

# Projet INFO0303

Version 30/09/2021

Ce projet compte pour 30% pour la partie code et 20% pour la partie rapport. **Les deux évaluations seront réalisées en tenant compte l'une de l'autre.**

## I. Présentation du sujet

### Sujet n°1 : gestion de formulaires.

Un formulaire est un document composé d'informations factuelles, d'informations à remplir (potentiellement par plusieurs personnes) et qui doit être signé par les différentes parties. Pour se donner une idée plus précise, nous pouvons prendre l'exemple de la convention de stage. L'étudiant remplit les informations sur le stage (informations sur l'entreprise, sur lui-même), puis le document doit être signé par l'étudiant, le responsable de la formation, l'entreprise et le Directeur de l'UFR SEN. À tout moment, les personnes impliquées dans le formulaire peuvent voir l'avancée du formulaire.

Nous souhaitons développer une application permettant de créer des formulaires et de les exposer à des personnes spécifiques.

### Sujet n°2 : gestion d'évènements.

Nous souhaitons développer une application permettant de gérer une liste d'évènements (des concours de bridge, des séminaires, des LAN). Au sein de chaque évènement, il est possible d'organiser des activités auxquelles les participants de l'évènement doivent s'inscrire. Cela peut être des séminaires ou des concours (concours de buvette, courses en sac, etc.). Pour ces derniers, il est possible d'avoir un classement où des points seraient attribués. Un comité d'organisation est en charge de chaque évènement. D'un point de vue de l'organisation, il y a un ensemble de tâches à effectuer qui sont attribuées à des organisateurs. Il est possible ensuite d'ajouter des messages associés à chaque tâche, d'afficher des états (terminé, en cours, non commencé), ainsi qu'une date limite.

## II. Travail à réaliser

Le projet doit être réalisé avec *Laravel* (tout projet qui n'utilisera pas *Laravel* ne sera pas évalué) dans le respect du framework. Il doit être réalisé en binôme. L'application doit être déployée et fonctionnelle sur la VM fournie. Le code doit être déposé dans un projet sur le *gitlab* (<https://gitlab-mmi.univ-reims.fr/>). Vous mettrez l'enseignant comme reporter de votre projet. Le code récupéré doit pouvoir être installé sur une machine tierce : les migrations ainsi que les *seeders* doivent être fonctionnels. Votre projet devra également utiliser AJAX.

Le rapport doit contenir toutes les modélisations du projet (diagrammes, MLD, etc.). Il contiendra également les fonctionnalités développées et les choix d'implémentation. Il doit être rédigé et ne pas être une succession de schémas. Ce n'est pas un mode d'emploi : il est déconseillé de faire des captures d'écran de toutes vos pages, sauf nécessité et préférentiellement en annexes. Le rapport doit contenir une partie concise pour expliquer comment utiliser l'application mise en ligne sur la VM (identifiants de connexion, actions permettant de tester les fonctionnalités développées) et une partie indiquant toutes les commandes que vous avez utilisées pour la création du projet (*composer* et *artisan*).

## III. Notation

L'application doit être fonctionnelle. L'ensemble des attendus doit être présent : l'application déployée sur la VM, le code sur le *gitlab* et le code/rapport déposés sur *Moodle*. Le sujet tel que présenté dans la première section est le minimum attendu (16/20 si le code et le rapport sont parfaits). Plus vous implémenterez de fonctionnalités (dans la limite du raisonnable), meilleure sera votre note. Vous devez cependant respecter le cadre du sujet.

## IV. Date

Le projet (code + rapport) doit être rendu pour le 12 décembre 2020 dernier délai. Une zone de dépôt sera mise en place sur *Moodle*. Pour le code, n'oubliez pas de supprimer le répertoire « `vendor` » qui peut être régénéré à l'aide de *composer*.