Une image contenant oiseau

Description générée automatiquement

GraVTunes

Rapport final

Projet tutoré 2019-2020

TARDIEU Loïc, SAYER Jules, PERNOT Anthony, CHEVALIER Nathan

Une image contenant dessin, alimentation

Description générée automatiquementS. Cruz-Lara, I. Debled-Rennesson

Table des matières

[Introduction 3](#_Toc36299877)

[Objectif du document 3](#_Toc36299878)

[Présentation du projet 3](#_Toc36299879)

[Présentation de l’équipe 3](#_Toc36299880)

[Répartition des tâches 4](#_Toc36299881)

[Planning du déroulement du projet 4](#_Toc36299882)

[Quelques statistiques 5](#_Toc36299883)

[Analyse 6](#_Toc36299884)

[Découpage fonctionnel du projet 6](#_Toc36299885)

[Modèles UML utilisés 6](#_Toc36299886)

[Evolutions par rapport à l’étude préalable 8](#_Toc36299887)

[Réalisation 9](#_Toc36299888)

[Architecture générale 9](#_Toc36299889)

[Technologies utilisées 11](#_Toc36299890)

[Fonctionnalités réalisées 12](#_Toc36299891)

[Difficultés rencontrées 20](#_Toc36299892)

Conclusion 14

# Introduction

## Objectif du document

Ce document a pour but de détailler le suivi du projet GraVTunes, de la répartition des tâches à la réalisation finale tout en étudiant les différences et les évolutions par rapport à l’étude préalable. Il sera détaillé en 4 grand points :

* Introduction
* Analyse
* Réalisation
* Conclusion du projet

## Présentation du projet

GraVTunes est un service permettant de créer des vinyles à partir de fichiers audios importés par l’utilisateur. Il s’inspire des plateformes de streaming musical puisque l’utilisateur peut gérer lui-même sa bibliothèque musicale et de sites web commerciaux permettant d’acheter un vinyle personnalisé contenant les pistes audios de son choix.

La plateforme permet à ses utilisateurs d’importer des fichiers audios, de les gérer et de les transformer en « playlist » ainsi que de créer un vinyle personnalisé qu’ils pourront commander. L’objectif étant qu’à la fin du projet nous ayons un site totalement fonctionnel, allant de l’inscription de l’utilisateur jusqu’au paiement de la commande, tout en réalisant une partie « administration ».

## Présentation de l’équipe

Notre équipe est constitué de :

* Jules SAYER,
* Anthony PERNOT,
* Loïc TARDIEU,
* Nathan CHEVALIER

## Répartition des tâches

Jules: Espace membres, Panier, Commandes, Paiement, Vinyles, Administration

Anthony: Titres, tableau de bord, Collaboration

Loïc: Playlists

Nathan: Tickets

## Planning du déroulement du projet

Initialement prévu :

Itération 1 :

* Les maquettes du site (Nathan, Loïc)
* Une base de données construite (Anthony)
* Structure du projet en PHP (Framework, MVC, ORM, Moteur de Template) (Jules)
* Page d’accueil (Jules)
* Premiers essais web (Nathan, Loïc, Anthony, Jules)

Itération 2 :

* Réalisations des fonctionnalités principales du site :
  + Création de vinyle
  + Importation de fichiers et gestion de bibliothèque
  + Création de playlists
  + Support client
  + Administration
  + Commandes/Panier
  + Tableau de bord
  + Lecteur audio

Itération 3 :

* Intégration en version mobile

Réalisé :

* + Itération 1 : réalisée comme initialement prévu.
  + Itération 2 : réalisation partielle, les tickets, les playlists, le lecteur et l’administration n’étaient pas finis.
  + Itération 3 : réalisation des parties manquantes, abandon de la version mobile mais nous nous sommes adaptés en rendant un site le plus responsive possible.

## Quelques statistiques

* Statistiques globales du projet :

Une image contenant ordinateur, noir, assis, jeu

Description générée automatiquement

* Statistiques individuelles de chaque membre du projet :

Une image contenant capture d’écran

Description générée automatiquement

# Analyse

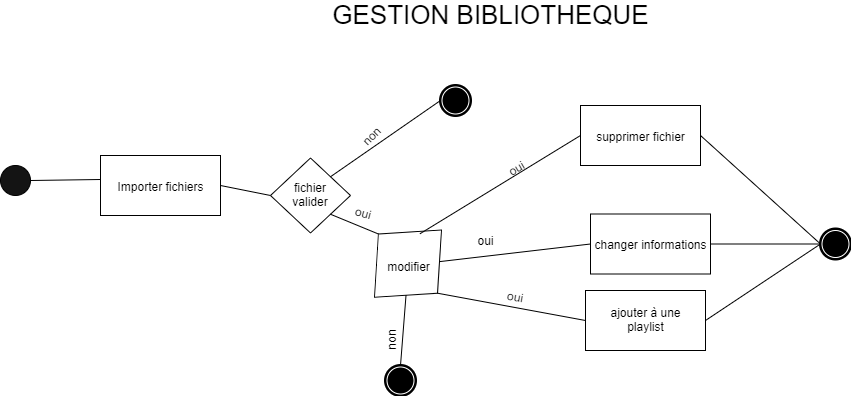
## Découpage fonctionnel du projet

Le projet se doit d’être découpé en plusieurs parties pour respecter le patron de conception MVC. Ce découpage permet d’avoir une cohérence dans la gestion des fonctionnalités de l’application. En effet, les données de la base de données (les tables User, Tracks, Vinyles, etc…) se retrouveront parmi les modèles alors que la gestion de leurs utilisations se trouvera à travers les contrôleurs (manipulation, contrôle des vues, etc…) tandis que l’interface du site et ses différentes pages se situeront dans les Vues.

## Modèles UML et diagrammes utilisés

Pour présenter certaines fonctionnalités, nous allons utiliser les modèles UML et diagrammes qui ont été réalisés lors de l’étude préalable pour montrer la cohérence du projet avec ce que nous avions prévues pour les fonctionnalités.

Pour la gestion des titres musicaux, nous pouvons illustrer cette fonctionnalité grâce à ce diagramme d’activité :



Une image contenant texte, carte

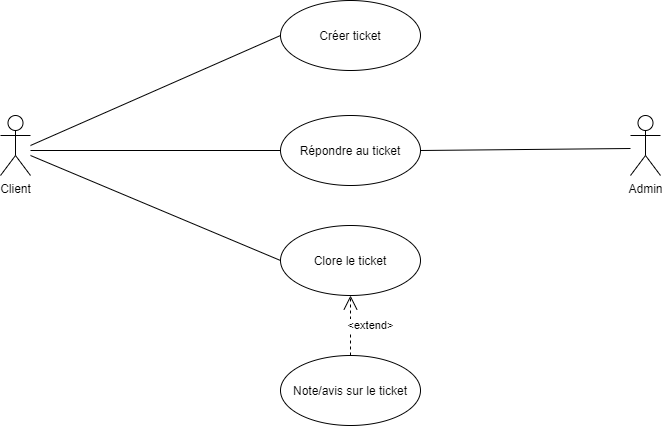
Description générée automatiquement

Nous pouvons désormais présenter la commande de vinyle avec ce cas d’utilisation :

Une image contenant périphérique

Description générée automatiquement

Si l’on s’intéresse aux fonctionnalités avec le support pour ce qui est des interactions avec les utilisateurs et les administrateurs du site, nous pouvons présenter celles-ci avec le cas d’utilisation suivant :



## Evolutions par rapport à l’étude préalable

Dans notre étude préalable, nous avions prévu de réaliser un site web proposant la gestion de bibliothèque personnelle de fichier audio avec également la possibilité de créer des playlists avec les titres enregistrés, et de pouvoir commander la gravure d’un vinyle à partir des fichiers de la bibliothèque. Le paiement de cette commande ne serait que simulé avec PayPal. Le vinyle pourrait avoir une pochette personnalisée.

Nous avions aussi prévu de faire en sorte que les utilisateurs puissent « s’entraider » lors de la création d’un vinyle : un utilisateur peut en inviter un autre pour collaborer avec lui, ce grâce à la clé du vinyle. Le collaborateur peut donc y ajouter ses propres musiques sur le vinyle de celui qui l’a invité. Il y aurait donc un espace client ou celui-ci peut gérer son compte et ses informations. La création du compte est obligatoire.

Pour que l’utilisateur puisse avoir un suivi de ses activités sur le site, nous avions prévu de faire une partie administration, où l’administrateur pourrait avoir un tableau de bord avec les informations globales du site. Un utilisateur pourrait s’adresser à un administrateur en cas de soucis, qui pourrait lui répondre.

Enfin, nous avions pensé faire de ce site web une application mobile disponible sur Android et IOS.

A l’heure actuelle, le projet est terminé, et l’étude préalable a été respectée, à l’exception de la transformation du site en application mobile. Tout le reste a pu être réalisé.

# Réalisation

## Architecture générale

Pour présenter l’architecture générale de GraVTunes, il est préférable d’utiliser l’arborescence que PhpStorm offre puisque cet outil a été l’IDE principal pour la conception et la réalisation du projet.

Une image contenant capture d’écran, ordinateur

Description générée automatiquement

Nous avons décidé d’utiliser le patron de conception MVC afin de séparer la gestion des fonctionnalités avec les données de la base de données et de l’interface web. Dans le dossier « src », nous avions créé un dossier « web » qui correspond au site internet ( il devait y avoir un dossier nommé « mobile » pour l’application mobile du projet ).

