

Rapport de conception

Adonis NAJIMI,
Valentin STERN,
Vincent ALBERT,
Théo GERRIET

4 janvier 2014

1 Présentation du projet

2 Client C++

2.1 Présentation du client

2.2 UML de l'interface graphique

L'interface graphique sera codée en C++ et respectera le modèle MVC. Ainsi seul le contrôleur modifiera le modèle, tandis que la classe NoSkin, représentant l'interface, ne fera qu'afficher les données. Ces deux classes sont donc représentées sur l'UML et sont complémentées par des classes complémentaires telles que la classe FormatException qui permet de gérer plus finement la conversion JSON/Objet. Les classes TabWidget, Options et Chord seront utilisées par NoSkin pour créer les fenêtres de l'accordeur, des paramètres et le widget particulier de la tablature.

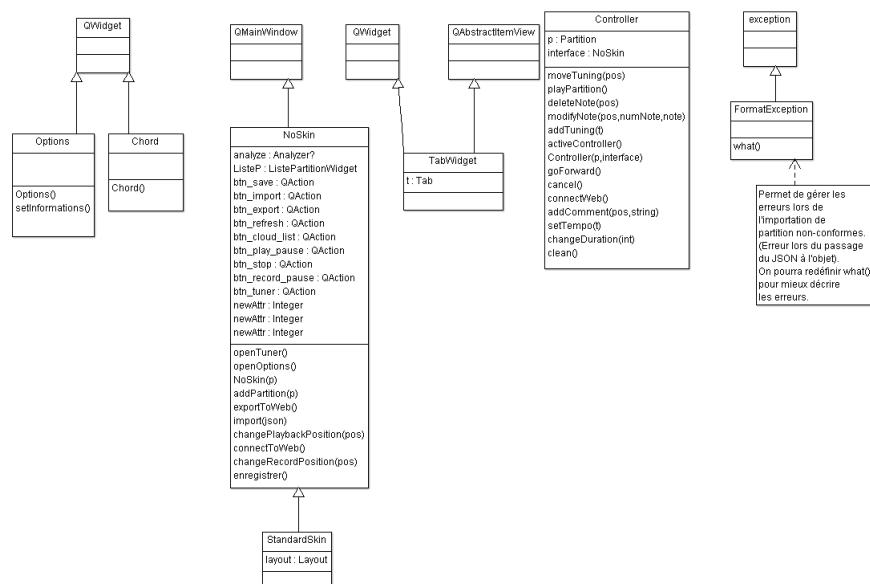


FIGURE 1 – UML de l'interface graphique

2.3 Maquettes du client

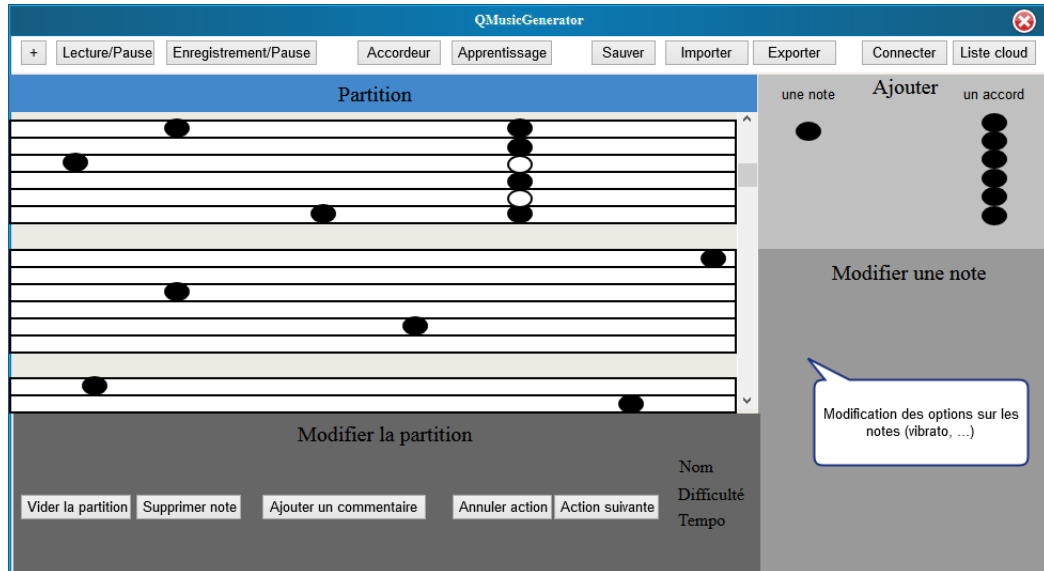


FIGURE 2 – Page principale du client

