

# **582-31F-MA**

## **Programmation d'interface Web 2**

### **TP 1**

# Description

Produire une application dynamique à l'aide de *JavaScript* utilisant la programmation orientée objet modulaire.

## Pondération

25%

## Modalité particulière

Travail individuel

## Sujet

Librairie

## Date de remise

Cours 12 - lundi 12 février 2024

## Modalités de remise

- Le dossier zippé, à votre nom (nomdefamille-prenom) doit m'être remis sur Léa avant le 12 février 23h59.
- Le site doit être en ligne avant le 12 février 23h59.
- Dans ce dossier, vous devez inclure un fichier **.txt** avec l'URL de votre site.

## Retard

Selon les règles du collège, 5 % par jour de retard seront enlevés, jusqu'à 5 jours de retard maximum.

# Énoncé

Vous avez obtenu le contrat pour la refonte du site d'une librairie en ligne. Pour le premier *sprint*, votre équipe vous a attribué le développement de la page listant les livres en inventaire. Celle-ci affiche les livres sous différentes catégories ainsi que le détail d'un livre suite au clic de sa tuile. Aussi, vous devez développer, côté client, la fonctionnalité d'ajouter au panier. Comme votre travail doit se greffer à celui de vos collègues, il vous faudra respecter le devis ainsi que les exigences techniques.

## Consignes

- Le *backend* n'est pas complété, du coup vous devrez travailler avec la structure de données statiques fournie (fichier **livres.js**).
- Toutes les images sont également données. Je vous invite à les placer suivant l'arborescence **./assets/img/**, autrement il vous faudra changer tous les chemins relatifs du fichier **livres.js**.
- Au chargement de la page, seuls les 12 premiers livres sont injectés dans la page.
- Il y a 8 filtres, ceux-ci sont :
  - Tous
  - Nouveautés
  - Littérature
  - Art de vivre
  - BD, Jeunesse, Humour
  - Culture et société
  - Loisirs, Tourisme, Nature
  - Savoir et science

## Consignes (suite)

- Vous aurez compris qu'il faudra afficher les livres correspondants au filtre cliqué.
- Remarquez que *Nouveauté* n'est pas une catégorie, mais bien une clé présente pour chaque livre. Bref, au clic de Nouveauté, vous devez injecter tout livre où sa clé **nouveaute** est **true**.
- Tâchez d'éviter la redondance, faire quatre boucles distinctes mais similaires pour les quatre différents cas (chargement de page (12 premiers livres), Tous, Nouveauté et les différentes catégories) serait peu optimal.
- Assurez-vous de faire la **gestion UX / UI** du filtre actif.
- Chaque tuile d'un livre doit afficher son image, son titre, son prix et un bouton pour l'ajouter au panier.
- Au clic d'une tuile (toute la tuile doit être cliquable), un modal affiche les informations de ce livre, soit : son image, son titre, son auteur, son éditeur, son nombre de pages et sa description.
- Une bonne stratégie serait d'injecter l'index du livre courant en **dataset** lors de l'injection de sa tuile.
- Prévenez le **scroll** en Y lorsque le modal est ouvert.
- Une fois le modal ouvert, l'utilisateur peut le fermer en cliquant n'importe où sur celui-ci. Prenez tout de même la peine de placer un X en haut à droite.
- Évidemment, si le **scroll** en Y a été prévenu en état ouvert, il faudra redéfinir ce comportement à la fermeture du modal.

## Consignes (suite)

- Au clic d'un bouton *Ajouter*, vous devez ajouter ce livre au `localStorage` assigné au panier. Les informations à conserver à l'intérieur du `localStorage` sont le titre et le prix du livre. Pour ce faire, vous devrez placer ces informations dans un objet littéral qu'il faudra `push` à l'intérieur d'un tableau. C'est ce tableau qui sera gardé en mémoire.
- Vous avez peu de gestion à faire sur ce panier, vous devez toutefois vous assurer de ne pas ajouter un livre déjà présent dans le panier.
- Au clic de l'icône panier (SVG) à la droite du `<header>`, affichez dans un petit modal le détail du panier ainsi que le prix total avant taxe à l'intérieur d'un élément `<table>`.
- S'il n'y a qu'un livre dans le panier, l'intitulé du premier `<th>` sera 'Livre'. S'il y a plus qu'un livre, ce même intitulé sera 'Livres'.
- Si toutefois le panier est vide, injectez uniquement le paragraphe 'Il n'y aucun livre dans votre panier.'
- Au clic suivant sur l'icône panier, si le modal est affiché, fermez-le.
- Assurez-vous d'utiliser le préfixe `#` pour déclarer comme *private* toutes propriétés et méthodes qui n'ont pas à être publiques.
- Comme la presque majorité des méthodes seront privées, le seul test automatisé demandé valide que chaque livre dans la structure de données `livres` possède les clés `titre`, `auteur`, `description`, `prix`, `editeur`, `pages`, `image`, `nouveaute` et `categorie`.

## Plus-value (nice- to-haves)

- Placer son projet sur `GitHub`.
- Changer la couleur du panier selon son état, par exemple l'icône est gris lorsque le panier est vide et vert dès qu'au moins un livre a été ajouté. Assurez-vous également de faire cette gestion au chargement de la page en interrogeant le `localStorage`.
- Ajouter des transitions animées pour les différents éléments de l'interface graphique.
- Utiliser une structure de fichiers CSS tel que ITCSS.

## Exigences techniques et critères d'évaluation

- Réussite des différentes fonctionnalités
- Utilisation stricte de la syntaxe des classes ES6 de type *Module*
- Structure et optimisation du code
- Qualité des algorithmes et qualité du code source
- Utilisation du préfixe `#` pour toutes propriétés et méthodes privées
- Absence de fonctions fléchées (*arrow functions*)
- Aucune librairie externe
- Code sémantique
- Commentaires fréquents et pertinents
- Ergonomie générale de l'interface
- Le site est en ligne

# Plagiat

Nous travaillons dans le Web, alors les réponses s'y trouvent et il va de soi que vous allez devoir chercher sur *Google*. Dans l'esprit *open-source* du Web, vous avez tout à fait le droit de copier-coller un bout de code que vous avez trouvé. Cela-dit, vous devez citer les extraits de code qui dépassent une ligne ou deux. Évidemment, générer du code via un outil d'intelligence artificiel tel que *ChatGPT* et/ou suivre un tutoriel (en tout ou en partie) sera considéré comme du plagiat. Pour prévenir le copier-coller direct d'Internet, aucune syntaxe autre que celles vues en classe ne sera acceptée.

Il s'agit d'un travail strictement individuel. Vous pouvez vous aider quelque peu entre vous, mais faites attention à ne pas partager ou récupérer des fonctionnalités complètes. Je rappelle encore une fois qu'un des objectifs principaux de la formation est de développer votre autonomie, c'est important. En cas de plagiat, je devrai appliquer les sanctions de la *PIEA* pour les étudiant.e.s ayant reçu et partagé le code.

Si vous êtes vraiment bloqué, contactez-moi.