



JavaScript & Node.js

Exercices

1 Révisions

1.1 Fin des exercices non-optionnels

Veuillez terminer tous les exercices non optionnels des fiches précédentes.

2 Exercice optionnel



2.1 Exercice récapitulatif couvrant toutes les compétences du cours

Pour info, cet exercice a été le sujet d'un examen associé à ce cours de JS en 2021. Il doit normalement pouvoir être réalisé dans un délai de 3h.

2.1.1 Contexte

Nous allons développer une simple application permettant d'apprendre à dactylographier.

« La **dactylographie** (du grec ancien « δάκτυλος » (doigt) et « γραφή » (écriture, - graphie)) est l'action de saisir un [texte](#) sur un [clavier](#) (de [machine à écrire](#) ou d'[ordinateur](#)). »

Plus particulièrement, durant ces 3 prochaines heures, nous allons offrir les moyens aux utilisateurs de sélectionner un texte à dactylographier, selon trois niveaux de difficulté.

2.1.2 Question 1 : « *frontend only* »

2.1.2.1 Développement d'une version statique des composants de votre SPA et de la navigation entre ces composants

Dans un premier temps, vous allez développer une version statique des composants visuels de votre SPA et vous assurer que la navigation entre ces composants fonctionne. Par la suite, vous allez donner vie à chacun des composants de votre SPA.

Veuillez prévoir deux boutons qui seront toujours affichés sur votre SPA.

Le bouton « Dactylographier » doit amener à afficher le composant visuel permettant la saisie d'un texte à dactylographier.

Le bouton « Créer un texte » amènera à afficher le composant visuel pour créer un nouveau texte à dactylographier.

Voici un exemple :

Dactylographier

Créer un texte



Exercices

Au chargement de la SPA, tout comme lors d'un clic sur le bouton « Dactylographier », veuillez afficher le composant visuel permettant la saisie d'un texte à dactylographier.

Le composant visuel de saisie d'un texte à dactylographier comprend :

- Une liste déroulante permettant de sélectionner le niveau d'un texte à dactylographier ; la liste déroulante affiche par défaut « Veuillez choisir un niveau » ; Elle permettra de sélectionner parmi 3 niveaux « facile », « moyen », « difficile ».
- Un texte à dactylographier ; pour cette étape, affichez « Le texte à dactylographier sera repris ici une fois le niveau sélectionné. » ;
- Un input pour pouvoir saisir le texte qui sera tapé par l'utilisateur. Par défaut, ce champ de saisie doit être désactivé (attribut **disabled**) et doit être suffisamment grand pour saisir un long texte.

Figure 1: exemple du composant visuel de saisie d'un texte à dactylographier

Le second composant visuel pour créer un nouveau texte à dactylographier comprend :

- un input pour le contenu textuel d'un texte à dactylographier ; ce champ doit être suffisamment grand pour saisir un long texte ;
- une liste déroulante pour indiquer le niveau de difficulté parmi 3 niveaux « facile », « moyen », « difficile » ; la liste déroulante affiche par défaut « Veuillez choisir un niveau » ;
- un bouton « Envoyer »
- un texte qui permettra d'afficher le statut de l'opération associée au formulaire ; à cette étape-ci, affichez simplement « Statut opération ».



JavaScript & Node.js

Exercices

Ajout d'un texte à dactylographier

Veuillez choisir un niveau

Envoyer

Statut opération

Figure 2: exemple du composant visuel de création d'un nouveau texte à dactylographier

Contrainte d'implémentation globale :

- Vous développez une SPA. Pour rappel, on ne charge qu'une seule fois le frontend au niveau d'une SPA.
- Vérifiez que votre navigation, via vos deux boutons, affiche bien l'un ou l'autre composant visuel ;
- Ce que vous développez ne doit pas être joli.
-

2.1.2.2 Chargement dynamique et aléatoire d'un texte à dactylographier

Vous allez rendre dynamique le composant visuel de saisie d'un texte à dactylographier développé au point précédent.

Nous vous offrons 6 textes à dactylographier.

```
const DEFAULT_TEXTS = [  
  {  
    id: 1,  
    content: "voici un texte a dactylographier sans caractere complique",  
    level: "facile",  
  },  
  {  
    id: 2,  
    content: "et voila que nous apprenons tous a taper sur un clavier",  
    level: "facile",  
  },  
  {  
    id: 3,
```



```
    content:
      "Si c'est ça être sage, alors je préfère rester un idiot pour le reste
nt de mes jours !",
    level: "moyen",
  },
  {
    id: 4,
    content:
      "J'étais devenu une relique du passé qu'ils souhaitaient tous voir dis
paraître. Jeune, je me suis demandé pourquoi j'existais.",
    level: "moyen",
  },
  {
    id: 5,
    content:
      "Les gens vivent en s'appuyant sur leurs convictions et leurs connaiss
ances et ils appellent ça la réalité : mais le savoir et la compréhension so
nt des concepts si ambigus que cette réalité ne pourrait être alors qu'une i
llusion.",
    level: "difficile",
  },
  {
    id: 6,
    content:
      "C'est vrai... Dans le monde des ninjas, ceux qui ne respectent pas le
s règles et transgressent les lois... Sont considérés comme des moins-que-
rien. Mais... Ceux qui ne pensent pas à leurs compagnons... Sont encore pire
s.",
    level: "difficile",
  },
];
```

Veuillez créer le fichier **default_texts.js** reprenant ces 6 textes.

