

École Polytechnique de Montréal  
Département Génie Informatique et Génie Logiciel  
INF3710 – Fichiers et Bases de données

**TP 4 – Applications et Bases de données**

**Objectif:**

**1. Informations générales**

<i>Pondération</i>	<b>20%</b>
<i>Taille de l'équipe</i>	<b>3 personnes</b>

**Notez bien:**

1. Tout retard dans la remise du TP entraîne automatiquement une pénalité comme discuté dans le plan de cours.
2. Aucun TP ne sera corrigé, s'il est soumis par une équipe dont la taille est différente **de trois (3) étudiants** sans l'approbation préalable du chargé de laboratoire. Cette approbation ne sera accordée qu'à une seule équipe et qu'en cas de nombre impair d'étudiants dans le laboratoire. Sinon, la note de zéro sera attribuée aux étudiants concernés.
3. Soumission du TP par **Moodle** uniquement (<https://moodle.polymtl.ca>). Aucune soumission "hors **Moodle**" ne sera corrigée. La note de zéro sera attribuée aux étudiants concernés.

**2. Évaluation**

<b>Rubriques</b>	<b>Points</b>
Clarté et présentation du rapport	10
Réponses aux questions (voir le détail)	90
<i>Total</i>	100

**Critères d'évaluation :**

1. Qualité de la modélisation et de la BD correspondante
2. Qualité du code SQL
3. Qualité du code de l'application
4. Exécution fluide de l'application avec gestion de toutes les exceptions (on ne devrait pas avoir à redémarrer l'application pour cause de bogues). Attention, une pénalité allant jusqu'à 15 points pourrait être appliquée si ce n'est pas le cas.
5. Professionnalisme de l'application et du rapport

### 3. Environnement et outils nécessaires

Vous devez utiliser :

- Le logiciel Visual Studio pour la modélisation UML
- Le SGBD PostgreSQL
- Angular (version 4 et +) et NodeJS pour l'application Web.

### 4. Etude de cas (inspirée de l'université de Lyon 1)

L'arrondissement du Plateau Mont-Royal vous charge de développer une base de données de ses différents jardins communautaires.

Chaque jardin possède un identifiant, un nom et une surface. Il existe trois types de jardins. Le jardin potager (légumes) se décrit avec un type de sol (e.g., argileux). Le jardin verger contient des arbres fruitiers et a une hauteur maximale autorisée. Finalement le jardin d'ornement contient uniquement des fleurs. Bien souvent, un jardin fait partie de plusieurs types.

Un jardin se compose de parcelles, localisées par des coordonnées et des dimensions dans le jardin. Les parcelles limitent l'épuisement des sols. Elles sont découpées en rangs, que l'on numérote à partir de 1 pour chaque parcelle. On stocke également les coordonnées géographiques des rangs.

Le jardin contient différentes plantes. Une plante possède un nom, un nom latin et une catégorie (e.g., racine, fleur), un type (e.g., arbre fruitier) et éventuellement un sous-type (e.g., fruits à noyau). L'association de certaines plantes peut être source de bénéfice (e.g., protection) ou au contraire d'inconvénient (e.g., épuisement du sol) et on désire conserver cette information. Par ailleurs, les plantes subissent des menaces (e.g., attaque de pucerons). Une description textuelle accompagne chaque menace et liste des solutions possibles.

Les plantes sont associées à une seule variété, qui elle-même peut contenir diverses plantes. On désire stocker de nombreuses informations sur les variétés de plantes : nom, année de mise sur le marché, différentes descriptions pour le semis, la plantation, l'entretien, et la récolte, période de mise en place, période de récolte et un commentaire général. Une variété est produite par un semencier, dont on ne stocke que le nom et le site web. Deux semenciers peuvent produire la même variété, et un même semencier peut proposer une même variété en version biologique ou non biologique. Enfin, une variété est plus ou moins adaptée aux différents types de sol, une information cruciale pour les jardiniers.

