

Projet Base de donnée Gestion d'un centre médical

LIMA DA SILVA Julie
NAUDIN Antoine
ODOR Mustafa
SIMMA Gridsada

L3 Informatique
Université méditerranée aix-marseille II

Responsable : Claude SABATIER

L'année universitaire : 2010/2011

Table des matières

ENONCE	2
INTRODUCTION.....	3
I. DONNEES THEORIQUES	4
1. MODELE CONCEPTUEL DE DONNEES (MCD)	4
Dictionnaire des données.....	4
Règles de gestion.....	5
Dictionnaire des propriétés	6
Détermination des objets.....	7
Détermination des relations.....	7
Le MCD de merise	8
2. PASSAGE DU SCHEMA CONCEPTUEL AU SCHEMA RELATIONNEL	9
Distinction des RG inter/intra schéma.....	9
Création des tables SQL.....	9
Déclarations des triggers.....	11
Autres triggers et vues	14
II. MISE EN PRATIQUE.....	16
Remplissage des tables.....	16
Test sur les triggers	18
III. MODULE FORMS Centre hospitalisé régional	19
Conclusion.....	22
Bibliographies.....	23

ENONCE

GESTION D'UN CENTRE MEDICAL **Les rendez-vous dans les différents services**

Un centre hospitalier, composé de différents services médicaux, reçoit des malades à la journée, pour effectuer une intervention prescrite par une ordonnance. Nous voulons automatiser l'établissement de ces ordonnances, c'est-à-dire trouver, pour une intervention donnée, une journée où les services concernés puissent recevoir le malade.

LES CONTRAINTES DE FONCTIONNEMENT SONT LES SUIVANTES :

Dans une journée donnée, un malade subit une et une seule intervention. Intervention qui impose l'exécution d'un certain nombre d'actes connus, dans un ordre chronologique fixé.

Chaque service :

- Est ouvert 4 jours sur 7, pendant les cinq créneaux suivants :
8h-10h, 10h-12h, 12h-14h, 14h-16h, 16h-18h ,
- Est compétent dans l'exécution de plusieurs actes de types différents (3 au maximum)
- Exécute au plus 6 actes (différents ou non) dans un créneau de 2 heures

Chaque ordonnance:

- Est relative à une seule intervention et à un seul client,
- Précise le jour où les actes composant cette intervention (4 au maximum) pourront être exécutés,
- Indique pour chaque acte, le créneau horaire et les services où le client est attendu.

Les traitements :

-A tout moment, le bureau planning doit être capable de proposer à un client des rendez-vous précis pour l'exécution d'une intervention qui lui a été prescrite.

-En début de journée, le bureau planning doit fournir à chaque service, la liste ordonnée des clients attendus et des actes à exécuter.

-Au départ du client, le service Facturation doit établir la facture globale de chaque intervention, en détaillant les coûts de chacun des actes qui la composent, ainsi que la part restant à la charge du malade.

Des informations statistiques sont également demandées.

-En fin de mois : pour chaque service, et chaque acte qu'il peut assurer, afficher le nombre total d'actes effectués et le pourcentage de ces actes par rapport à l'activité globale mensuelle du service.

-En fin de trimestre : pour chaque acte pris en charge par le centre, et pour chaque service assurant un tel acte, pourcentage réalisé par ce service relatif au nombre total d'actes de ce type réalisés au centre, pendant le trimestre écoulé.

INTRODUCTION

Une base de données est un lot d'information stockée de façon organisé et structuré où le contenu peut être facilement manipulé.

L'objectif du projet est de créer une base de données capable de gérer automatiquement le planning et les rendez-vous d'un centre médical selon ses différents services et créneaux disponible.

Plus précisément, un malade présente une ordonnance prescrivant des actes pour une intervention au centre médical. On interroge la base de données pour trouver un rendez-vous en fonction des disponibilités des services et des actes.

Pour cela, nous avons utilisé la base de données « oracle SQL » mais aussi le programme « FORMS Oracle ».

I. DONNEES THEORIQUES

1. MODELE CONCEPTUEL DE DONNEES (MCD)

Dictionnaire des données

Pour l'obtenir nous avons lus l'énoncé pour sortir les attributs et leurs contraintes. La constitution de ce dictionnaire est une épuration des données, c'est -à-dire la suppression des synonymes et des poly sèmes. Il permet aussi de créer une fiche descriptive où chaque donnée est décrite par plusieurs rubriques

Nom	Définition	Type	Contrainte règle de calcul
Nom	Nom du malade	élémentaire	
Prenom	Prénom du malade	élémentaire	
Adresse	Adresse du malade	élémentaire	
NomJournee	Jours de la semaine où le service est ouvert	élémentaire	
NomActe	Nom des actes proposé par le centre médical	élémentaire	
NomService	Nom des services du centre médical	élémentaire	
Creneau	Numéro des créneaux horaires composant la journée	élémentaire	1<Créneau<5 (5 Créneau/jour)
NumActe	Numéro attribué à l'acte pour une ordonnance	élémentaire	1<Actes<4 (4 Actes /ordonnance)
NumOrdonnance	Numéro d'une ordonnance d'un patient	élémentaire	
DateRDV	Date de passage des actes pour une ordonnance	élémentaire	
CoutActe	Coût d'un acte	élémentaire	>0
tauxremb	Pourcentage pris en charge par le sécurité sociale	élémentaire	0<tauxremb<100
PartRestMalade	Part restante du malade a payer	calculée	RC1
PourActeServCentr	Nombres total d'actes réaliser par mois par le centre médicale	calculée	RC2
NbActesServMensuel	Nombres total d'actes réaliser par mois et par service	calculée	RC3
NumSecuSociale	Numéro de sécurité sociale d'un malade	élémentaire	
CouTotal	Coût total des actes par ordonnance	calculée	RC4

