

## Dossier 2 : Développement Front End (24 pts)

Soit l'état initial du *store* redux :

```
InitialState = {  
  velo : {  
    id: 14,  
    Matricule: "1 A 4321",  
    Date_derniere_maintenance: "01/06/2025",  
    Date_prochaine_maintenance: "01/08/2025",  
    BatterieUtilisee: {  
      Id : 135,  
      Capacite : 73,  
      Numero_serie : "BAT-202",  
      sante_batterie: 95,  
      nombre_cycles: 2127,  
      statut: "En Utilisation",  
    }  
  },  
  Techniciens : [ {id : ..., Nom : ...}, {id : ..., Nom : ...}, ... ],  
  Batteries : [ {id : ..., Numero_serie : ...}, {id : ..., Numero_serie : ...}, ... ],  
  StatutBatterie : [ "En Stock", "En Utilisation", "Retiré" ]  
};
```

Remarque : Il n'est pas demandé à vous de créer le store Redux (actions, reducer, store).

Filière	DDOWFS	Variante	1	Page 4 sur 7
Examen	Fin de Formation	Session	Juin 2025	

re la composante **DétailsVelo.js** qui lit les informations depuis le store Redux -En utilisant le Selector- et les afficher comme indiqué le *figure 1*. (4pts)

## Velo Details

**Matricule:** 1 A 4321

**Capacité Batterie Actuelle:** 73%

**Santé Batterie Actuelle:** 95%

**Date Dernière Maintenance:** 01-06-2025

**Date Prochaine Maintenance:** 01-08-2025

Figure 1- Détails d'un Vélo (Composante DetailsVelo.js)

2. Ecrire le code de la composante **AjouterChangement.js** (voir figure 2) qui ajoute un changement de la batterie dans la base de données à travers l'API et en utilisant les données du store pour remplir le formulaire d'un nouveau changement de batterie : (6pts)

Méthode http	POST
URL de l'API	http://localhost:8000/api/AjouterChangement
Structure Objet	<pre>{   "id_velo":...,   "id_ancienne_batterie":...,   "id_nouvelle_batterie":...,   "date_changement":...,   "id_technicien":...,   "raison":... }</pre>

### Changement de Batterie

**Velo:** 1 A 4321

**Ancienne Batterie:** BAT-202

**Nouvelle Batterie:**

**Date de Changement:**

**Technicien:**

**Raison**

Ajouter

Figure 2- Ajout d'un changement d'une batterie

- Tous les champs sont obligatoires
- Les valeurs du champs (vélo, ancienne batterie, liste batteries, liste techniciens) sont lues depuis le store redux

Filière	DDOWFS	Variante	1	Page 5 sur 7
Examen	Fin de Formation	Session	Juin 2025	

Créer la composante **ListeBatteries.js** qui lit la liste des batteries depuis le store redux et l'affiche sous la forme de la *figure 3* : (6pts)

## Liste Batteries

Choisir le Statut de la Batterie:

En Stock

Id	Numero Serie	Statut	Nombre Cycles
232	BAT-11	En Stock	6
16	BAT-19	En Stock	24

2 batteries Trouvées

La moyenne des nombres de cycles est 15

*Figure 3 : Liste des Batteries - Lue depuis le store Redux*

- La liste déroulante (select) lit les données depuis le store redux (StatutBatterie).

-Après clique sur le bouton 'Filtrer' Les données sont filtrées selon l'option choisie au niveau de la liste déroulante.

-En Bas du tableau, on affiche le nombre total d'éléments trouvés ainsi que la moyenne des nombres de cycles

4. Ecrire la composante **Menu.js** qui définit les liens qui mènent vers les composantes : (4pts)

Lien	Composante	Titre
/AjouterChgmt	AjouterChangement.js	Nouveau Changement
/ListeBatteries	ListeBatteries.js	Liste Batteries
/DetailsVélo	DetailsVelo.js	Détails Vélo

5. Ecrire le code **App.js** qui : (4pts)

- Définit le routage (BrowserRouter, Routes, ...)
- Appel la composante Menu.js crée ci-dessus.
- Integer le provider du store redux

Filière	DDOWFS	Variante	1	Page 6 sur 7
Examen	Fin de Formation	Session	Juin 2025	