

Projet <h2>Gestion de système information</h2>	Reference Version Update Date Project	Esilv\Projets\ADBM 1.0 06/11/2025 MACSIN4A0325
--	--	---

Objectif:

Concevoir, sécuriser, optimiser et automatiser la gestion d'un système d'information (SI) pour une application donnée. Vous suivrez les étapes décrites dans les chapitres ci-dessous, en commençant par la préparation du schéma relationnel, la mise en œuvre des mesures de sécurité, l'optimisation des requêtes, et l'automatisation des tâches. Un chapitre bonus offre l'opportunité d'ajouter une interface graphique pour améliorer l'accessibilité de votre système.

Le projet est à réaliser en groupes de 3 ou 4 étudiants

Fichier à compléter : [Book 1.xlsx](#)

Chapitre 1 : Préparation du schéma relationnel:

1. Recherche de données:

- Trouvez un fichier CSV sur un site de données ouvertes comme [Kaggle](#). Ce fichier doit contenir suffisamment de colonnes pour permettre la création d'un schéma relationnel avec au moins 5 tables.
- Assurez-vous que les données sont pertinentes et exploitables pour le contexte de votre application.

2. Conception du schéma relationnel:

- Créez un schéma relationnel basé sur les données trouvées, en suivant les bonnes pratiques de modélisation de base de données.
- Identifiez les relations entre vos tables (clés primaires, clés étrangères) et définissez les contraintes d'intégrité (unicité, non-nullité, etc.).

3. Chargement des données:

- Importez les données Excel dans votre base de données selon le schéma défini.
- Vérifiez que vos tables sont correctement peuplées et que les relations entre les données sont cohérentes.

Chapitre 2 : Sécurité et gestion des utilisateurs

1. Création des utilisateurs:

- Créez les utilisateurs nécessaires à la gestion de votre système d'information, avec différents profils (administrateur, gestionnaire, utilisateur standard, etc.).

2. Attribution des priviléges:

- Assignez des priviléges appropriés à chaque utilisateur selon leur rôle (lecture seule, modification, suppression, etc.).
- Assurez une gestion stricte des accès pour sécuriser le système.

3. Sécurisation du système:

- Mettez en œuvre des mesures telles que le contrôle d'accès, la gestion des mots de passe, et la limitation de l'accès aux données sensibles.

Chapitre 3: Requêtes et optimisation

1. Conception de requêtes SQL:

- Proposez au moins 6 requêtes SQL complexes pour gérer votre base de données, comprenant :
 - Des jointures entre plusieurs tables.
 - Des fonctions d'agrégation (COUNT(), SUM(), AVG(), etc.).
 - Des sous-requêtes pour extraire des informations précises.
 - Des fonctions analytiques (RANK, POWNUM, ...)

2. Optimisation des requêtes:

- Proposez des techniques d'optimisation pour améliorer les performances des requêtes, telles que la création d'index ou de vues matérialisées.

Chapitre 4 (Bonus): Interface graphique

1. Création d'une interface graphique:

- Proposez une interface graphique pour faciliter l'accès et l'interaction avec votre système d'information. Vous pouvez utiliser le langage de programmation de votre choix (par exemple, C#, Java, C++).

- L'interface doit permettre aux utilisateurs d'effectuer des opérations sur la base de données tout en respectant les priviléges définis.

Livrables:

1. Rapport détaillant tous les chapitres du projet, comprenant :
 - Le lien du fichier CSV utilisé.
 - Le schéma relationnel avec explication des relations et contraintes d'intégrité.
 - Les utilisateurs créés et leurs fonctionnalités.
 - Les requêtes SQL et leur optimisation.
 - Les scripts pour les triggers, fonctions, et procédures.
 - Capture d'écran de l'interface graphique (bonus).
 2. Programme avec interface graphique (bonus) pour gérer le SI (exécutable et code source).
-

Critères d'évaluation:

- Pertinence et qualité du schéma relationnel. (6 points)
- Gestion correcte des utilisateurs et des priviléges. (6 points)
- Complexité et efficacité des requêtes SQL. (6 points)
- (Bonus) Qualité et fonctionnalité de l'interface graphique. (2 points)

Bonne chance!