

# Projet : Application de Gestion de Restaurant

## Description Générale

Développez une application de gestion de restaurant qui permet de gérer les réservations, les commandes, les factures, les employés et les menus. L'application supportera différents rôles d'utilisateurs (Administrateur, Serveur, Chef de Cuisine) avec des permissions spécifiques. Les données seront stockées dans des fichiers texte pour la persistance. L'application calculera également la TVA et les frais de service sur les factures.

## Structures de Données Utilisées

- **Listes** : Pour stocker des collections d'objets tels que les menus, les réservations, les commandes, etc.
- **Dictionnaires** : Pour associer des clés à des valeurs, par exemple, les détails des utilisateurs avec leurs rôles, les articles du menu avec leurs prix, etc.

```
users = {  
    "administrateurs" : [],  
    "serveurs" : [],  
    "chefs" : []  
}
```

```
menu = []  
reservations = []  
commandes = []  
factures = []
```

## Authentification Basée sur les Rôles

L'application doit gérer différents types d'utilisateurs avec des permissions spécifiques :

### 1. Administrateur

#### a. Permissions :

- Gérer les utilisateurs (ajouter, supprimer, modifier).
- Gérer les menus (ajouter, supprimer, modifier des articles).
- Consulter toutes les réservations et commandes.
- Générer des rapports financiers.

### 2. Serveur

#### a. Permissions :

- i. Prendre des réservations.
- ii. Prendre des commandes.
- iii. Générer des factures.
- iv. Consulter les réservations et commandes assignées.

### 3. Chef de Cuisine

#### a. Permissions :

- i. Consulter les commandes.
- ii. Mettre à jour le statut des commandes (en préparation, prêt).

## Classes Principales et Méthodes

### 1. Utilisateur (Classe Abstraite)

- **Attributs :**
  - `_nom` : Nom de l'utilisateur.
  - `_id_utilisateur` : Identifiant unique.
  - `_mot_de_passe` : Mot de passe pour l'authentification.
- **Méthodes Abstraites :**
  - `afficher_info()` : Affiche les informations de l'utilisateur.
- **Méthodes Concrètes :**
  - `modifier_mot_de_passe(nouveau_mot_de_passe)` : Modifie le mot de passe de l'utilisateur.

### 2. Administrateur (Hérite de Utilisateur)

- **Méthodes :**
  - `gerer_utilisateurs(action, utilisateur)` : Ajoute, supprime ou modifie des utilisateurs.
  - `gerer_menus(action, article_menu)` : Ajoute, supprime ou modifie des articles du menu.
  - `generer_rapports()` : Génère des rapports financiers et opérationnels.

### 3. Serveur (Hérite de Utilisateur)

- **Méthodes :**
  - `prendre_reservation(reservation)` : Enregistre une nouvelle réservation.
  - `prendre_commande(commande)` : Enregistre une nouvelle commande.
  - `generer_facture(commande)` : Crée une facture pour une commande.
  - `consulter_reservations()` : Affiche les réservations assignées.

- `consulter_commandes()` : Affiche les commandes assignées.

#### 4. ChefDeCuisine (Hérite de Utilisateur)

- **Méthodes :**

- `consulter_commandes()` : Affiche les commandes en cours.
- `mettre_a_jour_statut_commande(commande, statut)` : Met à jour le statut d'une commande.

#### 5. Menu

- **Attributs :**

- `_id_article` : Identifiant unique de l'article.
- `_nom` : Nom de l'article.
- `_description` : Description de l'article.
- `_prix` : Prix de l'article.
- `_categorie` : Catégorie de l'article (entrée, plat, dessert, boisson).

- **Méthodes :**

- `afficher_article()` : Affiche les détails de l'article.

#### 6. Reservation

- **Attributs :**

- `_id_reservation` : Identifiant unique de la réservation.
- `_nom_client` : Nom du client.
- `_nombre_personnes` : Nombre de personnes.
- `_date` : Date de la réservation.
- `_heure` : Heure de la réservation.
- `_table` : Numéro de la table réservée.

- **Méthodes :**

- `afficher_reservation()` : Affiche les détails de la réservation.

#### 7. Commande

- **Attributs :**

- `_id_commande` : Identifiant unique de la commande.
- `_reservation` : Objet Reservation associé.
- `_articles` : Liste des objets Menu commandés.
- `_statut` : Statut de la commande (en attente, en préparation, prêt).

- **Méthodes :**

- `ajouter_article(article_menu)` : Ajoute un article au menu de la commande.
- `afficher_commande()` : Affiche les détails de la commande.
- `calculer_total()` : Calcule le total de la commande avec TVA et frais de service.

## 8. Facture

- **Attributs :**

- `_id_facture` : Identifiant unique de la facture.
- `_commande` : Objet Commande associé.
- `_total_ht` : Total hors taxes.
- `_tvA` : Montant de la TVA.
- `_frais_service` : Montant des frais de service.
- `_total_ttc` : Total toutes taxes comprises.

