**Rapport du projet Vue, Express, MongoDB full-stack JS :**

**Magasin de plugins audio**

****

Ce projet est une application web avec : VueJS + ExpressJS + MongoDB inspiré d’une copie de ce site réel : <https://www.moddevices.com/gear-gallery>. Ce projet est réalisé par Ines NASR et Hawaou DIALLO.

Il contient toutes les fonctionnalités importantes du site web de référence:

* Une page d’authentification pour les utilisateurs.
* Une page d’accueil présentant tous les plugins audios disponibles.
* Un formulaire d’ajout d’un nouveau Plugin Audio.
* Une page contenant le catalogue des audios.
* Une page contenant les audios d’un Vendeur connecté.

**Caractéristiques :**

**Coté Server :**

* Node.JS v10.x.x
* Express
* MongoDB v4.0
* Bundled server-side code with Webpack 2.

**Coté Client**

* VueJS2.x
* Vuex
* Vue-router
* axios

**Usage :**

Installer les dépendances présentes dans le fichier package.json

$ npm install

ou

yarn

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Coté Client \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

La Commande pour exécuter la partie client

$ npm run dev

Le Lien de navigation coté client : <http://localhost:8081/>  
L’interface qui s’affiche, permet à un visiteur (internaute) de visualiser la liste des plugins mise en vente.

Les vendeurs des audios peuvent se connecter à notre plateforme à travers ce lien : <http://localhost:8081/login>

Un vendeur connecté (authentifié) peut ajouter des audios et les gérer (modifier / supprimer).

Exemple de compte de test:

Login: [Test@gmail.com](mailto:Test@gmail.com)

Password: test

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Coté Serveur \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

La Commande pour exécuter la partie serveur

$ npm start

Le Serveur est lancé sur le port : 8080

**Mongo Database:**

On a utilisé la base de données Mongo DB

Le fichier Json des vendeurs : [lien](https://github.com/InesNASRepOUHICHI/magasin-de-plugins-audio-ProjetVusJS/blob/master/audios/server/vendeur.json)

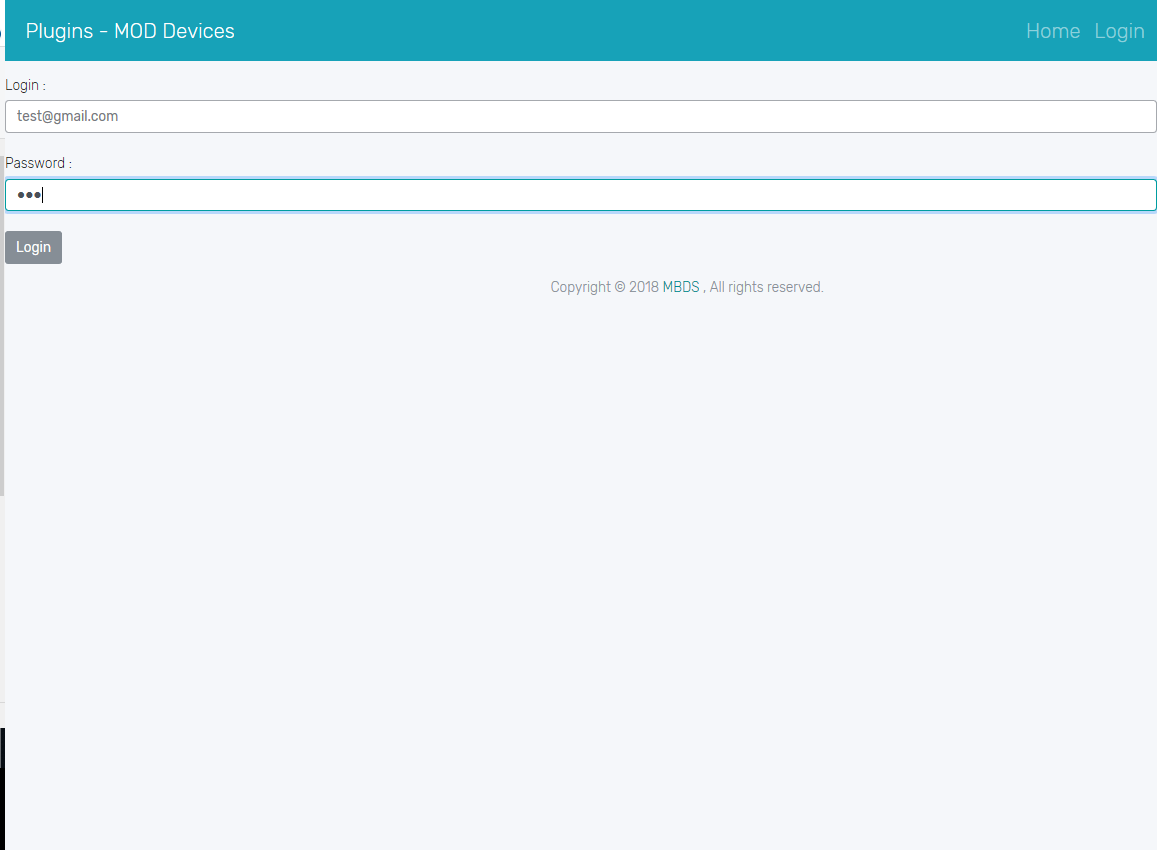
Le fichier Json des plugins audios : [lien](https://github.com/InesNASRepOUHICHI/magasin-de-plugins-audio-ProjetVusJS/blob/master/audios/server/audios.json)

Une jointure est faite entre les deux bases de données pour permettre à un vendeur d’afficher que ces propres plugins et d’ajouter son propre plugin.

