



**Université Abdelmalek Essaadi  
Ecole Nationale des Sciences Appliquées  
- Al Hoceima -**



**Filière : Génie informatique 2**

## **Rapport du projet : réalisation d'une application desktop java de gestion de crédits bancaire**

**Réalisé par :**

**BOUZID Kaoutar**

**CHACHI Ayoub**

**HASNAOUI Aymane**

**MAKHLOUFI Imane**

**Encadré par :**

**M. BAHRI Abdelkhalek**

**Année universitaire : 2020-2021**

## REMERCIEMENTS :

Avant tous développements sur cette expérience, il apparaît opportun de commencer ce rapport de projet par des remerciements.

Nous souhaitons remercier dans un premier temps Monsieur Bahri Abdelkhalek, notre respectueux professeur du module Java.

Nous tenons à saisir cette occasion pour vous dédier ce remerciement orné de respect et de reconnaissance pour le temps que vous nous avez donné tout au long du semestre afin de nous apprendre les atouts de la programmation en langage Java.

A travers des cours détaillés et bien conçus, des TDs et TP's extensifs guidés par vous tout au long de ces 4 mois nous avons pu acquérir les méthodes et les bases du langage Java avec souplesse et amusement.

Ce serait inévitablement votre encadrement qui a pu être tellement fructueux la raison pour laquelle on a pu réaliser le suivant projet en Java.

# TABLE DES MATIÈRES

<b>REMERCIEMENTS .....</b>	<b>1</b>
<b>TABLE DES MATIÈRES .....</b>	<b>2</b>
<b>INTRODUCTION GÉNÉRALE .....</b>	<b>3</b>
<b>ChAPITRE 1 : LE CREDIT BANCAIRE EN THEORIE.....</b>	<b>4</b>
Qu'est-ce qu'un crédit ?.....	4
Historique du crédit.....	4
Les types de crédits.....	5
La banque.....	5
Le système bancaire.....	5
Gestion d'une banque.....	6
<b>ChAPITRE 2 : CAHIER DES CHARGES ET SPÉCIFICATION DES BESOINS.....</b>	<b>7</b>
Présentation du projet .....	7
Principales fonctionnalités .....	8
Fonctionnalités détaillées .....	8
Spécifications non fonctionnelles.....	9
_Contrainte.....	11
Diagramme de cas d'utilisation .....	12
Diagramme de classes.....	12
Diagramme de séquences .....	13
Conclusion .....	16
<b>ChAPITRE 3 : RÉALISATION DU PROJET .....</b>	<b>17</b>
Les outils utilisés .....	17
Réalisation d'une base de données .....	20
Lancement du projet .....	21
Espace client .....	24
Deposite.....	25
Withdraw.....	25
Money Transfer .....	27
Change Details .....	28
Request a Loan .....	29
Loan Informations.....	30

Opérations .....	31
Sign out .....	31
Conclusion .....	32
<b>CONCLUSION GÉNÉRALE .....</b>	<b>32</b>

## INTRODUCTION GÉNÉRALE :

L'informatisation du système d'information est indispensable pour n'importe quel type d'organisation, ainsi le développement d'une organisation en matière de technologies de l'information exige de nouveaux moyens et supports pour échanger et diffuser l'information dans le but de réduire les contraintes de temps, d'espace et du cout et facilite la gestion de l'organisation.

Le crédit bancaire est l'un des moyens les plus importants du financement de l'économie, il joue un rôle important dans le développement des institutions financières et de l'activité économique.

D'une façon générale, le crédit résulte de la combinaison de trois caractéristiques : Le temps ou le délai pendant lequel le bénéficiaire dispose des fonds prêtés, la confiance faite par le créancier au débiteur, la promesse de restitution des fonds prêtés.

C'est dans ce sens que notre projet consiste au développement d'une application pour informatiser les fonctionnalités basiques de comptes bancaires et principalement la gestion de crédits bancaires.

# CHAPITRE 1 : LE CREDIT BANCAIRE EN THEORIE

## Qu'est-ce qu'un crédit ?

Un **crédit** est une avance de somme d'argent. Vous devez vous assurer de votre capacité de remboursement et prendre en compte la durée de votre engagement. Il existe différents **types de crédit** : La banque cherchera avec vous la formule la plus adaptée à votre situation.

## Historique du crédit :

Un crédit est une créance pour un prêt ou plus généralement une ressource pour l'entreprise. Le sens étymologique de crédit est la confiance accordée à autrui. Il s'agit du participe passé latin de credere, croire.

Le crédit englobe les diverses activités de prêt d'argent, que ce soit sous la forme de contrats de prêts bancaires ou de délais de paiement d'un fournisseur à un client.

Pour satisfaire les demandes de crédit, une Institution bancaire ou financière peut soit utiliser une épargne préalable dont elle dispose ou l'emprunter à son tour sur le marché monétaire, soit créer le montant emprunté par le mécanisme de création monétaire.

Même si elle est plus ancienne que le secteur bancaire, l'activité de crédit est l'une des activités bancaires les plus importantes. Nous supposons qu'il trouve ses origines à l'époque babylonienne et qu'il pourrait survivre jusqu'à présent à de nombreux obstacles qui étaient principalement liés

à la religion - comme le concept d'usure - pendant les civilisations grecque, romaine et arabe. Les différents produits de crédit qui existent de nos jours ont été engendrés par les mutations et les évolutions des besoins, de la technologie et de la structure économique de la société moderne.

## **Les types de crédit**

On peut distinguer le crédit à la consommation et le crédit immobilier par :

- La durée : en règle générale, les crédits à la consommation sont plutôt des crédits à court terme. Les crédits immobiliers sont plutôt des crédits à long terme (10 à 15 ans ou même plus), même si on trouve aussi dans cette catégorie des crédits relais qui peuvent durer moins de 2 ans.
- L'objet du crédit : un crédit à la consommation permet de financer les dépenses de la vie courante et d'équipement, au sens large puisqu'entrent dans cette catégorie les voitures et les bateaux. Un crédit immobilier permet de financer l'acquisition d'un terrain ou d'un logement ou bien encore des travaux de rénovation ou d'aménagement.

## **La banque**

### **Le système bancaire**

Le système bancaire peut être défini comme étant un ensemble d'institutions et de mécanismes permettant le fonctionnement des banques. Ces banques créent la monnaie en consentant des crédits. Elles procurent donc à l'économie des instruments de paiement, en

mettant à la disposition des clients leur propre monnaie ou la monnaie émise par la banque d'émission.

#### Gestion d'une banque :

La banque doit pouvoir atteindre plusieurs objectifs dans le cadre de la gestion quotidienne des comptes bancaires. Ces objectifs sont bien sûr complémentaires. Il lui faut :

Maîtriser les risques associés aux opérations bancaires réalisées sur le compte.

Améliorer la qualité des services rendus aux clients.

Rentabiliser ses comptes en équipant la clientèle des produits et services de la gamme.

Tout en gardant de côté que le domaine bancaire est tellement vaste, sujet de recherches et de mémoires, on ne pourrait pas le limiter dans le présent rapport.

# CHAPITRE 2 : CAHIER DES CHARGES ET SPÉCIFICATION DES BESOINS

## Présentation du projet :

- **Objectif**

L'objectif principal de ce projet est de développer une application desktop qui permet d'offrir au client de la banque un moyen efficace pour profiter des services bancaires et prendre des crédits. Pour aboutir à cette fin, nous allons tout d'abord effectuer une étude conceptuelle de l'application. Cette dernière nous permettra, en effet, d'accéder facilement à la réalisation de l'application en organisant les idées et en structurant le processus de codage suivant des diagrammes.

- **Cible**

Client de la banque.

- **Spécification des besoins fonctionnels**

L'application que nous allons développer devra regrouper toutes les fonctionnalités nécessaires pour :

- ✓ S'inscrire.
- ✓ S'authentifier.
- ✓ Déposer, débiter ou virer un montant.
- ✓ Afficher les informations de son compte.



- ✓ Demander et consulter ses crédits.

## Principales fonctionnalités

- Gestion des opérations bancaires.
- Virement d'un compte à un autre compte.
- Gestion des crédits.

