

**Rapport de Mini-Projet c++**  
**Gestion de Réservation Hôtelière**

Année universitaire : 2007/2008

# ***Table des matières***

<b>Tables des matières.....</b>	<b>1</b>
<b>Introduction .....</b>	<b>2</b>
<b>Chapitre 1: Etude des besoins .....</b>	<b>3</b>
Introduction.....	3
I. Besoins fonctionnels.....	3
II. Besoins non fonctionnels.....	3
Conclusion.....	3
<b>Chapitre 2 : Analyse et conception.....</b>	<b>4</b>
Introduction.....	4
I. Outil de conception UML .....	4
II. Diagramme des cas d'utilisation.....	5
III. Diagramme de classes.....	6
IV. Conception de la base de données.....	7
Conclusion.....	7
<b>Chapitre 3 :implémentation.....</b>	<b>8</b>
Introduction.....	8
I. Environnement de travail.....	8
II. Diagramme de classes adapté à l'implémentation.....	9
III. Utilisation de l'application.....	10
Conclusion.....	10
<b>Conclusion générale.....</b>	<b>12</b>

# Introduction

La gestion hôtelière est une vitalité indispensable dans le déroulement des activités normale d'un hôtel.

Notre travail consiste donc à la conception et l'implémentation d'une application de gestion de réservation hôtelière qui prendra en compte toutes les contraintes qui peuvent survenir lorsqu'un agent hôtelier établi des réservations.

A travers notre application, il est possibles de vérifier la liste des chambres disponible selon les critères souhaiter par le client ainsi de les réserver afin d'être occuper ultérieurement.

Notre travail est présenté par trois chapitres :

Le premier sera consacré à la présentation des besoins fonctionnels et non fonctionnels

Dans le second chapitre, nous nous intéressons à l'étude conceptuelle et nous détaillons les différents modèles adoptés ainsi la structure de la base de donnée.

Finalement le dernier chapitre sera consacré à la présentation et la description des différentes interfaces de l'application.

# **Chapitre1 : Etude des besoins**

## **Introduction**

Durant le premier chapitre intitulé cadre de travail, on présente les besoins fonctionnels et non fonctionnels de notre application.

Cette spécification nous permettra d'éclaircir notre objectif.

### **I. Besoins fonctionnels**

Le système comportera différentes fonctionnalités nécessaires pour une meilleure gestion.

L'application doit accomplir les traitements suivants :

L'ajout, la modification et la suppression d'un client d'où il est nécessaire d'avoir tous les renseignements qui lui concerne tel que son nom, son prénom, son numéro de CIN, son numéro de téléphone...

Le suivi des chambres des chambres selon leurs types et leurs catégories.

La manipulation des réservations : ajout, annulation, modification de la réservation.

La facturation de la réservation selon la saison.

Ces informations seront stockées dans une base de donnée qui peut être mise à jour au fur et à mesure des besoins.

## **Conclusion**

Au cours de ce chapitre nous avons étudié les besoins fonctionnels et non fonctionnels de notre application. Cette étude nous aidera dans l'étape de conception.

# **Chapitre2 : Analyse et Conception**

## **Introduction**

Le choix d'une méthodologie de conception va nous permettre de mettre en place un produit robuste, fiable et évolutif.

Dans ce chapitre nous détaillons la conception de notre application. Ainsi, nous présentons la conception de notre base de données et les modules de base.

