



## **RAPPORT DE PROJET CAS PRATIQUE :**

### **Réservation d'Hôtel et Bus Touristique**

#### **Solution :**

**Développement d'une Plateforme Web**

## **PRÉSENTÉ PAR :**

**Abdoulaye Gueye**

**N004AB020181**

**Khadim Sow**

**N0083E920181**

**Sur la supervision de Mr Lamine Faty**

# Introduction :

Avec l'évolution du numérique, les habitudes des voyageurs ont changé. Les plateformes de réservation en ligne permettent aux utilisateurs de planifier leurs séjours sans intermédiaire. Ce projet vise à concevoir une application web permettant de **réserver des chambres d'hôtel plus des réservations de restaurant** et des **places dans des bus touristiques**, de manière intégrée

## Contexte et Cahier de charge de ce Projet :

### 1. Contexte général

Le secteur du tourisme connaît une digitalisation rapide. Les touristes souhaitent réserver leurs chambres d'hôtel et circuits touristiques de façon centralisée, rapide et sécurisée. Ce projet vise à créer une **plateforme de réservation intégrée**, accessible 24h/24, destinée à la gestion simultanée :

- des **hôtels et chambres disponibles**
  - des **bus touristiques et circuits organisés**
- 

### 2. Problématique

Dans le contexte actuel du tourisme numérique, les voyageurs cherchent de plus en plus à planifier eux-mêmes leurs séjours sans passer par des agences physiques. Cela implique souvent de visiter **plusieurs plateformes distinctes** pour réserver séparément :

- leur **hébergement** (hôtels, auberges, locations saisonnières),
- puis leurs **activités ou circuits touristiques**, comme les excursions en bus, les visites guidées, etc.

Or cette **fragmentation** de la réservation présente plusieurs **inconconvénients majeurs** :

#### ► 1. Perte de temps

Les touristes doivent comparer différentes offres sur plusieurs sites, remplir plusieurs fois les mêmes informations (nom, date, nombre de personnes), ce qui allonge inutilement le processus. Cela peut décourager certains utilisateurs et les pousser à abandonner leur planification.

#### ► 2. Confusion

L'utilisateur peut perdre le fil des réservations qu'il a faites, ne plus se souvenir des horaires, des noms des établissements, ou encore des moyens de transport associés. De plus, la variété des interfaces utilisateurs entre les plateformes ajoute à la confusion générale.

### ► 3. Risque de surréservation ou d'incompatibilité horaire

Il est fréquent qu'un voyageur réserve une chambre d'hôtel pour un jour précis, puis un circuit touristique qui, par manque d'information centralisée, se chevauche avec l'heure d'arrivée ou de départ de l'hôtel. Cela crée une **désynchronisation** dans l'organisation du voyage. De plus, certaines plateformes ne mettent pas à jour en temps réel la disponibilité, ce qui peut entraîner des **surréservations** ou des **demandes annulées**, souvent à la dernière minute.

---

## 3. Objectifs spécifiques

### ✓ 1. Proposer une interface claire et simple pour la recherche et la réservation

L'une des priorités du projet est de fournir une **expérience utilisateur fluide**, intuitive, et cohérente. Pour cela, l'application web devra intégrer :

- Une **barre de recherche rapide** avec filtres multicritères : lieu, date, type de chambre ou circuit, nombre de personnes, budget, etc.
- Un **affichage ergonomique** des résultats sous forme de cartes ou de listes, avec photos, prix, note, disponibilité.
- Un **système de réservation simplifié**, en quelques étapes :
  - Choix d'une offre
  - Renseignement des informations nécessaires
  - Confirmation et paiement

L'objectif est de **réduire le nombre de clics** nécessaires pour effectuer une réservation, tout en offrant **toutes les informations importantes en un seul endroit**.

---

### ✓ 2. Gérer les disponibilités des chambres et des circuits automatiquement

Pour éviter les erreurs de surréservation et garantir la fiabilité des données, la plateforme doit intégrer un **mécanisme intelligent de gestion des disponibilités**, en temps réel :

- Chaque **chambre** aura un calendrier de disponibilité. Lorsqu'une réservation est confirmée, les dates réservées deviennent automatiquement indisponibles.
- De même, chaque **circuit touristique** aura un nombre de places limité. À chaque réservation, le **nombre de sièges disponibles est décrémenté**.
- Si la capacité est atteinte, la **réservation sera bloquée**, et une notification d'indisponibilité sera affichée à l'utilisateur.
- L'administrateur peut aussi **configurer les périodes d'indisponibilité** (ex : maintenance, vacances, etc.)

