

Projet

Application de gestion des clubs

Description du projet

L'administration de Efrei a demandé à votre groupe de concevoir et de développer l'application de gestion des clubs étudiants. On vous a donné la tâche de concevoir et implémenter la base de données associée.

Un club est caractérisé par un nom, une date de création, un type (culturel, technique, sportif, entrepreneurial, etc.), et une description. Chaque club a plusieurs membres, qui doivent être des étudiants de Efrei. Chaque membre a un nom, numéro d'étudiant, genre, classe, niveau, date de naissance. Plusieurs membres exercent des responsabilités dans le club (président, trésorier, secrétaire général, vice-président, etc.), une date de début et une date de fin d'affectation au poste (on aimerait garder l'historique de tous les membres qui ont des postes de responsabilité tout au long des années).

Un club peut organiser plusieurs évènements. Un évènement doit avoir un nom, une date de début et de fin, un horaire de début et de fin, un type (formation, workshop, soirée, bootcamp, hackathon, etc.). Certains évènements ont un comité d'organisation, qui sont des membres du club. Plusieurs postes temporaires sont définis pour ce comité (président, vice-président, responsable parrainage, trésorier, etc.), et chaque comité temporaire a une date de création, de dissolution, et doit bien sûr être affecté à l'évènement qu'il gère.

Un évènement doit avoir un budget défini au préalable (un budget peut être NUL pour les évènements gratuits), et le comité responsable de l'évènement doit maintenir le calendrier et l'affectation des rentrées d'argent (provenant du sponsoring) et des dépenses. Chaque dépense doit être associée à une tâche de l'évènement : par exemple, « location de la sono » est une tâche qui a un montant défini. Chaque tâche a un responsable, qui est un membre du comité d'organisation de l'évènement. Un évènement

peut également être commun à plusieurs clubs, le comité d'organisation peut alors être mixte (formé des membres de clubs différents).

Le club a un rapport d'activité qu'il doit générer chaque année. Le rapport doit être rédigé par le président du club et signé par les membres responsables (qui ont des postes) du club. La base de données doit consigner l'état du rapport d'activité (absent, en cours de création, en attente de signature, signé, soumis).

- Le rapport est à l'état *absent* s'il n'a toujours pas été entamé par le président du club. La valeur de l'état est mise à *absent* à la date d'affectation des nouveaux responsables à leurs postes pour l'année en cours.
- Le rapport est à l'état *en cours de création* si le président déclare dans l'application qu'il a commencé la rédaction.
- Le rapport est à l'état en attente de signature s'il est rédigé, et est en train d'être lu et validé par les membres responsables du club. On doit conserver dans ce cas la liste des membres qui ont signé et ceux qui ne l'ont pas encore fait.
- Le rapport est *signé* si tous les membres responsables ont signé.
- Le rapport est à l'état *soumis* s'il a été soumis à l'administration.

On doit conserver la date de modification de chaque état, pour faire plus tard des statistiques de la performance des membres (quel membre a mis le plus de temps à signer le rapport, ou quel président a mis le plus de temps à rédiger un rapport, etc.).

Un club peut gagner des récompenses, qui ont une provenance (l'école, une entreprise, une autre école ou un autre club...). Il peut également être sujet à des sanctions de la part de l'administration.

Travail à faire

1. Création de la base de données

1. À l'aide de votre outil de conception préféré, réaliser le schéma E/R puis générer le schéma relationnel de cette application.
2. Vérifier que votre schéma respecte les trois premières formes normales.
3. Générer le script SQL de la base de données.
4. Créer une nouvelle base de données MySQL sous le nom (clubs_efrei) et injecter le script SQL.

2. Les données de test

Créer un script qui permet d'insérer un jeu de test cohérent et représentatif (des données de test factices). Chaque table doit au moins contenir 10 enregistrements cohérents (ce script doit être inclus dans votre rendu).

Remarque importante pour le reste des questions

Pour chacune des questions suivantes vous devez proposer une requête qui permet de tester la fonctionnalité implémentée. Le test doit être fait avec un jeu de données suffisamment important pour illustrer clairement que votre implémentation est correcte et fonctionne pour toutes les situations (illustrer avec des exemples et des contre-exemples).

