

Application de Gestion des Voyages

I. Étude de Domaine

1. Introduction et Contexte Général

1.1. Contexte Global du Secteur

Le secteur du tourisme et des voyages constitue l'un des piliers de l'économie mondiale. Il intègre une grande diversité d'activités allant de la réservation de vols et d'hébergements à la planification d'itinéraires sur mesure. Dans ce contexte, les agences de voyages jouent un rôle essentiel en servant d'intermédiaire entre les fournisseurs de services (compagnies aériennes, hôtels, tour-opérateurs) et une clientèle de plus en plus exigeante.

1.2. Objectifs de l'Étude

Cette étude de domaine vise à :

- Comprendre les spécificités et les évolutions du marché touristique.
- Identifier les besoins et attentes des agents de voyages ainsi que les problématiques actuelles.
- Définir les axes de valorisation d'une application desktop centralisée pour améliorer la gestion des réservations et optimiser l'expérience client.
- Proposer une synthèse des exigences et contraintes qui serviront de socle à la solution envisagée.

2. Analyse du Secteur du Voyage

2.1. Évolution Historique et Tendances Actuelles

- a. **Historique :** Le tourisme s'est développé au fil des décennies, passant d'une offre essentiellement physique à une offre digitalisée, favorisée par l'essor d'Internet.

- b. **Tendances actuelles :**

- **Digitalisation :** Les plateformes en ligne et les applications mobiles transforment les modes de réservation.
- **Personnalisation :** La demande pour des voyages sur mesure et adaptés aux préférences individuelles se fait croissante.

- **Durabilité et éthique** : Les préoccupations environnementales et la recherche d'un tourisme responsable influencent désormais les choix des voyageurs.

2.2. Rôle et Importance des Agences de Voyages

- **Expertise** : Elles fournissent un service personnalisé et un accompagnement dans la planification de séjours complexes.
- **Confiance** : En centralisant des informations fiables, elles assurent la sécurité et la transparence des réservations.
- **Intermédiaires stratégiques** : Elles facilitent les relations entre les clients et les multiples prestataires du secteur, jouant un rôle de régulateur et de médiateur.

3. Analyse des Besoins et Attentes des Agents de Voyages

3.1. Gestion Efficace des Réservations

- **Centralisation des Données** : Les agents requièrent une interface unique qui regroupe toutes les informations relatives aux voyages, clients et réservations.
- **Accessibilité en Temps Réel** : La possibilité de mettre à jour et de consulter instantanément les informations est essentielle pour éviter toute incohérence.
- **Simplicité d'Utilisation** : Un outil ergonomique permet de réduire les erreurs et la charge administrative.

3.2. Amélioration de la Communication avec les Clients

- **Suivi Personnalisé** : Chaque client doit bénéficier d'un suivi adapté, tenant compte de son historique et de ses préférences.
- **Notifications et Rappels** : Un système automatisé de rappels (dates de départ, modifications, confirmations de paiement) contribue à une meilleure gestion et à une relation client renforcée.

3.3. Centralisation et Harmonisation des Outils

- **Élimination de la Fragmentation** : L'utilisation d'outils disparates est source de confusion et de perte de temps ; une application centralisée est une réponse efficace.
- **Interopérabilité** : Faciliter l'échange d'informations entre divers modules (réservations, voyages, facturation) améliore la cohérence des données.

4. Défis Actuels dans la Gestion des Voyages

4.1. Fragmentation et Multiplicité des Solutions Existantes

- **Outils Disparates** : De nombreuses agences se retrouvent à utiliser plusieurs logiciels non connectés, ce qui complique le suivi des réservations et la mise à jour des informations.
- **Interfaces Peu Intuitives** : La complexité des systèmes actuels demande une formation accrue et augmente le risque d'erreurs.

4.2. Problèmes de Communication et Synchronisation

- **Délais de Mise à Jour** : L'absence d'un système centralisé conduit souvent à des décalages dans la transmission des informations.
- **Communication avec les Prestataires** : Des problèmes de synchronisation entre l'agence et les partenaires (hôtels, compagnies aériennes) peuvent engendrer des conflits et des erreurs dans la réservation.

4.3. Impacts sur la Satisfaction Client

- **Insatisfaction et Perte de Confiance** : Des erreurs répétées ou des délais de réponse trop longs peuvent nuire à l'image de l'agence et faire perdre des clients.
- **Manque de Personnalisation** : L'impossibilité de disposer d'un historique complet et centralisé limite la capacité à adapter les offres aux besoins spécifiques des clients.

5. Identification de la Problématique et Besoins des Clients

5.1. Problématique Identifiée

Les agences de voyages se trouvent confrontées à un environnement où la fragmentation des outils de gestion et une communication inefficace compliquent la coordination des réservations. Cela entraîne :

- Des erreurs de saisie et de mise à jour.
- Une expérience client dégradée.
- Une perte de temps et une augmentation des coûts opérationnels.

5.2. Besoins Concrets des Clients et des Agences

- **Centralisation des Réservations** : Accès unique à l'ensemble des informations clients et voyages pour un suivi rigoureux.
- **Interface Dynamique** : Capacité à adapter les formulaires en fonction du type de réservation (vol ou voyage organisé).
- **Fiabilité et Rapidité** : Mise à disposition d'un outil qui minimise les délais de traitement et réduit les risques d'erreurs.

- **Personnalisation de l'Offre** : Possibilité d'offrir des services sur mesure en s'appuyant sur un historique client complet et détaillé.

6. Proposition de Valeur et Avantages de l'Application

6.1. Présentation de la Solution

L'application desktop proposée vise à centraliser l'ensemble des processus de gestion des voyages et des réservations en offrant une interface intuitive et réactive. Elle permettra aux agents de :

- Gérer efficacement les opérations CRUD (Création, Lecture, Mise à jour, Suppression) pour les voyages et les clients.
- Bénéficier d'un système de notifications et de rappels pour assurer le suivi des échéances importantes.
- Adapter dynamiquement les formulaires en fonction des besoins spécifiques du client (vol ou voyage organisé).

6.2. Avantages pour l'Agence et pour les Clients

- **Optimisation des Processus** : Réduction du temps passé sur la gestion administrative grâce à l'automatisation des tâches répétitives.
- **Amélioration de l'Expérience Client** : Un suivi personnalisé et une communication instantanée renforcent la satisfaction et la fidélisation.
- **Réduction des Coûts** : Une application centralisée limite les besoins en maintenance de multiples outils et simplifie les mises à jour.
- **Adaptabilité et Scalabilité** : La solution pourra évoluer en intégrant de nouvelles fonctionnalités (facturation, gestion documentaire, reporting) en fonction des besoins futurs.

7. Fonctionnalités Clés de l'Application

7.1. Gestion des Voyages et Réservations

- **Opérations CRUD** : Permettre la création, modification, suppression et consultation des voyages avec une recherche optimisée par destination.

7.2. Formulaires Dynamiques et Adaptatifs

- **Adaptation au Type de Réservation** : Modification des champs de saisie en fonction du type de produit réservé (vol ou voyage organisé), avec des informations spécifiques telles que les horaires de vol ou les détails d'hébergement.

- **Validation Automatique** : Contrôles de saisie pour garantir la cohérence et la fiabilité des données (formats de date, champs obligatoires, etc.).

