### **Rapport des aspects clés du projet**

Les fichiers sont : Entrainement.js, Jeux.js, Layout.js, ThemeToggle.js, ThemeContext.js, App.js, Jackpot.js et Jackpot.css.

### **Utilisation des Props**

### Il n’y a pas de passage de données entre les composants via des props, car les composants gèrent principalement leur propre état interne sans avoir besoin de communication inter-composants.

### **Utilisation des États**

Plusieurs fichiers utilisent useState pour gérer l'état local, comme dans Jackpot.js, Jeux.js ou Entrainement.js.

### **Gestion des Formulaires**

Il n'y a pas strictement de formulaire, mais nous nous sommes servis de <Form>. De cette façon, un formulaire est implémenté dans le fichier Entrainement.js, avec la gestion des états et des données.

### **Utilisation de React Router Dom v6**

Dans Entrainement.js et App.js, mise en place du routage avec les composants de React Router v6 tels que <Routes>, <Route>, et useNavigate.

### **Utilisation d'une librairie CSS rapide**

**Utilisation de Semantic-UI** pour le style de l'application.

### **Utilisation de l'Outlet**

Utilisation de Outlet dans les fichiers Layout.js et App.js.

### **Variables Calculées**

Utilisation de variables calculées dans Jeux.js : function numQstControl().

### **Utilisation du Hook Navigate**

useHistory a été remplacé par useNavigate dans React Router v6. Utilisation de useNavigate dans Jeux.js et Entrainement.js pour la navigation programmée.

### **Rendu Conditionnel**

Utilisation du rendu conditionnel dans le fichier Jackpot.js pour l’affichage de "JACKPOT!!!".

### **Création et Gestion du Contexte**

Dans ThemeContext.js pour l’affichage en thème sombre ou clair.

### **Pagination et Filtres**

L’application ne génère pas d'informations nécessitant une pagination. Un filtre est utilisé dans Jeux.js avec la constante paysRandom pour sélectionner les pays appartenant à une région.

### **Gestion des Événements**

Association aux entrées de Dropdown, Radio et Buttons dans Jeux.js pour gérer les fonctions handleOptionClick().