

# Rapport de projet

## *Base “Touriste”*

**Médéric HURIER**

<mederic.hurier@etudiant.univ-nancy2.fr>

**Jérémy ROBIN**

<jeremy.robin@etudiant.univ-nancy2.fr>

**Anne-Sophie DUHAUT**

<anne-sophie.duhaut@etudiant.univ-nancy2.fr>

**Apoté Vrénière AMBOUNDA ZEKPA**

<apote.ambounda-zekpa@etudiant.univ-nancy2.fr>



**Année 2011 – 2012**

# Table des matières

1 Introduction.....	3
2 Schéma conceptuel de la base de données.....	3
2.1 Modèle conceptuel de données.....	4
2.2 Modèle physique de données.....	4
2.3 Règles de gestion.....	4
3. Schéma conceptuel de la dynamique avec la démarche intégratrice.....	4
3.1 Demande de disponibilité.....	4
3.2 Confirmation de réservation.....	4
3.3 Versement des arrhes.....	5
3.4 Annulation d'une réservation.....	6
3.5 Analyse des échéances.....	7
3.6 Réception du paiement.....	8
3.7 Ajout d'une nouvelle location.....	9
3.8 Reprise de la gestion par le prestataire.....	10
3.9 Analyse des demandes en attente.....	11
4 Dictionnaire d'informations.....	12
4.1 Entités et attributs.....	12
4.2 Opérations et évènements.....	18
4.3 Conditions et facteurs.....	22
5 Requêtes répondant aux besoins en informations.....	23
6 Application.....	24
6.1 Présentation.....	24
6.2 Utilisation.....	25
6.3 Scripts.....	25
6.4 Captures d'écran.....	36
7 Conclusion.....	36
7.1 Réflexion sur les outils et méthodes utilisées.....	37
7.2 Difficultés rencontrées.....	37

# 1 Introduction

Ce rapport a pour but de présenter le schéma conceptuel complet de l'application "base touriste". Il comprend plusieurs parties:

- **Un modèle conceptuel de données** réalisé avec PowerAMC, et **son modèle physique de données** correspondant.
- **Un modèle conceptuel de traitement** respectant les spécifications de la démarche intégratrice.
- **Un dictionnaire d'information**, décrivant et commentant les entités, les attributs, les domaines, les événements, les opérations, les conditions et les facteurs.
- **Une liste de requête** répondant au besoin en informations
- Une **conclusion** sur l'utilisation des méthodes et des outils, ainsi que sur les difficultés rencontrées lors du projet.

Afin d'améliorer la consistance de l'analyse, plusieurs choix de modélisation ont été réalisés.

- **Conventions de nommage :**
  - Chaque table est composée d'une seule clé primaire préfixée dont l'attribut correspondant est préfixé de "no". Dans l'interface, l'attribut est nommé "N°".
  - Tous les attributs de la base sont uniques.
  - Toutes les dates sont au format "yyyy-mm-dd" (année-mois-date)
  - Il n'existe aucune association ternaire dans le modèle conceptuel de données.
- **Règles de gestion supplémentaires :**
  - Les périodes de disponibilité sont fixées par notre agence de manière hebdomadaire ou mensuel. Nos prestataires associent ensuite ces périodes à leurs hébergements. On consulte les disponibilités et les réservations pour vérifier que l'hébergement est respectivement accessible et libre.
  - Comme précisé dans l'énoncé, l'envoi de propositions d'hébergement et d'assurance est totalement manuel. Le système vérifie toutefois que l'hébergement utilisé pour confirmer la réservation satisfait les critères de la demande.
  - Les versements aux prestataires en fin de saison et les relances des paiements du client sont purement informatifs.
  - Une entité "Paiement" permet de gérer les versements et les remboursements du client. Les arrhes se versent en 1 fois, le montant restant en plusieurs fois mais jamais au dessus du montant total de la réservation.
- **Technique :**
  - Le moteur MySQL utilisé est InnoDB car il gère pleinement les clés étrangères et les suppressions en cascade.
  - Pour améliorer la qualité du code, le framework Yii a été intégré au projet.
  - Les déclencheurs et procédures stockées n'étant pas recommandés en MySQL, les enchaînements sont gérés directement en PHP.

## 2 Schéma conceptuel de la base de données

### 2.1 Modèle conceptuel de données

Voir annexe 1.

### 2.2 Modèle physique de données

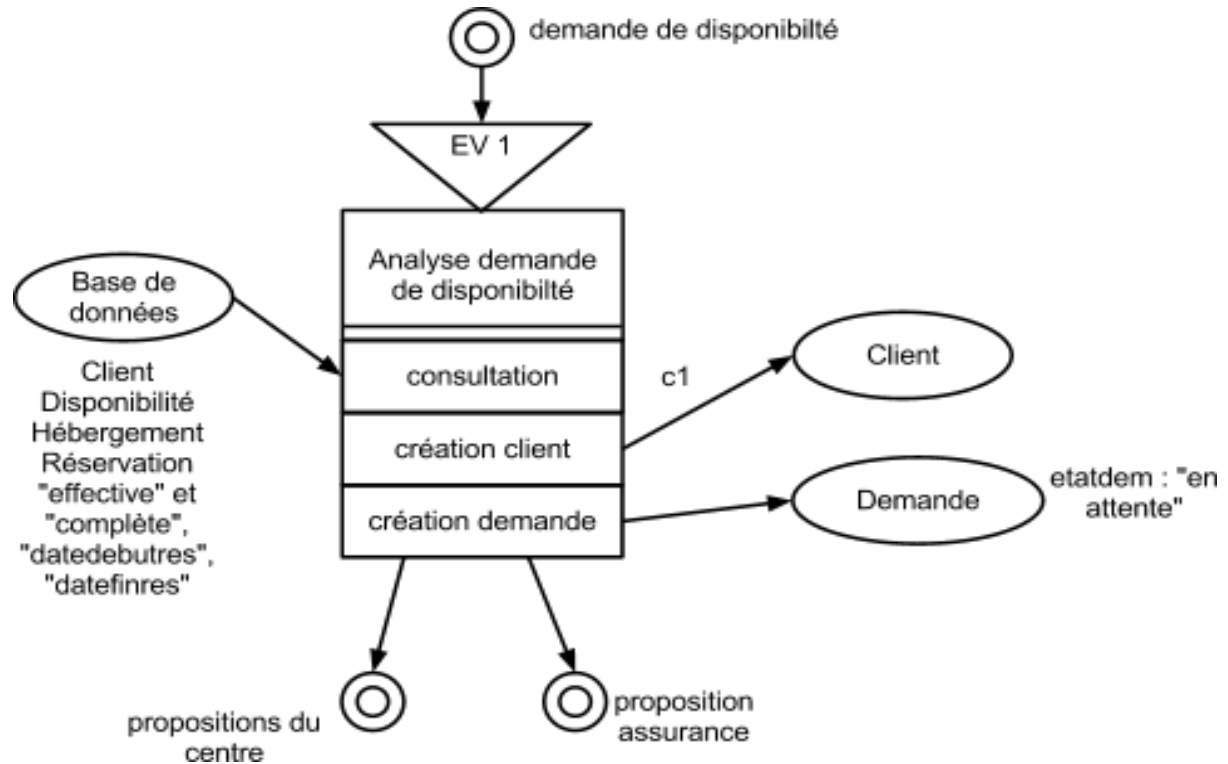
Voir annexe 2.

### 2.3 Règles de gestion

Voir annexe 3.

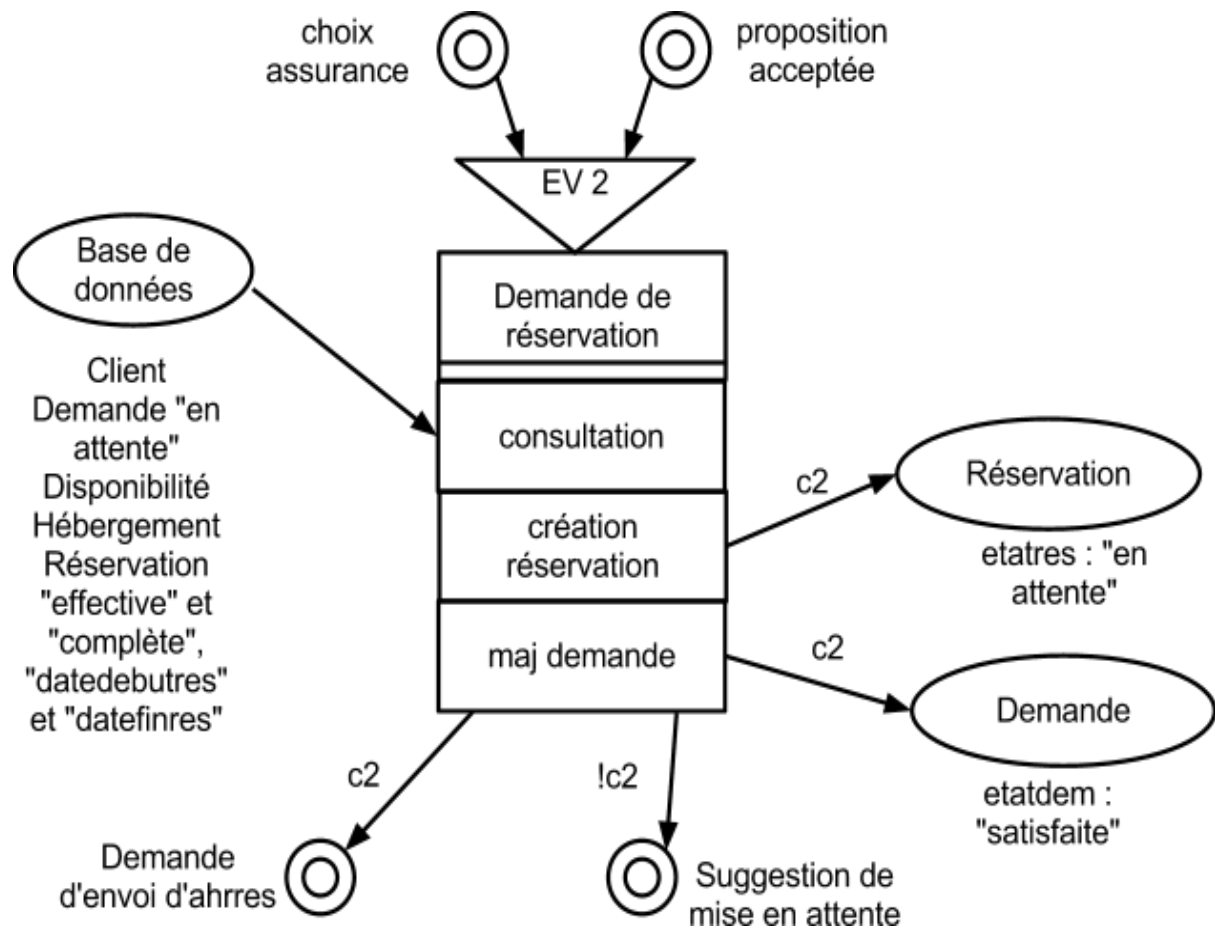
## 3. Schéma conceptuel de la dynamique avec la démarche intégratrice

