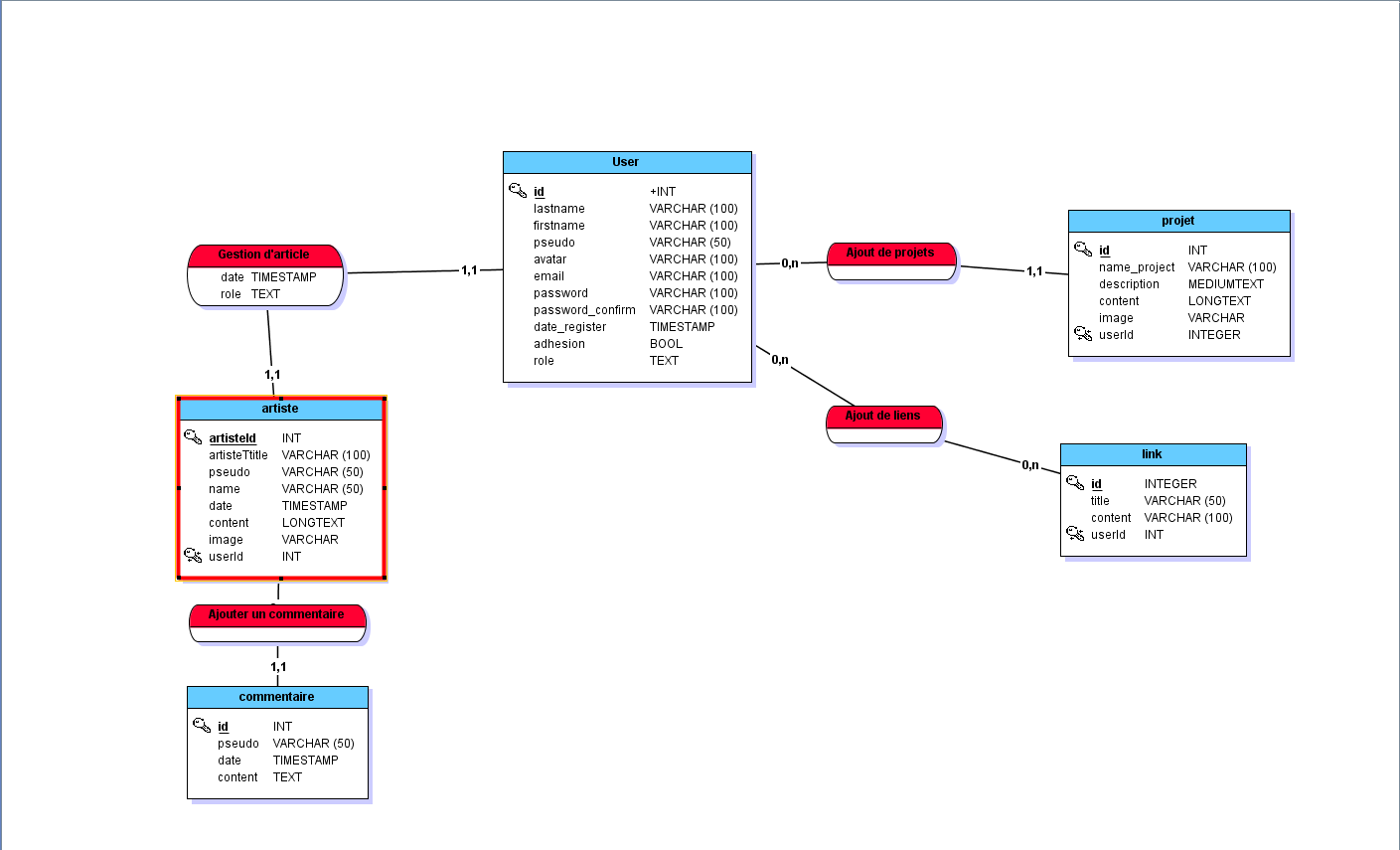
Use case, mcd, wireframe, maquette, charte graphique :

**Les uses cases** :

* Les artistes (users) : Les artistes pourront au sein du système gérer leurs projets (add, update et delete), ajouter des liens externes pour permettre aux visiteurs de savoir où se diriger pour les soutenir ou en découvrir davantage sur ce qu’ils font. Pour cela, ils ont besoin d’adhérer au collectif en s’inscrivant via le formulaire d’inscription, l’inscription sera d’abord vérifiée par l’admin.

