|  |
| --- |
| CFPT-I |
| WebradioManager |
| Travail de diplôme 2014 |
|  |
| **Simon Menetrey – T.IN E2A** |
| **31/03/2014** |

|  |
| --- |
| [Tapez le résumé du document ici. Il s’agit généralement d’une courte synthèse du document. Tapez le résumé du document ici. Il s’agit généralement d’une courte synthèse du document.] |

# Résumé

## Français

## English

Contenu

[1 Résumé 1](#_Toc384129510)

[1.1 Français 1](#_Toc384129511)

[1.2 English 1](#_Toc384129512)

[2 Introduction 4](#_Toc384129513)

[2.1 Mise en situation 4](#_Toc384129514)

[2.2 Sujet 4](#_Toc384129515)

[2.3 Qu’est-ce qu’une webradio ? 4](#_Toc384129516)

[2.4 Pourquoi ce sujet ? 4](#_Toc384129517)

[3 Cahier des charges 5](#_Toc384129518)

[4 Analyse fonctionnelle 6](#_Toc384129519)

[4.1 Schéma de l’application 6](#_Toc384129520)

[4.2 Interface principale 7](#_Toc384129521)

[4.2.1 Fenêtre d’administration 7](#_Toc384129522)

[4.2.2 Menu principale 7](#_Toc384129523)

[4.3 Gestion de plusieurs webradio 8](#_Toc384129524)

[4.3.1 Création d’une webradio 9](#_Toc384129525)

[4.3.2 Sélection d’une webradio 9](#_Toc384129526)

[4.4 Serveur de diffusion interne et externe 9](#_Toc384129527)

[4.5 Gestion des musiques 10](#_Toc384129528)

[4.6 Gestion des listes de lecture 10](#_Toc384129529)

[4.6.1 Listes de lecture musicales 11](#_Toc384129530)

[4.6.2 Listes de lecture publicitaires 11](#_Toc384129531)

[4.7 Gestion des horaires 11](#_Toc384129532)

[4.8 Gestion des transcoders 11](#_Toc384129533)

[5 Analyse organique 13](#_Toc384129534)

[5.1 Diagramme de classes 13](#_Toc384129535)

[5.2 Base de données 13](#_Toc384129536)

[5.3 Schéma de diffusion 13](#_Toc384129537)

[5.4 ShoutCast 14](#_Toc384129538)

[5.4.1 Présentation 14](#_Toc384129539)

[5.4.2 Serveur 14](#_Toc384129540)

[5.4.3 Transcoder 14](#_Toc384129541)

[5.5 Structures des dossiers/fichiers 14](#_Toc384129542)

[5.6 Webradio 14](#_Toc384129543)

[5.6.1 Création 14](#_Toc384129544)

[5.7 Transcoder 14](#_Toc384129545)

[5.7.1 Fichier de configuration 14](#_Toc384129546)

[5.7.2 Statut 14](#_Toc384129547)

[5.7.3 Gestion des processus 14](#_Toc384129548)

[5.8 Listes de lecture 14](#_Toc384129549)

[5.8.1 Musicale 15](#_Toc384129550)

[5.8.2 Publicitaires 15](#_Toc384129551)

[5.9 Musiques 15](#_Toc384129552)

[5.9.1 Ajout et indexation 15](#_Toc384129553)

[5.9.2 Analyse des tags 15](#_Toc384129554)

[5.10 Serveur de diffusion interne 15](#_Toc384129555)

[5.11 Grille horaire 15](#_Toc384129556)

[6 Plannings 15](#_Toc384129557)

[6.1 Prévu 15](#_Toc384129558)

[6.2 Final 15](#_Toc384129559)

[7 Conclusion 15](#_Toc384129560)

[8 Améliorations possibles 15](#_Toc384129561)

[9 Références 15](#_Toc384129562)

# Introduction

## Mise en situation

Je m’appelle Simon Menetrey, j’ai 20 ans et je suis actuellement en dernière année de formation technicien ES en informatique. J’ai effectué un CFC d’informaticien en 4 ans avant de commencer ma formation actuelle.

Cette documentation concerne mon travail de diplôme réalisé pour ma dernière année en tant que technicien ES en informatique. Ce travail à pour sujet la création d’un gestionnaire de webradio.

