

Document d'architecture technique : Blue Print Level 3

Sommaire

Informations	2
Contrôle de version	2
Historique des approbations	2
1. SOLUTION DESCRIPTION	3
1.1. GESTION RÉSUMÉ	3
1.2. ÉTAPES D'INTEGRATION AVEC PI	4
1.3. GLOSSAIRE	7
1.4. DOCUMENTS CONNEXES	9
1.5. Objectifs et restrictions de l'architecture	9
1.6. Spécifications fonctionnelles	11
1.6.1. Gestion des Alias :	11
1.6.2. Transferts P2P :	11
1.6.3. Paievements Marchands :	12
1.6.4. Test de Connectivité vers PI :	12
1.6.5. Sponsorisation de Participant Indirect :	12
1.6.6. Gestion de Compte des Participants :	13
1.7. Caractéristiques qualitatives	13
1.8. VUE DES CAS D'UTILISATION	14
1.9. SURVEILLANCE EXIGENCES	19
1.9.1. SURVEILLANCE DU SYSTÈME	19
1.9.2. EXIGENCES DU SYSTÈME	19
2. ARCHITECTURE DE LA SOLUTION	20
2.1. Vue logique	20
2.1.1. Architecture	20
2.1.2. Les composants du système.	21
2.1.3. Les dépendances externes	22
2.2. Diagrammes d'interaction	23
2.2.1. SEQ-001 : Création d'Alias	23
2.2.2. SEQ-002 : Mise à jour d'Alias	24
2.2.3. SEQ-003: Suppression d'Alias	25
2.2.4. SEQ-004: Recherche d'Alias	26
2.2.5. SEQ-005 : Transfert P2P Demande de transfert vers PI vers un autre participant.	27
2.2.6. SEQ-006 : Envoyer une demande de vérification d'identité	29
2.2.7. SEQ-007 : Envoyer une réponse à une demande de vérification d'identité	29



Projet : Intégration du système interopérable de la BCEAO avec FreeMoney.

Informations

Document Author:	Pape Samba NDOUR
Owner while current:	FREE SENEGAL

Contrôle de version

Document Author	Date	Author	Description
1.0.0	Pape Samba NDOUR	2023-11-27	Docs Initial

Historique des approbations

Approuvé par	
Date:	2023-11-27
Version approuvée	---



Dernière modification : 2024-01-02

1. SOLUTION DESCRIPTION

1.1. GESTION RÉSUMÉ

Description du Projet

Le projet d'intégration entre FreeMoney et le système interopérable de la Banque Centrale des États de l'Afrique de l'Ouest (BCEAO) représente une étape cruciale dans l'évolution des transactions monétaires dans la région. FreeMoney, en tant que plateforme monétique émergente, cherche à étendre sa portée en permettant à ses utilisateurs d'effectuer des transactions fluides et sécurisées avec d'autres systèmes de wallet et des comptes bancaires.

Ce document d'architecture technique vise à définir la structure, les composants et les interactions essentielles nécessaires à la réussite de cette intégration. En comprenant les exigences du système interopérable de la BCEAO, nous élaborons une architecture robuste et évolutive qui garantit la compatibilité, la sécurité et la performance.

Au fil des sections suivantes, nous explorerons les différents aspects de l'architecture technique, allant des vues statiques et logiques du système aux spécifications techniques détaillées, en passant par les décisions architecturales et les schémas de base de données. Cette documentation servira de guide fondamental pour les membres de l'équipe technique, les développeurs, et toute personne impliquée dans le processus d'intégration.

L'objectif ultime de cette initiative est de faciliter des transactions monétaires transparentes et efficaces, renforçant ainsi la confiance des utilisateurs et contribuant à la croissance durable du secteur financier dans la région de l'Afrique de l'Ouest.

Composants Clés :

Dans le cadre de l'intégration entre FreeMoney et le système interopérable de la BCEAO, plusieurs composants clés doivent être pris en compte pour assurer le bon fonctionnement du système.

Voici une liste de composants clés que vous pourriez considérer :

1. FreeMoney Platform:

- Mobiquity : Le cœur de la plateforme FreeMoney, gérant les comptes utilisateurs, les transactions, les dépôts et les retraits.
- Processeur de transactions : Les services interopérable de FreeMoney qui communique avec PI et responsable du traitement et de l'acheminement des transactions entre FreeMoney et la plateforme interopérable.

2. BCEAO Interoperability Platform (PI):

- Le Répertoire des Alias de Comptes (PI-RAC), gérant les alias dans le système interopérable(RAC).
- Le Système de Paiement Instantané (PI-SPI), facilitant les transactions et les paiements entre les systèmes de wallet et banques(API PI).
- Émission et de la gestion des certificats nécessaires à la sécurité des communications, y

compris les certificats de signature des messages ISO 20022, les certificats client pour l'authentification mutuelle (mTLS) et les clés d'APIs.

3. API PI (Option 1 - Connexion Directe à PI):

- Interface de programmation d'application permettant la communication directe entre le système FreeMoney et les API de la plateforme interopérable.

4. API PI AIP (Option 2 - Connexion via l'AIP):

- Interface de programmation d'application pour interagir avec l'Application d'Interfacage Participant (AIP), qui à son tour communique avec la plateforme interopérable.

5. Integration Layer:

- Couche d'intégration facilitant la communication entre les différents composants du système, assurant la cohérence des données et la gestion des erreurs.

6. Database Management System (DBMS):

- Système de gestion de base de données stockant les informations relatives aux comptes utilisateurs dans PI, aux transactions et aux autres informations nécessaires à intégrations.

7. Logging and Monitoring System:

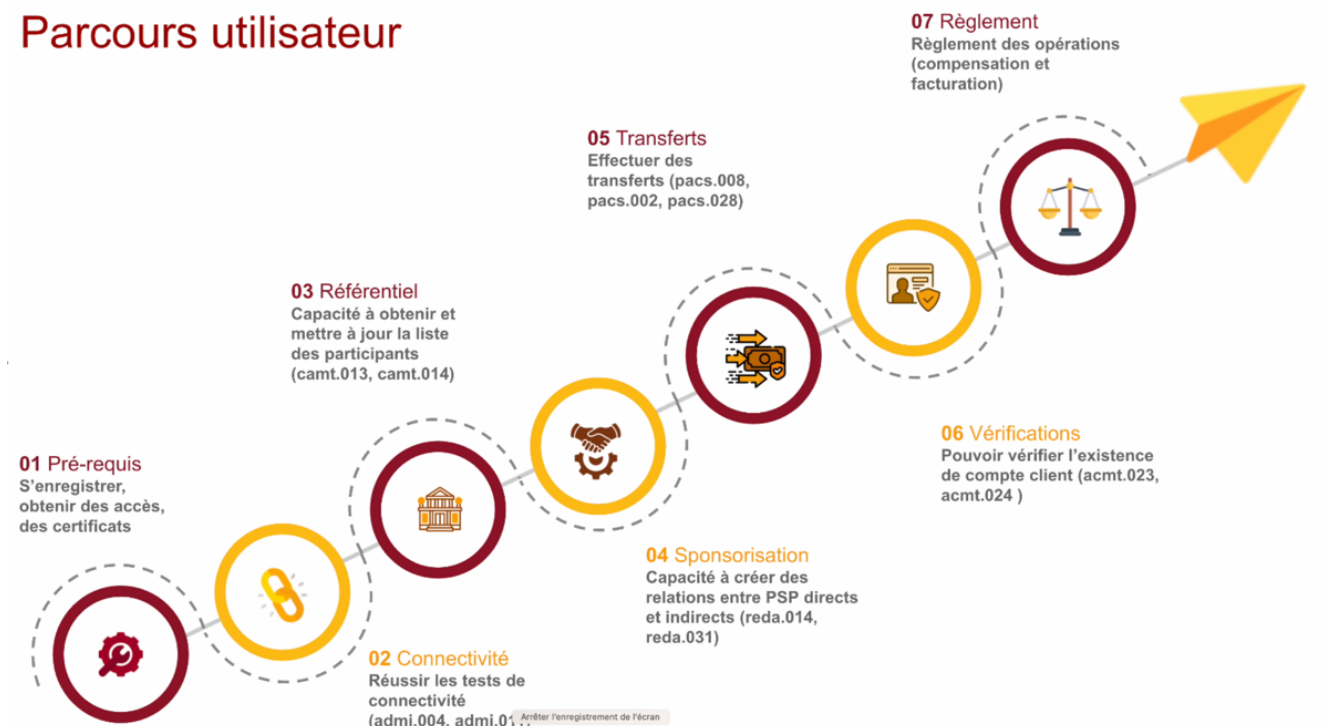
- Système de journalisation et de surveillance permettant de suivre les transactions, de détecter les anomalies et de générer des rapports pour l'analyse ultérieure.

Chaque composant joue un rôle crucial dans le fonctionnement global du système, contribuant à la sécurité, à la fiabilité et à la performance des transactions monétaires.

1.2. ÉTAPES D'INTEGRATION AVEC PI

L'intégration avec la BCEAO implique plusieurs étapes clés, allant de l'enregistrement des acteurs jusqu'à la configuration de la communication entre le système FreeMoney et la plateforme interopérable (PI). Voici un plan d'intégration détaillé :

Parcours utilisateur



1. Enregistrement des Acteurs:

- **Description** : Les acteurs, à savoir les banques et les systèmes de wallet, doivent s'enregistrer en tant que participants dans le système interopérable de la BCEAO (PI).
- **Actions** :
 - Remplir les formulaires d'enregistrement.
 - Recevoir un code membre unique pour identification.

2. Demande d'Accès à la Plateforme Interopérable:

- **Description** : Les participants enregistrés demandent des accès de connectivité vers la plateforme interopérable (PI).
- **Actions** :
 - Initier la demande d'accès.
 - Recevoir les informations de connectivité.

3. Demande de Certificats:

- **Description** : Les participants demandent les certificats nécessaires pour assurer la sécurité des communications.
- **Actions** :
 - Générer une demande de certificat (CSR - Certificate Signing Request).
 - Soumettre la demande au vérificateur.
 - Le vérificateur contrôle attentivement les informations avant de la transmettre au validateur.
 - Le validateur approuve la demande qui est ensuite transmise à l'autorité de certification.