### 3.1 Demande de disponibilité



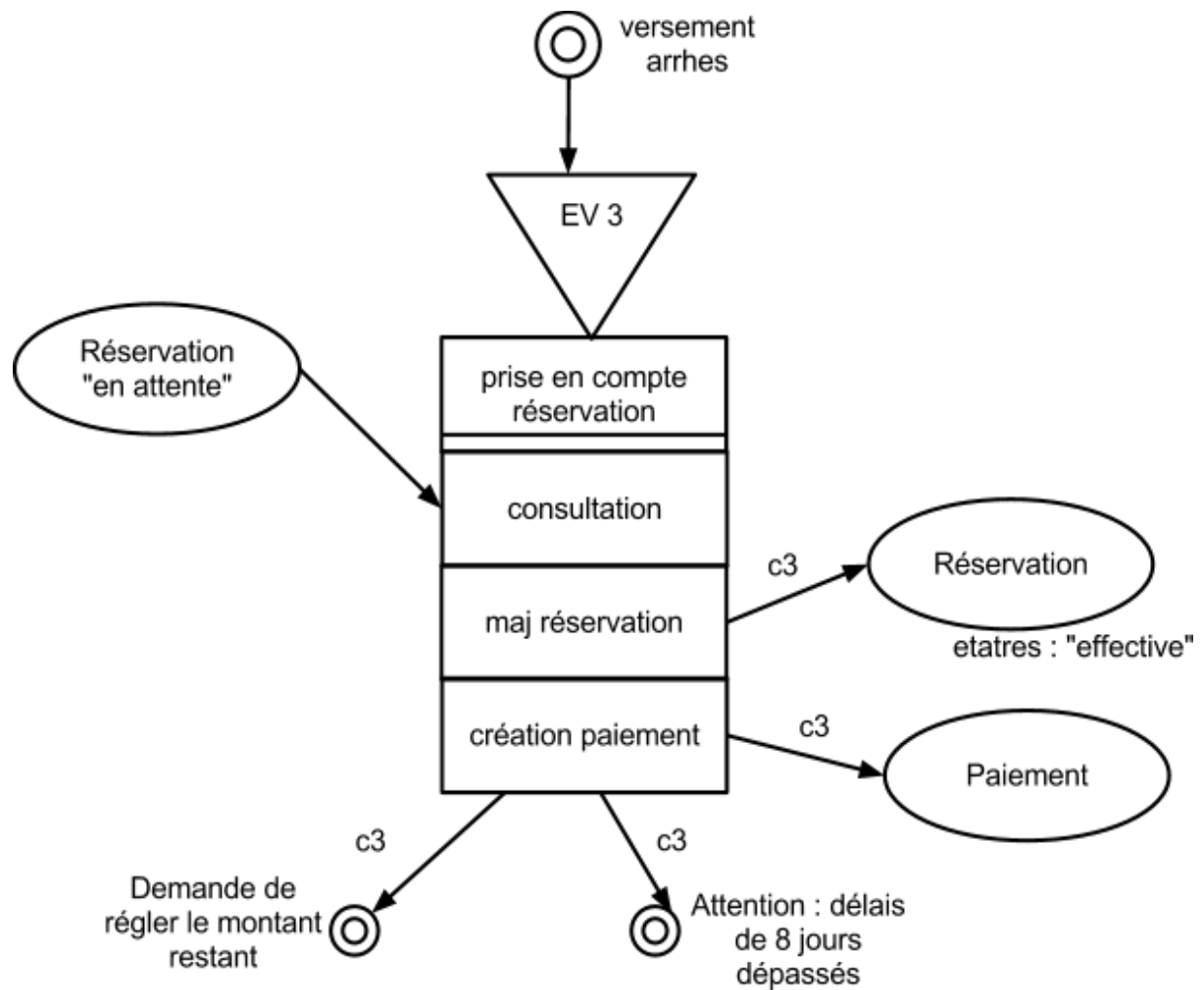
*c1 : le client n'existe pas*

### 3.2 Confirmation de réservation



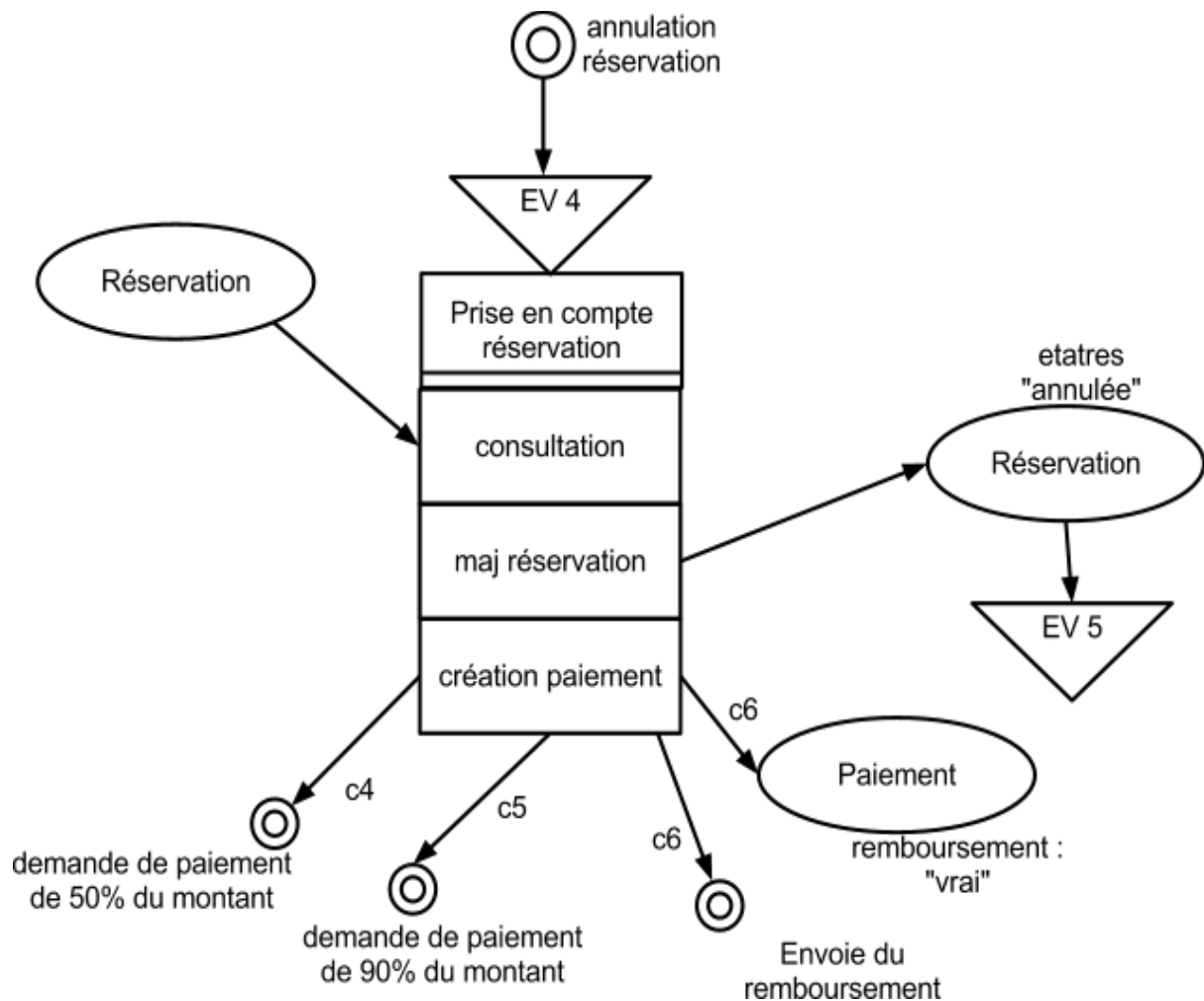
*c2 : l'hébergement est disponible*

### 3.3 Versement des arrhes



*c3 : si getDate() - dateres <= 8 jours*

### 3.4 Annulation d'une réservation



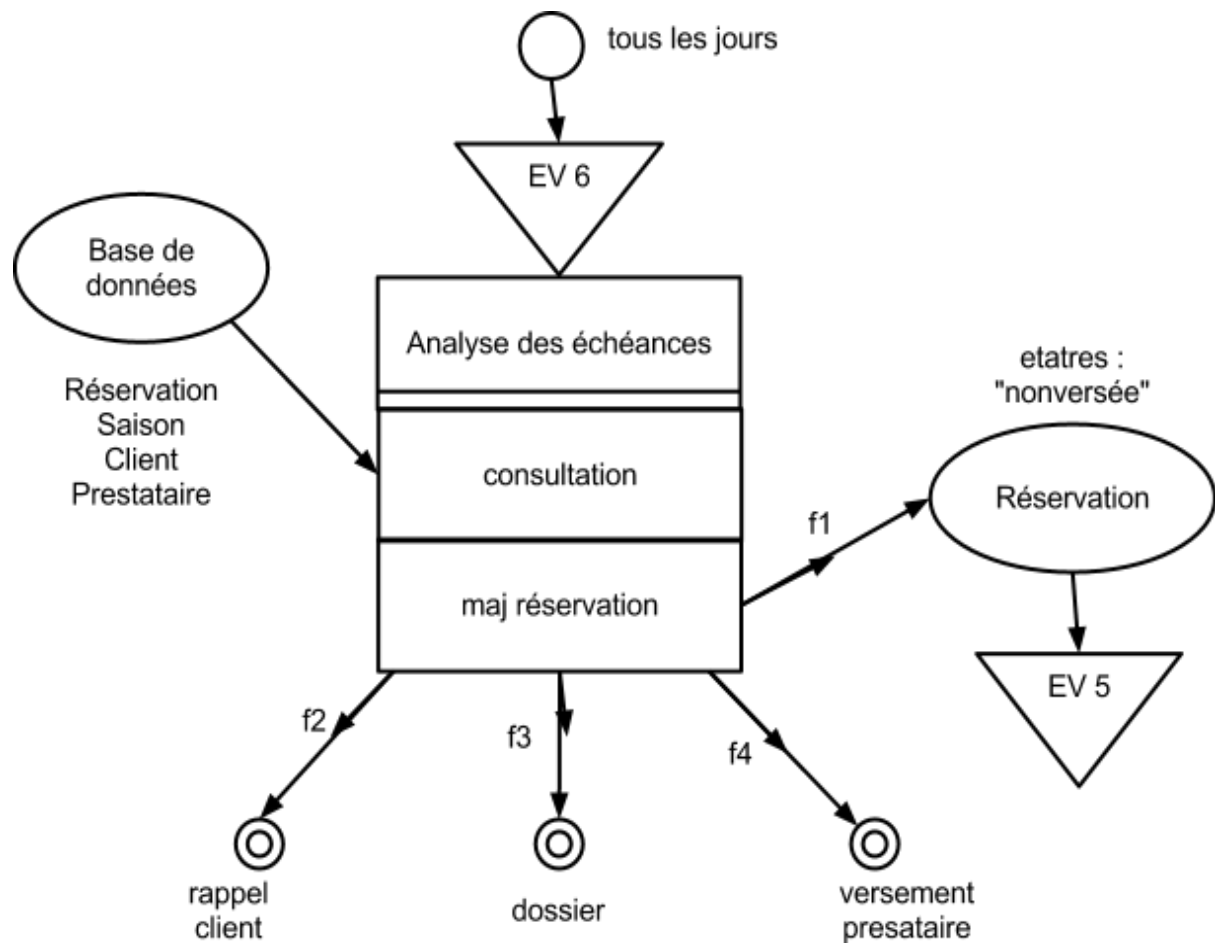
*dateanul : getDate()*

*c4 : si etatres = "effective" ou "complète" et (datedebutres - 30 jours) < dateanul < (datedebutres - 15 jours)*