Mon professeur de diplôme, Monsieur Garcia, m’a mis en contact avec une entreprise (KTFM) qui recherche un programme de gestion pour leur webradio. En effet, actuellement, c’est un tiers qui s’occupe de la diffusion de leur webradio via des émissions préalablement enregistrées dans leurs studios. Dans l’état actuel, l’entreprise n’a pas un contrôle direct sur la diffusion de son contenu et elle désirerait pouvoir gérer elle-même l’intégralité de leur webradio. Ainsi, elle supprimera un intermédiaire et aura pleinement contrôle de la diffusion.   
En plus de cela, KTFM a aussi besoin de pouvoir avoir une traçabilité des morceaux qu’elle diffuse avec des informations précises afin de pouvoir faciliter le paiement des droits d’auteur à la Suisa[[1]](#footnote-1).  
J’ai donc réalisé le cahier des charges avec KTFM afin de répondre au mieux à leurs besoins.



## Sujet

Gestionnaire de webradio :

* Permettre de diffuser directement depuis le logiciel
* Gérer les morceaux à diffuser/listes de lecture
* Gérer les plages horaires
* Historique de diffusion

## Qu’est-ce qu’une webradio ?

Une webradio est une radio diffusée sur internet via la technologie de lecture en continu. Cette technologie fournit ce que l’on appelle un « flux » que les auditeurs écoutent via leur lecteur multimédia préféré ou via un site web.

## Pourquoi ce sujet ?

Je suis passionné de musique. J’ai aussi toujours recherché un outil simple et gratuit pour gérer une webradio et sa diffusion. J’ai donc imaginé une application, tout-en-un, remplissant ce besoin. J’ai donc l’espoir que mon projet me sera autant utile à moi qu’à l’entreprise KTFM ainsi que de potentielles futures entreprises intéressées.

## Termes

Voici une liste de termes qui seront utilisés dans cette documentation :

* Playlist : Liste de lecture
* Stream : Flux

# Cahier des charges

Ce projet a pour but la création d’un gestionnaire de webradio (de type shoutCAST) complet. Les principales fonctionnalités sont les suivantes :

* Possibilité de gérer plusieurs webradio indépendamment
* Gestion des playlist, horaires et pubs
  + Génération automatique de playlist
  + Génération au format xml
* Gestion des serveurs de diffusions distants et transcoder interne.
* Indexation des fichiers musicaux (tags, chemin sur le disque dur)
* Historique des morceaux joués
  + Génération d’un compte-rendu afin de facilité la gestion des droits d’auteurs
* Serveurs de diffusion interne

Options :

* Si serveur de diffusion interne activé : Serveur WEB interne contenant un mini-site

Plus de détails dans le résumé et le *mindmap*.

Analyses

# Analyse fonctionnelle

## Schéma de l’application



Figure - Schéma application

Lors du lancement de l’application, la fenêtre de sélection de webradio se lance. L’utilisateur choisit la webradio qu’il veut gérer, puis une fenêtre d’administration s’ouvre avec les informations de la webradio sélectionnée. Il peut à tout moment réafficher la fenêtre de sélection et ouvrir une nouvelle fenêtre d’administration avec une autre webradio à gérer.

## Interface principale

### Fenêtre d’administration



Figure - Interface principale

Cette fenêtre est l’interface principale de l’application. Elle est associée à une webradio. Comme montré dans le schéma de l’application, il est possible d’avoir plusieurs fenêtres d’administration ouvertes simultanément, une pour chaque webradio lancée. Les différents onglets sont décrient dans les chapitre suivants.

### Menu principale

TODO : complétement en fonction du menu choisi

## Gestion de plusieurs webradio

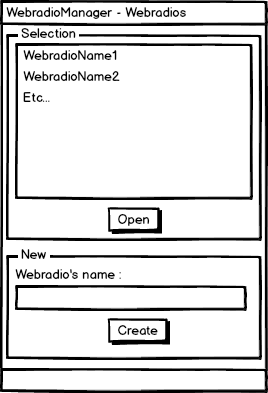


Figure - Interface gestion des webradios

La fenêtre de gestion des webradios s’affiche au démarrage de l’application pour sélectionner ou créer une webradio. Lorsque l’utilisateur clique sur « Open », la fenêtre principale s’ouvre avec les données de la webradio sélectionnée.

L’utilisateur peut gérer autant de webradio en même tant qu’il le souhaite, en effet, à tous moments, il peut ouvrir la fenêtre de gestion des webradios et en sélectionner une autre. Cela ouvrira une nouvelle fenêtre principale, sans pour autant fermer les autres déjà ouvertes, avec les informations de la webradio fraichement sélectionnée.

