

Devoir de contrôle continu
A rendre sur git avant le 14 février 2021 minuit
(Jour de la Saint Valentin)

Dans ce devoir, vous serez amené à mettre directement en application les compétences acquises dans les tp1-4. Les dernières questions, plus techniques, nécessiteront des 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. Créer votre fichier `README.md` au format `markdown` contenant les noms des membres du groupes. Pensez à ajouter dans `README.md` la commande `Symfony` utilisée 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 **Annonce**. Une annonce est donnée par un nom de produit, une description et un prix.

Question 2 *Créer l'entité **Annonce**, la base de données et la table associée à cette entité.*

Question 3 *Créer une fixture pour remplir votre base de données en utilisant `Faker`.*

Question 4 *Créer les pages pour visualiser l'ensemble des annonces, pour visualiser, pour modifier et pour supprimer une annonce particulière (**sans utiliser CRUD**) :*

Question 5 *Ajouter la possibilité d'écrire les descriptions au format `markdown`.*

Question 6 *Proposer la possibilité d'afficher toutes les annonces par prix croissant ou décroissant.*

Question 7 *Embellissez votre application en utilisation `Material Design`.*

Question 8 *Créer une entité **User** et lui associer un système d'authentification comprenant un login et la création d'un compte. L'entité **User** aura un attribut **nom** comme identifiant.*

Question 9 *Nous souhaitons enregistrer pour chaque annonce, le nom de l'auteur. Créer cette relation entre l'entité **Annonce** et l'entité **User**. Modifier votre fixture en conséquence.*

Question 10 *Faite en sorte que la création d'une annonce ne soit possible que par un utilisateur connecté. L'utilisateur connecté devenant ainsi l'auteur de l'annonce.*

Question 11 Mettez en place les autres contraintes qui vous paraissent pertinentes.

Question 12 Embellissez votre application en utilisation Material Design.

Question 13 Nous souhaitons créer des catégories de produits. Créer une entité **Categorie** avec un attribut **nom** et une relation entre les entités **Categorie** et **Annonce**. Partons du principe qu'un produit est associé à une catégorie. Modifier vos pages de visualisation, de création et de modification en conséquence (A vous de trouver comment, dans un formulaire, traiter la sélection d'une catégorie parmi un ensemble défini dans une table).

Question 14 Offrir la possibilité de visualiser les annonces d'une catégorie.

Question 15 Embellissez votre application en utilisation Material Design.