Dans le dossier « app », on retrouve les dossiers suivants :

* controllers : les fonctionnalités du site sont gérées ici ;
* exceptions : les exceptions sont créées dans ce dossier pour permettre de gérer les erreurs de l’application (critères de validation et authentification) ;
* extensions : on retrouve ici les extensions du moteur de template Twig : sécurité des formulaires (CSRF), messages de succès/erreur après soumission d’un formulaire ;
* helpers, middlewares ce dossier contient les classes PHP pour gérer les connexions au site et les sessions/cookies ;
* models : ce dossier correspond aux relations des tables de la base de données.
* views : les différentes pages du site internet sont stockées dans ce dossier.

Dans le répertoire « public\assets », nous pouvons retrouver les feuilles de style CSS, les scripts JavaScript, les fichiers multimédias et les polices de texte.

Pour observer les fichiers importés par les utilisateurs, ils se situent dans le dossier « uploads » :

* avatars : les images de profil des utilisateurs
* tracks : les fichiers audio des utilisateurs
* covers : les images/photos de couverture des vinyles

Le script SQL pour créer la base de données se nomme « gravtunes.sql » que l’on retrouve dans le dossier « sql ». Le dossier « vendor » sert à stocké les Frameworks PHP lors de l’installation/mise à jour du fichier composer.json à la racine du projet. Le fichier « .env » permet de stocker des données sensibles comme les identifiants de la base de données, les identifiants du Framework Braintree pour utiliser la fonctionnalité de paiement de Paypal, ainsi que les identifiants du serveur SMTP pour l’envoi de mail. Enfin, « index.php » permet de gérer les routes du site et « .htaccess » sert à réécrire et rediriger les URLs.

## Technologies utilisées

Puisque GraVTunes est un projet orienté Web dynamique avec la gestion d’une base de données, il nous semblait logique d’utiliser la langage PHP pour le côté applicatif du site avec l’utilisation des langages HTML/CSS pour la structure et la charte graphique du site. Également, JavaScript nous a été utile pour ajouter certaines fonctionnalités concernées avec le Document Object Model comme l’ajout d’un lecteur audio, la vérification du formulaire d’inscription du site et d’autres interactions avec l’utilisateur.

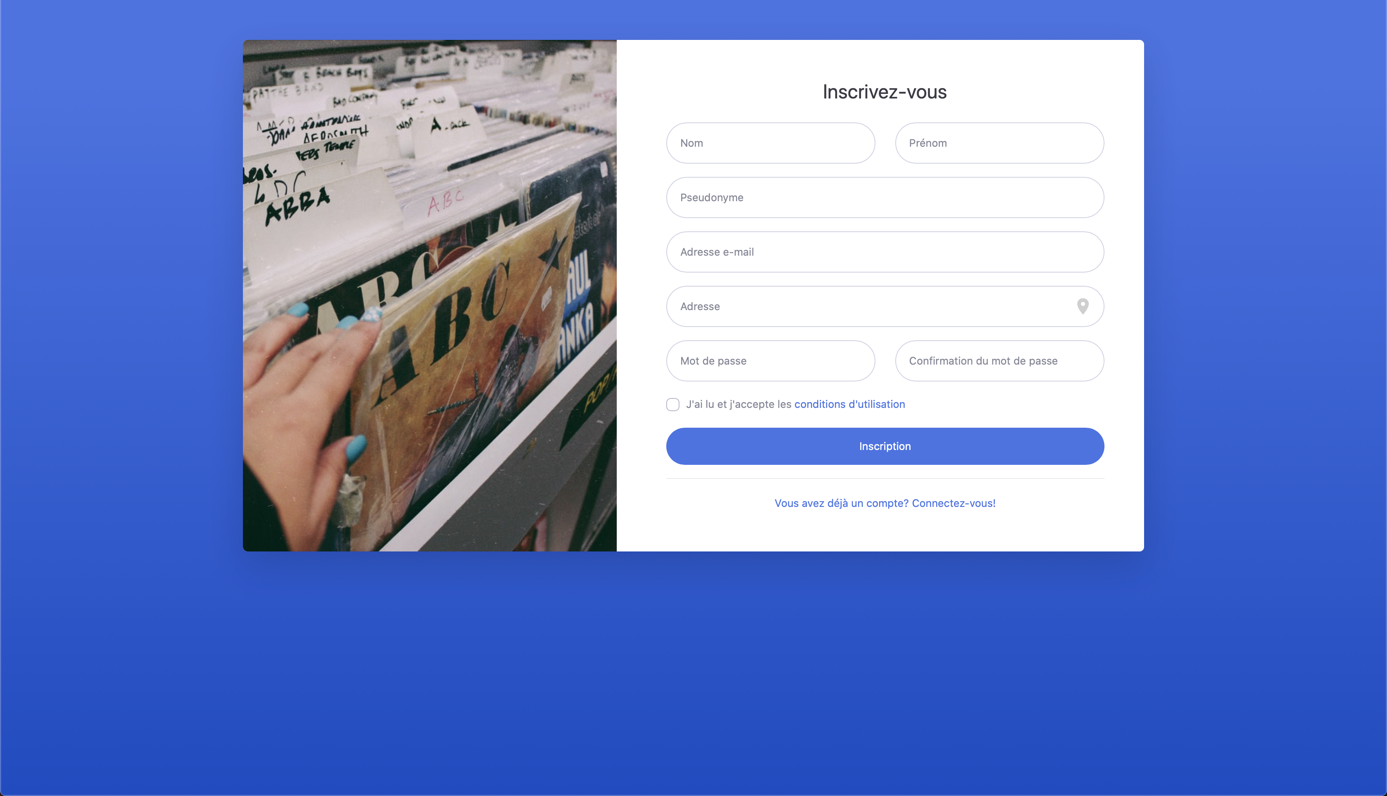
À partir de ces langages, nous avions ajoutés des Frameworks et des APIs pour simplifier et sécuriser certaines fonctionnalités à réaliser. Parmi les technologies qu’on a apporté, nous pouvons retrouver ceux-ci qui sont les plus exploitées :

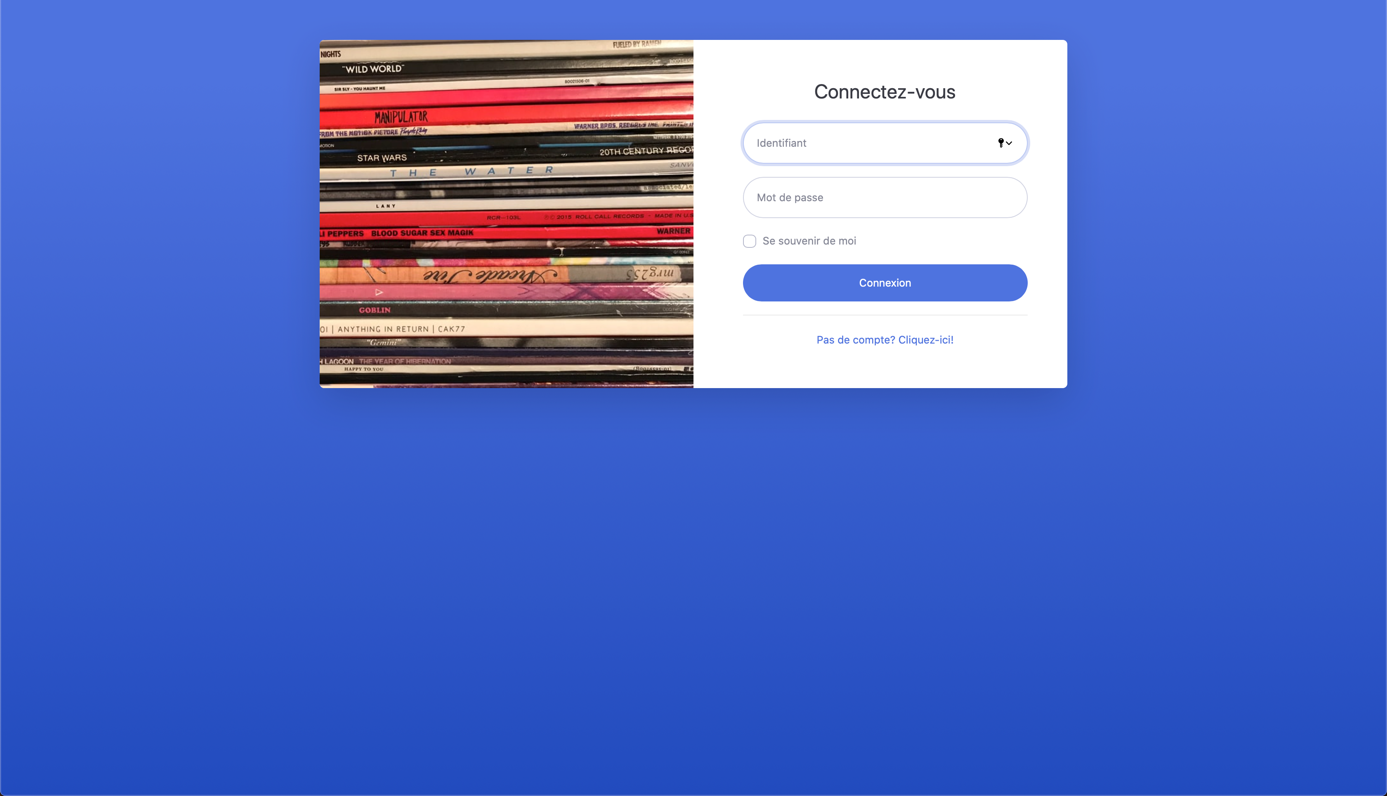
* Slim 4 : Micro-framework pour la structure de l’application web ;
* Twig : Moteur de template pour gérer les vues du projet ;
* Bootstrap : Framework CSS ;
* ORM Eloquent : Gestionnaire de relation entre la base de données et le site ;
* PHPDotenv : Framework PHP de gestion des variables d’environnements de l’application ;
* Braintree : Framework PHP appartenant à Paypal pour la gestion des paiements sécurisés ;
* Sticky Audio Player : API JavaScript pour implanter un lecteur audio sur le site ;
* FFMPEG : Framework PHP pour la manipulation de fichier audio/vidéo ;
* PHPMailer : Framework PHP pour l’envoi de mail ;