7.3. Tableaux de Bord et Reporting

- **Indicateurs Clés** : Visualisation des statistiques importantes (nombre de réservations, taux de conversion, indicateurs de satisfaction).
- **Rapports Personnalisables** : Génération de rapports pour analyser l'activité de l'agence et identifier les axes d'amélioration.

8. Exigences et Contraintes

8.1. Exigences Fonctionnelles

- **Cas d'Utilisation :**
 - Gestion complète des voyages et des clients via des opérations CRUD.
 - Système de recherche avancée permettant de filtrer par destination, type de réservation, ou date.
- **Flux de Travail :**
 - Processus de saisie simplifié pour accélérer les réservations.
 - Adaptation dynamique des formulaires en fonction des choix de l'utilisateur.

8.2. Exigences Non Fonctionnelles

- **Ergonomie et Interface** : Une interface intuitive et ergonomique, conçue pour une prise en main rapide par des utilisateurs non spécialisés.
- **Performance** : Temps de réponse optimisés, même en cas de forte charge de données.
- **Sécurité** : Accès sécurisé par authentification et gestion des autorisations pour éviter les modifications non autorisées.
- **Maintenance** : Facilité de mise à jour et d'extension du système pour intégrer de futures évolutions.

8.3. Contraintes Techniques et Organisationnelles

- **Environnement de Développement** : Utilisation de Java Swing dans un IDE compatible, garantissant une application desktop stable et performante.
- **Compatibilité** : L'application doit fonctionner sur les systèmes d'exploitation courants (Windows, Linux, éventuellement macOS).
- **Support et Formation** : Prévoir des modules de formation pour les agents et un support technique pour assurer une adoption rapide et efficace de la solution.

9. Analyse Comparative et Évaluation des Solutions Existantes

9.1. Panorama des Outils Actuels

- **Logiciels Multiples** : De nombreuses agences utilisent des outils spécialisés qui, s'ils offrent une grande variété de fonctionnalités, manquent souvent d'intégration et de simplicité d'utilisation.
- **Coûts et Complexité** : Les solutions existantes peuvent se révéler coûteuses et nécessiter une courbe d'apprentissage importante pour les utilisateurs.

9.2. Positionnement de la Nouvelle Application

a. Avantages Compétitifs :

- **Simplicité et Rapidité** : Une solution centralisée et facile à utiliser, adaptée aux besoins des petites et moyennes agences.
- **Personnalisation** : Possibilité d'adapter l'interface en fonction des réservations (vol ou voyage organisé) et de proposer des rapports personnalisés.
- **Coût Maîtrisé** : Une application desktop qui limite les dépenses en infrastructure et en maintenance.

b. Comparaison avec la Concurrence :

L'application se positionne comme une solution « lean » et efficace, évitant la complexité excessive des logiciels concurrents tout en garantissant une robustesse et une flexibilité adaptées aux évolutions du marché.

10. Risques et Stratégies d'Atténuation

10.1. Identification des Risques

- **Risque Technique** : Problèmes de compatibilité ou de performance liés à l'environnement desktop.
- **Risque Organisationnel** : Difficultés d'adoption par des utilisateurs peu familiers avec les nouvelles technologies.
- **Risque de Sécurité** : Possibilité d'accès non autorisé aux données sensibles de l'agence.

10.2. Mesures d'Atténuation

- **Tests Rigoureux** : Mise en place d'une batterie de tests unitaires et fonctionnels pour garantir la stabilité du système.

- **Formation et Support** : Organisation de sessions de formation et mise à disposition d'un support technique dédié.
- **Sécurisation de l'Accès** : Implémentation de protocoles d'authentification et de contrôle d'accès stricts pour protéger les données.

11. Plan de Déploiement et Perspectives d'Évolution

11.1. Phases de Déploiement

- **Phase Pilote** : Déploiement initial dans un environnement restreint pour valider le fonctionnement de l'application et recueillir les premiers retours des utilisateurs.
- **Déploiement Progressif** : Extension progressive de l'application à l'ensemble de l'agence, accompagnée d'un suivi personnalisé.
- **Retour d'Expérience** : Analyse des retours utilisateurs pour identifier des axes d'amélioration et planifier des mises à jour.

11.2. Perspectives d'Évolution

- **Intégration de Modules Complémentaires** : Ajout de fonctionnalités telles que la facturation, la gestion documentaire et le reporting avancé.
- **Adaptabilité au Marché** : Mise à jour régulière de l'application pour s'adapter aux évolutions réglementaires et aux nouvelles tendances du secteur touristique.
- **Extension Multicanale** : Possibilité d'envisager une version web ou mobile pour offrir une flexibilité supplémentaire aux agents et aux clients.

12. Conclusion et Synthèse

12.1. Synthèse de l'Étude de Domaine

L'analyse approfondie du secteur des voyages et des besoins spécifiques des agences a permis d'identifier une problématique majeure liée à la fragmentation des outils de gestion et à une communication insuffisante avec les clients. Cette étude met en évidence :

- L'importance de centraliser et de simplifier les processus de réservation.
- La nécessité d'un outil adaptable, capable de répondre aux spécificités des réservations (vols vs voyages organisés).
- Les bénéfices d'une solution ergonomique et sécurisée pour améliorer la performance opérationnelle et la satisfaction client.

12.2. Impact Attendu de la Nouvelle Application

La mise en place de cette application desktop devrait permettre aux agences de voyages de :

- Optimiser leur gestion interne en centralisant toutes les données pertinentes.
- Offrir une expérience client améliorée grâce à un suivi personnalisé et des réponses plus rapides.
- Réduire les coûts opérationnels en éliminant la complexité et les redondances des outils existants.
- Se positionner avantageusement dans un marché en constante évolution, en s'appuyant sur une solution flexible et évolutive.

II. Analyse

1. La méthodologie RUP :

Le Rational Unified Process (RUP) est une méthodologie de développement logiciel itérative et incrémentale, développée par Rational Software, désormais une division d'IBM. Il vise à fournir un cadre structuré pour la production de logiciels de haute qualité, répondant aux besoins des utilisateurs finaux, tout en respectant les contraintes de délais et de budget.

