

Projet Web ESIEA CFA 4A

Par Mohand ABDELOUHAB et Gabriel MANDRAUD

1. Description du projet

Suite à la consigne énoncée en cours, nous avons décidé de réaliser un site de « e-commerce » (gratuit) qui va nous permettre de télécharger des mods pour le célèbre jeu ARMA 3. Pour parler des fonctionnalités du site, il doit donc comporter un système d'authentification qui permet à un utilisateur ou à l'admin de se connecter. Il doit y avoir un système de téléchargement de fichiers. Concernant les pages, on a une page d'accueil avec un carrousel présentant les derniers articles ajoutés. Un page « A propos » qui résume quelques informations à propos du site. On retrouve une page Magasin qui va donc lister tous les articles disponibles. Chaque article dispose d'un bouton ajouter au panier qui permet donc d'ajouter l'article choisi au panier de l'utilisateur. Une pagination dynamique a aussi été ajoutée à la page du magasin pour permettre d'avoir plusieurs pages d'articles.

Un fois qu'un utilisateur se connecte il est autorisé à remplir son panier. Dans cette page panier, on va retrouver une liste des articles sélectionnés. On peut supprimer des articles du panier et télécharger le panier. Lorsque l'on se connecte en administrateur, un onglet administration apparaît qui va permettre d'ajouter de nouveaux articles.

2. Etapes de conception, réalisation et déploiement

2.1. Conception

Structure :

- **Accueil**
- **Magasin**
- **A propos**
- **Panier**
- **Connexion**
- **Administration (*droits admin requis*)**

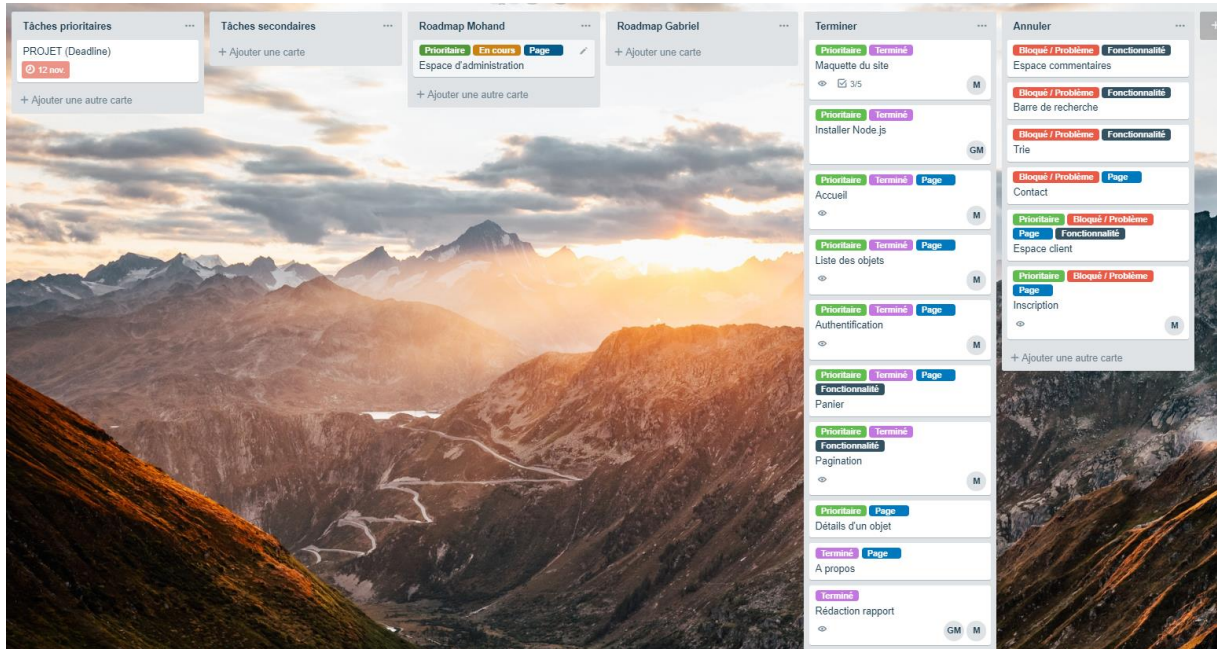
Lors de la réalisation de cette application, nous avons utilisé :