## Fonctionnalités réalisées

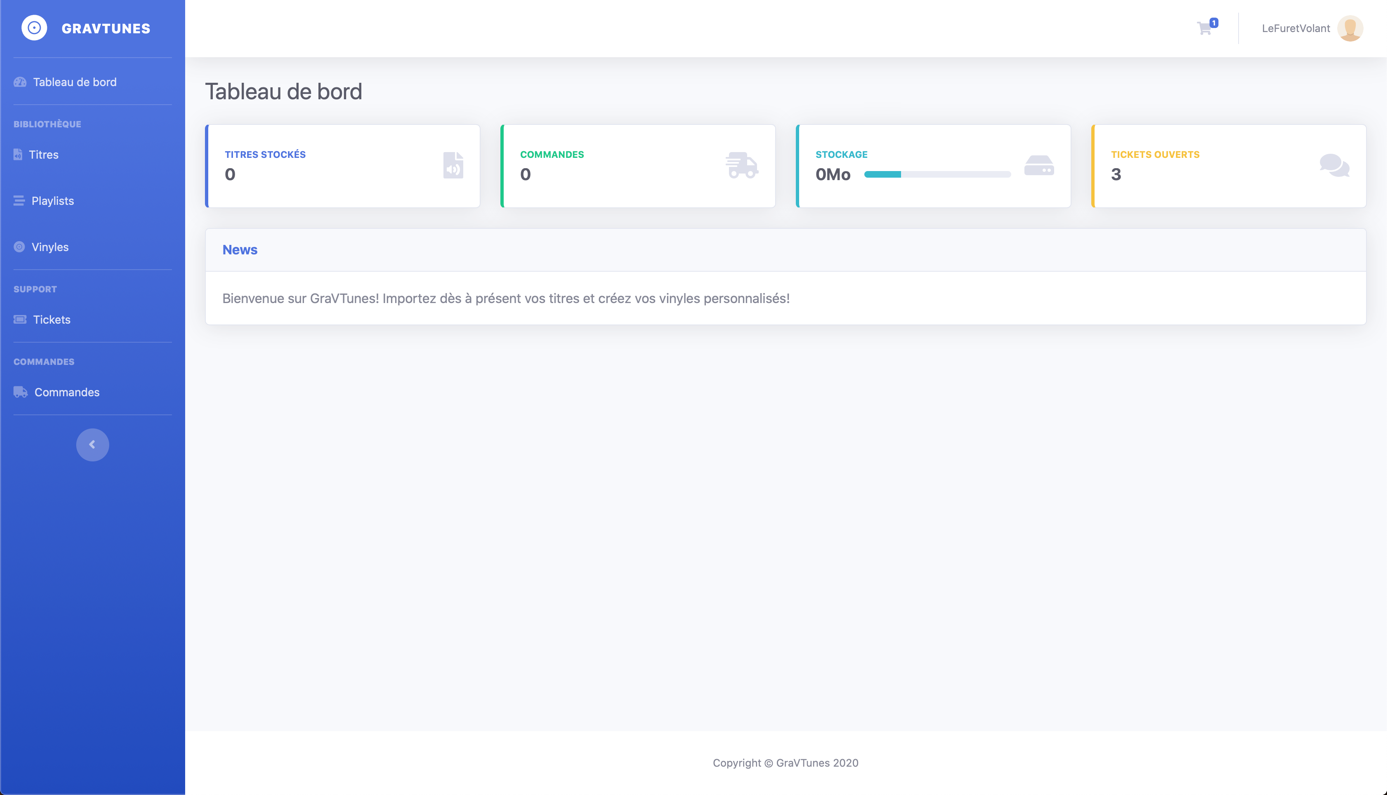
Pour la réalisation de ce projet, nous l’avions découpé en plusieurs parties de fonctionnalités. Les fonctionnalités réalisées ont été :

* Création de la base de données, pour assurer la persistance des données (comptes, musiques, playlists, vinyles, commandes…).
* Compte : inscription (avec critères de validation dans le formulaire) ; connexion ; modification du compte ; déconnexion.