Chaque webradio possède ses propres listes de lecture, sa propre grille horaire ainsi que ses propre transcoders. Un transcoder est un outils qui permet d’envoyer un flux musical à un serveur de diffusion. Pour plus d’informations, rendez-vous au chapitre concernant [Shoutcast](#_ShoutCast).

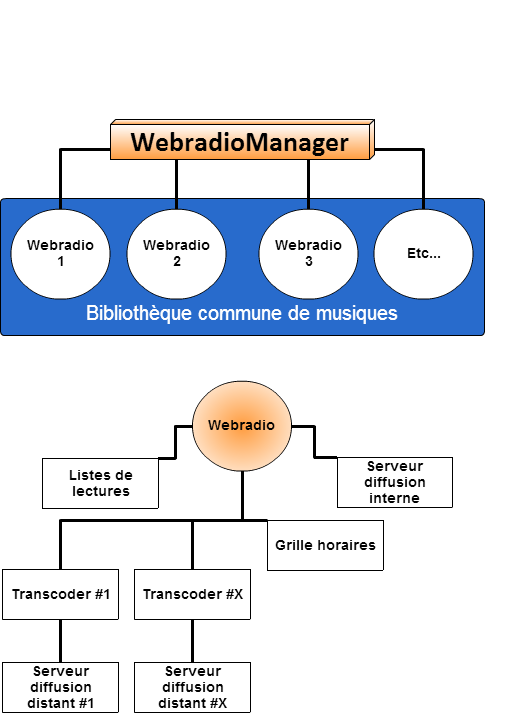


Figure - Schéma webradios

### Création d’une webradio

L’utilisateur entre le nom de la future webradio dans le champ correspondant, puis clique sur « create » pour créer la webradio et ouvrir la fenêtre d’administration liée à la nouvelle webradio fraichement créé.

### Sélection d’une webradio

L’utilisateur sélectionne une webradio parmi la liste proposée, puis clique sur « open » pour ouvrir une fenêtre d’administration liée la webradio séléctionnée.

## Serveur de diffusion interne et externe

Il faut bien différencier les serveurs de diffusion qui sont externes et ceux internes. Les externes sont des serveurs hébergés par un provider[[2]](#footnote-2) (exemple : infomaniak[[3]](#footnote-3) pour KTFM). L’application va donc se servir de ces serveurs pour diffuser la webradio.  
Les serveurs internes sont des serveurs de diffusion hébergés dans l’application elle-même.

## Gestion des musiques/pubs



Figure - Bibliothèque de musiques

Les musiques sont indexées pour former une bibliothèque commune dans l’application. Elle est commune car elle est accessible depuis n’importe quelle webradio créé. L’interface est doublée : Une partie pour les musiques et une autre pour les pubs. Elles sont identiques.

### Affichage

La partie principale de cette interface est l’affichage du contenu de la bibliothèque dans la partie inférieure. Il est affiché sous la forme d’une liste à entrées. Il est possible de sélectionner un ou plusieurs éléments.

### Importer et indexer

L’utilisateur peut importer des fichiers musicaux dans sa bibliothèque. Le bouton « Import from folder… » ouvre une boite de dialogue où l’utilisateur peut sélectionner le dossier à analyser afin d’importer les fichiers musicaux qui y sont présents. C’est la partie dite « d’indexation ». Cette indexation peut être récursive, c’est-à-dire que les sous-dossiers du dossier sélectionné vont être analysés aussi. Une boite de dialogue demande donc à l’utilisateur s’il veut effectuer une analyse récursive.   
Le bouton « Import from files… » permet d’importer un ou plusieurs fichiers sélectionnés manuellement. La sélection s’effectue via une boite de dialogue Windows standard.  
A gauche de ces 2 boutons, une barre de recherche permet d’effectuer une recherche dans la bibliothèque de l’application.  
Le bouton de droite « Delete selected » va supprimer les éléments sélectionnés dans la liste de morceaux affichés en dessous.

### Ajout à une playlist

Pour ajouter des morceaux à une playlist, l’utilisateur peut en sélectionner autant qu’il le souhaite dans la liste d’affichage puis sélectionner une playlist via le menu déroulant situé à droite du bouton « Add selected to ». Ce dernier permet donc de confirmer l’ajout à une playlist.

## Gestion des listes de lecture

La gestion des différentes listes de lecteur se fait dans l’onglet « playlists ».