4. Réception des Certificats :

- **Description** : Les certificats nécessaires à la signature des messages ISO 20022 et à l'authentification mutuelle (mTLS) sont émis par l'autorité de certification.
- **Actions** :
- Recevoir le certificat de signature des messages ISO 20022.
- Recevoir le certificat client pour l'authentification mutuelle (mTLS).

5. Configuration de la Communication:

- **Description** : Configurer la communication entre le système FreeMoney et la plateforme interopérable (PI).
- **Actions** :
- Choisir entre la connexion directe à PI (Option 1) ou la connexion via l'AIP (Option 2).
- Intégrer les certificats dans le système de sécurité.
- Configurer les paramètres de communication, tels que les points de terminaison et les protocoles.

6. Options d'Intégration des APIs:

- **Description** : Choisir l'option d'intégration des APIs en fonction des besoins spécifiques du projet.
- **Actions** :
- Option 1 : Intégrer pas à pas à partir des exemples de codes fournis.
- Option 2 : Générer un client d'API à partir de la spécification (OpenAPI).

7. Intégration Pas à Pas:

- **Description** : Intégrer progressivement les différentes interactions entre le système FreeMoney et la plateforme interopérable (PI) en suivant les guides d'intégration pas à pas.
- **Actions** :
- Suivre les scénarios de communication fournis.
- Utiliser les exemples de code pour faciliter l'intégration.

8. Génération du Client d'API:

- **Description** : Générer et configurer un client d'API à partir de la spécification (OpenAPI) pour simplifier l'intégration.
- **Actions** :
- Accéder à la documentation de l'API.
- Télécharger la spécification (OpenAPI).
- Utiliser l'outil OpenAPI Generator pour générer et configurer le client d'API.

9. Validation et Tests:

- **Description** : Valider l'intégration en effectuant des tests approfondis pour garantir la conformité et la robustesse du système.
- **Actions** :
- Tester les différentes fonctionnalités de communication.

- Vérifier la conformité avec les normes de la BCEAO.
- Effectuer des tests de charge pour évaluer les performances.

10. Documentation et Formation:

- **Description :** Documenter l'ensemble du processus d'intégration et fournir une formation aux membres de l'équipe.
- **Actions :**
- Créer une documentation complète de l'architecture et du processus d'intégration.
- Organiser des séances de formation pour les développeurs et les opérateurs.

11. Déploiement :

- **Description :** Déployer le système intégré dans l'environnement de production.
- **Actions :**
- Suivre les procédures de déploiement.
- Effectuer des tests finaux dans l'environnement de production.

12. Maintenance Continue :

- **Description :** Établir des mécanismes de surveillance continue, de gestion des incidents et de mise à jour du système.
- **Actions :**
- Mettre en place des outils de surveillance.
- Planifier des mises à jour régulières.

En suivant ces étapes de manière méthodique, nous devrons être en mesure d'assurer une intégration réussie entre FreeMoney et le système interopérable de la BCEAO. Chaque étape est cruciale pour garantir la sécurité, la conformité et la performance du système.

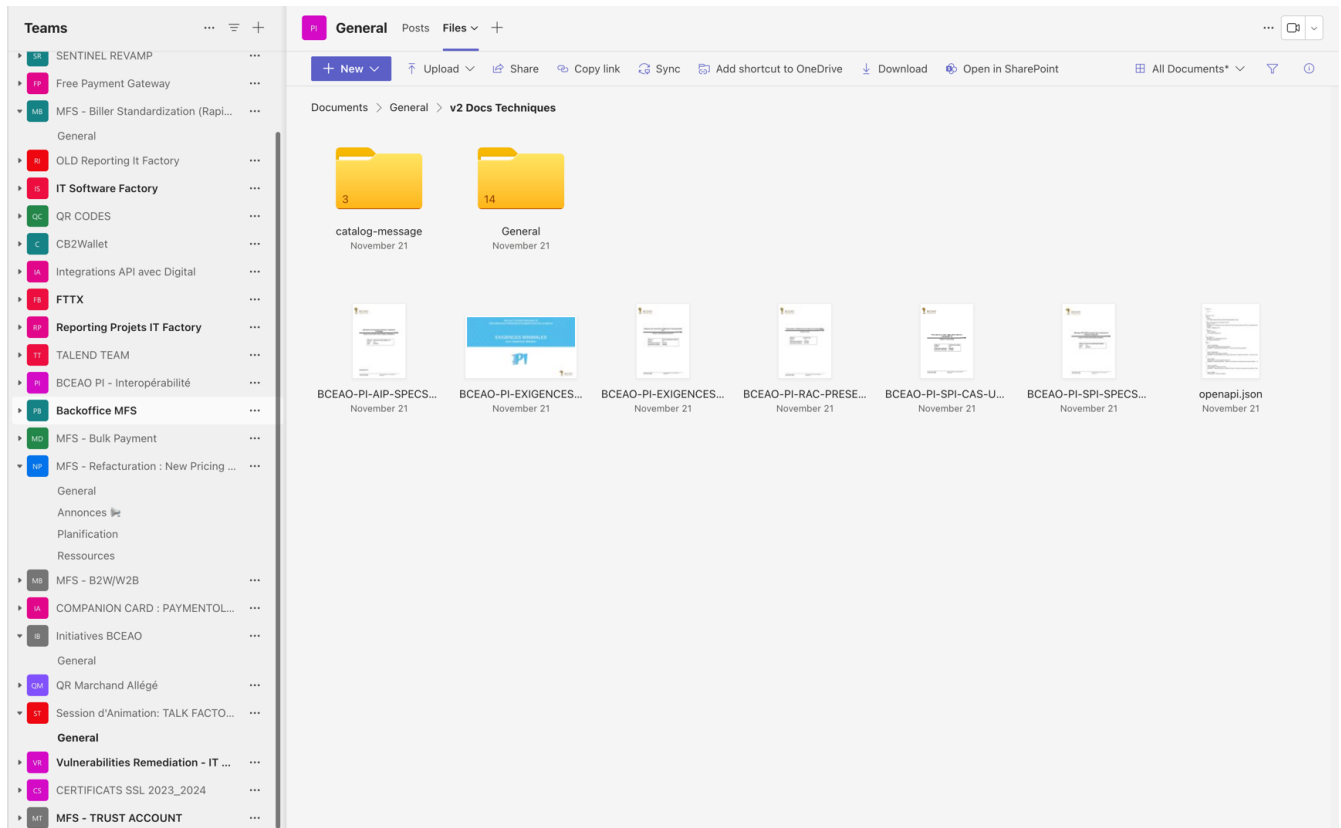
1.3. GLOSSAIRE

Terme	Définition
USSD	Données de service supplémentaires non structurées
MSISDN	Numéro RNIS de la station mobile
SMS	Service de messages courts
API	Interface de programmation d'applications
UG	Passerelle USSD
Talend	Plate-forme middleware
Interface graphique	Utilisateur graphique
DWH	Data Warehouse
FreeMoney	Système de mobile money de Free Sénégal, permettant aux utilisateurs de stocker de l'argent sur leur téléphone mobile pour des transactions financières.

Terme	Définition
Load Balancer (F5)	Dispositif qui répartit le trafic des utilisateurs entre les serveurs web Apache pour équilibrer la charge.
SQL Server	Système de gestion de base de données relationnelle utilisé pour stocker les données liées aux transactions et à la gestion des comptes.
Mobiquity	Plateforme FreeMoney fournissant des API pour les transferts compte à compte (P2P), les transferts d'un compte vers un numéro sans compte (P2C), et la vérification du statut des comptes.
Serveur SMS	Serveur envoyant des notifications SMS aux utilisateurs pour les informer des transactions et des mises à jour importantes.
Proxy	Serveur intermédiaire permettant l'accès sécurisé à la plateforme web partenaire depuis l'Internet.
SolarWinds	Outil de surveillance utilisé pour surveiller les performances des serveurs web et de la base de données.
Monnaie Mobile	Système permettant aux utilisateurs de stocker et de transférer de l'argent via leur téléphone mobile.
KYC (Knows Your Customer)	Processus de vérification de l'identité des clients, incluant la collecte d'informations personnelles et de documents d'identification.
DMZ (Zone Démilitarisée)	Zone de réseau intermédiaire entre le réseau interne et Internet, souvent utilisé pour héberger des serveurs accessibles depuis l'extérieur.
BCEAO	La Banque Centrale des États de l'Afrique de l'Ouest.
PI-SPI	Le Système de Paiement Instantané.
PI-RAC	Le Répertoire des Alias de Comptes.
BCEAO	Banque Centrale des États de l'Afrique de l'Ouest
API	Interface de Programmation d'Application
XMLDsig	XML Digital Signature
JWS	JSON Web Signature
CSR	Certificate Signing Request
mTLS	Mutual Transport Layer Security
PI	Plateforme Interopérable
AIP	Application d'Interfacage Participant
UML	Unified Modeling Language
IETF	Internet Engineering Task Force
W3C	World Wide Web Consortium
X.509	Standard de format de certificat électronique

1.4. DOCUMENTS CONNEXES

- [Lien Spécifications techniques PI](#) : La documentation de la plateforme interoperable de la BCEAO



- [Lien de la plateforme interoperable de la BCEAO](#)

1.5. Objectifs et restrictions de l'architecture

Architecture :

L'architecture repose sur une infrastructure comprenant plusieurs composants clés :

- Des serveurs d'applications Java EE avec répartition de charge via des load balancers F5 pour garantir une haute disponibilité des services web.
- Une base de données SQL Server centralisée pour le stockage transactionnel et la gestion des comptes.
- La plateforme Mobiquity fournissant des API core de transfert, dépôt, retrait et vérification de statut.
- Des interfaces web : une plateforme Backoffice(MFS BACKOFFICE) pour un meilleur suivi des transactions et paiements.
- Des outils de supervision (SolarWinds), de notification (SMS) pour l'administration, l'exploitation et l'usage des services transactionnels.

Objectifs :

L'objectif de l'architecture dans le contexte de l'intégration entre FreeMoney et le système interopérable de la BCEAO est de fournir une structure et un cadre technologique qui répondent aux exigences fonctionnelles et non fonctionnelles du système. Voici les principaux objectifs de l'architecture :

1. Assurer l'Interopérabilité :

- **Objectif :** Concevoir une architecture qui permet une communication transparente entre FreeMoney et le système interopérable de la BCEAO, garantissant l'échange efficace de données et de transactions.