Cette fenêtre est la fenêtre principale de notre client. A partir de celui-ci, il sera possible de créer des partitions vierges, d'en ouvrir des préexistantes, d'en lire et d'enregistrer des notes à partir de l'entrée audio ainsi que de les sauvegarder.

Mais il sera aussi possible d'ouvrir un accordeur, de télécharger des partitions hébergées sur le site web, et d'en uploader.

Il sera en effet possible de connecter directement le client avec le site à l'aide des identifiants de l'utilisateur.

De plus, il y aura aussi évidemment toutes les options permettant de modifier la partition en cours, tels que son nom, sa difficulté, son tempo, ...

Enfin, on pourra modifier la partition manuellement en déplaçant les notes ou en en ajoutant.

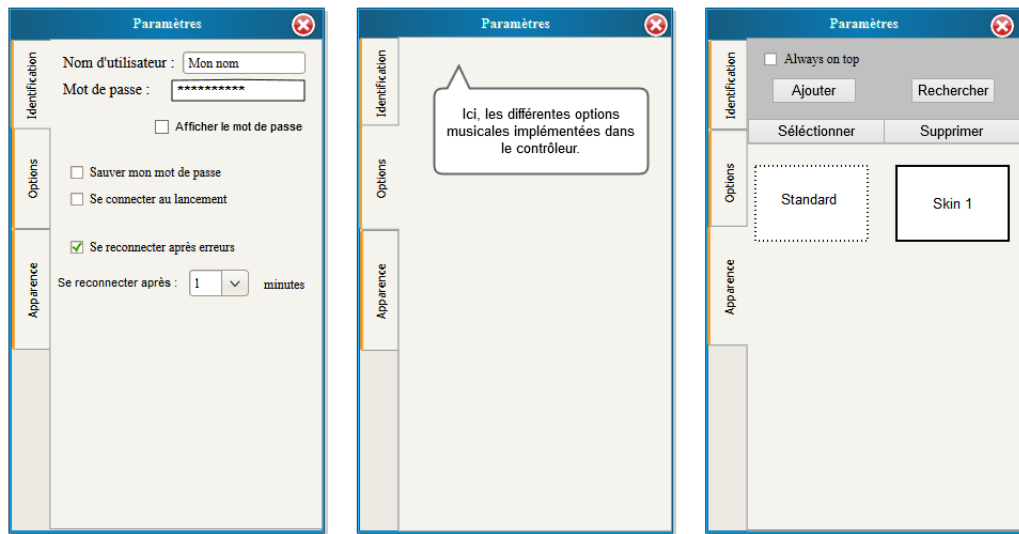


FIGURE 3 – Fenêtre des paramètres du logiciel

Cette fenêtre permettra de modifier les paramètres du logiciel. Cela recouvre les paramètres d'authentification, ceux spécifiques à la synthèse de musique en elle-même, mais aussi l'apparence du client via l'utilisation de skins.

