

Projet Gestion De Données 2022

Une école désire développer un logiciel d'aide à la construction de groupes lors de projets.

Avant son utilisation, le logiciel doit être initialisé par un administrateur.

- Il encodera tous les cours dans le système. Pour chaque cours, il encodera son code unique (le code d'un cours commence toujours par 'BINV' suivi de 4 chiffres), son nom, son bloc (1, 2 ou 3) et son nombre de crédits.
- Il encodera également les étudiants dans le système. Pour chaque étudiant, il introduit son nom, son prénom et son email vinci (qui se termine par @student.vinci.be). Il choisira également un mot de passe pour cet étudiant qu'il lui communiquera par mail.
- Finalement, il encodera pour chaque cours les étudiants inscrits. Un étudiant ne peut pas être inscrit plusieurs fois au même cours.

Après cette phase d'initialisation, les projets pourront être créés par les professeurs. Un projet est réalisé dans le cadre d'un cours et possèdera un identifiant (une chaîne de caractère), un nom, une date de début et une date de fin. Pour un projet, le professeur devra ensuite créer des groupes vides en spécifiant le nombre de groupes à créer et le nombre de place dans chaque groupe. Il pourra créer des groupes de tailles différentes (par exemple, 4 groupes de 4 étudiants et 3 groupes de 5 étudiants). Il ne pourra pas créer plus de places dans les groupes que le nombre de participants aux cours. Chaque groupe aura un numéro généré automatiquement. Pour un projet particulier, le premier groupe aura 1 comme numéro, le deuxième aura 2 comme numéro, ... Les numéros de groupes seront toujours compris entre 1 et le nombre de groupes du projet.

Lorsque les groupes vides sont créés, les étudiants inscrits au cours peuvent se rajouter à un groupe ou se retirer d'un groupe. Ils ne peuvent pas se rajouter dans plusieurs groupes pour un même projet. Lorsqu'un groupe est complet (le nombre de membre est égal au nombre de place du groupe), le professeur a la possibilité de valider le groupe ce qui aura comme conséquence de rendre le groupe définitif : il ne sera plus possible aux membres de se retirer du groupe. On retiendra explicitement dans la base de données le nombre actuel de membres au sein du groupe.

Notez que votre système ne doit être utilisé que pour une seule année académique. Pour l'année suivante, on vide simplement la DB.

Le système sera découpé en deux applications java : une pour les étudiants, l'autre pour les administrateurs et professeurs.

Application centrale

Cette application destinée aux administrateurs et professeurs offrira un menu d'accueil permettant d'accéder à dix points :

1. Ajouter un cours.
2. Ajouter un étudiant.
3. Inscrire un étudiant à un cours (en donnant l'adresse mail de l'étudiant et le code unique du cours) . On ne peut inscrire un étudiant à un cours si le cours contient déjà un projet.
4. Créer un projet pour un cours.
5. Créer des groupes pour un projet. Pour cela, le professeur devra encoder l'identifiant du projet, le nombre de groupes à créer et le nombre de places par groupe. S'il veut créer des

groupes de tailles différentes, il devra faire appel plusieurs fois à la fonctionnalité. La création échouera si le nombre de places dans les groupes du projet dépassent le nombre d'inscrits au cours. Dans ce cas, aucun groupe ne sera créé.

6. Visualiser les cours. Pour chaque cours, on affichera son code unique, son nom et les identifiants de ses projets (sur une ligne, séparé par des virgules). Si le cours ne possède pas de projet, il sera noté « pas encore de projet ».
7. Visualiser tous les projets. Pour chaque projet, on affichera son identifiant, son nom, le code unique de son cours, le nombre de groupes, le nombre de groupes complets et le nombre de groupes validés. Si un projet n'a pas encore de groupe, il devra également apparaître.
8. Visualiser toutes les compositions de groupe d'un projet (en donnant l'identifiant de projet). Les résultats seront affichés sur 5 colonnes. On affichera le numéro de groupe, le nom et prénom de l'étudiant, si le groupe est complet, et si le groupe a été validé. Tous les numéros de groupe doivent apparaître même si les groupes correspondants sont vides. Si un groupe est vide, on affichera null pour le nom et prénom de l'étudiant. Les résultats seront triés par numéro de groupe.

Ci-dessous, voici une visualisation pour un projet composé de 3 groupes de 2 étudiants. Le premier groupe n'est pas complet et est composé d'un seul étudiant. Le deuxième est complet et validé. Le troisième groupe est vide.

Numéro	Nom	Prénom	Complet ?	Validé ?
1	damas	christophe	false	false
2	ferneeuw	stéphanie	true	true
2	lehmann	brigitte	true	true
3	null	null	false	false

9. Valider un groupe. Pour cela, le professeur devra encoder l'identifiant du projet et le numéro de groupe. La validation échouera si le groupe n'est pas complet.
10. Valider tous les groupes d'un projet. Si un des groupes n'est pas complet, alors aucun groupe ne sera validé.