*c5 : si etatres = "effective" ou "complète" et (datedebutres - 15 jours) < dateanul < (datedebutres)*

*c6 : si etatres = "effective" ou "complète" et total paiement > montant à rembourser, montant du paiement à rembourser = total paiement - montant à rembourser*

### 3.5 Analyse des échéances



*f1 : pour toutes les réservations dont :*

- *etats = "en attente" et dateres + 8 jours > getDate()*
- *etats = "effective" et datedebutres < getDate()*

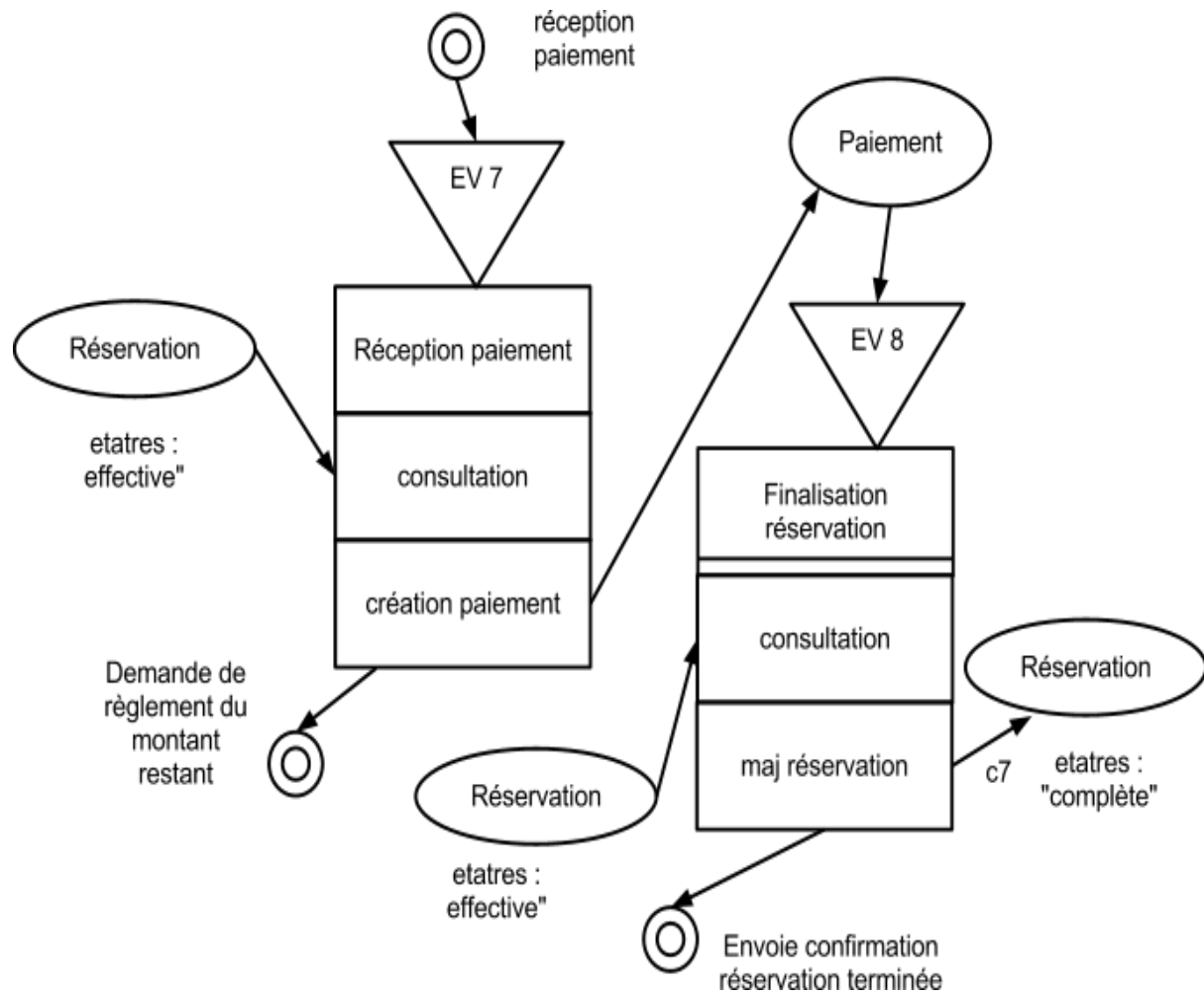
*f2 : pour toutes les reservations dont etats = "effective" et getDate() = datedebutres*

*f3 : pour toutes les réservations dont etats = "effective" et getDate() > datedebutres*

*f4 : pour toutes les réservations dont datedebutres > datedebutsais et datefinres < datefinsais et datefinsais = getDate()*

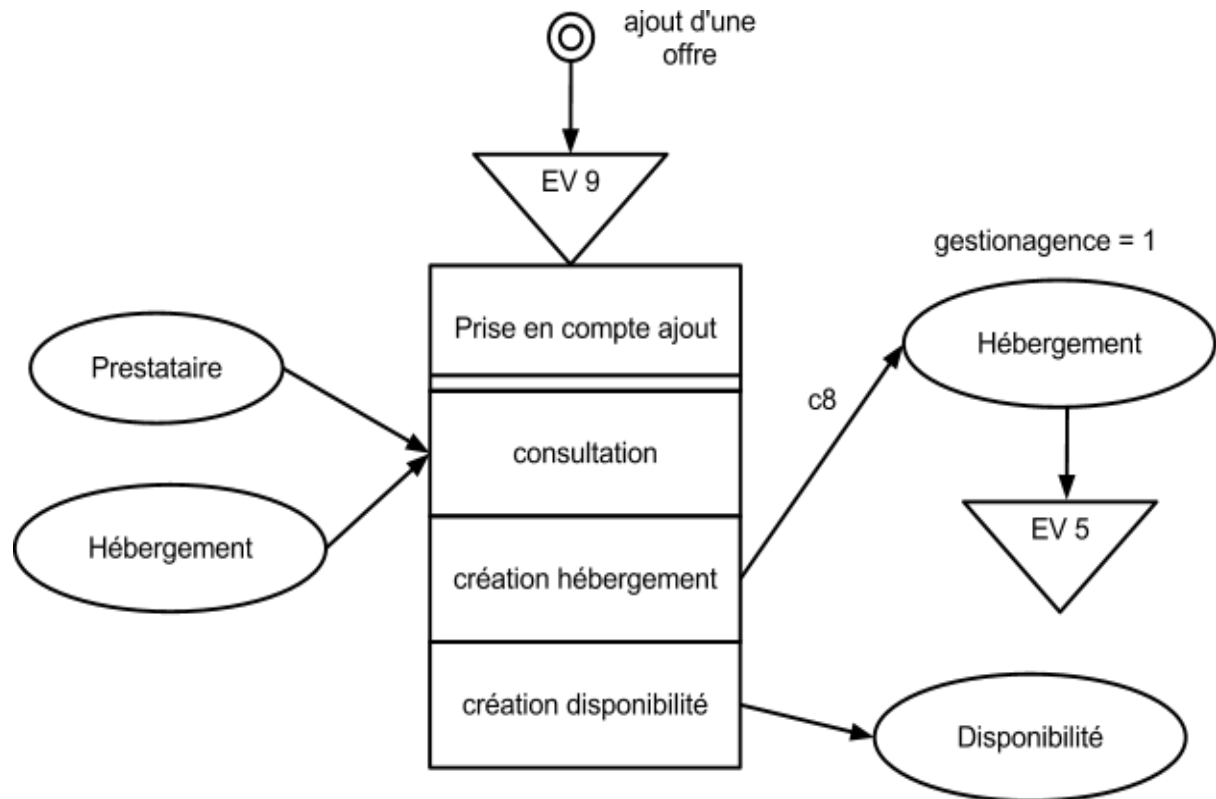


### 3.6 Réception du paiement



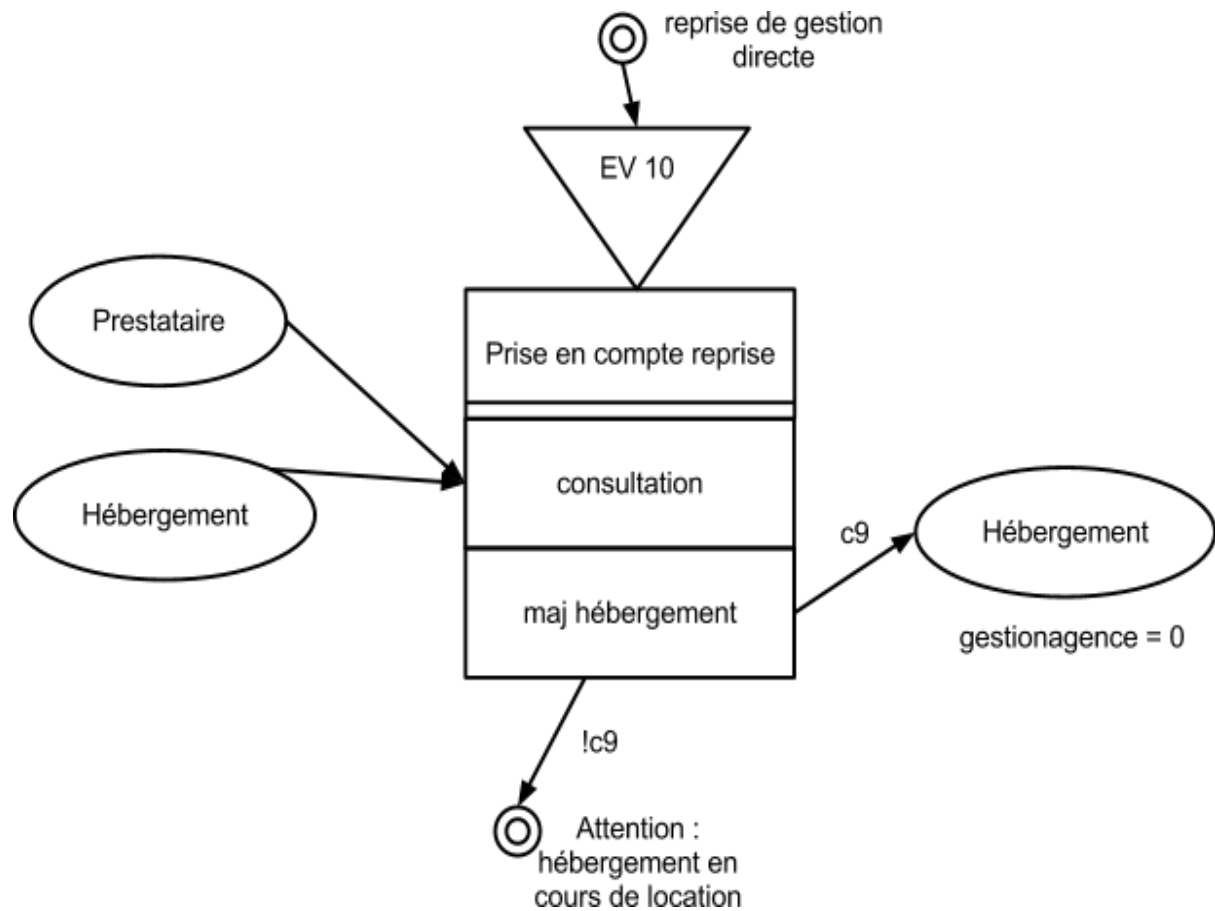
$c7 : \text{montant} = \text{somme des paiements}$