RC1: $PartRestMal = \Sigma (CoutActe \text{ de l'ordonnance}) * (1 - TauxRemb \text{ du malade} / 100).$

RC2: $PourActServCent = (\Sigma (\text{actes sur un mois}) \text{ par services} * 100) / \Sigma (\text{actes total sur ce mois}).$

RC3: $NbActServMens = // \text{IDEM que RD2 } \Sigma (\text{actes sur un mois}) \text{ pour chaque service.}$

RC4: $Couttotal = \Sigma (CoutActe \text{ d'une ordonnance}).$

Règles de gestion

Ces règles sont données dans l'énoncé et certaines sont des règles logiques ajoutées :

- ❖ *RG1: Une Intervention se passe sur une journée.*
- ❖ *RG2: Un malade reçoit une intervention par journée.*
- ❖ *RG3: Chaque service est ouvert 4 jours dans la semaine.*
- ❖ *RG4: Au plus 3 actes proposé par services.*
- ❖ *RG5: Un service exécute au plus 6 actes par créneau.*
- ❖ *RG6: Une Ordonnance par malade et par intervention.*
- ❖ *RG7: Une intervention est composée de 4 actes max.*
- ❖ *RG8: Chaque acte dans une ordonnance est fixé chronologiquement.*
- ❖ *RG9: Les services doivent être ouverts pour chaque acte prescrit.*

Dictionnaire des propriétés

Ce dictionnaire est la reprise du dictionnaire des données contenant uniquement nos données élémentaire. Ces données sont toutes des propriétés:

Nom	Définition	Type	Contrainte règle de calcul
Nom	Nom du malade	élémentaire	
Prenom	Prénom du malade	élémentaire	
Adresse	Adresse du malade	élémentaire	
NomJournee	Jour de la semaine où le service est ouvert	élémentaire	
NomActe	Nom de l'acte réalisé par le service du centre médical	élémentaire	
NomService	Nom du service du centre médical	élémentaire	
Creneau	Créneau horaire d'un acte pour un patient	élémentaire	$1 < \text{Creneau} < 5$ (5 Creneau/jour)
NumActe	Numéro attribué à l'acte pour une ordonnance	élémentaire	$1 < \text{Actes} < 4$ (4 Actes /ordonnance)
NumOrdonnance	Numéro d'une ordonnance d'un patient	élémentaire	
DateRDV	Date de passage des actes pour une ordonnance	élémentaire	
CoutActe	Coût d'un acte réalisé par le centre médical	élémentaire	
tauxremb	Pourcentage du prix pris en charge par la sécurité sociale	élémentaire	$0 < \text{tauxremb} < 100$
NumSecuSociale	Numéro de sécurité sociale d'un malade	élémentaire	

Détermination des objets

<u>Malade</u>
- NumSécuritéSociale
-Nom
-Prenom
-Adresse

<u>Actes</u>
- NomActe
-CoutActe

<u>Journée</u>
-NomJournée

<u>SécuritéSociale</u>
-tauxremb

<u>Service</u>
- NomService

<u>Ordre</u>
- NumActe

<u>Ordonnance</u>
- NumeroOrdonnance
-DateRendezVous
-CoutTotal
-PartRestMalade

Détermination des relations

EstOuvert :

Fait correspondre pour chaque service, ses jours d'ouvertures.

Clé primaire : (NomService et NomJournée)

Accueil :

Fait correspondre pour chaque service, les actes qu'il effectue.

Clé primaire : (NomService et NomActe)

Estcompose :

Relie les numeros d'ordonnances aux actes prescrits pour celles-ci, indique aussi l'ordre

Clé primaire : (NumOrdonnance, NomActe et NumActe)

RDV :

Donne le Jour et les créneaux pour chaque acte à effectuer dans les services.