## Fonctionnalités détaillées

Action	Acteur	Remarque
<i>Gestion des clients</i>		
S'inscrire : <ul style="list-style-type: none"> <li>• Nom</li> <li>• Prénom</li> <li>• Date de naissance</li> <li>• Email</li> <li>• Username</li> <li>• Password</li> <li>• Téléphone</li> <li>genre</li> </ul>	Client	
Se connecter : <ul style="list-style-type: none"> <li>• Username/Email</li> <li>• Password</li> </ul>	Client	
Se déconnecter	Client	
Consulter un compte	Client	
Supprimer un compte	Client	balance <= 1 MAD et pas de crédit courant
<i>Gestion opérations bancaires</i>		
Déposer un montant	Client	
Débiter un compte : Montant	Client	montant <= balance

Effectuer un transfert : <ul style="list-style-type: none"> <li>• Id de client récepteur</li> <li>• montant</li> </ul>	Client	montant <= balance
Consulter l'historique des opérations	Client	
<i>Gestion des crédits</i>		
Ajouter un crédit : <ul style="list-style-type: none"> <li>• Emprunt</li> <li>• Durée</li> </ul>	Client	Valeur acquise = emprunt + intérêts
Payer une mensualité	Client	Mensualité = valeur acquise / durée
Consulter un crédit	Client	

## Spécifications non fonctionnelles

### 1.Efficacité du rendement :

Le rendement du logiciel doit comporter comme le rapport entre les étudiants et le service d'apogée et qui permet de jouer le canal de transmission des documents entre deux côtés séparés.

Il s'agit d'optimiser le temps et la matière au sein du service apogée

Et aussi l'étudiant va recevoir le document sans avoir à se déplacer à l'école.

### 2.Compatibilité :

- Coexistence : La mesure dans laquelle le Logiciel peut coexister avec d'autres Logiciels indépendants dans un environnement commun partageant les ressources communes sans effets nuisibles.
- Interopérabilité : La mesure dans laquelle deux ou plusieurs systèmes ou composants peuvent échanger des informations et utiliser les informations qui ont été échangées.

### 3.Facilité d'utilisation :

- Pertinence : les utilisateurs peuvent reconnaître si le logiciel est adapté à leurs besoins.

- Facilité d'apprentissage : le logiciel peut être utilisé par les utilisateurs spécifiés pour atteindre les objectifs spécifiques avec efficacité, efficacité, sécurité et de satisfaction dans un contexte d'utilisation spécifiée.
- Opérabilité : le logiciel a des caractéristiques qui le rendent facile à utiliser et contrôler.
- Protection de l'utilisateur à faire une erreur : La mesure dans laquelle le système protège les utilisateurs à faire des erreurs.
- Interface utilisateur esthétique : La mesure dans laquelle l'interface utilisateur permet de plaire et de satisfaire l'interaction de l'utilisateur.
- Accessibilité : Le degré d'efficacité, d'efficacité, de sécurité et de satisfaction quand les gens possédant le plus vaste éventail de capacités utilisent un produit.

#### 4. Fiabilité :

- Maturité : Le degré auquel un système répond aux besoins en matière de fiabilité dans des conditions normales d'opération.
- Disponibilité : Le logiciel est opérationnel et accessible lorsqu'il doit être utilisé.
- Tolérance aux pannes : Le logiciel fonctionne comme prévu, malgré la présence de défauts matériels ou de défauts logiciels.
- Rétablissement : La mesure dans laquelle le produit peut récupérer les données affectées et ré-établir l'état désiré si une interruption ou une panne se produit.

#### 5. Sécurité :

- Confidentialité : Toutes les informations et des données sont protégées contre la divulgation non autorisée d'informations ou de données qu'elle soit accidentelle ou délibérée.
- Intégrité : Le logiciel empêche l'accès non autorisé ou la modification de programmes d'ordinateur ou de données.

- Authenticité : Les informations de l'étudiant doivent être correctes avant de passer à l'étape de la demande.

## 6. Maintenableté :

- Modularité : Le logiciel d'ordinateur est composé de composantes discrètes de sorte qu'un changement à l'une des composantes a un impact minime sur les autres composantes.
- Réutilisabilité : Les modules peuvent être utilisés dans plus d'un système ou dans la construction d'autres modules.
- Analysabilité : La facilité avec laquelle l'impact du projet de modification sur le reste du logiciel peut être évalué, ou le produit peut être diagnostiqué pour des déficiences ou pour les causes des échecs ou les parties qui doivent être modifiées peuvent être identifiées.

## 7. Portabilité :

- Installable: La facilité avec laquelle le logiciel peut être installé avec succès et/ou désinstallé dans un environnement spécifié.
- Remplaçabilité : Le logiciel peut être utilisé à la place d'un autre produit logiciel spécifié pour la même intention dans le même environnement.

## Contrainte

- Les logiciels doivent satisfaire aux contraintes de temps d'exécution

## Solutions :

- Logiciels plus simples
- Veiller à la complexité des algorithmes
- Machines plus performantes
- 

1. Un même logiciel doit pouvoir fonctionner sur plusieurs machines

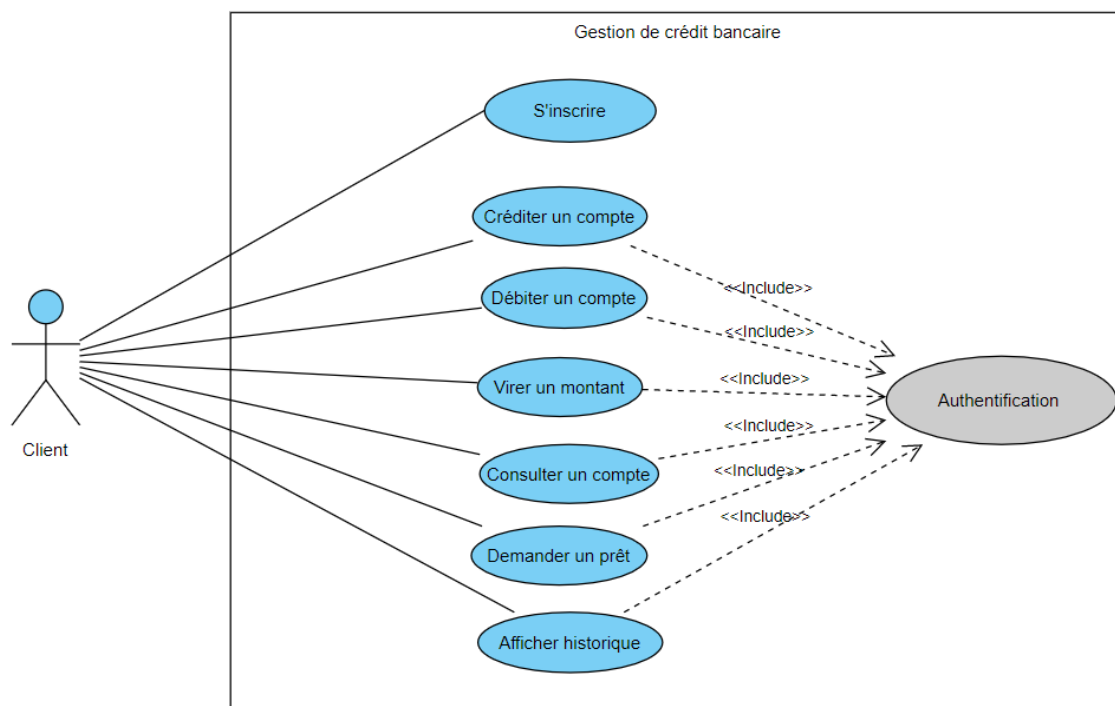
### Solutions :

- Rendre le logiciel indépendant de son environnement d'exécution (voir interopérabilité)
  - Machines virtuelles
2. Adéquation entre Le besoin effectif de l'utilisateur Les fonctions offertes par le logiciel

### Solutions :

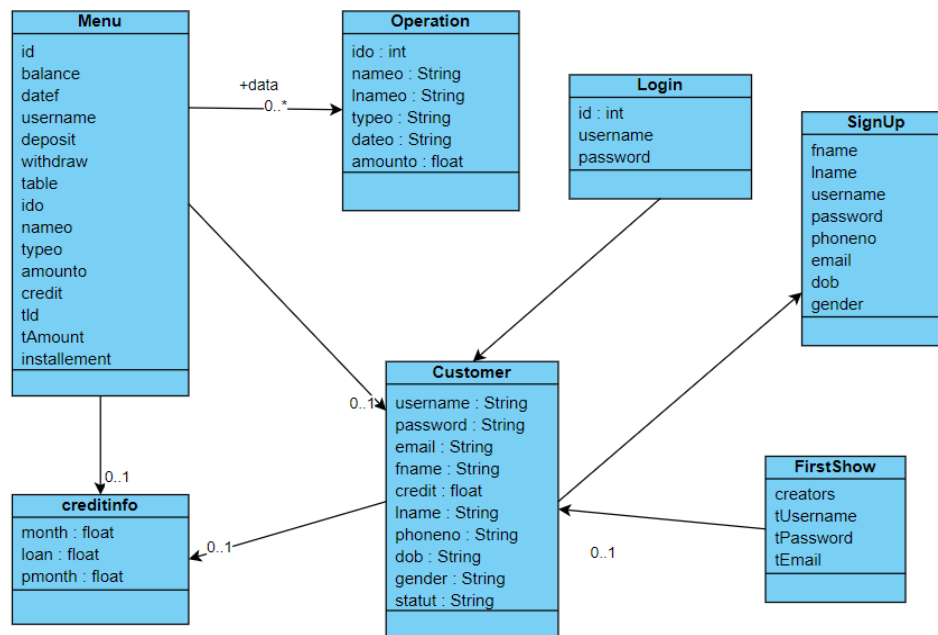
- Emphase sur l'analyse des besoins
- Améliorer la communication (langage commun, démarche participative)
- Travailler avec rigueur

### Diagramme de cas d'utilisation :



L'acteur de notre système est **le client**.

