

Centre de formation en informatique-Centre d'informatique  
et de recherche de l'armée et de la sécurité



# CONCEPTION ET REALISATION D'UNE APPLICATION DE TRANSFERT D'ARGENT

## RAPPORT DE PROJET

Binôme :

***NZILA Oscarvie Verdi***

***BAYONNEY YAVISHI Pierre-Destin***

Superviseur : ***OYERE Morel***

Matière : ***Oracle***

# SOMMAIRE

I. INTRODUCTION.....	3
II. CONCEPTION .....	5
1. Définition des concepts .....	5
a) Transfert d'argent .....	5
b) Sécurité des transactions .....	5
c) Interface utilisateur (UI) .....	5
d) Base de données.....	5
e) Backend et frontend.....	5
f) Architecture logicielle.....	5
g) Site web responsive.....	6
h) Bootstrap.....	6
2. Diagramme de cas d'utilisation.....	7
3. Diagramme de séquence .....	8
4. Diagramme de classe .....	9
5. Diagramme D'Activité .....	10
6. Déploiement.....	11
7. Modèle relationnel .....	12
8. Implémentation des fonctionnalités .....	13
a) Gestion des inscriptions .....	13
b) Authentification .....	13
c) Gestion des transactions.....	13
d) Gestion des notifications .....	14
e) Séquences et déclencheurs (triggers) dans les scripts SQL...	14
9. Difficultés rencontrées.....	15
III. REALISATION .....	16
1. Présentation des technologies Utilisées .....	16
2. Présentation du Code Source.....	17
3. Script SQL de la base de données .....	18
4. Capture de L'application.....	20
IV. CONCLUSION .....	22
V. BIBLIOGRAPHIE ET WEBOGRAPHIE .....	23

## I. INTRODUCTION

Dans le cadre de notre deuxième année de Licence en Informatique, avons entrepris un projet de fin d'année portant sur la conception et la réalisation d'une application de transfert d'argent intitulée TransMonnaie. Ce projet, mené en binôme, vise à mettre en pratique les compétences acquises au cours de notre formation, notamment en développement web, en bases de données, spécifiquement avec le Système de Gestion de Base de Données ORACLE (SGBDO), ainsi qu'en analyse et conception.

TransMonnaie est un site web de transfert d'argent conçu pour répondre aux besoins croissants de services de transfert d'argent rapides et sécurisés. Actuellement, TransMonnaie se concentre sur la gestion des transferts téléphoniques à partir du numéro de téléphone.

Le transfert d'argent est une activité essentielle dans le monde moderne, facilitant le soutien financier des familles, les paiements internationaux et les transactions quotidiennes. La nécessité de solutions de transfert d'argent efficaces et accessibles est amplifiée par la croissance de la globalisation et des migrations internationales. Les avancées technologiques ont permis de transformer ces processus, offrant des services plus rapides et plus sûrs grâce aux applications mobiles et aux plateformes en ligne. Dans ce contexte, TransMonnaie se positionne comme une solution potentielle pour répondre aux besoins des utilisateurs.

Pour valider les connaissances accumulées au cours de ces deux années au sein du CFI-CIRAS, nous avons été chargés de réaliser ce projet de fin d'année en binôme. Notre mission consiste à développer une solution informatisée sous forme d'application web, facilitant l'envoi d'argent entre deux ou plusieurs entités (particuliers).

Dans le cadre de ce projet de fin d'année, nous nous efforcerons de répondre à ces objectifs tout en appliquant les concepts théoriques appris au CFI-CIRAS. Nous adopterons une approche de conception orientée objet, en suivant la modélisation UML, et en utilisant diverses techniques web appropriées pour la réalisation de ce projet.

Ce rapport détaille les étapes que nous avons suivies pour réaliser cette application et se compose de trois parties distinctes.

Dans la première partie, nous définissons le contexte général de ce projet ainsi que les différents objectifs de l'application. Nous y exposons

également le plan suivi (méthodologie de résolution) et les difficultés rencontrées au cours de la réalisation.

La deuxième partie se concentre sur la phase d'analyse et de conception des données. Nous y détaillons la méthodologie adoptée pour la conception, présentons les acteurs, les différents diagrammes, et précisons les divers cas d'utilisation de l'application.

Enfin, dans la dernière partie, intitulée réalisation, nous exposons les divers outils de développement utilisés, ainsi que les principales interfaces graphiques créées, accompagnées de quelques scénarios applicatifs.

## II. CONCEPTION

### 1. Définition des concepts

#### *a) Transfert d'argent*

Le transfert d'argent est le processus par lequel un utilisateur peut envoyer des fonds à un autre utilisateur via une application ou un service en ligne. Les transferts peuvent être domestiques, c'est-à-dire à l'intérieur d'un même pays, ou internationaux, impliquant des conversions de devises et des frais supplémentaires.

#### *b) Sécurité des transactions*

La sécurité des transactions est essentielle pour protéger les données financières des utilisateurs et assurer la confiance dans le service. Cela inclut l'utilisation de technologies de cryptage pour protéger les informations sensibles, l'authentification multifactorielle pour vérifier l'identité des utilisateurs, et des systèmes de détection des fraudes pour identifier et prévenir les transactions suspectes.

#### *c) Interface utilisateur (UI)*

L'interface utilisateur est la partie visuelle de l'application avec laquelle les utilisateurs interagissent. Une bonne conception UI se concentre sur la simplicité et la clarté, permettant aux utilisateurs de naviguer facilement et d'effectuer des transactions sans confusion.

#### *d) Base de données*

La base de données Oracle utilisée pour TransMonnaie est une base de données relationnelle qui organise les données en tables interconnectées. Cela permet une gestion efficace et sécurisée des données transactionnelles et des informations utilisateur.

#### *e) Backend et frontend*

Le backend de l'application, développé en PHP et géré via XAMPP, traite la logique serveur, les transactions, et la communication avec la base de données. Le frontend, utilisant HTML, CSS et JavaScript, gère l'interaction utilisateur et la présentation des données.

#### *f) Architecture logicielle*

L'architecture de TransMonnaie est conçue pour être modulaire, permettant une maintenance et une évolutivité faciles. Les principaux composants incluent le frontend, le backend, la base de données, et les API d'intégration.

