

Université d'Ottawa

Faculté de génie École de génie électrique et d'informatique

Projet de cours

<i>Cours</i> CSI2532	Bases de données I
<i>Année académique</i>	2023-24
<i>Semestre</i>	Hiver
<i>Instructeur</i>	Kalonji Kalala H
<i>Annoncé</i>	05.02.2024
<i>Soumission du 1er livrable</i>	30.02.2024
<i>Soumission du 2^{ème} livrable</i>	27.03.2024
<i>Présentation du projet</i>	(06-07).04.2024 (ou aux heures des labos : dernière semaine des cours)

Les groupes de 1 à 4 étudiants peuvent soumettre des livrables de projet communs

e-Hôtels

Cinq des chaînes hôtelières les plus connues, avec des hôtels dans plus de 14 endroits différents en Amérique du Nord, ont décidé de collaborer et de développer une application qui permettra à leurs clients de réserver facilement des chambres dans leurs hôtels, en voyant la disponibilité des chambres en temps réel. Vous êtes invités à développer l'application qui permettra ce qui précède.

Pour chaque chaîne hôtelière, nous avons besoin de connaître l'adresse de ses bureaux centraux, le nombre de ses hôtels, les adresses électroniques (E-mails) de contact et les numéros de téléphones. Les hôtels des chaînes hôtelières sont classés (par exemple, 1 étoile à 5 étoiles). Pour chaque hôtel, nous avons besoin de connaître le nombre de chambres, l'adresse de l'hôtel et les numéros de téléphone et de courriel de contact de cet hôtel. Pour les chambres d'un hôtel, nous devons connaître leur prix, toutes les commodités (par exemple, TV, air conditionné, réfrigérateur, etc.), la capacité de la chambre (par exemple, simple, double, etc.), si elles ont une vue sur la mer ou la montagne, si elles peuvent être étendues (par exemple, ajouter un lit supplémentaire) et s'il y a des problèmes / dommages

dans la chambre. Pour les clients, nous devons stocker leur nom complet, leur adresse et leur numéro de sécurité sociale / NAS, la date de leur enregistrement dans notre système. Pour les employés des hôtels, nous devons stocker leur nom complet, leur adresse et leur numéro de sécurité sociale / NAS. Les employés peuvent avoir différents rôles / postes dans un hôtel. Chaque hôtel a besoin d'un gestionnaire. Les clients peuvent rechercher et réserver des chambres via l'application en ligne pour des dates spécifiques. Lorsqu'ils s'enregistrent à l'hôtel, leur réservation de chambre se transforme en location et ils peuvent également payer pour cette location. L'employé qui effectue l'enregistrement pour un client est responsable de la transformation de la réservation de la chambre en location. Un client peut se présenter physiquement dans un hôtel sans réservation et demander directement une chambre. Dans ce cas, l'employé de l'hôtel peut faire la location de la chambre immédiatement sans réservation préalable. Nous devons stocker dans la base de données l'historique des réservations et des locations (archives), mais nous n'avons pas besoin de stocker l'historique des paiements. Les informations sur une ancienne réservation / location de chambre (archivée) doivent exister dans la base de données, même si les informations sur la chambre elle-même n'existent plus dans la base de données. Nous devrions pouvoir supprimer de notre base de données les chaînes hôtelières, les hôtels et les chambres. Nous ne pouvons pas avoir dans la base de données des informations sur une chambre sans avoir dans la base de données les informations sur l'hôtel correspondant (c'est-à-dire l'hôtel auquel la chambre appartient aussi). De la même manière, nous ne pouvons pas avoir dans la base de données des informations sur un hôtel sans avoir dans la base de données les informations sur la chaîne hôtelière correspondante (c'est-à-dire la chaîne hôtelière à laquelle l'hôtel appartient également).

Nous vous demandons de procéder comme suit :

1. **(10%) Créer Diagramme E-R** qui correspond à la description ci-dessus.
2. **(8%) Créez le schéma de base de données relationnelle** qui correspond à votre diagramme ER.
3. **(7%) Définissez les contraintes nécessaires** qui garantiront l'exactitude de la base de données à créer selon votre schéma de base de données électronique. Il s'agit des clés primaires, des contraintes d'intégrité référentielle, des contraintes de domaine et des contraintes définies par l'utilisateur.
4. **(10%) Implémentez la base de données** en fonction de votre schéma de base de données relationnelle et des contraintes que vous avez définies.
5. **(5%) Insérez dans votre base de données les données** de chacune des 5 chaînes hôtelières. Chacun d'eux a au moins 8 hôtels, qui appartiennent à au moins 3 catégories. Deux des hôtels au moins devraient être dans la même zone. Chaque hôtel doit avoir au moins 5 chambres de capacité différente.
6. **(10%) Créer les modifications SQL nécessaires** (utiliser des requêtes et des triggers) : Votre base de données doit permettre d'insérer, de supprimer et de mettre à jour les opérations de données dans votre base de données en fonction des contraintes d'intégrité référentielle que vous avez définies. Donnez le code SQL pour au moins 4 requêtes et 2 déclencheurs (Triggers) de votre choix dans votre rapport.

