Une image contenant texte

Description générée automatiquement

**Conception et développement d’une application**

**de gestion de banque**

**Réalisé par :** salma el allam

Aya belhachmi

Année Universitaire 2023-2024 Groupe :4IIR8

# 

# **Remerciement**

Au terme de ce travail, nous tenons à exprimer nos profondes gratitudes à notre

Cher professeur et encadrant M.Omar El midaoui, enseignant au sein de l’Ecole

Marocaine des Sciences de L’ingénieur (EMSI-RABAT) pour nous avoir permis

de réaliser ce travail riche en nouveauté. Nous le remercions très sincèrement pour

son encadrement, son aide précieuse et sa confiance.

Nous ne nous laisserons pas cette occasion passer, sans remercier tous les

enseignants et le personnel de l’Ecole Marocaine des Sciences de L’ingénieur

(EMSI-RABAT), et particulièrement ceux du département génie informatique

pour leur aide et leurs précieux conseils et pour l’intérêt qu’ils portent à notre

formation.

**Résume**

Les principaux objectifs de ce projet se résument comme suit :

* Évaluer les compétences théoriques acquises durant toute ce semestre et les mettre en pratique.
* Renforcer son aptitude d’analyse et d’organisation
* Développer ses connaissances professionnelles.
* Améliorer le savoir-faire et le savoir être.
* Saisir la valeur du travail en équipe

Pour mieux vous tenir à proximité des différentes étapes suivies pour la réalisation de ce projet, on vous propose ce rapport qui se compose de trois chapitres principaux :

Dans le premier chapitre, on présente le contexte général du projet ainsi qu’une brève description du déroulement du projet, et les problématiques et objectifs du rapport.

# Sommaire

Table des matières

[**Remerciement** 2](#_Toc146014865)

[Sommaire 4](#_Toc146014866)

[Liste des figures 5](#_Toc146014868)

[INTRODUCTION 6](#_Toc146014869)

[**Chapitre 1 : Contexte général du projet** 7](#_Toc146014870)

[**I.** **Présentation du projet :** 8](#_Toc146014871)

[**1. Contexte :** 8](#_Toc146014873)

[**2.Objectif** 8](#_Toc146014874)

[II. Spécification des besoins (cahier de charge) : 8](#_Toc146014875)

[**1.** **Problématique** 9](#_Toc146014876)

[**2.** **Solution** 9](#_Toc146014877)

[**3.** **Besoins fonctionnels** 9](#_Toc146014878)

[**4.** **Besoins non fonctionnels** 9](#_Toc146014879)

[**Chapitre 2 : Analyse et conception** 11](#_Toc146014880)

[I. Méthode de conception 12](#_Toc146014881)

[1. Le choix UML 12](#_Toc146014882)

[2. Outils de modélisation (Lucidchart) 12](#_Toc146014883)

[II. Diagrammes 12](#_Toc146014884)

[1.Diagramme de cas d’utilisation 13](#_Toc146014885)

[2.Diagramme de séquence 13](#_Toc146014885)

[3.Diagramme de classe 16](#_Toc146014886)

[Chapitre 3 : Réalisation et présentation de l’application 17](#_Toc146014887)

[I. Technologies de programmation Back-end : 18](#_Toc146014888)

[1. JEE : 1](#_Toc146014889)8

[2. Spring boot : 18](#_Toc146014890)

[II. Technologies de programmation Front-end : 19](#_Toc146014891)

[1. Thymeleaf: 19](#_Toc146014892)

[2. CSS: 19](#_Toc146014893)

Outils de développement :.............................................................................................................19

Présentation des exemples des parties réalisées :........................................................................20

# Liste des figures

[FIGURE 1: Page de connexion](#_Toc75754542) 20

[FIGURE 2:Formulaire d’ajout banque](#_Toc75754542) 20

[FIGURE 3: Liste des banques 21](#_Toc75754543)

[FIGURE 4: Page qui affiche les informations sur une banque 21](#_Toc75754543)

FIGURE 5: Liste des clients……………………..……………………………………...................…22

FIGURE 6: Formulaire des clients ............................………………………………………………22

FIGURE7:Pagequiaffichelesinformationssurunclient............................……………..………………23

FIGURE 8: Modification des informations d’un client. …………………………………………………23

FIGURE9:Formulaire add un nouveau compte ……………………………………………..………24FIGURE10: formulaire edit un compte………………………….………..................................24

FIGURE 11 : un transfert d'argent de compte a autre.................……………………………………25FIGURE 12: l'historique des transactions par id ………..……..............……………………………25

# INTRODUCTION

Dans le cadre de notre formation universitaire, il est essentiel d'acquérir une expérience technique afin de mettre en pratique et de tester toutes les connaissances théoriques acquises au cours du JEE. L'objectif principal de ce projet est de réaliser une application de gestion bancaire. Cette application permettra de suivre la traçabilité des opérations effectuées sur les sauvegardes.

Le présent rapport est structuré en trois chapitres, détaillés comme suit :

- \*Premier chapitre : Présentation de l'organisme d'accueil, vue d'ensemble du projet et expression des nouveaux besoins.

- \*Deuxième chapitre : Analyse des spécifications fonctionnelles et conception de l'application.

- \*Troisième chapitre : Présentation du travail réalisé, des différents outils de développement utilisés et de l'implémentation de l'application.

Enfin, ce mémoire se conclut par une conclusion générale et propose des perspectives pour l'amélioration de l'application.

# **Chapitre 1 : Contexte général du projet**

## **Présentation du projet :**

### **1. Contexte :**

Le contexte d'une application de gestion bancaire repose généralement sur une plateforme permettant aux banques de gérer leurs activités quotidiennes. Elle offre une visibilité claire sur toutes les transactions effectuées et contribue également à améliorer l'efficacité opérationnelle.

**2.Objectif :**

Pour notre projet d, nous avons assigné à notre étude les objectifs suivants :

* Améliorer l'efficacité opérationnelle : Réduire les temps de traitement des transactions et augmenter la productivité grâce à l'automatisation.
* Réduire les erreurs : Minimiser les erreurs humaines en automatisant la gestion des comptes et des transactions.
* Améliorer le service client : Offrir des services plus rapides et fiables, permettant aux clients d'effectuer des transactions en ligne.
* Assurer une gestion centralisée : Permettre à l'administrateur de gérer efficacement les banques, les clients, les comptes et les opérations à partir d'une interface centralisée.

## Spécification des besoins (cahier de charge) :

Pour atteindre les objectifs de notre projet, l'étude préalable des besoins théoriques et spécifiques est une étape cruciale. En adoptant une approche par étapes, nous allons nous concentrer sur les concepts de base liés à notre travail.Nous allons spécifiée nos besoins en présentant la problématique, la solution proposée, les choix techniques adoptés, ainsi que l’architecture (design) globale du projet.

### **Problématique**

La gestion bancaire manuelle est sujette à des erreurs, prend du temps, coûte cher et limite le service client, nécessitant une automatisation pour améliorer l'efficacité et la satisfaction des clients.