Cela repose sur une **gestion centralisée** de la disponibilité dans la base de données, synchronisée avec les actions des utilisateurs.

---

### ✓ 3. Permettre la gestion des réservations (ajout, annulation, paiement)

Un module de gestion des réservations doit être intégré, permettant :

- **Ajout automatique** d'une réservation suite à une confirmation utilisateur
- **Paiement sécurisé** via une passerelle intégrée (Stripe ou PayPal)
- **Suivi des réservations** par le client dans son tableau de bord
- **Annulation manuelle** dans le respect des délais imposés
- **Statut dynamique** des réservations : en attente, confirmée, annulée, expirée

L'objectif est de **digitaliser l'ensemble du processus de réservation**, depuis la demande initiale jusqu'à la facturation, tout en offrant aux utilisateurs **plus de contrôle** sur leurs démarches.

---

### ✓ 4. Fournir un panneau de gestion complet pour les administrateurs

Les administrateurs disposeront d'un **panneau d'administration (backend)** sécurisé et complet, développé avec Laravel, pour leur permettre de gérer toutes les entités et les opérations du système.

Fonctionnalités attendues :

- **Dashboard statistique :**
  - Nombre total de réservations
  - Taux d'occupation des hôtels
  - Circuits les plus populaires
  - Revenus mensuels / annuels
- **Gestion des contenus :**
  - Ajout, modification, suppression des hôtels, chambres, bus et circuits
  - Téléversement d'images
- **Gestion des utilisateurs :**
  - Liste des clients
  - Suppression/bannissement d'un utilisateur en cas d'abus
- **Suivi et gestion des réservations :**
  - Affichage de toutes les réservations avec filtre (par date, par statut, par utilisateur)
  - Possibilité d'annulation administrative
- **Exportation des données :**
  - Possibilité d'exporter en Excel ou PDF les réservations, les statistiques, etc.

Ce module vise à **automatiser la gestion** du système, **réduire la charge humaine**, et fournir aux administrateurs tous les outils nécessaires pour piloter l'activité touristique de manière efficace.

---

## 4. Acteurs du système

Acteur	Rôles
Visiteur	Consulter les offres, créer un compte
Client	Réserver un hôtel ou un bus, payer
Admin	Gérer utilisateurs, hôtels, circuits, statistiques

---

## 5. Fonctionnalités détaillées

### a. Fonctionnalités Client

L'utilisateur (touriste) est l'acteur principal du système. L'interface qui lui est dédiée doit être conviviale, fluide et intuitive, afin d'offrir une expérience complète allant de la découverte à la finalisation de la réservation. Voici les fonctionnalités principales :

---

#### ✓ 1. Inscription / Connexion

Le système doit permettre à tout utilisateur de créer un compte personnel en fournissant les informations suivantes :

- Nom complet
- Adresse email (unique)
- Mot de passe
- Numéro de téléphone (optionnel)
- Confirmation de l'email (via lien ou code)

#### Fonctionnalités associées :

- Authentification sécurisée (hash bcrypt + CSRF protection)
- Récupération de mot de passe
- Gestion du profil utilisateur
- Middleware `auth` Laravel pour sécuriser les pages réservées

**Objectif :** Permettre une identification fiable de chaque utilisateur et sécuriser les données personnelles liées aux réservations.

---

#### ✓ 2. Parcourir les hôtels et circuits touristiques

Une fois connecté, le client peut naviguer dans une interface claire et hiérarchisée :

- Page **Hôtels** : liste paginée des hôtels avec aperçu rapide (photo, prix, localisation, note)
- Page **Circuits touristiques** : liste des circuits disponibles (bus + destination + date + tarif)

Chaque élément peut être **cliqué** pour ouvrir une fiche de détails comprenant :

- Description détaillée
  - Services inclus
  - Galerie d'images
  - Prix par nuit (hôtel) ou par personne (circuit)
  - Disponibilité (places restantes, chambres restantes)
- 

### ✓ 3. Filtres de recherche

Pour affiner sa recherche, l'utilisateur peut utiliser des **filtres dynamiques** :

- **Par date** : dates de séjour ou de départ
- **Par lieu** : destination ou localisation géographique
- **Par type** : type de chambre (simple, double, suite), type de circuit (historique, nature, culturel)
- **Par prix** : intervalle de prix à l'aide d'un curseur
- **Par note** (optionnel) : classement par avis utilisateurs

Ces filtres doivent être implémentés via **requêtes AJAX** ou avec **Livewire** pour un rendu rapide sans rechargement de page.

