



Dossier de spécifications techniques et fonctionnelles

Sommaire

Architecture de GeniArtHub	3
La page d'accueil	3
La page détail d'une oeuvre	3
La page panier	3
La page confirmation de commande.....	4
Endpoints de l'API.....	4
Paramètres.....	5
Validation de commande	5
Informations importantes	5
Le serveur back	6

Architecture de GeniArtHub

Le site web GeniArtHub est spécialisé dans la vente d'œuvres d'art réalisées par intelligence artificielle. Le site se compose de 4 pages spécifiques.

La page d'accueil

La page d'accueil permet d'afficher de manière dynamique toutes les œuvres d'art. Les œuvres affichées sur la page d'accueil proviennent d'un serveur back développé avec NodeJS et Express.

Le serveur back est fourni avec le projet GeniArtHub.

La page détail d'une oeuvre

La page de détail d'une œuvre d'art est une page qui s'affiche lorsque l'utilisateur clique sur une œuvre d'art sur la page d'accueil. Pour afficher une œuvre en particulier, il faut faire appel à serveur back en lui transmettant l'id de l'œuvre en question.

Sur la page de détail des oeuvres, on trouve :

- Le titre long
- Une description tronquée récupérée de la description longue
- Le prix de l'oeuvre par défaut (la première oeuvre
- Un champ text permettant de renseigner la quantité d'oeuvres à ajouter
- Un champ select qui permet de sélectionner une oeuvre parmi les différents formats disponibles
- La description complète

Le prix de l'œuvre dépend du format choisi.

Les déclinaisons ne possèdent pas d'identifiants propres, vous allez devoir utiliser leur taille comme identifiants.

La page panier

La page panier est en deux parties. La première partie contient la liste des produits qui sont présents dans le panier. La deuxième partie contient un formulaire de commande qui permet d'effectuer une commande.

La liste des produits doit afficher :

- L'image
- Le titre long
- Le format choisi
- Le prix unitaire
- La quantité choisie

- Un lien de suppression

Le total de la commande et le nombre d'articles dans le panier doivent être calculés automatiquement.

Lorsque la quantité d'un produit change, le nombre total d'articles et le montant doivent être recalculés automatiquement.

Le formulaire permettant d'effectuer la commande doit contenir des données bien formatées.

Les données doivent également être valides. Ce qui signifie :

- Prénom : Minimum 2 lettres, pas de caractères spéciaux
- Nom : Minimum 2 lettres, pas de caractères spéciaux
- Adresse : Au moins 10 caractères
- Ville : Au moins 3 caractères, pas de chiffres
- Email : Doit être un mail valide

!! ATTENTION – Chaque action effectuée (modification des quantités, suppression de produit, validation du formulaire) doit pouvoir fonctionner sans rechargement de la page.

La page confirmation de commande

La page de confirmation de commande doit afficher le numéro de commande récupéré depuis le back lors de la validation d'une commande.

Si le numéro de commande a bien été récupéré, les données dans les champs de formulaire de commande doivent être réinitialisées.

Les données du panier doivent également être vidées.

Endpoints de l'API

- <http://localhost:3000/api/products/>
- <http://localhost:3000/api/products/{product-id}>
- <http://localhost:3000/api/products/order>

Paramètres

Méthode	URL	Requête	Réponse
GET	/		Tableau d'objets de toutes les oeuvres
GET	/{{product-id}}		Tableau d'objet correspondant à l'élément dont l'id est transmis dans l'endpoint
POST	/order	JSON contenant un objet de contact et un tableau de produits	Objet de contact, tableau de produit et numéro de commande

Validation de commande

Afin de valider la commande, sur la route /order, les informations que le serveur doit recevoir doit être :

- Un objet contact avec les champs firstName, lastName, address, city et email
- Le tableau contient un array de strings avec les id des oeuvres

Pour le moment, le serveur back est en phase de développement, pour les commandes le système ne prend en compte que l'envoi d'id. Par la suite, il sera possible d'envoyer l'id, les déclinaisons et les quantités.

Informations importantes

Pour l'ajout au panier vous devez utiliser le localStorage pour stocker les données. Vous devez prendre en compte les informations suivantes :

- Si l'article n'existe pas dans le panier, vous devez l'enregistrer dans le localStorage
- Si l'article existe déjà dans le panier, vous ne devez pas l'avoir en double dans le localStorage, vous devez modifier la quantité dans le localStorage
- Vous devez enregistrer dans le localStorage les différentes déclinaisons de produit
- Vous ne devez pas dupliquer votre localStorage, vous devez enregistrer les données dans un tableau
- Pour des raisons de sécurité, vous ne devez PAS enregistrer le prix des produits dans le localStorage
- Vous devez limiter à 100 unités par produit dans le panier et sur la page d'ajout de produits

Le serveur back

Le serveur back nécessite NodeJS pour fonctionner. Vous devez vous positionner dans le dossier back et lancer la commande « npm install » pour installer les dépendances.

Une fois les dépendances installées, vous n'avez plus besoin de relancer la commande d'installation. Vous devez lancer le serveur en utilisant la commande « node server »

A chaque fois que vous devez travailler sur le projet, n'oubliez pas de lancer le serveur avec la commande « node server ».