Application étudiant

Pour utiliser cette application, l'étudiant devra d'abord se connecter en mentionnant son email et son mot de passe.

Une fois connecté, l'étudiant accèdera à un menu qui proposera six fonctionnalités :

1. Visualiser les cours auxquels il participe. Pour chaque cours, on affichera son code unique, son nom et les identifiants de ses projets (sur une ligne, séparé par des virgules). Si le cours ne possède pas de projet, il sera noté « pas encore de projet ».
2. Se rajouter dans un groupe en donnant l'identifiant de projet et le numéro de groupe. L'inscription échouera si le groupe est déjà complet ou si l'étudiant n'est pas inscrit au cours relatif à ce projet.
3. Se retirer d'un groupe en donnant l'identifiant de projet. Le retrait échouera si le groupe a été validé ou si l'étudiant n'est pas encore dans un groupe.

4. Visualiser tous les projets des cours auxquels il est inscrit. Pour chaque projet, on affichera son identifiant, son nom, l'identifiant du cours et le numéro de groupe dont il fait partie. S'il n'est pas encore dans un groupe, ce dernier champ sera à null.
5. Visualiser tous les projets pour lesquels il n'a pas encore de groupe. Cela n'affichera bien sûr que les projets faisant partie des cours où il participe. Pour chaque projet, on affichera son identifiant, son nom, l'identifiant du cours, sa date de début et sa date de fin.
6. Visualiser toutes les compositions de groupes incomplets d'un projet (en donnant l'identifiant du projet). On affichera le numéro de groupe, le nom et prénom de l'étudiant et le nombre de places restantes dans le groupe. Les résultats seront triés par numéro de groupe.

Ci-dessous, voici une visualisation pour un projet contenant 2 groupes incomplets. Le premier groupe incomplet (groupe numéro 4) est composé de 2 étudiants mais il reste encore une place. Le deuxième (groupe numéro 7) est vide ; deux places sont disponibles dans ce groupe.

Numéro	Nom	Prénom	Nombre de places
4	damas	christophe	1
4	ferneeuw	stéphanie	1
7	null	null	2

Contraintes

1. Utilisez en priorité les capacités de la DB là où elles sont prévues pour être utilisées. Ainsi il est préférable d'utiliser une contrainte d'intégrité plutôt qu'un test dans une procédure stockée. Il est préférable de placer le business au niveau des procédures stockées plutôt qu'au niveau du Java quand cela est possible, etc.
2. La partie Java doit être aussi simple que possible.
 - a. Par besoin d'une IHM, une simple console suffira : `scanner.nextXXX()` et `System.out.println()` .
 - b. Il ne faut, en Java, faire des **SELECT que** sur une seule chose à la fois. Cette chose peut être une table, une vue ou une procédure stockée.
3. Vous travaillez par paire et vous avez donc accès à deux bases de données différentes sur le même serveur (dbu2binXXX et dbu2binYYY). Pour ce projet, vous n'utiliserez qu'une seule de ces deux bases de données, à vous de choisir laquelle. Le login/password de la personne qui possède cette base de données sera utilisée par l'interface administrateur. Le login/password de l'autre personne (celui qui ne possède donc pas la base de données) sera utilisé par l'application étudiant. Il faudra donc donner les droits nécessaires (GRANT) à ce login pour que cette seconde application puisse fonctionner correctement.
4. Attention à la sécurité.
 - a. GRANT adéquats.
 - b. Injection de SQL.
 - c. Stockage du mot de passe adéquat.

Délivrables

1. Pour le vendredi 18 novembre à 23h59 : Remettre sur moodle sous format PDF le DSD correspondant à l'énoncé. À la lecture seule du DSD, les propriétés et associations doivent être compréhensibles. Donnez un nom aux associations : vous pouvez soit le placer sur celles-ci soit le donner via le nom de la clé étrangère. Respectez les conventions du cours de Conception.
2. Pour le samedi 10 décembre à 12h00 (midi) : Remettre sur moodle un fichier archive au format ZIP contenant :
 - a. un fichier projet.sql qui contient l'ensemble des commandes SQL à envoyer au serveur PostgreSQL ;
 - b. la copie de vos projets Java ;
 - c. si nécessaire, un pdf avec une explication des points/choix que vous estimez devoir expliquer dans votre solution ;
3. Du 12 au 16 décembre, une démonstration sera organisée sur les machines de l'institut pour chaque groupe. Des informations sur cette démonstration vous seront communiquées plus tard dans le semestre.

Le plagiat sera lourdement sanctionné (au minimum 0 à l'UE). Tous les projets seront automatiquement testés par un logiciel de détection de plagiat.