

MUSIC SHEET WRITER VISION

Jonathan Racaud - Antoine Simon - Jeremy Harrault - Julien Blondeel - Simon Daguenet - Florian Corradin

MUSIC SHEET WRITER

Objectifs du document

Résumé

Ce document est le document vision du projet Music Sheet Writer. Il se destine à montrer la vision globale sur le projet. Les fonctionnalités qui seront et ne seront pas réalisés.

Music Sheet Writer est un logiciel d'édition de partition se voulant innovant en permettant à l'utilisateur de générer une partition depuis son instrument branché à son ordinateur. Accompagné d'un site internet avec une composante communautaire et des applications mobiles, il s'agit donc d'un projet à destination des musiciens compositeurs ou non.

Nous avons deux types d'utilisateurs : les compositeurs, ayant besoin d'outils pour garder une trace de leurs compositions et les musiciens non compositeurs ayant besoin des partitions pour pouvoir rejouer les morceaux tels qu'ils ont été pensés.

Le projet est donc divisé en plusieurs livrables : un logiciel, un site internet et des applications mobiles. Chacun de ces livrables est décomposé en modules logiques regroupant des fonctionnalités appartenant à une même catégorie. Chacun de ces composants sont représentés à l'aide de diagrammes.

Le logiciel est le seul livrable à n'avoir aucune interaction avec les autres. Contrairement au site internet et les applications mobiles. Ces dernières utilisant les services développés sur le site internet pour donner à l'utilisateur final toutes les informations dont il a besoin.

Enfin, étant un projet vaste et dans un domaine avec de nombreux concurrents, de multiples librairies, SDK et autres modules ont déjà été créés. Nous ne nous privons pas de les réutiliser dans le cadre de notre projet. Il s'agit par exemple des librairies de gestion de l'audio et du MIDI ou encore de gestion du format MusicXML.

Description du document

Titre	Music Sheet Writer : VISION
Date	14/04/2015
Auteur	Jeremy Harrault
Responsable	Jonathan Racaud
E-mail	musicsheetwriter_2017@labeip.epitech.eu
Sujet	VISION
Version du modèle	1.0

Tableau des révisions

Date	Auteur	Section(s)	Commentaire
14/04/15	Jeremy Harrault	Toutes	Création du document.
17/04/15	Antoine Simon	4 Besoin du projet	Création de la partie et début de rédaction
18/04/15	Jonathan Racaud	Introduction	Rappel de l'EIP et contexte et périmètre du projet fait.
24/05/15	Simon Daguenet	5.1.1 Le logiciel	Rédaction.
24/04/15	Florian Corradin	5.1.2 Le site internet	Rédaction.
25/04/15	Jeremy Harrault	5.2 Flux et interaction	Ajout du diagramme de déploiement serveur, diagramme de communication logiciel.
25/04/15	Julien Blondeel	5.1.3 Applications mobiles	Rédaction de la partie.
25/04/15	Antoine Simon	4 Besoin du projet	Partie complétée.
26/04/15	Jonathan Racaud	Toutes	Remaniement du plan et rédaction des parties ajoutées.
26/04/15	Jeremy Harrault	Toutes	Relecture et mise en page.

Sommaire

1. Introduction.....	1
1.1. Rappel de l'EIP.....	1
1.2. Contexte et périmètre du projet.....	1
1.3. Définitions, Acronymes et abréviations	1
1.4. Références	2
2. Position du projet.....	3
2.1. Opportunité du marché	3
2.2. Position du problème	3
2.3. Position du projet	3
3. Description des utilisateurs	4
3.1. La démographie des utilisateurs.....	4
3.2. Les profils utilisateurs	4
3.3. Environnement utilisateur	4
3.4. Alternatives et concurrence	4
4. Besoins du projet	5
4.1. Le logiciel	5
4.2. Le site internet	6
4.3. Les applications mobiles	6
4.4. Modules principaux du projet	6
5. Produits et solution	8
5.1. Architecture logique	8
5.1.1. Le logiciel	8
5.1.2. Le site internet	8
5.1.3. Les applications mobiles	9
5.2. Flux et interactions	9
6. Hors périmètre.....	11
7. Choix des technologies	12
7.1. Logiciel.....	12
7.2. Serveur Web et base de données	12
7.3. Les applications mobiles	12