### 3.7 Ajout d'une nouvelle location



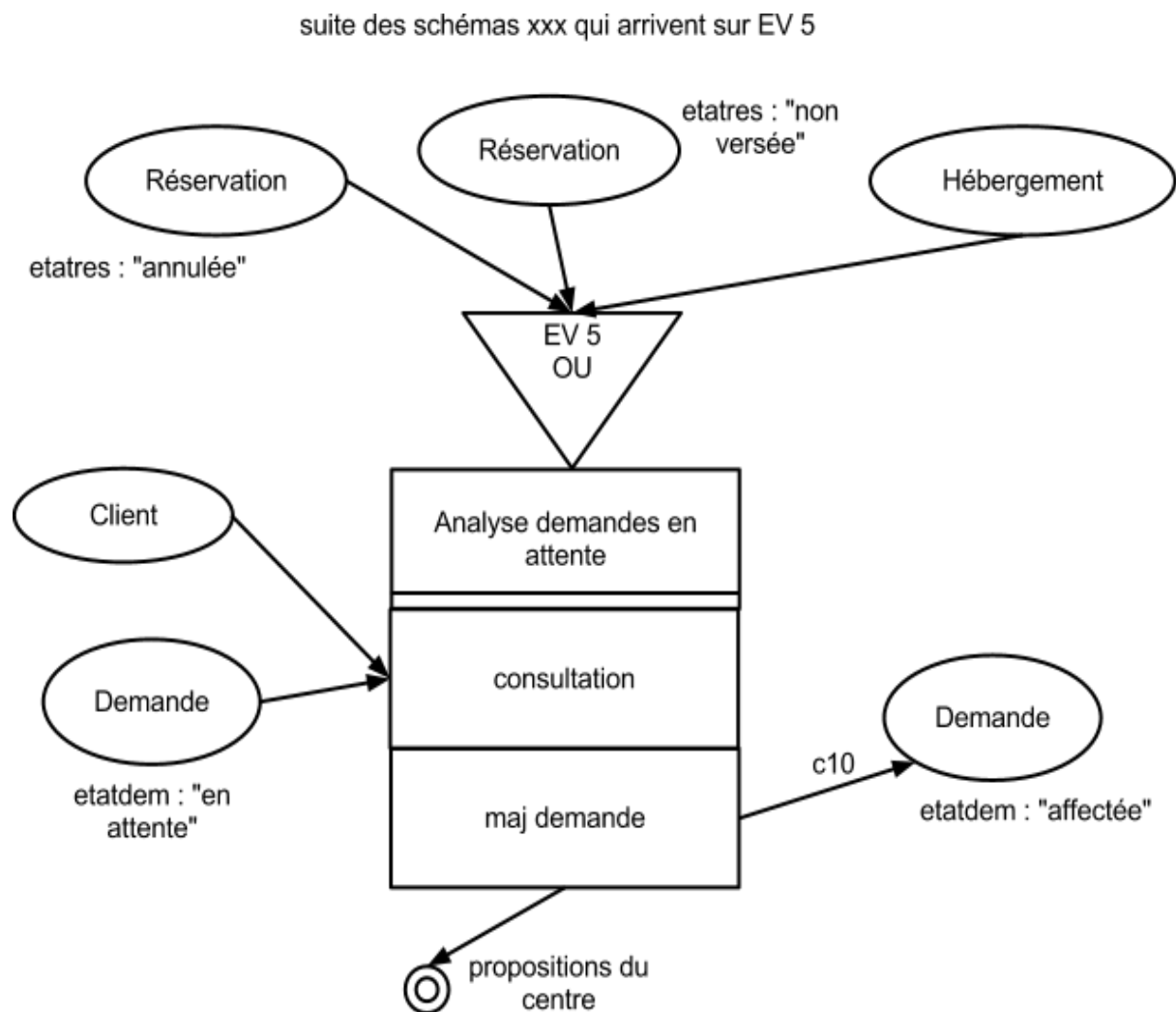
*c8 : Si l'hébergement n'existe pas*

### 3.8 Reprise de la gestion par le prestataire



*c9 : si l'hébergement n'a pas été loué (s'il n'y a pas de réservation "effective" ou "complète" pour l'hébergement).*

### 3.9 Analyse des demandes en attente



*c10 : s'il existe au moins une demande à satisfaire*

## 4 Dictionnaire d'informations

### 4.1 Entités et attributs

#### a) Description des tables

##### Entité **client**:

- **noclient** : l'identifiant du client
- nom : nom du client
- prenom : prénom du client
- datenaiss : date de naissance du client
- sexe : sexe du client
- sitmarital : statut marital
- telephoneclient : numéro de téléphone du client
- adresseclient : adresse fixe du client
- emailclient : adresse email du client

##### Entité **demande**:

- **ndem** : l'identifiant de la demande
- *noclient* : identifiant du client effectuant la demande
- *nostat* : numéro de station
- *nores* : numéro de la réservation
- *notyph* : identifiant du type d'hébergement
- *notypp* : identifiant du type de prestation
- etatdem : l'état de la demande
- codeattente : numéro de l'hébergement pour lequel la personne attend
- datedem : date de la demande

##### Entité **reservation**:

- **nores** : numéro de la réservation
- *ndem* : identifiant de la demande
- *nopaie* : identifiant du paiement
- *noheberg* : numéro de l'hébergement
- datreres : date de réservation
- montantres : montant de la réservation
- datedebutres : date de début de la réservation
- datefinres : date de fin de la réservation
- dateannul : date annulation de la réservation
- etatres: état de de la réservation
- nbpers : nombre de participants au séjour
- assurance : indique si une assurance a été prise pour la réservation
- montant\_total : montant total de la réservation

##### Entité **paiement**:

- **nopaie** : numéro du paiement
- *notypepaie* : numéro du type de paiement
- libellepaie : libellé du paiement
- montanpaie : montant du paiement
- datepaie : date à laquelle le paiement a été effectué
- remboursaie : indicateur de remboursement

##### Entité **hebergement**:

- **noheberg** : identifiant de l'hébergement
- *notypprest* : identifiant de la prestation

- *notyph* : identifiant du type d'hébergement
- *adresse* : adresse de l'hébergement
- *qualite* : qualité de l'hébergement
- *surface* : surface de l'hébergement
- *nblitadult* : nombre de lits pour adultes
- *nblitenfant* : nombre de lits pour enfants
- *wifi* : possibilité d'avoir le wifi
- *resatauration* : indique si la restauration est comprise
- *gestionagence* : indique si l'hébergement est géré par agence

#### Entité **disponibilite**:

- **nodisp** : identifiant de la disponibilité
- *datedebdisp* : date de début de la prestation
- *datefindisp* : date de fin de la prestation

#### Entité **saison**:

- **nosaison** : identifiant de la saison
- *datedebs* : date de début de la saison
- *datefins* : date de fin de la saison

#### Entité **prestataire**:

- **noprest** : identifiant du prestataire
- *nostat* : idetifiant de la station
- *notypp* : identifiant du type de prestation
- *nomprest* : libellé de la prestation
- *telephoneprest* : numéro de téléphone du prestataire
- *emailprest* : email du prestataire
- *particulier* : indique si le prestataire est un particulier

#### Entité **service**:

- **noserv** : identifiant du service
- *nomservice* : libellé du service
- *compris* : indique si le service est compris

#### Entité **station**:

- **nostat** : numéro de la station
- *nomstat* : nom de la station
- *adressestat* : adresse de la station
- *telephonestat* : numéro de téléphone de la réservation
- *emailstat* : adresse email de la station

#### Entité **particularite**:

- **nopart** : identifiant de la particularité
- *nompart* : libellé de la particularité
- *adressepart* : adresse de la particularité
- *descriptionpart* : description de la particularité
- *accessible* : indique l'accessibilité de la particularité

#### Entité **activite**:

- **noact** : numéro de l'activité
- *notypact* : numéro du type d'activité
- *nomact* : libellé de l'activité
- *interieur* : indique si l'activité se déroule à l'intérieur
- *enfantact* : indique si c'est une activité pour enfant

Entité **typepaiement**:

- **notypepaie** : numéro du type de paiement
- libelletypepaie : libellé du type de paiement

Entité **typeprest**:

- **notypp** : numéro du type de prestation
- nomtyp : libellé du type de prestation

Entité **typeheberg**:

- **notyph** : numéro du type d'hébergement
- nomtyph : libellé du type d'hébergement

Entité **typeactivite**:

- **notypact** : numéro du type d'activité
- nomtypact : libellé du type d'activité

Entité **voir**:

- **notypart** : identifiant de la particularité
- **nostat** : identifiant de la station

Entité **possible**:

- **notact** : identifiant de l'activité
- **nostat** : identifiant de la station

Entité **dispo**:

- **nodisp** : identifiant de la disponibilité
- **noheberg** : identifiant de l'hébergement

Entité **apour**:

- **nosaison** : identifiant de la saison
- **noheberg** : identifiant de l'hébergement
- prix : prix de l'hébergement selon la saison

Entité **offert**:

- **noserv** : identifiant du service
- **noprest** : identifiant de la prestation

**b) Contraintes de domaine**

Table	Attribut	Valeurs
demande	etatdem	'en attente', 'validé', 'annulé', 'renvoie proposition'
reservation	etatres	'en attente arrhes', 'effective', 'annule', 'refusé', 'complete'

