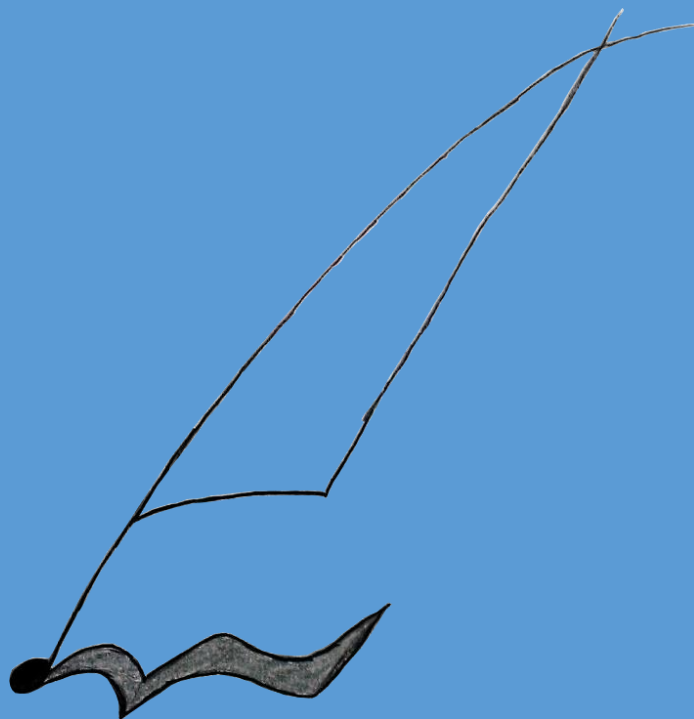




MUSIC SHEET WRITER SPÉCIFICATION DE L'API



J. Harrault

MUSIC SHEET WRITER

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Objectifs du document</i>	



Objectifs du document

Résumé

Ce document spécifie l'API de Music Sheet Writer. Cette API respecte une architecture REST et est utilisée communément par le site internet et les applications mobiles afin de stocker, récupérer et modifier des informations relatives à l'aspect communautaire de Music Sheet Writer.

L'aspect sécuritaire est respecté grâce à la mise en place de rôles et permissions associés aux différents utilisateurs. Chaque rôle donne accès à un ensemble de fonctionnalités au sein de l'API. Ces fonctionnalités sont accessibles en appelant l'API via des requêtes HTTP.

Chacune de ces requêtes est composée d'une URL, d'une méthode HTTP et optionnellement d'un corps et génère une réponse renvoyée par l'API. Ces réponses peuvent signaler soit la bonne réaction de l'API vis-à-vis d'une requête soit un problème au niveau du serveur ou de la requête HTTP.

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Objectifs du document</i>	

Glossaire

– A –

API (Application Programming Interface) : une API est un ensemble d'artéfacts servant de façade pour permettre à un logiciel d'offrir des services à d'autres logiciels. C'est une porte d'accès à des fonctionnalités en faisant abstraction de leur fonctionnement.

– C –

Client HTTP : un client HTTP est un logiciel conçu pour se connecter à un serveur HTTP, comme les navigateurs internet par exemple.

– H –

HTTP (*Hypertext Transfer Protocol*) : HTTP est un protocole de communication client-serveur. Il peut fonctionner sur n'importe quelle connexion fiable. Les clients HTTP les plus connus sont les navigateurs Web permettant à un utilisateur d'accéder à un serveur contenant les données.

HTTPS (*Hypertext Transfer Protocol Secure*) : HTTPS est l'agrégation du HTTP et d'une couche de chiffrement. Il garantit la fiabilité du serveur en assurant notamment une sécurité et une confidentialité des données envoyées par l'utilisateur.

– J –

JSON (JavaScript Object Notation) : est un format de données textuelles dérivé de la notation des objets du langage JavaScript.

– R –

REST (*Representation State Transfer*) : REST est un style d'architecture logiciel qui, la plupart du temps, est utilisé pour implémenter un Web Service. La communication se fait donc entre un client et serveur et utilise le protocole HTTP.

Ressource (REST) : Une ressource est une entité identifiée au sein d'une arborescence.

– S –

Serveur HTTP : un serveur HTTP est un logiciel servant des requêtes à un ou plusieurs clients en respectant le protocole HTTP.

– U –



URI (Uniform Resource Identifier) : une URI est l'identifiant d'une ressource sur un réseau.

URL (Uniform Resource Locator) : une URL est l'emplacement d'une ressource sur un réseau. Une URL est un sous-ensemble d'URI.

URN (Uniform Resource Name) : une URN est le nom d'une ressource sur un réseau. Une URN est un sous-ensemble d'URI.

– W –

Web Service : un Web Service est un programme informatique de la famille des technologies web permettant la communication et l'échange de données entre applications. Il s'agit donc d'un ensemble de fonctionnalités exposées sur internet ou sur un intranet.



