

se servir de l'IA en tant que traducteur et dictionnaire

Utiliser les LLM pour générer des paysages à partir de mes illustrations sur un site web interactif. Les utilisateur.ice pourront alors déplacer les images pour créer leur propre composition.

À partir de ces manipulations, l'IA analysera et interprétera les prompts fournis en s'appuyant sur ma base de données personnelle, afin de proposer des associations pertinentes.

La création de cette liste d'associations par l'IA s'est déroulée en plusieurs étapes:

Premier prompt pour vérifier si ChatGPT comprend ma demande: je lui fournis trois exemples d'images au format PNG et lui demande de les associer aux mots correspondants.

Ensuite, je lui propose davantage d'images et lui demande d'explorer d'autres méthodes pour faire correspondre les mots aux images sous format PNG.

clementine=	[«clementine1.png», «clementine2.png»]
amour=	[«heart1.png», «rose.png»]
anniversaire=	[«bougiebleu.png», «bougierose.png», «etoile1.png», «etoile2.png»]
fête=	[«bougiebleu.png», «bougierose.png», «clementine1.png», «fraise1.png», «pomme1.png»]
nature=	[«fraise1.png», «fraise2.png», «feuilles1.png», «feuilles2.png», «feuille4.png», «sapin6.png», «sapin5.png», «sapin4.png»]
hiver=	[«sapin6.png», «sapin5.png», «sapin4.png», «pommedepin1.png», «pommedepin2.png»]
plage=	[«coquillage1.png», «coquillage2.png», «coquillage3.png», «coquillage4.png», «coquillage5.png», «coquillage6.png»]
forêt=	[«sapin6.png», «sapin5.png», «sapin4.png», «pommedepin1.png», «pommedepin2.png», «feuilles1.png», «feuilles2.png», «feuille4.png»]
campagne=	[«maison1.png», «maison2.png», «pomme1.png», «pomme2.png», «pomme3.png», «poire1.png», «clementine1.png», «clementine2.png»]
lettre=	[«lettre.png», «timbre.png»]
promenade=	[«maison1.png», «sapin5.png», «feuille4.png», «fourmis.png»]
fruit=	[«fraise1.png», «fraise2.png», «pomme1.png», «pomme2.png», «pomme3.png», «poire1.png», «clementine1.png», «clementine2.png», «mûre.png»]
noël=	[«sapin6.png», «sapin5.png», «sapin4.png», «pommedepin1.png», «pommedepin2.png», «etoile1.png», «etoile2.png»]
désert=	[«briquet.png», «cigarette1.png», «cigarette2.png», «ecouteur.png»]
village=	[«maison1.png», «maison2.png», «saccourses.png», «lettre.png», «fourmis.png»]

Dans ma deuxième partie, j'écris un long prompt détaillant chaque élément pour que ChatGPT puisse tout intégrer. Ce prompt est divisé en deux parties : la première vise à associer les images à des mots, et la seconde à établir des liens entre différents types de mots existants et ceux associés aux fichiers PNG. Je fournis également l'intégralité de ma liste de fichiers PNG.

«Tu dois créer un fichier texte contenant un prompt en utilisant des mots simples. Ce prompt doit inclure les titres des fichiers PNG que je vais te fournir, en les associant de manière logique. Par exemple, pour «clementine», tu utiliseras les fichiers «clementine1.png» et «clementine2.png» ou fournis= fourmi.png et tu l'écrira cette première partie de cette manière la;
«pomme»: [«assets/pomme1.png», «assets/pomme2.png», «assets/pomme3.png»], (bien mettre le virgule à la fin)

Dans la deuxième partie tu fais la liaison de différents sortent de mots existant et ayant un rapport avec les mots affilié au png de la première partie, c'est à dire par exemple fruit= clementine et pomme puis l'écrire de cette manière la ;

«fruits»: [«clementine», «pomme»].

Tu me fera une très longue liste de mot genre vraiment où on peut écrire un maximum de mot ou de phrase en tête, la deuxième partie contient un maximum vraiment vraiment beaucoup de mots associés aux titres des fichiers PNG.