2. Garantir la Sécurité des Transactions :

- **Objectif :** Mettre en place des mécanismes de sécurité robustes, tels que la signature numérique des messages et le chiffrement, pour assurer l'intégrité, l'authenticité et la confidentialité des transactions.

3. Faciliter l'Extensibilité :

- **Objectif :** Créer une architecture flexible qui permet l'ajout de nouvelles fonctionnalités, de nouveaux services ou l'expansion vers de nouveaux marchés sans perturber les fonctionnalités existantes.

4. Optimiser les Performances :

- **Objectif :** Concevoir une architecture qui optimise les performances du système, minimisant les temps de réponse et maximisant la capacité à gérer un volume élevé de transactions.

5. Respecter les Normes et Standards :

- **Objectif :** S'aligner sur les normes et les meilleures pratiques définies par la BCEAO et d'autres organismes régulateurs, garantissant ainsi la conformité et la qualité du système.

6. Faciliter la Maintenance et la Gestion :

- **Objectif :** Concevoir une architecture qui simplifie les opérations de maintenance, les mises à jour du système et la gestion des configurations pour minimiser les temps d'arrêt.

7. Favoriser la Scalabilité :

- **Objectif :** Concevoir une architecture qui peut évoluer horizontalement ou verticalement pour répondre à la demande croissante, tout en maintenant des performances élevées.

8. Améliorer la Traçabilité des Transactions :

- **Objectif :** Mettre en place des mécanismes de journalisation et de traçabilité robustes pour faciliter la résolution des problèmes, la détection d'incidents et la conformité réglementaire.

9. Faciliter le Déploiement :

- **Objectif :** Créer une architecture qui simplifie le processus de déploiement dans différents environnements, favorisant ainsi la flexibilité opérationnelle.

10. Renforcer la Fiabilité :

- **Objectif :** Concevoir une architecture qui minimise les risques de pannes, garantit la disponibilité du système et offre une expérience utilisateur fiable.

11. Améliorer l'Expérience Utilisateur :

- **Objectif** : Concevoir une architecture conviviale qui offre une expérience utilisateur transparente et satisfaisante, renforçant ainsi l'adoption et la fidélité des utilisateurs.

L'atteinte de ces objectifs garantira le succès de l'intégration en créant un système robuste, sécurisé et performant, capable de répondre aux besoins actuels et futurs du paysage financier de l'Afrique de l'Ouest.

1.6. Spécifications fonctionnelles

Spécifications Fonctionnelles pour le Système d'Intégration FreeMoney - PI-RAC & PI-API

1.6.1. Gestion des Alias :

Un alias de compte est une identification choisie par un client qui lui permet de recevoir des virements sans fournir les informations d'identification de son compte.

1. Par Adresse de Paiement (SHID) :

Les utilisateurs peuvent créer un alias en associant une adresse de paiement générée par PI-RAC.

1. Par Numéro de Téléphone Mobile (MBNO) :

Les utilisateurs peuvent créer un alias en associant leur numéro de téléphone mobile.

1. Par Identifiant de Compte Marchand (MCOD) :

Les utilisateurs peuvent créer un alias en associant leur identifiant de compte marchand.

1. Recherche d'Alias :

Permet la recherche d'un alias dans le système.

1.6.2. Transferts P2P :

1. Demande de Transfert vers PI vers un Autre Participant :

Les utilisateurs peuvent initier une demande de transfert vers un autre participant via le PI en spécifiant l'alias du bénéficiaire.

1. Accepter le Transfert Provenant de PI :

Les participants peuvent accepter un transfert provenant du PI après réception de la notification correspondante.

1. Réception de la Notification de Paiement Irrévocable Provenant de PI :

Les participants reçoivent une notification de paiement irrévocable suite à l'acceptation du transfert.

1. Notifier Son Client de l'État du Transfert :

Les participants peuvent informer leurs clients de l'état réussi du transfert après réception de la notification de paiement irrévocable.

2. Vérifier le Statut d'un Transfert :

Les participants peuvent vérifier le statut actuel d'un transfert en émettant une demande spécifique au PI.

1.6.3. Paiements Marchands :

1. Envoi de Paiement par un Client :

Les clients de FreeMoney peuvent envoyer des paiements à des clients corporate participants en spécifiant l'alias du bénéficiaire.

1. Notification de Paiement :

Les clients et les marchands reçoivent une notification de paiement avec l'alias correspondant.

1. Demande de Paiement Immédiate par un Marchand :

Les marchands peuvent demander un paiement immédiat à un client en spécifiant l'alias du client.

1. Demande de Paiement avec Date de Péréemption par un Marchand :

Les marchands peuvent demander un paiement avec une date limite à un client en spécifiant l'alias du client.

1. Acceptation/Refus des Demandes de Paiement :

Les clients peuvent accepter, refuser ou ignorer une demande de paiement du marchand, et vice versa.

1.6.4. Test de Connectivité vers PI :

1. Échange de Messages Admi.004 et Admi.011 :

Le système effectue un échange de messages dans les deux sens pour tester la connectivité avec le PI.

1.6.5. Sponsorisation de Participant Indirect :

1. Créer la Relation :

Les participants directs peuvent créer une relation avec un participant indirect en spécifiant l'alias du participant indirect.

1. Modification de la Relation :

Les participants directs peuvent modifier les détails de la relation avec un participant indirect en

soumettant une demande spécifique.

1. Rompre une Relation :

Les participants directs peuvent rompre la relation avec un participant indirect en spécifiant l'alias du participant indirect.

1.6.6. Gestion de Compte des Participants :

1. Compensation :

Un message camt.053 est envoyé à PI après chaque compensation pour informer de la balance du participant.

1. Facturation :

Un message représentant la facture camt.086 est envoyé mensuellement à chaque participant.

1. Réconciliation :

Le participant peut envoyer un message camt.060 pour obtenir le détail des transactions.

Les spécifications fonctionnelles détaillées ci-dessus établissent le cadre opérationnel et fonctionnel du système d'intégration entre FreeMoney et le système interopérable de la BCEAO (PI-RAC). Ces spécifications offrent une compréhension approfondie des fonctionnalités clés, des processus d'interaction entre les participants et du flux global des transactions.

1.7. Caractéristiques qualitatives

Classification	Description
Fiabilité	Le système doit assurer une disponibilité élevée (99,99%) et minimiser les temps d'arrêt non planifiés.
Sécurité	Les transactions doivent être sécurisées via des mécanismes de chiffrement robustes et des processus d'authentification mutuelle.
Performances	Le système doit offrir des performances optimales, avec une réponse rapide aux requêtes et une gestion efficace du trafic, même en période de charge élevée.
Évolutivité	Le système doit être évolutif pour s'adapter à la croissance des utilisateurs et des transactions.
Interopérabilité	Le système doit être compatible avec les normes et protocoles établis pour garantir une interopérabilité réussie.
Maintenabilité	Le code doit être bien documenté, facilitant la maintenance et les mises à jour futures.
Extensibilité	Le système doit permettre l'ajout de nouvelles fonctionnalités sans perturber les fonctionnalités existantes.

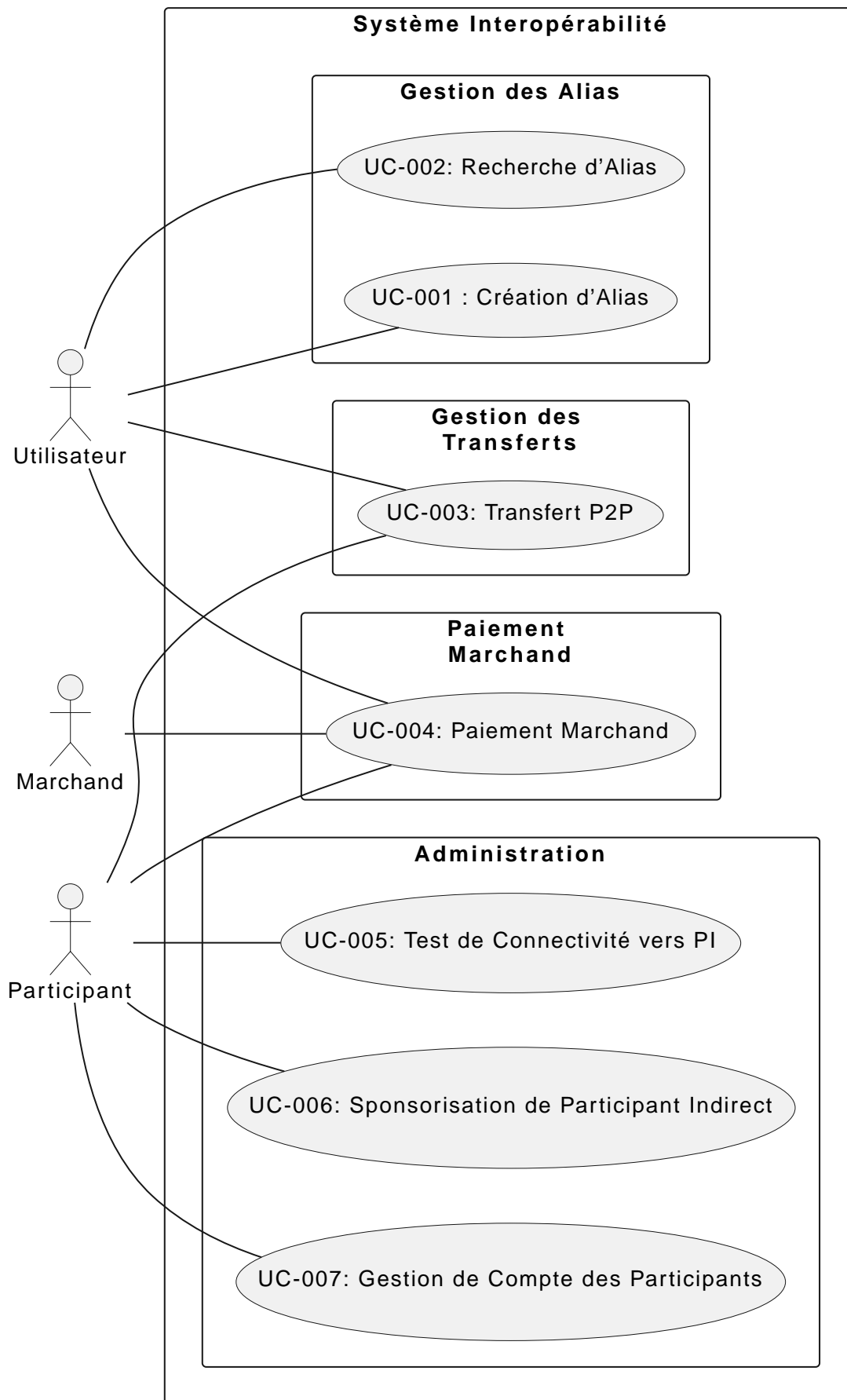
Classification	Description
Facilité d'utilisation	L'interface utilisateur doit être conviviale, facilitant l'utilisation du système par des utilisateurs de divers niveaux de compétence.
Disponibilité	Le système doit garantir une disponibilité 24/7, avec des procédures de sauvegarde et de récupération efficaces.