 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Description du document</i>	

Description du document



Titre	Music Sheet Writer : Spécification de l'API
Date de création	24/10/2015
Date de publication	25/10/2015
Auteur	J. Harrault
Responsable	Jonathan Racaud
E-mail	musicsheetwriter_2017@labeip.epitech.eu
Sujet	Spécification de l'API
Version du modèle	2.0

Tableau des révisions

Date	Auteur	Section(s)	Commentaire
24/10/2015	Jeremy HARRAULT	Toutes	Création du document
25/10/2015	Jeremy HARRAULT	Toutes	Première version
30/10/2015	Jeremy HARRAULT	3.2. Interface (/login)	Changement de « username » à « login »
02/11/2015	Jeremy HARRAULT	3.2. Interface (/test)	Ajout de la route /test
12/11/2015	Jeremy HARRAULT	Glossaire 3.2. Interface (/users GET)	Ajout de « Ressource (REST) » Limite les utilisateurs renvoyés aux comptes non clôturés.
16/11/2015	Jeremy HARRAULT	2.2. Vue en ressource 3.2. Interface (/contact POST)	Le contact est possible en mode visiteur.
18/11/2015	Jeremy HARRAULT	3.2. Interface (/users GET) 3.2. Interface (/users/[id] GET) 3.4.2. Liste des erreurs	Correction « status != CLOSED » Ajout explication sur « is_favourite » Ajout des erreurs GLO-USERNOTFOUND, GLO-SCORENOTFOUND, GLO-SUBNOTFOUND, GLO-FAVNOTFOUND
20/11/2015	Jeremy HARRAULT	3.2. Interface (/users/[id] GET)	Ajout de « is_subscription »
22/11/2015	Jeremy HARRAULT	3.2. Interface	« is_subscription » et « is_favourite » ne sont 0/1 mais true/false.
25/11/2015	Jeremy HARRAULT	3.2 Interface (/users/[id] PUT)	Correction du JSON sur le retour 200 de la requête
26/11/2015	Jeremy HARRAULT	3.2 Interface (/logout)	Changement de permission (MEM => VIS)
05/12/2015	Jérémy HARRAULT	3.2 Interface 3.2 Interface (/users/[id]/scores/own/[id] PUT)	Reponse vide et status code à 204 pour les requêtes DELETE Ajout de l'URL
11/01/2016	Jeremy HARRAULT	3.2. Interface (/users/[id]/password PUT)	Modification de route et méthode pour le changement de mot de passe
06/02/2016	Jeremy HARRAULT	2.2. Vue en ressources 3.2. Interface (/users/[id]/scores/own POST, /scores/[id] GET et /scores/[id]/preview GET) 3.3. Le téléchargement de fichier 3.4.2. Liste des erreurs	Ajout de la ressources /scores/[id]/preview GET Modification de la réponse de /scores/[id] GET. Modification du content-type autorisé pour cette ressource /users/[id]/scores/own POST

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Tableau des révisions</i>	

Date	Auteur	Section(s)	Commentaire
			Ajout du content-type <i>content_type_msw</i>
13/02/2016	Jeremy HARRAULT	3.2 Interface (/users POST et /users/[id]/personal_data PUT)	Ajout des explications sur firstname et surname Suppression du champ "photo" en entrée de la requête.
15/02/2016	Jeremy HARRAULT	2.2. Vue en ressources 3.2. Interface (/scores/[id] GET et /scores/[id]/preview GET) 3.4. Liste des erreurs 3.3. Le téléchargement de fichier	Suppression des requêtes /scores/[id] GET et /scores/[id]/preview GET Modification de l'objet « score » : « location » => « location_project » et « location_preview » Ajout de l'erreur « SCO-FAVALREADYADD » Suppression du content-type pour les partitions
21/02/2016	Jeremy HARRAULT	3.2. Interface	Ajout des champs de notoriété pour les utilisateurs, abonnements, abonnés et les partitions

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Sommaire	

Sommaire



1. Présentation de l'API.....	1
1.1. But de l'API	1
1.2. Choix architecturaux et composants.....	1
1.2.1. REST	1
1.2.2. Le Framework Symfony	1
2. Les besoins métiers	2
2.1. Utilisateurs et permissions.....	2
2.1.1. Les utilisateurs	2
2.1.2. Permissions	3
2.2. Vue en ressources.....	3
3. Implémentation	6
3.1. Les règles métiers du serveur	6
3.2. L'interface	6
3.3. Le téléchargement de fichier	25
3.4. Les erreurs.....	25
3.4.1. Le formatage	25
3.4.2. Liste des erreurs.....	26

Liste des Tableaux

Tableau 1 : Composition d'une requête-réponse HTTP	1
Tableau 2 : Description des utilisateurs de l'API	2
Tableau 3 : Description des différents niveaux de privilège	3
Tableau 4 : Vue en ressources de l'API.....	5
Tableau 5 : Règle métier de l'API	6
Tableau 6 : Détails des entrées-sorties des routes de l'API	25
Tableau 7 : Les différents "Content-Type" acceptés par l'API.....	25
Tableau 8 : Liste des erreurs possibles renvoyées par l'API.....	27

Liste des Figures

Figure 1 : Cas d'utilisation en fonction des utilisateurs de l'API	2
Figure 2 : Schémas des erreurs renvoyées par l'API.....	25

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Présentation de l'API</i>	

1. Présentation de l'API

1.1. But de l'API

L'API Music Sheet Writer rend accessible les fonctionnalités de partage et d'accès à la communauté à tous les clients HTTP à savoir les applications mobiles sur iPhone, Android et Windows Phone ainsi qu'aux navigateurs internet. L'unicité de l'API et le regroupement des fonctionnalités au sein d'un seul et même composant facilite la maintenance de l'application.

