

## Consignes

- Durée de l'examen: 3h30.
- Travail individuel
- Accès internet autorisé à l'exception des sites de communication et d'intelligence artificielle.
- À la fin de l'examen, montrez à un examinateur que votre site est bien déployé sur votre machine virtuelle.
- Uploader un zip sur Claco qui contient un rapport (partie Linux), votre table SQL (.sql) et votre code.

## Contraintes fonctionnelles

Le but du site est d'aider à la mémorisation de vocabulaire dans une langue étrangère. Il vous donne la possibilité d'entrer votre propre liste de vocabulaire, vous interroge sur un mot aléatoirement et vous donne du feedback.

### Écran principal

L'écran principal interroge l'utilisateur·rice sur un mot **choisi au hasard**. Lorsque vous entrez une réponse, l'interface vous donne du feedback et vous propose un autre mot choisi au hasard. Il n'est pas nécessaire de s'assurer que le mot soit différent.

#### Test yourself

How do you say "hello"?

Your answer:

Submit

#### Test yourself

Unfortunately, "hello" is "مرحبًا".

How do you say "television"?

Your answer:

Submit

L'écran principal doit avoir un lien vers l'écran "vocabulaire".

### Écran vocabulaire

Cet écran liste les mots que vous avez entrés et vous permet d'en ajouter et de les supprimer. Il donne également le **pourcentage de réponses correctes** pour chaque mot.

#### Vocabulary

Word	Translation		Actions
مرحبًا	hello	75%	Delete
التلفاز	television	100%	Delete

#### Add a new word

Word:

Translation:

Submit

## Contraintes non fonctionnelles

- Utilisation d'Express côté serveur
- Utilisation du design pattern MVC
- L'utilisation d'un ORM est autorisée

- Le site doit être déployé sur votre machine virtuelle sur le port 80
- Vous devez mettre en place une sauvegarde des données de la table de la manière suivante:
  - Créer un utilisateur databackup
  - Créer un script qui va sauvegarder les données du site contenues dans votre base de données (pas le code)
  - La table sera sauvegardée dans le dossier /var/backups/db\_web/
  - Créer un fichier db\_backup.log dans le même dossier qui indique quand le script a été lancé
  - Planifier ce script afin qu'il tourne tous les jours à l'heure de votre choix pour l'utilisateur databackup.

## Rapport pour la publication et le script de sauvegarde

Pour la publication et le script de sauvegarde, veuillez réaliser un court rapport dans lequel vous montrerez comment vous avez installé votre projet sur le serveur (commandes utilisées ou application utilisée) et le résultat obtenu (affichage de votre page web dans un navigateur). Vous indiquerez également le contenu de votre script (texte non formaté), sa planification avec l'outil adéquat et les résultats obtenus (ls du dossier /var/backups/db\_ecamair/ ainsi que le contenu du fichier db\_backup.log).

## Évaluation

Le seuil de réussite à 10/20 ne peut être atteint que si tous les éléments suivants sont présents:

- Un site web développé en NodeJS
- Le site est fonctionnel sur votre VM et accessible de l'extérieur à l'aide d'un navigateur sur le port 80

## Grille d'évaluation (partie web)

<b>Persistence des données</b> <ul style="list-style-type: none"> <li>• La structure de la table est appropriée et suffisante pour répondre aux besoins fonctionnels</li> <li>• Les noms des colonnes sont clairs, et la structure est optimale</li> </ul>	1
<b>Contraintes fonctionnelles</b> <ul style="list-style-type: none"> <li>• L'ajout se fait correctement en base de données après soumission du formulaire</li> <li>• La suppression en base de données se fait correctement</li> <li>• Les modifications en base de données sont faites correctement</li> <li>• Les lectures sont correctement implémentées</li> </ul>	4
<b>Design pattern et qualité de code Javascript</b> <ul style="list-style-type: none"> <li>• Il y a une séparation des responsabilités claires, avec un modèle qui s'occupe de la logique en rapport avec les données, et les vues de l'affichage à l'écran. Les requêtes SQL se font du côté du modèle.</li> <li>• Le code est clair, lisible et indenté; les noms des fonctions et variables sont bien choisis.</li> <li>• Le code n'a pas trop de redondance ou de répétitions</li> </ul>	3
<b>Interface utilisateur</b> <ul style="list-style-type: none"> <li>• L'interface est visuellement claire et attrayante.</li> <li>• Le code HTML/CSS est indenté, accessible et sémantique.</li> </ul>	2

Des points bonus seront octroyés si l'application se comporte comme une *Single Page Application*.