* L’administrateur (users) : l’admin pourra réaliser les articles de présentations des artistes et les gérer (add/update/delete), sur un back office il gérera également les demandes d’adhésions des artistes au collectif.
* Les visiteurs : les visiteurs pourront lire les articles de présentation et également les commenter, il l’en est de même pour les projets des artistes.

**Le mcd :**

* 

Il s’agit d’une représentation d’une base de données facilement compréhensible du fonctionnement du système de base de données.

J’ai défini dans un premier temps une table Users, la différence entre l’administrateur et l’user classique qui représente l’artiste se situe au niveau du champs rôle. Le champs rôle contient deux valeurs : « Admin » et « Artistes », selon le rôle qu’aura l’user la redirection ne serait pas identique.

* Un user ayant le rôle « Admin » pour gérer la création/la modification/la supression des articles de présentation des artistes. Il gère également les demandes d’adhésions au collectif.

L’user qui aura le rôle « artiste » accèdera au back-office lui permettant de gérer la création/la modification/la suppression de projets. Il a également la possibilité d’ajouter des liens externes.

**Les wireframes & les maquettes :**

Pour la réalisation de ce projet j’ai adopté la méthode mobile first en procédant à la réalisation, en priorisant sur l’aspect mobile du projet j’ai conçu les wireframes de la version mobile/tablette et dekstop de la page home, le back-office du user et ainsi que son profil. Pour la réalisation des maquettes j’ai utilisé le logiciel justimind pour le grand nombre de fonctionnalité qu’il propose et son accessibilité très intuitive. Il en est de même pour les maquettes.

**La charte graphique** :

**La typographie**: Comme pour l’ensemble du design du site web, j’ai souhaité rester sur des police sobre mais permettant tout de même d’attirer l’œil du visiteur.

* **LES TITRES**: les titres de type h1 ou h2 auront la font Teko, une police à la structure sobre se mariant correctement aux titre.
* **LES PARAGRAPHE**: pour garder cette synergie, la police font est utilisé pour les paragraphes.
* **LES MENU :** les menu avec la police **source sans pro.**

**Les couleurs :**

Les deux couleurs principales ont été choisit pour garder cette cohérence dans la sobriété du site, la dernière couleur a été choisit afin de faire ressortir certains éléments du site pour les rendre plus visibles (tels que le menu, etc)

**Le Back :**

* **Type de bdd :**
* **Les models/schémas :**
* **Les contrôleurs :** on insère dans les fichiers contrôleur les function call-back des routes où on effectue les requêtes. J’ai adopté cette méthode afin d’alléger et découper mon code au niveau du serveur qui commençait à gagner en volume et donc manquer en visibilité. Ces contrôleurs effectuent donc la logique des routes définis en appelant les méthodes définis dans le fichier contrôleur.

**Contrôleurs Auth :**

Le contrôleur Auth contient la logique d’inscription et de connexions de mes routes **register/login.**  Le contrôleur est passé en fonction callback de deux routes respectives afin « d’importer » les méthodes que j’ai conçu pour la connexion et l’inscription. Afin de sécuriser la phase de connexion, j’ai procédé à la méthode la jwt (json web token), pour cela dans un premier temps j’ai créé un fichier caché qui contient mes variables d’environnement, dans le cas présent la clé secrète utile à l’utilisation de token. L’utilisation de package **dotenv** va permettre de trouver ce fichier caché, pour pouvoir ensuite grâce au **process.env** récupérer ma variable d’environnement.

* **Les routes :**

**user/register :**

* Cette route permet aux utilisateurs (artistes) de procéder à l’adhésion au collectif en remplissant ce formulaire d’inscription. J’ai mis en place la gestion des cas d’erreurs au sein du contrôleur.
* Les cas d’erreurs :

**user/login :**

**Le Front :**