1.2. Choix architecturaux et composants

1.2.1. REST

L'API respecte une architecture REST garantissant une modularité élevée et une facilité d'implémentation. REST respecte une interface uniforme sans tenir compte d'un contexte particulier dans la requête envoyée. L'interface d'une architecture REST se définit par un ensemble de ressources indépendant permettant d'accéder ou d'effectuer des opérations sur des données spécifiques. Chaque requête s'effectue sur une et une seule ressource. L'opération à effectuer se détermine par la méthode HTTP. Enfin, des données supplémentaires peuvent être envoyées au serveur dans le « corps » de la requête. Une fois reçue, le serveur traite la requête et renvoie une réponse HTTP. Cette réponse est notamment caractérisée par un code de terminaison HTTP, définissant le type de réponse renvoyée, et, comme la requête, de données optionnelles dans le « corps » de la réponse.

Composition d'une :

<u>Requête</u>	<u>Réponse</u>
<ul style="list-style-type: none"> - URI - Méthode HTTP - donnée dans le corps 	<ul style="list-style-type: none"> - Code de terminaison - données dans le corps

Tableau 1 : Composition d'une requête-réponse HTTP

1.2.2. Le Framework Symfony2

TODO

Pourquoi utiliser symfony2 ? Avantage pour le projet ?

Quelle sont les fonctionnalités de symfony2 utilisées dans le projet ? (Élément de réponse : Gestion des routes, gestion des permissions, abstraction de la couche données et liaison avec la BDD, gestion de l'authentification de l'utilisateur)

2. Les besoins métiers

2.1. Utilisateurs et permissions

2.1.1. Les utilisateurs

Il existe plusieurs rôles parmi les utilisateurs de l'API.

Nom	Description
Visiteur	- Un utilisateur non connecté sur le système
Membre	- Un utilisateur connecté avec un compte Music Sheet Writer
Administrateur	- Un utilisateur connecté avec un compte administrateur Music Sheet Writer