## Diagramme de classes:



## Diagramme de séquences :

Le diagramme de séquence est une représentation intuitive qu'on a utilisée pour concrétiser des interactions entre nos entités.

Le diagramme de séquence décrit un scénario qui modélise une exécution particulière d'un cas d'utilisation du début jusqu'à la fin. Il correspond à une

sélection d'enchaînements du cas d'utilisation.

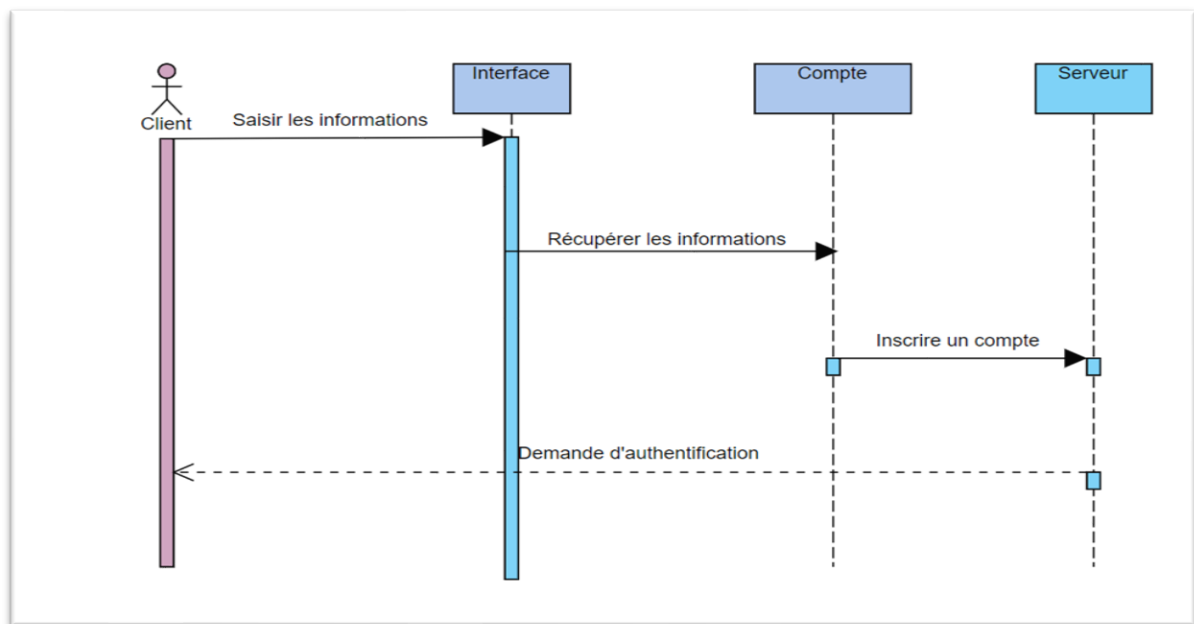


Diagramme de séquence de l'inscription

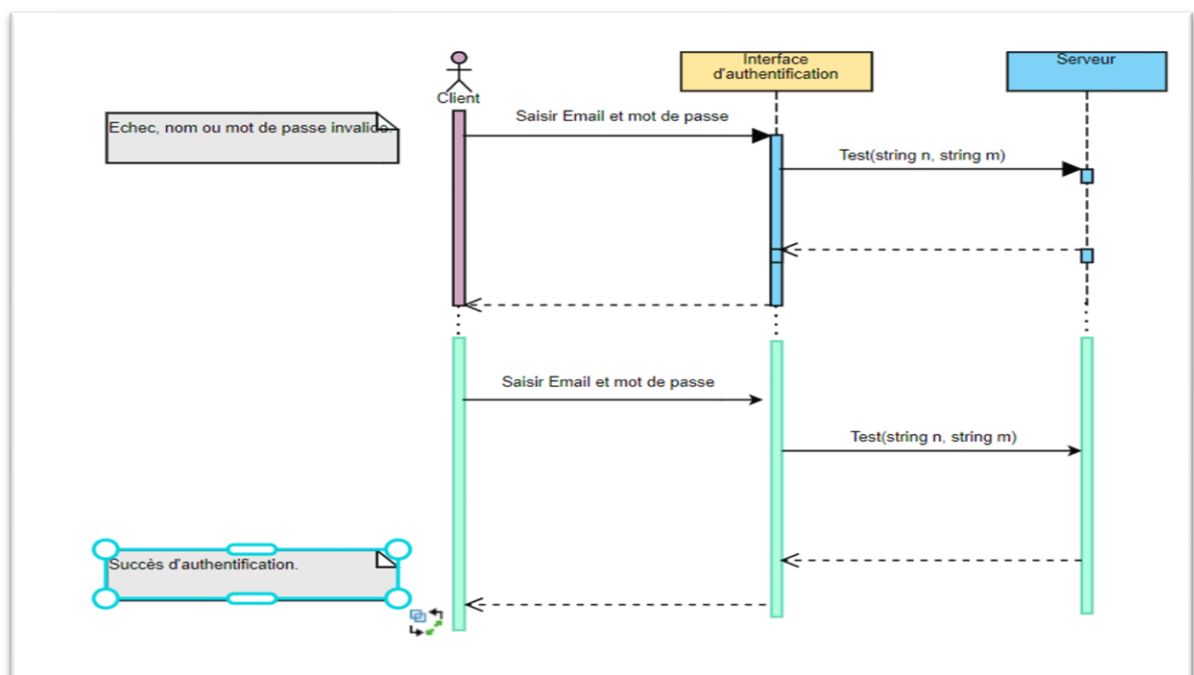


Diagramme de séquence d'authentification

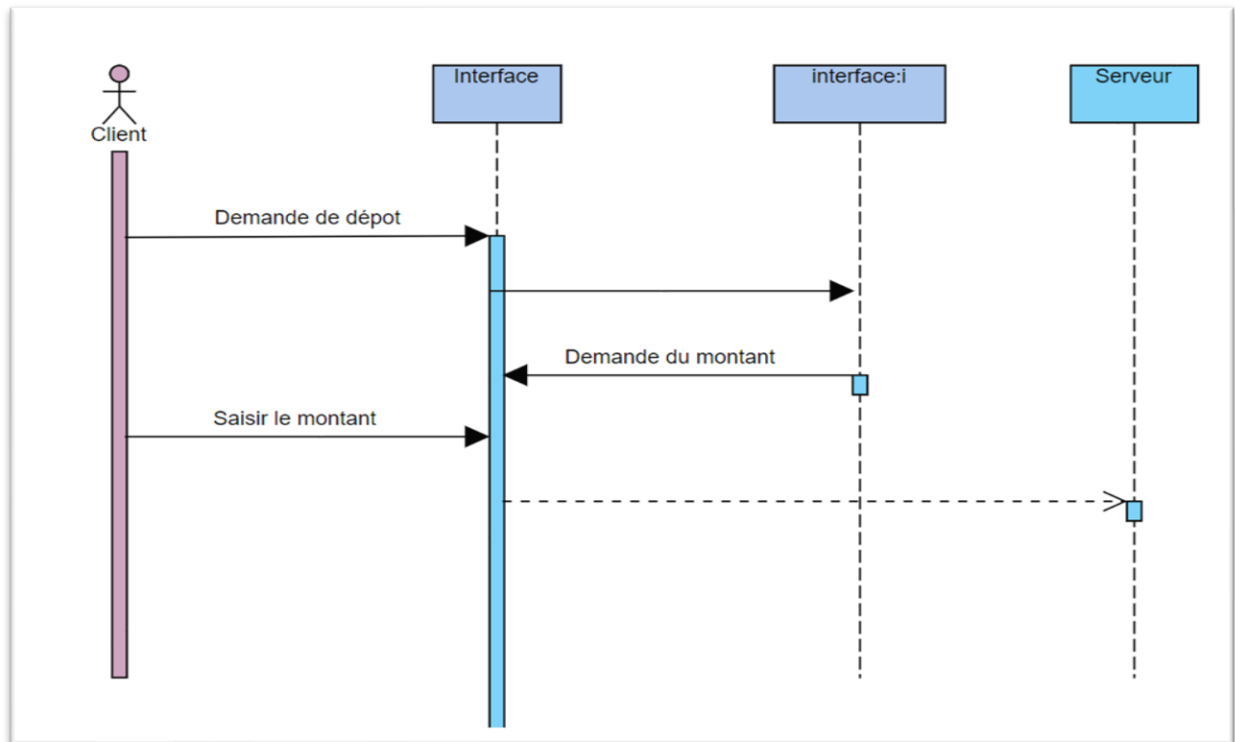


Diagramme de séquence de dépôt

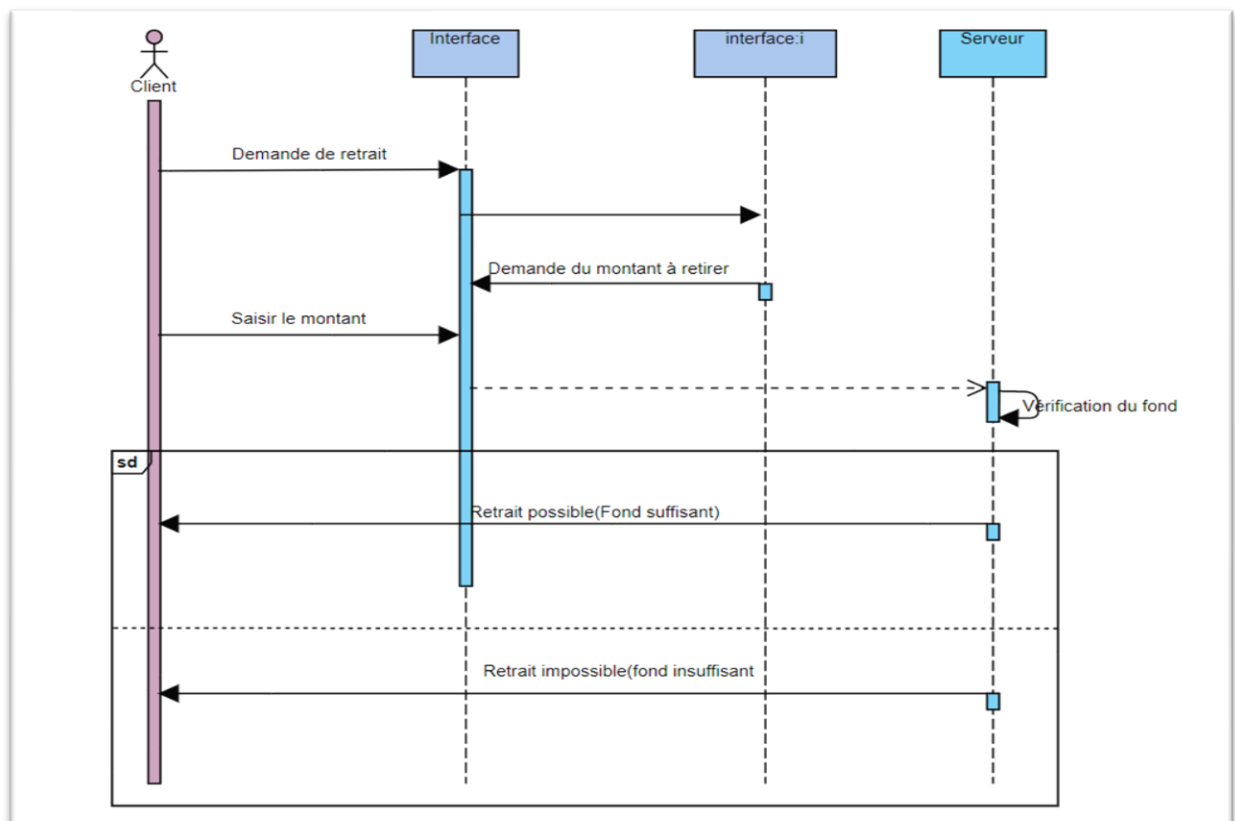
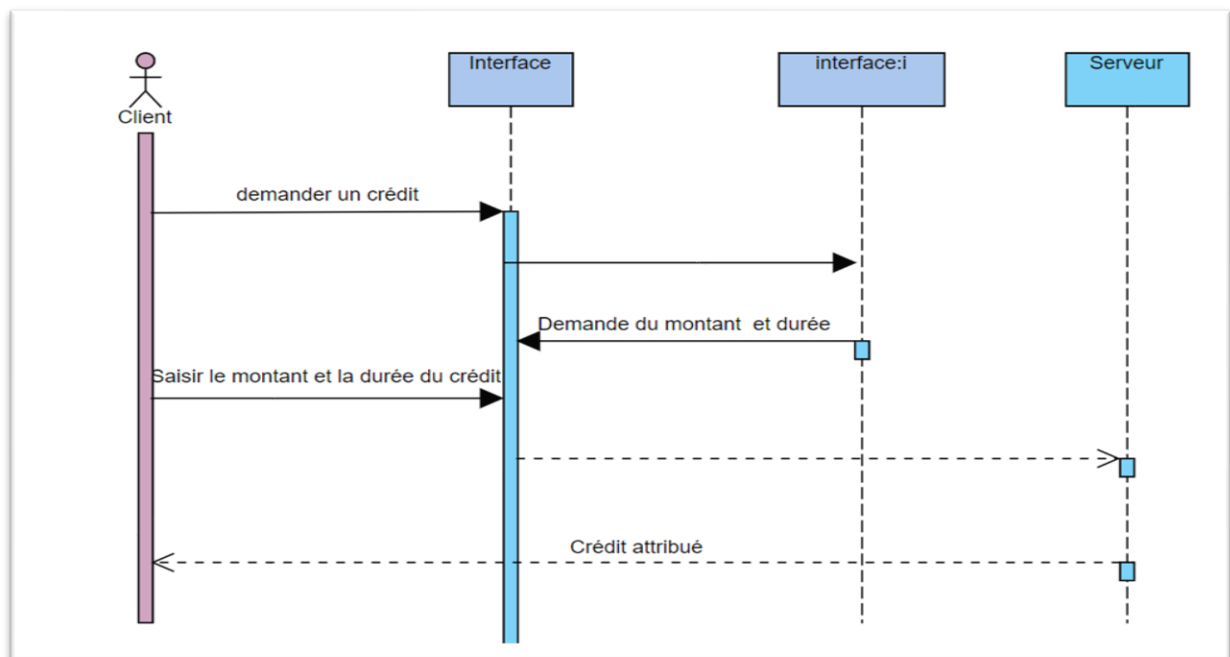


Diagramme de séquence de retrait





*Diagramme de séquence de crédit*

## Conclusion :

Dans ce chapitre nous avons présenté la modélisation de la structure statique et dynamique de notre système.

C'est une étape nécessaire et importante afin de pouvoir créer notre base de données et réaliser l'application.

Dans le prochain chapitre on attaquera les détails de l'application avec ses différents composants et interfaces.

# CHAPITRE 3 : RÉALISATION DU PROJET

## Les outils utilisés :

### ➤ Java



**Java** est à la fois un langage de programmation et un environnement d'exécution. Le langage Java a la particularité principale d'être portable sur plusieurs systèmes d'exploitation tels qu'UNIX, Microsoft Windows, Mac OS ou Linux... C'est la plateforme qui garantit la portabilité des applications développées en Java. Le langage reprend en grande partie la syntaxe du langage C++, très utilisé par les informaticiens. Néanmoins, Java a été épurée des concepts les plus subtils du C++ et à la fois les plus déroutants, tels que l'héritage multiple remplacé par l'implémentation des interfaces.

### Présentation du langage java

#### Bref historique

Développé par Sun Microsystems depuis la fin des années 1980, Java est un langage de programmation à usage général, évolué et orienté objet dont la syntaxe est proche du C. Il existe 2 types de programmes en Java : les applets et les applications. Une application autonome (standalone program) est une application qui s'exécute sous le contrôle direct du système d'exploitation.

Une applet est une application qui est chargée par un navigateur et qui est exécutée sous le contrôle d'un plug in de ce dernier.