### **Solution**

l'idée generale du projet consiste a recevoir un outil applicatif qui pourra de façon concrete

permettre les banques de gerer le système tout en offrant des outils pour faciliter mes processus de facturation et gagner plus de temps.

### **Besoins fonctionnels**

#### Liste des acteurs :

On doit réaliser une interface web qui fait la gestion des banques, pour cela on s’intéresse à un acteur :

* ADMIN
* Authentification
* authentification
* gestion des clients
* gestions des banques
* gestion des comptes
* gestion des operations
* Log out

### **Besoins non fonctionnels**

Les besoins non-fonctionnels sont des besoins optionnels, ou des besoins/contraintes liés à l'implémentation (contraintes de qualité, de langage ou de plate-forme, par exemple).

Le système à concevoir doit garantir :

➢ La disponibilité : Il est indispensable que l’application soit disponible à tout moment.

➢ La sécurité : l’application devra être hautement sécurisée, sur 3 niveaux :

✓ Le développement : limiter l’accès au code développé et sa modification.

✓ Le déploiement : la mise à jour de l’application sur un environnement donné

✓ L’accès : les informations ne devront pas être accessibles à tout le monde, c’est- à-dire accessible par un identifiant et un mot de passe attribué à la personne concernée.

➢ La performance : l’application devra être performante, c’est-à-dire que le système doit réagir dans un délai précis, quel que soit l’action de l’utilisateur.

➢ La convivialité : l’application doit être simple et facile à manipuler.

➢ L’ergonomie : l’interface utilisateur doit être présentable, bien clair, compréhensible, et facile à utiliser.

# **Chapitre 2 : Analyse et conception**

## Méthode de conception

### Le choix UML

Le langage UML (Unified Modeling Language) est constitué de diagrammes intégrés utilisés par les développeurs informatiques pour la représentation visuelle des objets, des états et des processus dans un logiciel ou un système. Le langage de modélisation peut servir de modèle pour un projet et garantir une architecture d’information structurée ; il peut également aider les développeurs à présenter leur description d’un système d’une manière compréhensible pour les spécialistes externes. UML est principalement utilisé dans le développement de logiciels orientés objet.

### Outils de modélisation (Lucidchart)

Lucidchart est une application web populaire utilisée pour créer des diagrammes et des schémas. Il s'agit d'un outil de modélisation visuelle qui permet aux utilisateurs de représenter graphiquement des concepts, des processus, des flux de travail, des architectures, des organigrammes et bien plus encore.

## Diagrammes

1.Diagramme de cas d’utilisation

Les diagrammes de cas d'utilisation (DCU) sont des diagrammes UML utilisés pour une représentation du comportement fonctionnel d'un système logiciel. Ils sont utiles pour des présentations auprès de la direction ou des acteurs d'un projet, mais pour le développement, les cas d'utilisation sont plus appropriés.

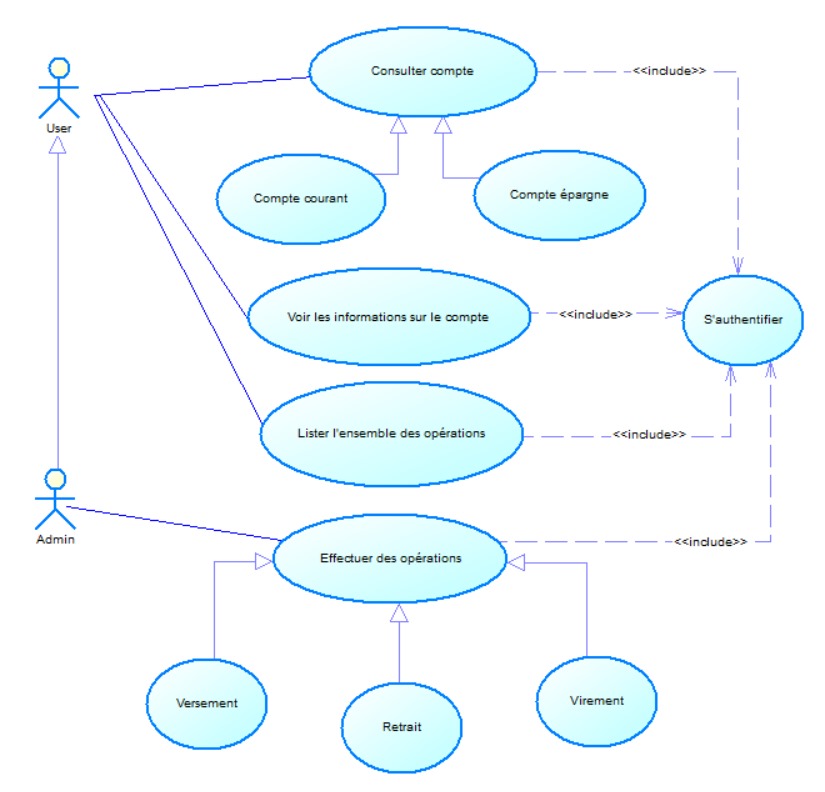


FIGURE 1 : DIAGRAMME CAS D’UTILISATION GLOBALE

### 2.Diagramme de séquence

Les diagrammes de séquences sont la représentation graphique des [interactions](https://fr.wikipedia.org/wiki/Unified_Modeling_Language) entre les [acteurs](https://fr.wikipedia.org/wiki/Acteur_(UML)) et le système selon un ordre chronologique dans la formulation [Unified Modeling Language](https://fr.wikipedia.org/wiki/Unified_Modeling_Language" \o "Unified Modeling Language)