1. Introduction

1.1. Rappel de l'EIP

EPITECH est l'école de l'expertise informatique, transformant une passion en véritable expertise. L'apprentissage à EPITECH est fondé sur une pédagogie par projets, individuels ou en groupe, validant un certain nombre de connaissances et de notions à assimiler. Tout au long de leur cursus, les étudiants se familiarisent avec le milieu professionnel, notamment grâce aux stages en première, troisième et cinquième année d'une période de quatre à six mois. L'école forme les étudiants à s'adapter à des situations inhabituelles avec la mise en place de rush (projets à réaliser sur un week-end, sur des sujets et notions dont les élèves n'ont aucune connaissance) ou le départ à l'international pendant leur quatrième année ; année durant laquelle l'étudiant va devoir faire preuve d'autonomie et de capacité d'adaptation.

Les Epitech Innovative Projects sont des projets à réaliser sur le cycle master du cursus Epitech. Ils sont conçus à la manière d'un véritable projet entrepreneurial, dans toutes ses composantes : business, techno, design & communication. Un EIP est appelé à devenir une start-up viable. Le but de l'EIP est donc de faire découvrir aux étudiants le monde de l'entrepreneuriat en leur demandant de mettre un peu un projet et de le réaliser en faisant face à des difficultés qu'ils n'avaient jusqu'alors pas rencontrées. Le principal obstacle est la gestion de groupe composé de membres dispersés dans des pays différents, faisant face alors aux problèmes de gestion du temps et des zones horaires pour leur quatrième année. Les problématiques de communication et de vente du produit sont aussi abordées.

1.2. Contexte et périmètre du projet

Ce document vision s'applique au projet Music Sheet Writer, qui sera développé par l'équipe Music Sheet Writer.

Music Sheet Writer est un logiciel d'édition de partition permettant à un utilisateur d'écrire une partition directement depuis un instrument branché à l'ordinateur à l'aide d'un câble JACK ou d'une interface audio USB. Il est aussi accompagné d'un site internet et d'applications mobiles.

Le site internet, servira de vitrine pour présenter le projet, mais aussi d'espace pour que la communauté puisse partager librement leurs créations. Les applications mobiles quant à elles, permettront à chaque utilisateur de consulter leurs partitions ou celles des autres.

1.3. Définitions, Acronymes et abréviations

– A –

API : Une API est une interface de programmation qui permet de se "brancher" sur une application pour échanger des données.

– B –

Bibliothèque : Les bibliothèques regroupent un ensemble de fonctions de base utilisées par de nombreux programmes.

– C –

Component : Composant.

– E –

EIP : Epitech Innovative Projects.

– F –

Front-End : interface utilisateur d'un site internet.

– J –

Jack : connectique permettant de transmettre du son entre différents périphériques (exemple : transmettre du son entre un ordinateur et un casque).

– L –

Librairies : ensembles de fonctions mise à disposition du développeur pour faciliter la tâche de développement d'un logiciel, site internet ou application mobile. Équivalent de *Bibliothèque*

– M –

MIDI : format numérique représentant la musique.

MSW : Music Sheet Writer.

MusicXML : format de fichier d'une partition de musique.

– R –

Responsive : méthode de développement du front-end d'un site internet afin que ce dernier s'affiche bien sur toutes les tailles d'écran.

REST : Style d'architecture pour les applications web.

– U –

USB : connectique permettant de transmettre différentes données, notamment du son, entre différents périphériques.

– S –

Solfège : système universel d'écriture de la musique.

1.4. Références

Se retrouvent dans cette section les références vers les différents documents cités dans le présent document.

- *Cahier des charges du projet Music Sheet Writer*: [2017_CDC1_musicsheetwriter.pdf](#)
- *Étude de l'existant* : [2017_EDE_musicsheetwriter.pdf](#)

2. Position du projet

2.1. Opportunité du marché

Ce projet a pour but de proposer une solution différente des autres logiciels d'édition de partitions existant. En effet, ces derniers proposent pour la majorité d'entre eux d'éditer une partition à l'aide du clavier et de la souris, ou bien à l'aide d'un clavier MIDI. Nous visons alors les musiciens souhaitant garder une trace de leurs compositions musicales de manière simple.