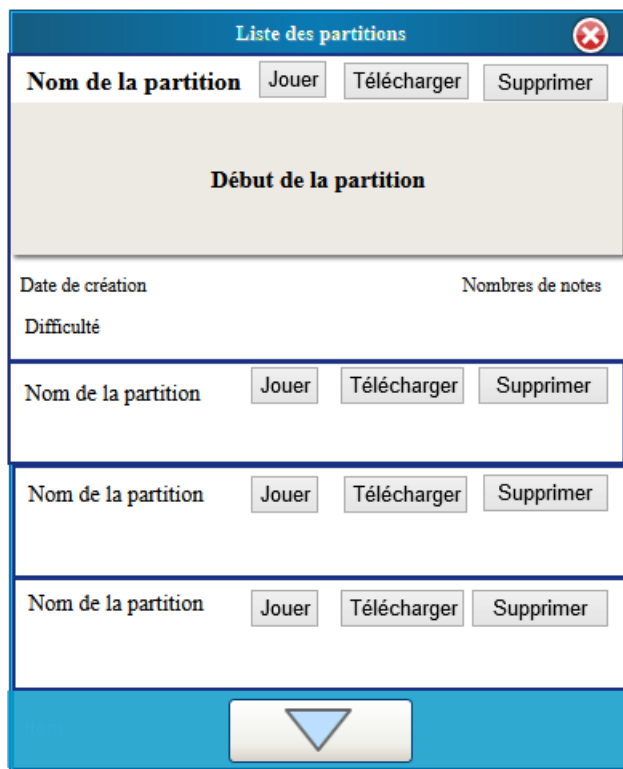


FIGURE 4 – Fenêtre des listes de partitions hébergées sur le site web
 Cette fenêtre listera toutes les partitions que l'utilisateur a uploadé sur le site internet si celui-ci est connecté.
 A partir de cette fenêtre, il pourra soit les supprimer du site web, soit les télécharger, ou encore les jouer.

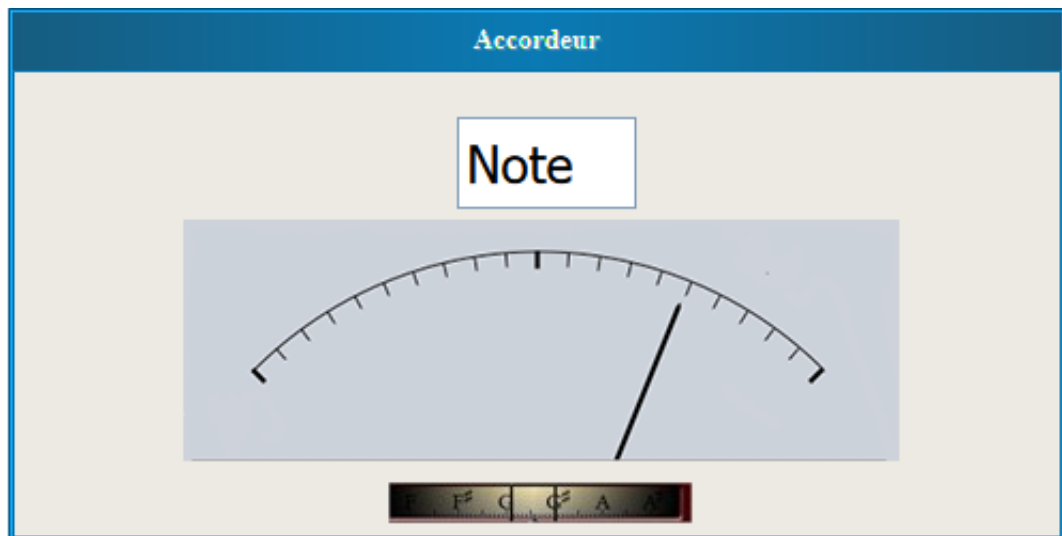


FIGURE 5 – Accordeur

Cette fenêtre affichera un accordeur qui analysera l'entrée sonore pour permettre à l'utilisateur de réaccorder son instrument directement depuis le logiciel.

2.4 Liaison avec l'application web

Afin de communiquer avec l'application web, le client sera doté d'un module réseau qui enverra et recevra des requêtes http. Les partitions seront envoyées et reçues en format JSON afin de faciliter ces échanges.

Le JSON aura format semblable à ceci :

```
{
  "Name" : "string",
  "Tempo" : 0,
  "private" : true/false
  "tunes" : [{
    "volume" : 0-1,
    "nb_notes" : 0,
    "temps" : "croche",
    "notes" : [{
      "name" : "do-la-...",
      "frequence" : 0
    }]
  }]
}
```

3 Application Web

3.1 Maquettes d'écran

Les différentes maquettes du site permettent d'améliorer la vision des besoins de l'application. Ces visuels ne sont pas définitifs et sont simplement là pour

nous aider à mieux structurer l'application et organiser les objectifs nécessaires à la réalisation de celle-ci.

