

Explications détaillées de l'implémentation des Wireframes et Mockups

Les Wireframes représentent des schémas réalisés dans le but de définir les zones et composants d'une interface. Ils ne doivent pas être suivis esthétiquement.

Les Mockups représentent la version finale de l'esthétique de l'application. L'objectif étant de réaliser l'intégration des interfaces en suivant au plus possible les détails des Mockups.

Accompagné de ce fichier, il sera fourni un fichier Figma(.fig), permettant de récupérer l'ensemble des détails de ces Mockups, tels que les ombres ou les polices. Figma n'étant pas parfait, nous ne vous recommandons pas de suivre les tailles de pixels de ces mockups, mais bien d'adapter votre développement au contenu que vous êtes en train d'implémenter.

Afin d'accéder au contenu du fichier Figma, nous vous invitons à créer un compte sur le site <https://www.figma.com/>, et à importer le fichier.

Nous vous fournissons aussi les Mockups en fichier pdf, pour si vous rencontrez des difficultés avec le fichier .fig de Figma.

Comme vous pouvez le voir, Il y a eu quelques changements de contenus entre la réalisation des Wireframes et la réalisation des Mockups. Merci de suivre en priorité les contenus d'interfaces des Mockups.

Quels technologies utilisés ?

Les besoins de cette application demandent un framework frontend. Nous vous laissons le choix libre entre Angular, React ou VueJs.

Avant d'attaquer la réalisation de ces interfaces directement sur un framework, nous vous demandons de faire ces implémentations uniquement en pages HTML et CSS, qui sont très facilement transférables sur un framework.

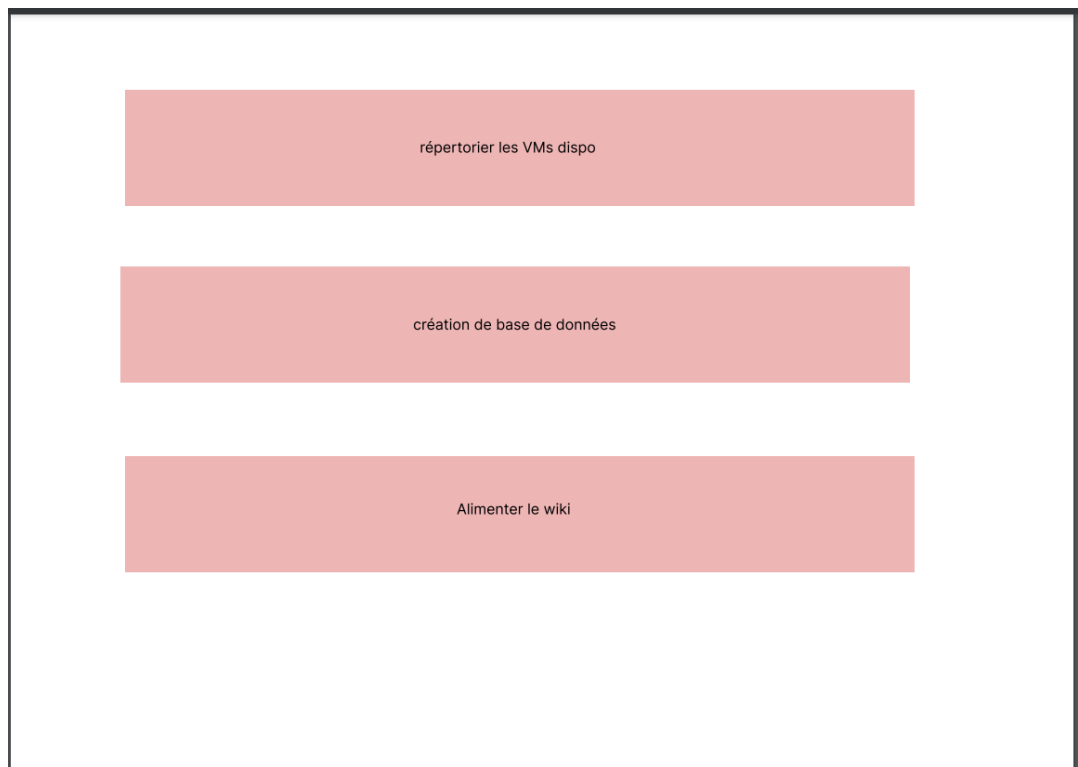
Vous ne devriez pas avoir besoin de Javascript pour le moment.