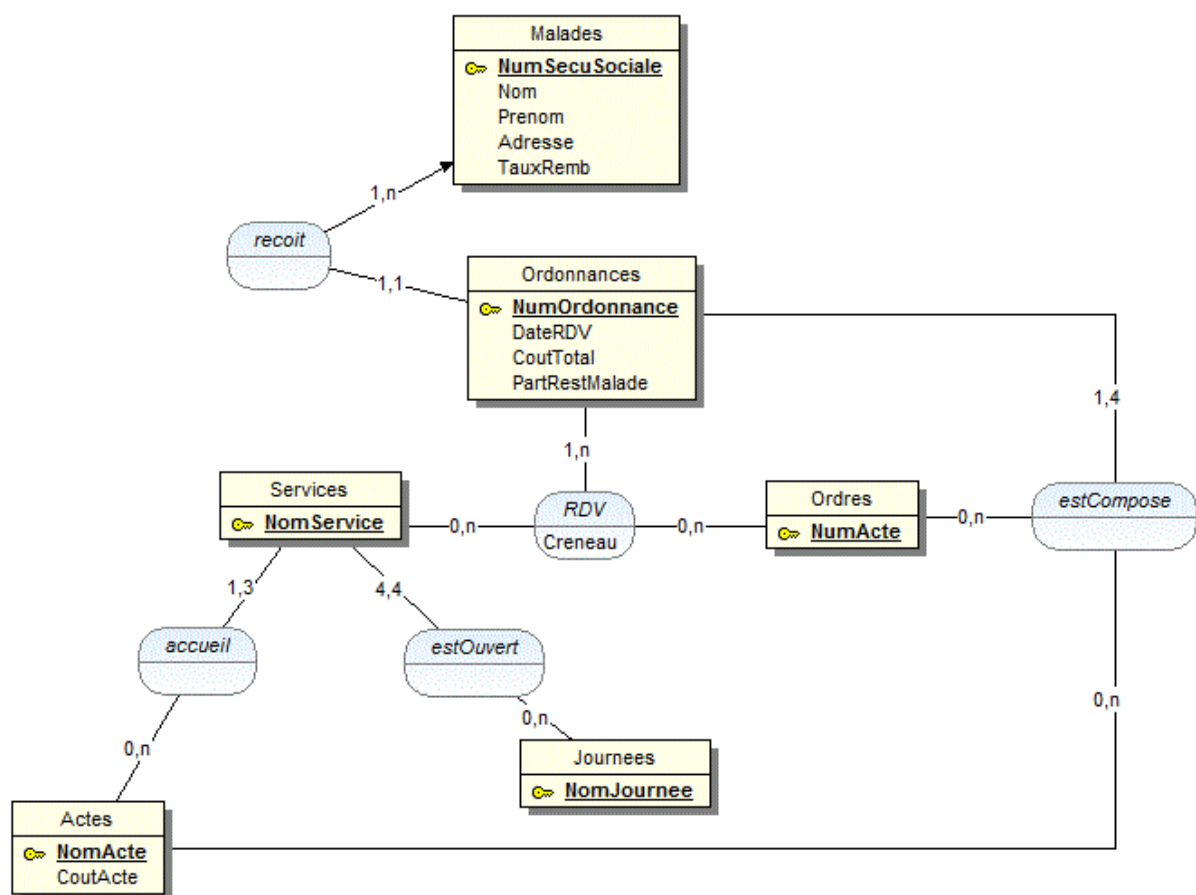
Clé primaire : (NumOrdonnance, NumActes et NomService)

Le MCD de merise

MERISE est une méthode de conception, de développement et de réalisation de projets informatiques. Le but de cette méthode est d'arriver à concevoir un système d'information. La méthode MERISE est basée sur la séparation des données et des traitements à effectuer en plusieurs modèles conceptuels et physiques

Le modèle conceptuel des données (**MCD**) a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information, ici sous forme de schéma. Il s'agit donc d'une représentation des données, facilement compréhensible, permettant de décrire le système d'information à l'aide d'entités. Le MCD repose aussi sur les notions d'association et sur les notions de relations.

Ce schéma est obtenu grâce à la détermination de notre objet ainsi que la détermination des relations. Il permet ainsi de corriger le dictionnaire et les règles en évitant les redondances.



2. PASSAGE DU SCHEMA CONCEPTUEL AU SCHEMA RELATIONNEL

Distinction des RG inter/intra schéma

RG inter schéma

RG10

RG5

RG intra schéma

RG1: dateRDV not null, dans ordonnances.

RG2: contrainte unique dans ordonnances.

RG6: clé primaire, table ordonnances.

RG7: table ordre + clé primaire est compose.

Création des tables SQL

CREATE TABLE Malades (

```
NumSecuSociale    NUMBER(20) constraint PK_Malades PRIMARY KEY,  
Nom               VARCHAR(15) NOT NULL,  
Prenom            VARCHAR(15) NOT NULL,  
Adresse           VARCHAR(60),  
TauxRemb          NUMBER(2) NOT NULL constraint Dom_Taux check (Tauxremb > 0  
and Tauxremb <= 100)  
);
```

CREATE TABLE Actes (

```
NomActe           VARCHAR(20) constraint DOM_NomActe check (NomActe =  
upper(NomActe)), CoutActe NUMBER(10,2) NOT NULL constraint  
Dom_Acte check (CoutActe > 0), constraint PK_Actes PRIMARY KEY  
(NomActe)  
);
```

CREATE TABLE Ordonnances (

```
NumOrdonnance     NUMBER(5) constraint PK_Ordonnances PRIMARY KEY,  
NumSecuSociale    NUMBER(20) constraint FK_Malade references Malades(NumSecuSociale),  
DateRDV           Date NOT NULL,  
CoutTotal         NUMBER(5) constraint Dom_CoutTotal check (CoutTotal >= 0),  
PartRestMalade    NUMBER(5) constraint Dom_PartRestMalade check (PartRestMalade >= 0),  
                  constraint Dom_Ordonnance check (PartRestMalade <= CoutTotal),  
                  constraint RG2 UNIQUE (NumSecuSociale, DateRDV)  
);
```

CREATE TABLE Journees (

```
NomJournee        VARCHAR2(15) constraint PK_Journees PRIMARY KEY,  
                  constraint DOM_Journees check (NomJournee in  
('LUNDI', 'MARDI', 'MERCREDI', 'JEUDI', 'VENDREDI', 'SAMEDI', 'DIMANCHE'))  
);
```