---

### ✓ 4. Réservation d'une chambre ou d'un bus

Depuis la page de détails de l'offre choisie, l'utilisateur peut initier une **demande de réservation** en quelques clics :

- Saisie des informations de séjour : nombre de personnes, date d'arrivée/départ ou date du circuit
- Affichage du prix total estimé
- Confirmation des conditions générales
- Appui sur le bouton **“Réserver maintenant”**

Une vérification en backend s'assure que les places/chambres sont toujours disponibles avant d'enregistrer la réservation dans la base de données.

## b. Fonctionnalités Admin

L'administrateur est responsable de la **gestion complète** de la plateforme. Il dispose d'un **accès sécurisé** à un panneau d'administration (backend) lui permettant de piloter l'ensemble du système. L'objectif est de garantir une **gestion fluide, efficace et centralisée** de tous les éléments (utilisateurs, contenus, réservations, paiements, etc.).

---

## ✓ 1. Tableau de bord (Dashboard)

Dès sa connexion, l'administrateur accède à un **tableau de bord global** regroupant les informations clés du système en temps réel, présenté sous forme de graphiques et de widgets.

### Informations affichées :

- Nombre total de réservations (hôtel / circuit)
- Revenus générés (aujourd'hui, ce mois, cette année)
- Nombre d'utilisateurs inscrits
- Circuits les plus réservés
- Hôtels les mieux notés
- Statuts des réservations (confirmées, en attente, annulées)

**Techniquement**, cela peut être réalisé avec Laravel + une bibliothèque JS comme Chart.js pour les graphiques.

---

## ✓ 2. Gestion CRUD (Create, Read, Update, Delete) + Upload/Download

L'administrateur a un contrôle complet sur les **entités fonctionnelles** du système via une interface CRUD (Laravel Resource Controllers ou Livewire).

### a. Utilisateurs

- Afficher tous les utilisateurs (clients)
- Rechercher un utilisateur par nom/email
- Modifier ou désactiver un compte
- Supprimer un utilisateur problématique

### b. Hôtels et chambres

- Ajouter un **nouvel hôtel** avec :
  - nom, description, localisation, nombre d'étoiles
  - photos (upload d'images)
- Gérer les **chambres** :
  - type (simple, double, suite), prix, disponibilité
- Mettre à jour ou supprimer un hôtel ou une chambre

### c. Bus et circuits touristiques

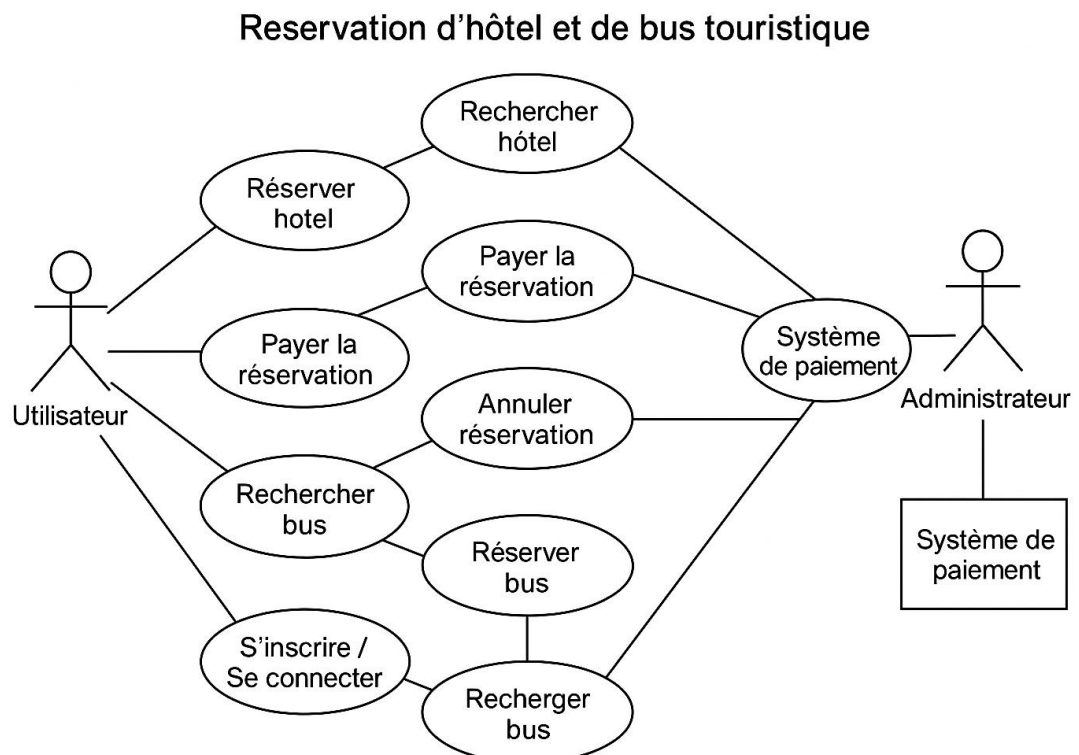
- Créer un **nouveau bus touristique** : nom, capacité, image
- Créer un **circuit touristique** associé à un bus :
- Modifier/supprimer les circuits obsolètes
- destination, date de départ, durée, prix
- Définir les **places disponibles**, horaires, etc.