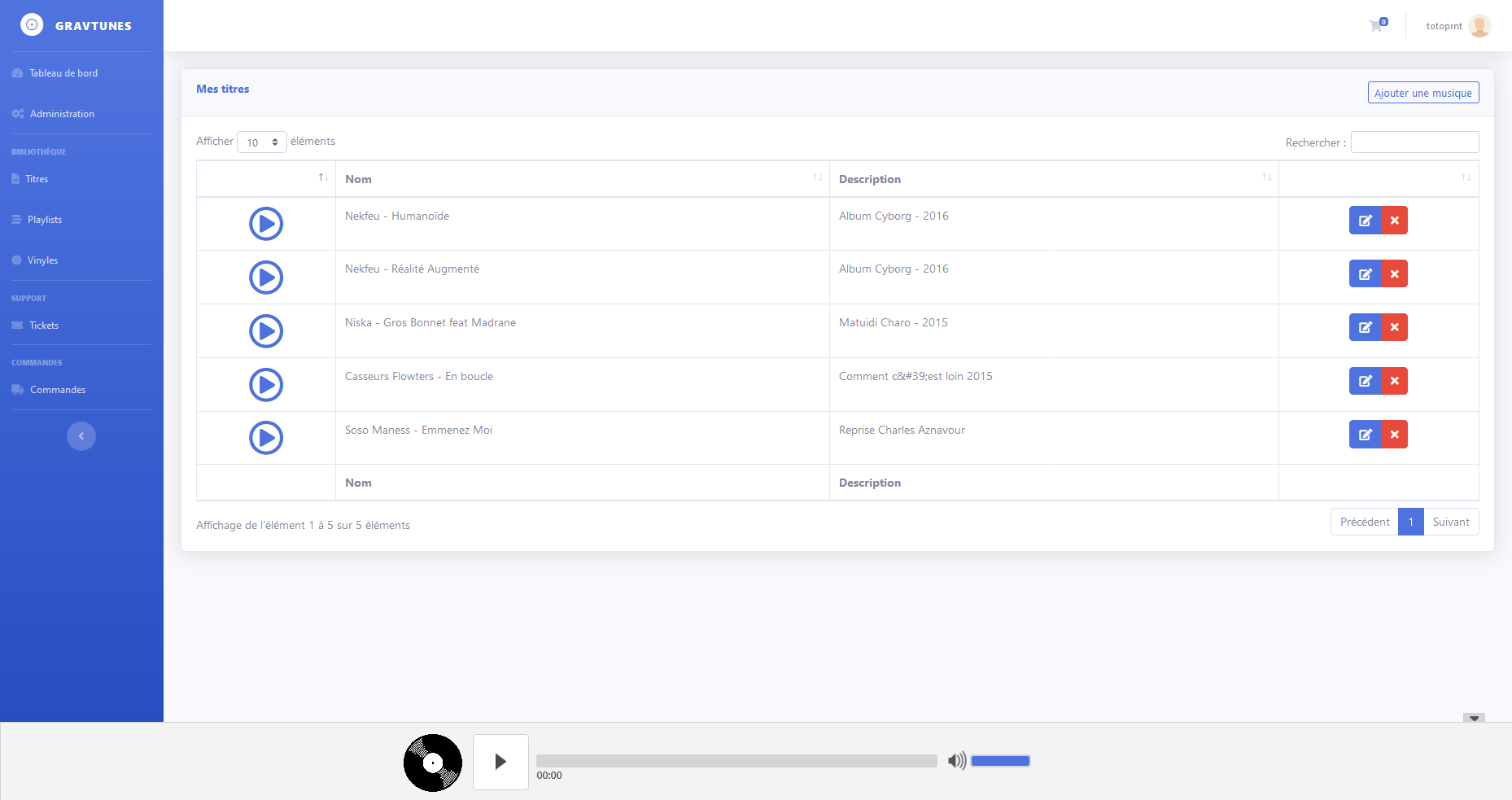




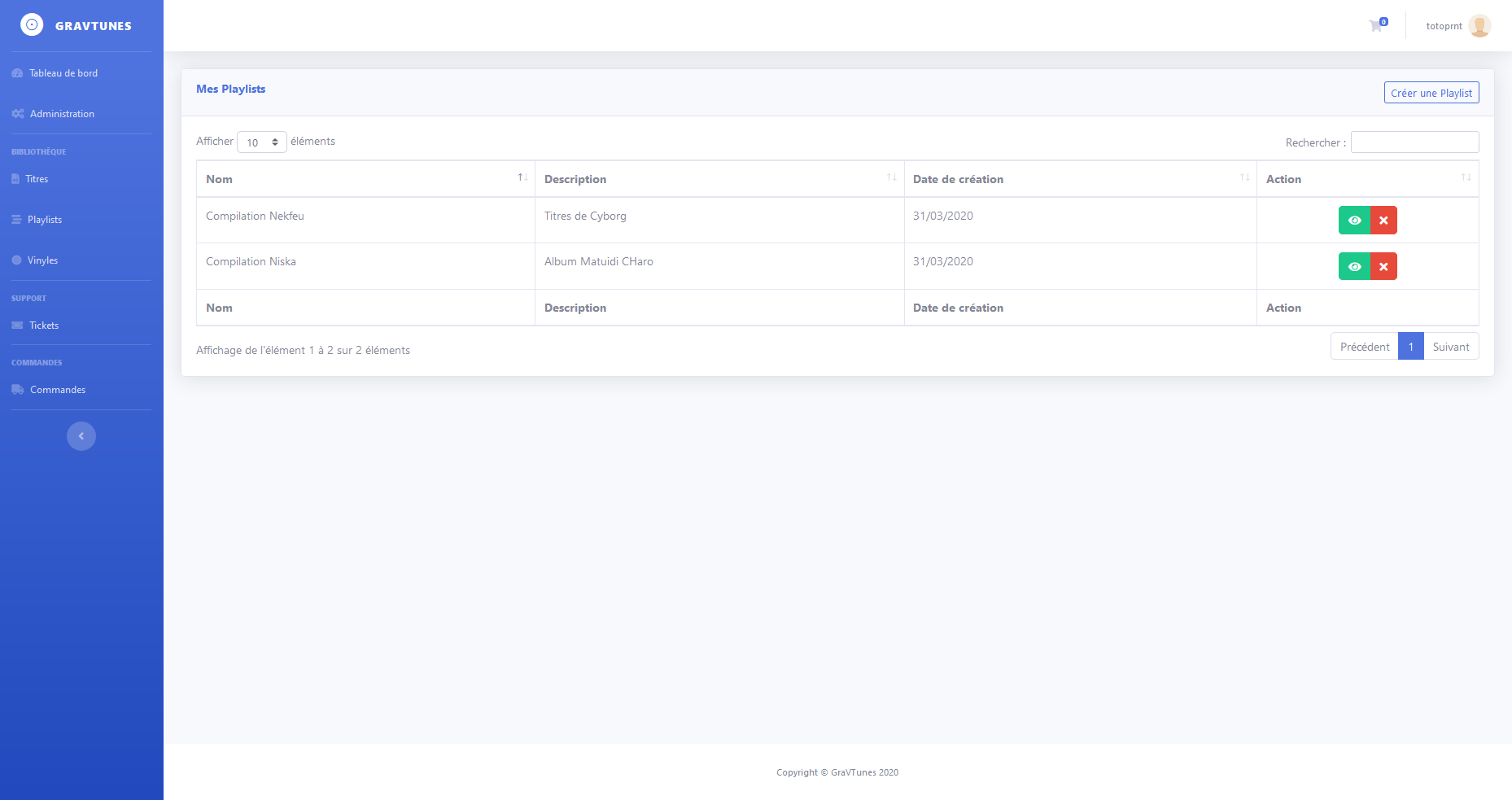
* Tableau de bord : affichage des statistiques de l’utilisateur connecté (nombre de titres stockés, nombre de tickets, nombres de commandes, espace de stockage occupé).

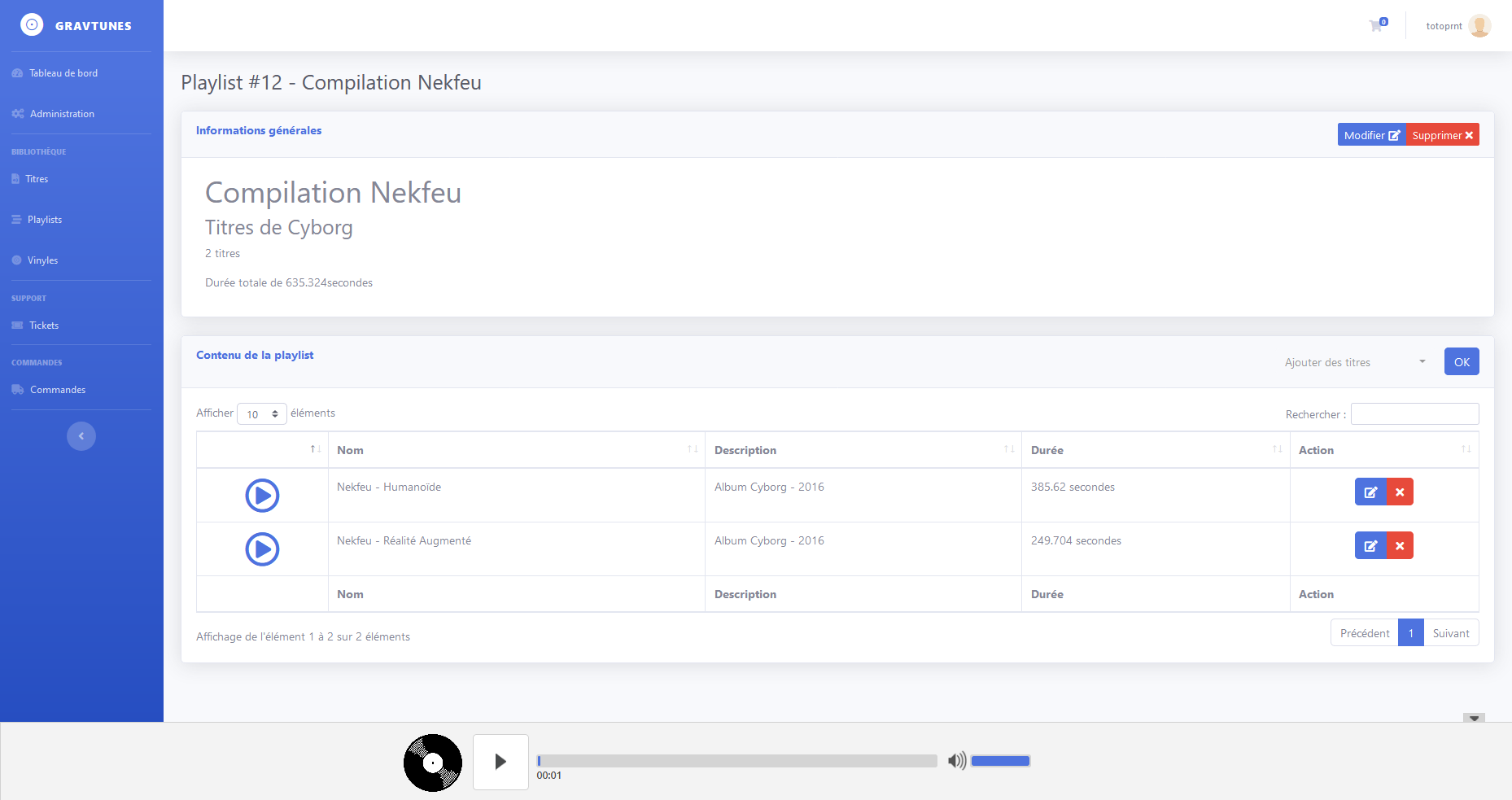


* Titre : affichage de ses titres musicaux ; création d’un titre (par importation de fichier) ; modification d’un titre ; suppression d’un titre ; possibilité d’écouter ses titres à l’aide d’un lecteur audio.

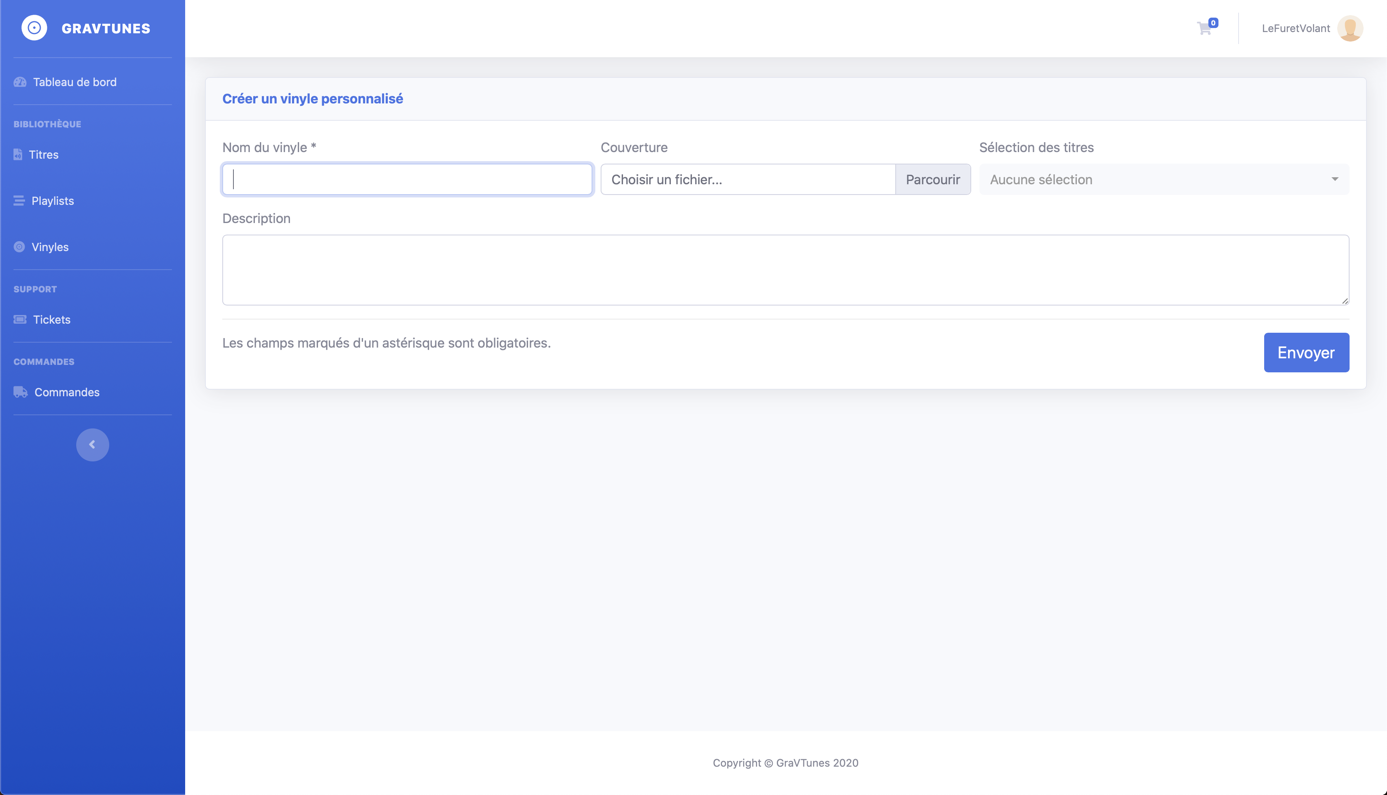


* Playlist : affichage des playlists ; création de playlist ; modification d’une playlist ; suppression de playlist ; ajout de titre dans une playlist ; suppression de titre dans une playlist.