2.2. Position du problème

Le problème de	L'écriture d'une partition qui se fait de manière lente par le couple clavier souris et qui peut se montrer fastidieux
Touche	Les musiciens voulant garder une trace de leurs compositions
L'impact est le suivant	Les musiciens peuvent oublier leurs compositions puisque n'ayant pas l'envie d'apprendre à maîtriser un système lourd
Une solution réussie	Permettrait de gagner du temps sur l'écriture de partition et de simplifier la sauvegarde d'une musique.

2.3. Position du projet

Pour	Les musiciens
Qui souhaitent	Garder une trace écrite de leurs compositions musicales de manière simple et intuitive
Music Sheet Writer est	Un logiciel
Qui	Permet de d'écrire et modifier une partition de musique
À la différence des	Logiciels existants
Notre produit	Reconnait les notes jouées depuis l'instrument de l'utilisateur branché à l'ordinateur pour écrire la partition.

3. Description des utilisateurs

Cette section décrit le profil d'un utilisateur de Music Sheet Writer. Il y a deux types d'utilisateurs pour Music Sheet Writer : les musiciens compositeurs et non compositeurs.

3.1. La démographie des utilisateurs

Les musiciens font partie d'une large communauté de personnes qui s'étend à travers le monde et qui bénéficient d'un système d'écriture universel, le solfège. Cependant, tous ne sont pas familiers avec les outils informatiques ou ne connaissent pas forcément ce système d'écriture.

3.2. Les profils utilisateurs

Compositeurs :

Les compositeurs ont besoin de garder une trace écrite de leurs compositions pour diverses raisons, que ce soit le partage de leur travail ou le dépôt de ce dernier afin de le protéger et de le vendre. Ils ne connaissent pas forcément le solfège.

Non compositeurs :

Les musiciens qui ne composent peuvent avoir besoin des partitions des morceaux afin de pouvoir rejouer lesdits morceaux.

3.3. Environnement utilisateur

Music Sheet Writer est un logiciel qui s'installe sur l'ordinateur de l'utilisateur. Ce dernier pourra le télécharger depuis le site internet officiel de Music Sheet Writer. Les applications mobiles et le site internet seront accessibles depuis n'importe quel accès à internet.

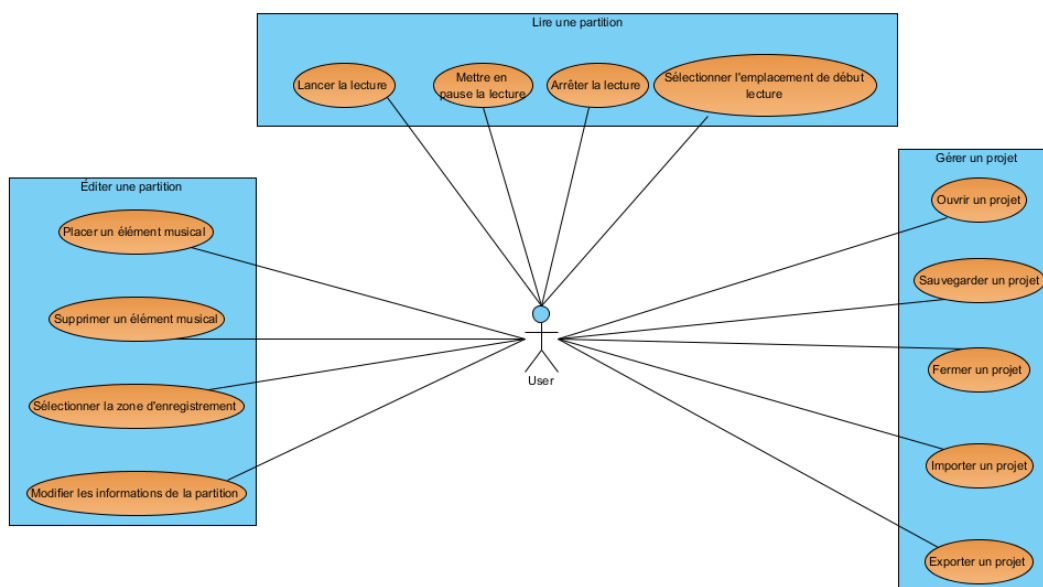
3.4. Alternatives et concurrence

De nombreux logiciels d'édition de partitions existent déjà, mais ne propose pas les fonctionnalités de création de la partition depuis l'instrument de l'utilisateur. Nous avons donc décidé de la développer.

