

* Bouchti NORDINE
* LAMINE Marouane
* phily TOM
* rostan ESTELLE
* Traoré AICHATOU

**Réaliser par :**

Encadrant : Fabrice huet

KARAOKE INVERSEE

Rapport Projet D’année

M1 MIAGE |DE NICE SOPHIA ANTIPOLIS

2016-2017

Résumé

Table des matières

[I. Présentation du sujet 3](#_Toc483347614)

[1. Contexte 3](#_Toc483347615)

[2. Taches planifiées 3](#_Toc483347616)

[3. Taches existant 3](#_Toc483347617)

[II. Travail accompli 4](#_Toc483347618)

[1. Choix techniques 4](#_Toc483347619)

[2. Taches effectuées **Erreur ! Signet non défini.**](#_Toc483347620)

[3. Résultat livré au client 5](#_Toc483347621)

[III. Gestion de projet 5](#_Toc483347622)

[1. Outils utilisés 5](#_Toc483347623)

[2. Organisation des taches dans le groupe 6](#_Toc483347624)

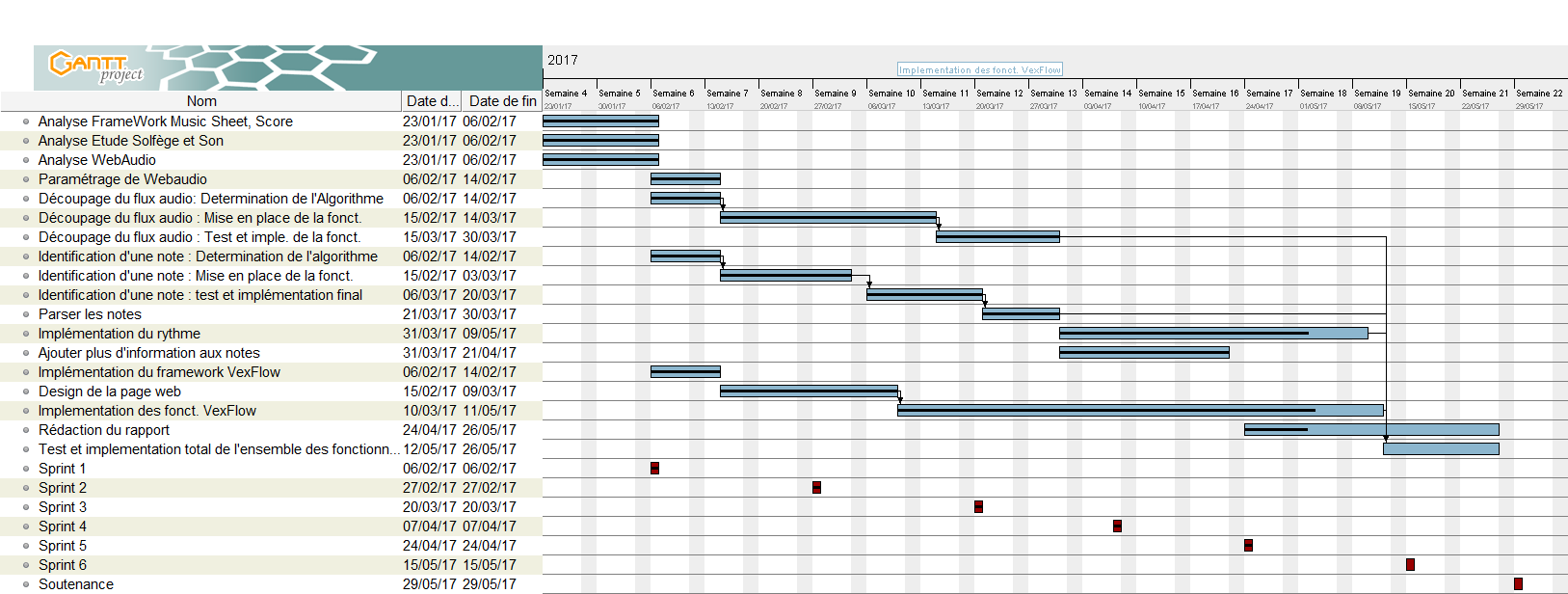
[IV. Conclusion 6](#_Toc483347625)

# Présentation du sujet

## Contexte

Dans le cadre de notre projet d’année nous nous intéressons à développer une application de karaoké inversée qui à partir d'une note de musique jouée ou un fichier audio, analyse les fréquences et affiche les notes correspondantes sur une partition à l'écran. Le fonctionnement est donc globalement celui d'un accordeur. Pour que le projet soit très XXIème siècle, Nous avons choisi de le réaliser avec l’API WEB AUDIO, HTML, JAVA SCRIPT, CSS, BOOSTRAP.