CREATE TABLE Services (

```
NomService        VARCHAR(15) constraint DOM_NomService check (NomService =  
upper(NomService)) constraint PK_Services PRIMARY KEY  
);
```

```
CREATE TABLE Ordres (  
  NumActe          NUMBER(1) not NULL constraint PK_NumAct PRIMARY KEY,constraint  
                  DOM_RG7 check (NumActe < 5 and NumActe > 0)  
);
```

```
CREATE TABLE estOuvert (  
  NomService       VARCHAR2(15) constraint FK_Service_Journee1 references  
                  Services(NomService),  
  NomJournee       VARCHAR2(15)constraint FK_Service_Journee2 references  
                  Journees(NomJournee), constraint PK_estOuvert PRIMARY KEY  
                  (NomService,NomJournee)  
);
```

```
CREATE TABLE estCompose (  
  NumActe          NUMBER(1) constraint FK_Ordre1 references Ordres(NumActe),  
  NomActe          VARCHAR2(20) constraint FK_Actes1 references Actes(NomActe),  
  NumOrdonnance    NUMBER(5) constraint FK_Ordonnance references  
                  Ordonnances(NumOrdonnance),constraint PK_estCompose PRIMARY KEY  
                  (NumActe,NomActe,NumOrdonnance)  
);
```

```
CREATE TABLE accueil (  
  NomActe          VARCHAR(50) constraint FK_Actes2 references Actes(NomActe),  
  NomService       VARCHAR(50) constraint FK_Services1 references Services(NomService),  
                  constraint PK_execute PRIMARY KEY (NomActe,NomService)  
);
```

```
CREATE TABLE RDV (  
  NumActe          NUMBER(1) constraint FK_Ordre2 references Ordres(NumActe),  
  NomService       VARCHAR(15) constraint FK_Services2 references Services(NomService),  
  NumOrdo          NUMBER(15) constraint FK_NumOrdo references  
                  Ordonnances(NumOrdonnance),  
  Creneau          NUMBER(1) constraint dom_Creneau check (Creneau between 1 and 5),  
                  constraint PK_RDV PRIMARY KEY (NumActe,NomService,NumOrdo)  
);
```

Déclarations des triggers

RG1: Une Intervention se passe sur une journée.

RG2: Un malade reçoit une intervention par journée.

RG3: Chaque service est ouvert 4 jours dans la semaine.

RG4: Au plus 3 actes proposé par services.

RG5: Un service exécute au plus 6 actes par créneau.

RG6: Une Ordonnance par malade et par intervention.

RG7: Une intervention est composée de 4 actes max.

RG8: Chaque acte dans une ordonnance est fixé chronologiquement.

RG9: Les services doivent être ouverts pour chaque acte prescrit.

RG3: Chaque service est ouvert 4 jours dans la semaine

CREATE OR REPLACE TRIGGER

```
RG3 AFTER INSERT OR UPDATE ON estouvert
DECLARE r varchar (15);
BEGIN
SELECT nomservice into r
FROM estouvert
GROUP BY nomservice
HAVING count (*)>4;
RAISE TOO_MANY_ROWS;
EXCEPTION
WHEN NO_DATA_FOUND THEN NULL;
WHEN TOO_MANY_ROWS THEN RAISE_APPLICATION_ERROR (-20001, 'on est fermé');
END;
/
```

RG4: Au plus 3 actes proposé par services

CREATE OR REPLACE TRIGGER

```
RG4 AFTER INSERT OR UPDATE ON accueil
DECLARE N integer;
BEGIN
SELECT 1 INTO N FROM accueil A
GROUP BY NomService
HAVING count (*)>3;
RAISE TOO_MANY_ROWS;
EXCEPTION
WHEN NO_DATA_FOUND THEN NULL;
WHEN TOO_MANY_ROWS THEN RAISE_APPLICATION_ERROR (-20002, 'Service dispensant
déjà 3 actes différents');
END;
/
```

RG5: Un service exécute au plus 6 actes par créneau

```
CREATE OR REPLACE TRIGGER
RG5 AFTER INSERT OR UPDATE ON rdv
DECLARE s varchar(15);
BEGIN
SELECT NomService into s
FROM Ordonnances o, RDV r
WHERE r.NumOrdo = o.NumOrdonnance
GROUP BY r.NomService,o.dateRDV,r.Creneau
HAVING count (*)>6;
RAISE TOO_MANY_ROWS;
EXCEPTION
WHEN NO_DATA_FOUND THEN NULL;
WHEN TOO_MANY_ROWS THEN RAISE_APPLICATION_ERROR (-20000,'ServicePlein');
END;
/
```

RG7: Une intervention est composée de 4 actes max.

```
CREATE OR REPLACE TRIGGER
RG7 AFTER INSERT OR UPDATE ON estcompose
DECLARE n number (15);
BEGIN
SELECT o.numordonnance into n
FROM ordonnances o, estcompose c
WHERE o.numordonnance=c.numordonnance
GROUP BY o.numordonnance
HAVING COUNT (*)>4;
RAISE TOO_MANY_ROWS;
EXCEPTION
WHEN NO_DATA_FOUND THEN NULL;
WHEN TOO_MANY_ROWS THEN RAISE_APPLICATION_ERROR (-20002, 'trop d actes pour
ordonnance');
End;
/
```

RG 8:Chaque acte dans une ordonnance est fixé chronologiquement.

```
CREATE OR REPLACE TRIGGER
RG8 AFTER insert or update on rdv
DECLARE o varchar(15);
BEGIN
SELECT r1.numordo INTO o
FROM rdv r1, rdv r2
WHERE r1.numordo=r2.numordo
AND r1.numacte>r2.numacte
AND r1.creneau<=r2.creneau
GROUP BY r1.numordo;
RAISE TOO_MANY_ROWS;
EXCEPTION
WHEN NO_DATA_FOUND THEN NULL;
WHEN TOO_MANY_ROWS THEN RAISE_APPLICATION_ERROR (-20002, 'Ordre non respecter
sur crenaux');
END;
/
```