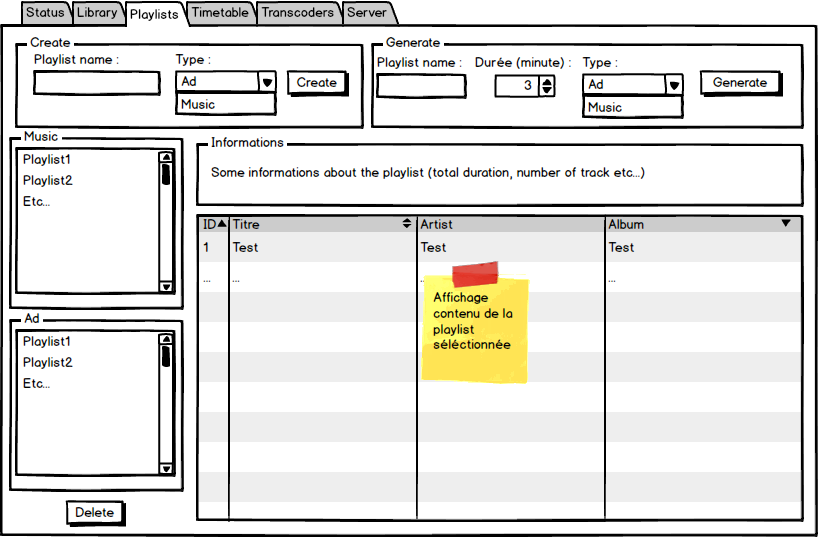


Figure - Onglet "playlists"

### Listes de lecture musicales

Une liste de lecture musicale contient exclusivement des musiques. Son type est « music ».

### Listes de lecture publicitaires

Une liste de lecture publicitaire contient des pubs, des annonces ou des jingles. Son type est « ad ».

### Création

La création d’une playlist se fait via le cadre situé en haut à gauche. Il lui faut un nom et un type. La playlist fraichement créé est ensuite ajouté à l’une des 2 listes situées à gauche en fonction de son type.

### Génération automatique

L’utilisateur a la possibilité de générer automatiquement une playlist en définissant une durée, un nom et un type. Le logiciel va ensuite prendre différentes musiques ou publicité afin de remplir le temps voulu.

### Affichage du contenu d’une playlist

Quand une playlist est sélectionnée (musique ou pub), son contenu est affiché dans la partie centrale de la vue. Au-dessus de cette partie, des informations concernant la playlist sont affichées telle que le temps total de lecture, le nombre de morceaux présents etc.

## Gestion des horaires

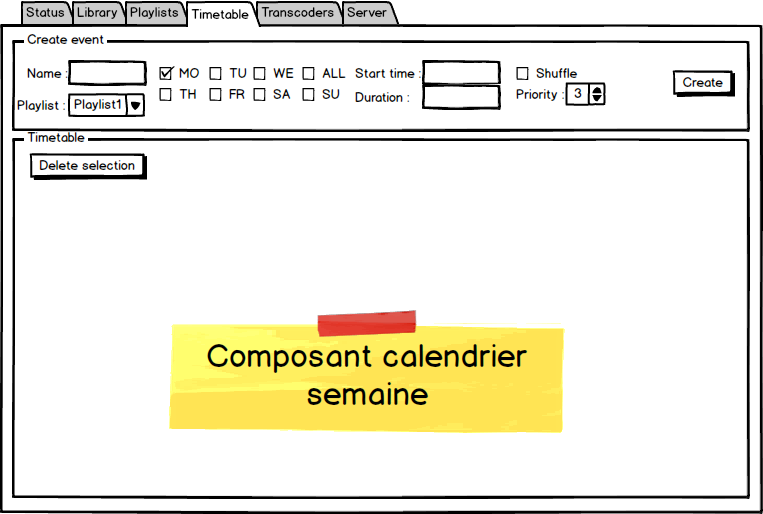


Figure - Onglet "timetable"



Figure - Calendrier

### Événement

### Remplissage manuel

TODO : priorityetc

TODO : clique sur le calandrier de la semaine = change les infos dans la partie supérieur de création en fonction du moment choisi