Chaque interface de cette application devra avoir sa propre page HTML ainsi que sa propre page CSS.

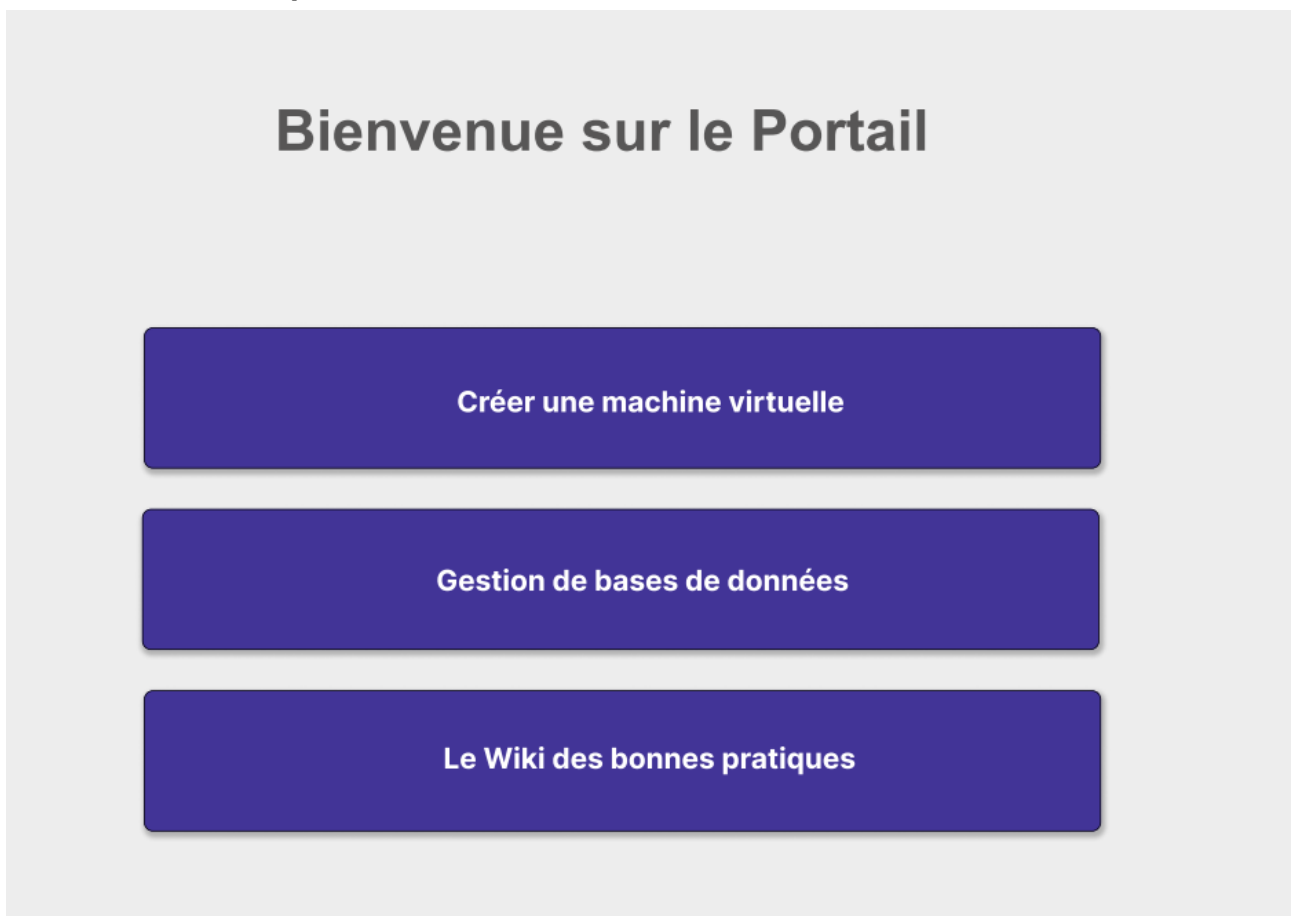
par exemple: "interface_portail.html" et "interface_portail.css" .