### **I. Outil de conception UML**

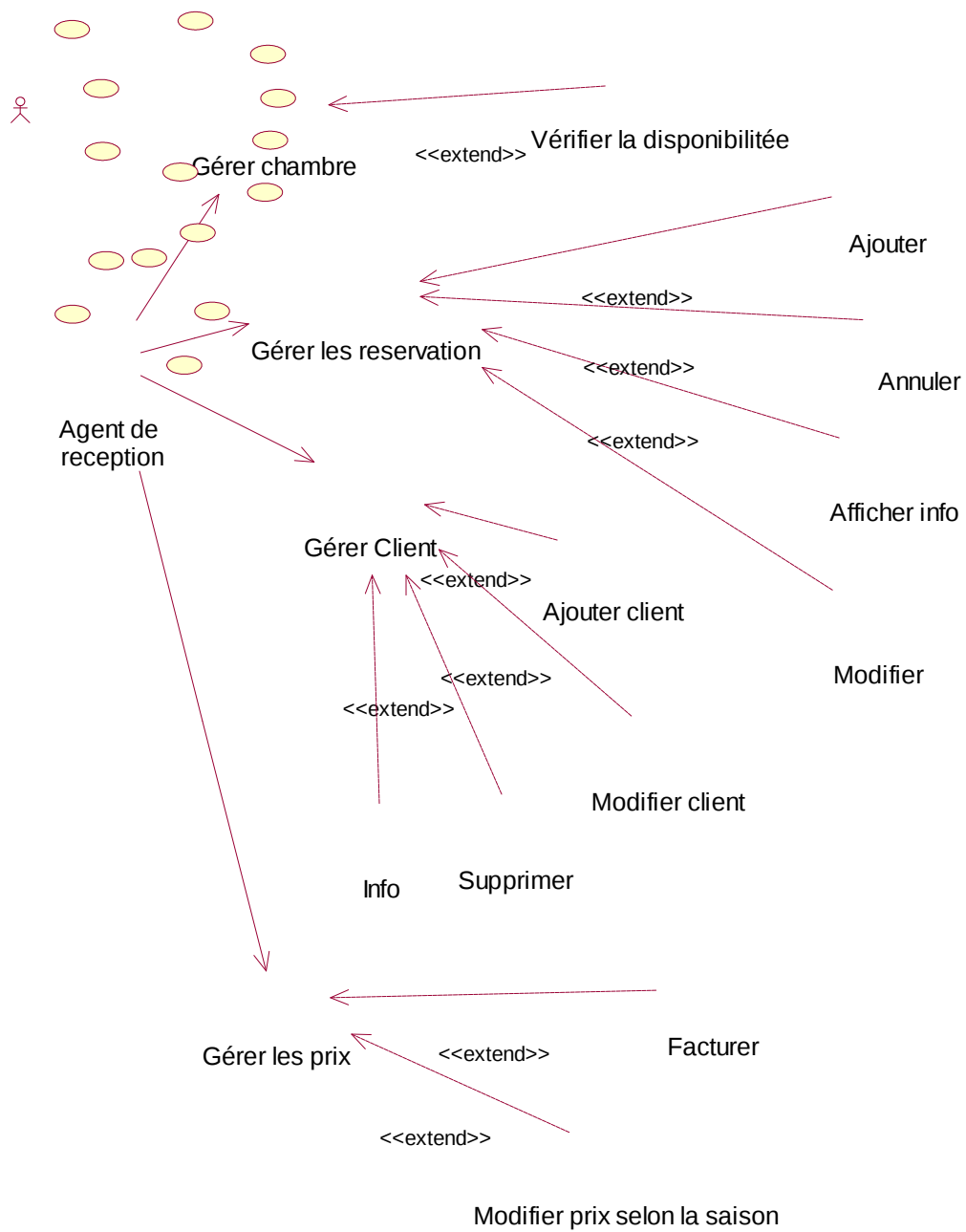
UML est une notation permettant de modéliser un problème de façon standard. Ce langage est né de la fusion de plusieurs méthodes existantes auparavant, et est devenu désormais la référence en terme de modélisation objet.

Nous avons choisi UML comme langage de conception vu sa simplicité et sa puissance à l'abstraction des objets réels. Parmi les facteurs de puissance, UML présente plusieurs diagrammes statiques et dynamiques facilitant la tâche de conception et offrant une vue globale sur la composition et le fonctionnement de notre logiciel.

### **II. Diagramme des cas d'utilisation**

Le diagramme de cas d'utilisation représente la structure des fonctionnalités nécessaires aux utilisateurs du système. Il est utilisé dans les deux étapes de capture des besoins fonctionnels et techniques.