#### Les caractéristiques

Java possède un certain nombre de caractéristiques qui ont largement contribué à son énorme succès :

- Java est interprété : le source est compilé en pseudo code ou byte code puis exécuté par un interpréteur Java : la Java Virtual Machine (JVM). Ce concept est à la base du slogan de Sun pour Java : WORA (Write Once, Run Anywhere : écrire une fois, exécuter partout). En effet, le bytecode, s'il ne contient pas de code spécifique à une plate-forme particulière peut être exécuté et obtenir quasiment les mêmes résultats sur toutes les machines disposant d'une JVM.

- Java est indépendant de toute plate-forme : il n'y a pas de compilation spécifique pour chaque plateforme. Le code reste indépendant de la machine sur laquelle il s'exécute. Il est possible d'exécuter des programmes Java sur tous les environnements qui possèdent une Java Virtual Machine. Cette indépendance est assurée au niveau du code source grâce à Unicode et au niveau du byte code.
- Java est orienté objet : comme la plupart des langages récents, Java est orienté objet. Chaque fichier source contient la définition d'une ou plusieurs classes qui sont utilisées les unes avec les autres pour former une application. Java n'est pas complètement objet car il définit des types primitifs (entier, caractère, flottant, booléen,...).
- Java est simple : le choix de ses auteurs a été d'abandonner des éléments mal compris ou mal exploités des autres langages tels que la notion de pointeurs (pour éviter les incidents en manipulant directement la mémoire), l'héritage multiple et la surcharge des opérateurs, ...
- Java est fortement type : toutes les variables sont typées et il n'existe pas de conversion automatique qui risquerait une perte de données. Si une telle conversion doit être réalisée, le développeur doit obligatoirement utiliser un cast ou une méthode statique fournie en standard pour la réaliser.
- Java assure la gestion de la mémoire : l'allocation de la mémoire pour un objet est automatique à sa création et Java récupère automatiquement la mémoire inutilisée grâce au garbage collector qui restitue les zones de mémoire laissées libres suite à la destruction des objets.
- Java est sûr : la sécurité fait partie intégrante du système d'exécution et du compilateur. Un programme Java planté ne menace pas le système d'exploitation. Il ne peut pas y avoir d'accès direct à la mémoire.
- Java est économe : le pseudo code a une taille relativement petite car les bibliothèques de classes requises ne sont liées qu'à l'exécution.
- Java est multitâche : il permet l'utilisation de threads qui sont des unités d'exécution isolées. La JVM, elle-même, utilise plusieurs threads. Ainsi à ce basant sur ces caractéristiques, nous avons porté notre choix sur ce langage pour le développement de notre application, dans le but de pouvoir déployé notre application largement dans n'importe quelle plateforme.

### ➤ *JavaFX*



**JavaFX** est un framework et une bibliothèque d'interface utilisateur issue du projet OpenJFX, qui permet aux développeurs Java de créer une interface graphique pour des applications de bureau, des applications internet riches et des applications smartphones et tablettes tactiles.

### ➤ *SceneBuilder*



**JavaFX Scene Builder** est un outil initialement créé par Oracle et désormais faisant partie de l'OpenJFX qui permet de créer des fichiers au format FXML via un éditeur graphique. Cet outil est disponible en tant qu'application autosuffisante qui peut être lancée depuis votre bureau ou en tant qu'API intégrable dans des outils tiers tels que NetBeans.

### ➤ *MySQL*



Le SGBD MySQL est supporté par un large éventail d'outils. MySQL est surtout installé pour les applications Web, ce SGBD est solide et utilisé par de grands groupes spécialisés dans l'Internet. Plusieurs pilotes natifs de type 4 sont disponibles pour MySQL et sont conseillés pour une utilisation en Java.

### ➤ *Eclipse*



Eclipse est l'environnement de développement (spécialisé pour le langage Java) qui sera utilisé dans ce projet. Le choix d'Eclipse repose essentiellement sur sa gratuité, sa facilité d'utilisation, sa puissance de développement et surtout ses nombreux plugins (bibliothèques additives).

### ➤ *PhpMyAdmin*



PhpMyAdmin est une application Web de gestion pour les systèmes de gestion de base de données MySQL réalisée principalement en PHP et distribuée sous licence GNU GPL.

## Réalisation d'une base de données :

Nous avons commencé par la création d'une base de données qu'on a nommé «Bank» ensuite dans cette base nous avons créé quatre tables: «login», «informations », « operations » et « crédits », comme indiqué ci-dessous:

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	idc	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	id	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	month	float			Non	Aucun(e)			Modifier Supprimer Plus
4	amount	float			Non	Aucun(e)			Modifier Supprimer Plus
5	pricem	float			Non	Aucun(e)			Modifier Supprimer Plus
6	date	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus

La table « crédits » comprend cinq champs :

- Id de client.
- Id de crédit.
- la durée.
- Le montant restant à payer et enfin la mensualité.

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	username	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
3	fname	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	lname	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
5	phoneno	int(30)			Non	Aucun(e)			Modifier Supprimer Plus
6	email	varchar(40)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
7	dob	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
8	gender	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
9	balance	float			Non	Aucun(e)			Modifier Supprimer Plus
10	credit	float			Non	Aucun(e)			Modifier Supprimer Plus
11	Status	varchar(40)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus

La table « information » contient des champs indiquant des informations sur le client et son compte comme le nom, le prénom, l'email, le genre, le solde ..

phpMyAdmin

Récentes | Préférées

bank

Nouvelle table

credits

information

login

operations

Serveur: 127.0.0.1 » Base de données: bank » Table: login

Parcourir | Structure | SQL | Rechercher | Insérer | Exporter | Importer | Privileges | Opérations | Suivi

Structure de table | Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	username	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
3	email	varchar(40)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	password	varchar(30)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus

Tout cocher Avec la sélection : Parcourir Modifier Supprimer Primaire Unique Index Texte entier

La table « login » contient les informations d'identification.

phpMyAdmin

Récentes | Préférées

bank

Nouvelle table

credits

information

login

operations

Serveur: 127.0.0.1 » Base de données: bank » Table: operations

Parcourir | Structure | SQL | Rechercher | Insérer | Exporter | Importer | Privileges | Opérations | Suivi

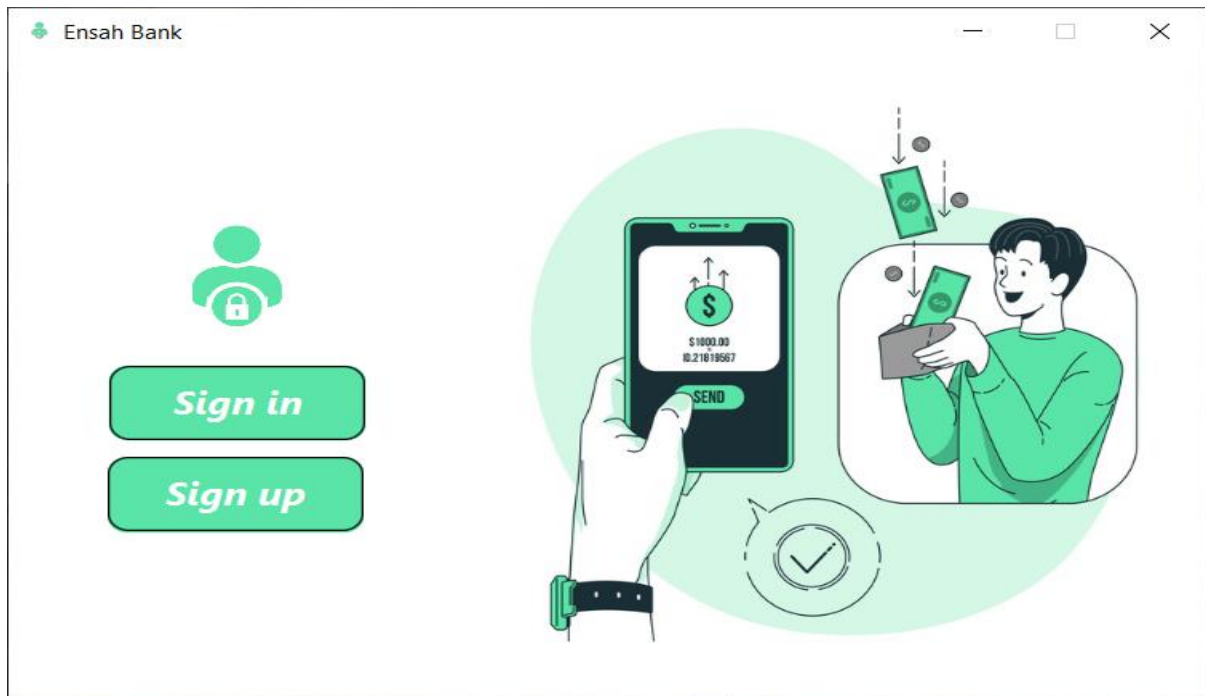
Structure de table | Vue relationnelle

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	ido	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	id	int(11)			Non	Aucun(e)			Modifier Supprimer Plus
3	nameo	varchar(40)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	lnameo	varchar(30)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
5	typeo	varchar(11)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
6	dateo	varchar(30)	utf8mb4_general_ci		Non	Aucun(e)			Modifier Supprimer Plus
7	amount	float			Non	Aucun(e)			Modifier Supprimer Plus

La table « crédits » comprend sept champs, entre lesquels il y a le nom d'opération, son type, sa date et son montant.