4. Besoins du projet

Nous décrivons dans cette section les différents besoins principaux du projet. Le projet menant à la réalisation de plusieurs produits, lesdits besoins sont regroupés par produit. Les besoins décrits sont la description des fonctionnalités exprimées à un haut-niveau. Pour avoir la liste détaillée des fonctionnalités, veuillez-vous référer au cahier des charges disponible dans la section 1.4 de ce document.

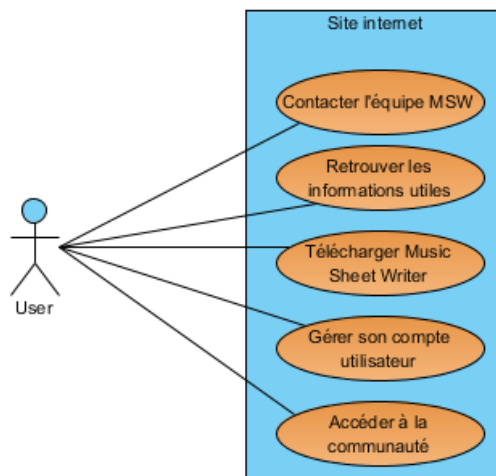
4.1. Le logiciel



L'utilisateur doit :

- Pouvoir lire une partition :
 - Pouvoir lancer la lecture
 - Mettre en pause
 - Arrêter la lecture
 - Sélectionner l'emplacement de début de lecture
- Pouvoir éditer une partition
 - Pouvoir placer un élément musical
 - Pouvoir supprimer un élément musical
 - Sélectionner la zone d'enregistrement
 - Modifier les informations de la partition
- Pouvoir gérer un projet
 - Pouvoir ouvrir un projet
 - Pouvoir sauvegarder un projet
 - Pouvoir fermer un projet
 - Pouvoir importer un projet
 - Pouvoir exporter un projet

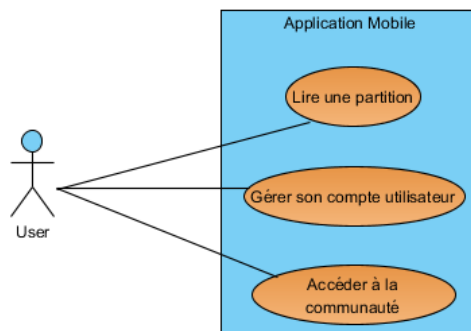
4.2. Le site internet



L'utilisateur doit :

- Contacter l'équipe de Music Sheet Writer
- Pouvoir se renseigner sur Music Sheet Writer
- Pouvoir télécharger Music Sheet Writer
- Pouvoir gérer son compte utilisateur
- Pouvoir accéder à la communauté Music Sheet Writer

4.3. Les applications mobiles



L'utilisateur doit :

- Pouvoir lire une partition (il s'agit du même besoin que celui du logiciel)
- Pouvoir gérer son compte
- Pouvoir accéder à la communauté

4.4. Modules principaux du projet

Afin de réaliser ce projet et les différents livrables le composant, nous nous basons sur certains modules existant pour nous aider dans cette tâche. Voici la liste de ces modules :

- Gestion de l'audio et du MIDI : Librairie PortAudio et PortMIDI

- Gestion du format MusicXML : libMusicXML
- Bases de données : MySQL et MariaDB
- Serveur : Apache
- Interface utilisateur logiciel : Qt
- Sécurité du site internet : Protocole HTTPS et certificats SSL