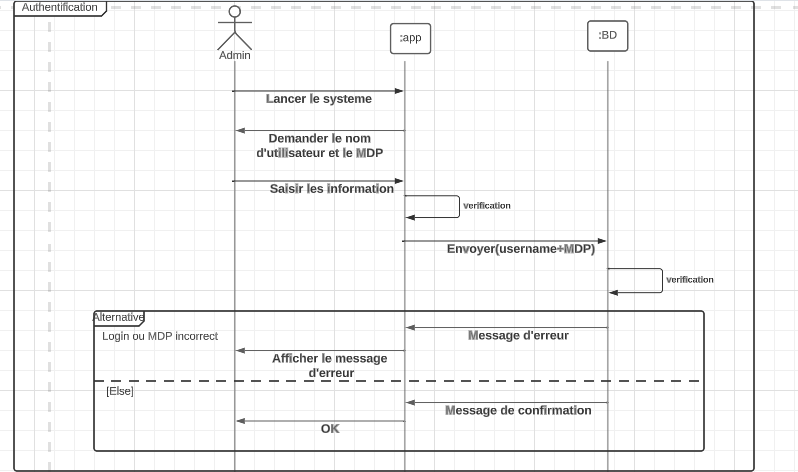


FIGURE 2 : DIAGRAMME DE SEQUENCE D’AUTHENTIFICATION

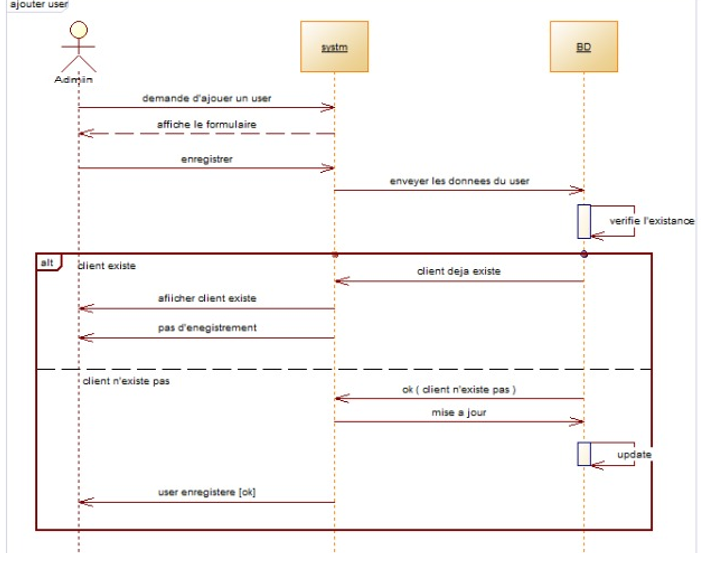


FIGURE 3: DIAGRAMME DE SEQUENCE DE AJOUTER USER

### 3.Diagramme de classe

Le diagramme de classe est un schéma utilisé en [génie logiciel](https://fr.wikipedia.org/wiki/G%C3%A9nie_logiciel) pour présenter les [classes](https://fr.wikipedia.org/wiki/Classe_(informatique)) et les [interfaces](https://fr.wikipedia.org/wiki/Interface_(informatique)) des systèmes ainsi que leurs relations. Ce [diagramme](https://fr.wikipedia.org/wiki/Diagramme) fait partie de la partie [statique](https://fr.wikipedia.org/wiki/Statique) d'[UML](https://fr.wikipedia.org/wiki/Unified_modeling_language), ne s'intéressant pas aux aspects temporels et [dynamiques](https://fr.wikipedia.org/wiki/Dynamique).

Une [classe](https://fr.wikipedia.org/wiki/Classe_(informatique)) décrit les responsabilités, le comportement et le type d'un ensemble d'objets. Les éléments de cet ensemble sont les [instances](https://fr.wikipedia.org/wiki/Instance_(programmation)) de la classe.

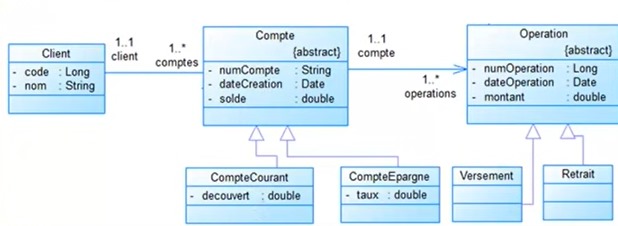


FIGURE 4 : DIAGRAMME DE CLASSE GENERAL

# Chapitre 3 : Réalisation et présentation de l’application

Apres avoir affecter l’étude de la conception de notre application, nous passons à la phase d’implémentation. Ce chapitre présente le résultat du travail effectue durant ce projet de fin d’année. Nous allons présenter aussi l’environnement matériel et les outils de développement utilisés. Nous clôturons ce chapitre par quelques captures d’écran démontrant les fonctionnalités de notre application

## Technologies de programmation Back-end :

### JEE :

 Java EE est une plateforme de développement logiciel utilisée pour créer des applications d'entreprise robustes et évolutives. Elle offre un ensemble d'API

et de spécifications pour le développement d'applications distribuées, de services web, de composants d'entreprise et bien plus encore. Java EE inclut des fonctionnalités

telles que la gestion des transactions, la persistance des données, la sécurité et la gestion des ressources. Cette plateforme permet aux développeurs de créer des applications

hautement performantes et sécurisées pour répondre aux besoins complexes des entreprises.

### Spring boot :

 Spring Boot est un framework Java qui facilite le développement d'applications en fournissant une configuration par défaut pour les projets Spring. Il simplifie la configuration et la mise en place de projets Spring en utilisant des conventions plutôt que des configurations explicites, ce qui permet aux développeurs de se concentrer sur l'écriture du code métier plutôt que sur la configuration de l'infrastructure.

## Technologies de programmation Front-end :

### Thymeleaf:

### Thymeleaf est un moteur de template Java qui permet de créer des vues HTML avec une syntaxe simple et intuitive.

### Il peut être utilisé pour générer des pages web dynamiques en intégrant des données provenant de votre application Java dans les templates HTML.

### Thymeleaf prend en charge la liaison de données bidirectionnelle entre le modèle Java et les vues HTML, ce qui facilite l'affichage et la manipulation

données dans vos pages web.

### CSS :

Une image contenant texte, trousse de secours, signe

Description générée automatiquement

Les feuilles de style en cascade[1](https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade#cite_note-1), généralement appelées CSS de l'[anglais](https://fr.wikipedia.org/wiki/Anglais) Cascading Style Sheets, forment un [langage informatique](https://fr.wikipedia.org/wiki/Langage_informatique) qui décrit la présentation des documents [HTML](https://fr.wikipedia.org/wiki/Hypertext_markup_language) et [XML](https://fr.wikipedia.org/wiki/Extensible_markup_language). Les [standards](https://fr.wikipedia.org/wiki/Standard_technique) définissant CSS sont publiés par le [World Wide Web Consortium](https://fr.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C). Introduit au milieu des [années 1990](https://fr.wikipedia.org/wiki/Ann%C3%A9es_1990), CSS devient couramment utilisé dans la [conception de sites web](https://fr.wikipedia.org/wiki/Conception_de_sites_web) et bien pris en charge par les [navigateurs web](https://fr.wikipedia.org/wiki/Navigateur_web) dans les [années 2000](https://fr.wikipedia.org/wiki/Ann%C3%A9es_2000).

1. Outils de développement :

 est un IDE populaire utilisé pour le développement de logiciels Java, ainsi que pour une variété d'autres langages de programmation,

tels que Kotlin, Scala, Groovy, et bien d'autres.Il offre une large gamme de fonctionnalités qui facilitent le processus de développement,