### c) Types

Nom	Domaine	Type de données	Taille
accessible		Booléen	
adresse		Caractère long variable (60)	60
adressecient		Caractère long variable (60)	60
adressepart		Caractère long variable (55)	55
adressestat		Caractère long variable (55)	55
assurance		Booléen	
codeattente		Entier	
compris		Booléen	
datannul		Date	
datdem		Date	
datedebdisp		Date	
datedebs		Date	
datedebutres		Date	
datefindisp		Date	
datefinres		Date	
datefins		Date	
datenaiss		Date	
datepaie		Date	
dateres		Date	
datmodif		Date	
descriptionpart		Caractère long variable (60)	60
emailclient		Caractère long variable (60)	60
emailprest		Caractère long variable (50)	50
emailstat		Caractère long variable (55)	55
enfantact		Booléen	
etatdem	etatdeman de	Caractère long variable (20)	20
etatres	etatreserva	Caractère long variable (20)	20



	tion		
gestionagence		Booléen	
interieur		Booléen	
libellepaie		Caractère variable (50)	50
libelletypaie		Caractère long variable (30)	30
montant_total		Réel	
montantpaie		Réel	
montantres		Réel	
nblitadult		Entier	
nblitenfant		Entier	
nbpers		Entier	
ndem		Entier	
noact		Entier	
noclient		Entier	
nodisp		Entier	
noheberg		Entier	
nom		Caractère long variable (60)	60
nomact		Caractère long variable (50)	50
nompart		Caractère long variable (20)	20
nomprest		Caractère variable (30)	30
nomservice		Caractère long variable (55)	55
nomstat		Caractère long variable (30)	30
nomtyp		Caractère long variable (30)	30
nomtypact		Caractère long variable (60)	60
nomtyph		Caractère long variable (30)	30
nopaie		Entier	
nopart		Entier	
noprest		Entier	
nores		Entier	

nosaison		Entier	
noserv		Entier	
nostat		Entier	
notypact		Entier	
notyph		Entier	
notypp		Entier	
notyppaie		Entier	
particulier		Booléen	
prenom		Caractère long variable (60)	60
prix		Réel	
qualite		Caractère long variable (50)	50
remboursepaie		Booléen	
restauration		Booléen	
sexe		Caractère long variable (10)	10
sitmarital		Caractère long variable (20)	20
surface		Réel	
telephoneclient		Caractère long variable (20)	20
telephoneprest		Caractère long variable (20)	20
telephonestat		Caractère long variable (20)	20
wifi		Booléen	

## 4.2 Opérations et évènements

### EV1 :

**Objet externe d'entrée :** demande de disponibilité, le centre reçoit de la part de ses clients des demandes de disponibilité par courrier ou par téléphone.

**Opération :** analyse demande de disponibilité, le centre analyse la demande du client.

**Actions :**

- consultation : plusieurs objets sont consultés dont le client, l'hébergement ainsi que ses disponibilités, les réservations effectives et complètes de dates correspondant à la demande du client. Ainsi, on écarte tous les hébergements déjà réservés aux dates demandées par le client.
- création client : le client demandeur est créé, s'il n'existe pas.
- création demande : la demande est créée avec un état "en attente".

**Objets externes de sortie :**

- propositions du centre : liste des locations correspondant à la demande du client.
- proposition assurance : on fait part au client qu'il a la possibilité de prendre une assurance "annulation".

### EV2 :

**Objets externes d'entrée :**

- proposition acceptée : le client accepte une des propositions faites par le centre
- choix assurance : il choisit de prendre l'assurance ou non.

**Opération :** demande de réservation, le centre prend en compte la demande de réservation.

**Actions :**

- consultation : plusieurs objets sont consultés dont le client, l'hébergement ainsi que ses disponibilités, les réservations effectives et complètes de dates correspondant à la demande du client et la demande du client, dans l'état "en attente". Ainsi, on écarte tous les hébergements déjà réservés aux dates demandées par le client afin de ne pas attribuer un hébergement déjà réservé.
- création réservation : une réservation est créée dans l'état "en attente" et avec ou sans assurance.
- maj demande : la demande est mise à jour, elle passe de l'état "en attente" à l'état "satisfaite".

### EV3 :

**Objets externes d'entrée :**

- versement arrhes : le client verse les arrhes, c'est-à-dire 20% du montant de la réservation.

**Opération** : prise en compte versement, le centre prend en compte le versement des arrhes.

**Actions :**

- consultation : on consulte la réservation dans l'état "en attente".
- maj réservation : la réservation passe de l'état "en attente" à "effective", si moins de 8 jours se sont passés entre la date de réservation et la date de versement des arrhes.
- création paiement : un paiement est créé, d'un montant égal au versement, si moins de 8 jours se sont passés entre la date de réservation et la date de versement des arrhes.

**EV4 :**

**Objets externes d'entrée :**

- annulation réservation : un client annule un réservation en attente ou acceptée.

opération : analyse annulation, le centre prend en compte la demande d'annulation de réservation.

**Actions :**

- consultation : la réservation est consultée.
- maj réservation : la réservation est mise à jour, son état est maintenant "annulée".
- création paiement : un paiement est créé, c'est un remboursement dû au client, si assurance = "vraie" est etatres = "effective" ou "complète", montant = somme de tous les paiements dont l'attribut remboursement = "faux".

**Objet interne de sortie** : réservation dans l'état "annulée" est source de l'événement interne EV5.

**Objets externes de sortie :**

- demande de paiement de 50% du montant, si etatres = "effective" ou "complète" et (dateres - 30 jours) < dateanul < (dateres - 15 jours).
- demande de paiement de 90% du montant, si etatres = "effective" ou "complète" et (dateres - 15 jours) < dateanul < (dateres).

.

**EV5 :**

**Objets internes d'entrée :**

- réservation dans l'état "annulée".

OU

- réservation dans l'état "non versée".

**Opération** : analyse demandes en attente, le centre reprend les demandes en attente.

**Actions :**

- consultation : les demandes en attentes sont consultées, ainsi que les clients concernés.

**Objet externe de sortie :**

- propositions du centre : liste des locations correspondant à la demande du client.

**EV6 :****Objets temporel :**

- jour : tous les jours.

**Opération :** analyse échéances, le centre analyse les échéances.

**Actions :**

- consultation : on consulte les réservations, les clients, les prestataires et les saisons.
- maj réservation : la réservation est mise à jour, son état est maintenant "non versée", traitement effectué pour toutes les réservations dont l'état est "en attente" et la date de réservation + 8 jours est supérieure à la date courante et pour toutes les réservations dont l'état est "effective" et la date de début de séjour est inférieure à la date courante.

**Objet interne de sortie :** réservation dans l'état "non versée" est source de l'événement interne EV5.

**Objets externes de sortie :**

- rappel client : un rappel est envoyé à tous les clients dont la réservation est dans l'état "effective" et dont la date courante est la date de début du séjour.
- transmission dossier : le dossier de tous les clients dont la réservation est dans l'état "effective" et dont la date courante a dépassé la date de début du séjour est transmis au service de contentieux.
- versement prestataire : le centre verse les sommes qui leur reviennent à tous les prestataires concernés par les réservations dont la date du début de séjour est supérieure à la date de début de saison, dont la date de fin de séjour est inférieure à la date de fin de saison et dont la date de fin de saison est la date courante.

**EV7 :****Objet externe d'entrée :**

- réception paiement : un client paie une réservation..

opération : réception paiement, le centre prend en compte le paiement du client.

**Actions :**

- consultation : la réservation dans l'état "effective" est consultée.
- création paiement : un paiement est créé, de montant égal au versement.

**Objet interne de sortie :** paiement qui est source de l'événement interne EV8.

**EV8 :****Objet interne d'entrée :**

- paiement, le paiement d'une réservation.

opération : finalisation paiement, le centre analyse le paiement d'une réservation afin de vérifier si le montant total de la réservation a été versé.

**Options :**

- consultation : la réservation dans l'état "effective" est consultée.
- maj réservation : l'état de la réservation passe à "complète" si le montant de la réservation est égal à la somme des paiements la concernant.

**EV9 :**

**Objet externe d'entrée :**

- ajout d'une offre : un prestataire confie une nouvelle location au centre ou une nouvelle disponibilité pour une location déjà existante.

**Opération :** prise en compte ajout, le centre prend en compte la nouvelle location.

**Actions :**

- consultation : on consulte le prestataire ainsi que l'hébergement.
- création hébergement : on crée la nouvelle location, si l'hébergement n'existe pas.
- création disponibilité : on crée une disponibilité pour l'hébergement

**EV10 :**

**Objet externe d'entrée :**

- reprise gestion directe : un prestataire décide de reprendre la gestion d'une de ses locations qui n'a pas été affectée.

**Opération :** prise en compte reprise, le centre prend en compte la reprise d'une location.

**Actions :**

- consultation : on consulte le prestataire ainsi que l'hébergement et les disponibilités de l'hébergement.
- suppression disponibilité : on supprime la disponibilité concernant l'hébergement repris, si aucune réservation dans l'état "effective" ou "complète" n'est faite sur l'hébergement .

## **4.3 Conditions et facteurs**

**EV1 :**

- c1 : si le client n'existe pas dans la base.

**EV2 :**

- c2 : si l'hébergement est disponible.

**EV3 :**

- c3 : si getDate() - dateres <= 8 jours, c'est-à-dire qu'il s'est passé maximum 8 jours entre la date de réservation et aujourd'hui.

**EV4 :** pour toutes les conditions de cet événement : `dateanul = getDate()`

- c4 : si `etatres = "effective" ou "complète"` et  $(datedebutres - 30 \text{ jours}) < dateanul < (datedebutres - 15 \text{ jours})$ , c'est à dire que l'annulation est faite entre 30 et 15 jours avant le premier jour de réservation.

- c5 : si `etatres = "effective" ou "complète"` et  $(datedebutres - 15 \text{ jours}) < dateanul < (datedebutres)$ , c'est à dire que l'annulation est faite 15 jours avant le premier jour de réservation.

- c6 : si `assurance = "vraie"` est `etatres = "effective" ou "complète"`, `montant = somme de tous les paiements dont l'attribut remboursement = "faux"`

**EV5 :**

- c10 : sil existe au moins une demande à satisfaire.

**EV6 :**

- f1 : pour toutes les réservations dont :

- `etatres = "en attente"` et  $dateres + 8 \text{ jours} > getDate()$

- `etatres = "effective"` et  $datedebutres < getDate()$

- f2 : pour toutes les reservations dont `etatres = "effective"` et  $getDate() = datedebutres$

- f3 : pour toutes les réservations dont `etatres = "effective"` et  $getDate() > datedebutres$

- f4 : pour toutes les réservations dont  $datedebutres > datedebutsais$  et  $datefinres < datefinsais$  et  $datefinsais = getDate()$

**EV8 :**

- c7 : `montant = somme des paiements`

**EV9 :**

- c8 : Si l'hébergement n'existe pas

**EV10 :**

- c9 : si l'hébergement n'a pas été loué c'est-à-dire qu'il n'y a pas de réservation "effective" ou "complète" pour l'hébergement

## 5 Requêtes répondant aux besoins en informations

Les requêtes suivantes permettent d'obtenir des informations concernant les entités de la base. Elles se trouvent dans la catégorie "interrogation" de maquette.