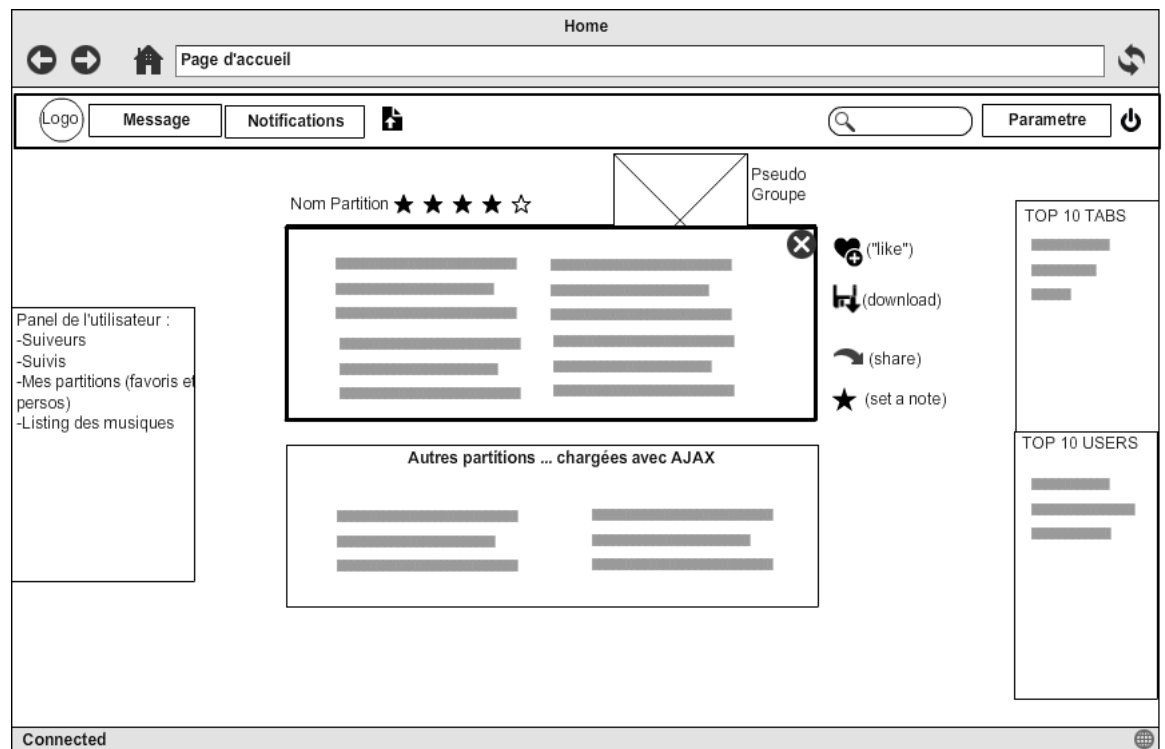


FIGURE 6 – Maquette de la page principale du site

La barre du haut est une barre de menu qui sera disponible sur toute les pages du site, elle n'est pas représentée sur toutes les maquettes par soucis de clarté. En effet, le principal sujet des maquettes suivantes n'est pas la présence de la barre de menu mais le contenu de celles-ci