D'autre précision si sur le noms des png il y-a des chiffres c'est que c'est le même éléments mais dessiné de différentes manière de même ce que j'ai mis entre parenthèse dans la liste représente la couleur de l'objet.

«fraise»:	[«assets/fraise1.png», «assets/fraise2.png»],	"fruits":	["fraise", "pomme", "clementine", "kaki", "poire", "mûre"],
«ruban»:	[«assets/ruban.png»],	"nature":	["sapin", "arbre", "buisson", "feuille", "champignon", "gland"],
«feuille»:	[«assets/feuille1.png», «assets/feuille2.png», «assets/feuille4.png»],	"maison":	["maison"],
«sapin»:	[«assets/sapin4.png», «assets/sapin5.png», «assets/sapin6.png», «assets/arbre3.png», «assets/sapin1.png», «assets/sapin2.png», «assets/sapin3.png»],	"plage":	["coquillage"],
«maison»:	[«assets/maison1.png», «assets/maison2.png»],	"animaux":	["fourmi", "escargot"],
«fourmi»:	[«assets/fourmi.png»],	"objets":	["lettre", "timbre", "bic", "briquet", "bougie", "cigarette", "ecouteur", "verre"],
«timbre»:	[«assets/timbre.png»],	"hiver":	["sapin", "buisson", "arbre", "etoile", "gland"],
«mûre»:	[«assets/mûre.png»],	"été":	["fraise", "clementine", "pomme", "coquillage"],
«coquillage»:	[«assets/coquillage1.png», «assets/coquillage2.png», «assets/coquillage3.png», «assets/coquillage4.png», «assets/coquillage5.png», «assets/coquillage6.png», «assets/coquillage.png»],	"automne":	["feuille", "champignon", "marron", "pomme", "sapin"],
«verre»:	[«assets/verre1.png», «assets/verre2.png», «assets/verre3.png», «assets/verre4.png», «assets/verre5.png»],	«textures»:	[«rugueux», «lisse», «pointu», «mou», «solide»],
«lettre»:	[«assets/lettre.png»],	"forêt":	["sapin", "arbre", "buisson", "champignon"],
«escargot»:	[«assets/escargot.png»],	"campagne":	["maison", "feuille", "buisson", "pomme", "radis"],
«gland»:	[«assets/gland.png»],	"plage":	["coquillage", "pissenlit"],
«arbre»:	[«assets/arbre.png», «assets/arbre1.png», «assets/arbre3.png»]	«fêtes»:	[«anniversaire», «cadeau», «fête», «noël»],
		«camping»:	[«briquet», «escargot», «pissenlit», «fourmi», «feuille», «coquillage», «radis», «tente», «feu», «cuisine», «lampe», «sac», «randonnée», «carte», «sac de couchage», «réchaud», «coussin», «insectes», «sleeping bag»],

Je remarque qu'en lui demandant de créer une longue liste, ChatGPT commence à inventer des mots qui ne font pas partie de la liste de base. Je revois donc mon prompt et y ajoute des précisions:

« rester strictement sur les mots donnés dans la première partie pour construire la deuxième partie, sans inventer de nouveaux termes ! »

```
{
«fruits»:      [«fraise», «pomme», «clementine», «poire», «kaki», «mûre»],
«nature»:      [«feuille», «sapin», «arbre», «champignon», «buisson», «pissenlit», «pomme
depin», «gland», «marron», «cailloux», «coquillage»],
«objets»:      [«ruban», «verre», «lettre», «briquet», «bic», «bougie», «timbre», «sac
courses», «cigarette», «ecouteur»],
«décorations»: [«bougie», «ruban», «sapin», «etoile»],
«animaux»:     [«fourmi», «escargot»],
«maison»:      [«maison», «lettre», «timbre»],
«végétation»:  [«feuille», «pissenlit», «radis», «buisson», «sapin», «arbre»],
«coquillages»: [«coquillage»],
«matières»:    [«cailloux», «verre», «marron»],
«arbres»:      [«arbre», «sapin», «pommedepin»],
«fleurs»:      [«pissenlit», «radis»],
«insectes»:    [«fourmi», «escargot»],
«petits_objets»: [«bic», «briquet», «lettre», «timbre», «ruban», «bougie»],
«plantes»:     [«feuille», «pissenlit», «buisson»],
«pierres»:     [«cailloux», «marron»],
«verres»:      [«verre»],
«légumes»:     [«radis»],
«écritures»:   [«lettre», «timbre»],
«forêt»:       [«arbre», «sapin», «champignon», «feuille», «gland», «pommedepin»,
«marron»],

«plage»:       [«coquillage», «cailloux»],
«étoiles»:     [«etoile»],
«anniversaire»: [«bougie», «ruban»],
«accessoires»: [«ruban», «bic», «bougie»],
«fêtes»:       [«bougie», «ruban», «sapin», «etoile»],
«littoral»:    [«coquillage», «cailloux», «verre»],
«bois»:        [«pommedepin», «marron», «gland»],
«matériaux»:   [«cailloux», «verre», «marron»],
«symboles»:    [«etoile», «ruban»],
«hiver»:       [«sapin», «marron», «pommedepin»],
«printemps»:   [«pissenlit», «radis», «feuille»],
«été»:         [«coquillage», «buisson»],
«automne»:     [«marron», «gland», «pommedepin»],
}
```