Certains rangs d'un jardin peuvent rester en jachère pendant une période donnée. Les rangs cultivés sont quant à eux occupés par des variétés de plantes, pour lesquelles on précise le type de mise en place (semis/graines, plant, greffe, etc.). Pendant la période de culture, il est courant de semer plusieurs variétés de graines ensemble, sur un même rang (e.g., carottes, radis et oignon).

## 5. Travail à faire (100 points)

- 1) Lisez attentivement l'étude de cas. Etablissez vos hypothèses si nécessaire. Assurez-vous de bien modéliser ce qui est demandé.
- 2) Proposez un modèle entités-associations (ou entités associations étendu selon le besoin) permettant de répondre aux besoins exprimés ci-dessus et en tenant compte des requêtes auxquelles vous devez pouvoir répondre. N'oubliez aucun composant du modèle. Utilisez le logiciel Visio pour créer le modèle en notation UML. **(5 points - il doit être soumis pour que la base de données relationnelle en SQL et les requêtes soient corrigées)**
- 3) Créez la base de données PostgreSQL correspondante. Indiquez vos clés primaires et étrangères. Notez que votre script doit être fait à la main et non généré avec PGADMIN. Enregistrez votre code SQL dans **bdschema.sql**. **(25 points)**
  - a. N'oubliez aucune contrainte nécessaire dans votre modèle (exemple : intégrité référentielle, valeurs non nulles, etc.).
  - b. En particulier, assurez-vous que tout jardin ne puisse contenir de rang qui reste en jachère pendant plus d'une année
- 4) Entrez des données dans la base de données dans **data.sql**. Notez que votre script doit être fait à la main et non généré avec PGADMIN. Assurez-vous d'avoir des données qui permettent de répondre aux requêtes tel qu'indiqué dans la liste des requêtes à implanter. **(5 points)**
- 5) Créez les requêtes ci-dessous et enregistrez-les dans un fichier **query.sql**. Notez que chaque requête SQL doit être précédée par un commentaire contenant la requête en Français. **(20 points)**
- 6) Créez un trigger dans un fichier *trigger.sql* qui sauvegarde dans une table l'historique les variétés de plantes dans les rangs d'un jardin donné. Ce déclencheur doit s'activer lorsqu'on ajoute une variété dans le rang d'un jardin et on doit sauvegarder aussi la date courante **(5 points)**
- 7) Créez une application Web pour que l'utilisateur puisse directement interroger la base de données (voir la section Application Web). **(30 points)**
- 8) Ecrivez un rapport nommé *matricule1\_matricule2\_matricule3\_TP3\_H22.pdf*. Voir la section *Rapport*. **(10 points)**

### Liste des requêtes à implanter

- 1) (2 points) Lister toutes les plantes qui sont actuellement dans les rangs d'un jardin
- 2) (2 points) Lister le nombre de rangs minimum et maximum sur les parcelles d'un jardin donné (choisissez-en dans vos données)
- 3) (2 points) Lister les détails des parcelles qui ont la variété de plante A et la variété de plante B
- 4) (2 points) Lister les détails des parcelles qui ont la variété de plante A ou la variété de plante B
- 5) (2 points) Lister les détails des parcelles qui ont la variété de plante A mais pas la variété de plante B
- 6) (2 points) Lister tous les rangs d'un jardin donné avec leurs variétés de plantes s'ils sont cultivés. Dans le cas contraire, affichez null.
- 7) (2 points) Quel est le nombre de jardins uniquement avec des semences biologiques ?
- 8) (2 points) Lister tous les jardins qui ont au moins un rang en jachère

- 9) (2 points) Quelles sont les menaces auxquelles sont sensibles les plantes fougères ?  
10) (2 points) Quelles sont les plantes de la variété *tuberosum* ?