Tableau 2 : Description des utilisateurs de l'API

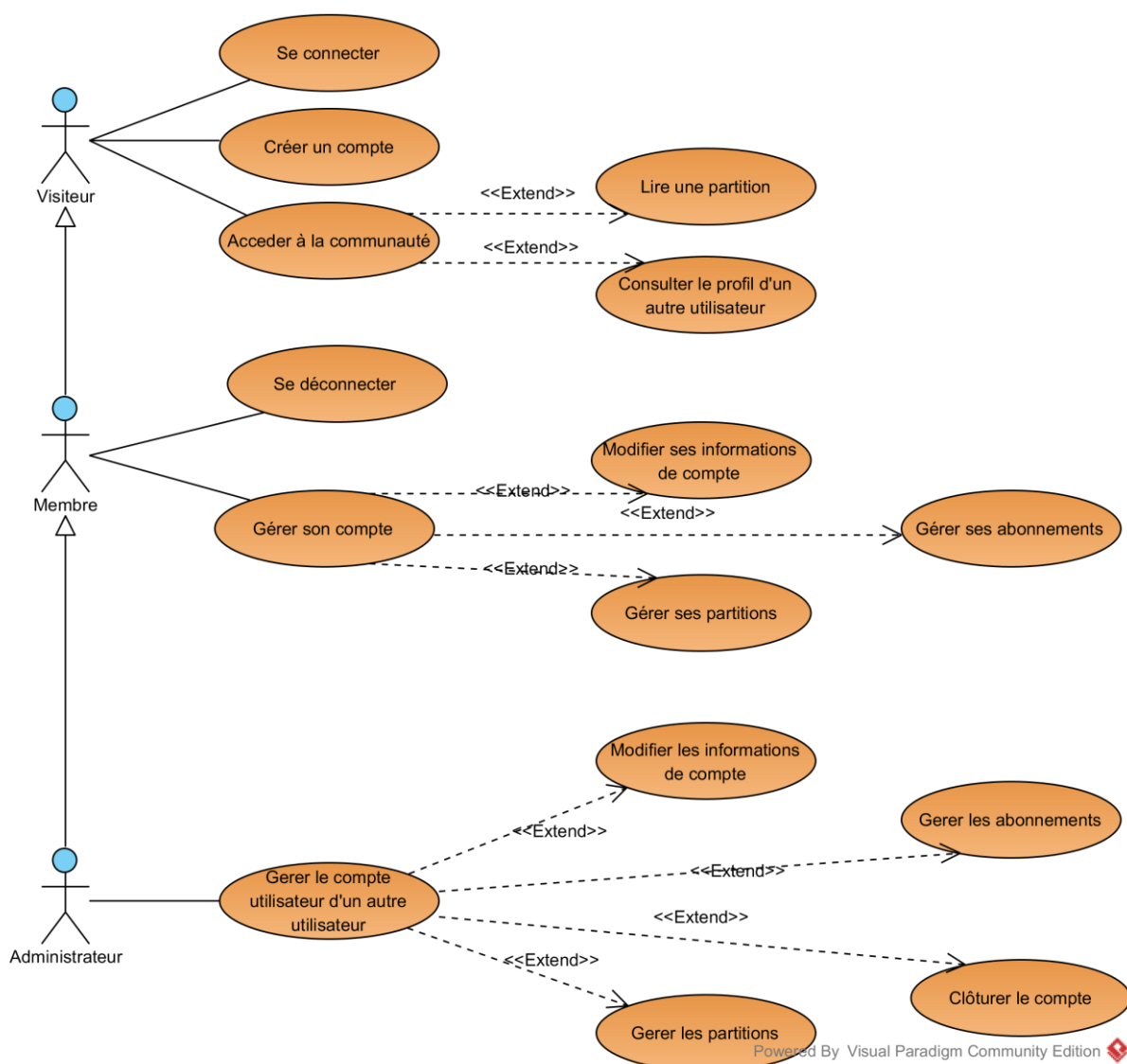




Figure 1 : Cas d'utilisation en fonction des utilisateurs de l'API

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Les besoins métiers</i>	

2.1.2. Permissions



Puisqu'il existe différents type d'utilisateurs, certains ayant accès à plus de fonctionnalités que d'autres d'utilisateur, l'API devra supporter plusieurs niveaux de permissions.

<i>Nom court</i>	<i>Nom</i>	<i>Description</i>
VIS	Droit Visiteur	L'utilisateur peut effectuer l'action, qu'il soit connecté ou pas.
MEM	Droit membre	L'utilisateur doit être connecté pour effectuer l'action. S'il est connecté à un compte membre, l'action ne peut être effectuée que sur son compte. L'identifiant utilisateur doit être vérifié. En revanche, si l'utilisateur est connecté sur un compte administrateur, l'action peut être effectuée sur n'importe quel compte.
ADM	Droit Administrateur	L'utilisateur doit être connecté à un compte administrateur pour effectuer l'action.



Tableau 3 : Description des différents niveaux de privilège

2.2. Vue en ressources

La liste ci-dessous définit les ressources disponibles sur le serveur, leur méthodes HTTP autorisées et le niveau de permission associé nécessaires à l'appel de la requête. Les permissions sont décrites dans la partie [2.1-Utilisateurs et permissions](#).



 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Les besoins métiers</i>	

Ressource	Méthode HTTP	Permission	Description
/login	POST	VIS	Connecte un utilisateur à partir de ses identifiants de connexion.
/logout	POST	VIS	Déconnecte un utilisateur.
/forgotten_password	POST	VIS	Envoie un mail à un utilisateur qui a perdu son mot de passe.
	PUT	VIS	Réinitialise le mot de passe d'un utilisateur en utilisant un jeton de réinitialisation.
/users	GET	VIS	Récupère la liste des utilisateurs. Des filtres peuvent être ajoutés.
	POST	VIS	Enregistre un nouvel utilisateur.
/users/[id]	GET	VIS	Récupère toutes les informations d'un utilisateur (informations personnelles, abonnement, abonnements et partitions).
	DELETE	ADM	Clôture un compte.
/users/[id]/password	PUT	MEM	Change le mot de passe d'un utilisateur en utilisant son mot de passe actuel.
/users/[id]/personal_data	GET	VIS	Récupère toutes les informations personnelles d'un utilisateur.
	PUT	MEM	Change les informations personnelles d'un utilisateur.
/users/[id]/photo	GET	VIS	Récupère la photo de profil d'un utilisateur.
	PUT	MEM	Change la photo de profil d'un utilisateur.
/users/[id]/subscriptions	GET	VIS	Récupère la liste des abonnements d'un utilisateur.
	POST	MEM	Ajoute un nouvel abonnement à un utilisateur.
/users/[id]/subscriptions/[uid]	DELETE	MEM	Retire un abonnement d'un utilisateur.
/users/[id]/subscribers	GET	VIS	Récupère la liste des abonnés d'un utilisateur.
/users/[id]/scores/own	GET	VIS	Récupère la liste des partitions d'un utilisateur.
	POST	MEM	Ajoute une nouvelle partition à un utilisateur.
/users/[id]/scores/own/[sid]	PUT	MEM	Modifie les informations d'une partition (titre).
	DELETE	MEM	Retire une partition d'un utilisateur.
/users/[id]/scores/favourites	GET	VIS	Récupère la liste des partitions favorites d'un utilisateur.
	POST	MEM	Ajoute une nouvelle partition favorite à un utilisateur.
/users/[id]/scores/favourites/[sid]	DELETE	MEM	Retire une partition favorite d'un utilisateur.
/users/[id]/purchases	GET	MEM	Récupère la liste des achats d'un utilisateur.
/scores	GET	VIS	Récupère la liste des partitions. Des filtres peuvent être ajoutés.
/scores/[id]	DELETE	ADM	Supprime le fichier contenant la partition.
/purchases	GET	ADM	Récupère la liste des achats. Des filtres peuvent être ajoutés.
/contact	POST	VIS	Envoie un email à l'équipe MSW.

