MSM).



## 6 Travail à accomplir pour les prochaines soutenances

### 6.1 Interface Graphique

L'interface graphique est presque fini mais les boutons traduire et telecharger ne marche pas encore. En effet, nous n'avons pas fini de coder les fonctions correspondantes. L'interface graphique sera donc entierement fonctionnel pour la 3ème soutenance mais pas avant. Le lien vers le site web ne revoit pas non plus sur notre site web car nous ne l'avons pas encore concu. Cette partie la sera prête pour la 2ème soutenance.

Il y aussi d'autres petits details à retravailler pour rendre notre application plus esthetic. Nous avons par exemple eu l'idée de mettre une scroll bar au niveau de l'image de la partition pour ne pas avoir à l'afficher en entier. Nous avons aussi eu l'idée de créer un bouton aide mènerais vers une fenètre qui expliquerai comment utiliser l'application. Ces idées seront mis en place pour la 2ème ou la 3ème soutenance, en fonction du temps de travail que nous demande les autres parties du projet.

### 6.2 Transformée rapide de Fourier

Nous devons trouver avec précision les fréquences de notre tableau représentant le spectre d'amplitude. Et calculer avec précision la note de musique qui s'en approche le plus. Sans oublier que notre programme doit rester rapide et léger.

### 6.3 Repérage des notes

Il faut encore savoir combien de temps dure une note avant de l'afficher. En effet, il faudra savoir combien un temps dure. Cela peut dépendre de la musique et à partir de la plus petite unité de temps nous pourrons déduire les autres. À partir de cela, il faudra afficher les croches, les blanches, etc.

### 6.4 La création de la partition

Pour la prochaine soutenance, les dièses et bémols seront présentent sur la partition, ainsi que certaine note avec des tonalités en dessous et au-dessus des lignes comme le do sur l'image si dessous.



Lorsque 2 croches se suivront elles seront dessinées d'une manière moins lourde comme si dessous. Il y aura également la présence de temps silencieux. Des pauses, des demi-pauses, des silences et des soupirs.



## 6.5 Site Web

Une fois le projet terminé, il faudra ajouter une manière de l'obtenir via le site web. Que cela soit un installeur ou au moins le code. Il faudrait aussi améliorer la présentation, pour l'instant très sobre.

## 7 Difficultés rencontrées

### 7.1 Camille

Pour cette première soutenance, j'ai eu à faire l'interface graphique. Nous avons décidé d'utiliser Glade pour la réaliser et, étant donnée que c'est une plateforme assez intuitive et facile à utiliser, je n'ai pas rencontré beaucoup de difficultés. J'ai néanmoins dû me documenter pour apprendre à l'utiliser car je n'avais jamais créé d'interfaces graphiques auparavant. J'ai donc fait des recherches sur internet et Youtube m'a beaucoup aidé. J'ai eu un peu de mal à ajouter des images à l'application mais je me suis vite rendu compte que je n'avais pas mis l'image au même endroit que le fichier Glade et que ça ne pouvait donc pas marcher.

J'ai aussi eu quelques problèmes quand je me suis rendu compte qu'il fallait quand même coder un peu pour pouvoir compiler l'application. En effet je pensais que Glade s'occupait de tout. La aussi j'ai dû faire des recherches et suis tombé sur beaucoup de codes qui ne marchaient pas. Heureusement après quelques essais j'ai réussi à faire quelque chose qui marchait.

J'ai passé beaucoup de temps à essayer de comprendre pourquoi mon bouton commencé ne marchait pas. En effet mon code était censé marcher mais quand je cliquais sur le bouton il ne se passait rien. J'ai eu peur que se soit parce que j'avais placé les deux fenêtres dans le même fichier glade car je n'avais rien vu de tel sur internet. Si ça avait été le cas j'aurais dû tout recommencer. Heureusement j'en ai parlé à mon groupe et il m'ont dit que ce n'était pas le problème. Finalement je me suis rendu compte que je n'avais pas activé l'option clicked sur Glade et que la était mon erreur.

Juste avant la soutenance nous avons décidé de tester l'application sur un autre ordinateur. L'écran n'étant pas de la même taille, nous nous sommes rendu compte qu'on ne voyait pas une partie de l'application. J'ai donc dû faire des modifications de dernière minute pour réduire la taille de l'application.

### 7.2 Amin

J'ai passé beaucoup de temps à lire des documents à propos de la transformée de Fourier sans trop comprendre comment ça fonctionne, puis pareil avec la transformée discrète, même si je comprend globalement comment ça fonctionne, dans la pratique je suis incapable de le redévelopper pour l'instant. C'était mon plus gros problème, essayer de développer l'algorithme de la transformée rapide de Fourier. Puis je suis tombé très malade quelques jours avant la soutenance. Par manque de moyen et de temps j'ai décidé d'utiliser FMOD qui permet d'obtenir le spectre d'amplitude beaucoup plus facilement.

### 7.3 Armand

Ma partie étant sur la reconnaissance des notes, je devais récupérer les données juste après la transformée de Fourier. Or nous pensions en premier lieu que je devrais travailler sur des fonctions continues. Or pour trouver le maximum (la note principal qui nous intéresse) il fallait passer par la dérivation de la fonction et trouver en quels points elle s'annule. Une fois avec ces extremums, pour savoir si il s'agit d'un minimum ou d'un maximum il faudrait trouver la dérivée de la dérivée (ou dérivée seconde). Enfin il faudrait comparer tous les résultats pour trouver lequel est le plus grand à partir de cette liste de points en lesquels la fonction est maximum localement.

Le plus gros problème étant pour calculer les dérivées car il n'existe pas de librairie de calcul de dérivée en C. J'aurais donc dû la créer moi-même. Une fois que nous avons convenu que je recevrais une liste de 20 000 amplitude pour chaque fréquence, le problème ne se posait plus et s'est résolu de lui-même.

## 7.4 Mickael

Au début je créer dessiner la partition vierge grâce à un parcours d'une image avec SDL et en dessinant chaque pixels noir et blanc. Cependant cette tâche est longue et facilement évitable. En effet, il m'a suffi de dessiner au préalable grâce à Photoshop une partition vierge puis de la télécharger sous le format Bitmap (.bmp). Ainsi le programme n'a plus qu'à remplir simplement cette partition aux bons endroits.



## 8 Conclusion

Bien que ce ne soit que la première soutenance, nous avons beaucoup avancé dans notre projet. Nous avons chacun travaillé de notre côté tout en essayant de communiquer le plus possible et le résultat est concluant. Nous formons une bonne équipe et chacun a su trouver sa place en fonction de ses qualités. Nous avons tous travaillé de manière égale et c'est assez prometteur pour les prochaines soutenances. Nous commençons déjà à réfléchir aux améliorations que nous pourrions apporter à notre application et allons travailler dur pour rendre un produit fini et qui sait, peut être commercialisable.