RG9 : Les services doivent être ouverts pour chaque acte prescrit.

```
CREATE OR REPLACE TRIGGER
RG9 AFTER INSERT OR UPDATE ON estcompose
DECLARE s number(15);

BEGIN

SELECT c.numordonnance into s
FROM estcompose c,accueil a1,accueil a2,ordonnances o
WHERE o.numordonnance=c.numordonnance
AND c.nomacte=a1.nomacte
AND c.nomacte=a2.nomacte
AND a1.nomservice<>a2.nomservice
AND a1.nomservice NOT IN (SELECT nomservice
                           FROM estouvert where nomjournee =
                           to_char(o.daterdv,'FMDAY','NLS_DATE_LANGUAGE=french'))

AND a2.nomservice NOT IN (SELECT nomservice
                           FROM estouvert
                           WHERE nomjournee =
                           to_char(o.daterdv,'FMDAY','NLS_DATE_LANGUAGE=french'))

GROUP BY c.numordonnance;
RAISE TOO_MANY_ROWS ;
EXCEPTION
WHEN NO_DATA_FOUND then null;
WHEN TOO_MANY_ROWS then raise_application_error(-20000,'ServiceFermeOuManquant');

END;
/
```

RG10: Pour une ordonnance, un seul service par acte et par créneau.

(notre table rdv, avec ce clé primaire <Numacte,Numordo,Nomservice> acceptait de ce fait pour une meme ordonnance, plusieurs services enregistrée pour le meme acte, ce qui est impossible.)

```
CREATE OR REPLACE TRIGGER
RG10 AFTER INSERT OR UPDATE ON rdv
DECLARE s number(15);

BEGIN

SELECT r1.numordo into s
FROM rdv r1, rdv r2
WHERE r1.numordo=r2.numordo
AND r1.numacte=r2.numacte
AND r1.nomservice<>r2.nomservice;
RAISE TOO_MANY_ROWS

EXCEPTION

WHEN NO_DATA_FOUND then null;
WHEN TOO_MANY_ROWS then raise_application_error(-20006,'Rendez-vous déjà
prit!');

END;
/
```

RG11: Pour chaque acte attribué à un service, celui-ci doit être en mesure de le faire.

Test si un service prescrit pour un acte, est bien habilité à le faire

```
CREATE OR REPLACE TRIGGER
RG11 AFTER INSERT OR UPDATE ON rdv
DECLARE s varchar(15);
BEGIN
SELECT r.nomservice into s
FROM estcompose c, rdv r
WHERE c.numordonnance=r.numordo
AND c.numacte=r.numacte
AND r.nomservice not in (select a.nomservice from accueil a where
a.numacte=c.numacte);
RAISE TOO_MANY_ROWS
EXCEPTION
WHEN NO_DATA_FOUND then null;
WHEN TOO_MANY_ROWS then raise_application_error(-20006,'Le Service en question
n'' accueille pas cette acte!');
END;
/
```

Autres triggers et vues

Trigger remplissage couttotal, partrestmalade (à mettre sur ordonnances, est compose)

```
CREATE OR REPLACE TRIGGER
Remplissagecout AFTER INSERT OR UPDATE OR DELETE ON estcompose
BEGIN
UPDATE ordonnances o
SET o.couttotal = (select sum(nvl(a.coutacte,0))
FROM estcompose c, actes a
WHERE c.numordonnance=o.numordonnance
AND c.numacte=a.numacte );
end;
/
```

Trigger remplissagepartsrestante

```
CREATE OR REPLACE TRIGGER
Remplissagepartsrestante AFTER INSERT OR UPDATE OR DELETE ON estcompose
BEGIN
UPDATE ordonnances oo
SET oo.partrestmalade = nvl(oo.couttotal,0)*(1-
(SELECT s.tauxremb
FROM malades s,ordonnances o
WHERE oo.numordonnance=o.numordonnance
AND s.numsecusociale=o.numsecusociale
)/100)
WHERE oo.couttotal is not null;
END;
/
```

GERER LA DATE :

On crée une table date_sys:

```
CREATE TABLE Date_SYS (  
  DateC date  
);
```

On crée une fonction date_courant():

```
CREATE or REPLACE function Date_Courant return date  
is dd date;  
BEGIN  
  SELECT dateC into dd from date_sys;  
  return dd;  
END;  
/
```

Pour finir, la procedure change 'date'.

```
CREATE or REPLACE procedure change_date(dd date)  
is  
BEGIN  
  UPDATE date_sys SET dateC=dd;  
  IF SQL%ROWCOUNT<>1 THEN  
    DELETE date_sys;  
    insert into date_sys values(dd);  
  END IF;  
END;  
/
```

Vues planning :

```
CREATE OR REPLACE view Planning (nomservice,nom,nomacte,creneau,daterdv) AS  
SELECT r.nomservice, m.nom, c.nomacte, r.creneau, o.daterdv  
FROM malades m, estcompose c, ordonnances o, rdv r  
  
WHERE o.numordonnance = r.numordo  
AND o.numsecusociale = m.numsecusociale  
AND r.numacte = c.numacte  
AND o.daterdv in ( select dateC from date_sys)  
GROUP BY r.nomservice, m.nom, c.nomacte, o.daterdv, r.creneau;
```

Vues facturation :

```
CREATE OR REPLACE view Facture AS  
SELECT e.numacte, e.nomacte, o.numordonnance, a.coutacte, m.numsecusociale  
FROM estcompose e, ordonnances o, actes a, malades m  
WHERE m.numsecusociale=o.numsecusociale  
      AND o.numordonnance=e.numordonnance  
      AND e.nomacte=a.nomacte;
```