## Lancement du projet :

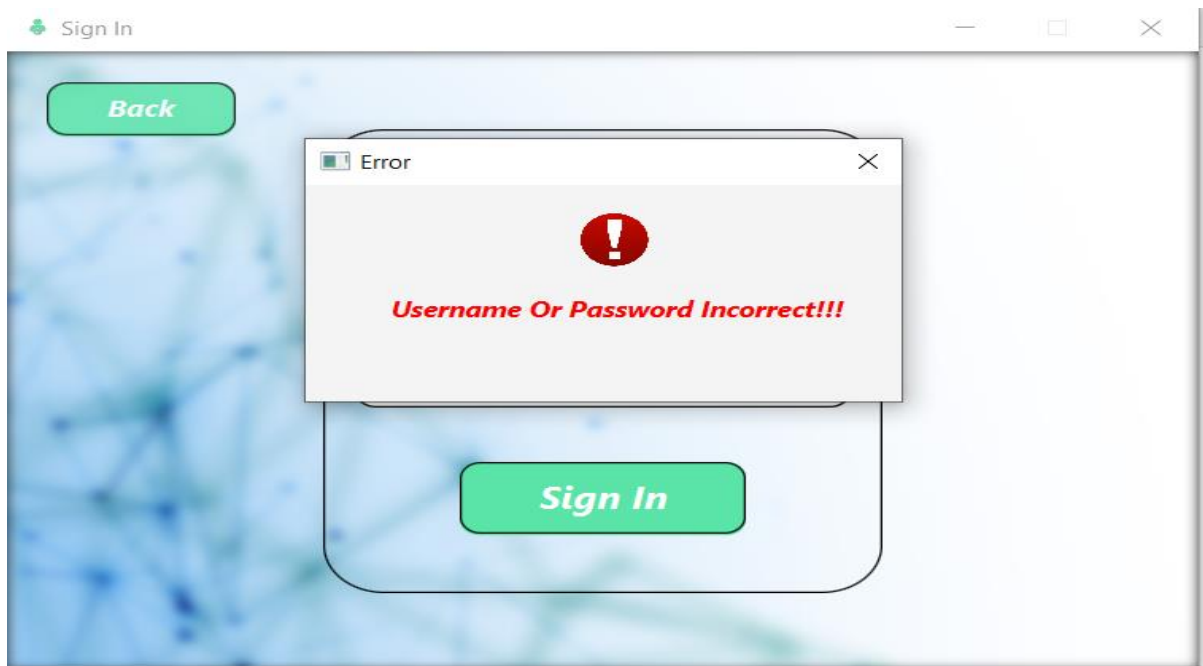
C'est la première page qui s'affiche après l'exécution d'application, elle contient le logo d'application, son nom " Ensah Bank ", et deux boutons: "Sign in" et "Sign up".



La sécurité est nécessaire pour n'importe quelle application, raison pour laquelle nous avons créé un mécanisme d'authentification pour l'ensemble des clients. Donc, il y a un login et mot de passe pour chaque client.

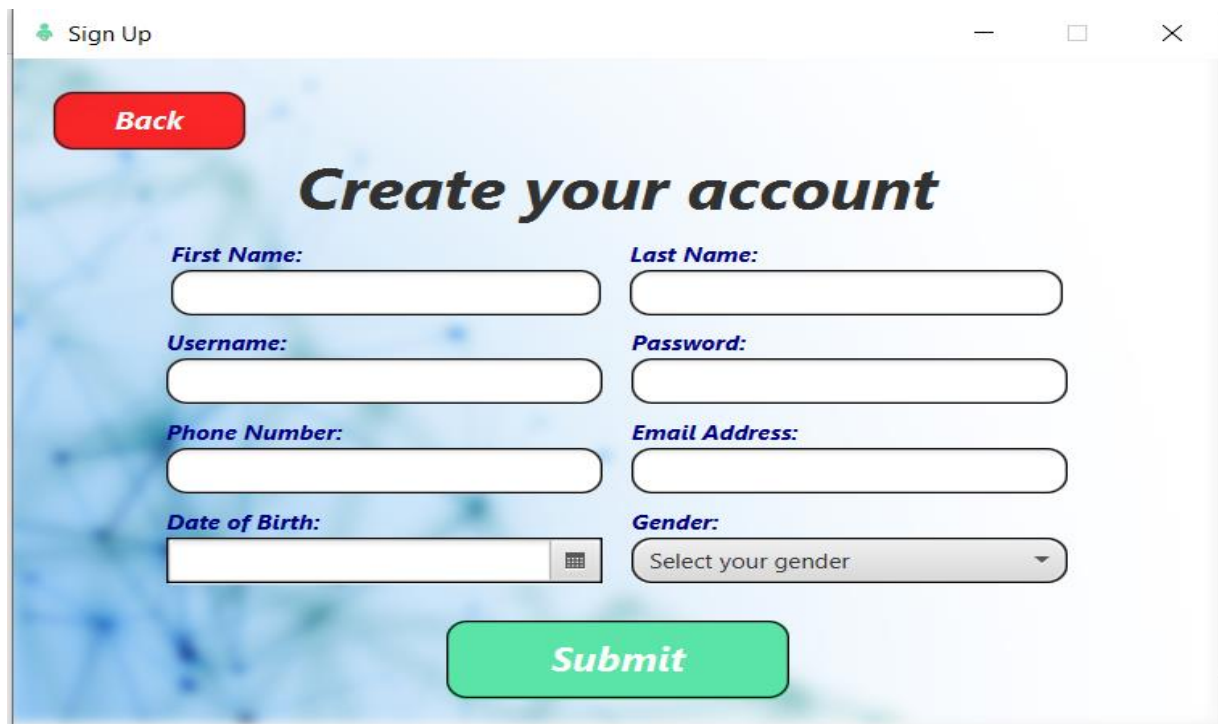
The screenshot shows the 'Sign In' form within a window titled 'Sign In'. In the top left corner, there is a green button labeled 'Back'. The form itself is a rounded rectangle containing two labels: 'Username/Email:' and 'Password:'. Below each label is a corresponding text input field. At the bottom of the form is a green button labeled 'Sign In'. The background of the window features a blue abstract pattern.

Un mauvais mot de passe ou login est aussi géré par notre application en cas d'erreurs de saisie comme indiqué dans la figure suivante :



Les clients obtiennent leurs login(s) et mots de passe après l'inscription.



A screenshot of a web application window titled "Sign Up". The window has a light blue background with a subtle pattern. At the top left, there is a green plus icon and the text "Sign Up". At the top right, there are standard window control icons (minimize, maximize, close). Below the title bar, there is a red button labeled "Back". The main heading is "Create your account" in a large, bold, black font. Below the heading, there are seven input fields arranged in two columns. The left column contains: "First Name:" with a text input field, "Username:" with a text input field, "Phone Number:" with a text input field, and "Date of Birth:" with a date picker. The right column contains: "Last Name:" with a text input field, "Password:" with a text input field, "Email Address:" with a text input field, and "Gender:" with a dropdown menu showing "Select your gender". At the bottom center, there is a large green button labeled "Submit".

Sign Up

[Back](#)

## Create your account

**First Name:**

**Last Name:**

**Username:**

**Password:**

**Phone Number:**

**Email Address:**

**Date of Birth:**

**Gender:**

[Submit](#)

### Espace client :

C'est la page qui s'affiche lorsque le client s'est identifié correctement. A partir de cette page, il peut accéder aux autres pages de l'application en utilisant le menu.

The screenshot shows the 'Ensah Bank' web application. At the top, there's a navigation bar with tabs: 'Deposit' (selected), 'Withdraw', 'Money Transfer', 'Change Details', 'Request a Loan', 'Loan Informations', and 'Operations'. The main content area has a light blue background with a network pattern. On the left, there are buttons for deposit amounts: '100 DH', '200 DH', '500 DH', and '1000 DH'. In the center, a box displays 'CURRENT BALANCE : 20725.0 MAD'. Below this, there's a 'Deposit : 0 MAD' input field with 'PROCEED' and 'RESET' buttons. On the right, there's a sidebar with 'Username: kao', 'ID: 5', and a 'Sign out' button. At the bottom right of the main area, there are buttons for '2000 DH', '3000 DH', '5000 DH', and '10000 DH'.

