

| TP noté | Création d'un site utile |

RENDU le 1^{er} avril à 22h59

ORAL - G1 : 01/04 ; G2 | G4 : 04/04

Objectifs du TP

- Approfondir le Framework Symfony
- Créer sa deuxième application en Symfony
- Travailler en équipe de 3 ou 4
- Présenter son travail devant un groupe

Rendu (par email à alexis@onylrocks.com)

- Support Présentation orale dans un dossier du projet
 - o Présentation du contexte (faut vendre votre appli !)
 - o Organisation du groupe et tâches réalisées
 - o Diagramme de classe
 - o Structuration du projet (si intéressant)
 - o Problèmes sur la réalisation / leviers utilisés
 - o Démo
 - o Retour d'expérience (quel bilan « je » et « nous » faisons)
- Lien du Repo Git de la forge de l'Université (avec les accès pour cloner le projet)
- README pour installer le projet (commandes à lancer qui sera suivi pour lancer votre projet), variables d'environnement à changer, etc...
- Attention : pour éviter toute mauvaise surprise, n'hésitez pas à cloner le projet, lancer les commandes du README, tester votre application avant le rendu.

Note

- Les points sont notés par fonctionnalités
- 1 point pour l'effort sur le CSS
- 1 point sur la structure globale, le nommage des variables, etc.
- Des bonus peuvent majorer la note (+ 1 point max) sans dépasser 20 mais surtout montrer votre compréhension du Framework, ses possibilités et votre implication dans ce cours (Voter, EventListener, Validator, Email, etc...)
- La présentation du groupe sera notée en plus de cette note technique

Contexte

Jack Asseur, entrepreneur ambitieux, a eu l'idée de développer une application dédiée à la gestion des listes de courses pour aider les consommateurs à mieux gérer leurs achats en période d'inflation. L'objectif est de créer un site web innovant qui simplifie la gestion des courses quotidiennes, hebdomadaires, et même mensuelles, un problème de plus en plus important pour les ménages. En effet, face à une augmentation constante des prix, de nombreux consommateurs cherchent des solutions pour optimiser leurs achats, éviter les gaspillages et maîtriser leur budget alimentaire.

Afin de concrétiser ce projet, Jack a embauché une équipe d'étudiants en BUT pour réaliser la version V0 de l'application, en tenant compte du fait que l'entreprise est encore en phase de démarrage et n'a pas de budget conséquent. L'équipe devra donc se concentrer sur le développement d'une preuve de concept (POC) fonctionnelle, qui permettra de démontrer le potentiel du projet aux financeurs. L'application devra intégrer diverses fonctionnalités essentielles, tout en restant simple et efficace pour répondre aux besoins des utilisateurs sans complexité excessive. L'objectif est d'attirer l'attention des investisseurs et de faire de cette idée une solution viable et rentable à long terme.

L'équipe devra donc se concentrer sur la création d'un produit avec un impact immédiat, en mettant l'accent sur la simplicité d'utilisation et la pertinence des fonctionnalités. Jack compte sur l'équipe pour développer un prototype qui pourra séduire les financeurs et l'aider à faire évoluer cette application en un produit phare dans le secteur de la gestion de courses.

Fonctionnalités attendues

Lors de l'arrivée de l'utilisateur sur le site <http://liste2courses.local>, l'utilisateur peut s'enregistrer sur l'application ou se connecter. Dans les deux cas l'adresse email et le mot de passe est obligatoire [ndp : 1 point/formulaire]. Si l'utilisateur s'inscrit, il sera redirigé vers un autre formulaire (après sa connexion) pour renseigner son pseudo [ndpⁱ : 1 point].

Dans le deuxième écran, et après une redirection suite au succès de la première page, l'utilisateur pourra *consulter l'ensemble de ses listes de courses existantes avec le détail des 3 premiers articles* [ndp : 2 points] (avec un bouton pour aller voir le détail de ses listes).

Un *bouton d'ajout* d'une nouvelle liste de course est aussi à prévoir sur cette page ainsi que la *suppression* d'une liste de courses [1 point].

Suite au clic sur le bouton « Ajouter une liste » (et avoir renseigné un champ *input* pour donner un nom à la nouvelle liste [ndp : 1 point] sur la même page), nous arrivons sur un troisième écran qui permet *d'ajouter des articles déjà définis* [ndp : 1 point] par un admin avec *une quantité* pour l'article sélectionné. Sur cet écran, l'utilisateur pourra *supprimer un article* ou *marquer cet article comme acheté* [ndp : 2 points].

Enfin, sur le deuxième écran, au-dessus de sa liste de liste de courses, Jack souhaite voir des statistiques relevant au minimum : la *totalité des sommes dépensées*, la *moyenne du coût par article*, *l'article le plus cher et celui moins cher* et la *répartition des dépenses* par type d'article (sucré/salé par exemple ou d'autres si l'admin en ajout) [ndp : 1 point/stat obligatoire].

Pour l'admin (qui a le pseudo *minh@ad.fr*), il aura accès au CRUD pour gérer les articles ainsi que les types d'articles [ndp : 1 point/CRUD + 1 points gestion admin].

Bonus

- Récupération des articles non achetés pour les ajouter dans la liste en cours de création
- Ajouter la notion de « Magasin » qui sera lié aux articles afin de gérer des statistiques par magasin par exemple
- Ajouter des images aux articles
- Permettre à plusieurs utilisateurices de collaborer sur la même liste de courses

ⁱ Ndp : Note du Prof (à ne pas confondre avec *ndlr* qui est la *note de la rédaction*)