#### d. Réservations et paiements

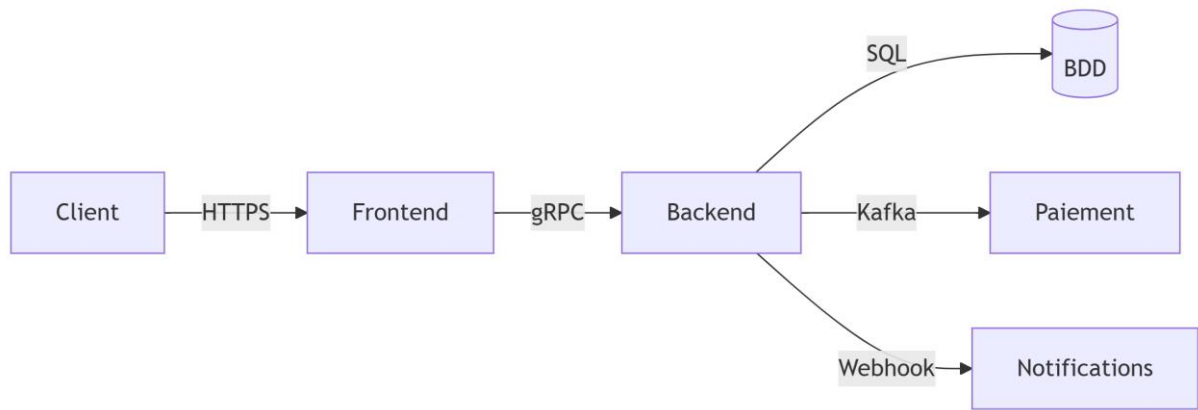
- Afficher la liste complète des réservations
- Filtrer par date, utilisateur, hôtel, circuit, statut
- Marquer une réservation comme confirmée / annulée