A. Principes fondamentaux du RUP :

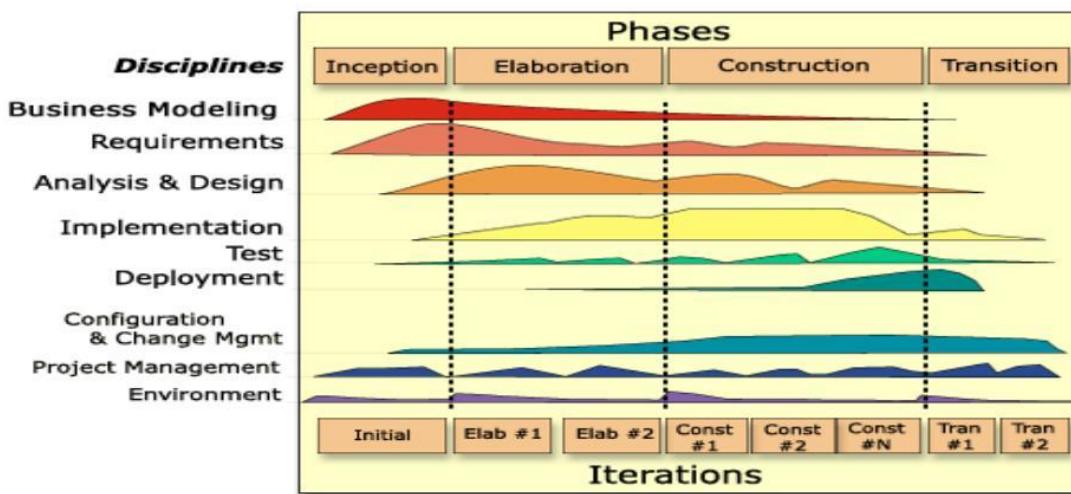
- **Approche itérative et incrémentale :** Contrairement aux modèles séquentiels traditionnels, RUP privilégie des cycles de développement courts et répétés, permettant une adaptation continue aux retours des utilisateurs et aux changements de besoins.
- **Pilotage par les cas d'utilisation :** Les cas d'utilisation (ou "use cases") servent de base pour capturer et définir les exigences fonctionnelles du système, garantissant que le développement reste centré sur les besoins réels des utilisateurs.
- **Centré sur l'architecture :** RUP met l'accent sur la définition et la validation précoce de l'architecture du système, assurant une base solide pour le développement ultérieur et facilitant la gestion de la complexité.
- **Gestion des exigences :** Une attention particulière est accordée à la collecte, à l'analyse et à la documentation des exigences, permettant de mieux comprendre les besoins des utilisateurs et de réduire les risques liés aux changements ultérieurs.
- **Vérification de la qualité :** RUP intègre des activités de test tout au long du cycle de développement, assurant que le produit final répond aux standards de qualité requis.
- **Gestion des changements :** La méthodologie prévoit des mécanismes pour gérer efficacement les modifications, qu'elles proviennent des évolutions des exigences ou des retours des parties prenantes.

2. Phases du RUP :

Le cycle de vie d'un projet RUP est divisé en quatre phases principales, chacune comportant une ou plusieurs itérations :

Rational Unified Process (RUP)

Detailed Diagram



- **Inception (Création)** : Définir la portée du projet, identifier les principales exigences et obtenir l'accord des parties prenantes.
- **Élaboration** : Analyser en profondeur les exigences, élaborer l'architecture du système et évaluer les risques majeurs.
- **Construction** : Développer le produit en construisant les composants et en intégrant les fonctionnalités, tout en effectuant des tests réguliers pour assurer la qualité.
- **Transition** : Déployer le produit chez les utilisateurs finaux, assurer la formation nécessaire et effectuer les ajustements post-déploiement.

B. Disciplines du RUP :

RUP structure les activités de développement en neuf disciplines principales :

- **Modélisation des besoins** : Capturer et gérer les exigences du système.
- **Analyse et conception** : Définir la solution technique répondant aux besoins.
- **Implémentation** : Coder et intégrer les composants du système.
- **Test** : Vérifier et valider que le système fonctionne conformément aux spécifications.
- **Déploiement** : Livrer le système aux utilisateurs finaux et assurer son installation.
- **Gestion de projet** : Planifier, suivre et contrôler le projet pour respecter les objectifs de coût, de délai et de qualité.

- **Gestion de la configuration et des changements** : Gérer les différentes versions des artefacts produits et contrôler les modifications.
- **Environnement** : Fournir les outils et les processus nécessaires au développement.
- **Gestion des risques** : Identifier, analyser et atténuer les risques tout au long du projet.

C. Avantages du RUP :

- **Flexibilité et adaptabilité** : Grâce à son approche itérative, RUP permet d'ajuster le développement en fonction des retours et des changements de contexte.
- **Réduction des risques** : L'identification et la gestion proactive des risques dès les premières phases minimisent les surprises en cours de projet.
- **Amélioration de la qualité** : Les tests continus et l'accent mis sur l'architecture assurent un produit final robuste et conforme aux attentes.

D. Limites du RUP :

- **Complexité** : La richesse et la formalisation du processus peuvent le rendre complexe à mettre en œuvre, notamment pour les petites équipes ou les projets de moindre envergure.
- **Ressources nécessaires** : La mise en place efficace de RUP peut nécessiter des ressources humaines et matérielles significatives.

3. Outils et Approches de Conception :

A. Utilisation de Visual Paradigm

Visual Paradigm est un outil très utile pour la modélisation UML. Voici comment il aide dans le processus :

- **Création de diagrammes clairs** : Vous pouvez dessiner des diagrammes de cas d'utilisation, des diagrammes de classes, des diagrammes de séquence, etc. Cela aide l'équipe à visualiser la structure du projet.
- **Collaboration** : L'outil facilite le travail en équipe grâce à des espaces collaboratifs.
- **Génération de documentation** : Certains diagrammes peuvent être utilisés pour générer automatiquement du code ou de la documentation, ce qui vous fait gagner du temps.

B. L'Approche AGL

Ici, l'AGL désigne une approche dans laquelle vous centralisez certaines logiques ou composants via des design patterns.

- **Exemple spécifique** : L'utilisation du Singleton qui permet de s'assurer qu'une classe n'a qu'une seule instance pour gérer des ressources critiques (comme une connexion à une base de données ou l'accès à une configuration).

4. Pattern MVC (Modèle-Vue-Contrôleur) :

L'approche MVC permet de séparer clairement les responsabilités dans votre application. Prenons le temps d'expliquer chaque couche :

A. Le Modèle (Model)

- Rôle : Gérer les données, la logique métier, et les règles de l'application.
- Fonctionnement : C'est la base de données ou l'ensemble des objets métiers.
- Avantage : L'indépendance vis-à-vis de l'interface utilisateur facilite la maintenance et les tests.

B. La Vue (View)

- Rôle : Afficher les données aux utilisateurs.
- Fonctionnement : Elle récupère les informations du modèle via le contrôleur pour les présenter sous une forme adaptée.
- Avantage : On peut modifier l'interface sans affecter la logique métier.

C. Le Contrôleur (Controller)

- Rôle : Faire le lien entre la vue et le modèle.
- Fonctionnement : Il reçoit les interactions de l'utilisateur (clics, saisie) et demande au modèle les données à afficher ou à modifier.
- Avantage : Facilite la répartition des rôles entre les différentes parties du code.

5. Mise en œuvre de Design Patterns : Focus sur le SingletonPourquoi le Singleton ?

- **Fonctionnement :** Ce pattern assure qu'une classe ne peut être instanciée qu'une seule fois et offre un point d'accès global à cette instance.
- **Utilité dans votre application :** Par exemple, pour centraliser la gestion d'une connexion à la base de données ou pour gérer une configuration globale.
- **Points à surveiller :**
 - Avantage : Simplicité d'accès à une ressource partagée.
 - Inconvénient : Peut rendre certains tests unitaires plus difficiles si le singleton est fortement couplé dans l'application.

6. Diagrammes UML pour la Documentation et la Communication :

A. Diagramme de cas d'utilisation global

Prenez le temps de représenter :

Les acteurs : Agents de voyages, clients .

- Gestion des clients (création, modification, suppression, consultation).
- Reporting et suivi des activités.

Chaque cas d'utilisation devra montrer comment chaque type d'utilisateur interagit avec le système.