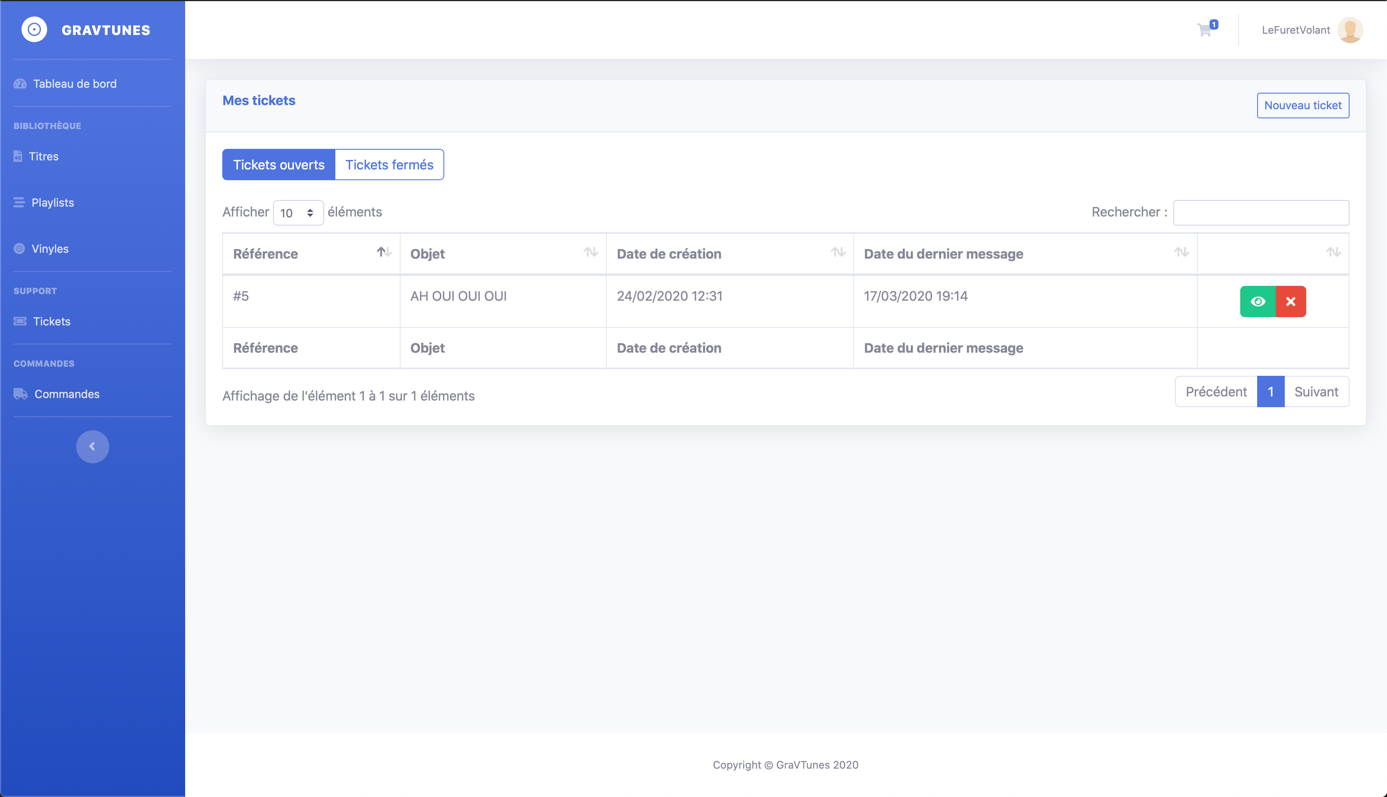


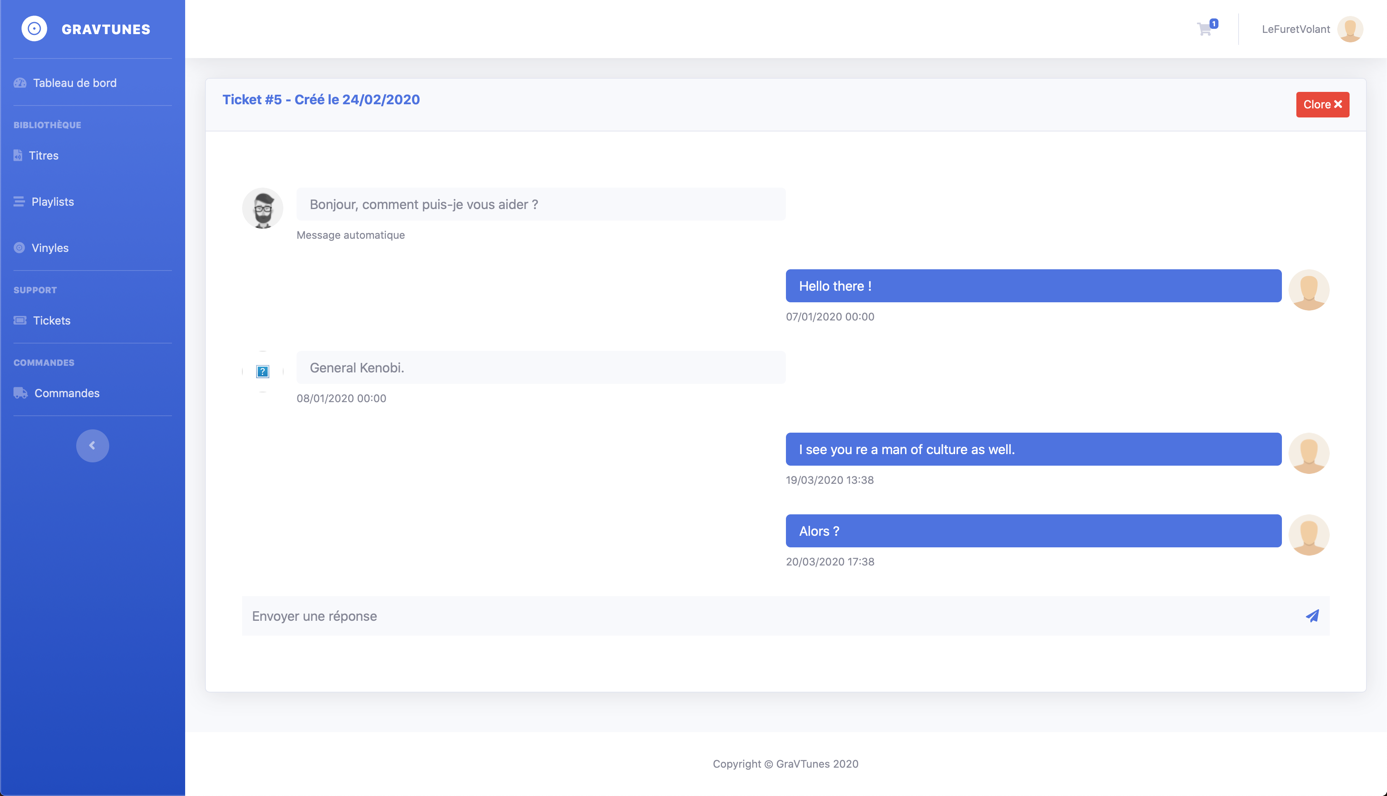


* Vinyle : affichage de ses vinyles ; création d’un vinyle ; modification d’un vinyle ; suppression d’un vinyle ; ajout de titre(s) à un vinyle ; ajout d’un vinyle au panier ; ajout d’un collaborateur à un vinyle ; ajout d’une musique de la part d’un collaborateur sur un vinyle.