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Les besoins métiers</i>	

<i>Ressource</i>	<i>Méthode HTTP</i>	<i>Permission</i>	<i>Description</i>
/test	ANY	VIS	Test la disponibilité de l'API.

Tableau 4 : Vue en ressources de l'API

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Implémentation</i>	

3. Implémentation

3.1. Les règles métiers du serveur

Ci-dessous se trouve la liste des règles métiers que le serveur doit implémenter.



<i>Nom</i>	<i>Règle</i>
Unicité des comptes	Le nom d'utilisateur et l'adresse email doivent être uniques à chaque compte. De plus, chaque compte est identifié par un « id » (numérique).
Mot de passe - format	Le mot de passe doit contenir plus de 6 caractères avec au moins 1 chiffre
Statuts de compte	Lorsqu'un compte est créé, son statut est à « PENDING ». Une fois validé, il passe en « ACTIVATED ». Si un compte a été clôturé, son statut est à « CLOSED ».

Tableau 5 : Règle métier de l'API



3.2. L'interface

Les données échangées dans les requêtes et les réponses de l'API sont au format JSON.



Ci-dessous se trouve le détail des requêtes prises en charges par le serveur.

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
Ressource	Méthode HTTP	Corps	Code HTTP	Corps
/login	POST	<pre>{ "login": "bob", "password": "azerty123" }</pre> <p>"login" can be either the username or the email</p>	200	<pre>{ "id": 1, "username": "bob", "last_activity_date": "10-29-2015" }</pre> <p>La date est sous la forme MM-DD-YYYY</p>
			401	<pre>{ "shortcode": "LOG-BADCREENTIALS", "message": "Credentials not valid" }</pre>
			403	<pre>{ "shortcode": "LOG-ACCNOTACTIVE", "message": "Account not activated" }</pre>
			403	<pre>{ "shortcode": "LOG-CLOSEDACC", "message": "Account closed" }</pre>
/logout	POST	EMPTY	204	EMPTY
/forgotten_password	POST	<pre>{ "email": "bob.marley@jamaica.jm" }</pre>	204	EMPTY
			404	<pre>{ "shortcode": "FOR-EMAILNOTFOUND", "message": "Email not found" }</pre>
	PUT	<pre>{ "token": "[token]", "password": "azerty123" }</pre>	204	EMPTY
			400	<pre>{ "shortcode": "GLO-BADFIELD", "message": "Field(s) not correct", "data": {</pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> "password": "password not valid", } </pre>
			404	<pre> { "shortcode": "FOR-TOKENNOTFOUND", "message": "Token not found" } </pre>
/users Des filtres peuvent être ajoutés en tant que paramètre GET - uname - fname - sname - email	GET	EMPTY	200	<pre> [{ "id": 1, "username": "bob", "firstname": "bob", "lastname": "marley", "email": "bob.marley@lamaica.jm", "message": "I have only one love", "status": "ACTIVATED", "last_activity_date": { "date": "2015-09-04 16:23:48", "timezone_type": 3, "timezone": "Europe/Paris" }, "creation_date": { "date": "2015-09-04 16:23:48", "timezone_type": 3, "timezone": "Europe/Paris" }, "photo": "[url_photo]", "nb_subscriptions": [nb_subscriptions], "nb_subscribers": [nb_subscribers], "nb_favourites": [nb_favourites], "nb_scores": [nb_scores], "is_subscription": false }] </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse
			<pre> }, { "id": 2, "username": "bobr", "firstname": "bobr", "lastname": "marleyr", "email": "bob.marleyr@lamaica.jm", "message": "I have only two love", "status": "PENDING", "last_activity_date": { "date": "2015-09-04 16:23:48", "timezone_type": 3, "timezone": "Europe/Paris" }, "creation_date": { "date": "2015-09-04 16:23:48", "timezone_type": 3, "timezone": "Europe/Paris" }, "photo": "[url_photo]", "nb_subscriptions": [nb_subscriptions], "nb_subscribers": [nb_subscribers], "nb_favourites": [nb_favourites], "nb_scores": [nb_scores], "is_subscription": true } ...] </pre> <p>Seuls les comptes non-clôturés (status != « CLOSED ») sont renvoyés. Le tableau d'utilisateurs contient les utilisateurs correspondants aux filtres de recherche.</p>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				Le champ « is_subscription » est à <i>true</i> si l'utilisateur retourné est un abonnement de l'utilisateur connecté. Sinon, il est à <i>false</i> . Si aucun utilisateur n'est connecté, le champ est à <i>false</i> .
	POST	<pre>{ "username": "bob", "firstname": "bob", "lastname": "marley", "email": "bob.marley@jamaica.jm", "password": "azerty123", }</pre> <p>Les champs "firstname" et "surname" sont facultatifs</p>	200	<pre>{ "id": 1, "username": "bob", "firstname": "bob", "lastname": "marley", "email": "bob.marley@jamaica.jm", "message": "I have only one love", "status": "PENDING", "last_activity_date": null, "creation_date": { "date": "2015-09-04 16:23:48", "timezone_type": 3, "timezone": "Europe/Paris" }, "photo": "[url_photo]", }</pre>
			400	<pre>{ "shortcode": "GLO-BADFIELD", "message": "Field(s) not correct", "data": { "username": "username not valid", "email": "email not valid" } }</pre>
			409	<pre>{ "shortcode": "REG-EALRUSED", "message": "Username or email already used", "data": {</pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> "username": "username already used.", "email": "email already used." } </pre>
/users/[id]	GET	EMPTY	200	<pre> { "is_subscription": true "personal_data": { "id": 1, "username": "bob", "firstname": "bob", "lastname": "marley", "email": "bob.marley@jamaica.jm", "message": "I have only one love", "photo": "[url_photo]" }, "subscription": [{ "id": 2, "username": "bobr", "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": true }, { "id": 3, "username": "bobs", "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": true }] } </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> } ...], "subscriber": [{ "id": 2, "username": "bobr", "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": true }, { "id": 4, "username": "bobt", "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": true }] ...], "score": { "own": [{ "id": 1, "title": "One love", "by": { "id": 1, "username": "bob" } }] "location_project": "[url_project]", </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Implémentation</i>	