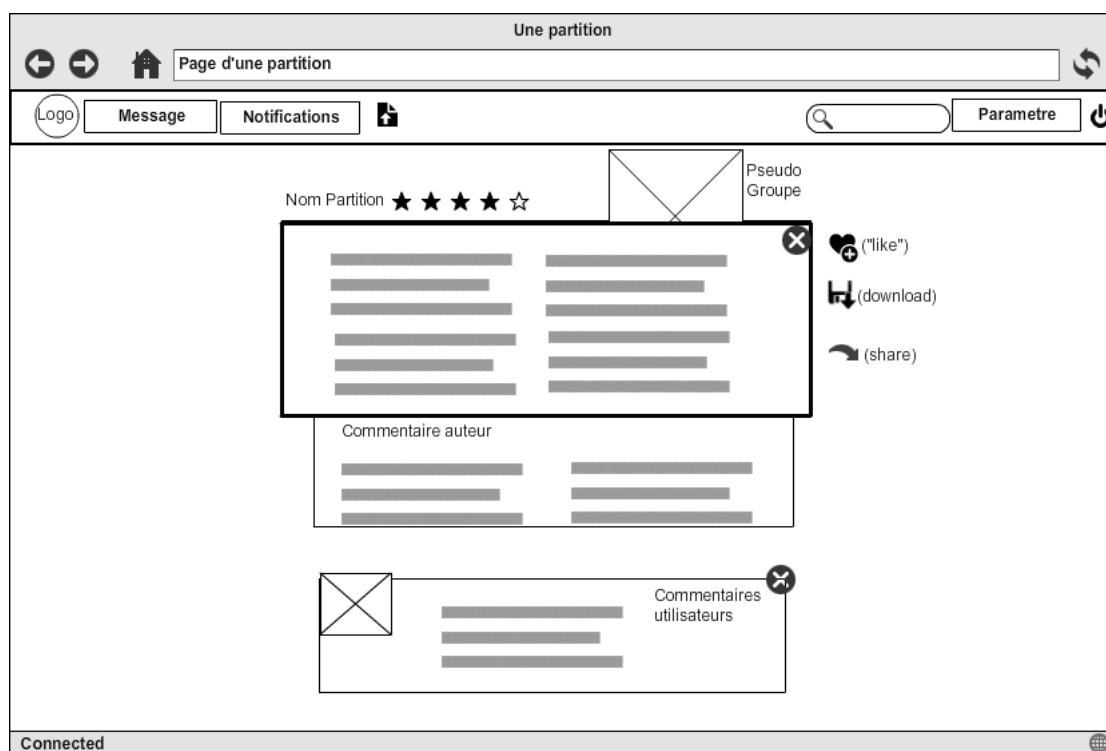


FIGURE 7 – Page principale d’une partition de musique

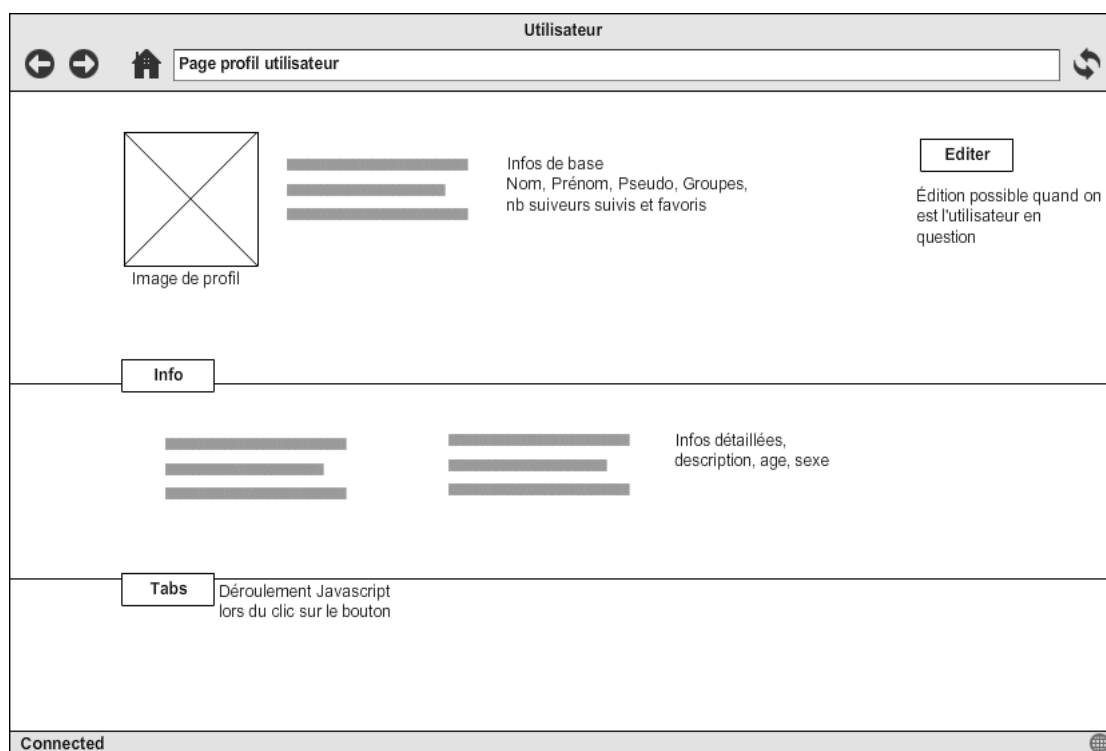


FIGURE 8 – Page de profil utilisateur

Cette page permettra à l'utilisateur concerné de modifier ses informations s'il le souhaite.

Les autres utilisateurs verront sur sa page de profil les informations