#### *g) Site web responsive*

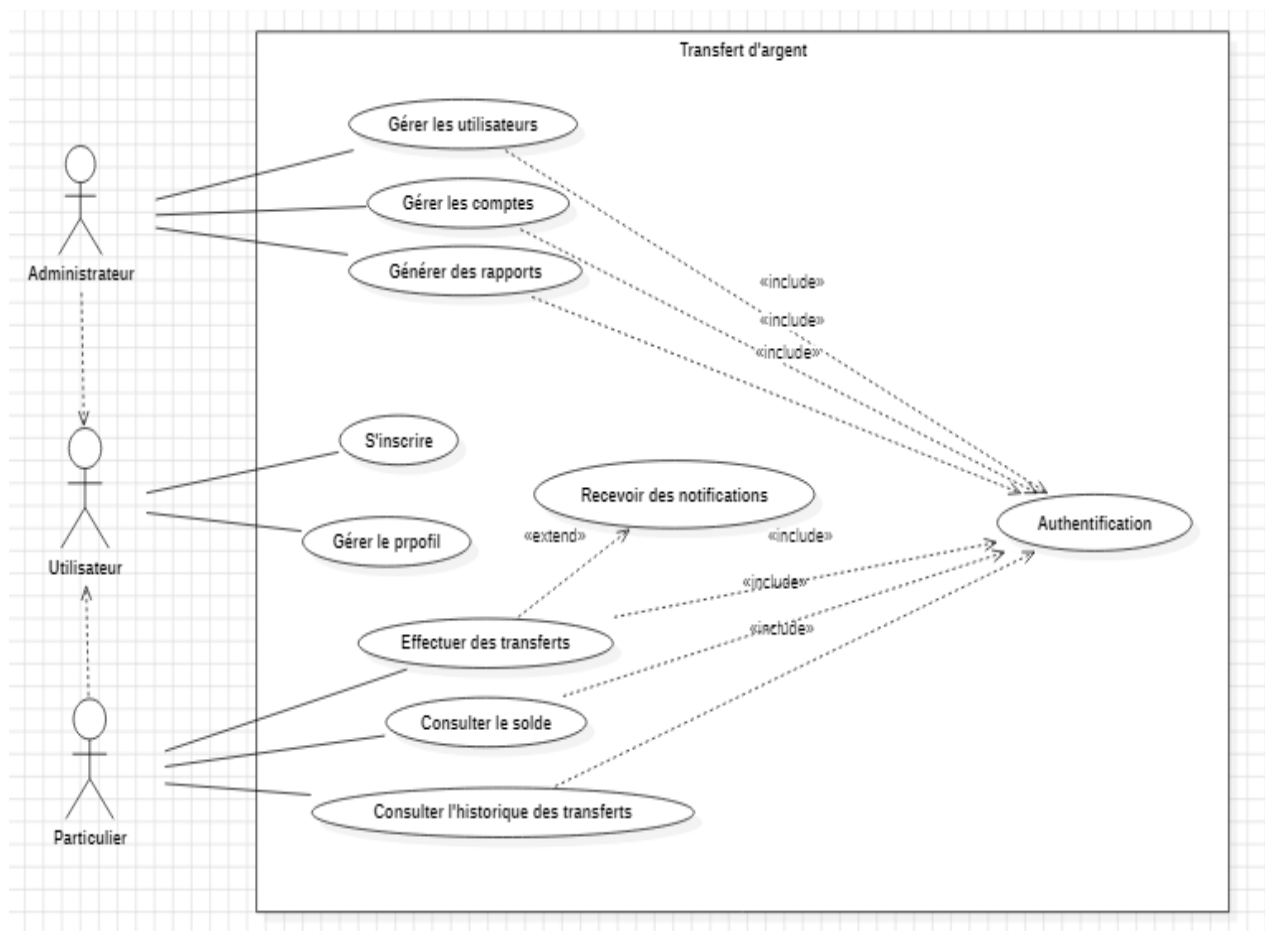
Un site web responsive, ou adaptatif, est un site web conçu pour offrir une expérience de navigation optimale quel que soit le type d'appareil utilisé (ordinateurs de bureau, tablettes, smartphones, etc.). Cela signifie que le site ajuste automatiquement sa mise en page, ses images, ses polices et ses autres éléments pour s'adapter à la taille et à la résolution de l'écran de l'utilisateur.

#### *h) Bootstrap*

Bootstrap est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. C'est l'un des Framework les plus populaires.