1. Présentation des exemples des parties réalisées :

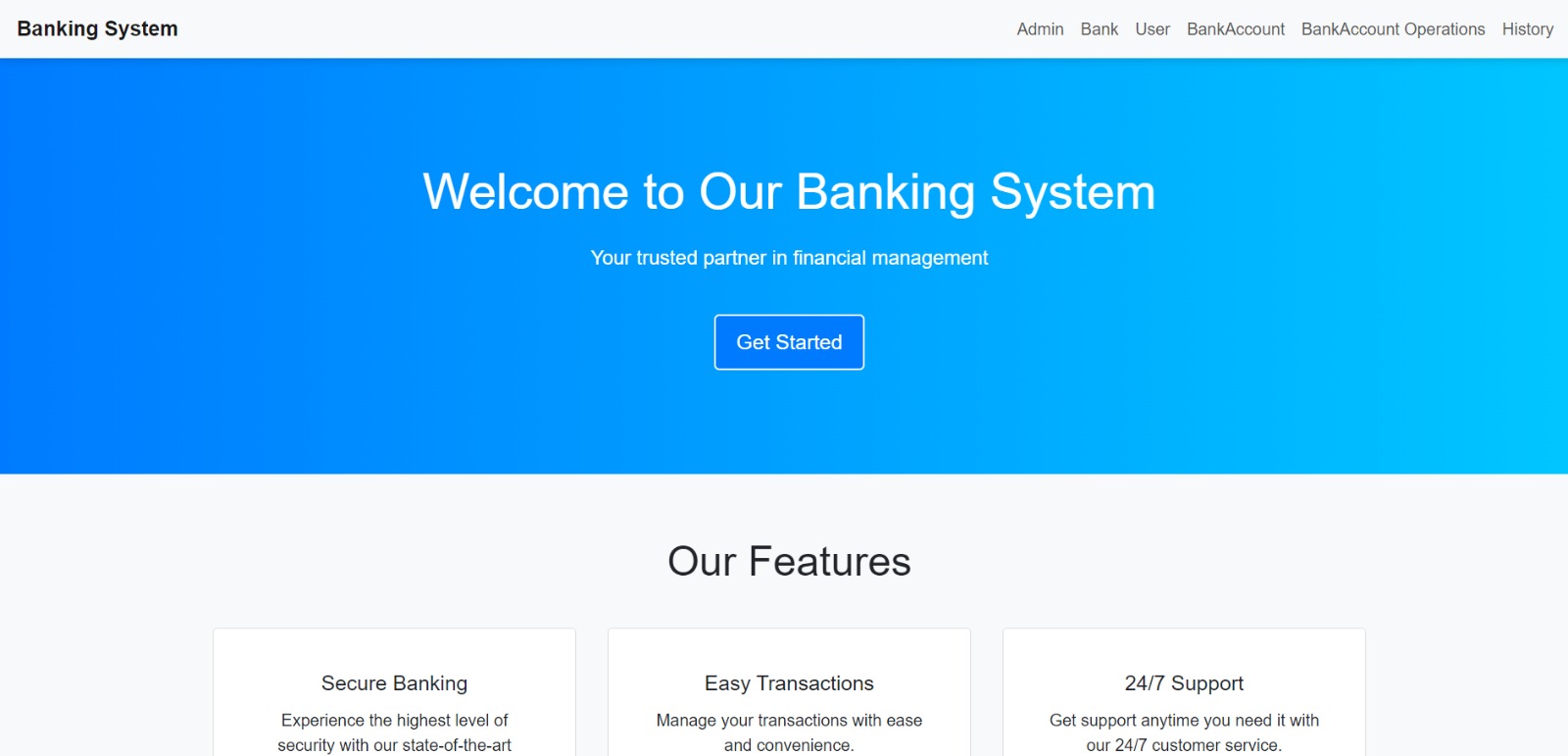


FIGURE 1: Page de connexion

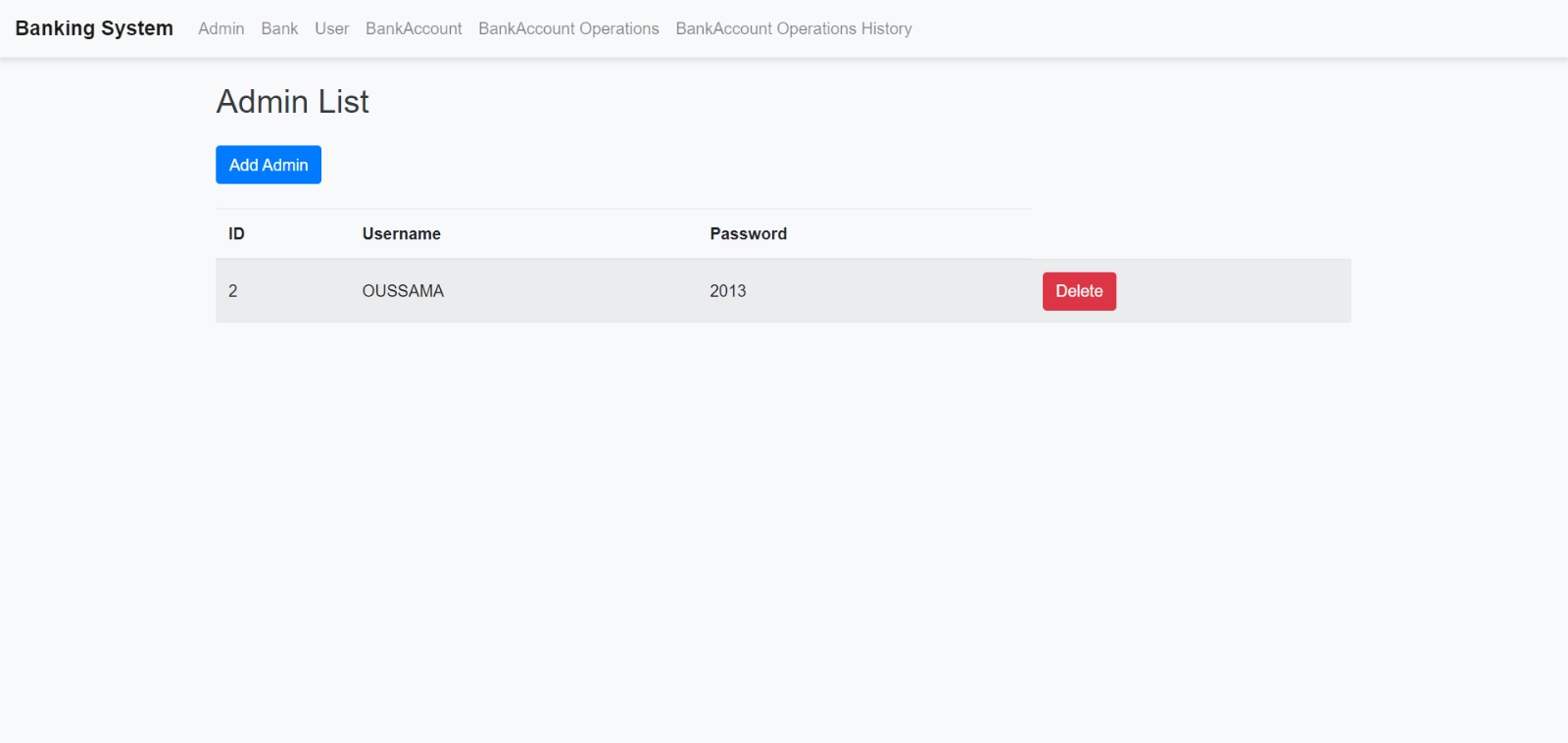


FIGURE 2:Formulaire d’ajout banque

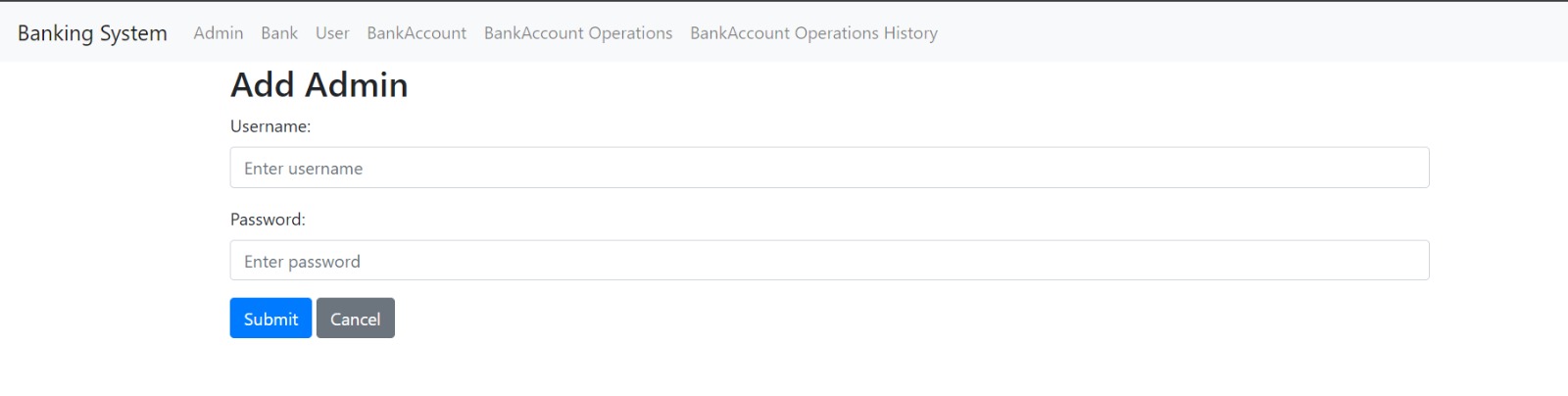


FIGURE 3: Liste des banques