Lors de la sélection du niveau de difficulté par l'utilisateur via une liste déroulante :

- Vous devez automatiquement afficher un texte correspondant au niveau de difficulté choisie. Le texte doit être choisi de manière aléatoire parmi les 2 textes associés au niveau de difficulté donné au sein de **default_texts.js**.
- Le champ de saisie doit d'office être activé et être vide. Pensez à vérifier que cela se produit lors d'une deuxième sélection d'un niveau de difficulté.



Texte à dactylographier

Moyen

Si c'est ça être sage, alors je préfère rester un idiot pour le restant de mes jours !

Si d

Figure 3: exemple du composant visuel de saisie d'un texte à dactylographier après sélection du niveau de difficulté, lors de la saisie de texte

Contrainte d'implémentation :

- Veuillez intégrer le fichier **default_text.js** à votre projet pour l'utiliser comme module permettant d'exporter et importer les textes à dactylographier ; vous devrez mettre à jour le contenu de ce fichier pour y parvenir.

2.1.3 Question 2 : RESTFul API et SPA

Vous allez maintenant développer une RESTFul API et une SPA monolithique consommant cette API.

Contrainte d'implémentation : votre SPA doit être accessible localement sur votre machine, via l'URL <http://localhost>

2.1.3.1 RESTFul API

Au niveau de la RESTFul API, tout en respectant les conventions REST appliquées dans le cadre de notre cours, veuillez développer ces deux opérations sur des ressources de type « texte à dactylographier » :

- **Lecture aléatoire** d'une ressource de type « texte à dactylographier » sur base d'un niveau de difficulté ;
 - o Le niveau de difficulté est spécifié lors de l'appel de l'opération de lecture ;
 - o Cette opération renvoie une ressource choisie aléatoirement parmi toutes les ressources associées au niveau de difficulté demandé ;



Exercices

- **Création d'une ressource** de type « texte à dactylographier » comprenant un identifiant, un contenu textuel (le texte qui sera plus tard à dactylographier) et le niveau de difficulté associé au contenu textuel.

Contrainte d'implémentation :

- Les ressources de type « texte à dactylographier » qui sont créées doivent persister, et donc survivre au redémarrage de votre application.

2.1.3.2 SPA et lecture aléatoire d'une ressource

Nous allons maintenant consommer l'opération de lecture aléatoire d'un texte à dactylographier au niveau de la page qui est chargée par défaut lors de l'accès à l'application.

Nous vous recommandons d'intégrer le frontend que vous avez développé dans le cadre de la question 1.

Le formulaire de saisie d'un texte à dactylographier est automatiquement chargé lors de l'accès au serveur, ou lors du clic sur le bouton « Dactylographier ».

Lors de la sélection du niveau de difficulté par l'utilisateur via une liste déroulante, vous devez automatiquement interroger l'API pour obtenir une ressource de type « texte à dactylographier » correspondant au niveau de difficulté choisie.

Veillez afficher le texte à dactylographier renvoyé par votre API au sein de votre page.



JavaScript & Node.js

Exercices



Figure 4 : exemple de lecture aléatoire d'un texte fourni par la RESTful API

2.1.3.3 SPA et création de ressource

Nous allons consommer l'opération de création de votre API au niveau d'un formulaire permettant d'ajouter des ressources de type « texte à dactylographier »

Pour ce faire, nous vous recommandons d'intégrer le frontend que vous avez développé dans le cadre de la question 1.

Le formulaire de création d'un texte est affiché, au niveau de votre SPA, lorsqu'on clique sur le bouton « Création d'un texte ».

Votre formulaire affiche :

- un input pour le contenu textuel d'un texte à dactylographier
- une liste déroulante pour indiquer le niveau de difficulté parmi 3 niveaux

Les données du formulaire ne peuvent pas être envoyées si le contenu textuel est vide ou si la liste déroulante n'indique pas de niveau de difficulté.



Exercices

Lors de l'envoi du formulaire, veuillez :

- afficher au sein d'une div si l'opération de création de la ressource a réussi ou non ; pour ce faire, affichez « Enregistré » en cas de succès, et, en cas d'erreur, affichez le code d'erreur renvoyé par le serveur.

The screenshot shows a web browser window with the address bar set to 'localhost'. Below the address bar is a navigation bar with several categories: Applications, IT, Recipes, Education, Hobbies, and Bank&Insurance&T... Each category is represented by a small yellow square icon. Below the navigation bar are two dark grey buttons: 'Dactylographier' and 'Créer un texte'. The main content area is titled 'Ajout d'un texte à dactylographier'. It features a large text input field containing the placeholder text 'Il était une fois ...'. Below the text field is a dropdown menu currently showing 'Facile'. At the bottom of the form is a blue 'Envoyer' button. Below the button is a light blue rectangular box containing the text 'Enregistré'.

Figure 5 : exemple de création d'une ressource de type « texte à dactylographier »

2.2 Déploiement

2.2.1 Déploiement de myMoovies

S'il vous reste du temps, n'hésitez pas à déployer votre frontend et votre RESTful API myMoovies sur Heroku.



2.3 Apprentissage personnel de technologies JS

S'il y a un concept que vous souhaitez approfondir et que vous êtes à jour avec les exercices des fiches précédentes, n'hésitez pas à approfondir, pour une compétence, un sujet qui vous tient à cœur.

Voici quelques sujets proposés :

- C1 : Frontend interactif et moderne en JS / HTML / CSS
 - Jeu en 2D : Phaser3
 - Animations via une librairie externe
- C2 : Backend & RESTfull API via Node.js & Express
 - Comment gérer correctement les rôles des utilisateurs ?
 - Comment gérer le rafraîchissement de token de la manière la plus sécurisée ?
- C3 : Single-Page Application (SPA)
 - Comment optimiser la gestion d'images ? Qu'afficher lors du chargement d'une page ? Comment faire du lazy loading ?