- Toutes les activités d'intérieurs pour les enfants pour une station donnée

```
SELECT * FROM Activite  
WHERE INTERIEUR = 1  
AND ENFANTACT = 1
```

- Tous les hébergements du prestataire

```
SELECT * FROM Hebergement h  
INNER JOIN Prestataire p ON h.NOPREST = p.NOPREST  
WHERE NOMPREST = "Robin Jeremy"
```

- Tous les clients en attente d'envoi de proposition

```
SELECT NOM, PRENOM, EMAILCLIENT FROM Demande d  
INNER JOIN Client c ON c.NOCLIENT = d.NOCLIENT  
WHERE ETATDEM = "renvoie proposition"
```

- Les disponibilités de l'année

```
SELECT * FROM Disponibilite  
WHERE DATEDEBDISP > "2012-01-01"  
AND DATEFINDISP < "2012-12-31"
```

- Le bénéfice sur la période

```
SELECT SUM(MONTANTPAIE) as benefice FROM Paiement
```

- Les particularités accessibles aux handicapées

```
SELECT NOMPART, ADRESSEPART, DESCRIPTIONPART FROM Particularite  
WHERE HANDI_ACCESSIBLE = 1
```

- Les réservations annulées

```
SELECT * FROM Reservation  
WHERE ETATRES = "annule"
```



- Les clients qui ont une réservation non versées (fraudeurs)

```
SELECT NOM, PRENOM FROM Client c
INNER JOIN Demande d ON c.NOCLIENT =
d.NOCLIENT
INNER JOIN Reservation r ON d.NDEM = r.NDEM WHERE ETATRES = "refusé"
```

## 6 Application

### 6.1 Présentation

Notre application tourne sur un service d'hébergement à l'adresse :  
<http://touriste.freaxmind.pro>

Vos informations d'authentification sont (nom d'utilisateur / mot de passe) :

- odile / tour
- asma / tour

Les technologies utilisées sont les suivantes:

- MySQL 5.1
- PHP 5.3 (objet)
- Yii 1.1.10 : framework PHP
- jQuery : framework JavaScript
- Blueprint : framework CSS

### 6.2 Utilisation

L'interface se divise en 3 modules, accessibles depuis la page d'accueil et le menu haut:

- **Gestion** : interface de gestion complètes des entités de la base. Permet de créer les objets statiques (types de prestataires, hébergements) et de corriger si besoin les objets dynamiques (demandes, réservations)
- **Interrogation** : permet d'exécuter des requêtes SQL, de les enregistrer et de les relancer plus tard
- **Traitement** : traduction des modèles conceptuels de traitement

## 6.3 Scripts

### a) Création des tables

```
CREATE TABLE IF NOT EXISTS `Activite` (  
  `NOACT` int(11) NOT NULL AUTO_INCREMENT,  
  `NOTYPACT` int(11) NOT NULL,  
  `NOMACT` varchar(30) NOT NULL,  
  `INTERIEUR` tinyint(1) NOT NULL,  
  `ENFANTACT` tinyint(1) NOT NULL,  
  PRIMARY KEY (`NOACT`),  
  KEY `NOTYPACT` (`NOTYPACT`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=9 ;  
CREATE TABLE IF NOT EXISTS `APour` (  
  `NOSAISON` int(11) NOT NULL,  
  `NOHEBERG` int(11) NOT NULL,  
  `PRIX` float NOT NULL,  
  PRIMARY KEY (`NOSAISON`,`NOHEBERG`),  
  KEY `NOHEBERG` (`NOHEBERG`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Client` (  
  `NOCLIENT` int(11) NOT NULL AUTO_INCREMENT,  
  `NOM` varchar(30) NOT NULL,  
  `PRENOM` varchar(30) NOT NULL,  
  `DATENAISS` date NOT NULL,  
  `SEXE` char(1) NOT NULL,  
  `SITMARITAL` varchar(30) NOT NULL,  
  `TELEPHONECLIENT` varchar(15) NOT NULL,  
  `EMAILCLIENT` varchar(50) NOT NULL,  
  `ADRESSECLIENT` varchar(100) NOT NULL,  
  PRIMARY KEY (`NOCLIENT`),  
  UNIQUE KEY `EMAILCLIENT` (`EMAILCLIENT`),  
  UNIQUE KEY `TELEPHONECLIENT` (`TELEPHONECLIENT`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;
```

```
CREATE TABLE IF NOT EXISTS `Demande` (  
  `NDEM` int(11) NOT NULL AUTO_INCREMENT,  
  `NOCLIENT` int(11) NOT NULL,  
  `NOSTAT` int(11) NOT NULL,  
  `NOTYPH` int(11) NOT NULL,  
  `NOTYPEPREST` int(11) NOT NULL,  
  `ETATDEM` varchar(30) NOT NULL,  
  `CODEATTENTE` int(11) NOT NULL,  
  `DATDEM` date NOT NULL,  
  `DATEDEBUTRES` date NOT NULL,  
  `DATEFINRES` date NOT NULL,  
  `NBPERSRES` int(11) NOT NULL DEFAULT '1',  
  PRIMARY KEY (`NDEM`),
```

```

KEY `NOCLIENT` (`NOCLIENT`),
KEY `NOSTAT` (`NOSTAT`),
KEY `NOTYPH` (`NOTYPH`),
KEY `NOTYPEPREST` (`NOTYPEPREST`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=59 ;

```

```

CREATE TABLE IF NOT EXISTS `Dispo` (
  `NODISP` int(11) NOT NULL,
  `NOHEBERG` int(11) NOT NULL,
  PRIMARY KEY (`NODISP`,`NOHEBERG`),
  KEY `NOHEBERG` (`NOHEBERG`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE IF NOT EXISTS `Disponibilite` (
  `NODISP` int(11) NOT NULL AUTO_INCREMENT,
  `DATEDEBDISP` date NOT NULL,
  `DATEFINDISP` date NOT NULL,
  PRIMARY KEY (`NODISP`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=17 ;

```

```

CREATE TABLE IF NOT EXISTS `Hebergement` (
  `NOHEBERG` int(11) NOT NULL AUTO_INCREMENT,
  `NOPREST` int(11) NOT NULL,
  `NOTYPH` int(11) NOT NULL,
  `ADRESSE` varchar(100) NOT NULL,
  `QUALITE` varchar(30) NOT NULL,
  `SURFACE` float NOT NULL,
  `NBLITADULT` int(11) NOT NULL,
  `NBLITENFANT` int(11) NOT NULL,
  `WIFI` tinyint(1) NOT NULL,
  `RESTAURATION` tinyint(1) NOT NULL,
  `GESTIONAGENCE` tinyint(1) NOT NULL,
  PRIMARY KEY (`NOHEBERG`),
  KEY `NOPREST` (`NOPREST`),
  KEY `NOTYPH` (`NOTYPH`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=19 ;

```

```

CREATE TABLE IF NOT EXISTS `Interrogation` (
  `noreq` int(11) NOT NULL AUTO_INCREMENT,
  `nomreq` varchar(100) NOT NULL,
  `sqlreq` text NOT NULL,
  PRIMARY KEY (`noreq`),
  UNIQUE KEY `nom` (`nomreq`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=22 ;

```

```
CREATE TABLE IF NOT EXISTS `Offert` (
  `NOSERV` int(11) NOT NULL,
  `NOPREST` int(11) NOT NULL,
  PRIMARY KEY (`NOSERV`,`NOPREST`),
  KEY `NOPREST` (`NOPREST`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Paiement` (
  `NOPAIE` int(11) NOT NULL AUTO_INCREMENT,
  `NORES` int(11) NOT NULL,
  `NOTYPPAIE` int(11) NOT NULL,
  `LIBELLEPAIE` varchar(50) NOT NULL,
  `MONTANTPAIE` float NOT NULL,
  `DATEPAIE` date NOT NULL,
  `REMBOURSEPAIE` tinyint(1) NOT NULL,
  PRIMARY KEY (`NOPAIE`),
  KEY `NOTYPPAIE` (`NOTYPPAIE`),
  KEY `NORES` (`NORES`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=46 ;
```

```
CREATE TABLE IF NOT EXISTS `Particularite` (
  `NOPART` int(11) NOT NULL AUTO_INCREMENT,
  `NOMPART` varchar(30) NOT NULL,
  `ADRESSEPART` varchar(100) NOT NULL,
  `DESCRIPTIONPART` text NOT NULL,
  `HANDI_ACCESSIBLE` tinyint(1) NOT NULL,
  PRIMARY KEY (`NOPART`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=12 ;
```

```
CREATE TABLE IF NOT EXISTS `Possible` (
  `NOACT` int(11) NOT NULL,
  `NOSTAT` int(11) NOT NULL,
  PRIMARY KEY (`NOACT`,`NOSTAT`),
  KEY `NOSTAT` (`NOSTAT`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Prestataire` (
  `NOPREST` int(11) NOT NULL AUTO_INCREMENT,
  `NOSTAT` int(11) NOT NULL,
  `NOTYPP` int(11) NOT NULL,
  `NOMPREST` varchar(30) NOT NULL,
  `TELEPHONEPREST` varchar(15) NOT NULL,
  `EMAILPREST` varchar(50) NOT NULL,
  PRIMARY KEY (`NOPREST`),
  UNIQUE KEY `EMAILPREST` (`EMAILPREST`),
  KEY `NOSTAT` (`NOSTAT`),
  KEY `NOTYPP` (`NOTYPP`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=10 ;
```

```

CREATE TABLE IF NOT EXISTS `Reservation` (
  `NORES` int(11) NOT NULL AUTO_INCREMENT,
  `NDEM` int(11) NOT NULL,
  `NOHEBERG` int(11) NOT NULL,
  `DATERES` date NOT NULL,
  `MONTANTRES` float NOT NULL,
  `DATANNUL` date DEFAULT NULL,
  `ETATRES` varchar(30) NOT NULL,
  `ASSURANCE` tinyint(1) NOT NULL,
  PRIMARY KEY (`NORES`),
  UNIQUE KEY `NDEM` (`NDEM`),
  KEY `NOHEBERG` (`NOHEBERG`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=38 ;

```

```

CREATE TABLE IF NOT EXISTS `Saison` (
  `NOSAISON` int(11) NOT NULL AUTO_INCREMENT,
  `DATEDEBS` date NOT NULL,
  `DATEFINS` date NOT NULL,
  PRIMARY KEY (`NOSAISON`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=9 ;

```

```

CREATE TABLE IF NOT EXISTS `Service` (
  `NOSERV` int(11) NOT NULL AUTO_INCREMENT,
  `NOMSERVICE` varchar(30) NOT NULL,
  `COMPRIS` tinyint(1) NOT NULL,
  PRIMARY KEY (`NOSERV`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=7 ;

```

```

CREATE TABLE IF NOT EXISTS `Station` (
  `NOSTAT` int(11) NOT NULL AUTO_INCREMENT,
  `NOMSTAT` varchar(30) NOT NULL,
  `ADRESSESTAT` varchar(100) NOT NULL,
  `TELEPHONESTAT` varchar(15) NOT NULL,
  `EMAILSTAT` varchar(50) NOT NULL,
  PRIMARY KEY (`NOSTAT`),
  UNIQUE KEY `EMAILSTAT` (`EMAILSTAT`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=7 ;

