

# Bases de données avancées

## Projet : Financement de projets

### Généralités

Le projet est à faire en binômes. Une modélisation préliminaire (avec une brève explication) doit être rendue sur Moodle le 5 avril à 9h00. Elle fera l'objet d'une présoutenance les 8 et 10 avril. L'inscription aux présoutenances se fera sur Moodle et celle-ci est obligatoire. Il s'agit de s'assurer que vous avez commencé votre travail et ne suivrez pas une mauvaise direction. Elle fera partie de la notation. Le projet sera à rendre en fin de semestre, les modalités définitives vous seront communiquées sur Moodle.

### Présentation générale du sujet

Il existe de nombreux sites d'applications offrant des services de développement à des projets originaux, ou personnels, qui ne répondent pas aux critères retenus habituellement par les institutions (banques, subventions, ...).

Ils proposent une organisation assez standardisée, d'appels aux dons, ou de prises de participations, soit auprès d'un public large, soit auprès d'un réseau coopté .

Les intérêts de chacun sont bien compris : le bénéficiaire pourra réaliser son projet, l'intermédiaire en tirera une commission, et le donateur une satisfaction (intérêt, publicité, philanthropie, ...)

En tant qu'intermédiaires, les sites se rémunèrent en touchant une commission directe sur les sommes collectées, ou indirecte sur les sommes dépensées auprès de partenaires. Ils récoltent des informations personnelles qui leur permettent de se construire une expertise sur la sociologie des grands donateurs, et pourront éventuellement proposer, contre tarification, des mises en relations.

Ces idées sont finalement assez anciennes, on les retrouve à l'origine des créations de coopératives, d'associations, de mutuelles, des monts de piétés ... Elles ont connu un renouveau récent par l'utilisation de l'informatique et des réseaux qui offrent un effet de levier important. Alors le vocabulaire a lui aussi été réinventé : micro-finance, financement participatif, cagnotte, et même facebook aujourd'hui propose de superviser du crowdfunding.

**Dans cette veine là, nous vous demandons de faire preuve d'un peu d'imagination dans le choix d'un thème original, puis de mettre en place une modélisation des données qui soit adaptée.**

En plus des cas évoqués précédemment, voici des exemples de thèmes suffisamment généraux qui pourraient se prêter à des développements spécifiques : la gestion de listes de naissance/mariage ; la collecte de fonds pour l'achat d'un cadeau d'anniversaire commun ; la collecte de fonds pour que des musiciens éditent un premier disque en échange d'un intérêt en nature (place de concert, exemplaire dédié ... ) ; des achats partagés d'appartements, de véhicules, de chevaux de course, d'électroménager (avec un partage des usages) ; etc ...

## Modélisation et tables

Le développement de ce projet comporte deux grandes étapes. La première consiste en une modélisation suffisamment riche pour qu'elle soit intéressante, puis viendra sa réalisation, où vous pourrez vous restreindre à une partie raisonnable.

Pour vous donner un ordre de grandeur de ce qui est attendu, le nombre de tables résultantes ne devrait pas dépasser 12, suivant les choix de modélisation et les simplifications entre le modèle et l'implémentation.

La modélisation, sur papier, sera défendue très tôt lors d'une présoutenance. Elle devra impérativement respecter le langage graphique utilisé en cours (et pas un autre). Une section ci-dessous vous en précise les modalités.

Vous devez pour la présoutenance identifier un maximum de contraintes d'intégrité et de règles de gestion au moment de la modélisation et préciser comment chacune pourra être gérée. Les contraintes devront être intégrées ensuite lors de la création des tables, celles-ci devront être munies de triggers le cas échéant. Définissez également les signatures des fonctions psql qui serviront à simplifier les opérations de gestion et les tests.

Les tables devront être peuplées de données pertinentes permettant d'apprécier les résultats des requêtes — volume indicatif : le nombre total de lignes (somme sur toutes les tables) pourra être de l'ordre de 100. Portez une attention particulière aux cas particuliers afin que vos tests soient significatifs.

## Points communs

Quel que soit le thème que vous aurez choisi de développer, vous aurez à aborder les aspects suivant :

### a) Aspects obligatoires

- **Les utilisateurs** : on distinguera probablement d'une façon ou d'une autre les bénéficiaires, les initiateurs, les participants.
- **La mise en place d'un nouveau projet** : faites en sorte qu'elle soit assez facile, et claire. Ajoutez une description, le/les bénéficiaires, les dates, ...
- **Dons et commissions** : précisez les modalités que vous retenez : y a t'il une participation fixe/minimum/maximum ? Comment se finance l'intermédiaire ?
- **La clôture d'un projet** : lorsque l'objectif de collecte sera atteint, le projet rentrera dans une phase de conclusion (résolution, redistribution ...). Mais qu'arrive t'il si à terme le projet n'est pas suffisamment financé ? Il vous faut alors décider de ce que deviennent les fonds : seront-ils versés tout de même, seront-ils remboursés aux donateurs, ou transférés à des projets tiers ? Que devient la part de l'intermédiaire ?
- **L'archivage** : toutes les opérations importantes devront être archivées.

Ces points sont essentiels, et sont donc attendus pour espérer avoir la moyenne au projet.

