

# Projet javascript – 2023-2024

Pour clore le parcours javascript, vous développerez un mini-projet (hors SAE).

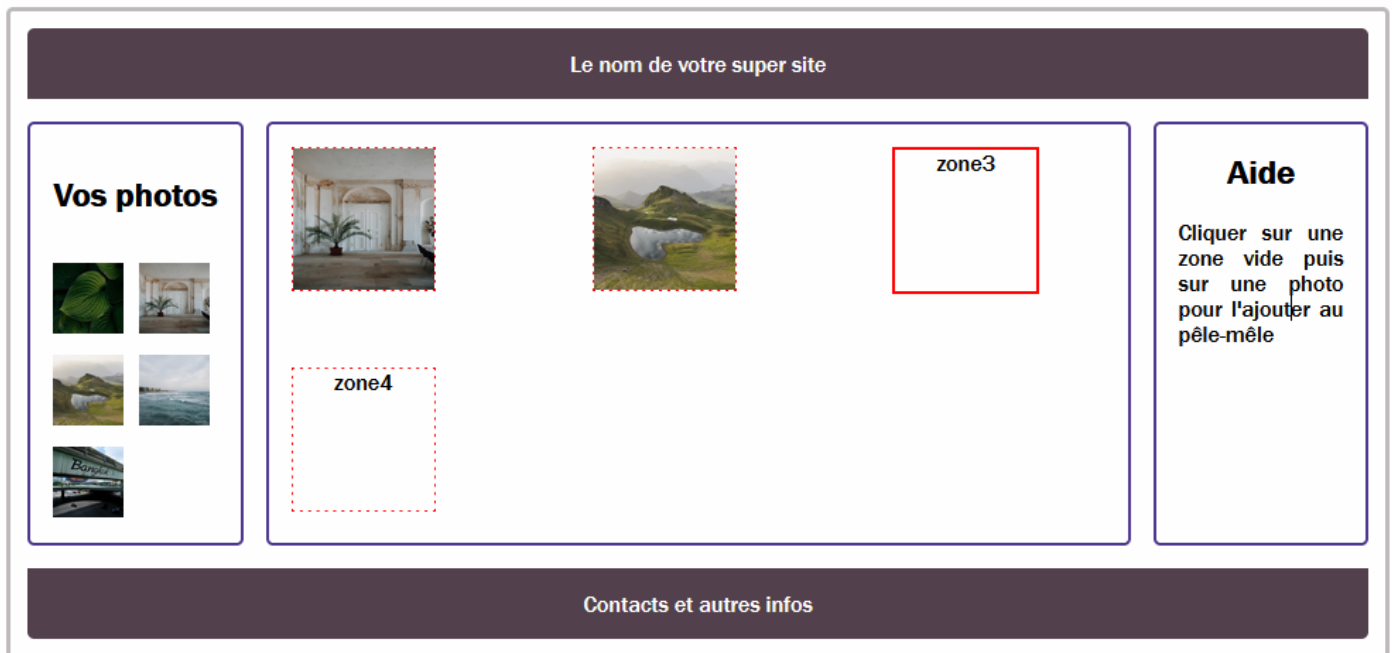
**Equipes** : binômes au choix à constituer au plus tard la première séance dédiée au projet. Il est aussi autorisé de le faire seul.

**Livrables** :

- lien vers votre page mise en production sur le serveur IUT (public\_html) ou sur le Gitlab de la forge de l'université.
- Lien vers votre dépôt (repository) sur Gitlab. Accès aux enseignants en rôle « Reporter »

**Echéance** : **lundi 04 décembre**. Le projet doit être développé en classe et préparé au sujet du DS. Pénalité en cas de retard.

**Sujet** : Développer une application d'édition d'un "poster pêle-mêle" à partir de photos récupérées via une API.



## Fonctionnalités principales :

- ✓ Les images récupérées sur le site Unsplash ou Pexels apparaissent dans le cadre de gauche.
- ✓ Les zones de dépôt apparaissent au milieu.
- ✓ Quand un utilisateur veut ajouter une image dans le poster, il clique sur la zone à sélectionner dont le cadre devient rouge, puis il clique sur une photo de la partie gauche.
- ✓ Quand l'utilisateur clique à côté des images ou des zones, toutes les zones redeviennent désélectionnées.