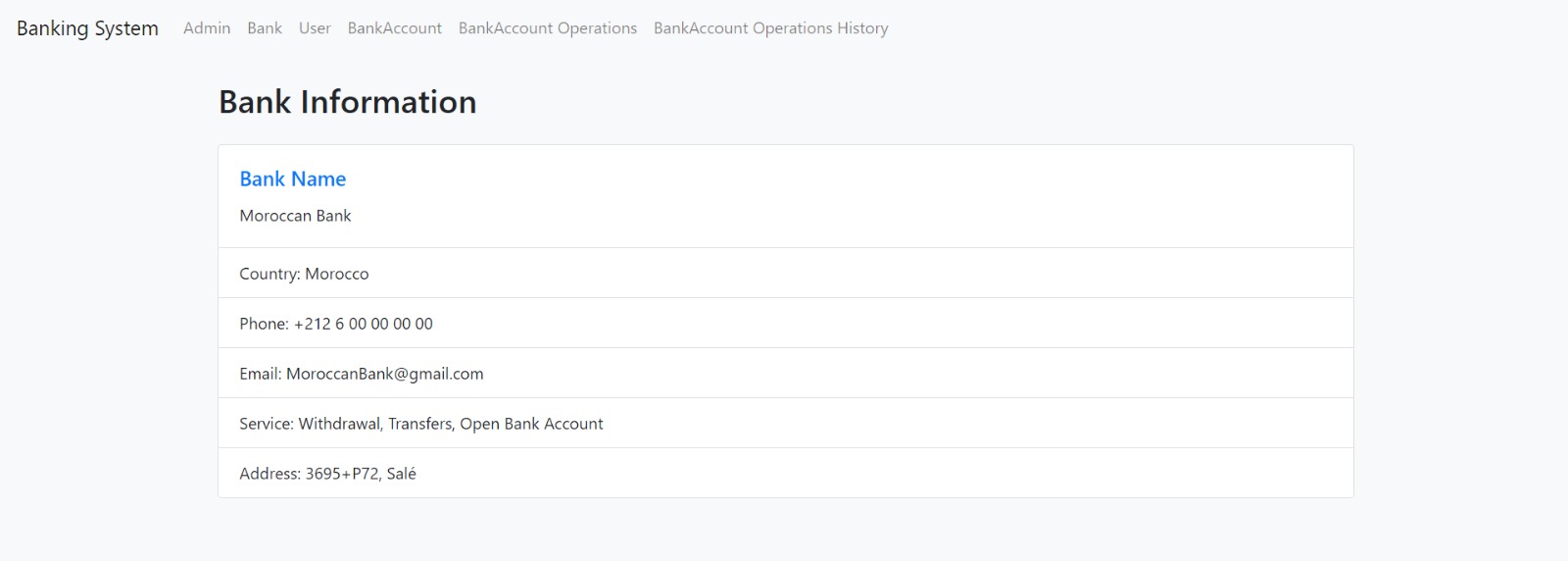


FIGURE 4: Page qui affiche les informations sur une banque

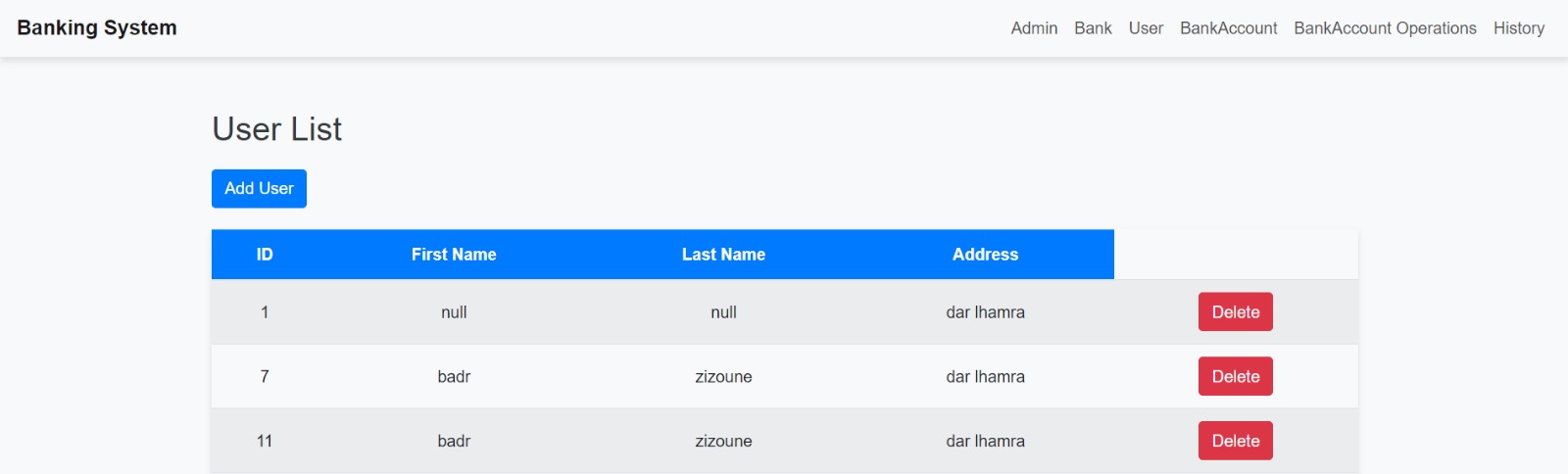


FIGURE 5: Liste des clients

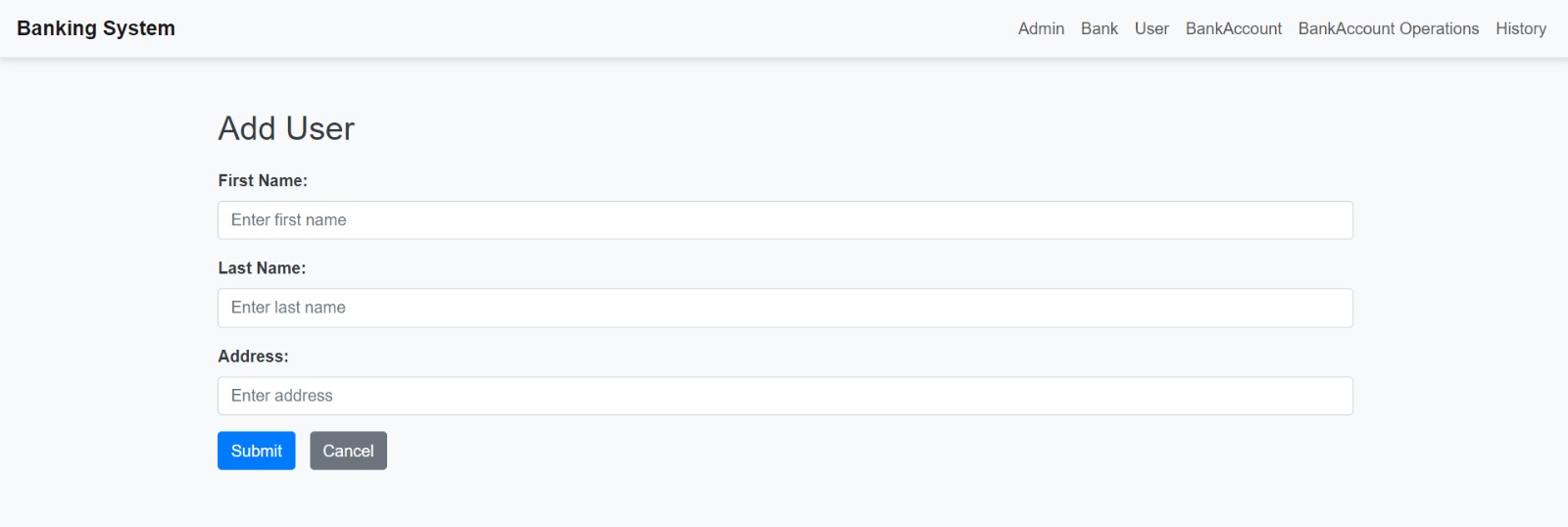


FIGURE 6: Formulaire des clients