## Les différents diagrammes pour ce projet:

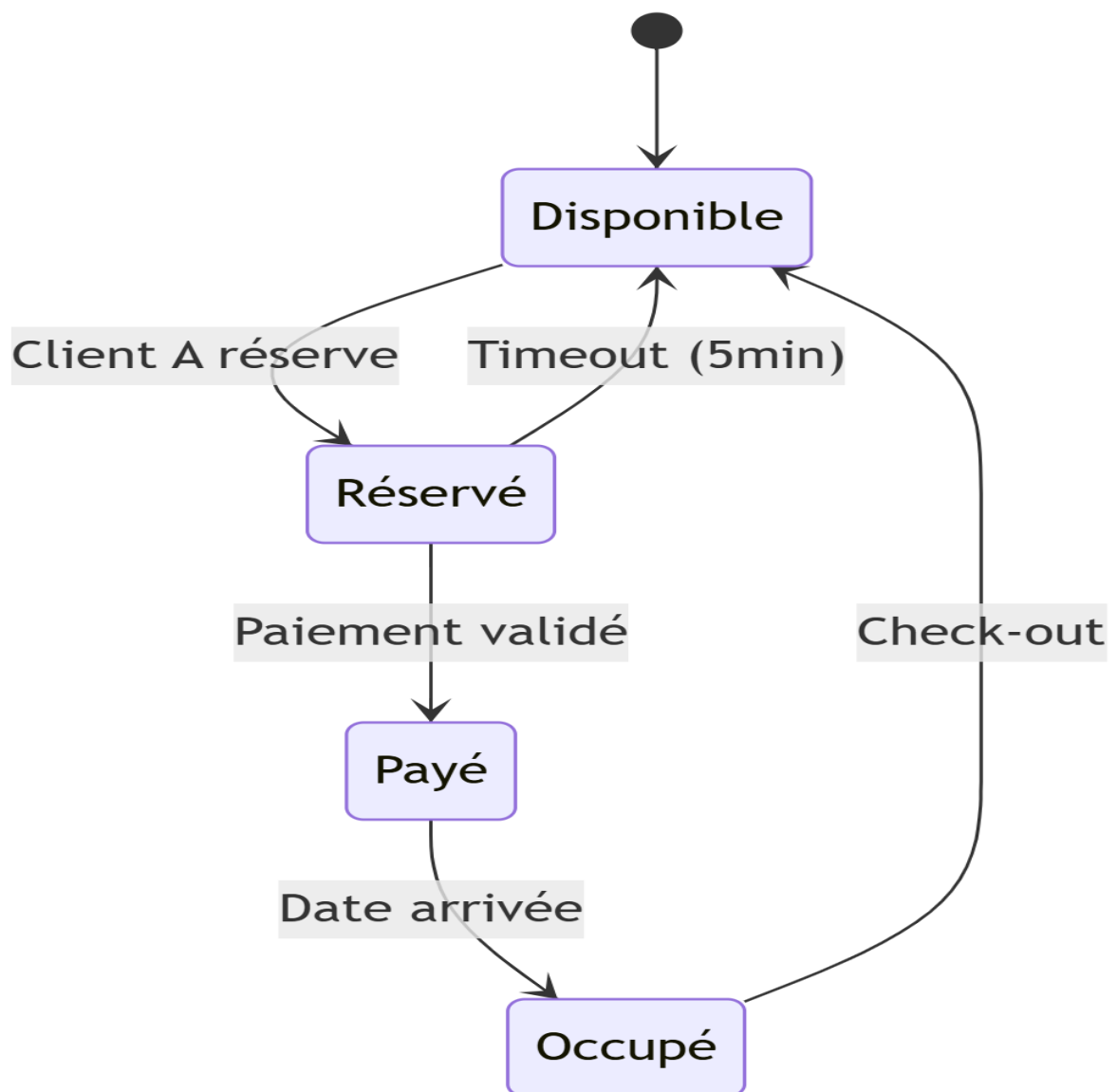
### a. Diagramme de cas d'utilisateur :