1.8. VUE DES CAS D'UTILISATION

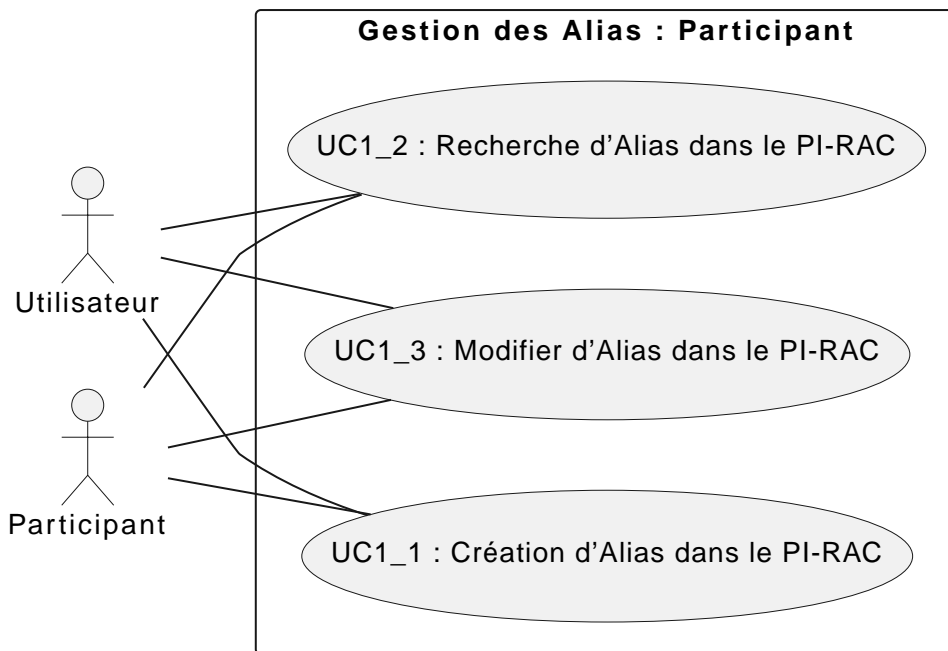
Cas d'Utilisation	Description
UC-001: Création d'Alias	Les utilisateurs peuvent créer un alias en associant une adresse de paiement (SHID), un numéro de téléphone mobile (MBNO) ou un identifiant de compte marchand (MCOD).
UC-002: Recherche d'Alias	Permet aux utilisateurs de rechercher un alias dans le système(PI-RAC).
UC-003: Transfert P2P	<ul style="list-style-type: none"> - Demande de transfert vers PI vers un autre participant. - Accepter le transfert provenant de PI. - Réception de la notification de paiement irrévocable provenant de PI. - Notifier son client de l'état du transfert. - Vérifier le statut d'un transfert.

UC-004: Paiement Marchand	<ul style="list-style-type: none"> - Envoi de paiement par un client. - Notification de paiement vers le client et le marchand. - Demande de paiement immédiate par un marchand vers un client. - Demande de paiement avec date de péremption par un marchand vers un client. - Accepter une demande paiement immédiat. - Refuser une demande paiement immédiat. - Accepter une demande paiement avec date de péremption. - Refuser une demande de paiement avec date de péremption. - Ignorer une demande de paiement avec date de péremption. - Ignorer une demande de paiement immédiat.
UC-005: Test de Connectivité vers PI	<ul style="list-style-type: none"> - Échange de messages Admi.004 et Admi.011 dans les deux sens pour tester la connectivité.
UC-006: Sponsorisation de Participant Indirect	<ul style="list-style-type: none"> - Créer la relation avec un participant indirect. - Modifier la relation avec un participant indirect. - Rompre la relation avec un participant indirect.
UC-007: Gestion de Compte des Participants	<ul style="list-style-type: none"> - Compensation : Envoi d'un message camt.053 à PI après chaque compensation pour informer de la balance du participant. - Facturation : Envoi d'un message représentant la facture camt.086 mensuellement à chaque participant. - Réconciliation : Le participant peut envoyer un message camt.060 pour obtenir le détail des transactions.

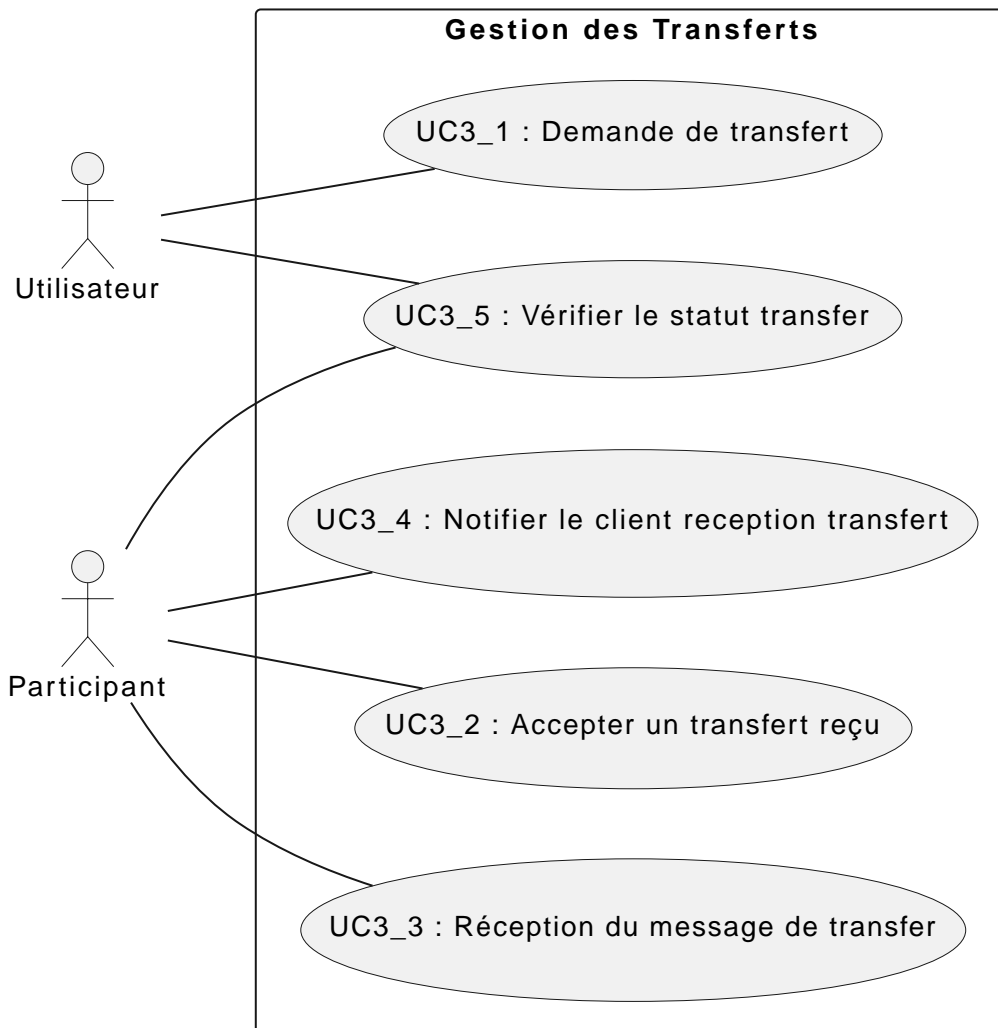
- Cas d'utilisation de la plateforme interoperable : PI