De même, je lui demande d'ajouter de nouvelles fonctions dans le fichier texte .json, permettant d'affilier les mots ou images à leur taille, ainsi qu'à leur mouvement (par exemple, un arbre reste plutôt fixe, tandis qu'une clémentine est malléable).

```
«sizes»: {  
  
  «petite»: [«feuille»,»fraise», «ruban», «fourmi», «pommedepin»,»timbre»,  
  pomme»,»champignon», «mûre»,»bougie», «coquillage», «escargot»,»lettre», «kaki»,  
  «poire», «bic», «briquet», «cigarette», «ecouteur», «pissenlit», «radis», «marron»,  
  «gland»],  
  
  «moyenne»: [«sapin», «arbre»,»cailloux», «verre»,»saccourses», «clementine»,  
  «saccourses», «buisson», «ambre», «roche»,»etoile»],  
  
  «grande»: [«maison», «sapin», «arbre» ]  
},  
  
«rotation»: {  
  
  «rotatable»: [«bic», «pommedepin», «clementine», «gland», «coquillage»,»bougie»,  
  »feuille»,»kaki»,»poire»,»pomme»,»radis»,»marron»,»etoile»]  
},
```

Je lui demande de créer une interaction entre mon fichier .json (fichier texte) et mon script .js, afin de prendre en compte les mots présents sur le site web et de leur ajouter des options, comme leur position sur la page, ou agir sur la taille des images ou les rotations des images à chaque chargement de page.

De plus, dans ce javascript on peut retrouver d'autres fonction pour introduire une dimension interactive au site, notamment en permettant de déplacer des éléments de créer un <canvas> pour permettre que les images générées restent dans leur limites définies.

Voici un exemple d'une partie du code .js

```
fetchDataJSON().then(data => {  
  let mapping = data.mapping; // Récupère les informations des images  
  let synonyms = data.synonyms; // Récupère les informations association mots/images  
  let stopWords = data.stopWords; // Récupère les informations des mots à ignorer  
  let sizes = data.sizes; // Récupère les informations de tailles (petite, moyenne, grande)  
  let rotationData = data.rotation; // Récupère les données de rotation
```