Nous vous invitons à modifier votre implémentation existante afin d'arriver au résultat attendu.

Wireframe “interface du portail” :



Mockup “interface du portail” :



Explication de “interface du portail” :

Voici la page d'accueil du portail. C'est la page de redirection par défaut.

Le titre est réalisé en police GoogleFont “Arial”.

Les 3 boutons bleu, durant le clique, doivent avoir une animation d'enfoncement lors du clique.

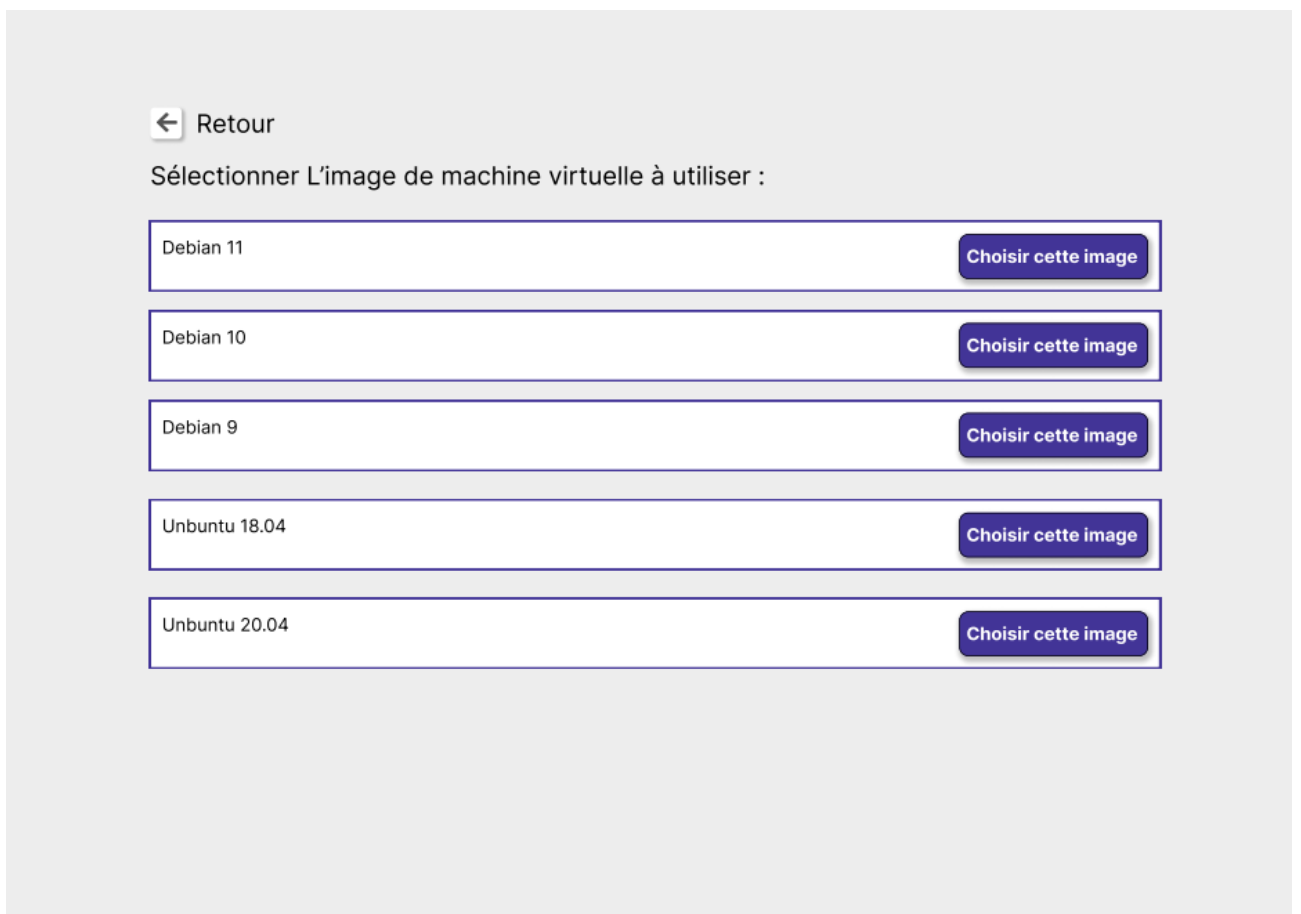
Les 3 boutons bleu sont réalisés en police GoogleFont “Inter”.