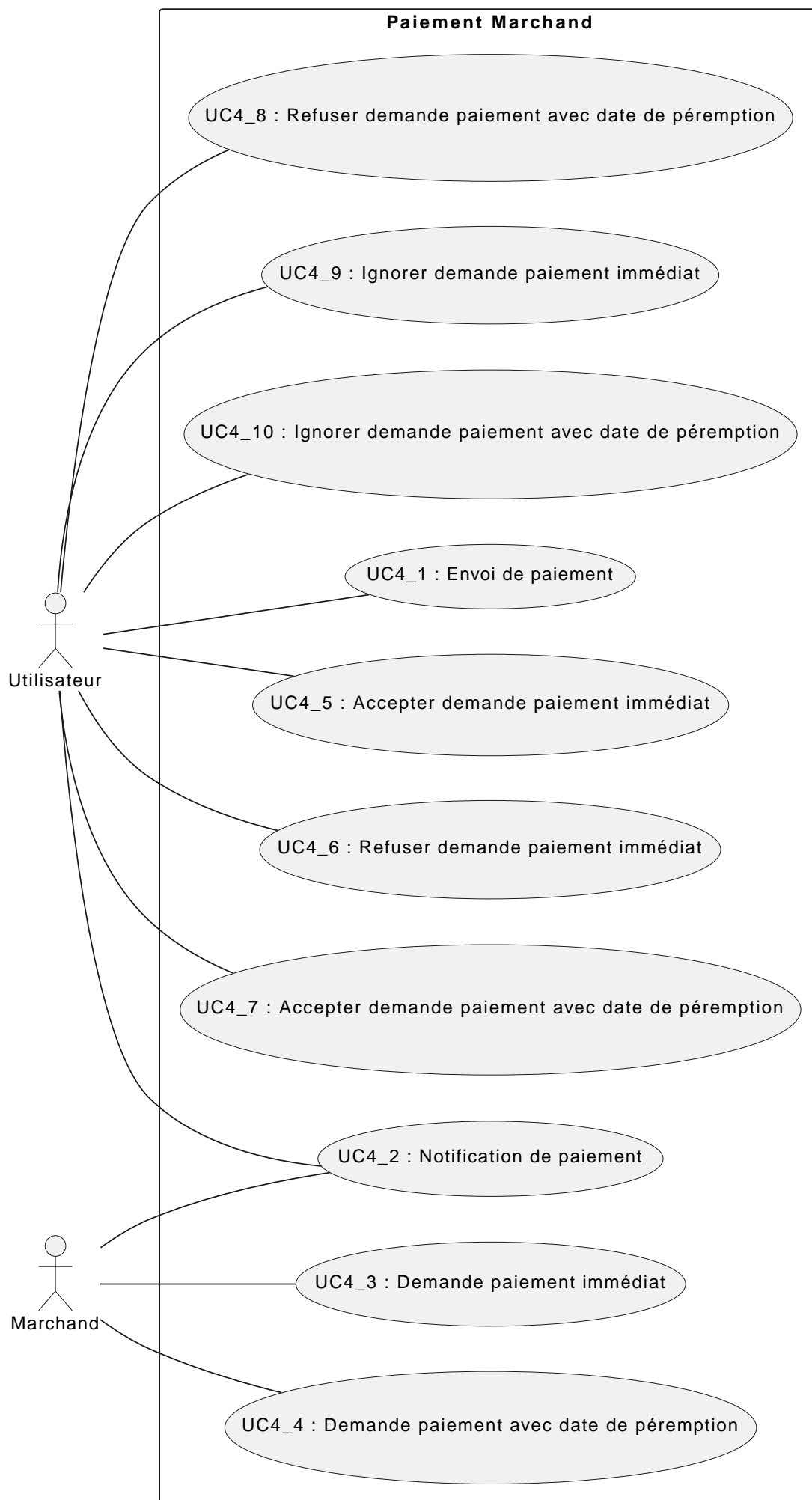
- Gestion des Alias



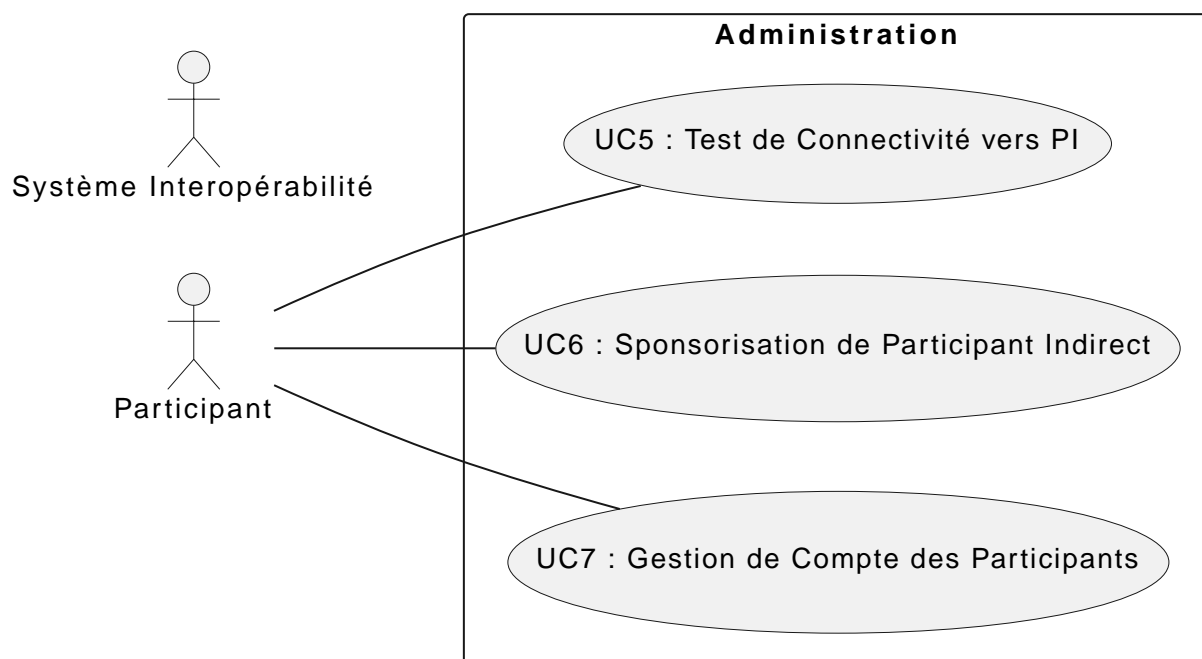
- Gestion des Transferts



- Paiement Marchand



- Administration



1.9. SURVEILLANCE EXIGENCES

1.9.1. SURVEILLANCE DU SYSTÈME

Aspect de Surveillance	Description
Journalisation des Événements	Enregistrement des opérations, erreurs, temps de réponse, etc., pour une analyse approfondie.
Surveillance des Performances	Surveiller les temps de réponse, la charge du serveur, l'utilisation de la mémoire, etc.
Surveillance de la Connectivité	Suivre la disponibilité et la stabilité de la connexion avec la plateforme interopérable (PI).
Alertes et Notifications	Mettre en place des alertes pour informer les administrateurs des anomalies, erreurs, etc.
Surveillance de la Sécurité	Examiner les journaux de sécurité pour détecter toute activité suspecte et garantir la conformité aux normes de sécurité.
Intégrité des Données	Mettre en œuvre des mécanismes de surveillance pour garantir l'intégrité des données lors des transferts, paiements, et autres opérations critiques.

1.9.2. EXIGENCES DU SYSTÈME

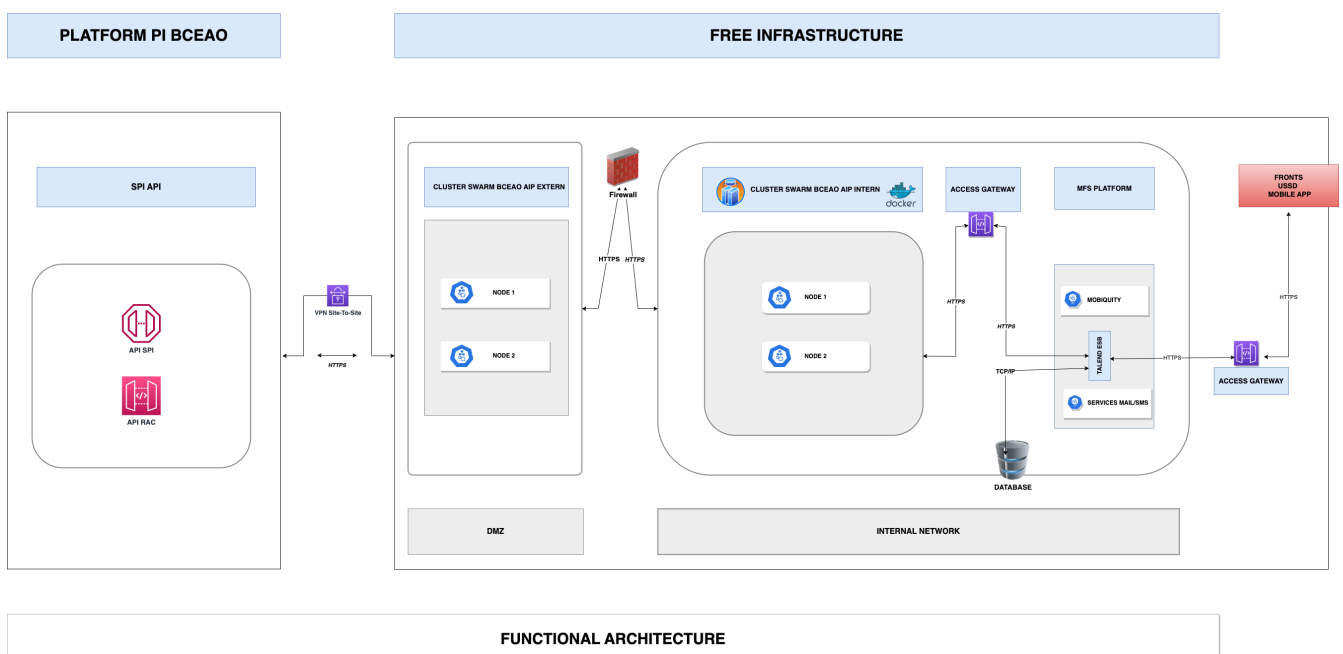
Exigence du Système	Description
Sécurité	Utilisation de protocoles sécurisés (HTTPS), gestion sécurisée des certificats.

Performances	Réponse aux demandes dans des délais spécifiés, charge maximale spécifiée.
Interopérabilité	Compatibilité avec la plateforme interopérable (PI) selon les normes de la BCEAO. Connexions directes à l'API PI et utilisation de l'AIP PI prises en charge.
Fiabilité	Disponibilité fiable, mécanismes de sauvegarde et de récupération pour prévenir la perte de données.
Scalabilité	Conception pour évoluer avec le nombre d'utilisateurs et de transactions.
Journalisation et Audit	Génération de journaux détaillés, conservation conforme aux réglementations.
Documentation	Maintenance d'une documentation complète du système, y compris architecture, intégration, sécurité, etc.
Tests et Validation	Mise en œuvre de procédures de test complètes, tests de sécurité réguliers.
Formation et Support	Programmes de formation pour utilisateurs et administrateurs. Support technique efficace.
Conformité Réglementaire	Conformité aux réglementations en vigueur, notamment aux normes de la BCEAO.

2. ARCHITECTURE DE LA SOLUTION

2.1. Vue logique

2.1.1. Architecture



2.1.2. Les composants du système.

Composant	Description
Module Gestion des Alias	Gère la création et la recherche d'alias, incluant les types d'alias tels que l'adresse de paiement, le numéro de téléphone mobile, et l'identifiant de compte marchand.
Module Gestion des Transferts	Gère les transferts peer-to-peer (P2P), incluant les demandes de transfert, l'acceptation, la notification de paiement irrévocable, la vérification du statut des transferts, etc.
Module Paiement Marchand	Gère les transactions de paiement marchand entre un client end user et un client corporate des participants.
Module Administration	Gère les opérations administratives, y compris le test de connectivité vers PI, la sponsorship de participant indirect, et la gestion des comptes des participants.
API PI	Interface permettant la connexion directe au système interopérable de la BCEAO pour la communication avec les APIs de PI.
AIP PI	Interface permettant au système d'appeler les APIs de l'AIP (Application d'Interfacage Participant) pour communiquer avec PI.
Base de Données	Stocke les données nécessaires au fonctionnement du système, y compris les informations sur les alias, les transferts, les paiements marchands, etc.
Service de Sécurité	Fournit des fonctionnalités de sécurité, y compris la gestion des certificats, la signature des messages, et l'authentification mutuelle (mTLS).
Logs	Gère la journalisation des événements, permettant le suivi et l'analyse des opérations du système.
F5	Composant réseau fournissant des fonctionnalités de gestion du trafic, d'équilibrage de charge et de sécurité.
Access Gateway	Solution de sécurité et de proxy permettant de sécuriser les connexions et d'agir en tant que point d'accès centralisé.