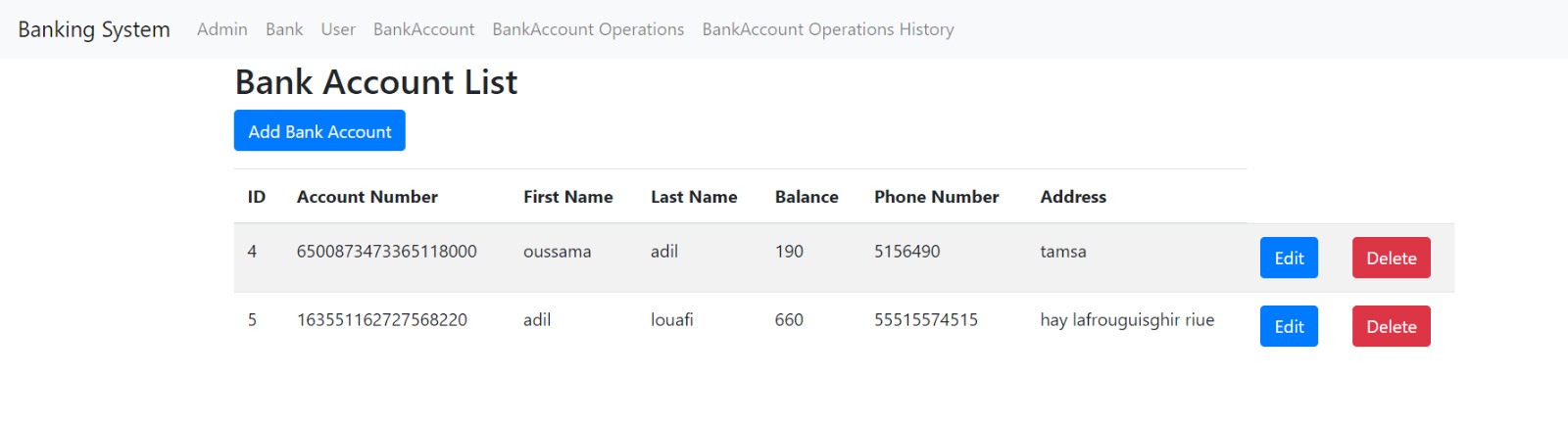


FIGURE 7: Page qui affiche les informations sur un client

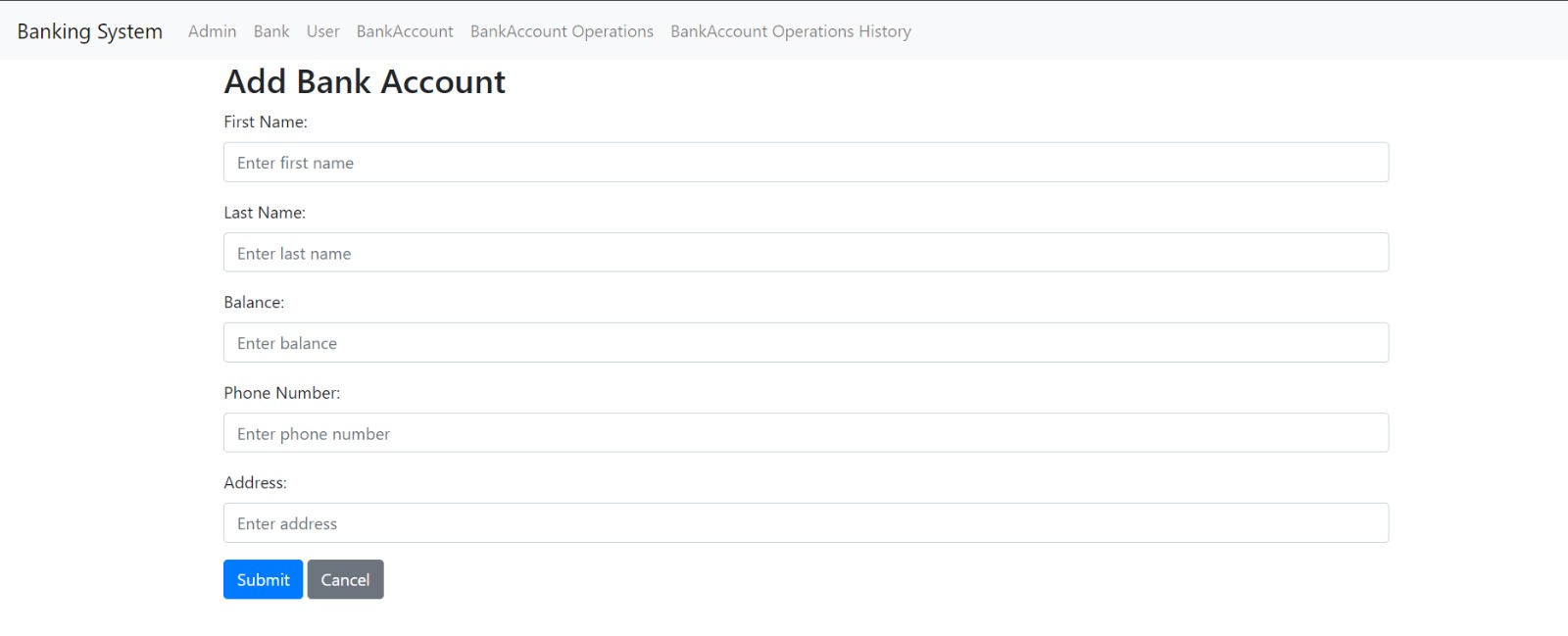


FIGURE 8: Formulaire add bank account

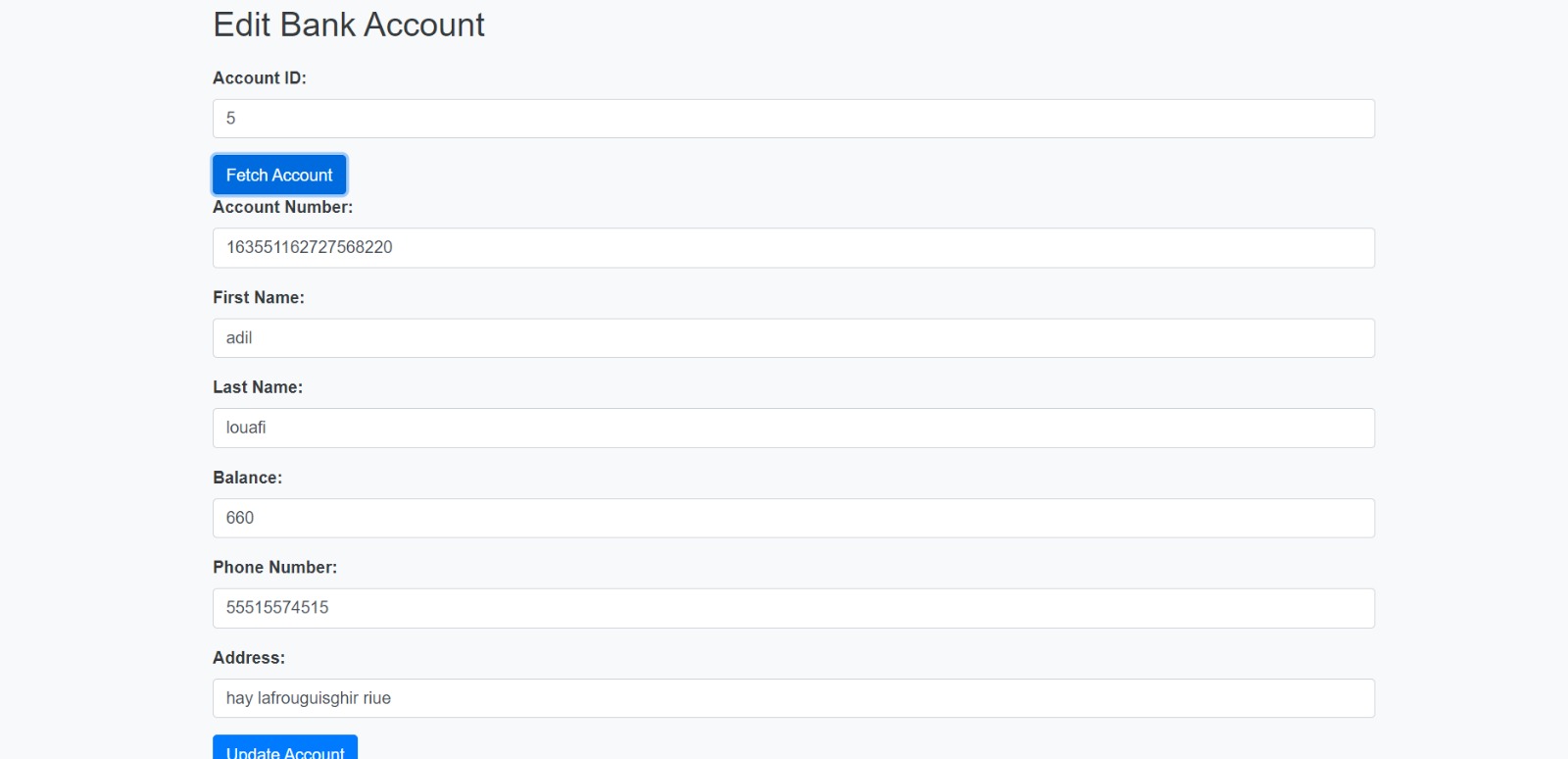


FIGURE 9: Formulaire edit bank account

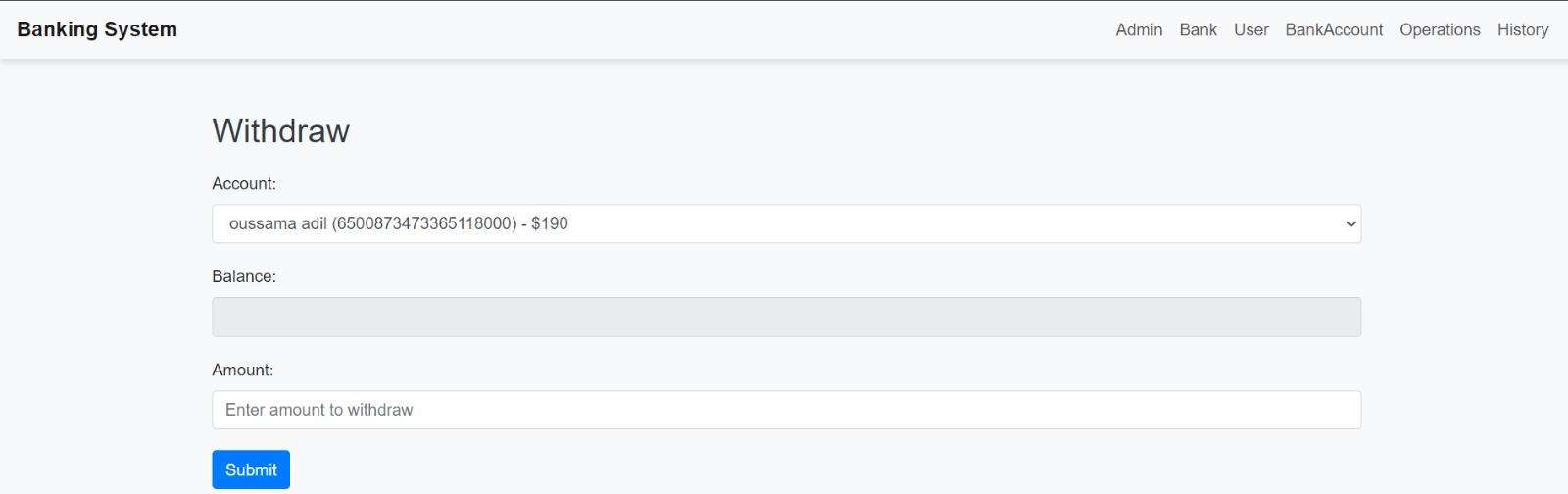