- Le Framework VueJS (version complète avec Webpack)
- NodeJs pour la partie Serveur
- Bootstrap pour VueJS
- Photoshop pour la réalisation de la maquette
- Trello pour la gestion de projet

Un planning de projet a été réalisé avec l'outil Trello afin de nous permettre d'avoir une visibilité sur l'avancement du projet.

Une maquette du site a été réalisée sous Photoshop pour avoir un aperçu du rendu final de l'application et ainsi mieux anticiper la répartition des tâches.

Une liste des fonctionnalités a été déterminée suivie par une répartition de tâches.



Lien du Trello : <https://trello.com/b/5QHXmcJv/web-esiea-project>

2.2. Réalisation

Partie VueJS

Après la réalisation de la maquette sous Photoshop vient l'étape de l'intégration.



En s'appuyant sur la documentation officielle de VueJS, nous avons commencé à réaliser les fonctionnalités du site.

Tout d'abord, nous avons mis en place un carrousel dynamique pour la page d'accueil qui représente les 3 derniers articles ajoutés à la base de données. Vient ensuite la réalisation du magasin où nous avons listé tous les articles envoyés depuis la base de données tout en utilisant une pagination dynamique qui va nous permettre de garder une certaine mise en forme du magasin et d'éviter que les articles débordent de leur cadre.

La page Panier gère les articles qui ont été ajoutés à l'aide d'une variable globale depuis la page du Magasin. Il est impossible de remplir le panier tant que l'utilisateur n'est pas authentifié.

La page d'authentification permet à l'utilisateur de s'authentifier et de pouvoir utiliser les fonctionnalités du site. Si cet utilisateur est référencé en tant qu'administrateur, il aura accès à un panneau d'administration qui va lui permettre de poster des articles.

Par manque de temps, il nous a été impossible d'ajouter la gestion des utilisateurs.

La page A propos permet de présenter le site ainsi que l'équipe de développement.

Partie NodeJS

Un serveur NodeJS a été mis en place afin d'assurer l'authentification et l'envoi de données aux différents clients.

Structure :

Fonction GET /items : permet d'afficher les items

Fonction GET /len : permet de renvoyer le nombre d'items

Fonction POST /upload : permet d'upload les images

Fonction POST /uploadMod : permet d'upload les mods

Fonction POST /post : permet d'ajouter un article

Fonction POST /auth : permet l'authentification des utilisateurs

Fonction GET/getLastImgs : permet d'alimenter le carrousel avec les trois derniers articles ajoutés

Fonction GET /download : permet de créer une archive zip avec les mods sélectionnés lors de la validation du panier.

2.3. Déploiement

Pour le déploiement, nous avons mis en place une Raspberry Pie qui héberge le serveur NodeJS et l'application.

3. Difficultés rencontrées et solutions apportées

Lors de la réalisation de ce projet, nous avons rencontré quelques problèmes. Notamment pour le partage des informations entre les différents composants de l'application. En effet les composants sont étanches et pour un projet de ce type il est indispensable qu'ils puissent communiquer entre eux. La solution que nous avons mise en place consiste en l'utilisation d'un fichier Javascript qui contient l'objet et qui est accessible par tous les composants par une simple importation.

Un autre problème est lié à l'utilisation de la version complète de VueJS ou à l'utilisation du CDN. En effet, au début du projet on a utilisé la version CDN mais on s'est rendu compte que sur le long terme cette méthode n'est pas adaptée notamment pour maintenir l'application.