Composant	Description
Mobiquity	Composant central de la plateforme monétique de FreeMoney, fournissant des fonctionnalités liées aux transactions monétaires et à la gestion des comptes.

2.1.3. Les dépendances externes

Dépendance Externe	Description
Réseau de Communication	Les opérations dépendent d'un réseau de communication fiable pour l'échange de données avec la PI, d'autres participants, et les services externes tels que Mobiquity.
Mobiquity	Composant externe au système, jouant un rôle central dans la plateforme monétique de FreeMoney. Les fonctionnalités monétaires, y compris les transactions et la gestion des comptes, dépendent de Mobiquity.
Solution F5	Utilisée pour la gestion du trafic, l'équilibrage de charge et la sécurité. Le bon fonctionnement du système dépend de la disponibilité et de la performance de la solution F5.
Access Gateway	Solution de sécurité et de proxy agissant comme point d'accès centralisé. Les connexions sécurisées et la gestion des flux de données dépendent du bon fonctionnement de l'Access Gateway.
Environnement d'Exécution	Le système dépend d'un environnement d'exécution stable, compatible avec les technologies utilisées (Spring Boot Cloud, SQL Server, etc.).
AIP PI	L'Application d'Interfacage Participant est une interface permettant au système d'appeler les APIs de l'AIP pour communiquer avec la PI.
PI API	L'API de la Plateforme Interopérable utilisée pour la communication directe avec les autres participants et la BCEAO.
PI RAC	Le Référentiel d'Adresses des Clients de la Plateforme Interopérable, utilisé pour la gestion des relations participant-participant.

Dépendance Externe	Description
Serveur de Notification SMS	Permet l'envoi de notifications SMS pour informer les utilisateurs des transactions, des mises à jour de compte, ou d'autres événements importants.

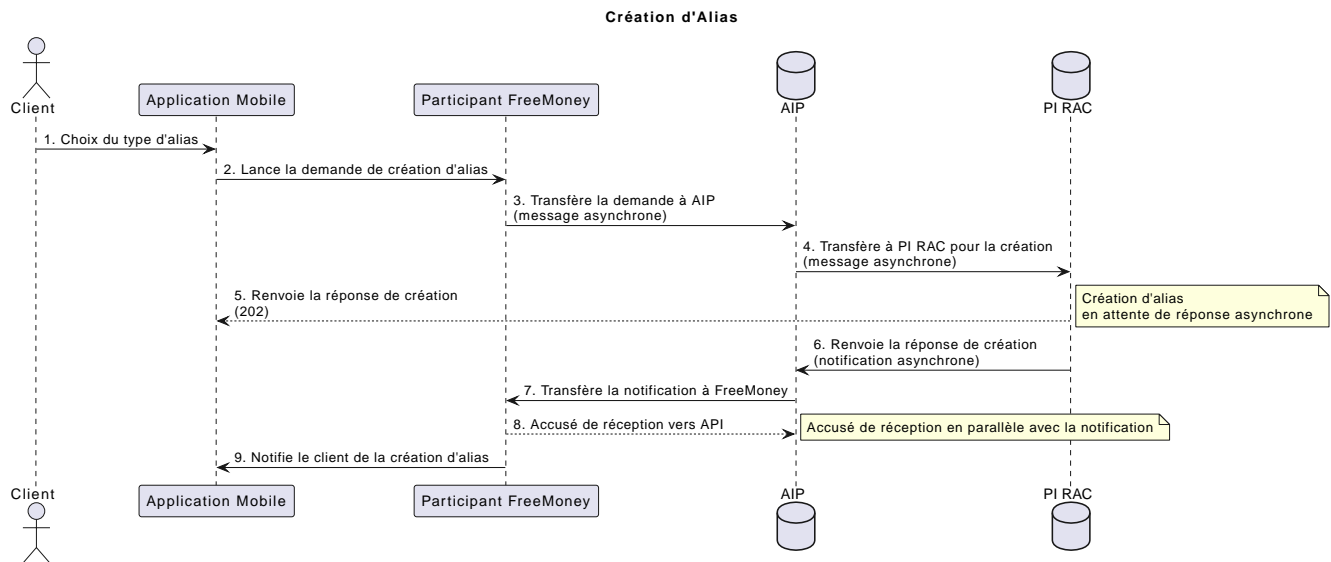
2.2. Diagrammes d'interaction

2.2.1. SEQ-001 : Création d'Alias

La description textuelle de la séquence du cas d'utilisation "Création d'Alias" avec les mêmes acteurs que dans le cas du Transfert P2P est la suivante :

1. Le client, à travers l'application mobile, décide de créer un alias et choisit le type d'alias qu'il souhaite créer (par exemple, une adresse de paiement, un numéro de téléphone mobile, ou un identifiant de compte marchand).
2. L'application mobile communique avec le participant FreeMoney pour lancer la demande de création d'alias.
3. Le participant FreeMoney, à travers son Application d'Interfacage Participant (AIP), transfère la demande de création d'alias vers PI RAC (Répertoire des Alias de Compte) en utilisant un message asynchrone.
4. PI RAC reçoit la demande de création d'alias de manière asynchrone et procède à la création de l'alias dans son système et envoie une réponse avec code http 202.
5. Une fois l'alias créé avec succès, PI RAC envoie une notification de création d'alias en utilisant un message asynchrone.
6. L'AIP reçoit la notification de création d'alias de PI RAC.
7. L'AIP transfère la notification au participant FreeMoney.
8. Le participant FreeMoney prend connaissance de la notification de création d'alias et finalise le processus de création d'alias pour le client.

Cette séquence illustre comment un client, à travers l'application mobile, peut initier la création d'un alias, et comment cette demande est gérée de manière asynchrone par les différents composants du système, notamment l'AIP et PI RAC.



2.2.2. SEQ-002 : Mise à jour d'Alias

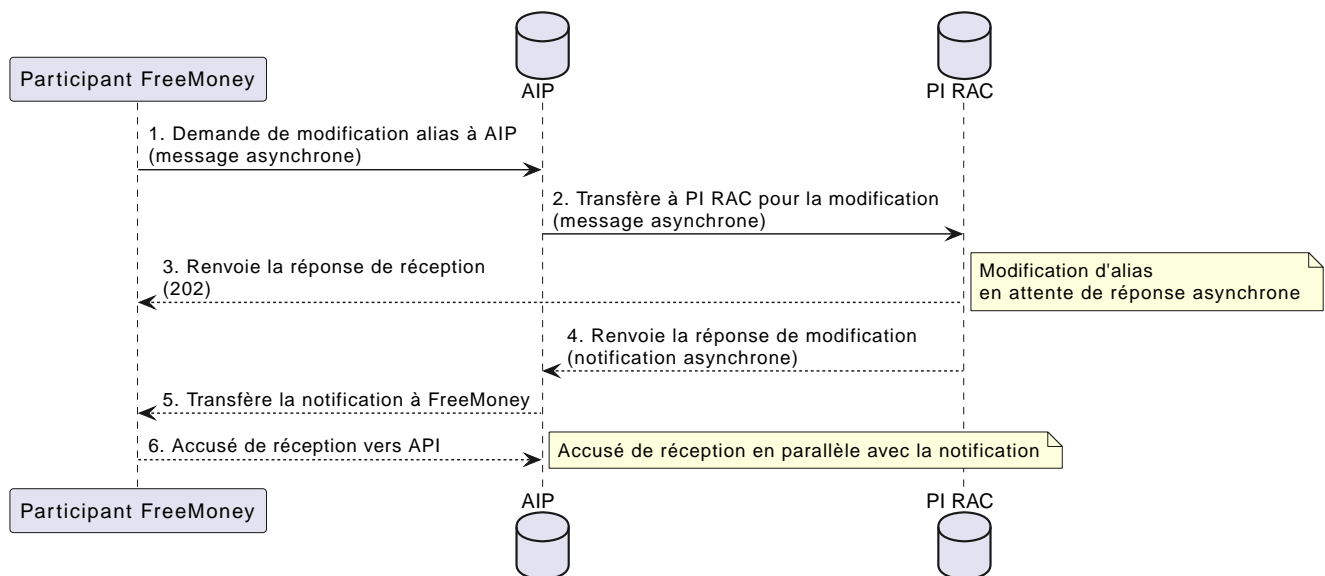
Cette opération permet de mettre à jour les informations du client propriétaire de l'alias ou les informations sur le compte du client. Traitement de la réponse Mise à jour des informations clientèles dans le SI du participant(FREE-MONEY). Il est important de mettre à jour la date de modification (renvoyé par PI) de l'alias dans notre Système d'Information pour le client donné.

La description textuelle de la séquence du cas d'utilisation "Modification d'Alias" en se basant sur la création d'alias est la suivante :

1. Le participant FreeMoney, via son Application d'Interfacage Participant (AIP), transfère la demande de modification d'alias vers PI RAC en utilisant un message asynchrone.
2. PI RAC reçoit la demande de modification d'alias de manière asynchrone et effectue la modification de l'alias dans son système et envoie une réponse avec code http 202.
3. Une fois la modification effectuée avec succès, PI RAC envoie une notification de modification d'alias en utilisant un message asynchrone.
4. L'AIP reçoit la notification de modification d'alias de PI RAC.
5. L'AIP transfère la notification au participant FreeMoney.
6. Le participant FreeMoney prend connaissance de la notification de modification d'alias et finalise le processus de modification d'alias pour le client.

Cette séquence illustre comment un client peut demander la modification d'un alias existant, et comment cette demande est traitée de manière asynchrone par les composants du système, notamment l'AIP et PI RAC.