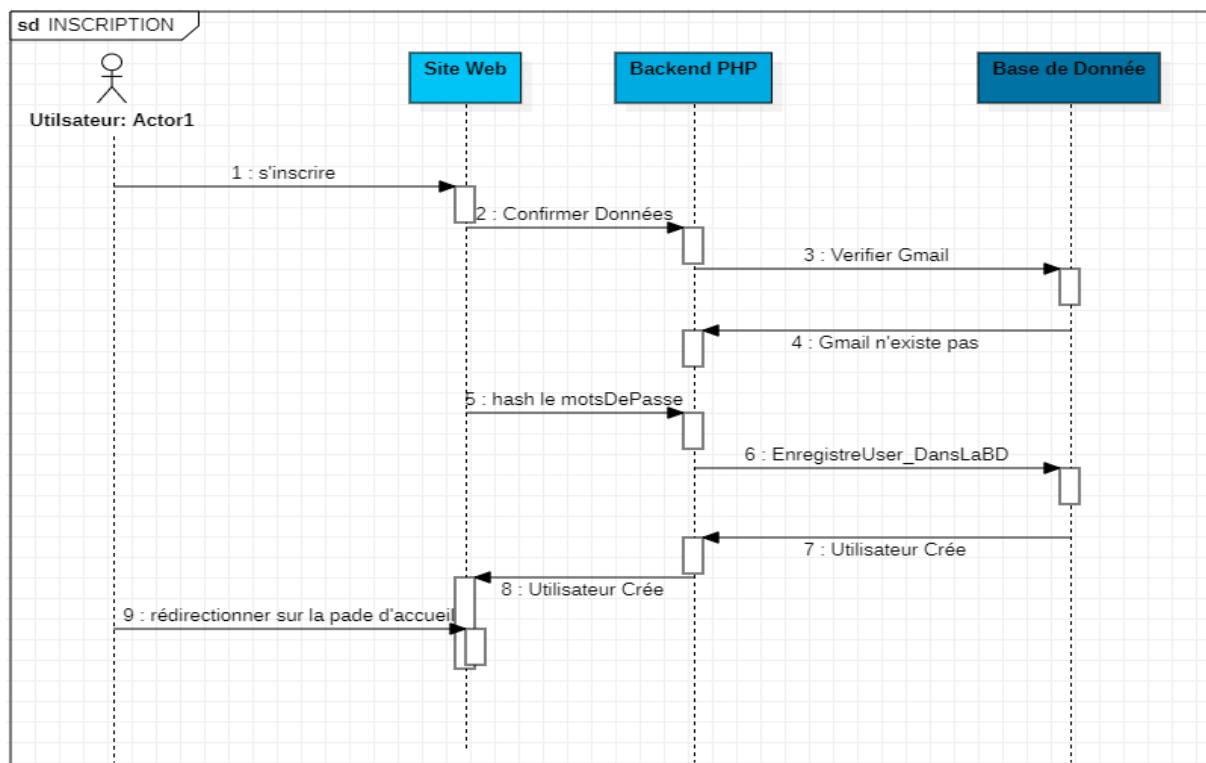
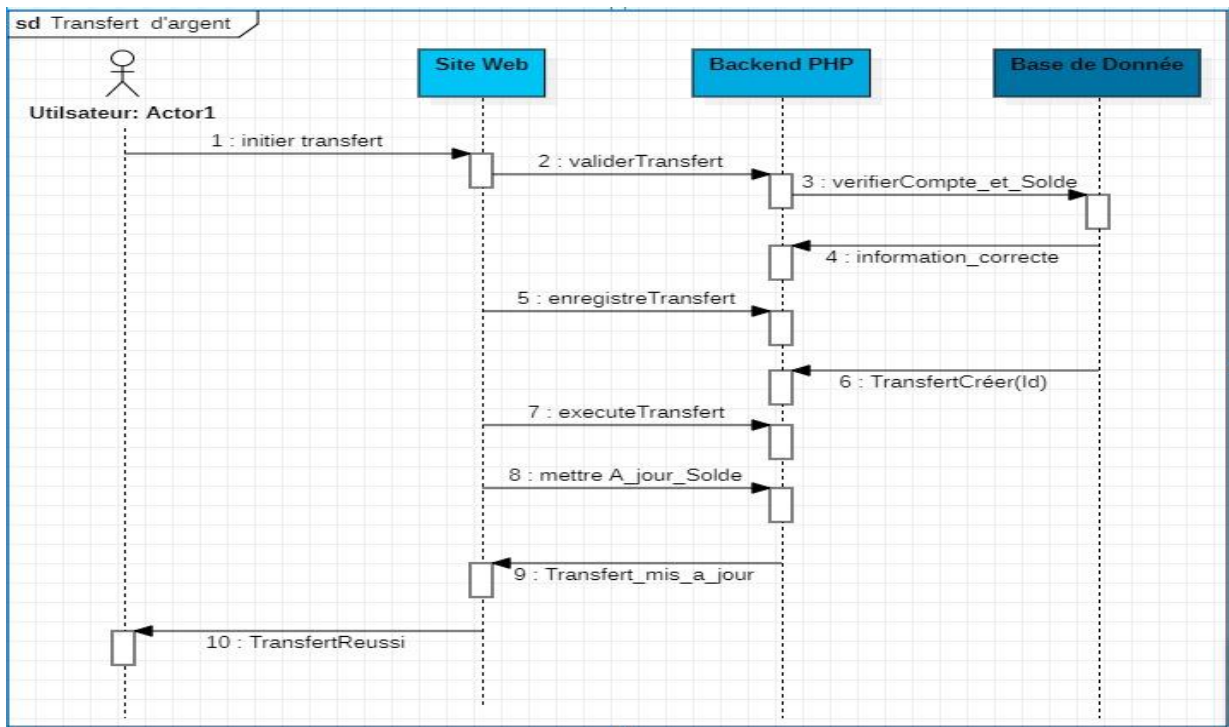
## 2. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation décrit les interactions entre les utilisateurs (particuliers, administrateurs) et le système. Il inclut des cas d'utilisation comme la création de compte, l'authentification, l'initiation et la validation de transferts, et la consultation de l'historique des transactions.



### 3. Diagramme de séquence

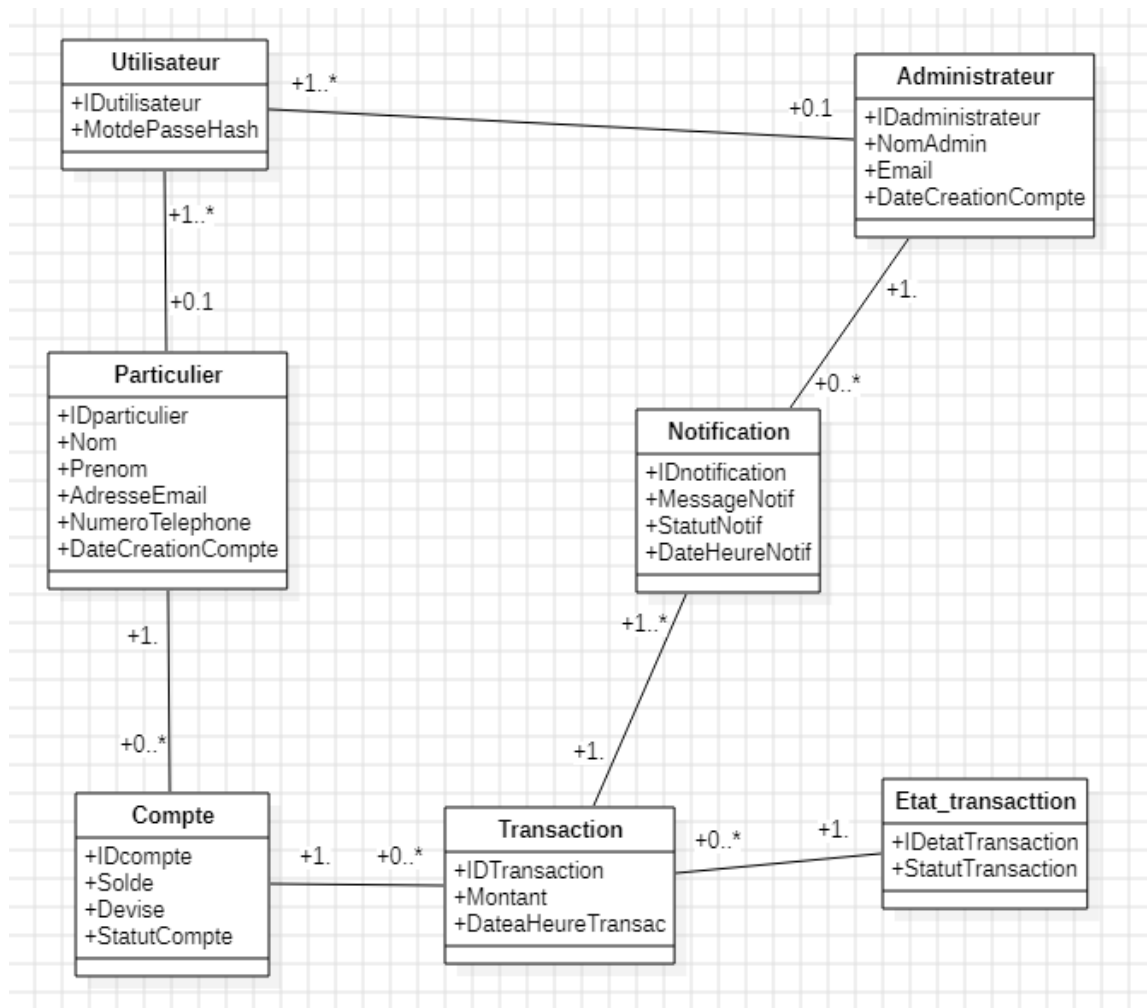
Le diagramme de séquence illustre l'interaction temporelle entre les différents composants du système lors de l'exécution d'un transfert d'argent ou d'une inscription. Il montre l'ordre des messages échangés entre l'utilisateur, le serveur web, la base de données.