Le bouton “Créer une machine virtuelle”, au clique, renvoi vers “interface de listing des images”.

Wireframe “interface de listing des images” :



Mockup “interface de listing des images” :



Explication de “interface de listing des images” :

Voici la page qui liste les différentes images disponibles pour la création de machine virtuelle.

L'intégralité des textes sur cette interface sont réalisés en police GoogleFont “Inter”.

Le bouton blanc avec la flèche, durant le clique, doit avoir une animation d'enfoncement lors du clique.

Le bouton blanc avec la flèche, au clique, renvoie vers “interface du portail”.

Le texte “Retour” n'est pas interactif.

Nous retrouvons sur le mockup un élément récurrent, représenté en blanc avec une bordure bleu.

Nous appellerons cet élément un “composant-image”.

Chaque “composant-image” représente une image de machine virtuelle disponible pour l'utilisateur. Il doit donc y avoir autant de “composant-image” qu'il y a d'images disponibles pour l'utilisateur.

Le texte noir dans ce “composant-image” est le nom de l'image qu'il représente. Il n'est pas interactif.

Chaque “composant-image” contient un bouton bleu. Cela sert à l'utilisateur à sélectionner l'image qu'il souhaite utiliser.

Ce bouton, durant le clique, doit avoir une animation d'enfoncement lors du clique.

Ce bouton, au clique, renvoie vers “interface de création d'instance”.

Nous vous invitons à simuler les informations de images de machine virtuelle lors de l'implémentation de cette interface.

Wireframe “interface de création d’instance” :

Lancer La VM

formulaire

nom de la machine:

taille de la machine:

taille de la machine(processeur):

taille de la machine(memoire):

Lancer La VM

(Le formulaire doit rassembler tous les champs utiles pour pouvoir créer une VM à partir de l'image préalablement choisi.

Mockup “interface de création d’instance” :

[← Retour](#)

Caractéristiques techniques de la machine virtuelle :

Nom de l'image sélectionnée : **Debian 11**

Nom de la machine :

Nombres de processeurs :

Taille de la mémoire vive : MB

Créer une instance de Machine virtuelle

Explication de “interface de création d’instance” :

Voici la page qui permet d’insérer les caractéristiques techniques de l’instance de machine virtuelle que l’utilisateur est en train de créer.

Le bouton blanc avec la flèche, durant le clique, doit avoir une animation d’enfoncement lors du clique

.Le bouton blanc avec la flèche, au clique, renvoie vers “interface du portail”.

Le texte “Retour” n’est pas interactif.

Il y a dans cette interface un formulaire avec 4 champs.

Le premier champ, représenté par l’intitulé “nom de l’image sélectionnée”, contient l’information du nom de l’image sélectionnée.

L’input de ce champ doit esthétiquement ressembler à un texte classique, et ne doit pas être cliquable ni modifiable. Il sera quand même pris en compte lors de l’envoi du formulaire.

Les 3 autres champs de ce formulaire ont des inputs cliquables et modifiables par l’utilisateur.

L’input relié à l’intitulé “Nom de la machine” sera de type texte.

L’input relié à l’intitulé “Nombre de processeurs” ainsi que l’input relié à l’intitulé “Taille de la mémoire vive” sera de type number.

Le bouton bleu avec l’intitulé “créer une instance de machine virtuelle” est le bouton qui va valider et envoyer le formulaire. Les 4 inputs doivent être envoyés à la validation de ce formulaire.

Ce bouton bleu, durant le clique, doit avoir une animation d’enfoncement lors du clique
Ce bouton bleu, au clique, va générer l’instance de machine virtuelle, puis va renvoyer vers “interface du portail”.

Nous vous invitons à simuler l’information du nom de l’image sélectionnée(en utilisant l’attribut de l’input “value”) lors de l’implémentation de cette interface.