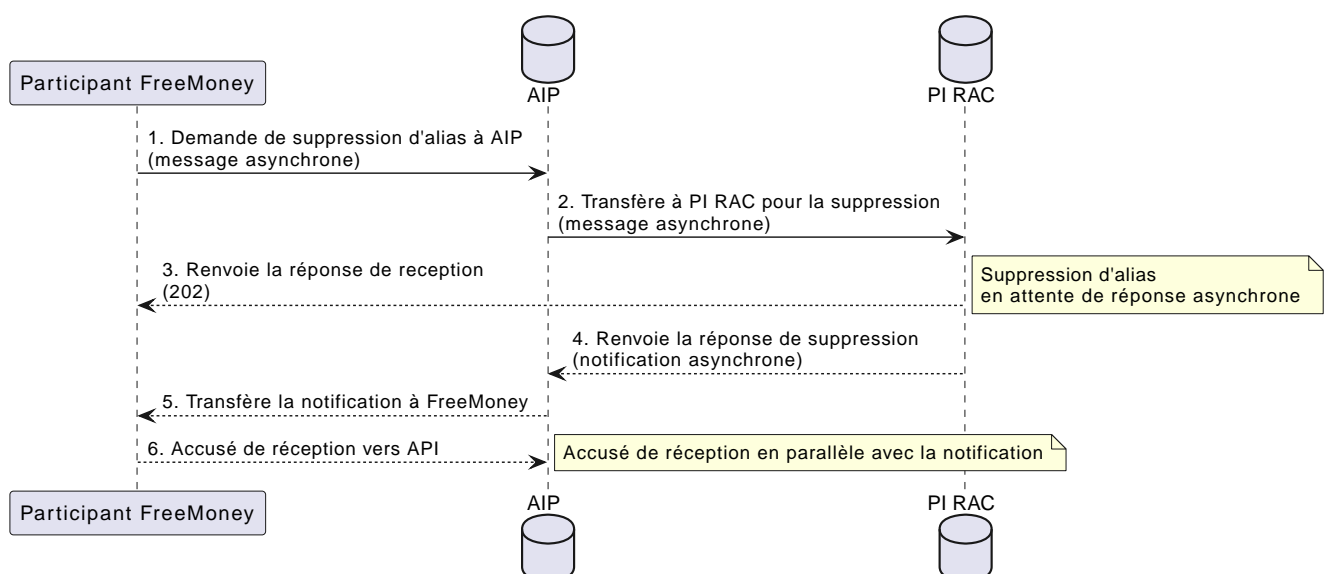
Modification des informations liée à l'alias



2.2.3. SEQ-003: Suppression d'Alias

1. Le Participant FreeMoney, via son Application d'Interfacage Participant (AIP), envoie la demande de suppression d'alias à PI RAC en utilisant un message asynchrone.
2. PI RAC reçoit la demande de suppression d'alias de manière asynchrone et procède à la suppression de l'alias dans son système et envoie une réponse avec code http 202.
3. Une fois la suppression effectuée avec succès, PI RAC envoie une notification de suppression d'alias en utilisant un message asynchrone.
4. L'AIP reçoit la notification de suppression d'alias de PI RAC.
5. L'AIP transfère la notification au Participant FreeMoney.

Le Participant FreeMoney notifie l'application mobile du client de la suppression réussie de l'alias.



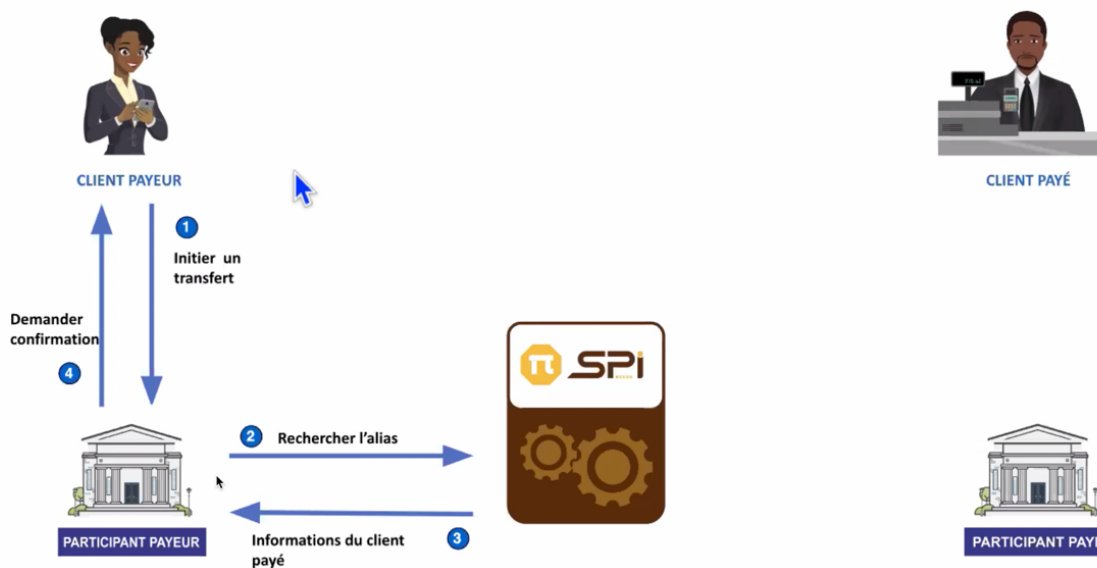
2.2.4. SEQ-004: Recherche d'Alias

La description textuelle du cas d'utilisation "Recherche d'Alias" est la suivante :

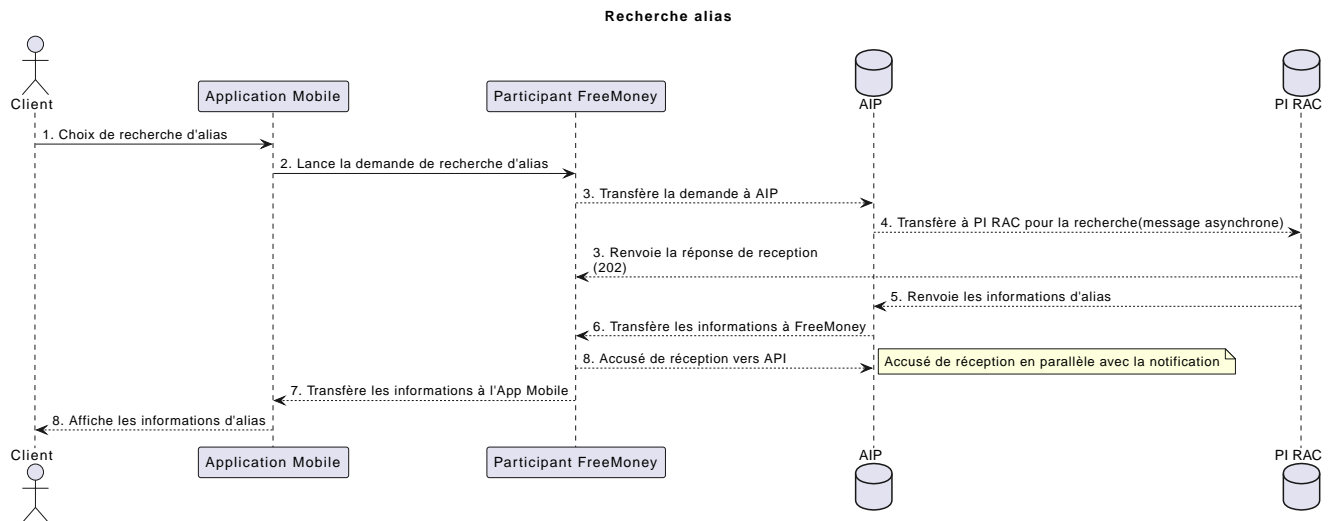
1. Le client, via l'application mobile, sélectionne l'option de recherche d'alias.
2. L'application mobile communique avec le Participant FreeMoney pour lancer la demande de recherche d'alias.
3. Le Participant FreeMoney, via son Application d'Interfacage Participant (AIP), transfère la demande de recherche d'alias à PI RAC.
4. PI RAC traite la demande de recherche d'alias et renvoie les informations associées à l'alias recherché.
5. L'AIP reçoit les informations de PI RAC.
6. L'AIP transfère les informations au Participant FreeMoney.
7. Le Participant FreeMoney affiche les informations d'alias sur l'application mobile du client.

Cette séquence illustre comment un client peut effectuer une recherche d'alias, et comment cette demande est traitée par les composants du système, en particulier l'AIP et PI RAC, pour fournir les informations associées à l'alias recherché.

- Flow simplifié



- Flow détaillés



2.2.5. SEQ-005 : Transfert P2P | Demande de transfert vers PI vers un autre participant.

Description textuelle du scénario de transfert P2P :

