

# Mini-Projet 2024 | Base de Données XML

## I/ Présentation du projet

Ce projet vise à développer une base de données XML pour une entreprise proposant des "boîtes à recettes". Le but est de gérer efficacement les recettes, ingrédients, commandes, et commentaires des clients, ainsi que les informations sur les livreurs. L'objectif principal est de faciliter la recherche, la sélection et la commande de recettes adaptées à divers besoins des clients, tout en assurant une gestion fluide des commandes et des livraisons.

La décision d'utiliser une base de données XML pour ce projet repose sur la nature dynamique et structurée des données de recettes, où chaque recette peut comporter un nombre variable d'ingrédients et d'étapes. De plus, la flexibilité du XML en fait un choix adapté pour représenter les complexités des données liées à la gastronomie, à la santé et au budget, aspects essentiels pour une entreprise de boîtes à recettes.

## II/ Choix de Modélisation

### Structure XML

La base de données est structurée autour de quatre éléments principaux : recettes, commandes, commentaires et livreurs. Chaque recette est détaillée par son titre, sa description, sa catégorie de cuisine, son profil de santé, son budget, ses ingrédients et ses étapes de préparation. Les commandes associent les clients aux recettes sélectionnées, incluant les détails de livraison et le statut de la commande.

### Schéma XML (XSD)

Le schéma XML (XSD) joue un rôle crucial dans la validation de la structure des données, assurant l'intégrité et la conformité des informations stockées. Il définit les types de données, les éléments requis, et les restrictions pour chaque composant de la base de données, facilitant ainsi le maintien de la qualité des données et la prévention d'erreurs.

### Justifications

Les choix de conception ont été guidés par la nécessité d'une base de données flexible, facilement extensible pour accueillir de nouvelles recettes ou modifier des commandes, et capable de supporter des recherches variées, par exemple par budget, santé ou cuisine. Cette flexibilité est essentielle pour répondre rapidement aux évolutions du marché et aux préférences des clients.

### III/ Avantages et Inconvénients

#### Avantages

- **Structuration claire et hiérarchisée** : La modélisation fait une distinction nette entre les différentes entités (recettes, commandes, commentaires, livreurs), ce qui facilite la navigation dans les données et leur traitement.
- **Utilisation d'identifiants uniques (xs:ID et xs:IDREF)** : Cela permet d'associer de manière fiable les commandes aux recettes et les commentaires aux recettes correspondantes. C'est un choix de conception robuste pour maintenir l'intégrité des données.
- **Détails complets pour chaque recette** : La structure inclut des informations détaillées pour chaque recette, telles que le titre, la description, la cuisine, l'état de santé, le budget, les ingrédients et les étapes de préparation. Cela enrichit la base de données et améliore l'expérience utilisateur.
- **Flexibilité pour ajouter plusieurs ingrédients et étapes** : La modélisation permet d'ajouter un nombre variable d'ingrédients et de recettes grâce à `minOccurs="0"` et `maxOccurs="unbounded"`, offrant une grande flexibilité pour représenter des recettes de complexité variable.
- **Normalisation des données** : L'utilisation des énumérations assurent la cohérence et la fiabilité des données, facilitent la validation et la saisie, tout en simplifiant la logique de traitement et de présentation des informations dans notre système de gestion des recettes et des commandes.

#### Inconvénients

- **Redondance des informations** : L'utilisation répétée de certains éléments, comme les ingrédients dans différentes recettes, peut entraîner une redondance des données. Un système de référencement des ingrédients pourrait être plus efficace mais on a préféré opter pour la flexibilité pour ne pas complexifier le schéma.
- **Gestion des commandes limitée** : La structure actuelle des commandes ne permet pas de spécifier la quantité de chaque recette commandée, se limitant à lister les identifiants des recettes. Cela pourrait être limitatif pour des commandes plus complexes.