II. MISE EN PRATIQUE

Remplissage des tables

```
insert into malades values(numsecusociale, nom,prenom,adresse,tauxremb);
insert into malades values(1,'naudin','antoine','luminy',25);
insert into malades values(2,'limash','julie','luminy',50);
insert into malades values(3,'grisada','deng','luminy',80);
insert into malades values(4,'naudin','lulu','kingston',99);
insert into malades values(5,'Robert','Pierre','Paris',10);
insert into malades values(6,'Prince','Junior','151, av.Boisson',50);
insert into malades values(7,'King','Shiloh','Netherland',12);
```

```
insert into actes values(nomacte,coutacte);
insert into actes values('BANDAGE',500);
insert into actes values('BANDAGE',500);
insert into actes values('AMPUTAGE',300);
insert into actes values('REPOS',1000);
insert into actes values('POINTS',20);
insert into actes values('RASAGE',1);
insert into actes values('ECOUTE',15);
insert into actes values('SCANNER',70);
insert into actes values('SOIN',10);
```

```
insert into actes values(nomjournee);
insert into journees values('LUNDI');
insert into journees values('MARDI');
insert into journees values('MERCREDI');
insert into journees values('JEUDI');
insert into journees values('VENDREDI');
insert into journees values('SAMEDI');
insert into journees values('DIMANCHE');
```

```
insert into Ordonnances values (NUMORDONNANCE,NUMSECUSOCIALE,DATERDV,COUTTOTAL
, PARTRESTMALADE);
insert into Ordonnances values (1,1,to_date('10/05/1988 00:00'),0,0);
insert into Ordonnances values (2,1,to_date('11/05/1988 00:00'),0,0);
insert into Ordonnances values (3,2,to_date('10/05/1988 00:00'),0,0);
insert into Ordonnances values (4,2,to_date('16/05/1988 00:00'),0,0);
insert into Ordonnances values (5,3,to_date('10/05/1988 00:00'),0,0);
insert into Ordonnances values (6,2,to_date('19/05/1988 00:00'),0,0);
insert into Ordonnances values (7,2,to_date('04/01/0001 00:00'),0,0);
insert into Ordonnances values (8,1,to_date('04/01/0001 00:00'),0,0);
```

```
insert into services values('nomservice');
insert into services values('PSYCHIATRIE');
insert into services values('DERMATOLOGIE');
insert into services values('CHIRURGIE');
insert into services values('TRAUMATOLOGIE');
insert into services values('GYNECOLOGIE');
```

```
insert into ordres values (numacte);
insert into ordres values (1);
insert into ordres values (4);
insert into ordres values (3);
insert into ordres values (2);
```

```

insert into estOuvert values(nomservice,nomjournee);
insert into estOuvert values('TRAUMATOLOGIE','LUNDI');
insert into estOuvert values('TRAUMATOLOGIE','MARDI');
insert into estOuvert values('TRAUMATOLOGIE','MERCREDI');
insert into estOuvert values('TRAUMATOLOGIE','JEUDI');
insert into estOuvert values('DERMATOLOGIE','LUNDI');
insert into estOuvert values('DERMATOLOGIE','MARDI');
insert into estOuvert values('DERMATOLOGIE','MERCREDI');
insert into estOuvert values('DERMATOLOGIE','JEUDI');
insert into estOuvert values('CHIRURGIE','VENDREDI');
insert into estOuvert values('CHIRURGIE','SAMEDI');
insert into estOuvert values('CHIRURGIE','DIMANCHE');
insert into estOuvert values('CHIRURGIE','LUNDI');
insert into estOuvert values('PSYCHIATRIE','MARDI');
insert into estOuvert values('PSYCHIATRIE','SAMEDI');
insert into estOuvert values('PSYCHIATRIE','DIMANCHE');
insert into estOuvert values('PSYCHIATRIE','LUNDI');
insert into estOuvert values('GYNECOLOGIE','VENDREDI');
insert into estOuvert values('GYNECOLOGIE','SAMEDI');
insert into estOuvert values('GYNECOLOGIE','DIMANCHE');

insert into estcompose values(numacte,nomacte,numordonnance);
insert into estcompose values(1,'RASAGE',1);
insert into estcompose values(1,'RASAGE',2);
insert into estcompose values(1,'RASAGE',3);
insert into estcompose values(1,'RASAGE',4);
insert into estcompose values(2,'SCANNER',1);
insert into estcompose values(2,'SCANNER',2);
insert into estcompose values(2,'SCANNER',3);
insert into estcompose values(2,'SCANNER',4);
insert into estcompose values(3,'SOIN',1);
insert into estcompose values(3,'SOIN',2);
insert into estcompose values(3,'SOIN',3);
insert into estcompose values(3,'SOIN',4);
insert into estcompose values(4,'POINTS',1);
insert into estcompose values(4,'POINTS',2);
insert into estcompose values(4,'POINTS',4);

insert into accueil values(nomacte,nomservice);
insert into accueil values('BANDAGE','CHIRURGIE');
insert into accueil values('AMPUTAGE','CHIRURGIE');
insert into accueil values('POINTS','CHIRURGIE');
insert into accueil values('REPOS','TRAUMATOLOGIE');
insert into accueil values('SCANNER','TRAUMATOLOGIE');
insert into accueil values('RASAGE','TRAUMATOLOGIE');
insert into accueil values('SOIN','GYNECOLOGIE');
insert into accueil values('AMPUTAGE','GYNECOLOGIE');
insert into accueil values('RASAGE','GYNECOLOGIE');
insert into accueil values('SOIN','DERMATOLOGIE');
insert into accueil values('SCANNER','DERMATOLOGIE');
insert into accueil values('POINTS','DERMATOLOGIE');
insert into accueil values('ECOUTE','PSYCHIATRIE');
insert into accueil values('SOIN','PSYCHIATRIE');

insert into rdv values(numacte,nomservice,numordo,numcreneux);
insert into rdv values(2,'DERMATOLOGIE',2,2);
insert into rdv values(2,'DERMATOLOGIE',3,3);
insert into rdv values(2,'DERMATOLOGIE',4,2);
insert into rdv values(2,'DERMATOLOGIE',5,2);
insert into rdv values(2,'DERMATOLOGIE',6,2);
insert into rdv values(4,'DERMATOLOGIE',1,5);