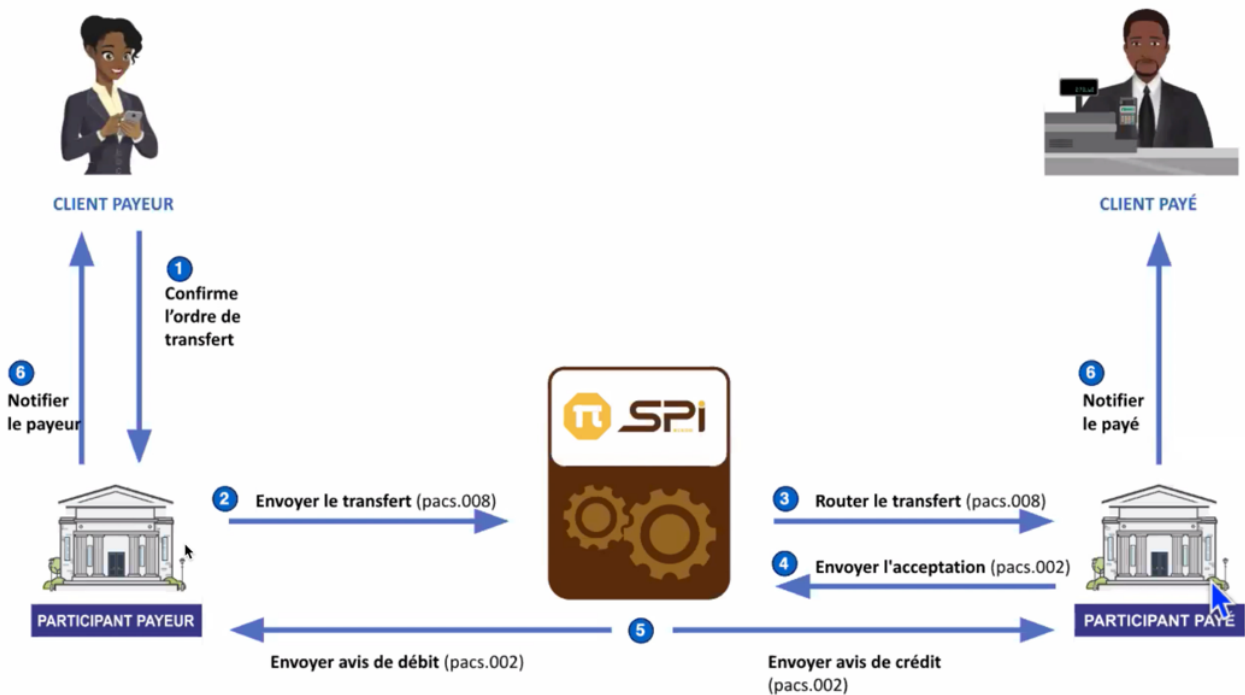
1. Le **Client Particulier Payeur** lance l'application mobile.
2. Le **Client Particulier Payeur** initie un transfert P2P en spécifiant l'alias du client payé.
3. L'**Application Mobile** interroge le **Participant Client Payeur (FREE-MONEY)** pour obtenir des informations sur le client payé.
4. Le **Participant Client Payeur** interroge l'**Application d'Interface Participant (AIP)** pour obtenir des informations sur le client payé.
5. L'**AIP** interroge le **PI RAC (Repertoire des Alias de Compte)** pour obtenir des informations sur le client payé.
6. Le **PI RAC** répond en fournissant les informations du client payé.
7. Les informations du client payé sont renvoyées de manière synchrone de l'**AIP** au **Participant Client Payeur**, puis affichées dans l'**Application Mobile**.
8. Le **Client Particulier Payeur** confirme le paiement dans l'application mobile.
9. L'**Application Mobile** envoie la demande de transfert au **Participant Client Payeur (FREE-MONEY)**.
10. Le **Participant Client Payeur** envoie la demande de transfert au **Système PI SPI (Système de Paiement Instantané)**.
11. Le **Système PI SPI** route le transfert au **Participant Client Payé (OTHERS)** en utilisant un message ISO 20022 pacs.008. Le statut 202 est renvoyé au **Participant Client Payeur**.
12. Le **Système PI SPI** route le transfert au **Participant Client Payé**.
13. Le **Participant Client Payé** accepte le paiement en envoyant un message ISO 20022 pacs.002 au **Système PI SPI**.
14. Le **Système PI SPI** notifie l'irrévocabilité du paiement au **Participant Client Payeur** et au **PI (Banque Centrale)**.
15. L'**AIP** notifie l'irrévocabilité du paiement au **Participant Client Payeur**.

16. Le **Participant Client Payeur** notifie le succès au **Client Particulier Payeur**.

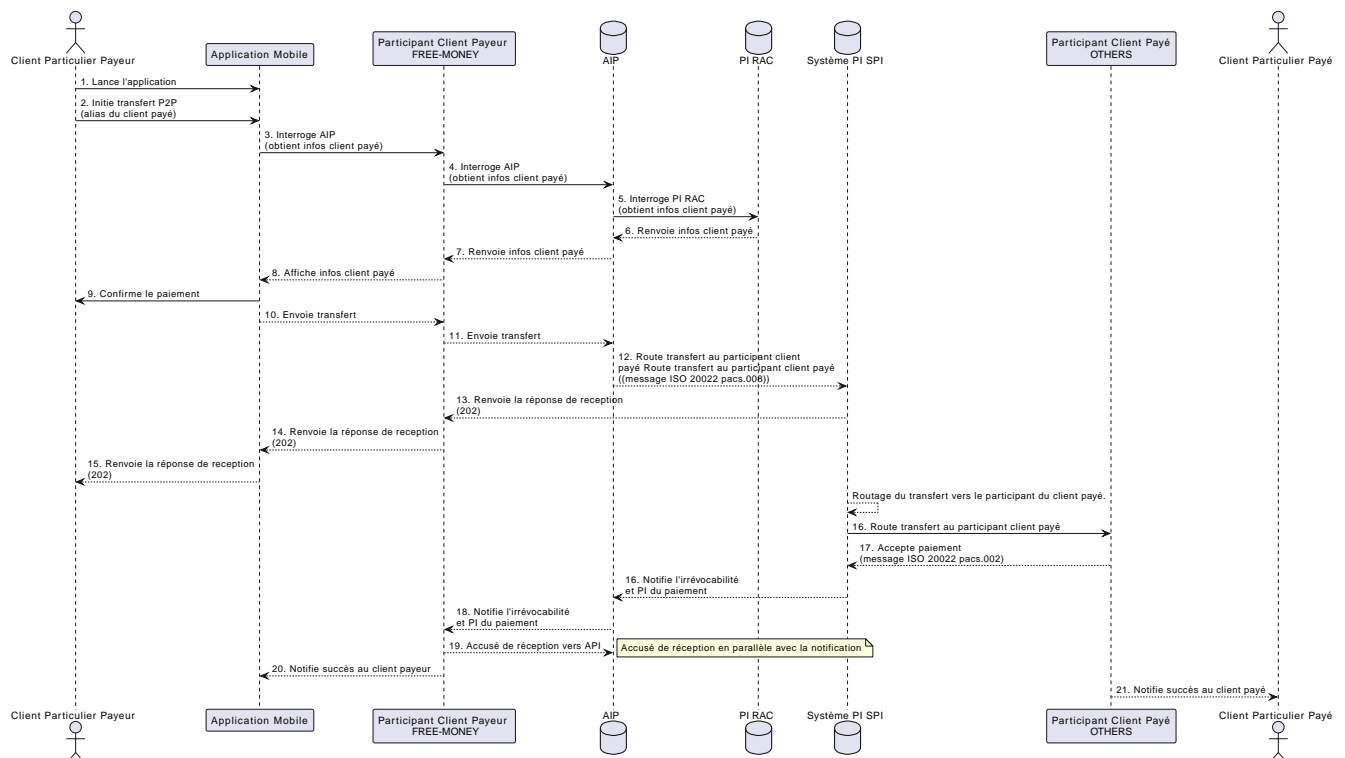
17. Le **Participant Client Payé** notifie le succès au **Client Particulier Payé**.

Ce scénario décrit le processus de transfert P2P entre un client payeur et un client payé, en utilisant le système interopérable avec la BCEAO (Banque Centrale des États de l'Afrique de l'Ouest).

- Flow simplifié



- Flow détaillés

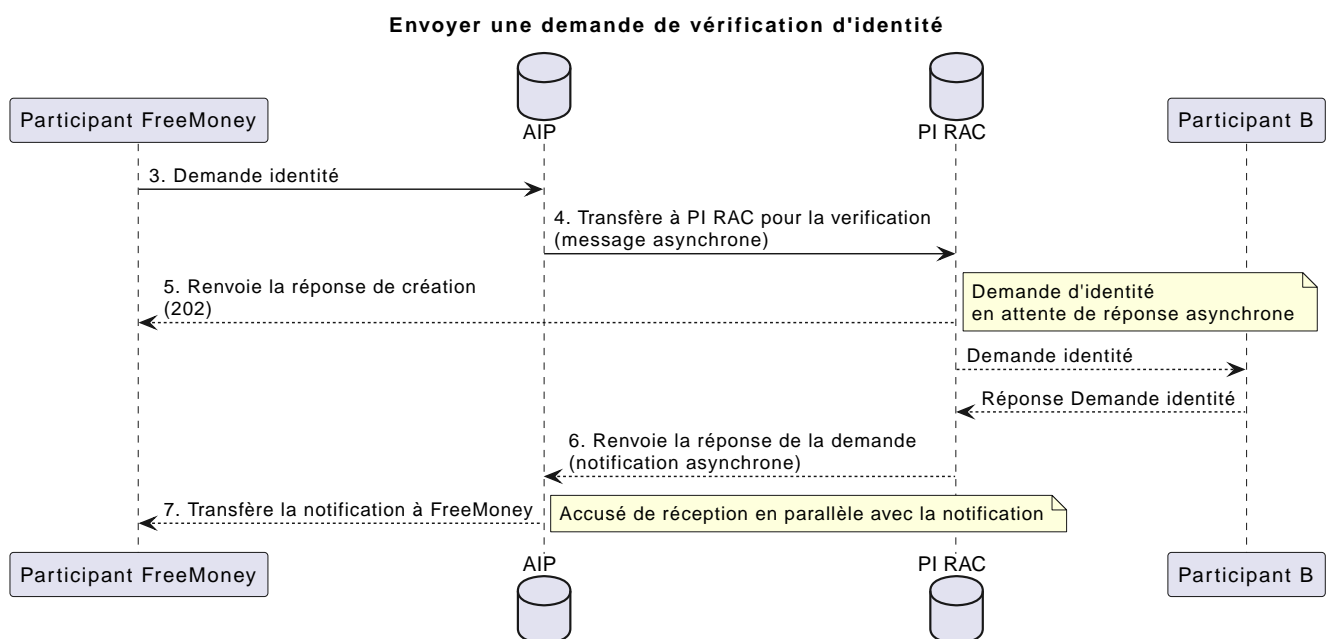


2.2.6. SEQ-006 : Envoyer une demande de vérification d'identité

1. Le Participant FreeMoney initie le processus en envoyant une demande de vérification d'identité à AIP.
2. AIP transfère la demande à PI RAC pour effectuer la vérification d'identité de manière asynchrone.
3. PI RAC reçoit la demande et donne une réponse avec un code HTTP 202.
4. En parallèle, PI RAC envoie une demande d'identité à d'autres participants concernés.
5. Le participant requis, répondent à la demande d'identité de PI RAC.
6. PI RAC reçoit la réponse du participant.
7. PI RAC envoie la réponse de la demande d'identité à AIP.
8. AIP transfère la notification de réponse à FreeMoney.

Ce processus garantit une vérification d'identité sécurisée et asynchrone, permettant une communication fluide entre les participants du système.

- Flow détaillés



2.2.7. SEQ-007 : Envoyer une réponse à une demande de vérification d'identité

- Description textuelle

1. L'API reçoit une demande d'identité de la part de PI RAC.
2. L'AIP transfère la demande d'identité au Participant FreeMoney.
3. Le Participant FreeMoney reçoit la demande d'identité.
4. Le Participant FreeMoney transfère la réponse identité ver l'AIP.
5. L'AIP' transfère la réponse identité vers PI RAC.

- Flow détaillés

Envoyer une réponse à une demande de vérification d'identité