- **Méthodes :**

- `afficher_facture()` : Affiche les détails de la facture.
- `calculer_total()` : Calcule les montants HT, TVA, frais de service et TTC.

## 9. Table

- **Attributs :**

- `_numero_table` : Numéro de la table.
- `_capacite` : Capacité de la table (nombre de personnes).
- `_est_occupee` : État de la table (libre ou occupée).

- **Méthodes :**

- `__init__(numero_table, capacite)` : Initialise la table avec un numéro et une capacité.
- `afficher_info_table()` : Affiche le numéro et la capacité de la table.
- `occuper_table()` : Change l'état de la table en occupée.
- `liberer_table()` : Change l'état de la table en libre.

## 10. GestionFichiers (Classe Utilitaire)

- **Méthodes :**

- `lire_fichier(nom_fichier)` : Lit et retourne le contenu d'un fichier texte.
- `ecrire_fichier(nom_fichier, contenu)` : Écrit du contenu dans un fichier texte.

- `chercher_dans_fichier(nom_fichier, critere)` : Recherche des entrées dans un fichier texte selon un critère.

## 11. Authentification (Classe Utilitaire)

- **Méthodes :**
  - `se_connecter(id_utilisateur, mot_de_passe)` : Authentifie un utilisateur et retourne son rôle.
  - `se_deconnecter()` : Déconnecte l'utilisateur actuel.

## 12. CalculateurFinancier (Classe Utilitaire)

- **Méthodes :**
  - `calculer_tva(montant, taux=0.20)` : Calcule la TVA sur un montant donné.
  - `calculer_frais_service(montant, taux=0.05)` : Calcule les frais de service sur un montant donné.

## Concepts Financiers

- **TVA** : Taxe sur la valeur ajoutée, par exemple, 20%.
- **Frais de Service** : Pourcentage supplémentaire sur le total, par exemple, 5%.

Ces concepts seront intégrés dans le calcul des factures.

## Instructions de Test du Projet

Pour tester ce projet de manière efficace, suivez les étapes ci-dessous :

### 1. Initialisation du Système

- a. Créez une instance de `GestionFichiers` pour gérer les opérations de lecture et d'écriture des fichiers.
- b. Créez une instance de `Authentification` pour gérer les connexions des utilisateurs.
- c. Créez une instance de `CalculateurFinancier` pour les calculs liés à la TVA et aux frais de service.

### 2. Gestion des Utilisateurs

- a. Ajoutez différents utilisateurs avec des rôles variés (Administrateur, Serveur, Chef de Cuisine) en tant qu'Administrateur.

- b. Testez l'authentification en vous connectant avec chaque rôle et en vérifiant les permissions (par exemple, un Serveur ne doit pas pouvoir gérer les utilisateurs).

### **3. Gestion des Menus**

- a. En tant qu'Administrateur, ajoutez, modifiez et supprimez des articles du menu.
- b. Vérifiez que les modifications sont correctement reflétées dans le système et stockées dans les fichiers texte.

### **4. Gestion des Tables et Réservations**

- a. En tant que Serveur, prenez des réservations pour différentes tables et heures.
- b. Vérifiez que les réservations sont enregistrées et affichées correctement.
- c. Testez la gestion des erreurs, par exemple, réserver une table déjà occupée à une heure donnée.

### **5. Prise de Commandes**

- a. En tant que Serveur, prenez des commandes pour les réservations existantes.
- b. Ajoutez des articles du menu à la commande.
- c. Calculez le total de la commande avec TVA et frais de service en utilisant `CalculateurFinancier`.
- d. Générez une facture et vérifiez les calculs.

### **6. Gestion des Factures**

- a. Affichez les factures générées et vérifiez l'exactitude des montants HT, TVA, frais de service et TTC.
- b. Testez la modification des prix des articles du menu et vérifiez l'impact sur les factures futures.

### **7. Gestion des Commandes par le Chef de Cuisine**

- a. En tant que Chef de Cuisine, consultez les commandes en cours.
- b. Mettez à jour le statut des commandes (en préparation, prêt).
- c. Vérifiez que les statuts sont correctement mis à jour et affichés.

### **8. Persistances des Données**

- a. Vérifiez que toutes les données (utilisateurs, menus, réservations, commandes, factures, tables) sont correctement sauvegardées dans les fichiers texte.
- b. Redémarrez l'application et assurez-vous que les données sont chargées correctement depuis les fichiers.

### **9. Gestion des Erreurs**

- a. Testez des scénarios où des actions non autorisées sont tentées, par exemple, un Serveur essayant de modifier les utilisateurs.
- b. Vérifiez que le système gère correctement les entrées invalides et affiche des messages d'erreur appropriés.