```

```

insert into rdv values(4, 'DERMATOLOGIE', 2, 4);
insert into rdv values(4, 'DERMATOLOGIE', 3, 5);
insert into rdv values(4, 'DERMATOLOGIE', 4, 5);
insert into rdv values(4, 'DERMATOLOGIE', 5, 4);
insert into rdv values(3, 'PSYCHIATRIE', 1, 4);
insert into rdv values(3, 'PSYCHIATRIE', 2, 3);
insert into rdv values(3, 'PSYCHIATRIE', 3, 4);
insert into rdv values(3, 'PSYCHIATRIE', 4, 3);
insert into rdv values(3, 'PSYCHIATRIE', 5, 3);
insert into rdv values(3, 'PSYCHIATRIE', 6, 3);

```

Test sur les triggers

RG3:

```

insert into estOuvert values(nomservice,nomjournee);
insert into estOuvert values('GYNECOLOGIE','MERCREDI');
insert into estOuvert values('GYNECOLOGIE','JEUDI'); //Ne marche pas

```

RG4:

```

insert into accueil values(nomacte,nomservice);
insert into accueil values('REPOS','PSYCHIATRIE');
insert into accueil values('BANDAGE','PSYCHIATRIE'); //Ne marche pas

```

RG5:

```

insert into rdv values(numacte,nomservice,numordo,numcreneux);
insert into rdv values(1, 'TRAUMATOLOGIE', 1, 1);
insert into rdv values(1, 'TRAUMATOLOGIE', 2, 1);
insert into rdv values(1, 'TRAUMATOLOGIE', 3, 1);
insert into rdv values(1, 'TRAUMATOLOGIE', 4, 1);
insert into rdv values(1, 'TRAUMATOLOGIE', 5, 1);
insert into rdv values(1, 'TRAUMATOLOGIE', 6, 1);
insert into rdv values(1, 'TRAUMATOLOGIE', 7, 1); // ne marche pas

```

RG8:

```

insert into rdv values(numacte,nomservice,numordo,numcreneux);
insert into rdv values(4, 'DERMATOLOGIE', 6, 1); // ne marche pas
insert into rdv values(4, 'DERMATOLOGIE', 6, 4);

```

RG9:

```

insert into estcompose values(numacte,nomacte,numordonnance);
insert into estcompose values(4, 'AMPUTAGE', 3); // ne marche pas
insert into estcompose values(4, 'POINTS', 3);

```

RG10:

```

insert into rdv values(numacte,nomservice,numordo,numcreneux);
insert into rdv values(2, 'DERMATOLOGIE', 1, 3);
insert into rdv values(2, 'TRAUMATOLOGIE', 1, 3); //ne marche pas

```

RG11:

```

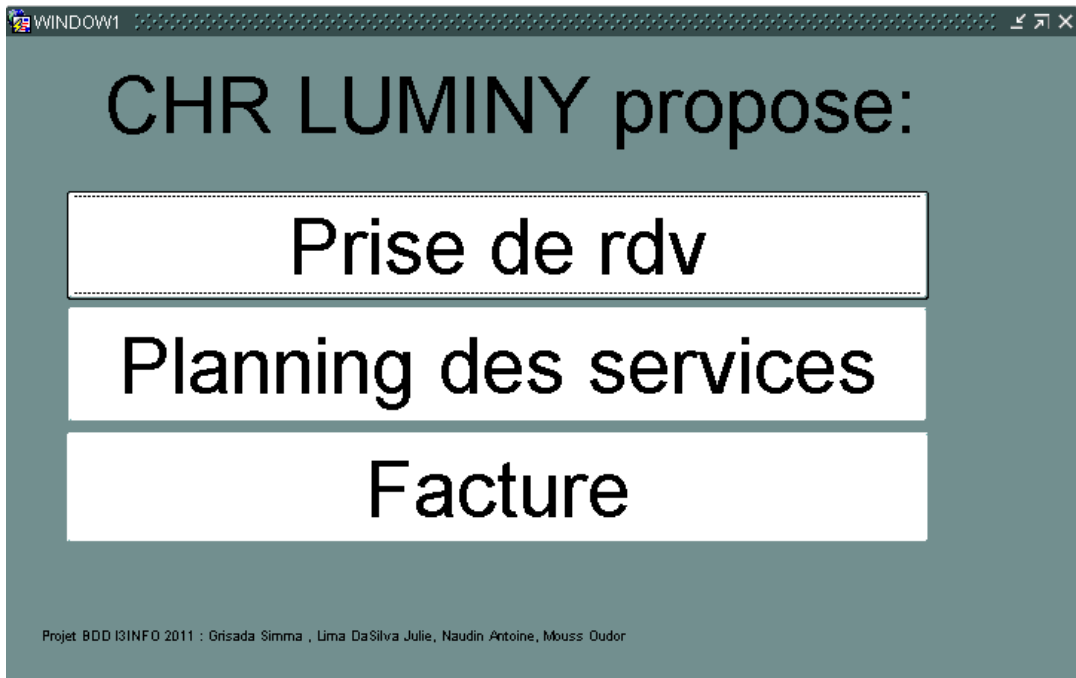
insert into rdv values(numacte,nomservice,numordo,numcreneux);
insert into rdv values(2, 'CHIRURGIE', 1, 3); // ne marche pas
insert into rdv values(2, 'DERMATOLOGIE', 1, 3);

