

**Projet Applications Web 2.0****MIAGEin**

On souhaite mettre en œuvre une application web d'organisation d'événements. Cette application doit permettre de créer des événements (type soirée de remise de diplômes, forums ou autres), d'inscrire des participants et d'en tirer des statistiques.

**Version 1 : BackOffice**

- Le système fonctionne sur la base d'événements qui sont constitués à minima :
- Acronyme de l'événement. Ex « AL3C2019 » ;
- Nom de l'événement. Ex : « Forum AL3C 2019 » ;
- Lieu de l'événement : Adresse postale où l'événement se déroulera ;
- Description de l'événement ;
- Date d'ouverture des inscriptions ;
- Date de clôture des inscriptions ;
- Nombre maximal de participants ;
- Type de participants à choisir. Ex : « Etudiant, Diplômé, Professionnel, Administratif, Enseignant, Accompagnant... »
  - Pour chaque type, on doit associer un nombre d'accompagnants maximal.

Dans cette première version, nous nous focalisons sur le Back Office de l'application qui concerne principalement la création des événements.

Ce Back Office ne doit être accessible qu'à un utilisateur particulier que l'on nommera « administrateur du service ».

Plus techniquement, nous définissons le Back Office comme l'ensemble des pages et API permettant à l'administrateur du service de faire son office.

Lors de la connexion à l'application, l'administrateur du service doit se voir présenter la liste totale des événements. Il peut alors : créer un événement, consulter les détails d'un événement ou en supprimer un.

**Version 2 : FrontOffice**

Maintenant, nous nous intéressons au FrontOffice. Nous définissons le Front Office comme l'ensemble des pages et API permettant à un utilisateur lambda de s'inscrire.

Le système doit présenter la liste des événements dont la date de clôture n'est pas encore passée.

Un utilisateur n'a pas besoin de créer un compte pour s'inscrire mais doit renseigner certaines informations :

- Nom
- Prénom
- @mail
- Numéro de téléphone
- Type d'inscription (fait écho au type de participant de l'événement)
  - Selon le type d'inscription, l'utilisateur doit être invité à renseigner des éléments complémentaires tel que « Promotion » pour les étudiants ou diplômés, nom de l'entreprise pour le type « Professionnel » ou Administratif par exemple.
- Si l'événement le permet, l'utilisateur peut inscrire des accompagnants (dans la limite fixée par l'administrateur du service). Les accompagnants doivent renseigner les mêmes informations que l'utilisateur à l'exception du type d'inscription. Les accompagnants ne doivent pas pouvoir inscrire des accompagnants...

Si toutes les informations sont saisies et qu'il y a suffisamment de place, l'inscription est validée.

Dès lors qu'un utilisateur est inscrit à un événement, il est considéré comme « participant ».

### **Version 3 : Statistiques et Tableaux de bord**

Afin de faciliter la gestion des événements présents et futurs, on souhaite compléter le Back Office avec les éléments suivants :

- Général au Back Office :
  - Statistiques : Nombre d'événements et nombre moyen de participants par événement
- Spécifique à un événement :
  - Statistiques : Nombre de participants, pourcentage de participants par type d'inscription...
  - Participants : Consulter la liste des participants ainsi que les détails d'un participant donné (accompagnant compris s'il y a lieu)
  - Edition de la liste de participants à un événement
  - Edition de badges de participants à un événement

### **Bonus : ...et au-delà**

- a) Comptes de participants. Lors de sa première inscription, un identifiant est donné à l'utilisateur. Il pourra dès lors s'inscrire à un événement en utilisant son id et après affichage des informations connues et confirmation. En cas de mise à jour des informations (comme l'adresse mail par exemple), les informations seront prises en compte dans le Back Office.
- b) Gestion des authentifications.
  - a. On souhaite que les applications frontales (parties web Angular) utilisent une clé (statique et fixée à l'avance par événement) pour pouvoir utiliser l'API du système.
  - b. La sécurité du compte de l'administrateur du service doit être assurée par authentification
- c) Internationalisation de l'application via Angular et envoi de mail de confirmation de participation à un événement. Le participant, dès son inscription, reçoit un mail lui confirmant son inscription.

### **Organisation du projet et travail attendu**

La réalisation du projet comporte deux parties :

#### **Partie I : Conception de l'application**

Concevez et rédigez l'application permettant de répondre aux spécifications précédemment énoncées. Constituez un dossier qui comportera les diagrammes UML (ou avec un autre formalisme que vous connaissez) adéquats exprimant le résultat de l'analyse de cette application.

#### **Partie II : Développement d'une maquette de l'application**

Une maquette de la solution envisagée doit être développée. Celle-ci devra donner une image fidèle de la solution réelle ensuite déployée.

### **Évaluation**

Le projet sera évalué le 26/03/2018 pour les FI et le 28/05/2018 pour les FA. A cette échéance, chaque **trinôme** devra présenter son projet et faire une démonstration. Ce même jour, vous devez rendre le rapport et votre code sous chamilo sous la forme d'un fichier ZIP. Votre projet sera évalué en fonction des choix techniques que vous effectuerez, de la qualité technique du code produit, de la qualité de la documentation, de son respect du sujet, du respect de l'échéance, de l'ergonomie de votre application et de son allure générale.

Tout retard dans la fourniture des livrables sera sanctionné dans l'évaluation.

Consignes diverses :

- Les fonctionnalités supplémentaires que vous mettrez en œuvre seront aussi comptabilisées à la condition que tous les éléments du projet aient été correctement traités au préalable ET sont soumises à l'accord préalable de l'équipe enseignante.

- **RAPPEL DES ÉCHEANCES :**
- **Date de l'évaluation : 26/03/2018 (FI) – 28/05/2018 (FA)**
- **Date de remise des livrables : 2/04/2018 (FI) – 3/06/2018 (FA)**