

Devoir de contrôle continu

A rendre sur git avant le samedi 17 février 2024 à minuit

Dans ce devoir, vous serez amené à mettre directement en application les compétences acquises en cours et en tp. Les dernières questions nécessiteront quelques recherches dans la documentation de Symfony. Le rendu se fera exclusivement sur le git dans les dépôts créés à cet effet. **A chaque question du devoir, vous créerez une version sous la forme d'une étiquette (tag).** Vous ajouterez à votre dépôt, un fichier `README.md` au format `markdown`. Ce fichier sera mise à jour pour contenir la liste des commandes `Symfony` utilisées pour répondre à la question traitée. Par soucis de portabilité, nous utiliserons comme système de base de données, `SQLite`.

Question 1 *Créer un projet vide Symfony. Mettre en place `webpack` encore et `bootstrap` dans votre projet. Créer votre fichier `README.md` au format `markdown` contenant les noms des membres du groupes. Pensez à ajouter dans `README.md` les commandes `Symfony` utilisées pour répondre à la question. Taguer votre dépôt avec l'étiquette **Question 1**. Pensez à toujours taguer votre dépôt après avoir traité les questions suivantes !*

Nous considérons un système d'information composé uniquement d'une entité `leçon`. Une `leçon` est définie par un nom et une description.

Question 2 *Créer l'entité `leçon`, la base de données et la table associée à cette entité. [Je complète le readme et je tague "question 2"]*

Question 3 *Créer une fixture pour remplir votre base de données en utilisant `Faker`. [Je complète le readme et je tague "question 3"]*

Question 4 *Créer un `CRUD` pour gérer les leçons. [Je complète le readme et je tague "question 4"]*

Question 5 *Ajouter une barre de navigation. Adapter et embellissez votre application avec `Bootstrap`. [Je complète le readme et je tague "question 5"]*

Question 6 *Ajouter la possibilité d'écrire les descriptions au format `markdown`. [Je complète le readme et je tague "question 6"]*

Introduisons le rôle de professeur.

Question 7 *Créer une entité `User` et lui associer un système d'authentification comprenant un login et la création d'un compte. L'entité `User` aura les attributs `nom` et `prénom`. Par défaut, un `User` a le rôle de professeur. [Je complète le readme et je tague "question 7"]*

Question 8 *Nous souhaitons enregistrer pour chaque leçon, le nom de l'unique professeur. Créer cette relation entre l'entité `leçon` et l'entité `User`. Modifier votre fixture en conséquence. [Je complète le readme et je tague "question 8"]*

Question 9 *Faite en sorte que la création d'une leçon ne soit possible que par un professeur connecté. Le professeur connecté devenant ainsi l'auteur d'une leçon. [Je complète le readme et je tague "question 9"]*

Question 10 *Mettez en place les autres contraintes qui vous paraissent pertinentes. [Je complète le readme et je tague "question 10"]*

Question 11 *Embellissez votre application et pensez, si cela n'est pas déjà fait, à modifier votre fixture. Afin de faciliter le test de votre application, créer tous vos users de votre fixture avec le mot de passe "secret". [Je complète le readme et je tague "question 11"]*

Nous souhaitons mettre en place une gestion des inscriptions des élèves aux leçons. Dans un premier temps, nous ne distinguons pas les professeurs et des élèves. Ils sont tous les deux du type User.

Question 12 *Offrir la possibilité pour un élève de s'inscrire (et se désinscrire) facilement à une leçon. [Je complète le readme et je tague "question 12"]*

Question 13 *Offrir la possibilité pour un professeur de voir la liste des élèves inscrits à une leçon et pour un élève de voir ses inscriptions. [Je complète le readme et je tague "question 13"]*

Question 14 *Embellissez votre application et pensez, si cela n'est pas déjà fait, à modifier votre fixture. [Je complète le readme et je tague "question 14"]*

Apportons une gestion plus saine des rôles et introduisons le rôle d'administrateur. Attention ! Un rôle en Symfony est juste une chaîne de caractères commençant par `ROLE_`. Nous aurons donc trois rôles : `ROLE_ADMIN`, `ROLE_PROFESSEUR` et `ROLE_ELEVE`.

Question 15 *Introduire le rôle administrateur pour les professeurs. Un professeur ayant en plus le rôle d'administrateur aura la possibilité d'ajouter et de supprimer des professeurs, ainsi que d'octroyer le rôle d'administrateur à un professeur. [Je complète le readme et je tague "question 15"]*

Question 16 *Revoir votre application pour que le site réservé aux professeurs et celui réservé au public soit clairement séparés en gérant correctement les droits d'accès aux pages. [Je complète le readme et je tague "question 16"]*

Question 17 *Embellissez votre application final et pensez, si cela n'est pas déjà fait, à modifier votre fixture. [Je complète le readme et je tague "question 17"]*