// Fonction pour générer des images

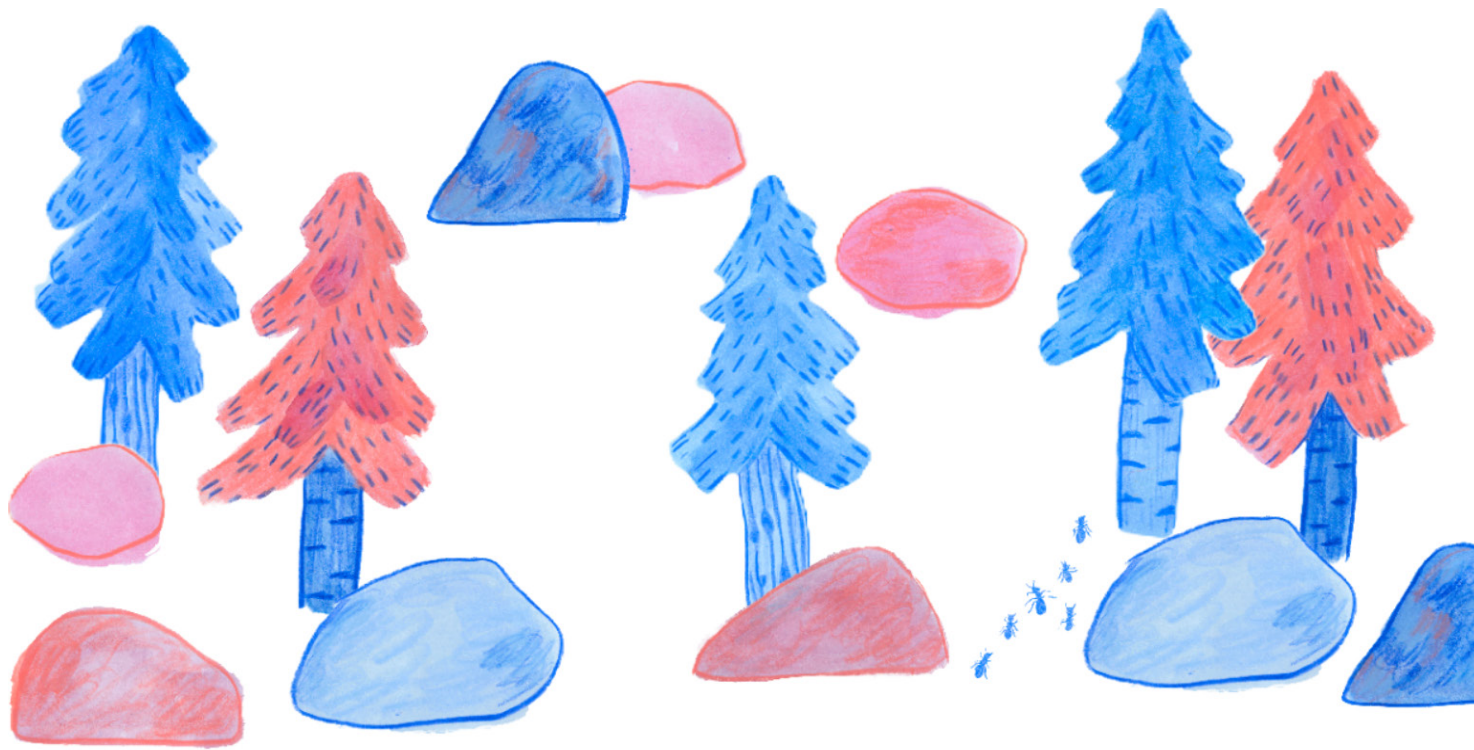
```
function generateImage(assets) {  
  ctx.clearRect(0, 0, canvas.width, canvas.height);  
  imagePositions.length = 0; // Réinitialiser les positions des images  
  
  const keywordCounts = assets.reduce((counts, assetSrc) => {  
    counts[assetSrc] = (counts[assetSrc] || 0) + 1;  
    return counts;  
  }, {});  
  Object.entries(keywordCounts).forEach(([assetSrc, count]) => {  
    for (let i = 0; i < count; i++) {  
      const img = new Image();  
      img.src = assetSrc;  
  
      img.onload = () => {  
        let imageWidth = img.width;  
        let imageHeight = img.height;  
  
        const keyword = Object.keys(mapping).find(key => mapping[key].includes(assetSrc));  
        const size = getSizeForKeyword(keyword); // Récupérer la taille du mot-clé (petite, moyenne, grande)  
  
        const sizeFactors = {  
          petite: 0.05,  
          moyenne: 0.12,  
          grande: 0.1  
        };  
  
        const scale = sizeFactors[size] || 0.6; // Par défaut, taille moyenne si non définie  
        imageWidth *= scale;  
        imageHeight *= scale;  
      });  
    }  
  });  
}
```



je met la table pour mon anniversaire



mes vacances en campagne



je me balade en montagne



dans mon jardin je regarde les etoile