



Devoir de Synthèse
en « Développement Framework Côté Client »

Durée : 1 Heure 30 Minutes

- ➔ Barème : Partie I (3 + 3 + 3 + 3 + 2 points) / Partie II (3 + 3 points)
- ➔ Remarque : On peut donner le code complet de la page sans avoir à décomposer les réponses.

Problème :

On se propose de développer une application web, en utilisant le Framework Vue.js, pour gérer les actualités d'une entreprise.

On commence par créer le projet en utilisant la commande « npm create vue@latest ». La structure du projet contient, entre autres, le dossier « views » qui contient les pages. La structure de chaque page est comme suit :

Structure générale d'une page dans le dossier « /views »

```
<script setup>
  import { ref } from 'vue'

  const ... = ref(...)

  function ...() {
    ...
  }
</script>

<template>
  <div>
    <h1>...</h1>
    ...
  </div>
</template>
```

Pour simplifier, on s'intéresse en premier lieu à l'écran d'ajout des actualités (« actualites.vue »). Cette interface permet à l'utilisateur de saisir le titre, le résumé et la nature de l'article. Au dessous, un bouton nommé « ajouter » permet d'ajouter les données saisies à la liste en bas. Chaque élément de la liste contient le titre, suivi de la nature dans la même ligne et dans la ligne suivante, on affiche le résumé.

La page est similaire à cette interface :

Nos actualités

Titre

Résumé

Nature

technique ▼

ajouter

Actualités

Filtrer par nature :

Choisir une ▼

Sortie du nouveau produit [commerciale]

On vient de sortir la nouvelle version de notre logiciel

Correction d'un bug dans v2 [technique]

Ce correctif permet de corriger un problème parvenu dans la version 2

Partie I :

1. Donner le code pour dessiner l'interface y compris l'affichage de la liste.
2. On suppose qu'on a un tableau en mémoire (« Array »), contenant l'ensemble des natures des articles (commerciale, technique et évènement). Donner le code nécessaire pour remplir la liste des natures à afficher.
3. Suite à l'appui sur le bouton « ajouter », on ajoute la nouvelle actualité dans la

liste en bas. Donner le code nécessaire à ajouter. Notons que l'élément ajouté sera affiché à la suite des anciens éléments.

4. Pour chaque élément de la liste, on veut ajouter un bouton « supprimer » (avec ou sans icône). Ce dernier permet d'enlever l'actualité en cours. Donner le code nécessaire à ajouter.
5. On désire rajouter un filtre en haut de la liste. Il permet d'afficher les actualités uniquement selon la nature sélectionnée. Si aucune n'est sélectionnée, on affiche toute la liste. Donner le code nécessaire.

Partie II :

1. On suppose qu'on dispose d'une API REST pour gérer les données dans un serveur. L'API est accessible via cette URL : « `https://api.entreprise.tn/actualites/` ». On veut ajouter un bouton « Synchroniser » qui appelle la méthode « `synchroniser()` » qui permet de supprimer les données dans le serveur et ajouter toutes les actualités qui existent actuellement dans l'interface. Donner le code de cette méthode. Notons que chaque actualité dans le serveur est caractérisée par « id », « titre », « résumé » et « nature ».
2. Si on veut améliorer le filtre par l'affichage, à côté de chaque nature (chaque élément du « SELECT »), du nombre d'éléments dans cette liste : par exemple, « technique - 3 », ... Donner le code nécessaire.

On donne :

Exemple de code pour appeler une API

```
methode() {  
  fetch('https://api.entreprise.tn/actualites/', {  
    method: 'POST',  
    headers: {  
      'Content-Type': 'application/json'  
    },  
    body: JSON.stringify({  
      propriete1: valeur1,  
      propriete2: valeur2,  
      propriete3: valeur3  
    })  
  })  
  .then(response => response.json())  
  .then(data => console.log(data))  
}
```

Bon travail...