B. Focus sur la gestion des clients (premier cas d'utilisation)

Pour ce cas d'utilisation, procédez de la manière suivante :

1. Phase d'Inception

- Identifier les acteurs : Par exemple, l'agent de voyages et le client.
- Décrire le besoin : Permettre la gestion (création, modification, consultation) des informations clients.

2. Phase d'Élaboration

- Diagramme de cas d'utilisation spécifique : Dessinez les interactions spécifiques à la gestion des clients.
- Diagramme de classes : Montrez comment les entités (Client, Adresse, etc.) sont reliées.
- Diagramme de séquence : Illustrer pas à pas comment se déroule, par exemple, l'ajout d'un nouveau client.

3. Phase de Construction

- Implémentation et tests : Développez la fonctionnalité en vous assurant que chaque itération est correctement testée (tests unitaires, tests d'intégration).

4. Phase de Transition

- Déploiement et retours utilisateurs : Mettez la fonctionnalité en production, formez les utilisateurs et adaptez en fonction des retours.

C. Exemple de diagramme de séquence pour la gestion des clients

Imaginez le scénario suivant :

Étapes :

1. L'agent saisit les informations d'un nouveau client.
2. L'interface (Vue) envoie ces informations au Contrôleur.
3. Le Contrôleur valide les données, puis demande au Modèle de créer le nouveau client.
4. Le Modèle transmet la requête au service de persistance qui effectue l'enregistrement.
5. Une fois le client créé, la confirmation est renvoyée du Modèle au Contrôleur puis à la Vue pour informer l'agent.

7. Application de RUP sur le projet :

Le projet a été piloté en suivant les principes et les phases de la méthodologie RUP, permettant une approche itérative et incrémentale. Voici comment RUP a été mis en œuvre tout au long du projet :

3.1. Phase d'Inception (Inception) :

- **Objectif** : Définir la vision du projet, établir le business case et identifier les parties prenantes.
- **Activités réalisées** :
 - **Recueil des besoins** : Réunions avec les utilisateurs finaux et les responsables métier afin de déterminer les fonctionnalités principales (gestion des voyages, clients, réservations, etc.).
 - **Définition de la portée** : Clarification des objectifs du projet, définition des limites de l'application et identification des risques majeurs.
 - **Architecture préliminaire** : Ébauche d'un diagramme d'architecture global intégrant la séparation des couches (Vue, Contrôleur, Modèle, DAO, Base de Données).
- **Livrables** :
 - **Document de vision et business case**.
 - **Diagrammes préliminaires (cas d'utilisation, architecture système)**.

3.2. Phase d'Élaboration (Elaboration) :

- Objectif : Définir l'architecture détaillée du système, affiner l'analyse des besoins et réduire les risques techniques.
- Activités réalisées :
 - Analyse fonctionnelle détaillée : Modélisation complète des cas d'utilisation, identification des acteurs et des scénarios principaux.
Par exemple, le cas d'utilisation « Ajouter un voyage » a été précisément décrit, incluant l'interaction entre la Vue (IHM), le Contrôleur (VoyageController) et la couche de persistance (VoyageDAO).
 - Architecture technique : Conception d'un diagramme de classes et de séquence intégrant le modèle MVC pour le traitement et la persistance des données.
 - Prototype fonctionnel : Développement rapide d'un prototype pour valider les principaux flux d'interaction (comme la création, la modification et la suppression d'un voyage).

- Livrables :
 - Spécifications fonctionnelles et techniques.
 - Modèles UML (diagramme de classes, diagramme de séquence, diagramme de cas d'utilisation).
 - Prototype de validation de l'architecture et des flux de données.

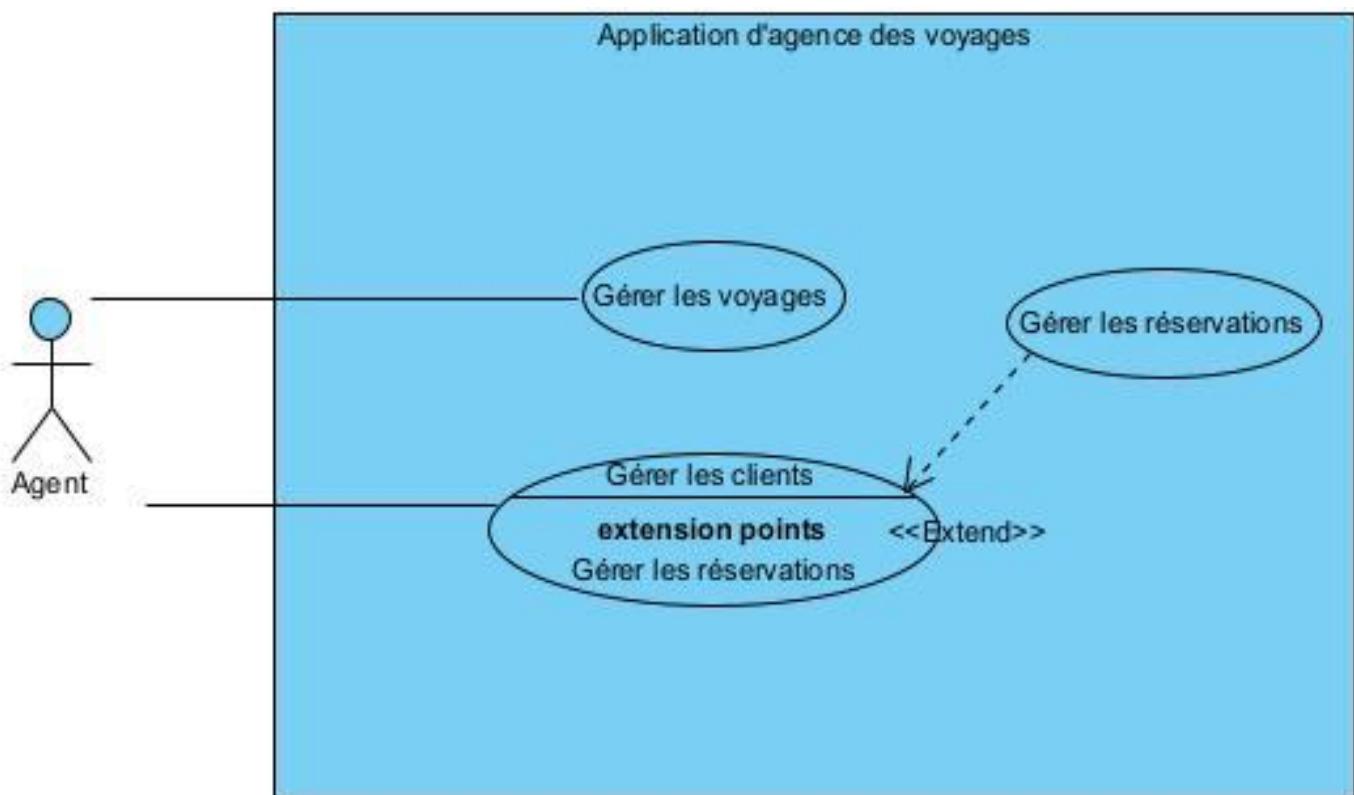
III. Conception

Diagramme de cas d'utilisation

- Le diagramme met en avant les interactions principales entre l'acteur et le système. Il inclut les cas d'utilisation mentionnés ci-dessus.

Acteurs : Agent.