3. Les requêtes

Votre base de données doit vous permettre de répondre (entre autres) à ce type de requêtes.

1. Afficher la liste de tous les clubs de Efrei en précisant leur type, président et description. Les informations relatives au président doivent aussi être affichées, notamment son nom, classe et niveau et numéro étudiant.
2. Afficher la liste des évènements qui ont eu lieu l'année passée, avec leur comité d'organisation, budget alloué et dépenses.
3. Afficher l'état des rapports d'activité de chaque club.
4. Pour les rapports en cours de signature, afficher la liste des membres qui n'ont pas encore signé.
5. Afficher les étudiants qui participent à plus d'un seul club, en indiquant le nom du club et leur position au sein de celui-ci (s'ils n'ont pas de responsabilité, la position doit simplement mentionner *membre*).
6. Afficher la liste des dix tâches les plus coûteuses, avec leurs prix, et responsable.
7. Donner les évènements communs à plusieurs clubs, avec la liste de leurs responsables, et leurs affectations.
8. Afficher la liste des récompenses allouées à chaque club, ainsi que leurs provenances.

4. Les vues

Créer les vues suivantes :

1. Créer une vue appelée « top_clubs » qui permet de donner la liste des clubs qui respectent toujours leur budget (dont la dépense est inférieure au budget et aux rentrées d'argent).
2. Créer une vue appelée « top_présidents » qui les présidents de clubs qui rédigent le rapport d'activité dans les délais les plus courts.
3. Créer une vue appelée « aujourd'hui », qui donne la liste des évènements prévus pour aujourd'hui, classés par horaire, et en indiquant le (ou les) clubs responsables.

4. Créer une vue appelée « fainéant », qui donne le nom du membre du comité d'organisation d'un club donné qui signe le rapport en dernier.
5. Créer une vue appelée « trouble_fête », qui donne le nom du club ayant eu le plus de sanctions.
6. Créer une vue appelée « vaut_mieux_acheter » qui donne les tâches de type « location » les plus coûteuses.
7. Créer une vue appelée « teachers_pet » qui donne le nom du club ayant eu le plus de récompenses de la part de l'administration.
8. Créer une vue appelée « perf » qui donne, pour un club donné, la liste de tous ses présidents classés par date, avec pour chacun les évènements qu'il a organisé, l'argent qu'il a pu obtenir du sponsoring, le temps pris pour la rédaction du rapport, et les récompenses et sanctions obtenues pendant son mandat.

5. Bonus

Ce défi rapportera jusqu'à +3 pts sur la note finale aux membres du groupe, avec un plafond de 20/20 pour la note globale du projet.

Seule condition : Toutes les fonctionnalités de base doivent être mises en œuvre.

Créer une petite application dans le langage de votre choix, qui se connecte à la base de données créée, et qui permet d'afficher les résultats des différentes vues créées.

Consignes d'organisation

- Il faudra vous organiser en groupes de 2 membres minimum et 3 membres au maximum par groupe (liste à finaliser auprès de votre enseignant(e) de TP).
- Vous aurez une validation partielle de votre avancement pendant la séance de TP de la **semaine 7-12 Mars**, où vous devez présenter votre MCD.
- La validation finale du projet aura lieu pendant la dernière séance de type Projet, c'est-à-dire pendant la **semaine du 11-16 Avril**. Pendant cette séance, vous aurez à faire une démo du fonctionnement de votre base de données à votre enseignant(e) de TP.

- Le rapport ainsi que les scripts sont à rendre dans l'espace qui vous est créé sur Moodle avant le **dimanche 17 Avril 2022, à 23h59**.
-

Rendus attendus

Les livrables suivants sont à rendre dans un seul fichier zip sur Moodle.

- Les scripts SQL pour la création et remplissage de la base de données.
 - Le MCD et MLD sous forme d'image.
 - Un rapport décrivant votre conception, vos choix, les réponses aux requêtes et toute explication que vous jugez intéressante.
-

Barème

Le barème suivant sera utilisé :

- Conception de la base de données (6 pts)
- Création de la base et Remplissage par des données significatives (4 pts)
- Requetes (4 pts)
- Vues (4 pts)
- Présentation et Rapport (2 pts)
- Bonus (jusqu'à 3 pts)

Note maximale : 20/20