## Application Web

Vous devez programmer une application Web en utilisant le stack Node et Angular. Votre application Web doit permettre, au moyen d'une interface, d'insérer, supprimer, modifier et d'interroger les données de votre base de données. En particulier, vous devez permettre :

- 1) **Afficher la liste des jardins** : Vous devez afficher la liste des jardins et lorsqu'un jardin est sélectionné, afficher ses parcelles, ses rangs ainsi que les variétés de plantes actuelles dans les rangs ; (9 points)
- 2) **Ajouter/Modifier/Supprimer une variété de plante** : Vous devez créer une page qui affiche l'ensemble des variétés de plante et qui permet d'ajouter, supprimer ou modifier une nouvelle variété. L'ajout et la modification doivent avoir leur propre page.  
Dans la page d'ajout, on inclut un formulaire avec les informations à saisir pour la variété. Pensez à proposer des valeurs par défaut pour chaque champ de saisie, afin de ne pas avoir à remplir tous les champs du formulaire lors des tests. Quand c'est nécessaire, on utilisera des listes pour sélectionner une valeur (pour le nom de la plante, du semencier, les types de sols, etc.) obtenue de la base de données.  
Dans la page de modification, on s'attend à charger toutes les informations reliées à la variété à partir de la base de données et on permet leur modification ; (15 points)
- 3) **Rechercher les informations d'une plante** en utilisant son nom ou une partie de son nom. Notez que vous pouvez avoir plus d'un résultat pour ces recherches et que vous devez le gérer adéquatement – on doit pouvoir voir tous les résultats correspondants ; (6 points)

## Rapport

Votre rapport doit contenir les informations suivantes :

- La page de garde
- Une brève introduction résumant le projet et les contributions de chaque membre de l'équipe
- Le modèle conceptuel UML incluant les hypothèses et commentaires si nécessaire
- Le modèle relationnel en syntaxe abstraite
- La présentation de l'application développée avec des copies d'écran permettant de démontrer toutes ses fonctionnalités. N'hésitez pas à souligner les aspects novateurs de votre application tel que l'utilisation de certains patrons de conceptions, ou un effort d'ergonomie, etc.
- Un guide d'installation et de configuration qui permette au chargé de laboratoire d'installer et d'exécuter votre application

## Informations supplémentaires

- Notez que les projets doivent être faits en groupes de trois. Les projets ne respectant pas cette condition ne seront pas acceptés et mèneront à une note de 0.
- Il est fortement conseillé de compléter les étapes de la base de données et des requêtes le plus tôt possible. L'application **vous prendra du temps** et vous devrez compter sur votre effort personnel et votre expérience en programmation pour la compléter.

- Notez bien que le chargé de laboratoire doit être capable d'installer et d'exécuter votre application sans problèmes en suivant le guide d'installation et d'utilisation que vous préparerez (voir section Rapport).
- Notez également que votre application doit aller ajouter/modifier/chercher les données via des requêtes SQL appropriées. En aucun cas vous ne devez effectuer des traitements sur les données au niveau de l'application (exemple en utilisant des filtres Javascript).

### **Description des livrables à la fin de la session**

- 1) Le rapport .pdf
- 2) Un modèle relationnel dans un fichier bdschema.sql qui permet de créer votre base de données
- 3) Un fichier data.sql qui ajoute des données à votre BD avec des instructions INSERT (suffisamment pour qu'il y ait au moins deux tuples dans les réponses aux requêtes)
- 4) Un fichier query.sql qui rassemble le code SQL de l'ensemble des requêtes. Notez que chaque requête doit être précédée par un commentaire indiquant le texte de la requête (référez-vous à la liste des requêtes)
- 5) Un fichier trigger.sql
- 6) Le code de votre application Web, soit :
  - a. Le code de votre serveur Node dans un répertoire server
  - b. Le code de votre client Angular dans un répertoire client
  - c. Un ReadMe pour installer et lancer votre application Web

### **Modalités de remise**

Vous devez soumettre sur Moodle à la fin du cours (la date exacte sera spécifiée sur Moodle) un fichier nommé `équipeNum_GRNum_TP4.zip` qui contient tous les livrables demandés.