- Cas :**
 - Ajouter un client
 - Modifier un client
 - Supprimer un client
 - Lister les clients
 - Rechercher par CIN

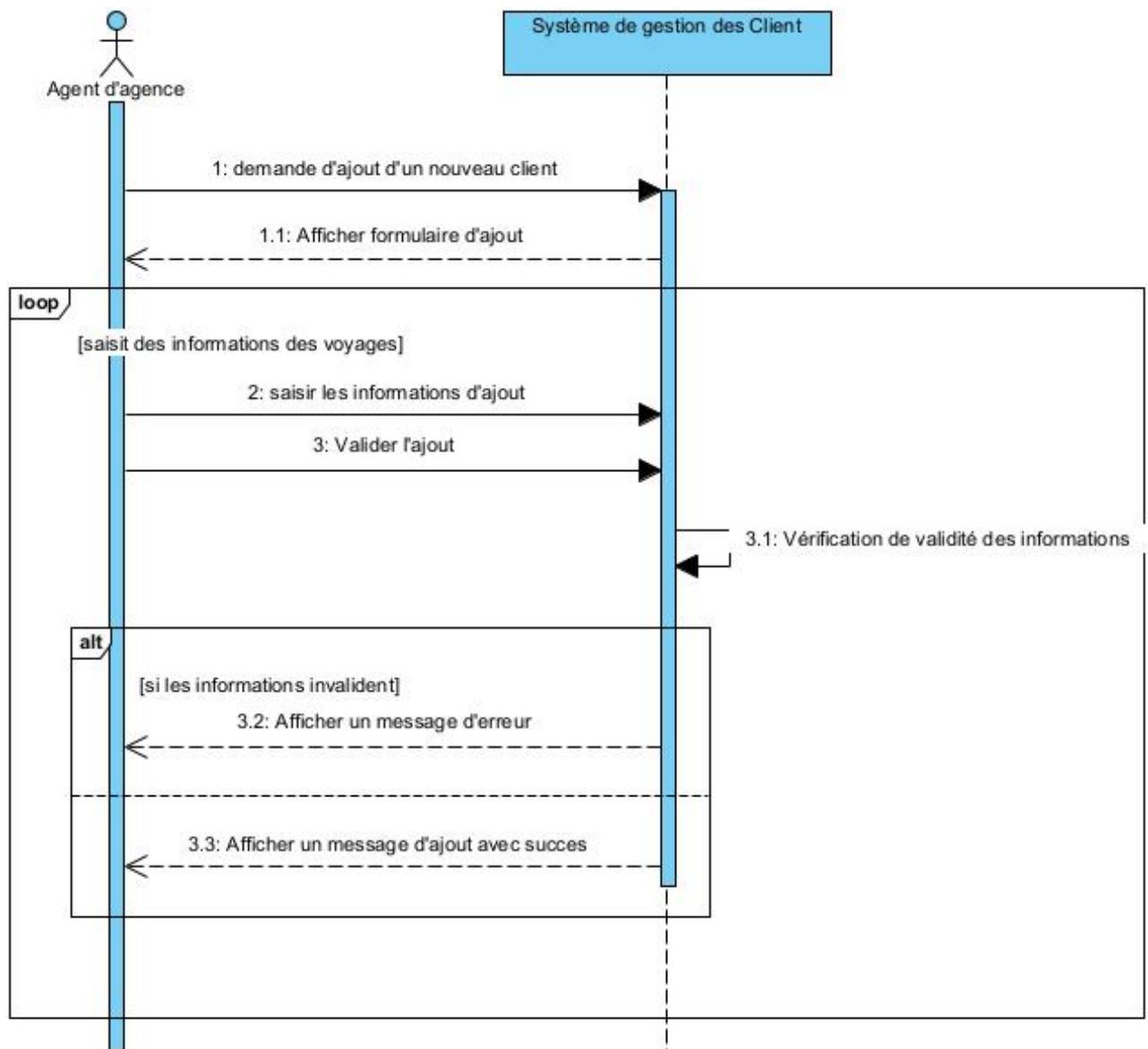


1. Diagramme de séquence système de UC1 :

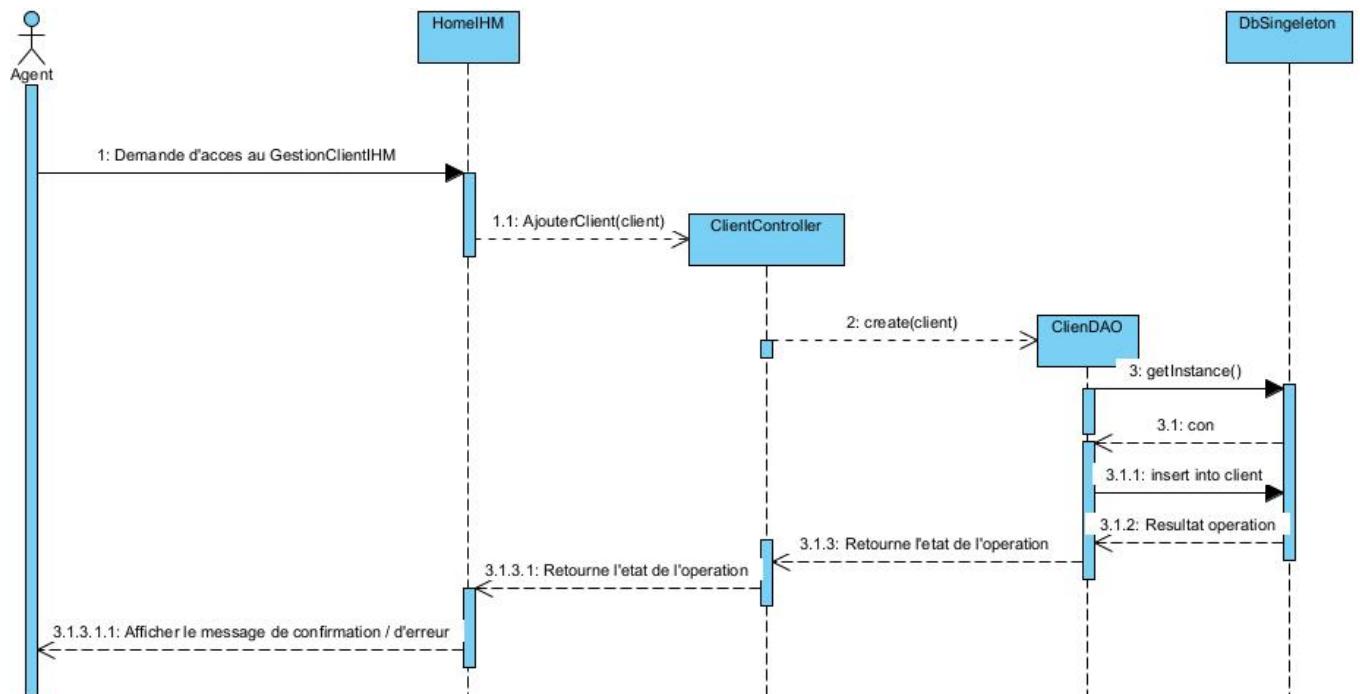
Pour le cas d'utilisation de l'itération :

1. Spécifier le flots d'événement (Flow of events)
2. Générer le diagramme de séquence
3. Spécifier les besoin IHM par un diagramme User Interface