5. Produits et solution

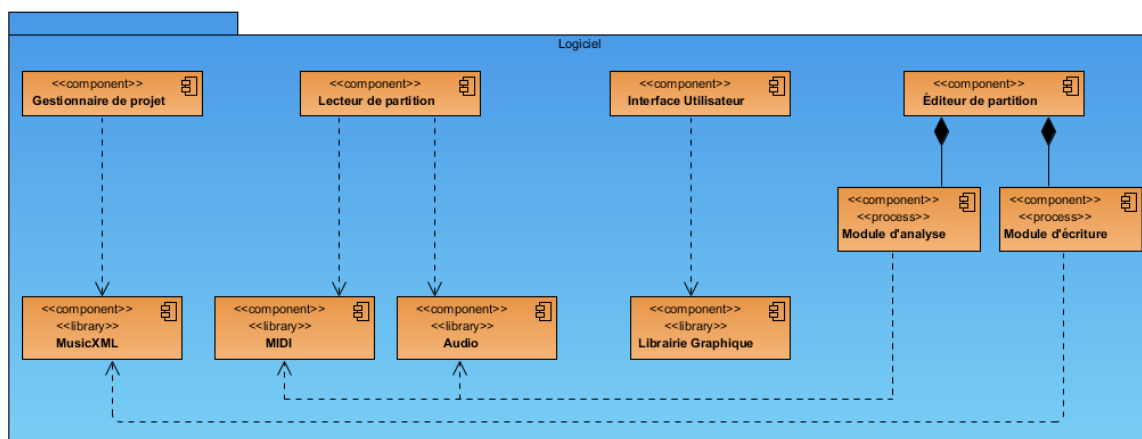
5.1. Architecture logique

Dans cette section sont décrits les différents composants de chacun des livrables du projet.

5.1.1. Le logiciel

Le logiciel est le cœur principal du projet Music Sheet Writer. Ce dernier permet la création et la modification de partitions de musique. Il regroupera toutes les fonctionnalités majeures présentes chez la concurrence et ajoutera la modification et création d'une partition depuis l'instrument de l'utilisateur branché à son ordinateur à l'aide d'un câble JACK ou d'une interface audio USB.

Architecture du logiciel PC

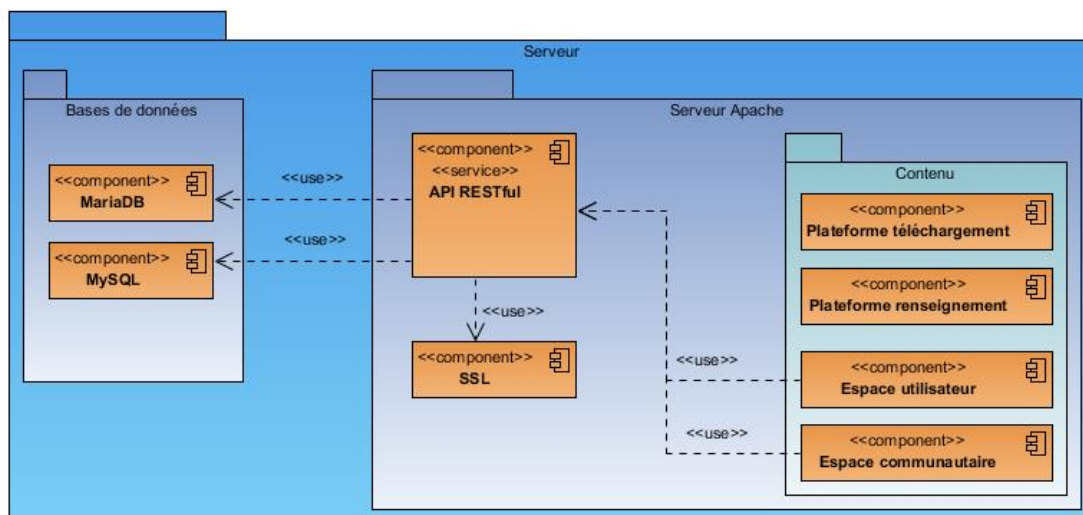


5.1.2. Le site internet

Le site internet est le second produit principal du projet Music Sheet Writer. Ce dernier permettra à un utilisateur de se renseigner, de télécharger le logiciel Music Sheet Writer et de partager ses travaux. En effet, le site internet se décompose en trois composants principaux : la vitrine, l'espace utilisateur et communautaire, l'espace d'administration.

La vitrine permettra à l'utilisateur de trouver les informations sur le logiciel et les applications mobiles à développer. L'espace utilisateur et communautaire sera l'aspect communautaire du projet Music Sheet Writer.

