

Démarche de Test des Fonctionnalités de l'Application

Les tests des fonctionnalités de l'application sont essentiels pour garantir son bon fonctionnement et la qualité du code. Voici une démarche de test qui peut être appliquée pour tester les principales fonctionnalités de l'application **INTIA Assurance**.

3.1 Objectifs des Tests

- Vérifier que toutes les fonctionnalités de l'application fonctionnent comme prévu.
- Assurer que les interactions avec la base de données (ajout, modification, suppression) sont correctes.
- Tester l'interface utilisateur pour garantir une expérience utilisateur fluide.
- Vérifier la sécurité et la gestion des erreurs, notamment pour les données sensibles.

3.2 Types de Tests

3.2.1 Test Unitaire

Les tests unitaires permettent de tester des fonctions ou des méthodes spécifiques pour vérifier leur comportement.

- **Test de la création d'un client** : Vérifier si la fonction qui crée un client dans la base de données fonctionne correctement.
- **Test de la suppression d'un client** : Vérifier si la fonction de suppression de client retire bien le client de la base de données.
- **Test de la modification d'un client** : Tester si les informations d'un client peuvent être mises à jour avec succès.
- **Test de la consultation de la liste des clients** : Vérifier si la fonction qui récupère tous les clients retourne les résultats corrects.

3.2.2 Test Fonctionnel

Les tests fonctionnels visent à tester les différentes fonctionnalités de l'application telles que l'ajout, la modification et la suppression de clients, en simulant des actions utilisateur.

- **Test d'ajout d'un client:**
 - Ouvrir le formulaire d'ajout de client.
 - Remplir le formulaire avec des informations valides.
 - Soumettre le formulaire.

- Vérifier que le client a bien été ajouté à la base de données et qu'il apparaît dans la liste des clients.
- **Test de modification d'un client :**
 - Sélectionner un client existant.
 - Modifier les informations (par exemple, changer le numéro de téléphone).
 - Soumettre les modifications.
 - Vérifier que les informations du client sont correctement mises à jour dans la base de données.
- **Test de suppression d'un client :**
 - Sélectionner un client existant dans la liste.
 - Supprimer ce client.
 - Vérifier que le client est supprimé de la base de données et qu'il n'apparaît plus dans la liste.
- **Test de consultation de la liste des clients :**
 - Accéder à la page d'accueil.
 - Vérifier que la liste des clients est affichée correctement.
 - Vérifier que chaque client affiché possède toutes les informations (nom, email, téléphone, etc.).

3.2.3 Test d'Interface Utilisateur (UI)

Les tests d'interface utilisateur permettent de vérifier que les éléments de la page sont bien positionnés et fonctionnent comme prévu. Cela inclut :

- **Vérification de la présentation de la page d'accueil :** Tester si la liste des clients est correctement affichée.
- **Vérification de la présentation du formulaire d'ajout :** Tester si le formulaire d'ajout de client est bien affiché et fonctionnel.
- **Vérification de la validité des boutons et liens :** Tester si tous les boutons et liens fonctionnent comme prévu (par exemple, les boutons "Ajouter", "Modifier", "Supprimer").

3.2.4 Test de Sécurité

La sécurité est cruciale dans une application de gestion des clients, surtout lorsqu'il s'agit de données sensibles. Voici quelques tests de sécurité importants à réaliser :

- **Test de validation des entrées :**
 - Vérifier que des données invalides (par exemple, un email incorrect ou un numéro de téléphone mal formé) ne sont pas acceptées par le formulaire.
 - Vérifier que l'application n'accepte pas les entrées malveillantes (injection SQL, scripts malveillants, etc.).
- **Test de la gestion des erreurs :**
 - Simuler des erreurs comme la soumission de champs vides ou la tentative de suppression d'un client inexistant.
 - Vérifier que l'application retourne des messages d'erreur appropriés sans exposer d'informations sensibles.

3.2.5 Test de Performance

Les tests de performance permettent de vérifier que l'application est capable de gérer un nombre important d'utilisateurs ou de données.

- **Test de charge :** Simuler un grand nombre de requêtes pour voir comment l'application réagit sous une charge importante.
- **Test de performance des requêtes :** Vérifier que les requêtes à la base de données (ajout, modification, suppression) ne prennent pas trop de temps.