

Classe : LGLSI-2	Examen Technologies programmation web	Nombre de pages :2 Documents non autorisés
Enseignants : Mohamed Seifeddine Azzabi, Raoudha Souabni		Date : 05-2023; Durée : 1h30

- Soit le projet web dans la figure ci-dessous (arborescence du projet et code de la page index.html)

EXPLORER

> SOURCE CONTROL

> OPEN EDITORS

> OUTLINE

> TIMELINE

BOX

css

bootstrap.css

js

JS app.js

<> index.html

<> index.html

1 <!DOCTYPE html>

2 <html>

3 <head>

4 <meta charset="UTF-8">

5 <meta name="viewport" content="width=device-width, initial-scale=1.0">

6 <title>Examen, LGLSI2</title>

7 <link rel="stylesheet" href="./css/bootstrap.css">

8 </head>

9 <body>

10 <form id="game">

11 <label for="participant">Nombre de participants</label>

12 <input id="participant" name="participant" type="number" min="1">

13 <label for="type">Type d'activité</label>

14 <select name="type" id="type">

15 <option value="education">Education</option>

16 <option value="social">Social</option>

17 <option value="cooking">Cuisine</option>

18 <option value="relaxation">Relaxation</option>

19 </select>

20 <button id="bOk" type="submit">I'm bored</button>

21 </form>

22 <div class="container"><div id="contenu" class="row"></div></div>

23 <script src="./js/app.js"></script>

24 </body>

25 </html>

Cahier des charges

- Soit l'API "The Bored API", un service qui permet de proposer des activités lorsque quelqu'un s'ennuie, en se basant sur le nombre de participants ou le type d'activité. Les types d'activités possibles sont les suivants : ['education', 'social', 'cooking', 'relaxation'].
 - URL de l'API: <http://www.boredapi.com/api/activity?participants=1&type=education>
 - Exemple de la réponse:


```
{
    "activity": "Learn Morse code",
    "type": "education",
    "participants": 1,
    "price": 0,
    "link": "https://en.wikipedia.org/wiki/Morse_code",
    "key": "3646173",
    "accessibility": 0
  }
```

}

- Nous souhaitons créer un composant Web (BORED) qui respecte les spécifications suivantes :
 - Nom de la classe : BoredActivity
 - Nom du script : bored.js
 - Exemple d'utilisation de la balise :

```
<bored-app
  activite="La description de l'activité"
  prix="Le prix qui se trouve dans price"
  lien="le lien vers le jeu qui se trouve dans link"
  type="le type d'activité">
</bored-app>
```

- La balise doit respecter le design suivant (Bootstrap) :
 - .col-md-6
 - div.Card
 - ✓ div.card-body
 - ✓ h4.card-title → Le prix
 - ✓ h5.card-title → Le type
 - ✓ p.card-text → La description de l'activité
 - ✓ button btn .btn-primary (Valeur="Afficher Lien")
 - Lors du clique sur le bouton "Afficher Lien", une ALERTE sera affichée indiquant la valeur de l'url

Travail demandé

Q1. Écrire le code du script bored.js permettant de :

- Créer sous forme de module le composant web (BORED)

Q2. Écrire le code nécessaire (dans app.js) permettant d'utiliser le module créé et de déclarer une nouvelle balise respectant le cahier des charges.

Q3. Écrire le code nécessaire (dans app.js) permettant :

- Lors de la soumission du formulaire, déclencher une fonction getBored()
- La fonction getBored() :
 - Récupère les valeurs saisies par l'utilisateur
 - Utilise l'API **The Bored API** pour récupérer les informations sur l'activité proposée
 - En utilisant DOM, créer une nouvelle balise **<bored-app>** à partir des informations récupérées
 - En utilisant DOM insérer cette balise dans la DIV ayant l'id **#contenu**

Q4. Ecrire les lignes de code nécessaires pour enregistrer un nouveau service worker (sw.js) pour cette application web

Q5. Où est ce qu'il faut insérer ce code ? Expliquer.

Q6. Identifier les événements qui doivent être gérés par le service worker et expliquer le traitement à faire par le service worker pour chaque événement.

N.B : *il est important décrire un code clair et lisible (sinon la réponse ne sera pas comptabilisée)*

N.B : *il est important de respecter l'ordre des questions dans la réponse (chaque question à part avec son nom Q1, Q2...) (sinon la réponse sera sanctionnée)*