

Évaluation 3

582-B54-RI - INTÉGRATION D'INTERFACE UTILISATEUR



Date de remise : Lundi 12 Février 2024 à 23h55

Travail à faire individuellement

I. Travail à faire

Vous devez concevoir à l'aide de Vue.js un site web sur le sujet de votre choix. Le site web devra permettre de se connecter à l'API en rapport à votre sujet.

Le site devra être fait avec la Composition API.

Il est de votre responsabilité de vous assurer qu'il existe une API qui répond aux besoins spécifiés plus bas en lien avec votre sujet.

La personne utilisant votre site pourra :

- Afficher les éléments
- Voir les détails d'un élément choisi
- Faire une recherche
- Avoir une liste de favoris

Plusieurs API sont disponibles et concernent différents sujets.

Quelques API par exemple :

- Star Wars
- Pokémon
- Harry Potter
- Cuisine (exemple <https://www.themealdb.com/>)
- Rapid API (site contenant une multitude d'API sur différents sujets)

Ne prenez pas d'api payante. Cela n'en vaut pas la peine. Aussi, l'api des pays n'est pas autorisée.

II. Exigences du travail

Votre site doit avoir une barre de navigation permettant d'aller sur chacune des pages principales du site.

1. Page d'accueil

Cette page doit présenter votre sujet. Elle doit avoir au moins deux images en lien avec votre sujet.

La page d'accueil doit également avoir une section d'inscription aux nouvelles de votre site. Il s'agit ici d'un formulaire permettant de rentrer le nom, l'email. En cliquant sur le bouton de validation, vous devez vous assurer que les champs sont correctement remplis et afficher un message d'erreur dans le cas contraire.

Si les informations sont correctement entrées et que l'on clique sur le bouton, la zone contenant les inputs et le boutons doit être cachée, et un message de remerciement doit être affiché.

2. Page de liste

Vous devez avoir une deuxième page permettant d'afficher les données de l'API en lien avec votre sujet. Les informations devront être affichées sous forme de grille.

Cette page doit afficher au moins 10 cartes.

Les cartes doivent comporter les informations importantes et une image.

Exemple : s'il s'agit d'un site de cuisine, cette page doit afficher au moins 10 recettes.

3. Page de zoom

Lorsqu'on clique sur une carte de votre API dans la page liste, on doit voir un « zoom » sur cet objet avec plus de détails. Pensez à un clic dans Netflix où l'on voit ensuite les acteurs, le synopsis et des images. On doit pouvoir revenir à la page de grille facilement.

4. Page de recherche

Vous devez permettre de faire une recherche en lien avec votre sujet en interrogeant votre API.

Si la recherche est concluante, on doit avoir les cartes répondant à la recherche en question.

Si la recherche ne donne pas de résultat, votre page doit le mentionner.

Aussi, on doit pouvoir cliquer sur l'un des résultats de recherche pour aller sur le détail.

Exemple : si nous recherchons dans un site de cuisine le mot « poulet », on doit avoir des recettes à base de poulet. Si on clique sur une des recettes trouvées, on doit avoir le détail de celle-ci.

5. Page des favoris

Vous devez avoir une page de vos favoris. Sur cette page, on doit avoir uniquement les éléments que vous avez marqués comme favoris.

On doit pouvoir marquer un élément comme favori au niveau la carte (page liste ou page de recherche). Un bouton devra donc se trouver sur la carte.

Ces favoris devront être stockés dans un tableau. Ce tableau devra lui-même être déclaré soit au niveau de App.vue, soit dans un store accessible par tout le programme.

Si le tableau de favoris est déclaré dans App.vue, il faudra le « provide » et l'injecter dans les parties où nécessaire. Il faudra en faire de même pour la fonction qui ajoute dans le tableau.

Si le tableau est déclaré dans un store, il faudra également avoir la fonction qui permet d'ajouter dans les favoris.

6. Exigences du travail

Votre travail doit comporter les éléments suivants :

- Des composants pour chacune des pages
- Les cartes doivent provenir d'un composant dédié à cet effet
- Gestion des routes
- Un composable pour les appels vers l'API
 - Les fetchs ne doivent pas se trouver ailleurs
- Provide/inject OU store

III. Critères de correction

- Le site doit comporter tous les éléments demandés plus haut
- Le site devra être esthétique
- Le site n'a pas d'erreur dans la console

IV. Remise du travail

Vous devez remettre votre travail dans un dossier compressé dans les délais indiqués.

Tout travail remis en retard ne sera pas accepté.

Afin d'alléger votre fichier compressé, vous pouvez supprimer le dossier « node_modules » à l'intérieur du fichier compressé. Il sera réinstallé lors de la correction.