### b) Aspects complémentaires

Pour un projet plus personnel (et obtenir une note au dessus de la moyenne) vous pouvez traiter des aspects complémentaires en reprenant à votre compte quelques-unes des idées

suivantes :

- **Progressivité** : lorsqu'un projet atteint un seuil, il peut automatiquement s'ouvrir sur un autre plus ambitieux.
- **Communication** : sur des échéances à définir, l'outil peut envoyer un rapport aux donateurs, ou au demandeur.
- **Relance** : à certains moments critiques, on peut imaginer que l'outil se charge de faire des relances à ceux qui n'ont pas répondu.
- même dans le micro-crédit à visée sociale certains principes se sont imposés. Peut-être vous inspireront-ils également :
  - **Restrictions** : personne ne peut obtenir un nouveau prêt s'il n'a pas remboursé le précédent
  - **Solidarité** : si quelqu'un n'est pas en mesure de rembourser son prêt, celui-ci est pris en charge par la communauté
  - **Rentabilité** : globalement votre système doit être rentable pour garantir son existence. Vous pouvez vous poser la question de la rentabilité, en fixer des paramètres, en esquisser une étude.
- **Prospection** : il peut être utile que le système prennent de lui même l'initiative de contacter de nouveaux prospects (en marketing on définit aussi des notions de prospects chauds ou froids ...)
- **Réseau de contacts** : dans le cas d'une recherche de nouveaux donateurs, faut-il désinscrire ceux qui sont inactifs ? Peut-on proposer d'inviter en groupe les amis directs des participants actuels ?

## Présoutenance

D'une durée courte (une dizaine de minutes) elle devra montrer que vous avez réfléchi en amont à tous les aspects du sujet, et que vous vous êtes organisés pour la phase de développement.

- La discussion aura pour support votre diagramme entité/relation qui devra déjà être soigné et quasiment définitif. Vous pourrez le faire sur papier et le scanner pour le déposer. Attention en effet aux outils que vous seriez tentés d'utiliser, ils ne respectent pas tous la convention fixée en cours. Soyez consciencieux !
- Si votre diagramme est très complet, vous aurez à fixer précisément la partie sur laquelle vous allez ensuite vous concentrer. Elle devra être suffisamment riche pour vous permettre de démontrer (lors de la soutenance finale) votre maîtrise des problématiques de cette année.
- Vous devrez avoir établi la liste de toutes les contraintes, et décidé de comment elles seront gérées : (NOT NULL, CHECK, trigger, par construction, ...) ainsi que les grandes thématiques de vos tests.
- Listez toutes les règles de gestion et préciser si celles-ci seront gérées par un trigger ou par une fonction.
- Listez les principales fonctions de gestion que vous mettrez à la disposition de l'utilisateur.
- Ces éléments peuvent tenir sur environ 3 pages, c'est ce petit rapport qu'il vous faudra déposer sur Moodle. Le nom du fichier sera **rapport\_XXX\_YYY.pdf** en précisant vos noms dans le nom du fichier mais aussi sur une page de garde au début

du rapport.

## Soutenance Finale

### Dispositions générales

La soutenance se déroulera par binôme, mais la notation et les questions pourront être individualisées. Vous devrez **chacun** maîtriser l'ensemble de ce qui est présenté, quelle que soit la façon dont vous vous êtes réparti le travail.

Vous pourrez être amené à modifier une partie de ce que vous avez fait pour que nous puissions apprécier votre réactivité.

Nous devrions passer un minimum de temps au clavier, pouvoir enchaîner rapidement les questions, retrouver des lignes de codes très rapidement etc... Cela n'est possible que si vous avez pensé à tous les cas de figure, conservé et ordonné vos propres tests.

Soyez prêt : ayez de quoi faire une démonstration complète, avec des scripts préparés à l'avance qui montrent chaque règle de gestion implémentée.

**Dans le barème nous évalueront cette fluidité (4 ou 5 points). Pensez à des jeux de tests significatifs, améliorez l'interface en écrivant des fonctions utiles, parfois un simple PREPARE suffit.**

Pour simplifier la gestion du temps qui passe, n'utilisez pas `CURRENT_DATE`, mais créez une table qui contiendra une seule date, correspondant à une date courante fictive, associez lui des fonctions pour faire avancer le temps.

Attention : votre code doit pouvoir s'exécuter **sans aucune modification** sur nivose. Veillez à le tester avant de soumettre votre projet.

### Documents

Pour vous aider, voici une liste de ce qu'il faudra rendre sur Moodle. Vous la déposerez sous la forme d'une archive compressée dont le nom sera la concaténation de vos deux noms.

- Un rapport étendant celui de la présoutenance. Il contiendra à présent la description sql de votre schéma dans la base, et des considérations générales sur le développement.
- Une liste commentée des fonctions, des triggers, des règles de gestion ;
- La liste et la nature des index qui seront éventuellement ajoutés ainsi qu'une justification des bénéfices attendus ;
- Un fichier séparé permettant de distinguer clairement l'étape de création des tables de la base appelé `create_All.sql` ;
- Un fichier qui créera les triggers appelé `create_Trigger.sql` ;
- Un fichier `insert_Data.sql` qui permettra de peupler les tables
- Un fichier pour chaque scenario. Chacun permettra de tester comment se comporte votre base de données sur chacune des fonctionnalités, vérifier le déclenchement de chaque trigger, etc. Utilisez des commandes `prompt` et `echo` pour qu'on voie ce qu'il se passe. Chaque fichier s'appellera `test_XXX`. Vous pouvez répartir les tests dans des sous-répertoires.
- Un fichier appelé README qui décrira brièvement le rôle de chacun des autres fichiers et éventuellement les instructions permettant de démarrer.