## IV/ Scénarios et Implémentation

### Scénarios de Visualisation

#### 1. Visualisation des Recettes par Catégorie de Santé

- **Intérêt du scénario**

Ce scénario permet aux utilisateurs de filtrer rapidement les recettes en fonction de critères de santé spécifiques (par exemple, sans gluten, équilibré, occasionnel), répondant ainsi à des besoins diététiques particuliers.

- **Implémentation**

Une feuille de style XSLT est appliquée pour trier et afficher les recettes selon leur catégorie de santé. Ce choix permet aux utilisateurs de facilement identifier les recettes qui correspondent à leurs préférences ou restrictions alimentaires.

#### 2. Affichage des Commentaires Récents

- **Intérêt du scénario**

Mettre en avant les commentaires récents des utilisateurs permet un aperçu en temps réel de l'appréciation des recettes, encourageant la communauté des consommateurs à partager leurs expériences et conseils. Cette fonctionnalité permet à la marque de fortement augmenter l'engagement de ses clients

- **Implémentation**

Une feuille de style XSLT sélectionne et ordonne les commentaires par date, du plus récent au plus ancien.

#### 3. Liste des Commandes et Leur Statut

- **Intérêt du scénario**

Fournir une vue d'ensemble des commandes et de leur statut (par exemple, "livré", "en préparation") est nécessaire pour la gestion des attentes des clients et l'optimisation du processus de livraison. Cela permet aux clients de suivre l'avancement de leurs commandes et aux gestionnaires de mieux coordonner les livraisons.

- **Implémentation**

À l'aide d'une feuille de style XSLT, les commandes sont affichées de manière intuitive en indiquant leur statut.

### Réorganisation des Données et Transformation en JSON

#### 4. Regroupement des Recettes par Budget

- **Intérêt du scénario**

Ce scénario aide les utilisateurs à planifier leurs repas en fonction de leur budget. En regroupant les recettes par tranche de budget, les utilisateurs peuvent avoir le choix selon leur moyen financier, rendant la cuisine à domicile plus accessible et agréable.

- **Implémentation**

Une transformation XSLT réorganise les données XML pour les grouper par budget. Cela facilite la découverte de recettes correspondant à différents budgets, enrichissant l'expérience de planification des repas.

## **5. Transformation d'une Partie des Données en Format JSON**

- **Intérêt du scénario**

La transformation d'informations spécifiques (par exemple, les détails des livreurs) en JSON permet une intégration facile avec des systèmes externes ou des applications mobiles, permettant d'étendre l'utilisation de cette base XML en dehors de notre site WEB.

- **Implémentation**

Utilisant XSLT, ce scénario convertit une sélection de données XML en JSON, format largement utilisé pour l'échange de données entre serveurs et applications web. Cette flexibilité ouvre la porte à des intégrations diversifiées et à de nouveaux services pour les utilisateurs.

## **6. Programme Java qui exploite les données XML**

- **Intérêt du scénario**

Le programme de plusieurs fonctionnalités telles que l'ajout, la suppression et le filtrage des recettes. L'ajout permet à l'utilisateur d'ajouter de nouvelles recettes à la base de données. La suppression permet de retirer des recettes existantes. Le filtrage offre des options comme le tri par prix, la sélection par préférence diététique et par type de cuisine, permettant à l'utilisateur de trouver rapidement des recettes spécifiques selon ses besoins culinaires.

- **Implémentation**

Pour l'ajout de recettes, nous avons une méthode qui permet à l'utilisateur de saisir les détails de la nouvelle recette, puis de les ajouter au fichier XML existant en utilisant des classes telles que `DocumentBuilderFactory` et `DocumentBuilder`. La suppression des recettes se fait en affichant toutes les recettes disponibles avec un numéro associé à chacune. L'utilisateur choisit ensuite le numéro de la recette à supprimer, et nous utilisons des fonctions de suppression XML pour mettre à jour le fichier. En ce qui concerne le filtrage des recettes par prix, préférence diététique et type de cuisine, nous utilisons des méthodes de parcours du document XML pour extraire les données correspondant aux critères choisis par l'utilisateur, puis nous affichons les résultats pertinents.