7. (10%) **Implémentez au moins trois index** sur les relations de votre base de données et justifiez pourquoi vous avez choisi ces index : expliquez quel type de requêtes et de mises à jour de données vous attendez sur votre base de données et en quoi ces index sont utiles pour accélérer l'interrogation de la base de données.
8. (30%) **Application web** : Concevoir et mettre en œuvre une interface utilisateur appropriée, à travers laquelle un utilisateur pourra voir les chambres disponibles en donnant des critères différents, multiples et combinés afin de choisir la chambre qui l'intéresse et de la réserver ou de la louer. Ces critères doivent être : les dates (début, fin) de réservation ou de location, la capacité des chambres, la superficie, la chaîne hôtelière, la catégorie de l'hôtel, le nombre total de chambres dans l'hôtel, le prix des chambres. L'utilisateur doit être en mesure de voir les choix disponibles lorsqu'il modifie la valeur de l'un de ces critères. L'interface utilisateur doit permettre l'insertion/suppression/mise à jour de toutes les informations relatives aux clients, aux employés, aux hôtels et aux chambres. L'utilisateur peut être soit un client (qui utilisera l'interface pour rechercher des chambres et faire des réservations) ou un employé de l'hôtel (qui utilisera l'interface pour transformer une réservation en location lorsqu'un client s'enregistre à l'hôtel, ou faire directement une location lorsqu'un client se présente physiquement à l'hôtel). Un employé devrait être en mesure d'insérer un paiement client pour une location via l'interface. L'interface utilisateur doit être conviviale, ce qui signifie que l'utilisateur n'est pas obligé de connaître SQL. Toutes les informations doivent être présentées à l'utilisateur au moyen de formulaires conçus de manière appropriée. Chaque fois que nécessaire, vous devez utiliser des éléments appropriés, tels que des listes déroulantes, des boutons radio, etc.
9. (10%) L'utilisateur doit pouvoir également voir deux vues spécifiques. Vous devez implémenter les vues en tant que vues SQL. Vue 1: la première vue est le nombre de chambres disponibles par zone. Vue 2: la deuxième vue est la capacité de toutes les chambres d'un hôtel spécifique.

Remarque : Si vous décidez de ne pas implémenter l'Interface Utilisateur comme décrit ci-dessus, mais que vous implémentez directement sur le SGBD les modifications et requêtes décrites, vous recevrez tout de même 40% de la note de cette exigence, soit 12% de la note totale du projet au lieu de 30%.

LIVRABLES

1^{er} livrable

Veuillez soumettre un rapport qui comprend vos réponses aux exigences 1 à 3 de la liste ci-dessus. Ainsi, votre rapport devrait inclure les éléments suivants :

1. Le diagramme ER. Veuillez inclure une brève justification.
2. Schéma de base de données relationnelle. Veuillez inclure une brève justification.
3. Les contraintes que vous avez définies. Veuillez inclure une brève justification pour chaque contrainte.

2^{ème} livrable

Veillez soumettre un fichier .zip qui comprend les éléments suivants :

1. Un rapport qui comprend les éléments suivants :
 - a. Le SGBD et les langages de programmation que vous avez utilisés dans votre implémentation de l'application.
 - b. Étapes spécifiques pour guider quelqu'un dans l'installation de vos applications
 - c. Une liste avec les DDL qui créent votre base de données
2. Votre code SQL qui prend en charge toutes les fonctionnalités de votre application
3. Tout le code nécessaire à l'exécution de votre application

Pour le développement de votre application, nous vous suggérons d'utiliser les éléments suivants : **PostgreSQL/MySQL** pour l'implémentation de la base de données, Apache Tomcat pour la communication client-serveur, PHP ou Java pour le côté serveur de l'application et HTML pour le côté client de l'application. Néanmoins, vous êtes également autorisé à choisir d'autres technologies.

Veillez télécharger vos livrables sur la page du cours aux dates indiquées au début de ce document. Chaque groupe disposera d'un **créneau de 20 minutes** pour présenter son projet **les samedi-dimanche 6 et 7 avril 2024 (ou aux heures des labos de la dernière semaine du cours)**. Veillez venir avec votre application installée et en cours d'exécution sur votre ordinateur portable pour la présentation de votre projet.

Note : Veillez contacter les TA (labos et tutoriel) ou le professeur aux heures de bureau si vous avez des questions.

Bonne application
Kalonji Kalala