## Taches planifiées



## Taches existant

# Travail accompli

## Choix techniques

Nous avons réalisé l’application karaoké inversé avec les technologies suivants :

* Traitement du son audio

**Capture de l’audio en HTML5**

Pendant de nombreuses années, nous étions obligés de compter sur des plugins de navigateur tels que Flash ou Silverlight pour capter le son. La montée de l’HTML5 a apporté une transition remarquable en puissance de l’accès au matériel du périphérique. La géolocalisation (GPS), L'API d'orientation et l'API Web Audio (matériel audio), sont des exemples de montée. Ces fonctionnalités sont extrêmement puissantes, révélant des API JavaScript de haut niveau qui s'ajoutent aux fonctionnalités matérielles principales du système.

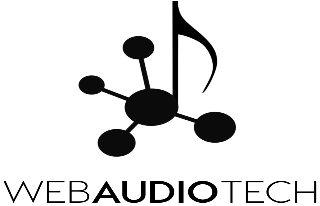
Notre recherche dans la nouvelle API est focalisé sur navigator.getUserMedia(), qui permet aux applications web d’accéder à camera et microphone de l’utilisateur.

* *getUserMedia()*

Le rythme de développement pour trouver une API de capture plus adaptée a accéléré grâce à l’effort WebRTC (Web Real Time Communication). Cette spécification est supervisée par le groupe de W3C WebRTC. GetUserMedia () est lié à WebRTC car elle est la passerelle pour cet ensemble d'API. Il fournit les moyens d'accéder au flux local de l'utilisateur notamment le microphone et la caméra.

**Support des navigateurs :**

getUserMedia() a été pris en charge depuis Chrome 21, Opera 18, et Firefox 17.



La Web Audio API propose un système puissant et flexible pour contrôler les données audio sur internet. Elle permet notamment de sélectionner des sources audio (microphone, flux media), d'y ajouter des effets, de créer des visualisations, d'appliquer des effets de spatialisation (comme la balance), etc.

* Structuration de l’interface de l’application



* Mise en forme de l’interface de l’application

* Affichage des notes de musique identifiées



VexFlow est une API de rendu de notation de musique en ligne open source. Il est entièrement écrit en JavaScript, et fonctionne correctement dans le navigateur. VexFlow prend en charge HTML5 Canvas et SVG.

* Algorithme pitch détection

C’est un algorithme conçu pour estimer le pitch ou la fréquence fondamentale d'un signal quasi- périodique ou oscillant, généralement un enregistrement numérique de la parole ou une note musicale ou une tonalité. Cela peut se faire dans le domaine temporel ou le domaine fréquentiel ou les deux domaines. Nous l’avons utilisé pour l’analyse des fréquences des sons audio.

**Pourquoi pitch detect ? :**

Les techniques d'estimation par la fréquence fondamentale peuvent être séparées en deux catégories :

* Des estimateurs basés sur le domaine de temps dont on observe la forme du flux de son entrant.
* Des estimateurs basés sur le domaine de fréquence dont on observe le Spectrum de la fréquence.

On trouve aussi des méthodes qui combinent les deux techniques.

**Les informations extraites par l’algorithme :**

***La note :*** on récupère en notation (abc/sol) les symboles permettant de représenter un fragment de musique par une convention d'écriture de la hauteur et de la durée d'un son.

***Le cent :*** il permet d'indiquer avec précisions les petits écarts entre la hauteur d'une note particulière et une référence donnée. Voir : <https://en.wikipedia.org/wiki/Cent_(music)>

***La durée :*** d'un son musical est le laps de temps pendant lequel on peut identifier ses vibrations (les sons pouvant évidemment durer plus ou moins longtemps).

## Problèmes techniques rencontrés

## Résolutions des problèmes

## Résultat livré au client

# Gestion de projet

## Outils utilisés

* Gestion des versions



* Définition des taches



## Organisation des taches dans le groupe

|  |  |
| --- | --- |
| Taches | Contributeurs |
| Analyse Web Audio | Traoré & Marouane |
| Analyse son & solfèges | Phily & Bouchti |
| Analyse Framework score HTML5 | Rostan |
| Paramétrage de web Audio | Traoré |
| Découpage du flux audio | Marouane & Bouchti |
| Identification d’une note | Traoré & Phily |
| Implémentation du Framework Vexflox | Rostan |
| Design de la page | Traoré |

3. Différence entre planning initial et final

# Conclusion