L'application donne aux clients la possibilité de gérer leurs comptes et faire quelques opérations tel que: Déposer, Débiter, Virement vers un autre compte, Consulter, Créditer, et enfin consulter son crédit et son historique d'opérations .

Donc nous allons présenter dans ce qui suit l'ensemble des IHMs proposées aux clients en fonction de leurs demandes :

### Deposit

Un client peut choisir un montant entre les propositions ou écrire le montant qu'il souhaite là-bas comme indiqué dans la figure ci-dessus. Il a le droit de réinitialiser et récrire avant de cliquer sur 'PROCEED'.

### Withdraw

La même remarque est valable pour l'opération Withdraw.

Ensah Bank

Deposite Withdraw Money Transfer Change Details Request a Loan Loan Informations Operations

Username: LOL  
ID: 5  
**Sign out**

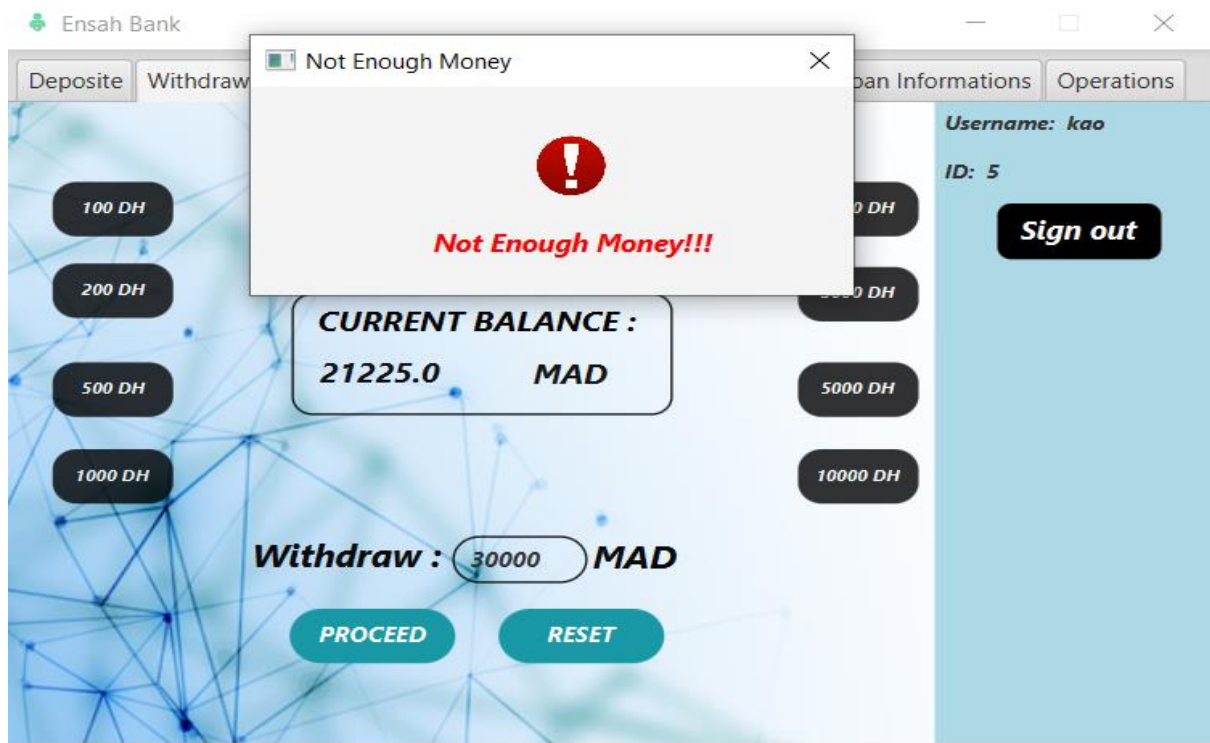
**CURRENT BALANCE :**  
**99999.0 MAD**

**Withdraw :**  **MAD**

**PROCEED** **RESET**

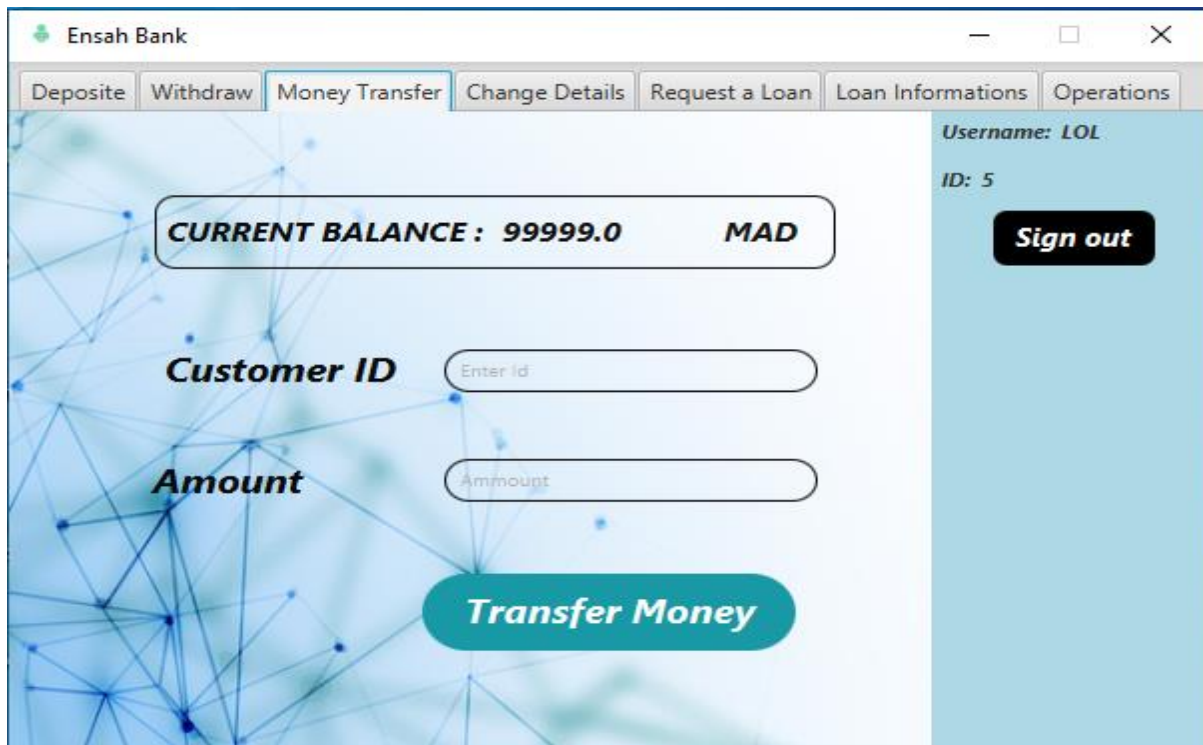
100 DH 200 DH 500 DH 1000 DH 2000 DH 3000 DH 5000 DH 10000 DH

On ne peut pas débiter un compte sauf si son solde 'Balance' le permet comme indiqué dans la figure suivante.