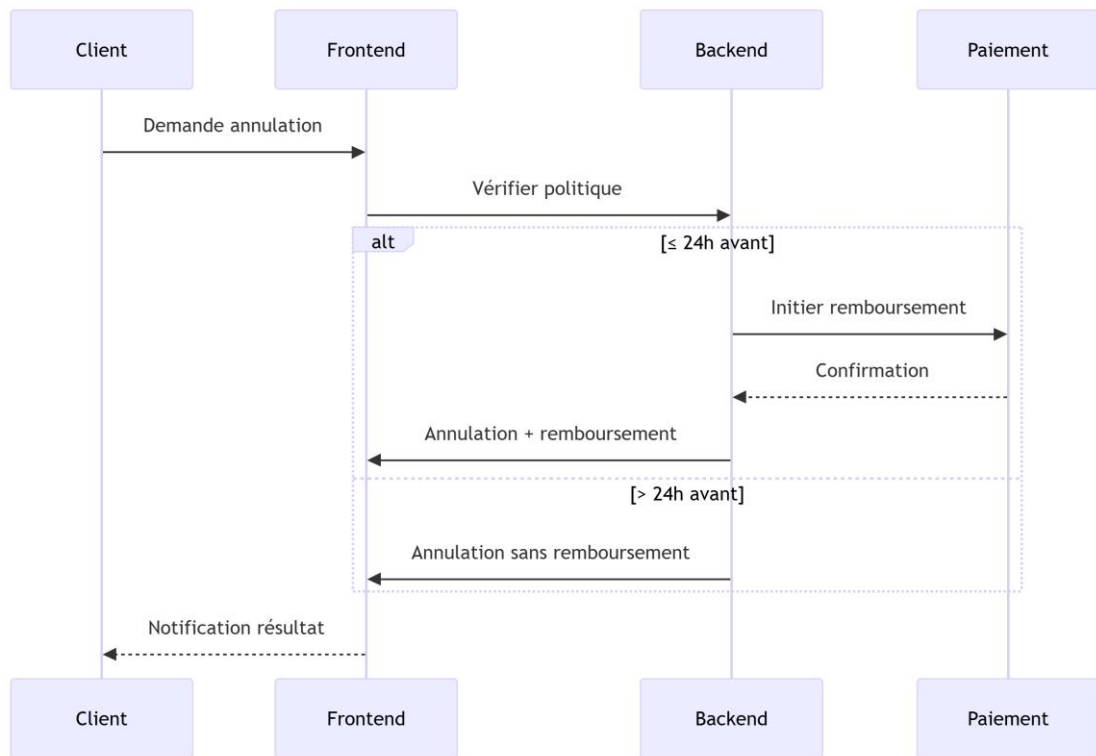




### Relation technique

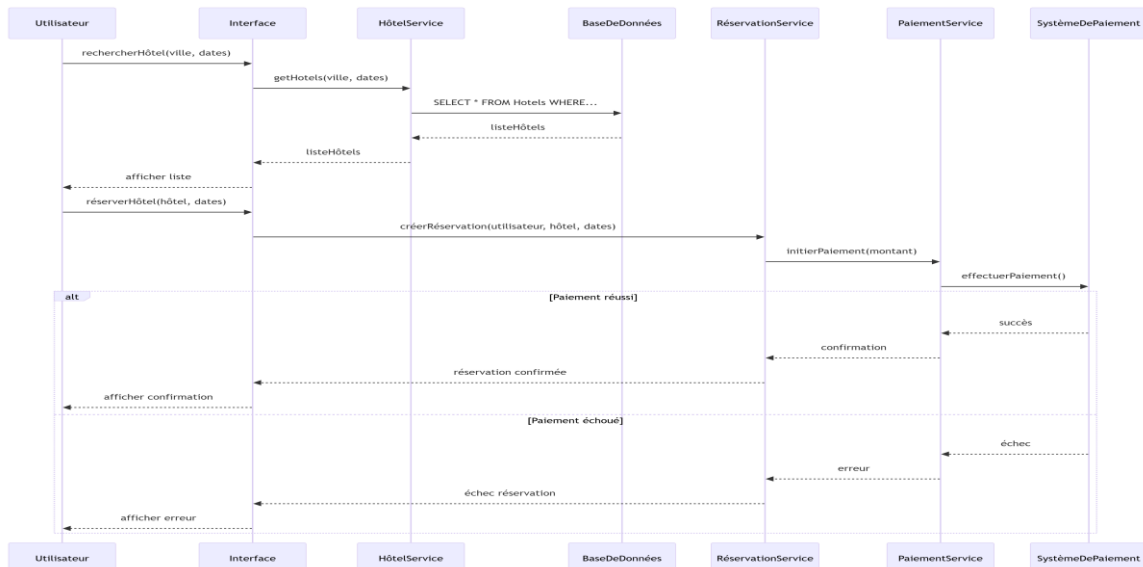


### Cas de conflit