### Remplissage automatique

### Suppression d’un événement

### Historique

TODO : générer historique

## Gestion des transcoders

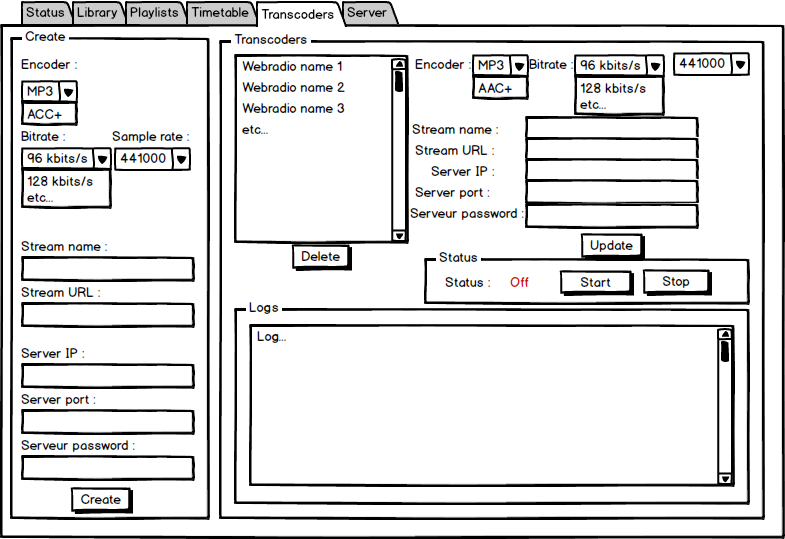


Figure - Onglet "Transcoders"

Cet onglet permet de gérer les différents transcoders d’une webradio.

### Création

La partie de gauche offre la possibilité de créer un nouveau transcoder avec ses différents réglages. Pour des mesures de simplicités, le minimum requis est demandé à l’utilisateur. Les informations suivantes sont demandées :

* Le type de flux (mp3 ou ACC+)
* Le bitrate[[4]](#footnote-4) du flux
* Le sample rate[[5]](#footnote-5)
* Le nom du flux
* L’URL du flux (par exemple : site web de la webradio)
* Adresse IP du serveur
* Port du serveur
* Mot de passe du serveur

### Affichage

Dans la partie droite, la liste des transcoders de la webradio est affichée. L’utilisateur peut cliquer et sélectionner un des transcoder afin de la supprimer à l’aide du bouton « Delete ». Les informations du transcoder sélectionné sont affichées à droite de la liste. Le statut (on ou off) est affiché en dessous de ces dernières. Tout en bas, le log du transcoder est affiché.

## Gestion du serveur

# Analyse organique

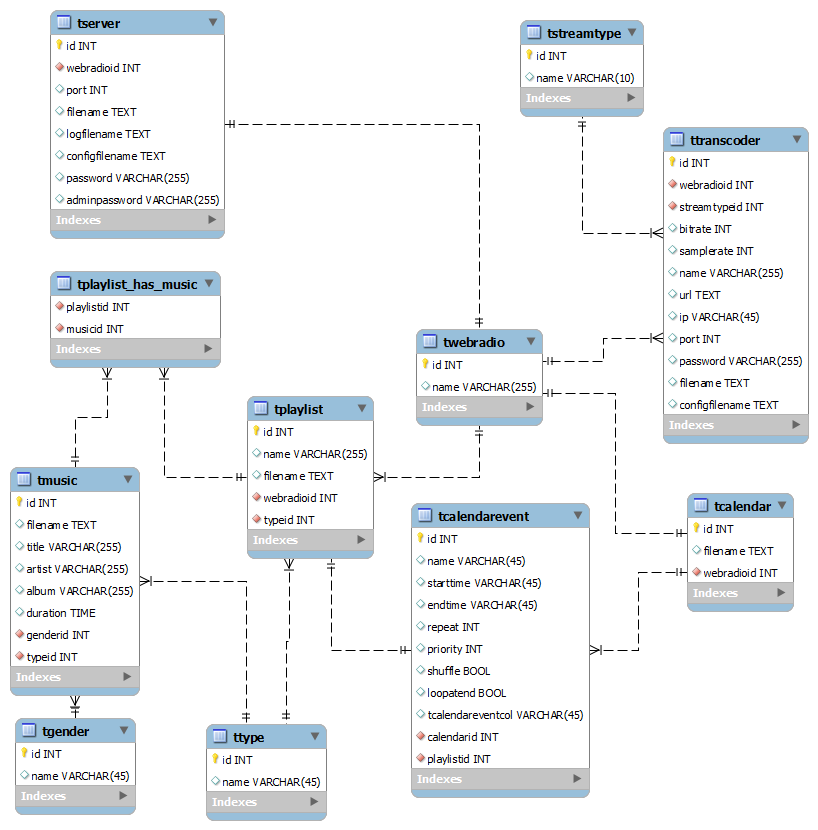
## Diagramme de classes

## Base de données

### SQLite

<http://sqlitestudio.pl/>

### Schéma



## Schéma de diffusion



Figure - Schéma de diffusion



Figure - Schéma de diffusion 2

Ces 2 schémas montrent le fonctionnement de la diffusion des webradio via infomaniak. Le logiciel avec les transcoders sont les « sources audio ». Infomaniak demande seulement un flux audio, peut importer le logiciel utilisé pour le diffuser.