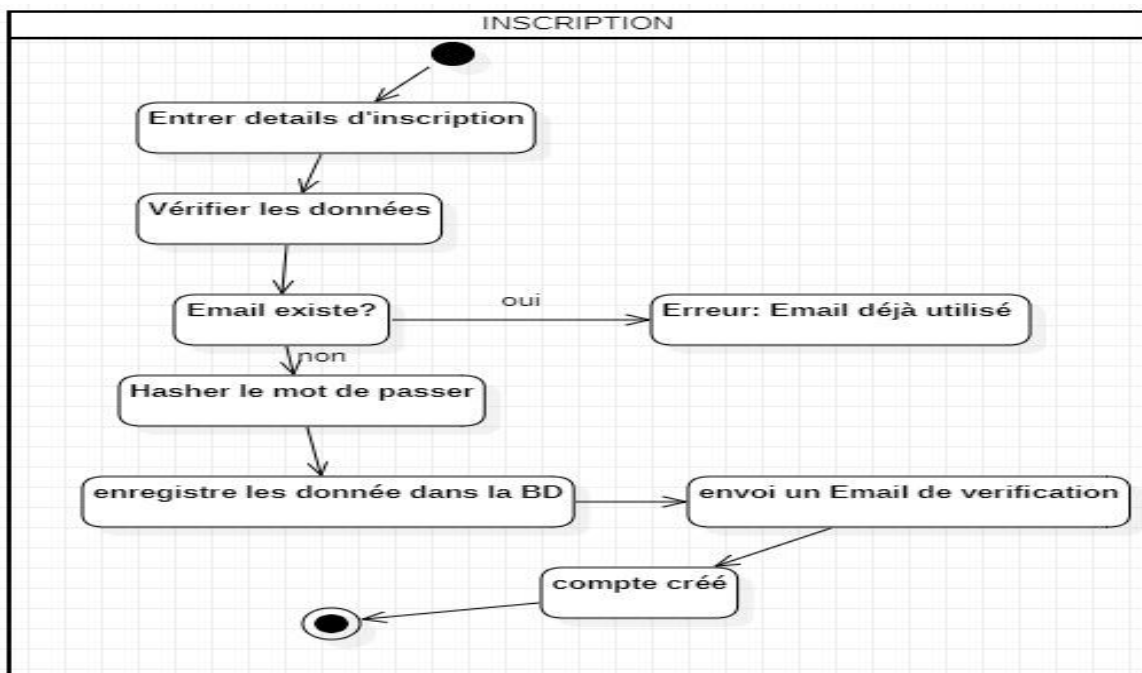
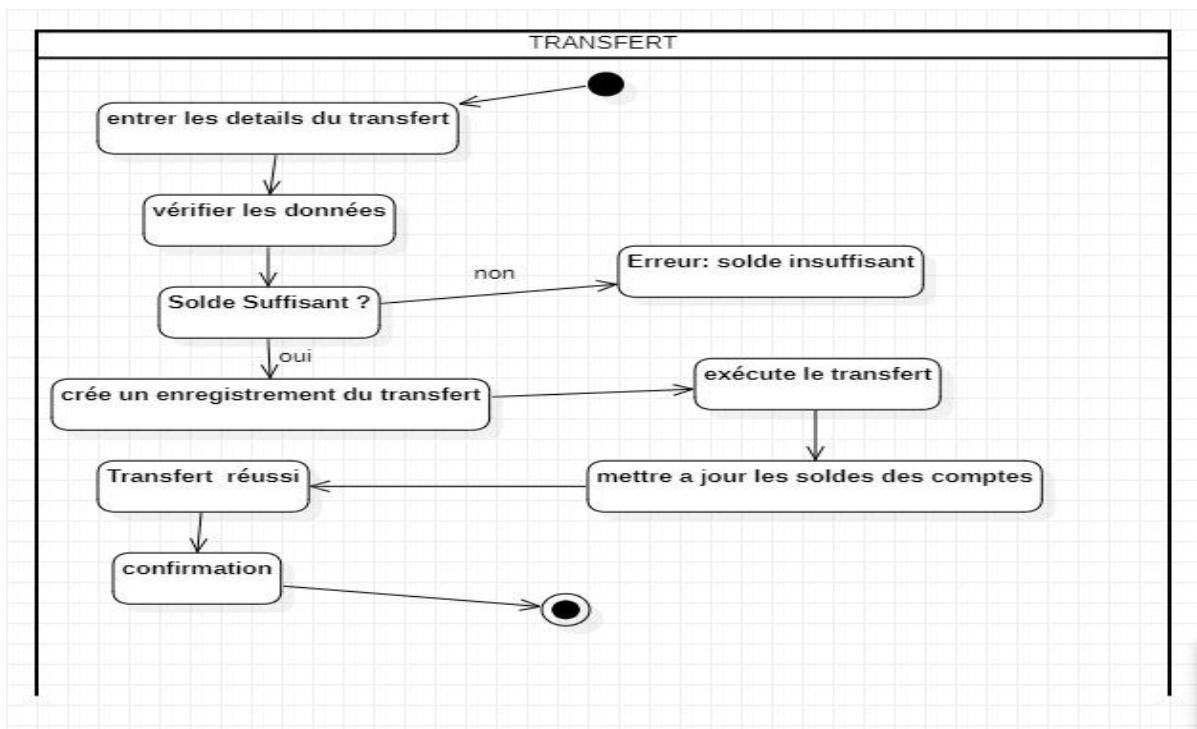
#### 4. Diagramme de classe

Un diagramme de classe est un type de diagramme structurel utilisé en génie logiciel pour décrire la structure d'un système en représentant ses classes, leurs attributs, leurs méthodes ainsi que les relations entre les objets. Ce diagramme est une composante clé de la modélisation orientée objet et de la méthode Unified Modeling Language (UML).



