
Projet : Développement d'une Application Système Bancaire

Interface graphique – Windows Forms

1. Introduction

Objectif : Développer une application de gestion bancaire comprenant la gestion des clients, des devises, des transactions et des utilisateurs.

Technologie : C# avec Windows Forms

2. Description Fonctionnelle

Gestion des Clients :

- Formulaire AjouterNouveauClient : Permettre l'ajout d'un nouveau client avec les informations essentielles (nom, adresse, numéro de téléphone, etc.).
- Formulaire SupprimerClient : Suppression d'un client existant par identifiant unique.
- Formulaire RechercherClient : Recherche d'un client à partir de critères définis (nom, identifiant).
- Formulaire AfficherTousLesClients : Affichage de la liste complète des clients.
- Formulaire MettreAJourClient : Mise à jour des informations d'un client.

Gestion des Devises :

- Formulaire CalculeurDeDevise : Calcul des conversions entre différentes devises.
- Formulaire RechercherDevise : Recherche d'informations sur une devise.
- Formulaire AfficherToutesLesDevises : Affichage de toutes les devises supportées.
- Formulaire MettreAJourTaux : Mise à jour des taux de change.

Gestion des Transactions :

- Formulaire Dépôt : Permettre le dépôt d'argent sur un compte.
- Formulaire RechercherTransaction : Recherche de transactions spécifiques.
- Formulaire Transfert : Transfert de fonds entre comptes.
- Formulaire Retrait : Retrait de fonds d'un compte.

Gestion des Utilisateurs :

- Formulaire AjouterNouvelUtilisateur : Ajout d'un nouvel utilisateur du système.
- Formulaire SupprimerUtilisateur : Suppression d'un utilisateur existant.
- Formulaire RechercherUtilisateur : Recherche d'un utilisateur à partir de critères définis.
- Formulaire AfficherTousLesUtilisateurs : Affichage de la liste de tous les utilisateurs.
- Formulaire MettreAJourUtilisateur : Mise à jour des informations d'un utilisateur.

Interfaces Principales :

- Formulaire Connexion : Interface de connexion à l'application avec authentification.
- Formulaire ÉcranPrincipal : Écran principal après connexion, offrant des accès aux différentes fonctionnalités.

Contraintes Techniques

- Utiliser le Framework .NET et C# pour le développement.
- Adopter une structure et une convention de nommage cohérente.
- Implémenter des tests unitaires pour chacune des fonctionnalités clés.
- Respecter les principes de programmation orientée objet (encapsulation, héritage).