## Plan de travail suggéré

### Préambule

Récupérer le dossier compressé sur Moodle contenant le fichier html, le fichier css, les images de tests.

Analyser la structure du code html.

Mise en place du Git sur la forge en utilisant le template pages/Plain HTML. Documentation : <https://perso.liris.cnrs.fr/pierre-antoine.champin/enseignement/intro-web/iut.html>

### Mise en place du fichier js

Faire apparaître 6 photos (du dossier img) dans la partie gauche (rappel : il n'est pas autorisé de modifier le code html). Pensez à imposer une largeur et une hauteur de px.

Faire apparaître les 4 zones dans la partie centrale.

## Interaction

Mettre en place des listeners au clic pour la partie gauche et pour la partie centrale. Inutile de mettre un listener sur chaque image et chaque zone, il faudra utiliser target.

Au clic sur une image ou sur une zone, l'élément visé est encadré en rouge (cf classe selected du css).

Au clic sur une image de la partie gauche, on parcourt les zones de la partie centrale. Si l'une d'elle est sélectionnée, alors on récupère la source de l'autre image pour la faire apparaître.

## Intégration d'images externes

Se rendre dans la partie documentation pour développeurs du site de partage de photos « Unsplash » ou « Pexels ».

Créer un compte et récupérer une clé d'accès. En effet, l'accès aux photos depuis un autre serveur suit le standard CORS (cf chap 5).

### Avec « Unsplash »

L'url pour une requête vers une photo au hasard sur le site est la suivante :

`https://api.unsplash.com/photos/random/?client_id=votre clé sans guillemet`

### Avec « Pexels »

L'url pour une requête vers des photos sur un thème donné est la suivante :

`https://api.pexels.com/v1/search?query=sport`

Votre fetch doit contenir un deuxième paramètre sur ce modèle :

`fetch(url, {Header : {Authorization: votre_Clé} })`

Modifier votre code pour que les photos de la partie gauche soient issues du site unsplash.

Le nombre de requête est limité à 50 par heure sur Unsplash et 20 000 par mois sur Pexels. (cf documentation). Prévoir que les images de secours sont affichées quand la requête déclenche une erreur.

## Partie facultative (attention votre SAE S3 est prioritaire !)

Le projet se prête à beaucoup d'améliorations :

- Rendre les zones redimensionnables par l'utilisateur.
- Mettre en place un drag and drop
- Permettre à l'utilisateur de modifier la couleur du fond du poster (en utilisant la partie droite)
- Par un clic sur une photo, la photo est agrandie et affichée devant le poster avec la mention de l'auteur et un petit drapeau indiquant le pays de la prise de vue (autre API).
- Interroger unsplash avec un thème choisi par l'utilisateur.
- Améliorer les parties html et css.
- Interroger d'autres API de photos (Flickr, Pexels, ...)

## Grille d'évaluation

Critères (notes sur 10)	Coefficient	Echelle
Fonctionnalités principales	4	0-2 : le site ne fonctionne pas du tout ou très peu 3-7 : le site fonctionne globalement mais pas certaines fonctionnalités 8-10 : le site fonctionne parfaitement et résiste aux différents tests.
Qualité du code	5	0-2 : le code n'est pas du tout structuré correctement. <b>Il ne franchit pas le test jshint.</b> 3-5 : le code est correct mais il y a trop de morceaux de codes récupérés et mal compris. 6-8 : le code est tout à fait conforme à ce qui est attendu (commentaire, structure) 9-10 : le code peut être publié à titre d'exemple
Gestion du Git	1	0-3 : le git n'est pas tout utilisé correctement : pas de commit des deux développeurs, pas de structure correcte. 4-7 : utilisation basique du git mais encore beaucoup d'erreurs

		8-10 : Utilisation optimale du commit avec des merge, request...
Investissement continu	2	0-3 : Aucun avancement pendant les séances 4-7 : Avancement correct mais fortes modifications du code entre les séances 8-10 : Projet intégralement fait pendant les heures de cours.
Fonctionnalités facultatives	Bonus : 2 points maximum	Il s'agit de valoriser les étudiants ayant développé d'autres fonctionnalités mais nous ne voulons pas que ce projet prenne trop d'importance.