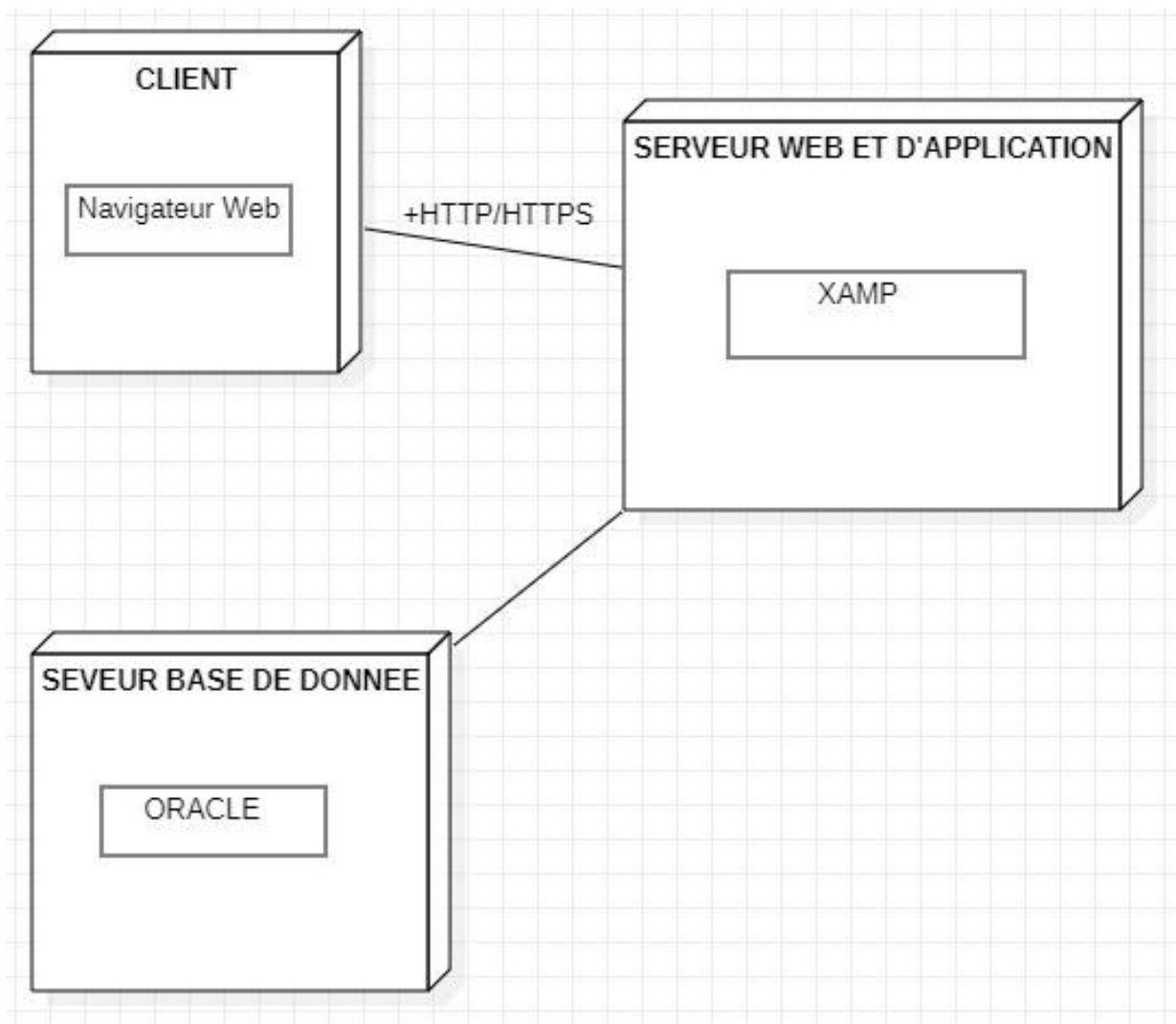
## 5. Diagramme D'Activité

Le diagramme d'activité décrit le flux de travail pour les processus principaux, tels que l'authentification utilisateur et le processus de transfert d'argent. Il montre les étapes séquentielles et les décisions impliquées.



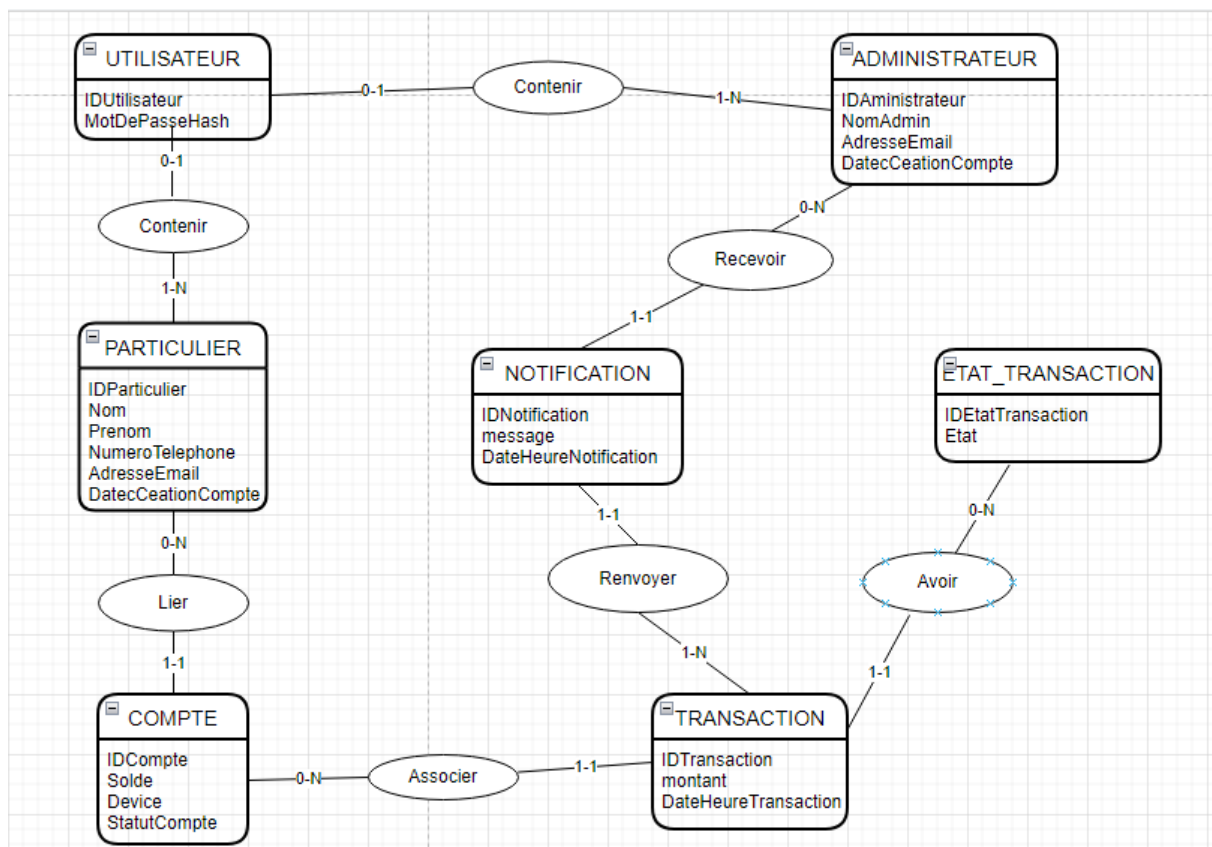
## 6. Déploiement

Le diagramme de déploiement illustre la distribution physique des composants logiciels sur les différents nœuds du réseau, y compris le serveur web, le serveur de base de données, et les dispositifs utilisateurs.