Requête			Réponse	
				<pre> "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": true }, { "id": 2, "title": "Generation", "by": { "id": 1, "username": "bob" } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": false }, ...], "favorite": [{ "id": 1, "title": "One love", "by": { "id": 1, "username": "bob" } } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": true }, { </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> "id": 2, "title": "Macumba", "by": { "id": 6, "username": "Renand" } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": false }, ...] } </pre> <p>Le champ « is_subscription » est à <i>true</i> si l'utilisateur retourné est un abonnement de l'utilisateur connecté. Sinon, il est à <i>false</i>. Si aucun utilisateur n'est connecté, le champ est à <i>false</i>.</p> <p>Le champ « is_favourite » est à <i>true</i> si la partition retournée est une partition favorite de l'utilisateur connecté. Sinon, il est à <i>false</i>. Si aucun utilisateur n'est connecté, le champ est à <i>false</i>.</p>
	DELETE	EMPTY	200	<pre> { "id": 1, "closure_date": { "date": "2015-09-11 18:26:56", "timezone_type": 3, "timezone": "Europe/Paris" }, "status": "CLOSED" } </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
			403	{ "shortcode": "CLO-USERALRCLO", "message": "User account already closed" }
/users/[id]/password	PUT	{ "current_password": "azerty123", "new_password": "qwerty456" }	204	EMPTY
			400	{ "shortcode": "GLO-BADFIELD", "message": "Field(s) not correct", "data": { "password": "password not valid" } }
			403	{ "shortcode": "USR-WRONGPASS", "message": "Current password not correct" }
/users/[id]/personal_data	GET	EMPTY	200	{ "id": 1, "username": "bob", "firstname": "bob", "lastname": "marley", "email": "bob.marley@jamaica.jm", "message": "I have only one love", "photo": "[url_photo]", "nb_subscriptions": [nb_subscriptions], "nb_subscribers": [nb_subscribers], "nb_favourites": [nb_favourites], "nb_scores": [nb_scores] }
	PUT	{ "firstname": "bob",	200	{ "id": 1,

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
		<pre> "lastname": "marley", "email": "bob.marley@jamaica.jm", "message": "I have only one love" </pre>		<pre> "username": "bob", "firstname": "bob", "lastname": "marley", "email": "bob.marley@jamaica.jm", "message": "I have only one love", "photo": "[url_photo]", "nb_subscriptions": [nb_subscriptions], "nb_subscribers": [nb_subscribers], "nb_favourites": [nb_favourites], "nb_scores": [nb_scores] </pre>
			400	<pre> { "shortcode": "GLO-BADFIELD", "message": "Field(s) not correct", "data": { "email": "email not valid" } } </pre>
/users/[id]/photo	GET	EMPTY	200	<pre> { "photo": "[url_photo]" } </pre>
	PUT	En-tête : Content-type: [content_type_picture]	200	<pre> { "photo": "[url_photo]" } </pre>
		Corps : Le contenu du fichier	406	<pre> { "shortcode": "UPL-BADCONTENTTYPE", "message": "ContentType not acceptable" } </pre>
/users/[id]/subscriptions	GET	EMPTY	200	<pre> [{ "id": 1, "username": "bob", </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": true }, { "id": 2, "username": "bobr", "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": true } ...]</pre> <p>Le champ « is_subscription » est à <i>true</i> si l'utilisateur retourné est un abonnement de l'utilisateur connecté. Sinon, il est à <i>false</i>. Si aucun utilisateur n'est connecté, le champ est à <i>false</i>.</p>
	POST	<pre> { "id": 2 }</pre>	200	<pre> { "id": 2, "username": "bobr", "photo": "[url_photo]" }</pre>
			404	<pre> { "shortcode": "SUB-BADUSERID", "message": "User not found" }</pre>
/users/[id]/subscriptions/[uid]	DELETE	EMPTY	204	EMPTY
/users/[id]/subscribers	GET	EMPTY	200	<pre> [{ "id": 1,</pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> "username": "bob", "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": true }, { "id": 2, "username": "bobr", "photo": "[url_photo]", "nb_subscribers": [nb_subscribers], "nb_scores": [nb_scores], "is_subscription": false } ...] </pre> <p>Le champ « is_subscription » est à <i>true</i> si l'utilisateur retourné est un abonnement de l'utilisateur connecté. Sinon, il est à <i>false</i>. Si aucun utilisateur n'est connecté, le champ est à <i>false</i>.</p>
/users/[id]/scores/own	GET	EMPTY	200	<pre> [{ "id": 1, "title": "One love", "by": { "id": 1, "username": "bob" } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": true }] </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> }, { "id": 2, "title": "Generation", "by": { "id": 5, "username": "Simple surface" } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": false }, ...] </pre> <p>Le champ « is_favourite » est à <i>true</i> si la partition retournée est une partition favorite de l'utilisateur connecté. Sinon, il est à <i>false</i>. Si aucun utilisateur n'est connecté, le champ est à <i>false</i>.</p>
	POST	En-tête : Content-type: [content_type_msw] Corps : Le contenu du fichier	200	<pre> { "id": 1, "title": "Generation", "by": { "id": 5, "username": "Simple surface" } "location_project": "[url_project]", "location_preview": "[url_preview]", } </pre>
			406	<pre> { "shortcode": "UPL-BADCONTENTTYPE", "message": "ContentType not acceptable" } </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
/users/[id]/scores/own/[sid]	PUT	{ "title": "Generation" }	200	{ "id": 1, "title": "Generation", "by": { "id": 5, "username": "Simple surface" } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], }
	DELETE	EMPTY	204	EMPTY
/users/[id]/scores/favourites	GET	EMPTY	200	[{ "id": 1, "title": "One love", "by": { "id": 1, "username": "bob" } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": true }, { "id": 2, "title": "Generation", "by": { "id": 5, "username": "Simple surface" } }]