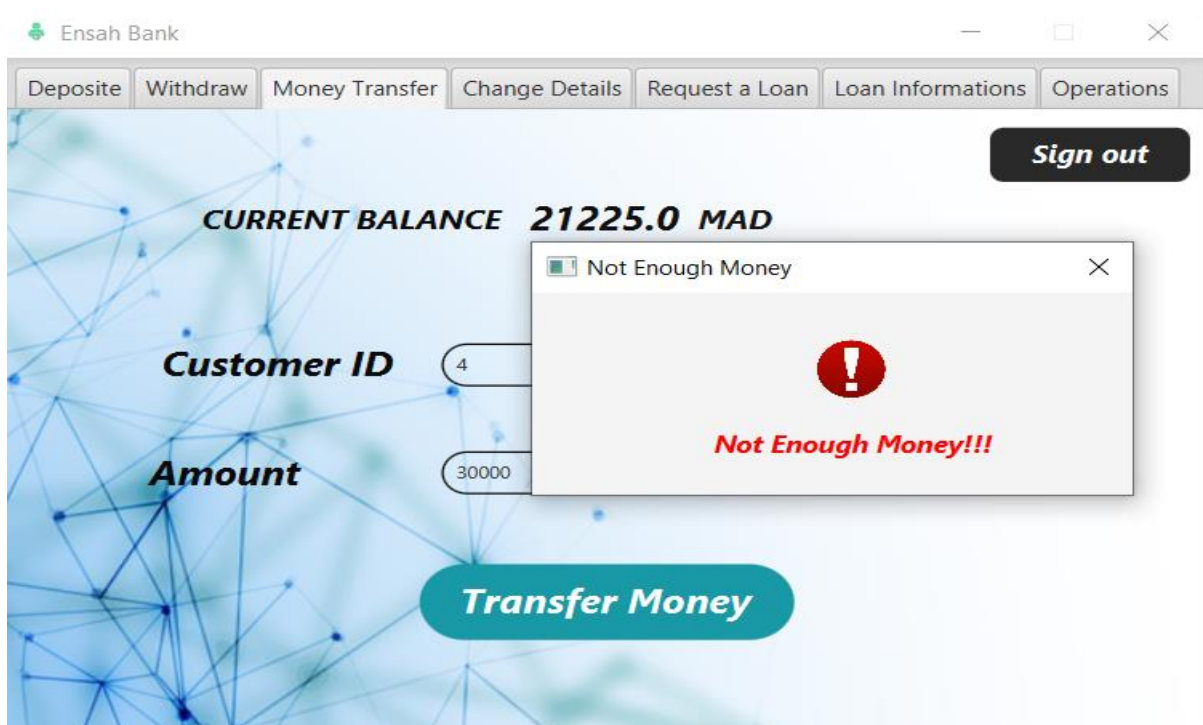


## Money Transfer

Cette opération demande de saisir l'id de client récepteur et le montant du virement.



Il est aussi impossible de faire le virement si le compte émetteur ne contient pas le montant indiqué.



## Change Details

La consultation d'un compte est comme suit :

Ensah Bank

Deposit Withdraw Money Transfer **Change Details** Request a Loan Loan Information Operations

**First Name:** LOL

**Last Name:** LOL

**Phone Number:** 34567890

**Email Address:** SDJGS@GMAIL.COM

**Date of Birth:** 1/14/2021

**Gender:** Male

**Close Account** **Update**

**Username:** LOL

**ID:** 5

**Sign out**

Tout titulaire d'un compte en banque peut décider de sa clôture quand il le souhaite. Il doit seulement cliquer sur le bouton **"Close Account"**, à condition que son solde ne dépasse pas 1 MAD et qu'il n'a pas un crédit.

### Request a Loan

Le client a le droit de demander un crédit en indiquant la durée et le montant souhaité.



Ensah Bank

Deposit Withdraw Money Transfer Change Details Request a Loan Loan Information Operations

## APPLY FOR A LOAN

**Installment on:** Months

**Amount:** 0

**TAKE A CREDIT**

Username: LOL  
ID: 5  
**Sign out**

### Loan Informations

Ce service donne au client la possibilité de voir les informations sur son crédit : les mois restants, le montant restant, la mensualité et la date du crédit. Mais le plus important est de payer ses mensualités.

Ensah Bank

Deposit Withdraw Money Transfer Change Details Request a Loan Loan Information Operations

## LOAN INFORMATION

**Remaining Months:** 1 Months

**Remaining Loan:** 0 MAD

**Price Per Month:** 0 MAD

**Date of Credit:** 02/11/2021 17:03:09

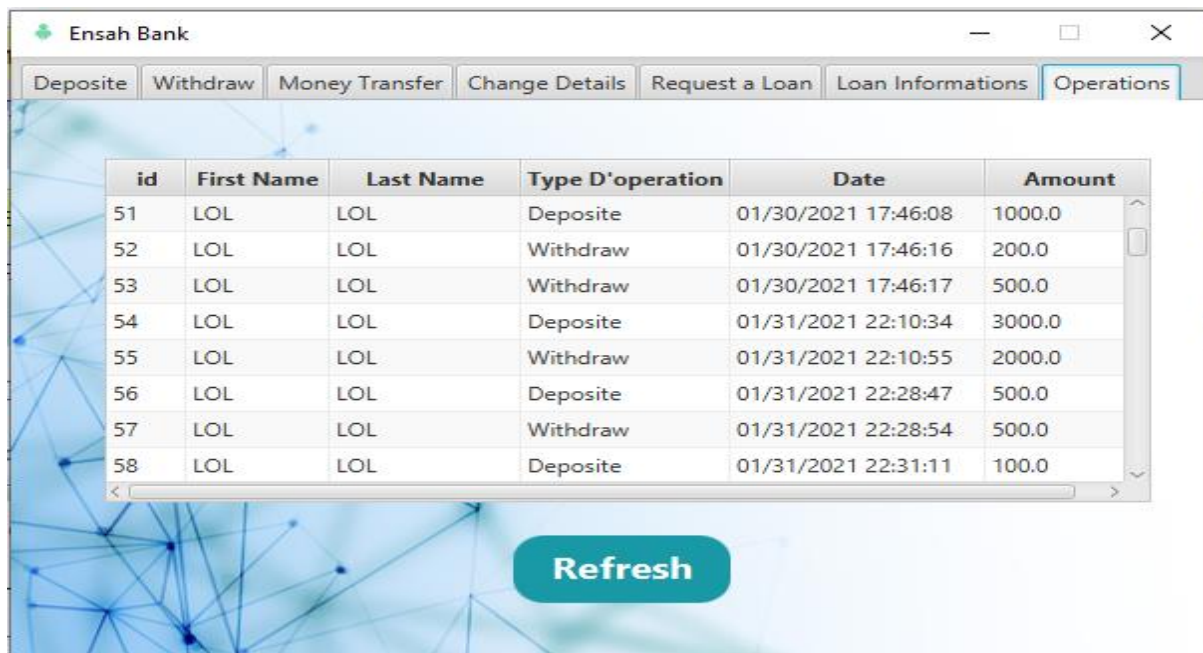
**Pay An Installment**

Username: LOL  
ID: 5  
**Sign out**

Le client va payer en total la valeur acquise (emprunt + intérêts) divisée en la durée qui a choisi.

### Opérations

« opérations » comprend la liste de tous les opérations effectuées par le client en spécifiant le type d'opération, sa date et son montant.



The screenshot shows a web application window titled 'Enсах Bank'. It has several tabs: 'Deposit', 'Withdraw', 'Money Transfer', 'Change Details', 'Request a Loan', 'Loan Informations', and 'Operations'. The 'Operations' tab is selected, displaying a table of transactions. Below the table is a 'Refresh' button.

id	First Name	Last Name	Type D'operation	Date	Amount
51	LOL	LOL	Deposit	01/30/2021 17:46:08	1000.0
52	LOL	LOL	Withdraw	01/30/2021 17:46:16	200.0
53	LOL	LOL	Withdraw	01/30/2021 17:46:17	500.0
54	LOL	LOL	Deposit	01/31/2021 22:10:34	3000.0
55	LOL	LOL	Withdraw	01/31/2021 22:10:55	2000.0
56	LOL	LOL	Deposit	01/31/2021 22:28:47	500.0
57	LOL	LOL	Withdraw	01/31/2021 22:28:54	500.0
58	LOL	LOL	Deposit	01/31/2021 22:31:11	100.0

### Sign out

Quand l'utilisateur veut se déconnecter, il clique sur le bouton

**"Sign out"** comme indiqué dans la figure suivante.





### Conclusion :

Ce chapitre présente tous les résultats de notre projet et montre tous les cas et les opérations possibles et impossibles lors de la gestion de crédits bancaire.

## CONCLUSION GÉNÉRALE :

Dans ce projet, nous avons conçu et réalisé une application de gestion de crédits bancaire avec Java.

On a eu recours dans ce projet à la technologie JavaFX vu que celle-ci contient des outils très divers.

Cette application offre des services bancaires à ses clients. Nous avons défini et évoqué les outils utilisés pour réaliser notre application, ainsi que le déroulement de l'application et les résultats obtenus. La réalisation de l'application s'est déroulée d'une manière itérative. Pendant chaque itération un produit partiel était développé tout en mettant en évidence les activités facultatives (la conception, le développement et le test).

La réalisation de ce projet a été une bonne occasion pour nous d'une part pour acquérir de nouvelles connaissances, et d'autre part, d'assimiler les différents outils acquis durant ce semestre en module de programmation JAVA.

Notre rapport est constitué de trois chapitres. Le premier chapitre en guise de vue théorique du sujet traité, le deuxième chapitre est consacré à la phase de la conception, et le troisième chapitre présente l'application réalisée dans le cadre de ce projet, ainsi les outils utilisés comme JAVA et JavaFX.

Vidéo drive :