## 7. Modèle relationnel

Le modèle relationnel est un modèle de gestion de bases de données qui organise les données en utilisant des relations, souvent représentées sous forme de tables. Chaque table, aussi appelée relation, est composée de lignes (enregistrements ou tuples) et de colonnes (attributs ou champs). Le modèle relationnel repose sur l'algèbre relationnelle et la théorie des ensembles, permettant de gérer les données de manière structurée et cohérente.



## 8. Implémentation des fonctionnalités

L'implémentation des différentes fonctionnalités de notre application "TransMonnaie" a été un processus structuré et méthodique. Nous avons divisé le projet en plusieurs modules pour faciliter le développement et la gestion des différentes tâches. Voici un aperçu détaillé de l'implémentation des principales fonctionnalités :

### *a) Gestion des inscriptions*

Pour assurer un processus d'inscription sécurisé et efficace, nous avons mis en place les éléments suivants :

**Formulaire d'inscription :** Création d'un formulaire d'inscription avec des champs obligatoires pour les informations utilisateur, y compris nom, adresse e-mail, numéro de téléphone et mot de passe.

**Validation des données :** Implémentation de la validation côté client (JavaScript) et côté serveur (PHP) pour vérifier la conformité et la complétude des données saisies.

**Sécurisation des mots de passe :** Utilisation de la fonction PHP `password_hash()` pour hasher les mots de passe avant de les stocker dans la base de données Oracle.

### *b) Authentification*

Pour garantir un accès sécurisé à l'application, nous avons développé un système d'authentification robuste :

**Formulaire de connexion :** Création d'un formulaire de connexion avec des champs pour l'adresse e-mail et le mot de passe.

**Vérification des identifiants :** Comparaison des identifiants saisis avec les données stockées dans la base de données en utilisant la fonction PHP `password_verify()`.

**Gestion des sessions :** Utilisation des sessions PHP pour maintenir l'état de connexion de l'utilisateur et protéger les pages sécurisées.

### *c) Gestion des transactions*

Le cœur de notre application repose sur la gestion des transactions financières, incluant :

Formulaire de transaction : Développement d'un formulaire pour l'initiation des transferts d'argent, incluant des champs pour le montant, le bénéficiaire et le mode de transfert.

Validation et traitement des transactions : Implémentation de la logique métier pour vérifier la validité des transactions (solde suffisant, informations correctes, etc.) et les enregistrer dans la base de données.

Historique des transactions : Création d'une interface pour afficher l'historique des transactions de l'utilisateur, avec des options de filtrage et de tri.

#### *d) Gestion des notifications*

Pour informer les utilisateurs en temps réel, nous avons mis en place un système de notifications :

Envoi de notifications : Utilisation des notifications concernant les transactions, les mises à jour de compte, etc.

#### *e) Séquences et déclencheurs (triggers) dans les scripts SQL*

Pour automatiser certaines opérations et garantir l'intégrité des données, nous avons utilisé des séquences et des déclencheurs SQL :

Séquences : Création de séquences pour générer des identifiants uniques pour les enregistrements de transactions.

Déclencheurs : Développement de déclencheurs pour effectuer des opérations automatiques lors de l'insertion, de la mise à jour ou de la suppression de données dans la base de données, par exemple, pour mettre à jour les soldes des comptes après une transaction.

## 9. Difficultés rencontrées

Dans la conception de notre projet, nous avons été confrontés à plusieurs défis, notamment dans le développement du backend. Les principaux obstacles rencontrés incluent :

**Gestion des inscriptions :** Assurer un processus d'enregistrement utilisateur sécurisé et efficace, intégrant la vérification des données et la prévention des inscriptions frauduleuses.

**Authentification :** Mettre en place un système d'authentification robuste, garantissant la protection des informations d'identification et la gestion des sessions utilisateur.

**Gestion des transactions :** Développer un système fiable pour le traitement des transactions financières, incluant la vérification, le suivi et l'enregistrement des opérations de transfert d'argent.

**Gestion des notifications :** Implémenter un mécanisme de notifications permettant d'informer les utilisateurs en temps réel sur l'état de leurs transactions et autres activités pertinentes.

**Séquences et déclencheurs (triggers) dans les scripts SQL :** Concevoir et intégrer des séquences et des triggers efficaces dans notre base de données pour automatiser certaines opérations et garantir l'intégrité des données.

### III. REALISATION

#### 1. Présentation des technologies Utilisées

Dans le cadre de notre projet "TransMonnaie", nous avons employé une variété de technologies modernes et éprouvées pour assurer la robustesse, la sécurité et la facilité d'utilisation de l'application. Voici un aperçu des principales technologies utilisées :

- **Langages de Programmation**

**PHP** : Utilisé pour le développement du backend, PHP est un langage de script côté serveur qui permet de créer des applications web dynamiques et interactives. Il a été choisi pour sa flexibilité, sa facilité d'intégration avec les bases de données et sa large communauté de développeurs.

**JavaScript** : Utilisé pour le développement frontend, JavaScript permet de rendre l'interface utilisateur interactive et réactive.

- **Framework**

**Bootstrap** : Un Framework CSS, Bootstrap a été utilisé pour concevoir des interfaces utilisateur modernes, réactives et attrayantes. Il facilite le développement rapide de designs cohérents et adaptatifs grâce à ses composants prédéfinis et ses grilles flexibles.

- **Bases de Données**

**Oracle Database** : Pour la gestion des données, nous avons utilisé Oracle Database, un système de gestion de bases de données relationnelles (SGBDR) robuste et performant. Oracle offre des fonctionnalités avancées pour la gestion des transactions, la sécurité des données et l'intégrité référentielle.

- **Outils de Développement**

**Visual Studio Code** : Un éditeur de code source puissant et extensible, Visual Studio Code a été notre principal environnement de développement intégré (IDE). Il supporte une multitude de langages et de Framework grâce à ses extensions.



- Serveurs

Apache : Utilisé comme serveur web, Apache a servi de plateforme pour héberger notre application PHP et gérer les requêtes HTTP des utilisateurs.