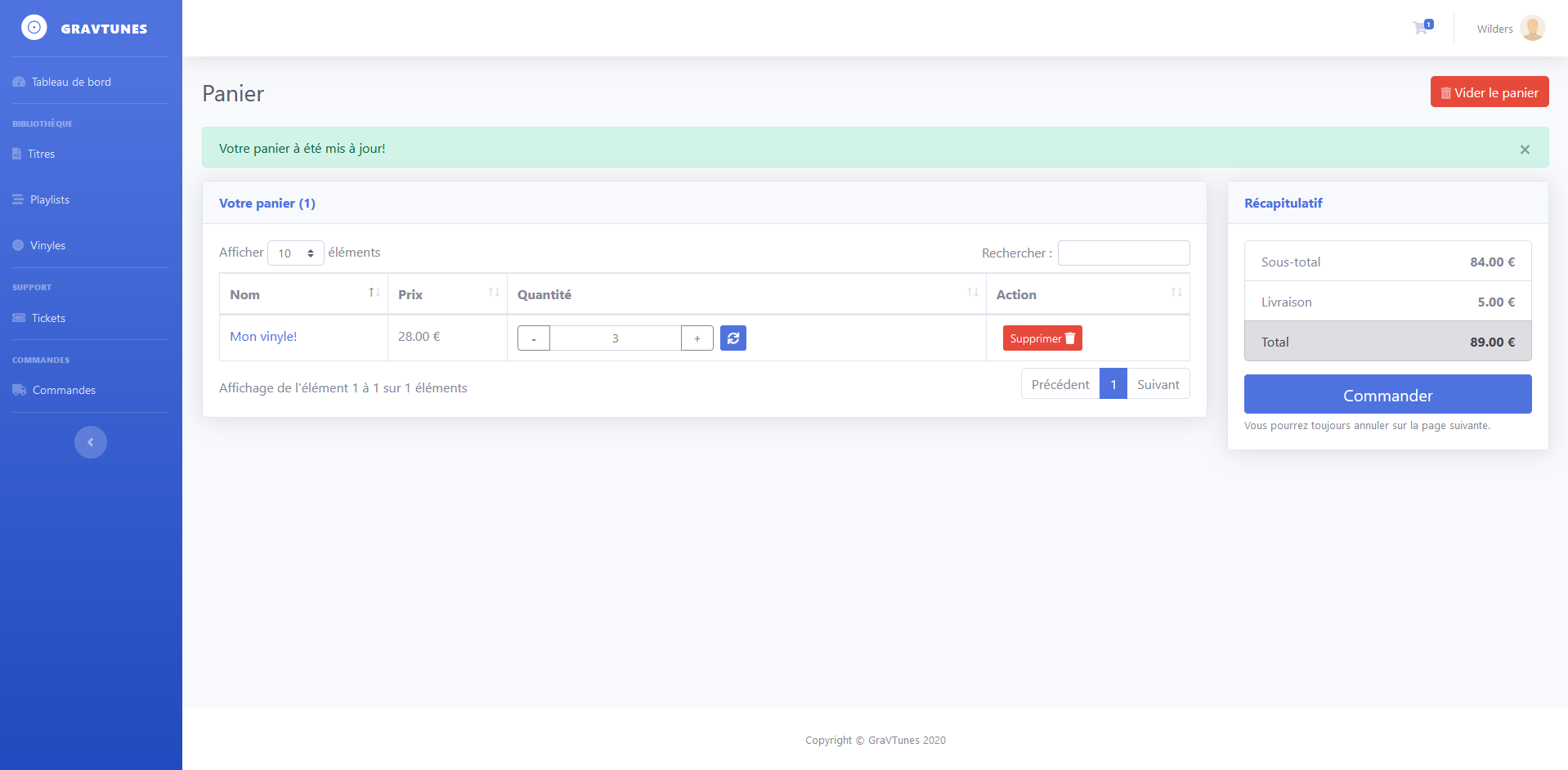


* Ticket : affichage de ses tickets ; ouverture/fermeture d’un ticket ; dialoguer à un administrateur à travers un ticket (interface de messagerie) .

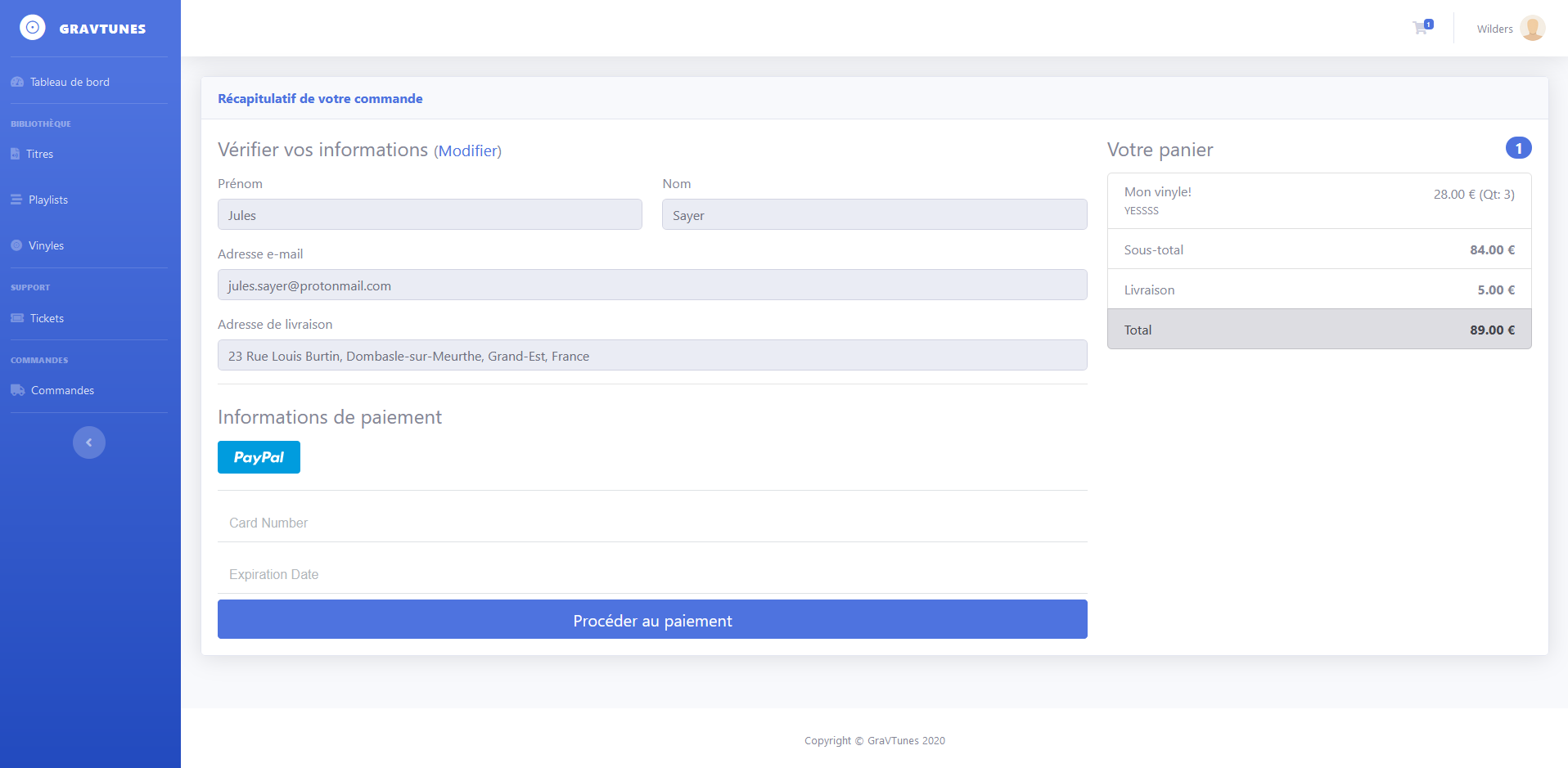


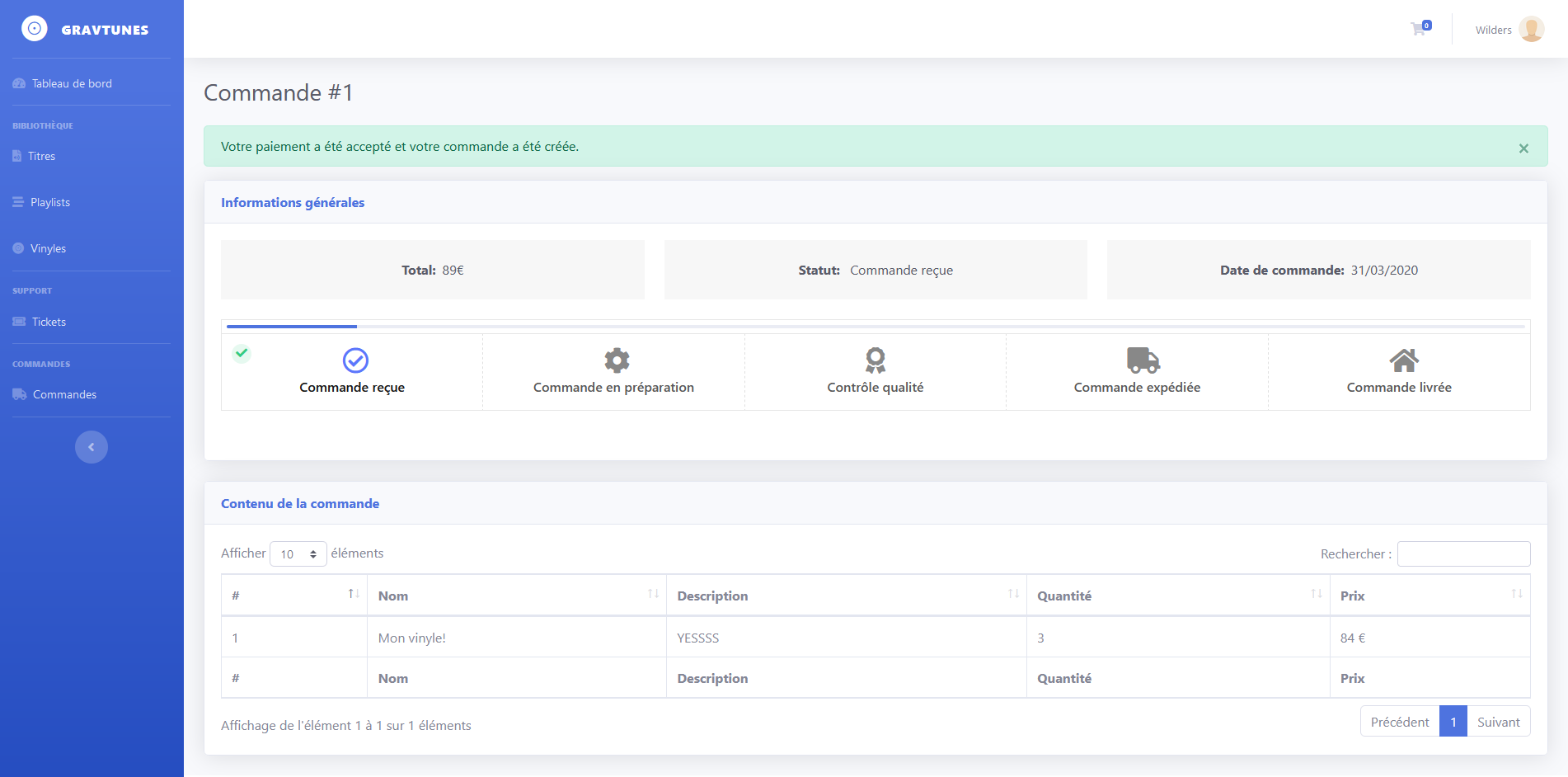


* Panier : affichage du panier ; modification de la quantité d’un article dans le panier ; suppression d’un article du panier ; possibilité de vider son panier.