**Captures d’écrans :**

**Page d'authentification**

Un utilisateur doit s’authentifier pour accéder à la plateforme.



**Page d'accueil**

Cette page contient tous les plugins disponibles :

Une image contenant capture d’écran

Description générée automatiquement

Si on scrolle on voit le catalogue des plugins disponibles :

Une image contenant capture d’écran

Description générée automatiquement

En cliquant sur le bouton « Go To Plugins Shop » en bas de la page, on peut voir tous les plugins.

**Plugins Shop**

Dans cette page, on peut chercher tous les plugins existants, naviguer entre les pages de plugins et choisir combien de page on veut afficher. L’utilisateur peut aussi voir le détail de chaque audio.

**Une image contenant capture d’écran

Description générée automatiquement**

**Mes Plugins**

Cette page contient les plugins que l’utilisateur connecté a ajoutés.

Une image contenant capture d’écran, intérieur

Description générée automatiquement

Il peut afficher les détails d’un plugin. En cliquant sur le bouton « Détails », un modèle des détails est affiché.

Une image contenant capture d’écran

Description générée automatiquement

Il peut modifier un plugin. En cliquant sur le bouton « Edit », un formulaire de modification est affiché.

Une image contenant capture d’écran

Description générée automatiquement

Il peut supprimer un plugin. En cliquant sur le bouton « Supprimer », une alerte de confirmation de la suppression est affichée.

Une image contenant capture d’écran

Description générée automatiquement

Il peut ajouter un nouveau plugin en cliquant sur le lien « Add new Audio » dans le menu. Ce lien lui redirige vers le formulaire d’ajout d’un nouveau plugin.

Une image contenant capture d’écran

Description générée automatiquement

**Structure du répertoire**

+---client

| +---build

| +---config

| +---node\_modules

| +---src

| | +---assets

| | +---css

| | +---img

| | +---js

| | +---components

| | +---AddAudios.vue

| | +---AppAudio.vue

| | +---Author.vue

| | +---Login.vue

| | +---NavBar.vue

| | +---PluginShop.vue

| | +---router

| | +---index.js

| | +---App.vue

| | +---main.js

| +---static

| | +---images

| +---index.html

| +---package-lock.json

| +---package.json

|

+---server

| +---app\_modules

| | +---audios-crud-mongo.js

| +---node\_modules

| +---audios.json

| +---package-lock.json

| +---package.json

| +---vendeur.json

Contenu des fichiers coté client :

* **AppAudio :** composant VueJS qui gère la construction de la page d’accueil où n’importe quel visiteur peut voir la liste des plugins existants.
* **PluginShop :** composant VueJS responsable de la présentation du catalogue des plugins. Un visiteur peut chercher un plugin. Il peut aussi définir le nombre maximum de plugins à afficher par page et naviguer entre les différentes pages.
* **Author :** ce composant offre à l’utilisateur connecté de modifier, supprimer ou consulter ses propres plugins.
* **AddAudio :** c’est le composant pour ajouter un nouveau plugin.
* **NavBar :** composant affichant le menu de la plateforme.
* **App :** le composant principal englobant tous les autres composants de l’application.
* **Le fichier index.html :** le point d’entrée de l’application.
* **Le fichier main.js :** dans lequel est créée l’instance principale / racine de Vue correspondant à l’entité.
* **Index.js :** ce fichier de configuration contient toutes les routes vers les différents pages de la plateforme.

Contenu des fichiers coté server :

* **Audios-crud-mongo :** ce fichier contient la configuration vers la base de données (le nom, l’url vers la base de données MongoDB) et les fonctions d’ajout, modifier, supprimer et afficher un plugin a partir de la base de donnée.
* **ServerWithMongo :** ce fichier définit les différents web services et méthodes exposées pour nos ressources (Audio et User).

**Structure d’un Plugin**

- L'audio est composé de :

-author : {

- avatar Url : URL du site du créateur,

-name: nom du vendeur

},

-brand : la marque du plugin,

-catégories : [] table de tags (mots clés qui serviront d’indexes),

-control Ports : [] table avec le détail des paramètres (nom, valeur par défaut, min et max),

-description : détail sur le plugin,

-imageHeight : dimension de plugin,

-imageWidth : dimension de plugin,

-label : label plugin,

-name : nom du créateur,

-pedalboardCount : nombre de pedalboard qui l’utilisent,

-screenshot Url : image du plugin,

-stable : stabilité du plugin,

-thumbnailUrl : url du thumbnail d’un plugin,

-uri : Lien vers l’image du plugin,

-version : version du plugin

}