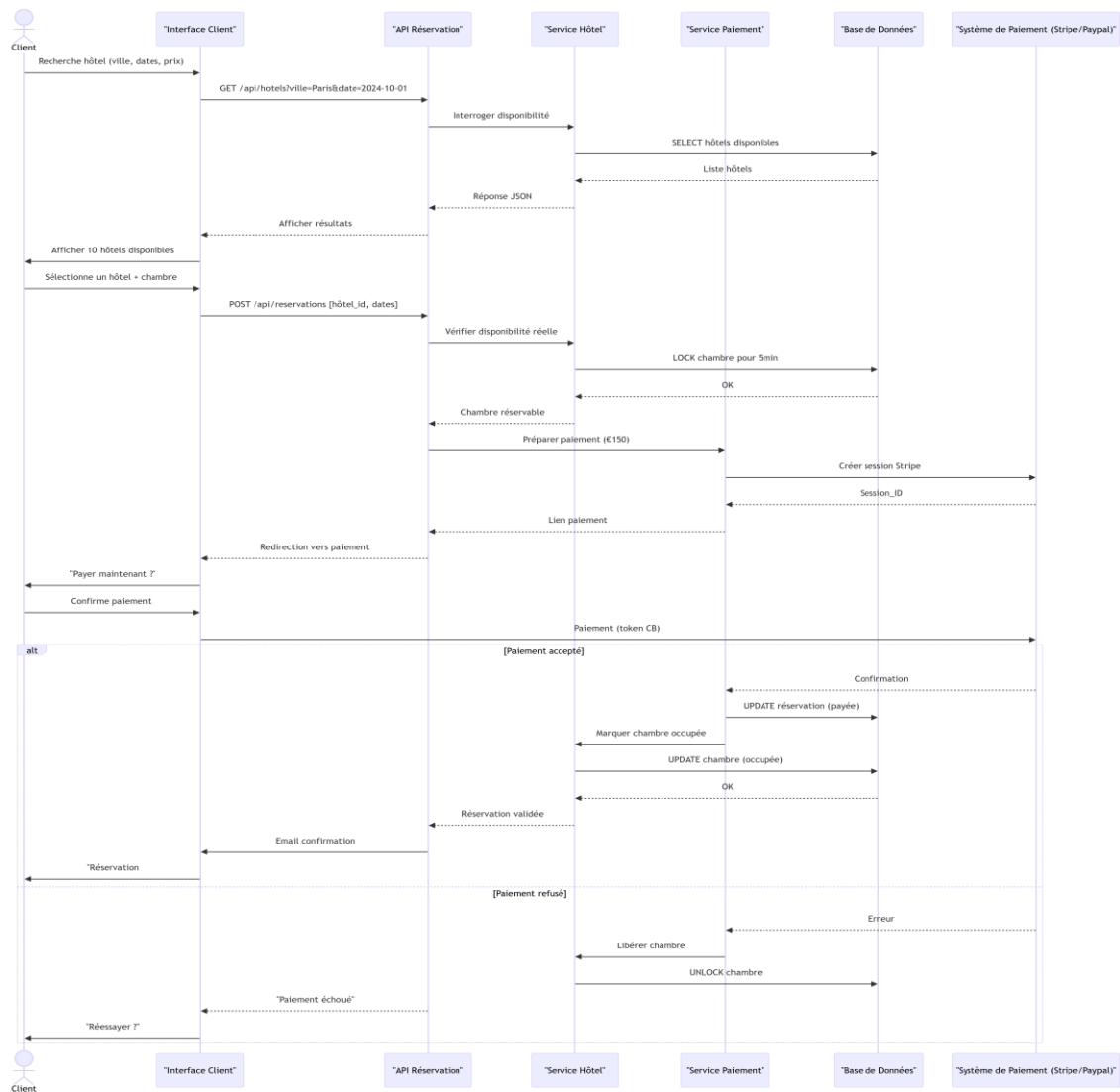


## Annulation de réservation

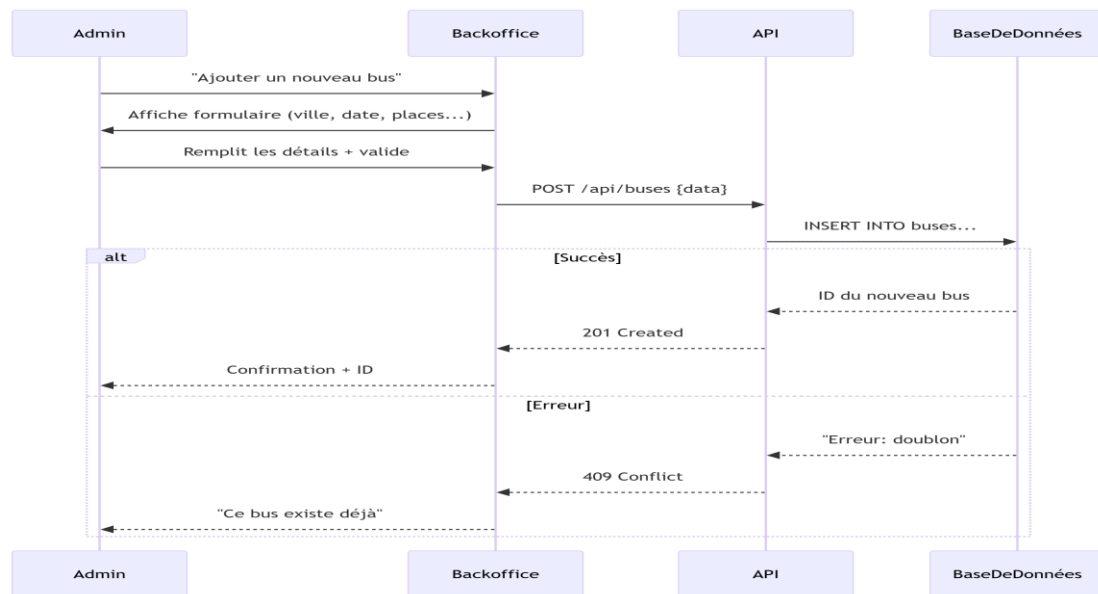
### b. Diagramme de séquence :