rendues visibles par l'utilisateur en question.

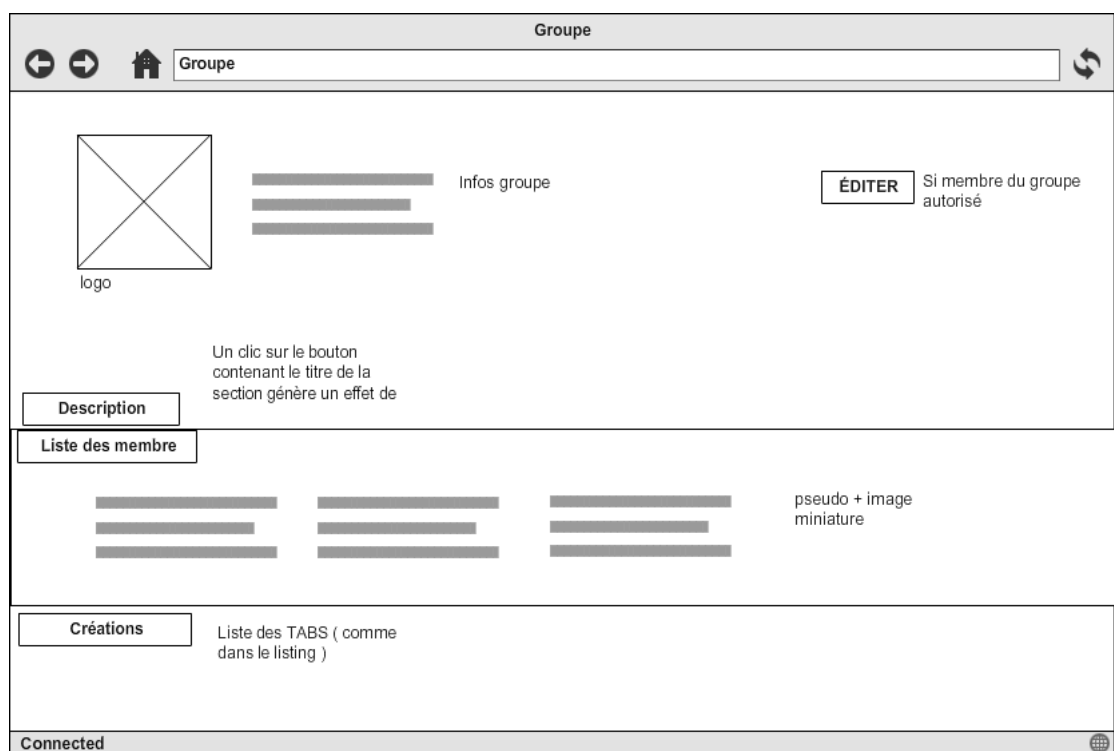


FIGURE 9 – Page d'un groupe de musique
 Cette page permettra de modifier le groupe si l'utilisateur en possède les droits.
 Cette page contiendra la liste des membres du groupe ainsi que les différentes
 partitions liées au groupe concerné.