Architecture du serveur WEB

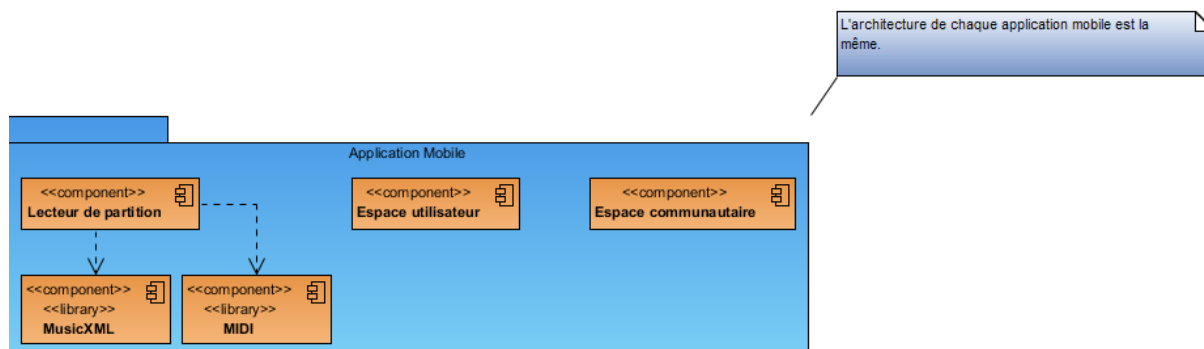


5.1.3. Les applications mobiles

Les applications mobiles sont les derniers livrables du projet Music Sheet Writer et permettent à l'utilisateur de manager son compte MSW ainsi que d'accéder à la communauté MSW. Il pourra aussi retrouver les partitions mises en favoris ou partagées sur son compte afin de les consulter n'importe où.

Les applications mobiles seront disponibles sur les plateformes iOS, Android et Windows Phone et comporteront la même architecture et les mêmes fonctionnalités. Bien qu'elles soient développées en natif sur chacune de leurs plateformes respectives.