Flow of Events



3. Diagramme de Séquence Détailé UC1 :

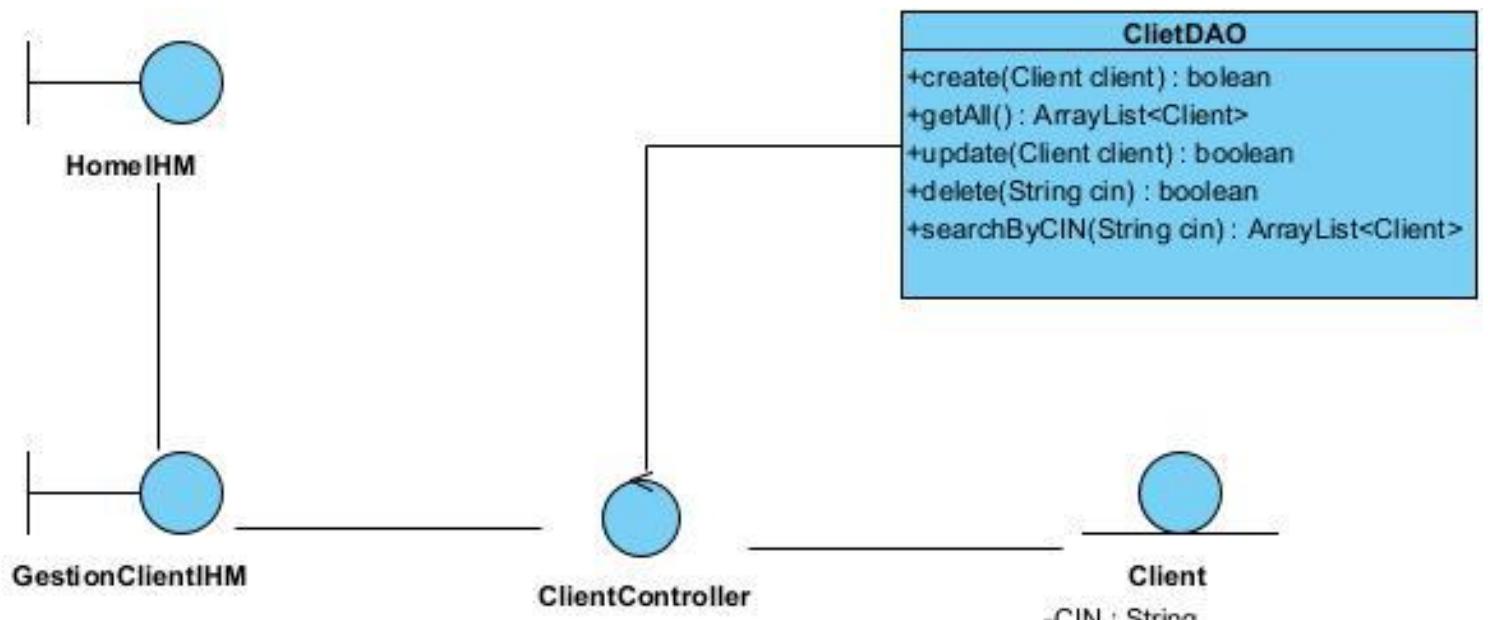


4. Diagramme de Classes Participantes UC1:

- Classes :** Client, ClientController, ClientDAO, GestionClientIHM.

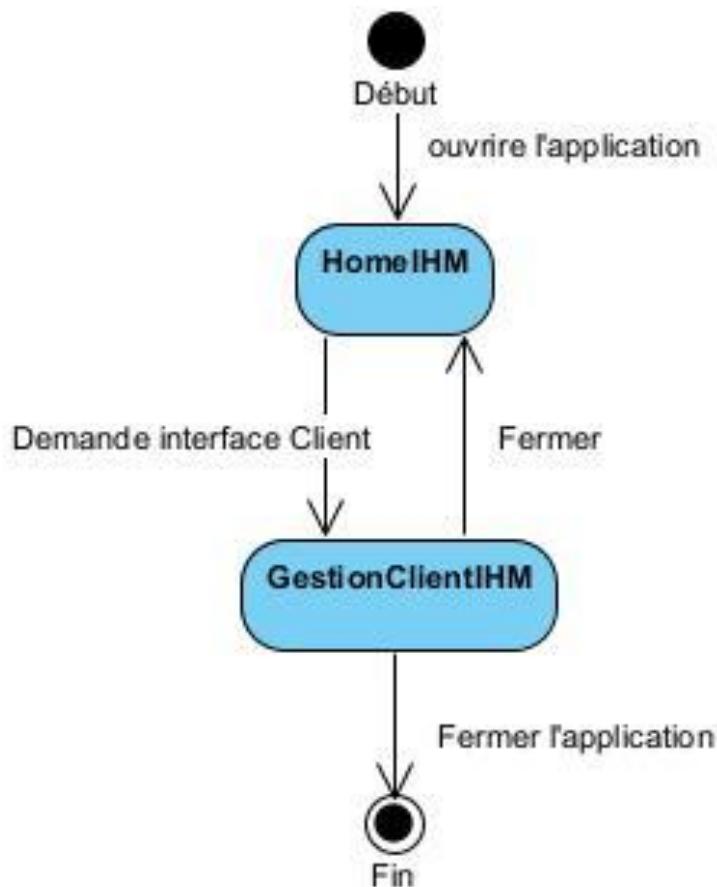
- Relations :**

- ClientController dépend de ClientDAO.
- ClientDAO manipule Client.