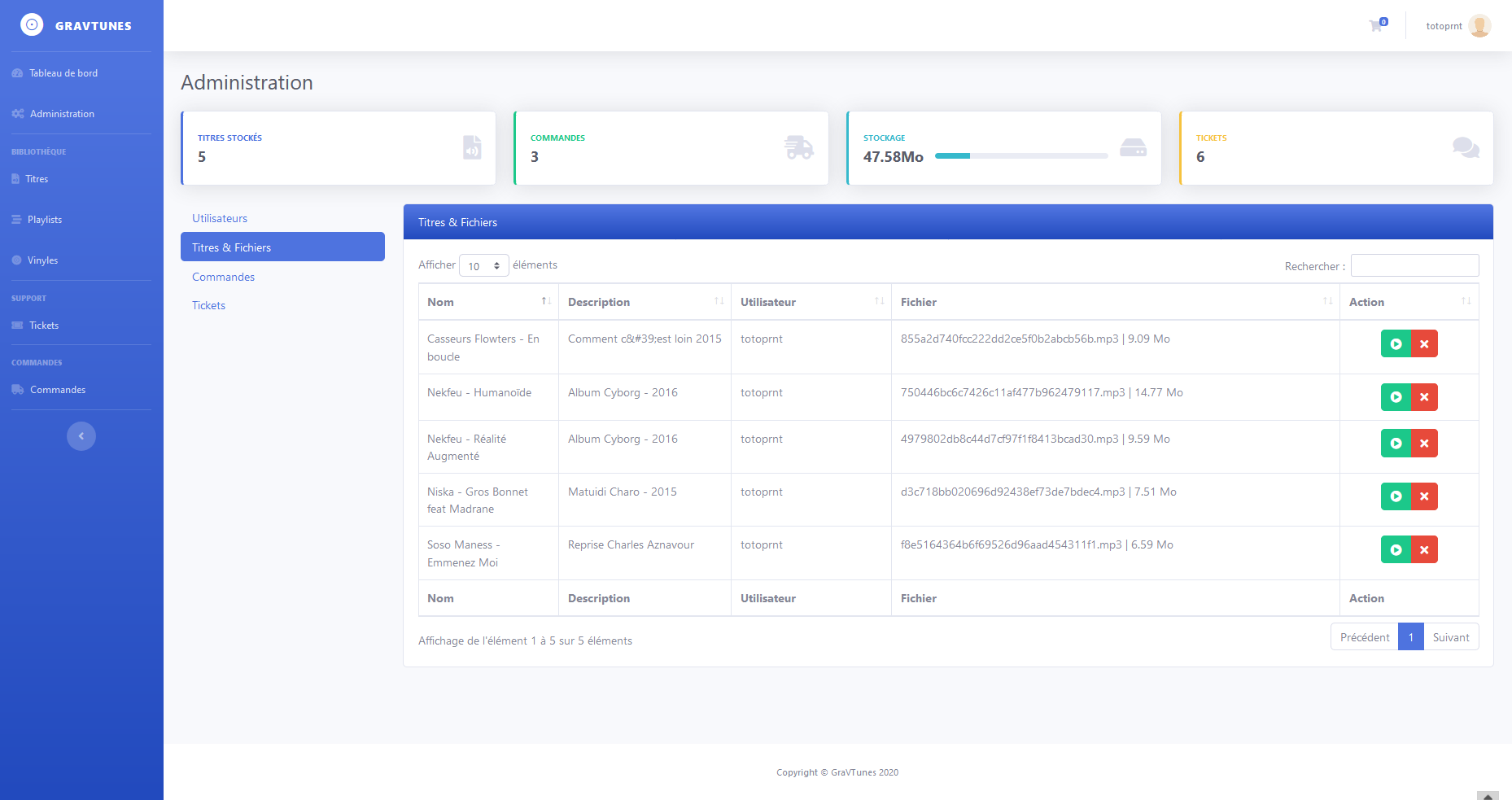


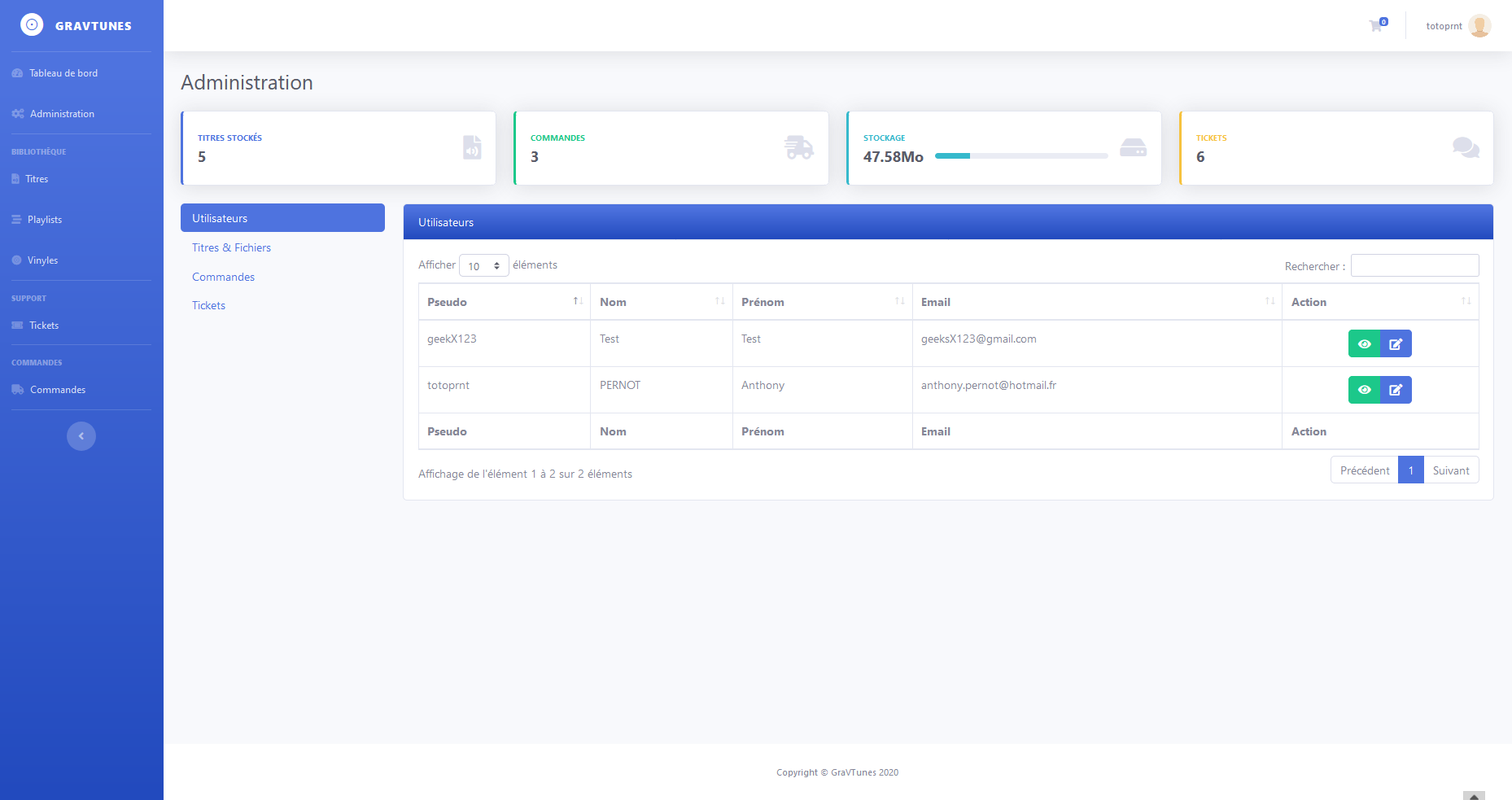
* Commande : affichage de ses commandes ; possibilité de commander un vinyle ; réaliser le payement avec PayPal ; suivre une commande (différentes étapes d’évolution d’une commande).