## ShoutCast



Figure - Logo Schoutcast

### Présentation

### Pourquoi cet outil ?

J’ai choisi Shoutcast car il propose des outils en ligne de commande ainsi qu’une gestion facilitée via des fichiers xml ou texte basiques. Cela permet de pouvoir interagir plus facilement avec via des application externe telle que la mienne.

### Serveur

TODO : descriptif

### Transcoder

TODO schema du principe de diffusion avec les transcoder et serveurs

TODO explication du system de fichier xml pour gerer le transcoder etc ( détails dans des nouveau chapitre sur les playlist etc ) et qu’il gère tout seul els calendrier etc

TODO : type de formatd esortie ( mp3 ou aac+ )

## Structures des dossiers/fichiers

TODO : expliquer que pour chaque webradio créé, un sous dossier est crée avec les fichier transcoder dedans etc

TODO : dans la base de donnée, les filenames sont enregistrés.

TODO : expliquer que des fichier de « base » seront déjà présent dans les dossier de l’application pour pouvoir s’en servir comme base a chaque création

TODO : pour une webradio, tous les transcoder pointe sur les meme fichier calandrier et playlist (grace au chemin relatif inscrit dans les fichier de config)

## Webradio

### Création

## Transcoder

### Fichier de configuration

TODO : explication des différents elements du fichier de conf ainsi que les element modifiable par l’utilisateur

### Statut et logs

TODO : se référé au processus ou alors requete ajax dessus pour avoir toutes les informations

<http://wiki.winamp.com/wiki/SHOUTcast_Transcoder_AJAX_api_Specification#LogData>

### Gestion des processus

TODO : avoir une liste de processus où les transcoders sont lancé afin de pouvoir récupéré leur log et statut facilement

## Listes de lecture

TODO : explication avec les fichiers transcodeur

TODO : explication génération xml et génération automatique

### Musicale

### Publicitaires

### Ajout à une playlist

TODO : depuis l’onglet library

TODO : refraichissement de la vue

### Génération automatique

TODO : genre et durée

### Suppression

## Musiques

### Ajout et indexation

TODO : recursivité ou pas

### Tags ID3

### Analyse des tags

### Suppression

TODO : supprimer des playlist

## Serveur de diffusion interne

TODO : explication du principe via schema

## Grille horaire

TODO : expliquer qu’il faudra jouer sur les duration pour générer les playlist et le calendrier

<http://wiki.winamp.com/wiki/SHOUTcast_Calendar_Event_XML_File_Specification#Calendar_Tag>

c’est dans repeat que l’on va définir les jours, chaque joiur = une valeur donc il faudra un tableau de valeur

TODO : event lié à calandrier et a une playlist

TODO : explication avec les fichier transcodeur

<http://calendar.codeplex.com/> utilisation du composant externe

<http://www.devcomponents.com/kb2/?p=404>

### Affichage du calendrier

TODO : avec le composant

TODO : masque de bit comme pour le chmod pour la valeur repeat afin de savoir quel jour est cooncernépar l’event

### Génération automatique

### Génération de l’historique

## Serveur de diffusion

### Log

TODO : affichage depuis fichier log (filename dans bdd)

# Tests

# Plannings

## Prévu

## Final

# Apports personnels

# Conclusion

# Améliorations possibles

* Multi-serveur de diffusion par transcoder
* Capture live

# Références

* <https://cacoo.com> : Création de diagrammes en ligne
* <http://balsamiq.com/> : Création de « mokup »
* <http://calendar.codeplex.com/> : Composant C# pour l’affichage du calendrier sur une semaine
* <http://sqlitestudio.pl/> : Logiciel de gestion de base de données SQLite

# Annexes

1. SUISA est la coopérative des auteurs et éditeurs de musique <http://www.suisa.ch/fr/> [↑](#footnote-ref-1)
2. Fournisseur de services internet [↑](#footnote-ref-2)
3. <http://www.infomaniak.com/> [↑](#footnote-ref-3)
4. Débit binaire : une mesure de la quantité de données numériques transmises par unité de temps. [↑](#footnote-ref-4)
5. Taux d’échantillonnage [↑](#footnote-ref-5)