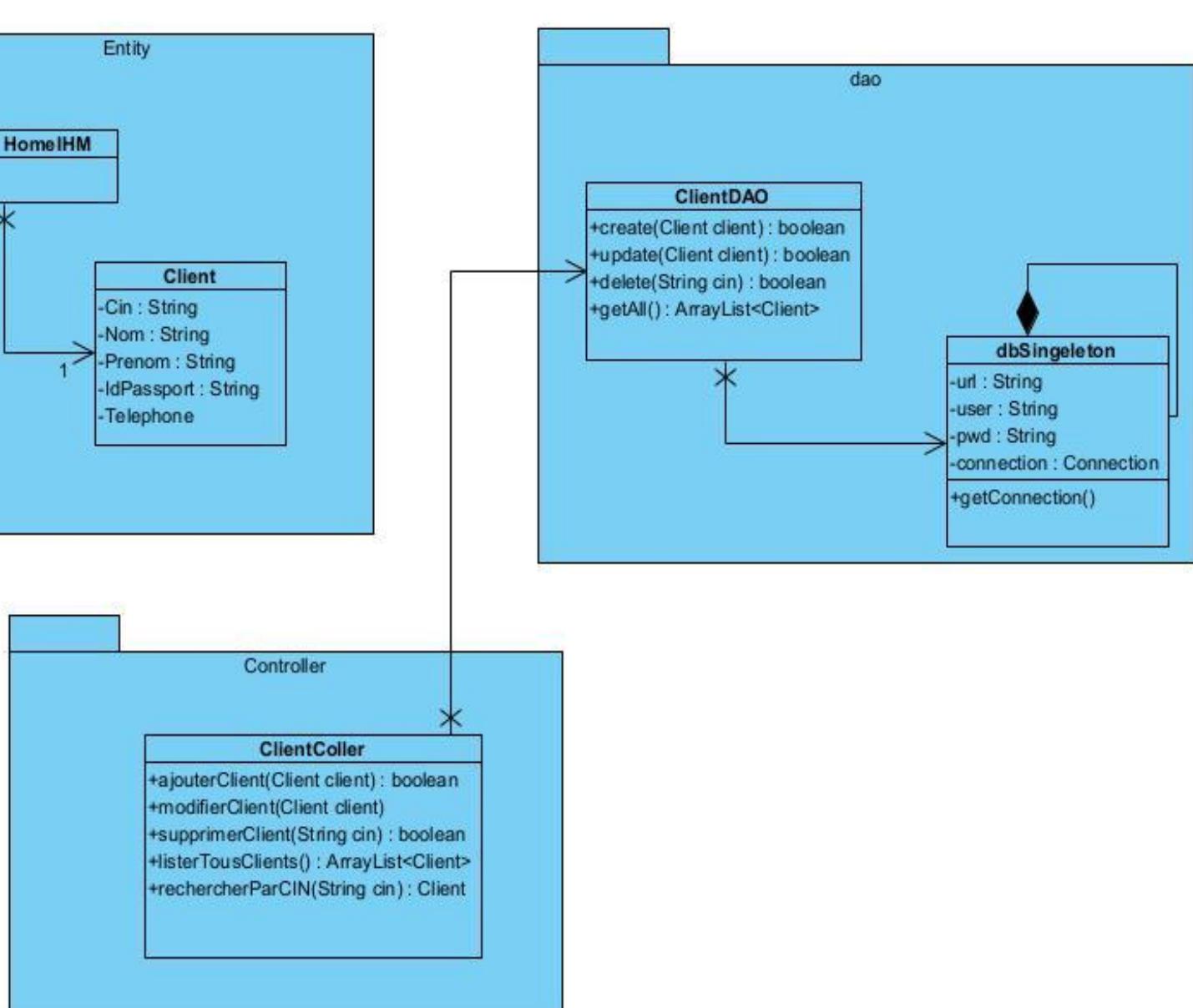


5. Diagramme de Navigation UC1 :

zFlux : IHM -> Contrôleur -> DAO -> Base de données



6. Diagramme de classe UC1:



5. Capture d'Écran de l'IHM Gestion Clients UC1:

Design Preview [GestionClientIHM]

Gestion Clients

Formulaire à remplir :

CIN :	<input type="text"/>
Nom :	<input type="text"/>
Prénom :	<input type="text"/>
id Passport :	<input type="text"/>
Telephone :	<input type="text"/>

Tableau de gestion :

CIN	Nom	Prenom	Passport	Téléphone

Actions :