## 2. Présentation du Code Source

```
TransMon > index.php
1  <?php
2  include 'header.php';
3
4  $is_logged_in = isset($_SESSION['utilisateur']);
5  ?>
6
7  <div class="hero-section py-5">
8      <div class="container">
9          <div class="row align-items-center">
10             <div class="col-md-6 order-md-2">
11                 
12             </div>
13             <div class="col-md-6 order-md-1">
14                 <h2>Bienvenue sur TransMonnaie</h2>
15                 <p>TransMonnaie est votre solution de transfert d'argent de confiance. Envoyez de l'argent
16                 rapidement et en toute sécurité à vos proches, où qu'ils soient.</p>
17                 <?php if ($is_logged_in): ?>
18                     <p>Bonjour, <?php echo htmlspecialchars($_SESSION['utilisateur']['Prenom']) . ' '
19                     . htmlspecialchars($_SESSION['utilisateur']['Nom']); ?> !</p>
20                     <p>Numéro de compte : <?php echo htmlspecialchars($_SESSION['utilisateur']['IDCompte']); ?></p>
21                     <p>Solde : <?php echo htmlspecialchars($_SESSION['utilisateur']['solde']); ?> FCFA</p>
22                 <?php else: ?>
23                     <a href="login.php" class="btn btn-primary">Connexion</a>
24                     <a href="inscription.php" class="btn btn-outline-primary">Créer un compte</a>
25                 <?php endif; ?>
26             </div>
27         </div>
28     </div>
29 </div>
30
```

```
TransMon > header.php
1  <?php
2  include 'configuration/infoBD.php';
3  session_start();
4
5  if (isset($_SESSION['utilisateur'])) {
6      $utilisateur = $_SESSION['utilisateur'];
7      $conn = new PDO("oci:dbname=$DB_Host", $DB_User, $DB_Pass);
8      $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
9
10     $sqlNotifications = "SELECT COUNT(*) AS nbr
11     FROM Notification
12     WHERE IDParticulier = (SELECT IDParticulier FROM Utilisateur WHERE Email = :email)
13     AND StatutNotif = 'non lue'";
14     $stmtNotifications = $conn->prepare($sqlNotifications);
15     $stmtNotifications->bindParam(':email', $email, PDO::PARAM_STR);
16     $stmtNotifications->execute();
17     $rowNotifications = $stmtNotifications->fetch(PDO::FETCH_ASSOC);
18     $unread_count = $rowNotifications['nbr'];
19
20     // Fermer les connexions
21     $stmtNotifications->closeCursor();
22     $stmt->closeCursor();
23 }
24 ?>
25
```

```

<?php if (isset($unread_count) && $unread_count > 0): ?>
    <span class="badge"><?php echo $unread_count; ?></span>
<?php endif; ?>
</a>
<?php if (isset($utilisateurDetails)): ?>
    <div class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" id="monCompteDropdown" role="button" data-bs-toggle="dropdown"
            aria-expanded="false">
            <i class="bi bi-person icon_user"></i>
        </a>
        <ul class="dropdown-menu dropdown-menu-end" aria-labelledby="monCompteDropdown">
            <li class="dropdown-item">
                <div class="d-flex flex-column align-items-start">
                    <div class="mb-2">
                        <strong>Utilisateur :</strong> <?php echo htmlspecialchars($utilisateurDetails['Prenom']) . ' '
                            . htmlspecialchars($utilisateurDetails['Nom']); ?>
                    </div>
                    <div class="mb-2">
                        <strong>Compte :</strong> <?php echo htmlspecialchars($utilisateurDetails['IDCompte']); ?><br>
                        <span class="text-muted">Solde :</span> <strong>
                            <?php echo number_format($utilisateurDetails['Solde'], 2, ',', ' '); ?> ECFA</strong>
                        </div>
                    </div>
                </li>
            </li>
        </ul>
    </div>
</div>

```

### 3. Script SQL de la base de données

```

58 CREATE TABLE Notification (
59     IDNotification NUMBER PRIMARY KEY,
60     IDTransaction VARCHAR2(8),
61     IDAdministrateur VARCHAR2(4),
62     MessageNotif CLOB,
63     DateHeureNotif TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
64     StatutNotif VARCHAR2(10) DEFAULT 'non lue',
65     CONSTRAINT FK_Notification_Transaction FOREIGN KEY (IDTransaction) REFERENCES Transaction(IDTransaction) ON DELETE CASCADE,
66     CONSTRAINT FK_Admin_notif FOREIGN KEY (IDAdministrateur) REFERENCES Administrateur(IDAdministrateur) ON DELETE CASCADE
67 );
68
69
70 -- Création des Séquences
71
72 CREATE SEQUENCE Administrateur_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
73 CREATE SEQUENCE Particulier_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
74 CREATE SEQUENCE Utilisateur_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
75 CREATE SEQUENCE Compte_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
76 CREATE SEQUENCE EtatTransaction_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
77 CREATE SEQUENCE Transaction_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
78 CREATE SEQUENCE Notification_SEQ START WITH 1 INCREMENT BY 1 NOCACHE NOCYCLE;
79
80
81 -- Création des Triggers
82

```

```

29 CREATE TABLE Compte (
30     IDCompte VARCHAR2(20) PRIMARY KEY,
31     Solde NUMBER(10,2) DEFAULT 0.00,
32     Devise VARCHAR2(10) NOT NULL,
33     StatutCompte VARCHAR2(10) DEFAULT 'actif',
34     IDParticulier VARCHAR2(4),
35     CONSTRAINT FK_Compte_Partulier FOREIGN KEY (IDParticulier) REFERENCES Particulier(IDParticulier) ON DELETE CASCADE,
36     CONSTRAINT CHK_Solde_Positif CHECK (Solde >= 0)
37 );
38
39 CREATE TABLE EtatTransaction (
40     IDEtatTransaction NUMBER PRIMARY KEY,
41     StatutTransaction VARCHAR2(10) DEFAULT 'en attente'
42 );
43
44 CREATE TABLE Transaction (
45     IDTransaction VARCHAR2(8) PRIMARY KEY,
46     IDExpéditeur VARCHAR2(20),
47     IDDestinataire VARCHAR2(20),
48     Montant NUMBER(10,2) NOT NULL,
49     DateHeureTransaction TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
50     IDEtatTransaction NUMBER,
51     CONSTRAINT FK_Expe_Compte FOREIGN KEY (IDExpéditeur) REFERENCES Compte(IDCompte) ON DELETE CASCADE,
52     CONSTRAINT FK_Desti_Compte FOREIGN KEY (IDDestinataire) REFERENCES Compte(IDCompte) ON DELETE CASCADE,
53     CONSTRAINT FK_EtatTransac FOREIGN KEY (IDEtatTransaction) REFERENCES EtatTransaction(IDEtatTransaction) ON DELETE CASCADE,
54     CONSTRAINT CHK_Montant_Positif CHECK (Montant >= 500),
55     CONSTRAINT CHK_Expe_Desti_Diff CHECK (IDExpéditeur <> IDDestinataire)
56 );