```

```

CREATE TABLE IF NOT EXISTS `TypeActivite` (
  `NOTYPACT` int(11) NOT NULL AUTO_INCREMENT,
  `NOMTYPACT` varchar(30) NOT NULL,
  PRIMARY KEY (`NOTYPACT`),
  UNIQUE KEY `NOMTYPACT` (`NOMTYPACT`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=7 ;

```

```

CREATE TABLE IF NOT EXISTS `TypeHeberg` (
  `NOTYPH` int(11) NOT NULL AUTO_INCREMENT,
  `NOMTYPH` varchar(30) NOT NULL,
  PRIMARY KEY (`NOTYPH`),

```

```

    UNIQUE KEY `NOMTYPH` (`NOMTYPH`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=13 ;

```

```

CREATE TABLE IF NOT EXISTS `TypePaiement` (
  `NOTYPPAIE` int(11) NOT NULL AUTO_INCREMENT,
  `LIBELLETYPPAIE` varchar(30) NOT NULL,
  PRIMARY KEY (`NOTYPPAIE`),
  UNIQUE KEY `LIBELLETYPPAIE` (`LIBELLETYPPAIE`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=4 ;

```

```

CREATE TABLE IF NOT EXISTS `TypePrest` (
  `NOTYPP` int(11) NOT NULL AUTO_INCREMENT,
  `NOMTYP` varchar(30) NOT NULL,
  PRIMARY KEY (`NOTYPP`),
  UNIQUE KEY `NOMTYP` (`NOMTYP`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

```

```

CREATE TABLE IF NOT EXISTS `Voir` (
  `NOPART` int(11) NOT NULL,
  `NOSTAT` int(11) NOT NULL,
  PRIMARY KEY (`NOPART`,`NOSTAT`),
  KEY `NOSTAT` (`NOSTAT`),
  KEY `NOPART` (`NOPART`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

## **b) Contraintes**

```

ALTER TABLE `Activite`
  ADD CONSTRAINT `Activite_ibfk_1` FOREIGN KEY (`NOTYPACT`) REFERENCES
`TypeActivite` (`NOTYPACT`);

```

```

ALTER TABLE `APour`
  ADD CONSTRAINT `APour_ibfk_4` FOREIGN KEY (`NOHEBERG`) REFERENCES
`Hebergement` (`NOHEBERG`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `APour_ibfk_3` FOREIGN KEY (`NOSAISON`) REFERENCES
`Saison` (`NOSAISON`) ON DELETE CASCADE ON UPDATE CASCADE;

```

```

ALTER TABLE `Demande`
  ADD CONSTRAINT `Demande_ibfk_10` FOREIGN KEY (`NOSTAT`) REFERENCES
`Station` (`NOSTAT`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Demande_ibfk_6` FOREIGN KEY (`NOTYPH`) REFERENCES
`TypeHeberg` (`NOTYPH`),
  ADD CONSTRAINT `Demande_ibfk_7` FOREIGN KEY (`NOTYPEPREST`)
REFERENCES `TypePrest` (`NOTYPP`),
  ADD CONSTRAINT `Demande_ibfk_9` FOREIGN KEY (`NOCLIENT`) REFERENCES
`Client` (`NOCLIENT`) ON DELETE CASCADE ON UPDATE CASCADE;

```

```

ALTER TABLE `Dispo`
  ADD CONSTRAINT `Dispo_ibfk_2` FOREIGN KEY (`NOHEBERG`) REFERENCES
`Hebergement` (`NOHEBERG`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Dispo_ibfk_1` FOREIGN KEY (`NODISP`) REFERENCES
`Disponibilite` (`NODISP`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `Hebergement`
  ADD CONSTRAINT `Hebergement_ibfk_3` FOREIGN KEY (`NOPREST`) REFERENCES
`Prestataire` (`NOPREST`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Hebergement_ibfk_2` FOREIGN KEY (`NOTYPH`) REFERENCES
`TypeHeberg` (`NOTYPH`);

ALTER TABLE `Offert`
  ADD CONSTRAINT `Offert_ibfk_2` FOREIGN KEY (`NOPREST`) REFERENCES
`Prestataire` (`NOPREST`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Offert_ibfk_1` FOREIGN KEY (`NOSERV`) REFERENCES `Service`
(`NOSERV`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `Paiement`
  ADD CONSTRAINT `Paiement_ibfk_3` FOREIGN KEY (`NORES`) REFERENCES
`Reservation` (`NORES`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Paiement_ibfk_1` FOREIGN KEY (`NOTYPPAIE`) REFERENCES
`TypePaiement` (`NOTYPPAIE`);

ALTER TABLE `Possible`
  ADD CONSTRAINT `Possible_ibfk_2` FOREIGN KEY (`NOSTAT`) REFERENCES
`Station` (`NOSTAT`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Possible_ibfk_1` FOREIGN KEY (`NOACT`) REFERENCES `Activite`
(`NOACT`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `Prestataire`
  ADD CONSTRAINT `Prestataire_ibfk_2` FOREIGN KEY (`NOTYPP`) REFERENCES
`TypePrest` (`NOTYPP`),
  ADD CONSTRAINT `Prestataire_ibfk_1` FOREIGN KEY (`NOSTAT`) REFERENCES
`Station` (`NOSTAT`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `Reservation`
  ADD CONSTRAINT `Reservation_ibfk_3` FOREIGN KEY (`NDEM`) REFERENCES
`Demande` (`NDEM`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Reservation_ibfk_4` FOREIGN KEY (`NOHEBERG`) REFERENCES
`Hebergement` (`NOHEBERG`) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE `Voir`
  ADD CONSTRAINT `Voir_ibfk_1` FOREIGN KEY (`NOPART`) REFERENCES
`Particularite` (`NOPART`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `Voir_ibfk_2` FOREIGN KEY (`NOSTAT`) REFERENCES `Station`
(`NOSTAT`) ON DELETE CASCADE ON UPDATE CASCADE;

```

### c) Insertions

```
INSERT INTO `Activite` (`NOACT`, `NOTYPACT`, `NOMACT`, `INTERIEUR`,  
`ENFANTACT`) VALUES
```

```
(1, 1, 'tennis', 0, 1),  
(2, 4, 'ski', 0, 1),  
(3, 1, 'piscine', 0, 1),  
(4, 1, 'golf', 0, 0),  
(5, 2, 'karaoke', 1, 1),  
(6, 3, 'speed dating', 1, 0),  
(8, 5, 'contes de noel', 0, 1);
```

```
INSERT INTO `APour` (`NOSAISON`, `NOHEBERG`, `PRIX`) VALUES
```

```
(6, 2, 4000),  
(6, 3, 1000),  
(6, 4, 1500),  
(6, 6, 500),  
(8, 5, 30000);
```

```
INSERT INTO `Client` (`NOCLIENT`, `NOM`, `PRENOM`, `DATENAIS`, `SEXE`,  
`SITMARITAL`, `TELEPHONECLIENT`, `EMAILCLIENT`, `ADRESSECLIENT`) VALUES  
(2, 'Duo', 'Julia', '1989-02-01', 'F', 'Mariée', '0258963254', 'jul@ul.com', 'adresse julia duo'),  
(3, 'Hur', 'Ben', '1979-11-22', 'H', 'célibataire', '0102030405', 'ben.hur@gladiateur.com', 'Rue  
du petit peplum'),  
(4, 'Ron', 'Anthony', '1966-05-16', 'H', 'divorcé', '0908070605', 'anthony.ron@star.com',  
'Avenue du touriste'),  
(5, 'Apo', 'Junior', '1955-10-10', 'H', 'célibataire', '0761819834', 'apo.fils@stargate.com', '7e  
porte des étoiles');
```

```
INSERT INTO `Demande` (`NDEM`, `NOCLIENT`, `NOSTAT`, `NOTYPH`,  
`NOTYPEPREST`, `ETATDEM`, `CODEATTENTE`, `DATDEM`, `DATEDEBUTRES`,  
`DATEFINRES`, `NBPERSRES`) VALUES
```

```
(44, 2, 1, 11, 1, 'validé', 0, '2010-03-01', '2010-04-01', '2010-04-07', 5),  
(45, 5, 1, 4, 1, 'validé', 0, '2012-01-01', '2012-02-01', '2012-02-07', 2),  
(56, 2, 1, 5, 2, 'validé', 0, '2012-04-27', '2012-05-15', '2012-05-30', 1),  
(57, 4, 1, 5, 2, 'renvoie proposition', 1, '2012-04-27', '2012-05-15', '2012-05-30', 1),  
(58, 2, 3, 2, 2, 'validé', 0, '2012-04-27', '2012-05-15', '2012-05-30', 1);
```

```
INSERT INTO `Dispo` (`NODISP`, `NOHEBERG`) VALUES
```

```
(9, 2),  
(10, 2),  
(11, 2),  
(16, 2),  
(11, 3),  
(12, 3),  
(13, 3),  
(14, 3),  
(9, 4),  
(10, 4),
```



(11, 4),  
 (12, 4),  
 (13, 4),  
 (14, 4),  
 (9, 5),  
 (10, 5),  
 (11, 5),  
 (12, 5),  
 (13, 5),  
 (14, 5),  
 (15, 5),  
 (9, 6),  
 (10, 6),  
 (11, 6),  
 (12, 6),  
 (13, 6),  
 (14, 6),  
 (9, 16),  
 (10, 16),  
 (11, 16),  
 (9, 17),  
 (10, 17),  
 (11, 17),  
 (9, 18),  
 (10, 18),  
 (11, 18);

```

INSERT INTO `Disponibilite` (`NODISP`, `DATEDEBDISP`, `DATEFINDISP`) VALUES
(9, '2012-03-01', '2012-03-31'),
(10, '2012-04-01', '2012-04-30'),
(11, '2012-05-01', '2012-05-31'),
(12, '2012-06-01', '2012-06-30'),
(13, '2012-07-01', '2012-07-31'),
(14, '2012-08-01', '2012-08-30'),
(15, '2010-01-01', '2010-12-31'),
(16, '2012-02-01', '2012-02-28');
  
```

```

INSERT INTO `Hebergement` (`NOHEBERG`, `NOPREST`, `NOTYPH`, `ADRESSE`,
`QUALITE`, `SURFACE`, `NBLITADULT`, `NBLITENFANT`, `WIFI`, `RESTAURATION`,
`GESTIONAGENCE`) VALUES
(2, 1, 4, 'aa', 'neuf', 10, 2, 3, 1, 1, 1),
(3, 2, 5, 'zz', 'récent', 15, 4, 1, 1, 0, 1),
(4, 5, 2, 'Toulon', 'neuf', 50, 0, 0, 0, 1, 1),
(5, 1, 11, '30 avenue de la mer\r\nToulon', 'neuf', 250, 10, 5, 1, 0, 1),
(6, 6, 2, '25 rue du pont\r\nToulon', 'neuf', 300, 0, 0, 0, 0, 1),
(16, 2, 5, 'Rue cyfflé', 'neuf', 24, 2, 1, 1, 0, 0),
(17, 2, 5, 'Rue cyfflé', 'neuf', 24, 2, 1, 1, 0, 0),
(18, 2, 5, 'rue cyfflé', 'neuf', 35, 2, 1, 1, 0, 0);
  
```