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	



Requête			Réponse	
				<pre> } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": false }, ...] </pre> <p>Le champ « is_favourite » est à <i>true</i> si la partition retournée est une partition favorite de l'utilisateur connecté. Sinon, il est à <i>false</i>. Si aucun utilisateur n'est connecté, le champ est à <i>false</i>.</p>
	POST	<pre> { "id": 2 } </pre>	200	<pre> { "id": 1, "title": "Generation", "by": { "id": 5, "username": "Simple surface" }, "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], } </pre>
			404	<pre> { "shortcode": "FAV-BADSCOREID", "message": "Score not found" } </pre>
			409	<pre> { "shortcode": "FAV-SCOREALREADYFAV", "message": "Score already in favourites" } </pre>
/users/[id]/scores/favourites/[fid]	DELETE	EMPTY	204	EMPTY

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	

Requête			Réponse	
/users/[id]/purchases	GET	EMPTY	200	[<pre> { "id": 1, "date": { "date": "10-29-2015 00:00:00", "timezone_type": 3, "timezone": "Europe/Paris" } }, { "id": 1223, "date": { "date": "10-30-2016 00:00:00", "timezone_type": 3, "timezone": "Europe/Paris" } }, ...] </pre>
/scores Des filtres peuvent être ajoutés en tant que paramètre GET - title - by	GET	EMPTY	200	[<pre> { "id": 1, "title": "One love", "by": { "id": 1, "username": "bob" }, "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": true }, </pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	

Requête			Réponse	
				<pre>{ "id": 2, "title": "Generation", "by": { "id": 5, "username": "Simple surface" } "location_project": "[url_project]", "location_preview": "[url_preview]", "nb_favourites": [nb_favourites], "is_favourite": false }, ...</pre> <p>Le tableau de partition contient les partitions correspondantes aux filtres de recherche.</p> <p>Le champ « is_favourite » est à <i>true</i> si la partition retournée est une partition favorite de l'utilisateur connecté. Sinon, il est à <i>false</i>. Si aucun utilisateur n'est connecté, le champ est à <i>false</i>.</p>
/scores/[id]	DELETE	EMPTY	204	EMPTY
/purchases	GET	EMPTY	200	<pre>[{ "id": 1, "by": { "id": 1, "username": "bob" }, "date": { "date": "10-29-2015 00:00:00", "timezone_type": 3, "timezone": "Europe/Paris" } }]</pre>

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	

Requête			Réponse	
				<pre> } }, { "id": 1223, "by": { "id": 1, "username": "bob" }, "date": { "date": "10-30-2016 00:00:00", "timezone_type": 3, "timezone": "Europe/Paris" } }, ...] </pre>
/contact	POST	<pre> { "email": "bob.marley@jamaica.jm", "subject": "Connexion Impossible", "message": "je n'arrive pas à me connecter" } </pre>	201	EMPTY
/test	ANY	Whatever	200	<pre> { "status": "1" "method": "ANY" "body": "Whatever" } </pre> <p>Le champ "status" est à un 1 si l'API est fonctionnelle. Sinon, 0. Le champ "method" est égal à la méthode de la requête. Le champ "body" est égal aux données envoyées dans le corps de la requête, sous forme de chaîne de caractères. Chaîne vide si le corps de la requête est vide.</p>



 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	

Tableau 6 : Détails des entrées-sorties des routes de l'API

3.3. Le téléchargement de fichier

Lors d'un téléchargement de fichier, il est demandé de spécifier un « Content-type ». Ci-dessous se trouve la liste des types de contenu acceptés.