```


Transaction > SQL.sql

```

1
2  ⚡ Création des Tables
3
4  CREATE TABLE Particulier (
5      IDParticulier VARCHAR2(4) PRIMARY KEY,
6      Nom VARCHAR2(50) NOT NULL,
7      Prenom VARCHAR2(50) NOT NULL,
8      AdresseEmail VARCHAR2(100) UNIQUE NOT NULL,
9      NumeroTelephone VARCHAR2(20) NOT NULL,
10     DateCreationCompte TIMESTAMP DEFAULT CURRENT_TIMESTAMP
11 );
12
13 CREATE TABLE Administrateur (
14     IDAdministrateur VARCHAR2(4) PRIMARY KEY,
15     NomAdmin VARCHAR2(50) NOT NULL,
16     Email VARCHAR2(100) UNIQUE NOT NULL,
17     DateCreationCompte TIMESTAMP DEFAULT CURRENT_TIMESTAMP
18 );
19
20 CREATE TABLE Utilisateur (
21     IDUser NUMBER PRIMARY KEY,
22     IDParticulier VARCHAR2(4),
23     IDAdministrateur VARCHAR2(4),
24     MotDePasseHash VARCHAR2(255) NOT NULL,
25     CONSTRAINT FK_Part_User FOREIGN KEY (IDParticulier) REFERENCES Particulier(IDParticulier) ON DELETE CASCADE,
26     CONSTRAINT FK_Admin_User FOREIGN KEY (IDAdministrateur) REFERENCES Administrateur(IDAdministrateur) ON DELETE CASCADE
27 );
28

```

#### 4. Capture de L'application

 **Transmonnaie**



## Connexion

[Mot de passe oublié ?](#)

Connexion

ou

Créer un compte

 **TransMonnaie** 

Connexion



## Bienvenue sur TransMonnaie

TransMonnaie est votre solution de transfert d'argent de confiance. Envoyez de l'argent rapidement et en toute sécurité à vos proches, où qu'ils soient.

Connexion

Créer un compte


**TransMonnaie**
Connexion



**ENTREPRISE**

- [Blog](#)
- [À propos](#)
- [Devenez agent](#)
- [Carrières](#)

**À PROPOS DE NOUS**

- [Politique de confidentialité](#)
- [Conditions générales](#)
- [Droits du consommateur](#)
- [Sécurité & fraude](#)

**NOUS SUIVRE**








**Transmonnaie**
5




## Bienvenue sur TransMonnaie

TransMonnaie est votre solution de transfert d'argent de confiance. Envoyez de l'argent rapidement et en toute sécurité à vos proches, où qu'ils soient.


**Transmonnaie**
5


## Bienvenue sur Transmonnaie

Transmonnaie est votre solution de transfert d'argent de confiance. Envoyez de l'argent rapidement et en toute sécurité à vos proches, où qu'ils soient.



**Utilisateur : Jean Dupont**  
**Compte 1 : 066666666**  
**Somme : 1.500Fcfa**

- Transaction
- Tableau de bord
- Profil
- Historique des Transactions
- Paramètres
- Déconnexion

**Pourquoi choisir Transmonnaie ?**

**Transmonnaie assure la satisfaction totale de nos clients**

## IV. CONCLUSION

La réalisation de ce projet de fin d'année s'est avérée extrêmement enrichissante, nous permettant de consolider et d'approfondir nos connaissances en programmation et en conception de systèmes. Grâce à ce projet, nous avons non seulement pu renforcer nos compétences en développement web avec PHP et le Framework Bootstrap, mais aussi perfectionner notre compréhension de la gestion de projets informatiques.

Ce projet nous a offert une opportunité précieuse de travailler dans un contexte réel, nous confrontant à des problématiques authentiques et complexes. La gestion des inscriptions, l'authentification des utilisateurs, la manipulation des transactions financières, la gestion des notifications et la mise en œuvre de séquences et de déclencheurs (triggers) dans les scripts SQL ont été autant de défis que nous avons relevés. Cependant, certains défis, n'ont pas pu être entièrement surmontés.

Le projet "TransMonnaie" a pour vocation de répondre aux besoins croissants de services de transfert d'argent rapides et sécurisés, particulièrement adaptés aux utilisateurs. En nous concentrant sur la gestion des transferts téléphoniques via des numéros de téléphone, nous avons développé une solution informatique, capable de transformer les processus de transfert d'argent grâce à l'utilisation de technologies avancées.

Enfin, ce projet reste ouvert à des améliorations futures. Il offre une base solide que d'autres développeurs pourront étendre, enrichir et adapter aux besoins émergents. De nouvelles fonctionnalités peuvent être ajoutées pour répondre aux évolutions du marché et des technologies, garantissant ainsi la pérennité et la pertinence de l'application.

Ainsi, ce projet a non seulement été un exercice académique, mais aussi une expérience formatrice qui nous a préparés à affronter les défis réels du monde professionnel. Nous sommes désormais mieux équipés pour contribuer de manière significative à des projets informatiques complexes et innovants.

- Documentation PHP
  - [PHP Documentation](<https://www.php.net/docs.php>)
- Documentation Bootstrap
  - [W3Schools Bootstrap](<http://www.w3schools.com/bootstrap/>)
  - [MDBBootstrap Documentation](<https://mdbootstrap.com/docs/>)
- Documentation Oracle SQL
  - [Oracle Database Documentation](<https://docs.oracle.com/en/database/>)
  - [Oracle SQL Developer User's Guide](<https://docs.oracle.com/en/database/oracle/sql-developer/>)
- Forums & Astuces
  - [Stack Overflow](<http://stackoverflow.com>)
  - [PHP Freaks](<https://www.phpfreaks.com/>)
  - [Oracle Technology Network Community](<https://community.oracle.com/>)
- Cours en ligne
  - [Dynamisez vos sites web avec JavaScript](<https://openclassrooms.com/courses/dynamisez-vos-sites-web-avec-javascript>)
  - [Simplifiez vos développements JavaScript avec jQuery](<https://openclassrooms.com/courses/simplifiez-vos-developpements-javascript-avec-jquery>)
  - [Learn PHP from Scratch](<https://www.codecademy.com/learn/learn-php>)
  - [PHP for Beginners - Become a PHP Master](<https://www.udemy.com/course/php-for-beginners/>)
  - [Bootstrap 4 from Scratch](<https://www.udemy.com/course/bootstrap-4-from-scratch-with-5-projects/>)
  - [The Complete Oracle SQL Certification Course](<https://www.udemy.com/course/the-complete-oracle-sql-certification-course/>)
- Tutoriels
- Wikipédia