FIGURE 10 un retrait d'un compte bancaire

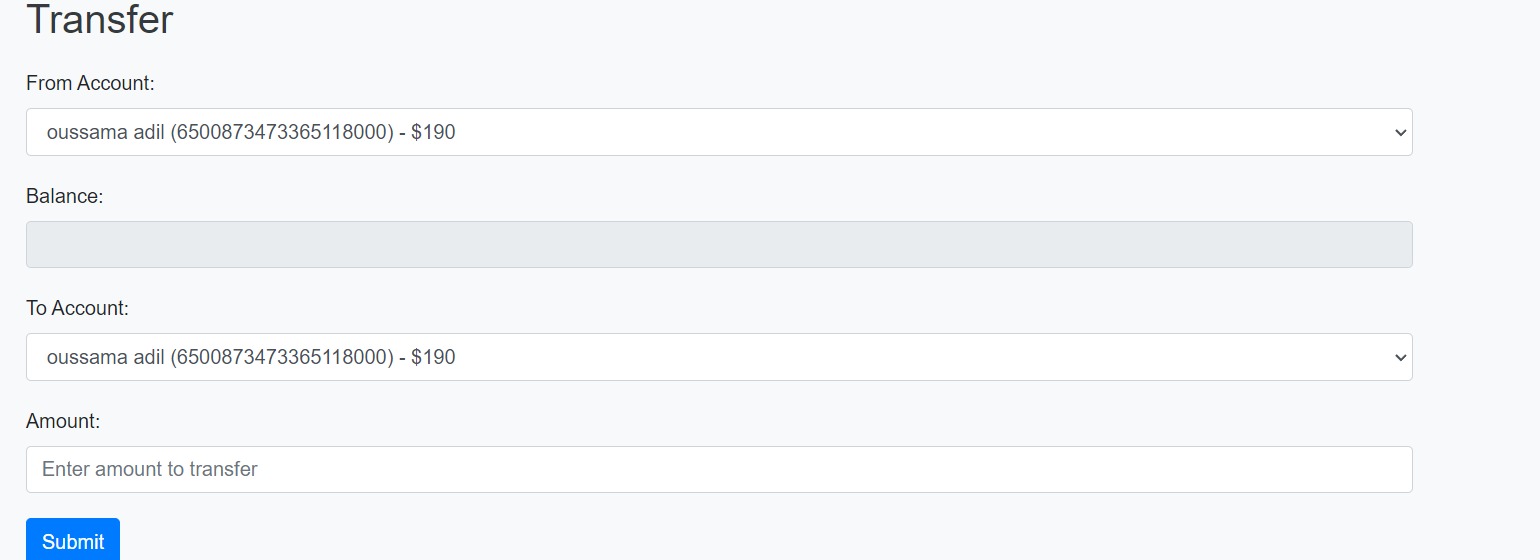


FIGURE 11 un transfert d'argent de compte a autre

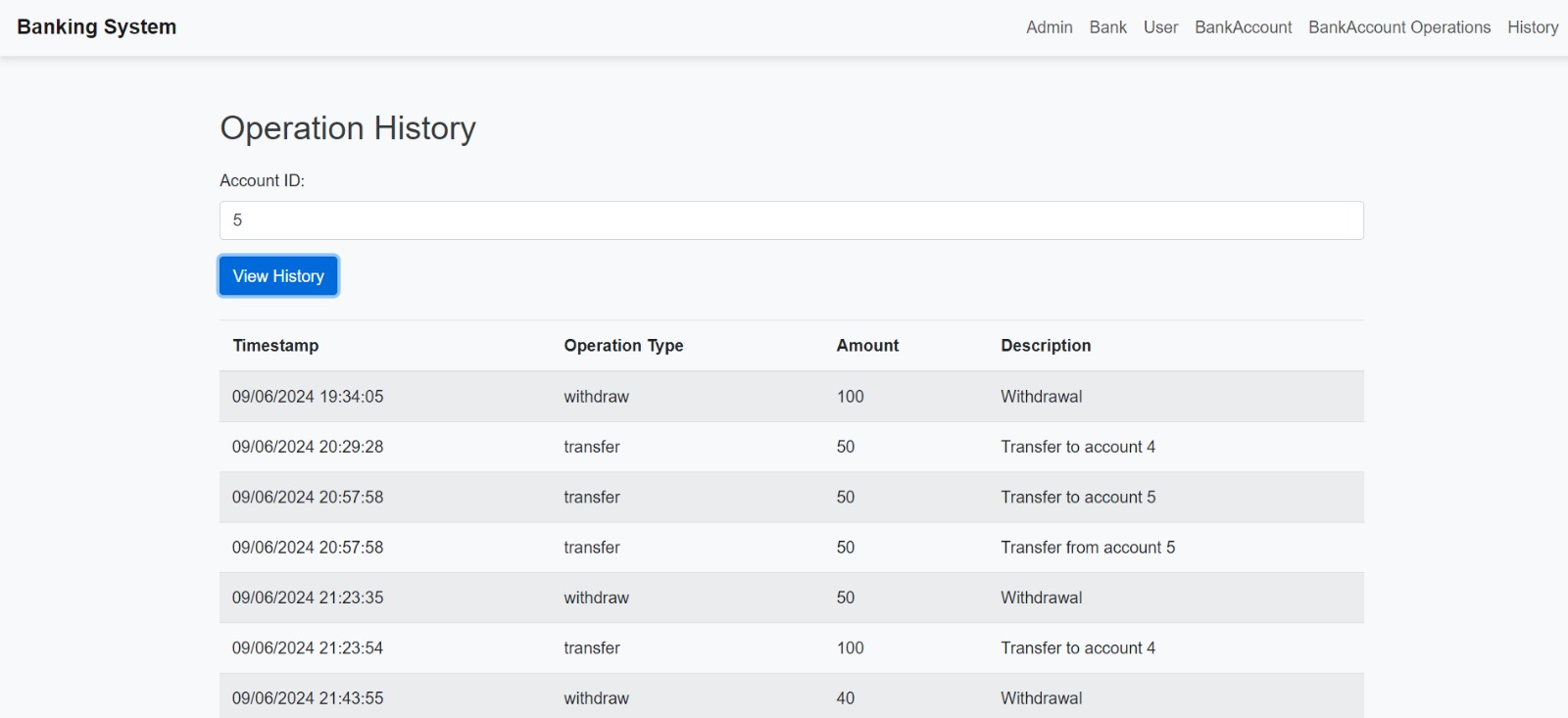


FIGURE 12 l'historique des transactions par id

Conclusion

La réalisation de ce projet a été une expérience extrêmement enrichissante, tant sur le plan pratique qu'académique. Ce projet m'a permis d'appliquer et de renforcer mes connaissances en développement web, tout en acquérant de nouvelles compétences et en relevant des défis techniques inédits.