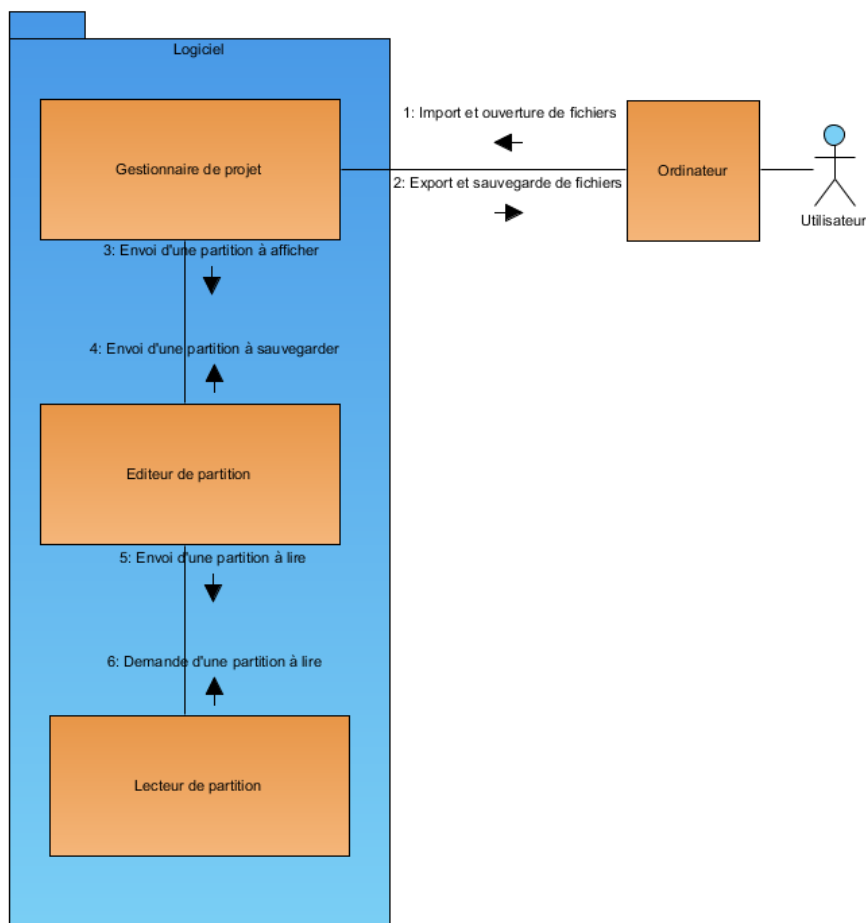
Architecture des applications mobiles



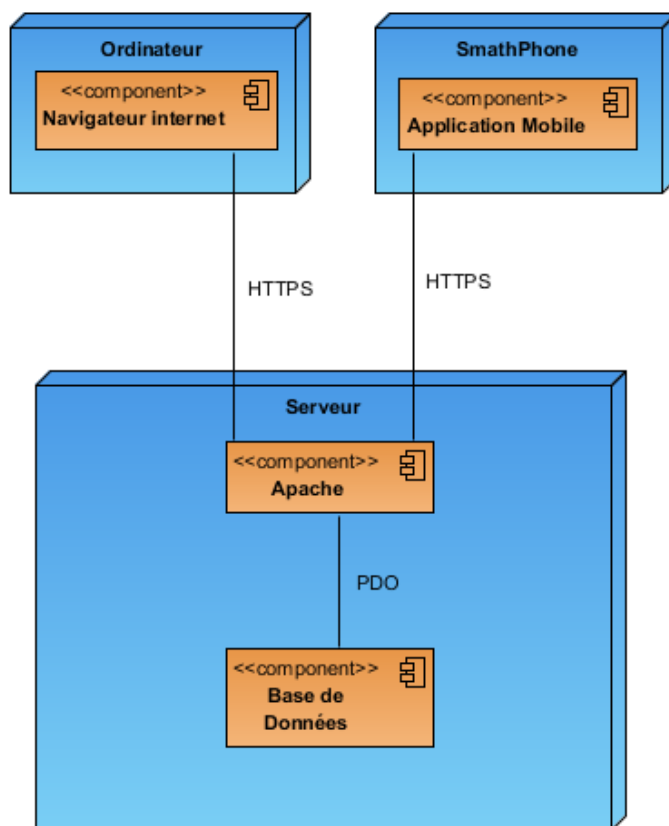
5.2. Flux et interactions

Cette section reprend l'architecture décrites précédemment et établit les interactions existantes entre elles.

Interaction des composants du logiciel



Interaction avec le serveur WEB



6. Hors périmètre

Ne sera pas réalisés dans ce projet :

- Un système de partage des partitions intégré au logiciel
- Un système d'édition de partition depuis les applications mobiles
- Un système de stockage des partitions sur le site internet sans le partage de ces dernières
- Un système d'abonnement premium sur la partie communautaire du projet

7. Choix des technologies

7.1. Logiciel

Le logiciel sera présent sur les plateformes Windows et Mac OS. Le choix de ces plateformes est déterminé par l'aspect cross-plateforme du langage, des librairies audio et MIDI ainsi que de la librairie MusicXML que nous utiliserons lors du développement.

7.2. Serveur Web et base de données

Le serveur Web sera accessible depuis n'importe quel accès internet et sera compatible avec les bases de données suivantes : MySQL et MariaDB. Choix effectué pour répondre à la contrainte « le serveur Web doit être compatible avec plusieurs bases de données ».

	MariaDB	MySQL	PostgreSQL	MSSQL
Compatibilité PHP5 et PDO	Native	Native	Native	Nécessite un Driver ODBC
Syntaxe SQL	Identique à MySQL	Identique à MariaDB	Unique	Unique

Le serveur hébergeant le site internet sera par ailleurs un serveur Apache. Les critères déterminant sont pour les bases de données, leur syntaxe similaire ; pour le serveur, les nombreux plugins, la communauté, sa robustesse mais surtout son langage de programmation : PHP.

	Apache2 HTTP Server	Apache Tomcat	Node.js
Langage utilisé	PHP	Java	JavaScript
Familiarité avec le serveur	Bonne	Moyenne	Mauvaise

7.3. Les applications mobiles

Les applications mobiles seront développées en natif sur chacun des plateformes iOS, Android et Windows Phone soit respectivement Objective-C, Java, C#. Cela afin de pouvoir bénéficier de l'ensemble de fonctionnalités prises en charge par chacune des plateformes et d'adapter l'interface utilisateur aux paradigmes de chaque plateformes. Ce qui n'aurait pas été le cas avec la réalisation d'applications ayant un code commun.