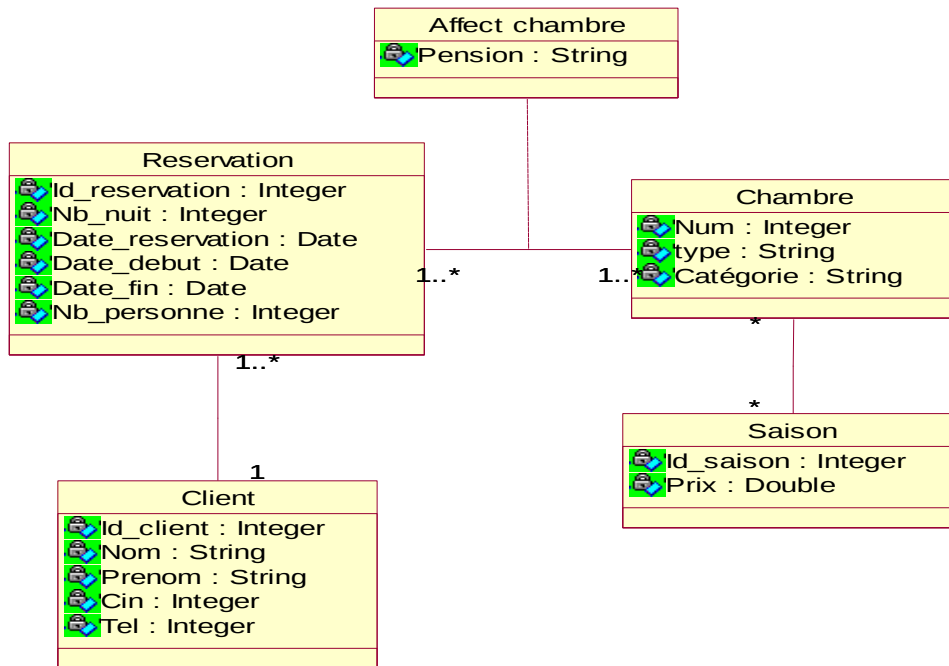
A partir de l'étude préliminaire, nous avons pu dégager le diagramme des cas d'utilisation général suivant :