Chercher par CIN : Veillez saisir le CIN du client

USE CASE 2 : gestion des voyages

1. Diagramme de sequence UC2:

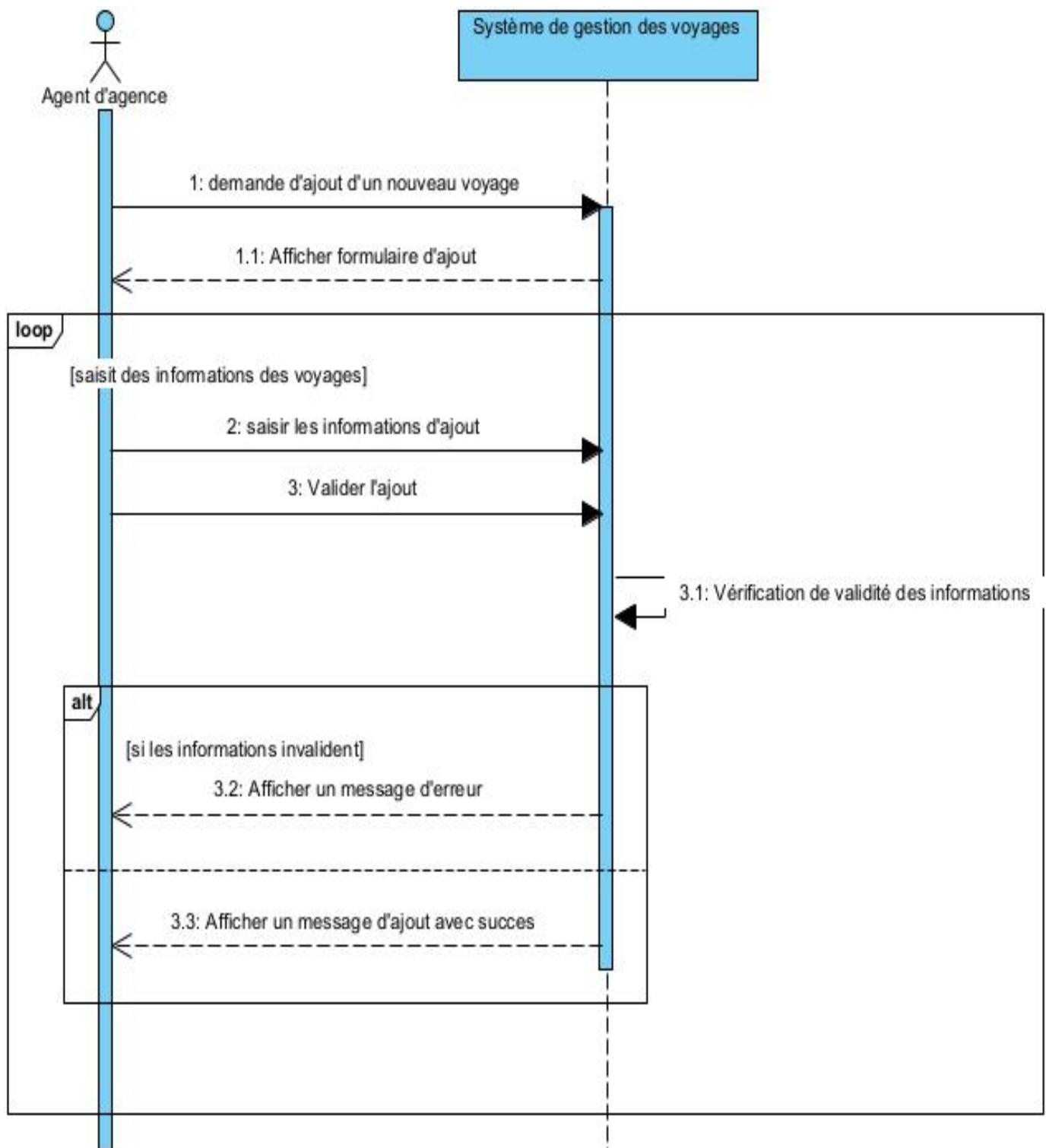
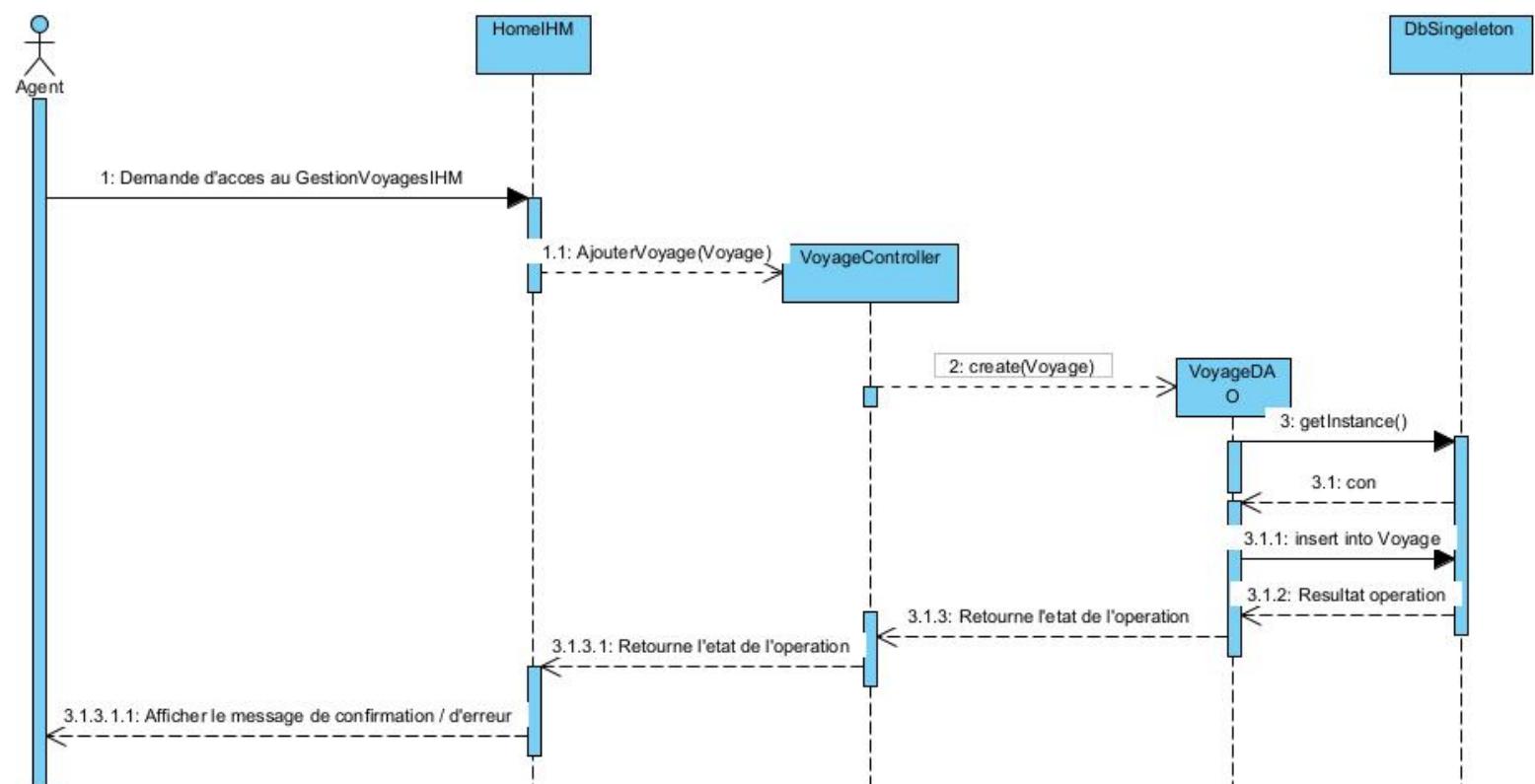
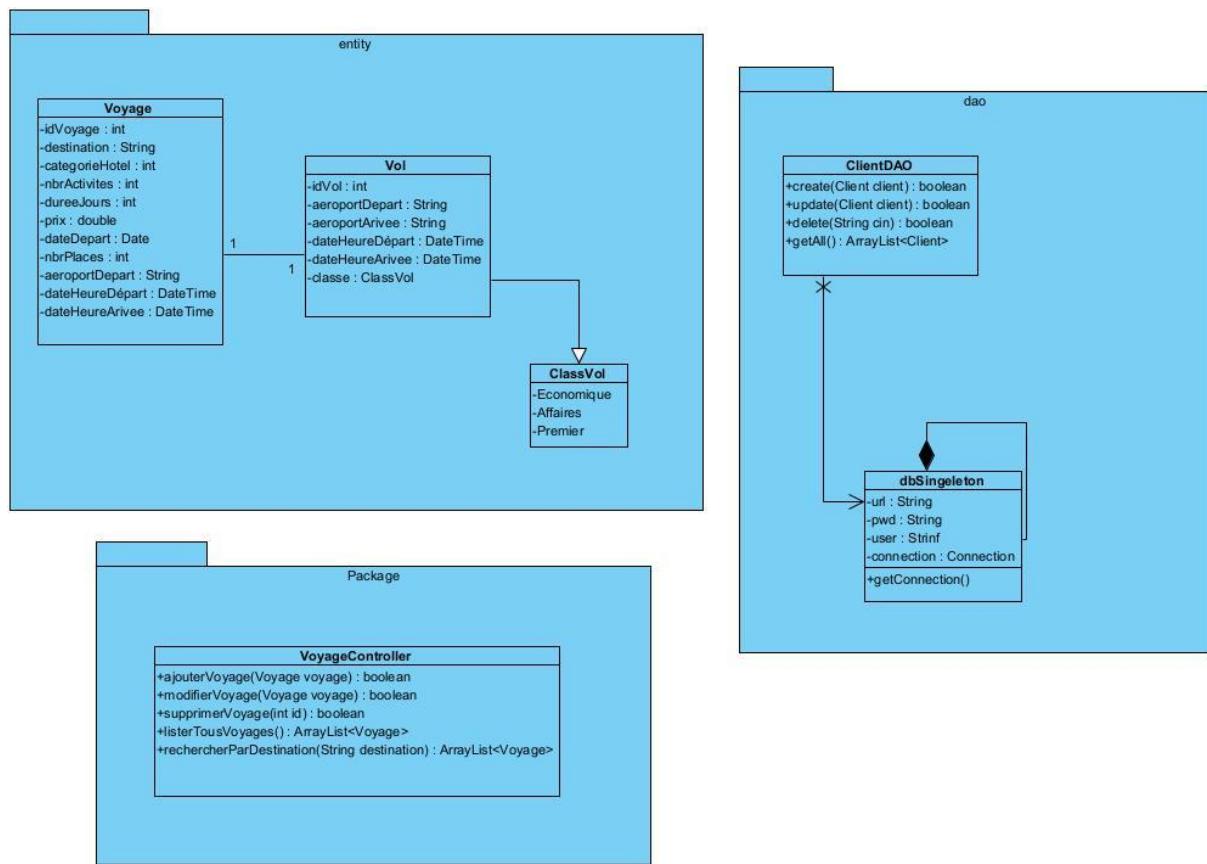


Diagramme de sequence detailler UC2 :



4. Diagramme de Classes UC2:



5. Capture d'Écran de l'IHM Gestion Voyages UC2:

Design Preview [GestionVoyagesIHM]

Gestion Voyages

Formulaire à remplir :

id	destination	prix	aeroport_depart	aeroport_arrivee	nbr_places

Destination :

Prix :

Aéroport de départ :

Aéroport d'arrivée :

Nombre des places :

Ajouter **Modifier** **Supprimer**

Chercher Voyages : Veuillez saisir une destination

IHM de d'Acceil :

Design Preview [HomeIHM]

Gestion d'agence de voyage



Gestion des voyages



Gestion des clients