```
INSERT INTO `Offert` (`NOSERV`, `NOPREST`) VALUES
```

```
(4, 1),  
(5, 1),  
(4, 4),  
(5, 4),  
(6, 5);
```

```
INSERT INTO `Paiement` (`NOPAIE`, `NORES`, `NOTYPPAIE`, `LIBELLEPAIE`,  
`MONTANTPAIE`, `DATEPAIE`, `REMBOURSEPAIE`) VALUES
```

```
(27, 26, 2, 'versement arrhes', 6000, '2010-03-07', 0),  
(28, 26, 1, 'reception paiement', 24000, '2010-03-14', 0),  
(42, 36, 2, 'versement arrhes', 200, '2012-04-27', 0),  
(43, 36, 2, 'reception paiement', 400, '2012-04-27', 0),  
(44, 36, 2, 'reception paiement', 400, '2012-04-27', 0),  
(45, 37, 2, 'versement arrhes', 100, '2012-04-27', 0);
```

```
INSERT INTO `Particularite` (`NOPART`, `NOMPART`, `ADRESSEPART`,  
`DESCRIPTIONPART`, `HANDI_ACCESSIBLE`) VALUES
```

```
(1, 'dune', 'La Baule', 'La dune du Guézy s"élève derrière celle de Mazy jusqu"à une altitude  
de 25 mètres et se raccorde au sillon de Guérande vers la route de Nérac.', 0),  
(2, 'plage', 'La Baule', 'La plage, que la ville partage avec les communes du Pouliguen et de  
Pornichet, mesure plus de huit kilomètres et considérée par beaucoup comme « l"une des  
plus belles plages d"Europe ». Elle borde les quartiers de La Baule-les-Pins, La Baule-  
Centre et Casino-Benoît. Près du Pouliguen, la plage Benoît est réputée pour son sable  
blanc très fin. Son estran recèle de grandes quantités de coques', 0),  
(3, 'Aiguille du midi', 'Chamonix', 'L"aiguille du Midi fait partie des aiguilles de Chamonix,  
dans le massif du Mont-Blanc. Culminant à 3 842 mètres, elle est la plus haute des aiguilles  
de Chamonix', 1),  
(4, 'Mer de Glace', 'Chamonix', 'La Mer de Glace est un glacier situé sur la face nord du  
massif du Mont-Blanc, formé de la jonction de trois glaciers plus petits que sont le glacier du  
Tacul, le glacier de Leschaux et le glacier de Talèfre.', 0),  
(5, 'Cathédrale Notre-Dame-de-la-Se', 'Toulon', 'La cathédrale Sainte-Marie-de-la-Seds de  
Toulon est la cathédrale du diocèse de Toulon, créé au Ve siècle.', 1),  
(6, 'Opéra de Toulon', 'Toulon', 'L"opéra de Toulon est un bâtiment spécialement conçu pour  
la représentation des opéras. Il est situé dans la ville de Toulon en France.', 1),  
(7, 'magasin LVMH', 'Luxe', 'Vins, spiritueux, mode, maroquinerie, parfums, cosmétiques,  
montres, joaillerie, distribution sélective, médias', 1),  
(8, 'Yves Saint Laurent Beauté', 'Luxe', 'Yves Saint Laurent Beauté est une maison de  
parfums française qui crée ou commercialise les marques YSL, Boucheron, Stella  
McCartney, Maison Martin Margiela, Viktor&Rolf, Cacharel et Diesel.', 1),  
(9, 'Musée du bonbon', 'Uzès', 'Le Musée du bonbon d"Uzès est centré sur l"histoire et la  
fabrication des bonbons de la marque Haribo.', 1);
```

```
INSERT INTO `Possible` (`NOACT`, `NOSTAT`) VALUES
```

```
(2, 1),  
(3, 1),  
(5, 1),  
(8, 1),  
(2, 2),
```

(3, 2),  
(6, 2),  
(8, 2),  
(1, 3),  
(2, 3),  
(1, 4),  
(3, 4),  
(2, 5);

```
INSERT INTO `Prestataire` (`NOPREST`, `NOSTAT`, `NOTYPP`, `NOMPREST`,  
`TELEPHONEPREST`, `EMAILPREST`) VALUES  
(1, 1, 1, 'Robin Jeremy', '0589653287', 'r.jeremy@ul.com'),  
(2, 1, 2, 'Apoté', '0258785412', 'apote@ul.com'),  
(3, 2, 1, 'Anne-Sophie Duhaut', '0369875421', 'as@ul.com'),  
(4, 2, 3, 'Mederic', '0125859632', 'mederic@ul.com'),  
(5, 3, 1, 'Jean Durand', '0589632154', 'jd@ul.com'),  
(6, 3, 2, 'Jeanne Smith', '0418589632', 'js@ul.com'),  
(7, 4, 2, 'Paul Mesange', '0852541256', 'mesange@ul.com'),  
(8, 5, 1, 'Julien Dupont', '0235698541', 'jul@ul.com');
```

```
INSERT INTO `Reservation` (`NORES`, `NDEM`, `NOHEBERG`, `DATERES`,  
`MONTANTRES`, `DATANNUL`, `ETATRES`, `ASSURANCE`) VALUES  
(26, 44, 5, '2010-03-07', 30000, NULL, 'complete', 0),  
(27, 45, 2, '2012-01-07', 4000, NULL, 'refusé', 1),  
(36, 56, 3, '2012-04-27', 1000, NULL, 'complete', 0),  
(37, 58, 6, '2012-04-27', 500, '2012-04-27', 'annule', 0);
```

```
INSERT INTO `Saison` (`NOSAISON`, `DATEDEBS`, `DATEFINS`) VALUES  
(6, '2011-01-01', '2012-12-31'),  
(8, '2010-01-01', '2010-12-31');
```

```
INSERT INTO `Service` (`NOSERV`, `NOMSERVICE`, `COMPRIS`) VALUES  
(2, 'garde d'enfant', 0),  
(3, 'téléphone', 1),  
(4, 'télévision', 1),  
(5, 'service de chambre', 0),  
(6, 'alcool', 1);
```

```
INSERT INTO `Station` (`NOSTAT`, `NOMSTAT`, `ADRESSESTAT`, `TELEPHONESTAT`,  
`EMAILSTAT`) VALUES  
(1, 'La Baule', '35 avenue de la libération 44500 La Baule', '0445859631', 'labaule@ul.com'),  
(2, 'ChamonixSki', '22 rue du ski 74000 Chamonix', '0528745123', 'chamonixski@ul.com'),  
(3, 'ToulonDetente', '78 rue de la mer 83690 Toulon', '0258963274',  
'toulondetente@ul.com'),  
(4, 'Luxe station', '96 rue du luxe', '0148756963', 'luxestation@ul.com'),  
(5, 'Station coeur', '58 avenue de l'amour 88000 Epinal', '0369875421',  
'stationcoeur@ul.com');
```

```
INSERT INTO `TypeActivite` (`NOTYPACT`, `NOMTYPACT`) VALUES  
(2, 'chanson'),  
(5, 'excursion'),  
(3, 'rencontre'),  
(1, 'sport été'),  
(4, 'sport hiver');
```

```
INSERT INTO `TypeHeberg` (`NOTYPH`, `NOMTYPH`) VALUES  
(5, 'appartement'),  
(4, 'chambre'),  
(12, 'chambre d"hôtes'),  
(2, 'emplacement'),  
(11, 'villa');
```

```
INSERT INTO `TypePaiement` (`NOTYPPAIE`, `LIBELLETYPPAIE`) VALUES  
(2, 'carte bleue'),  
(1, 'chèque'),  
(3, 'comptant');
```

```
INSERT INTO `TypePrest` (`NOTYPP`, `NOMTYP`) VALUES  
(2, 'camping'),  
(1, 'centre de vacance'),  
(3, 'hôtel');
```

```
INSERT INTO `Voir` (`NOPART`, `NOSTAT`) VALUES  
(6, 1),  
(8, 1),  
(9, 1),  
(3, 2),  
(4, 2),  
(5, 3),  
(6, 3),  
(7, 4),  
(8, 4),  
(2, 5),  
(3, 5),  
(4, 5),  
(9, 5);
```

## 6.4 Captures d'écran

Voir annexe 4.

# 7 Conclusion

## 7.1 Réflexion sur les outils et méthodes utilisées

Notre équipe était déjà familière à l'utilisation de MySQL, de PHP et des méthodes Merise pour réaliser des applications métiers.

La principale nouveauté de ce projet concerne l'emploi de la démarche intégratrice et de l'outil Power AMC pour atteindre ces mêmes objectifs.

L'un des plus grands avantages de la Démarche Intégratrice est sa compatibilité avec Merise. Les méta-éléments sont similaires mais ne sont pas associés de la même façon. Cela rend l'apprentissage plus simple et permet de chaîner les événements internes plus efficacement. On peut ainsi retrouver tous les déclencheurs internes et externes à partir d'un schéma de synthèse pour simplifier la conception.

Pour rappel, ces liens ont été représentés par du code PHP dans le cadre de notre application.

Le principal bémol de la démarche reste sa popularité, qui limite le choix des outils et le suivi de projet. A ce désavantage près, cette méthode offre une alternative convaincante.

L'outil Power AMC a été un facteur de productivité important pour ce projet. La génération automatique des schémas permet de gagner du temps, tout en maintenant la vue abstraite et physique des données. On peut juste lui reprocher de ne pas fonctionner sous Linux ou Mac, à l'heure où des alternatives libres tels que PostgreSQL (et ses outils) percent sur le marché des serveurs de base de données.

Il est difficile de juger une méthode sur la réalisation du premier projet, car les savoirs ne sont pas pleinement acquis. Toutefois, cette expérience laisse envisager les bénéfices à long terme pour un besoin gardant de nombreuses étapes manuelles.

## 7.2 Difficultés rencontrées

L'étape de conception de ce projet n'a pas été d'une grande difficulté, mais s'est pourtant révélée plus longue que ce que nous avions estimé. Les discussions ont été nombreuses concernant les choix de conception, et notre groupe hétérogène a dû convenir de conventions tout au long de la réalisation pour favoriser le travail de groupe.

Pour nous aider, nous sommes souvent revenus à des cas concrets et logiques afin d'illustrer les règles de gestion. Nous sommes conscients que notre implémentation n'est qu'une possibilité parmi d'autres, mais nous espérons que nos choix répondront aux exigences de l'énoncé.

Au niveau de la réalisation, l'intégration d'un framework PHP nous a permis d'augmenter notre productivité pour réaliser davantage de contraintes d'intégrité. L'appréhension a été difficile, surtout qu'il n'existe aucun cours pour nous former à ce genre d'outil dans la formation.

Passée la période d'apprentissage, les traitements ont été rapides à implémenter.

Nous espérons que vous apprécierez notre travail. Vous pouvez nous contacter pour obtenir davantage d'informations.