**Diagramme des cas d'utilisation**

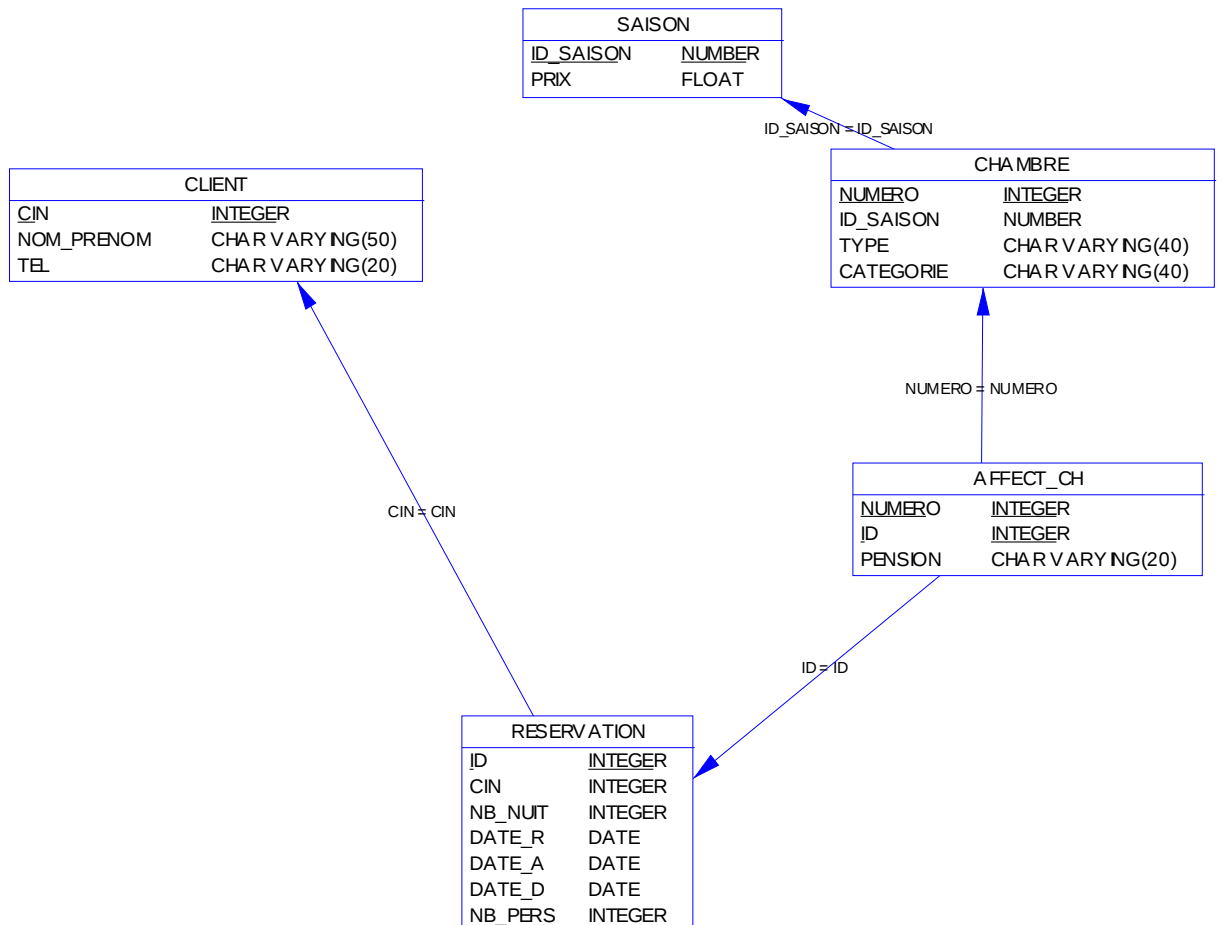
### III. Diagramme de classes

Dans cette section, nous allons présenter le diagramme des classes de la base de données :



## II Conception de la base de données

Les figures suivantes représentent le modèle physique de notre base de données :



## Conclusion

Cette phase nous a permis de comprendre le contexte du système, identifier les principaux cas d'utilisation, ce qui nous a facilité la modélisation de notre base de données. Le chapitre suivant sera consacré à la présentation de la partie développement de ce projet.

## Chapitre3 : Implémentation



## Introduction

Dans ce chapitre, nous détaillons l'implémentation de notre application. Ainsi, nous développons les différents modules de l'application. Nous présentons également l'utilisation de l'application via les fenêtres de base.

## I Environnement de travail

L'application comprend une base de données qui, pour être facilement manipulée, interagit avec une interface graphique. Nous aurons recours pour son développement aux outils suivants :

### ➤ POWER AMC DESIGNOR :

Avec cet outil de conception on peut :

- Concevoir un système d'information en utilisant un diagramme Entité Association appelé Modèle Conceptuel de Données (MCD).
- Générer le Modèle Physique de Données (MPD) correspondant, pour un système de gestion de base de données (SGBD), en tenant compte des spécificités du SGBD choisi.
- Personnaliser le MPD afin de respecter les contraintes physiques et les performances du produit.
- Générer un script de création de base de données pour le SGBD cible.

### ➤ ORACLE

C'est un système de gestion des bases de données relationnel. Pour notre application on va s'intéresser à ORACLE 10g qui est adaptable avec les règles de gestion de notre application.

La connexion avec la base a été réalisée grâce à l'intermédiaire d'un OCL :

- C'est une interface de programmation d'application performante spécifiée par un accès rapide à la base de données.
- Possède les mêmes fonctionnalités d'un ODBC.

➤ RATIONNAL ROSE

Est un logiciel de conception nous permettons de réaliser le diagramme des classes de notre application.

➤ MICROSOFT VISUAL STUDIO 6.0

Est utilisé pour l'implémentation de notre application avec le langage de programmation orientée objet c++.