	Valeur du « Content-type »	Description	Extension
<i>content_type_msw</i>	application/msw	MusicSheetWriter	.msw
<i>content_type_picture</i>	image/jpeg	Image	.jpg .jpeg
	image/png	Image	.png
	image/x-icon	Image	.ico
	image/gif	Image	.gif

Tableau 7 : Les différents "Content-Type" acceptés par l'API

3.4. Les erreurs

3.4.1. Le formatage

En cas d'erreur, la réponse de l'API est de la forme



```
{
    « shortcode » : [valeur],
    « message » : [valeur],
    « data » : [valeur/objet JSON]
}
```

Figure 2 : Schémas des erreurs renvoyées par l'API

Le champ « shortcode » est l'identifiant de l'erreur. Il permet d'identifier précisément l'erreur survenue.



Le champ « message » est l'explication de l'erreur en anglais.

Le champ « data » est optionnel. Il n'est présent que lorsque l'erreur a besoin d'être plus spécifique que le message et/ou lorsqu'elle doit renvoyer des données.

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	Implémentation	

3.4.2. Liste des erreurs

Short code	Code HTTP	Message	Explication (optionnel)
GLO-MISSING	400	<i>The following required parameter are missing: [param1] [param2]...</i>	Il manque des paramètres dans le corps de la requête. Seuls les paramètres manquants et obligatoires peuvent être retournés avec cette erreur
GLO-BADFIELD	400	<i>Field(s) not correct</i>	Le format de certains paramètres dans le corps de la requête est incorrect.
GLO-UNAUTHORIZED	401	<i>You are not allowed to perform this action</i>	L'utilisateur n'a pas les permissions requises pour effectuer cette action.
GLO-RESNFOUND	404	<i>The requested resource is not found</i>	L'URI de la requête est invalide.
GLO-USERNFOUND	404	<i>The user ID is not found</i>	L'ID d'un utilisateur inséré dans la requête /users/[id] (et dérivée) est introuvable
GLO-SCORENOTFOUND	404	<i>The score ID is not found</i>	L'ID d'une partition inséré dans la requête /users/[id]/scores/own/[id] ou /scores/[id] est introuvable.
GLO-SUBNOTFOUND	404	<i>The subscription ID is not found</i>	L'ID d'un abonnement inséré dans la requête /users/[id]/subscription/[id] est introuvable.
GLO-FAVNOTFOUND	404	<i>The favourite score ID is not found</i>	L'ID d'une partition favorite inséré dans la requête /users/[id]/scores/favourites/[id] est introuvable.
GLO-METHNALLOW	405	<i>The method used is not allowed on the resource</i>	L'URI est correct mais la méthode HTTP associée n'est pas autorisée.
GLO-BADJSON	415	<i>The JSON sent is not correct</i>	Le corps de la requête n'est pas sous format JSON
GLO-DBSERERR	502	<i>The database server does not respond</i>	La base de données n'est accessible
GLO-DBSCHERR	500	<i>An SQL error occurred</i>	Une anomalie dans le schéma de la base de données est survenue. Elle ne devrait pas être retournée en production
LOG-BADCREDENTIALS	401	<i>Credentials not valid</i>	Les identifiants de connexion sont incorrectes
LOG-ACCNOTACTIVE	403	<i>Account not activated</i>	Le compte n'a pas été validé via le mail de confirmation d'inscription
LOG-CLOSEDACC	403	<i>Account closed</i>	Le compte a été clôturé
FOR-EMAILNOTFOUND	404	<i>Email not found</i>	L'adresse email n'appartient à aucun compte
FOR-TOKENNOTFOUND	404	<i>Token not found</i>	Le jeton de réinitialisation de mot de passe est introuvable

 Music Sheet Writer	Date de publication	25/10/2015	 Epitech Innovative Project
	Nom du projet	Music Sheet Writer	
	Objet du document	Spécification de l'API	
	Nom du chapitre	<i>Implémentation</i>	

Short code	Code HTTP	Message	Explication (optionnel)
REG-EALRUSED	409	<i>Username or email already used</i>	Le nom d'utilisateur et/ou l'adresse email appartiennent déjà à un autre utilisateur
USR-WRONGPASS	403	<i>Current password not correct</i>	Le mot de passe actuel du compte n'est pas correct
CLO-USERALRCLO	403	<i>User account already closed</i>	Le compte est déjà clôturé
UPL-BADCONTENTTYPE	406	<i>ContentType not acceptable</i>	Le champ « Content-type » dans l'en-tête de la requête n'est pas accepté par le serveur pour cette requête.
SUB-BADUSERID	404	<i>User not found</i>	L'utilisateur identifié n'existe pas.
FAV-BADSCOREID	404	<i>Score not found</i>	La partition identifiée n'existe pas.
FAV-SCOREALREADYFAV	409	<i>Score already in favourites</i>	La partition est déjà marquée comme favoris.

Tableau 8 : Liste des erreurs possibles renvoyées par l'API