

Gromangin Clément
Holder Jules

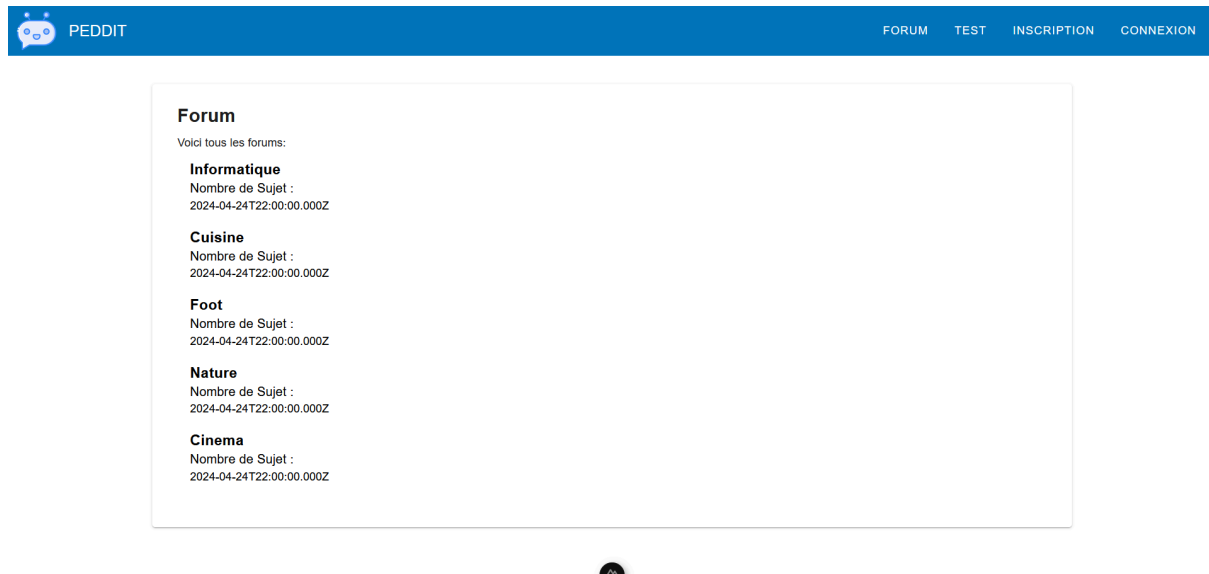
Projet Nuxt: Forum

Github : <https://github.com/ClemGrom/Gromangin-Holder-ProjetNuxt>

Sommaire :

- 1. L'application**
- 2. Fonctionnalités réalisées**
- 3. Installation**
- 4. Les choix techniques**
- 5. Difficultés rencontrées**

1- L'application



Forum interactif qui s'actualise en temps réel.

2- Fonctionnalités réalisées :

- L'application s'articule sous la forme d'une simple hiérarchie Forums \Rightarrow Sujets \Rightarrow Messages. On peut donc avoir plusieurs forums, qui eux-même contiennent plusieurs sujets, qui eux-même contiennent plusieurs messages. Un sujet contient toujours un message initial, qui est écrit par l'utilisateur qui ouvre le sujet.
- Les utilisateurs doivent créer un compte et s'authentifier à l'application pour pouvoir créer des sujets ou poster des messages. Toutefois, il n'est pas nécessaire d'être authentifié pour consulter les forums, sujets et messages postés. Deux niveaux de droits existent, un utilisateur simple, et un administrateur. Un compte par défaut "admin" (avec le mot de passe "admin") sera créé automatiquement par la plate-forme lorsqu'elle est lancée pour la première fois
- Les utilisateurs doivent pouvoir se déconnecter via un bouton "Déconnexion"
- Lorsque l'on arrive sur la page d'accueil, celle-ci doit lister l'ensemble des forums disponibles, avec le nombre de sujets au sein de chaque forum.
- Toujours sur la page du forum, un bouton "Nouveau sujet" permettra de créer un nouveau sujet. L'utilisateur doit alors saisir un titre, et un premier message pour le sujet (champ texte multi-lignes). Ce bouton n'est accessible que si l'utilisateur est connecté.
- Les sujets doivent être paginés, à raison de 20 sujets par page.
- Lorsque l'on sélectionne un sujet, on doit arriver sur une page qui liste tous les messages du sujet, par ordre de date de message croissant. Le premier

message (initial) est donc en haut de la page 1, le dernier message posté en réponse sera en bas de la dernière page. Les utilisateurs connectés peuvent cliquer sur un bouton Répondre pour pouvoir apporter une réponse au sujet, sous la forme d'un champ texte, comme pour le message initial à la création d'un sujet.

- Les messages doivent indiquer pour chacun le nom de l'utilisateur qui l'a posté, la date à laquelle le message a été posté, et évidemment le message en lui-même.
- Les messages doivent être paginés, à raison de 20 messages par page.
- Un utilisateur doit pouvoir modifier un de ses propres messages via un bouton "Modifier" présent au niveau de chaque message qu'il aura posté. Un administrateur peut supprimer un message posté
- Un espace administrateur supprimer un forum. Lorsqu'un forum est supprimé, tous ses sujets et messages doivent l'être également **si le bouton ne s'affiche pas : localhost:3000/admin**

3- Installation:

Pré-requis: docker lancé

- Cloner le dépôt avec la commande
"git clone [git@github.com:ClemGrom/Gromangin-Holder-ProjetNuxt.git](https://github.com/ClemGrom/Gromangin-Holder-ProjetNuxt.git)"
- Se rendre dans le dossier "Forum_Holder_Gromangin"
- "cd Forum_Holder_Gromangin"
- Effectuer la commande
"docker compose up -d"
- Effectuer ensuite la commande
"npm i && npm run dev"
- attendez une dizaine de secondes
- Se rendre sur <http://localhost:3000>

Email de admin : **admin@gmail.com**

mdp : admin

4 - Choix techniques :

Nous avons principalement utilisé les technologies vues en cours, à savoir une application réalisée avec le framework NuxtJS, Pinia comme système de store, bcrypt pour crypter les mots de passe, bluebird et mysql pour l'API avec Nuxt. Comme framework pour le style, vuetify a été utilisé.

5- Difficultés rencontrées

websocket : nous avons d'abord réalisés l'application sans utiliser websocket pour d'abord vérifier que toutes les requêtes api marchent.

Ce qui a été une erreur, car nous n'avons pas eu le temps de bien utiliser le websocket.

Docker : La base de donnée et l'adminer tournent tous deux sur le docker, mais nous avons rencontré des difficultés quant à la dockerisation de node pour faire tourner le serveur nuxt dans un conteneur : une erreur de format de module nous empêchait de lancer correctement celui-ci et nous n'avons ainsi pas réussi à intégrer totalement notre appli sous conteneur Docker.