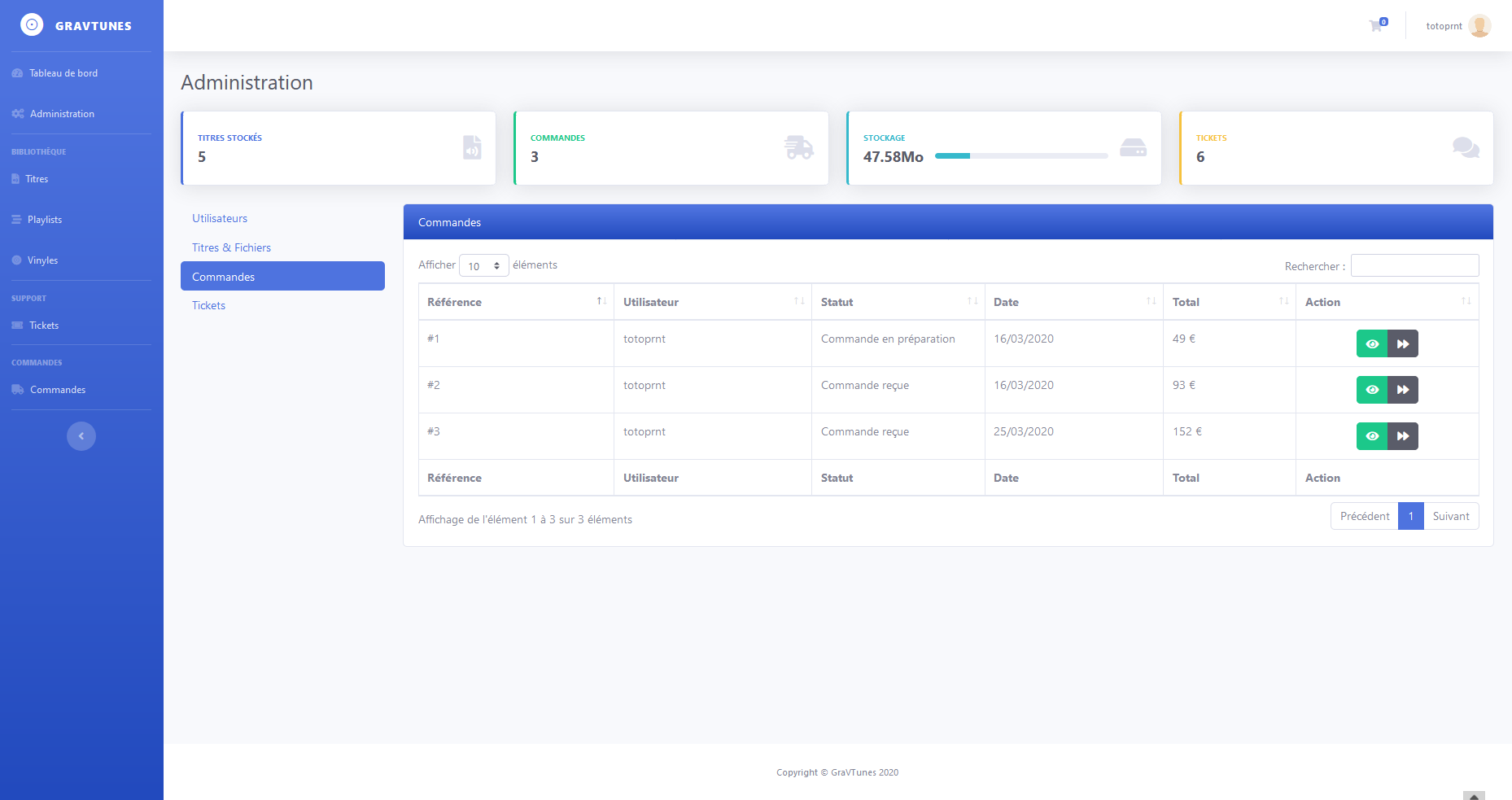


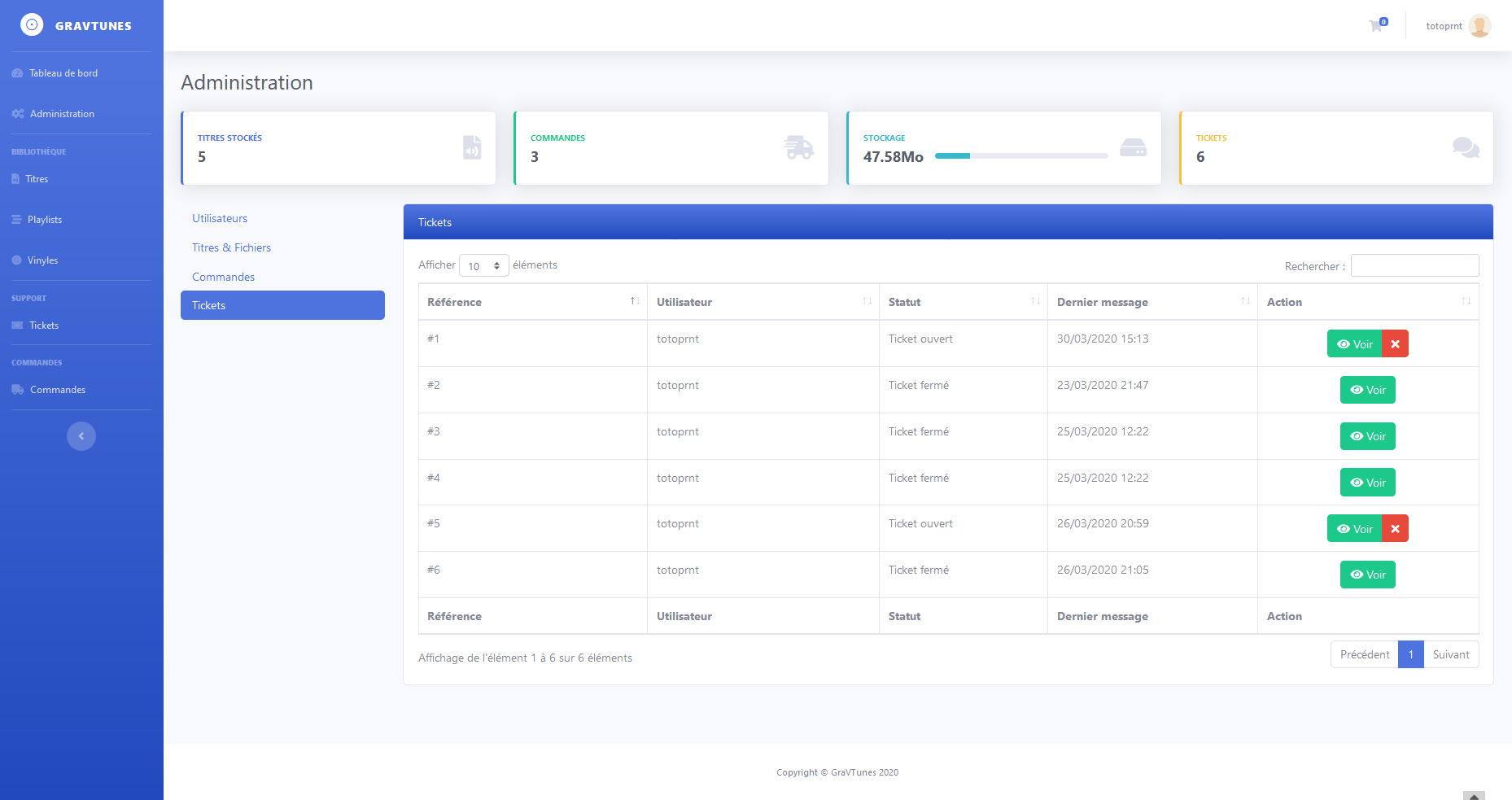


* Administration : affichage des utilisateurs, commandes, tickets, titres importés ; modification des données de l’application ; suppression des données de l’application ; répondre à un ticket utilisateur côté administrateur.









* Sécurité : protection CSRF ; affichage de messages d’erreurs ou de succès ; restriction d’accès en fonction de l’authentification ; l’envoi de mail pour les invitations à des collaborations sur un vinyle contenant la clé privée d’un vinyle.

## Difficultés rencontrées

Durant la réalisation de notre projet, nous avons été confrontés à certaines difficultés, notamment celle du travail de groupe. En effet, cela demander une bonne communication, ce qui n’a pas toujours été le cas. Mais nous avons réussi à faire face à ce problème. En effet, nous avons utilisé chacun une branche sur BitBucket pour pouvoir avancer en parallèle, et ainsi éviter les conflits. Cela nous a aussi permis de bien nous familiariser avec les Pulls Requests, qui sont une bonne pratique pour que le code soit approuvé par tous. D’ailleurs cela nous a permis d’améliorer notre productivité. La deuxième difficulté que nous avons rencontré est l’utilisation de certaines API.

En effet, cela nécessite du temps pour les comprendre et les intégrer correctement au projet. La dernière difficulté a été d’être dans les temps pour finaliser le projet et de respecter au mieux les fonctionnalités prévues puisque nous avons eu d’autres projets à faire en parallèle sur les dernières semaines.

# Conclusion

Afin de conclure ce projet, nous pouvons souligner qu’on a pu apporter un produit pratiquement fini pour ce qui est du site internet. Malheureusement, nous avons eu un manque de temps pour commencer le développement de la partie mobile de GraVTunes. Nous avons rencontré différents problèmes d’organisation qui nous a empêché d’optimiser le temps de travail à consacré. En effet, il y a eu premièrement un problème de répartition des tâches. À ce moment-là, on souhaitait surtout répartir les tâches selon les compétences de chacun. Le problème, c’est qu’il y a eu un déséquilibre sur l’apport de travail de chacun et cela a entrainé un retard sur le projet au point qu’on a dû renoncer à l’application mobile du projet. Pour remédier à ce problème, il était préférable que chacun touche un peu à tous sur le projet. Cela a poussé tout le monde à se documenter sur des fonctionnalités que l’on n’a pas vu en cours et à s’investir davantage. Finalement, nous avons préféré de travailler chacun sur sa branche BitBucket concernée par les fonctionnalités qui nous ont été attribués. Nous avons alors mis en place les Pull Requests car cela permet de faire vérifier son code aux autres membres du groupe de projet et d’avancer chacun en parallèle sur le dépôt distant. À la suite de ça, nous avons réussi à finir en grande partie les fonctionnalités initialement prévues et d’en ajouter pour compenser la partie mobile (comme l’envoi de mail ou bien l’interface de messagerie pour les tickets).

Tout de même, ce projet nous a apporté des satisfactions personnelles car on a réussi à utiliser et à mettre en place des nouvelles technologies que nous ne connaissions pas dans le cadre des cours pour certain (ce qui a ajouté un côté exploratoire au projet), et cela a permis d’améliorer nos méthodes de travail en groupe. Également, ce projet nous a éclairci sur nos poursuites d’études en trouvant nos compétences ainsi que nos faiblesses. Pour ce qui est de la suite de GraVTunes, nous pouvons éventuellement voir pour livrer le produit à l’IUT Brabois pour qu’ils puissent continuer leur service de gravure de vinyle grâce à notre outil pour gérer les interactions avec les utilisateurs.