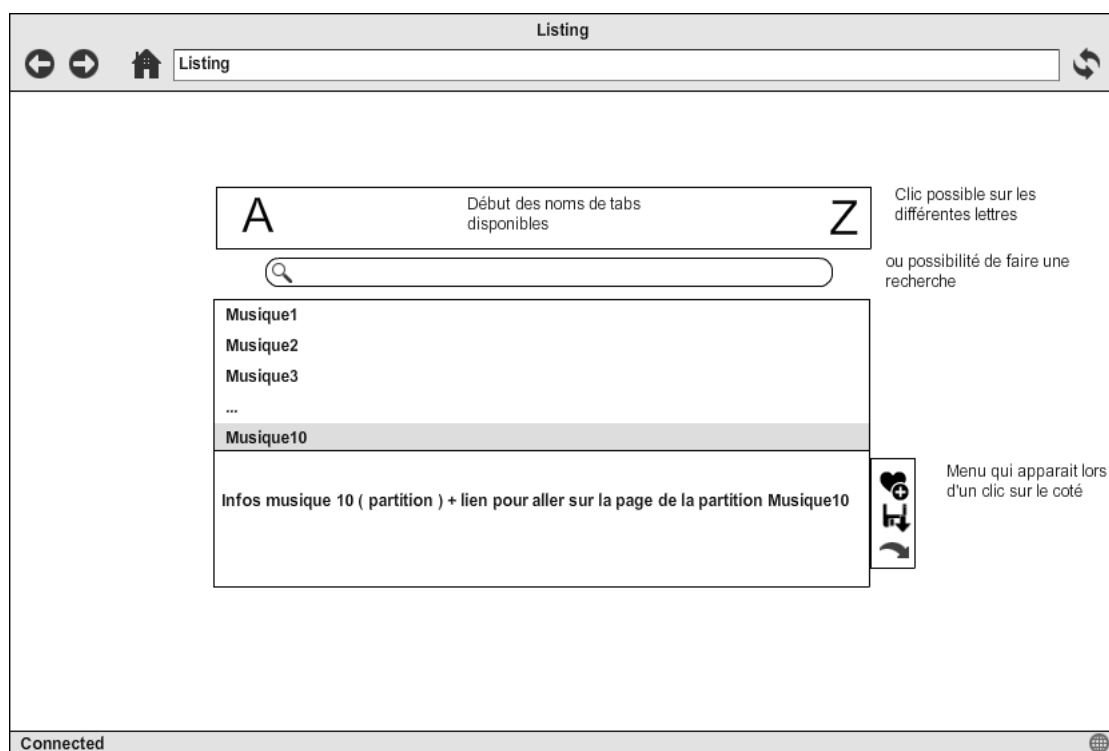
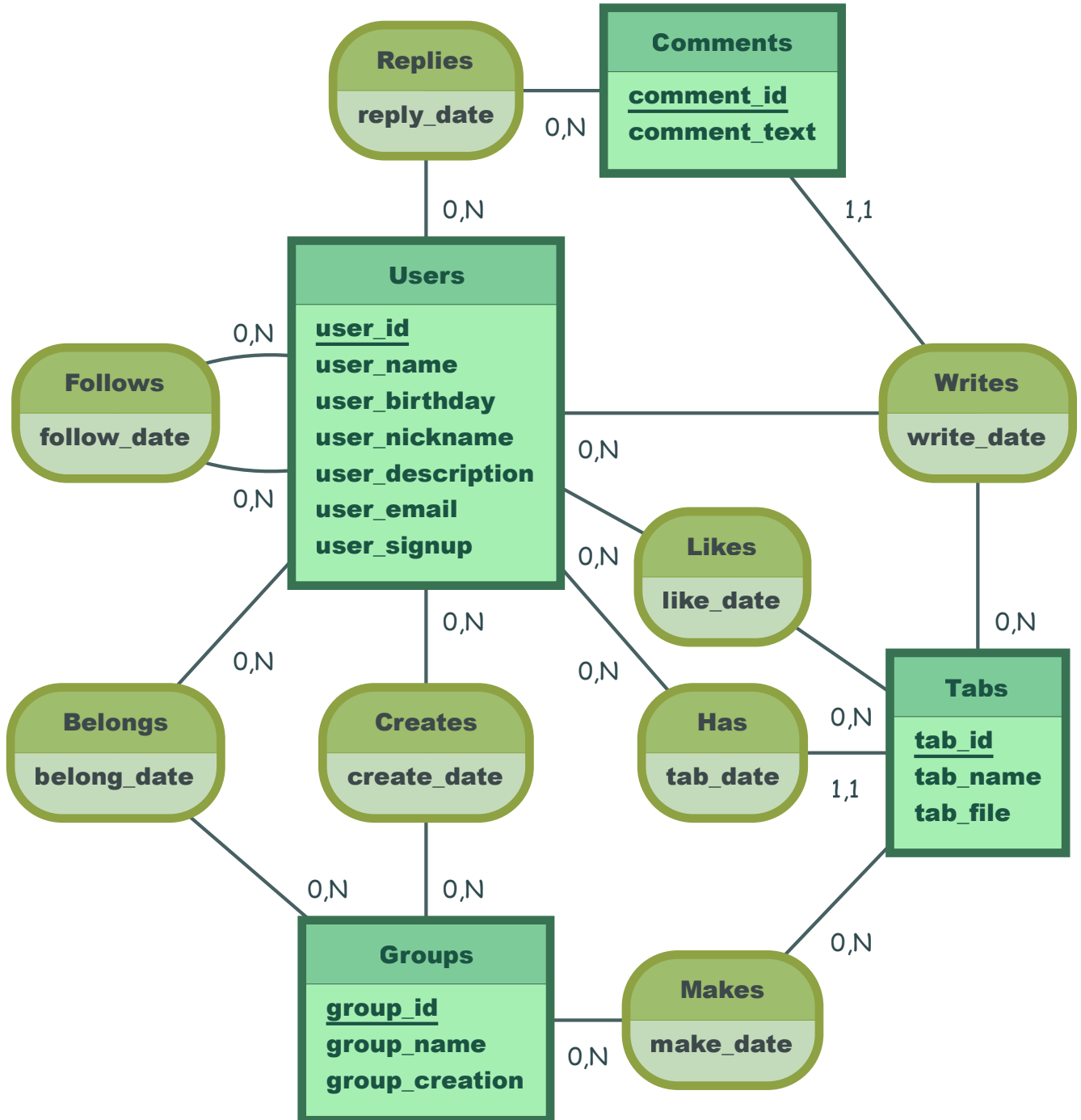