## Séquence générale

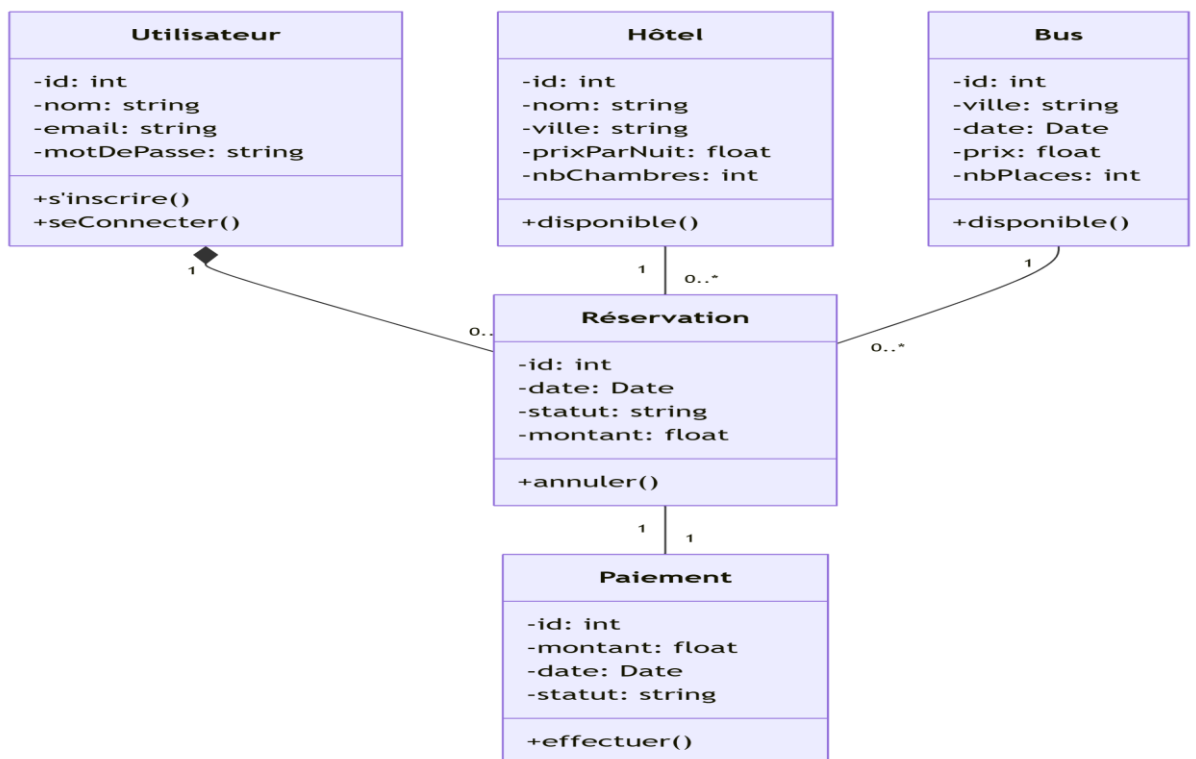


## Séquence hôtel



## Séquence Bus

### c. Diagramme de classe :



## 5. Conclusion

Ce cahier des charges définit les besoins fonctionnels et techniques pour la réalisation d'une plateforme de réservation d'hôtels et de bus touristiques en Laravel. Le respect des bonnes pratiques garantira un produit final performant, sécurisé et facile à maintenir.