

# Test technique - Front-end



Hello padawan !

Pour chaque environnement, les langages attendus sont :

- Front-End Web : HTML5, CSS3 et JavaScript
- Backend : Language libre (le back-end doit seulement servir à rendre le front)

## Principe du mini-projet

Tu devras développer la page d'accueil de notre site internet à partir de la maquette jointe (en format JPG et PSD).

Pour ouvrir le fichier PSD, tu auras besoin de Photoshop, ou à défaut tu pourras utiliser un outil en ligne gratuit comme Photopea. Tu devras donc exporter les images directement depuis la maquette, ou depuis le site <https://studiozerance.fr>.

## Consignes

**La page** comporte 5 sections : un header, une bannière, un portfolio, un slider et un « text infinite ».

Tu as l'entière liberté sur le choix de la grid CSS, à l'exception d'Avalanche CSS qui est interdit.

Tu ne peux pas utiliser de JS, sauf pour la section « slider ». Pour cette section, tu peux faire appel à la lib que tu souhaites à l'exception de Flickity.

Le rendu devrait ressembler au maximum à notre maquette !

Tu devras ajouter la font « Karla », qui se trouve en pièce jointe.

Les sections devront être séparées de manière distincte dans le code.

1. Section « header » :
  - Les menus Nos clients, Notre offre et Le studio doivent respectivement renvoyer vers les section 3, 4 et 5. L'onglet « Contact » doit ouvrir la popup de contact dont le fonctionnement est détaillé plus bas
2. Section « Bannière »
  - Le texte doit être entre balises H1, le mot Shopify souligné comme sur la maquette.
3. Section « portfolio »
  - L'image Yoko doit rediriger vers le site <https://yokoshop.com>
4. Section « slider »
  - Au hover, le bouton « Voir plus » doit apparaître en italique
  - il y a 2 slide dans le slider, dont les 2 images de fond sont en pièce jointe
  - le contenu textuel des 2 slides est identique
  - La flèche doit donc mener au slide suivant
  - On doit voir apparaître le début du slide suivant sur la droite
  - Il est interdit d'utiliser Flickity. Cependant, on te recommande l'utilisation d'une lib ;)
5. Section « text infinite »
  - Le texte doit loop à l'infini de la droite vers la gauche, de manière fluide.
  - il est interdit d'utiliser du JavaScript pour cette section

**Sur mobile**, tu as l'entière liberté concernant le responsive. Le contenu doit être compréhensif et cohérent, sans bug (chevauchement, image surdimensionnées, etc...) Tu peux aussi décider de cacher une partie du contenu si ce choix te paraît pertinent, justifie le en quelques mots dans le ReadMe.

**La popup de contact** doit s'ouvrir quand on clique sur « contact » et se fermer quand on clique en dehors de la popup ou à l'aide d'une croix.

Son design est totalement libre, mais la popup devra comporter les éléments suivants :

- un input text « prénom » pour entrer le prénom
- un input mail « mail » pour entrer l'adresse mail
- un input text « message » afin d'entrer le message
- un bouton « valider » afin d'envoyer le message.

Son comportement devra être le suivant :

- il doit y avoir une gestion d'erreur qui empêche de valider si on a pas rempli toutes les cases.
- quand on clique sur « Valider », tu devras récupérer le genre de la personne grâce à son prénom , en envoyant une requête à une des nombreuses « Gender API » existant gratuitement en ligne. Ces API permettent de déterminer le sexe d'un prénom.

En fonction de la réponse tu devras afficher une popup de confirmation avec écrit :

- « Rosebud » si le sexe est masculin
- « We rob banks » si le sexe est féminin
- « It's a trap ! » si le sexe n'est pas déterminé ou s'il y a une erreur

Ainsi qu'un bouton OK fermant la popup de confirmation ET la popup de contact.

*Si tu ne sais vraiment pas à quoi ressemble une popup de contact, tu peux chercher **ici**.*

**Optionnellement** et si tu connais l'environnement Shopify, tu peux coder directement dans Shopify. Sur demande nous t'enverrons les accès d'une boutique vierge dans laquelle est installée un thème vierge.

Nous analyserons les schémas dans tes sections, la structure de ton code dans Shopify, et nous t'encourageons à laisser des options.

## **Rendu**

Pour que nous puissions évaluer ton code tu devras héberger ton mini-projet sur un serveur git public comme GitHub. Tu devras rédiger un fichier README.md expliquant comment installer et lancer ton application. Tu nous fourniras donc simplement un lien vers ton répertoire GitHub.

Dans ce README.md, nous souhaitons aussi que tu détailles le fonctionnement technique de ta page, que tu nous parles des difficultés que tu as rencontré et de la manière de laquelle tu les as surmontées.

Si tu n'as pas réussi certaines parties de l'exercice, ce n'est pas grave ! Explique nous le problème, les solutions que tu as envisagées et les workaround que tu as trouvés.

Tu indiqueras également le temps que tu as passé à développer cette page, idéalement section par section.

Si tu estimes à plus de 7h le temps de développement de cet exercice, nous te déconseillons de te lancer.

Au plaisir de te rencontrer. Nous restons à ta disposition en cas de questions.

Bon code et à très vite !

L'équipe Zerance

PS : Bien sûr, la triche est interdite et éliminatoire ;)