FIGURE 10 – Liste de toutes les partitions
Le listing des partitions permettra une recherche plus approfondie qu’avec la recherche disponible dans la barre de menu (la barre du haut).

3.2 Modèle Merise MCD



3.3 Modèle Merise MLD

Replies (#user_id, #comment_id, reply_date)
Comments (comment_id, comment_text, #tab_id, #user_id, write_date)
Follows (#user_id, #user_id, follow_date)
Users (user_id, user_name, user_birthday, user_nickname, user_description, user_email, user_signup)
Likes (#tab_id, #user_id, like_date)
Belongs (#user_id, #group_id, belong_date)
Creates (#group_id, #user_id, create_date)
Tabs (tab_id, tab_name, tab_file, #user_id, tab_date)
Groups (group_id, group_name, group_creation)
Makes (#group_id, #tab_id, make_date)

3.4 Structure de l'application

—Structures— views : -home //invité ou non -compte // config ou non -groupe // config ou non -listing -barre en haut? (gérer upload) -page partition -partition elle même

controllors (gèrent les animations et DATA ?) ANGULAR : home.js || controleur général compte.js || controleur du compte => gère les effets groupe.js || controleur gère effet listing.js || gèrent effet bare.js partition.js partition-page.js

home.js => gestion des animations + gestion des datas à afficher

DATA => voir base de données Active record => une table = un module -methode de base (findAll,byId,insert,update,delete)+ sorted by ...

les routes express gèrent les chemins de navigations

Routes : chaque lien est une route qui peut contenir une action et qui actionne une méthode du controller lié à la route

Table des matières

1	Présentation du projet	2
2	Client C++	2
2.1	Présentation du client	2
2.2	UML de l'interface graphique	2
2.3	Maquettes du client	3
2.4	Liaison avec l'application web	6
3	Application Web	6
3.1	Maquettes d'écran	6
3.2	Modèle Merise MCD	12
3.3	Modèle Merise MLD	13
3.4	Structure de l'application	13