```

III. MODULE FORMS

Centre hospitalisé régional

Module Menu:



WINDOW1

CHR LUMINY propose:

Prise de rdv

Planning des services

Facture

Projet BDD I3INFO 2011 : Grisada Simma , Lima DaSilva Julie, Naudin Antoine, Mouss Oudor

Module Rendez-Vous

Prise de rendez-vous

[Retour accueil](#)

Malades

N° secusociale

Nom

Prenom

Adresse

Tauxremb

[Sauver Malades](#)

Ordonnance

Daterdv

N° ordonnance

N° secusociale

[Sauver Ordonnance](#)

RDV

	N° Acte	N° Ordo	Nom Acte	Nom Service	N° Acte	N° Ordo	Creneau
Sauver Rendez ...	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Sauver Rendez ...	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Sauver Rendez ...	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Sauver Rendez ...	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Ce module fonctionne en mode insertion, il permet à l'utilisateur d'insérer de nouvelles ordonnances Ainsi que des patients et des rendez-vous pour des actes et des services sur les créneaux de la journée.

Les triggers inclus dans la base, empêcheront toutes insertions s'ils ne sont pas respecté.

Le bloc RDV est basé sur deux tables : estcompose et rdv.

Ils sont en relations l'un et l'autre par « numacte » et sont liés par une relation maître/détails au bloc ordonnance par le « numordonnance » qui joint les trois tables.

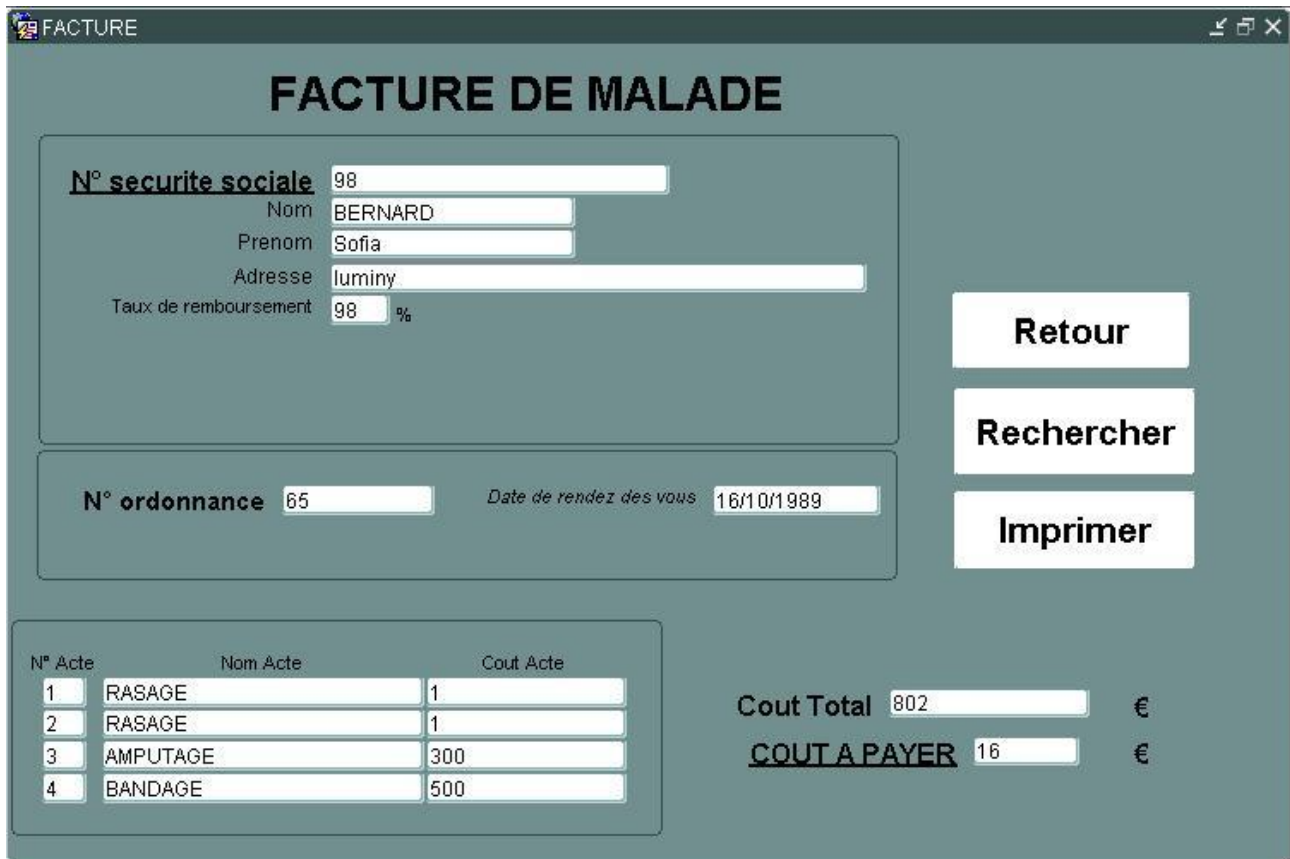
Le bloc malades est lié par « numsecuSociale » au bloc ordonnance.

Il faut rentrer les données dans l'ordre: malade->ordonnance->actes+services+créneaux.

Chaque bouton exécute un commit sur la base pour valider les insertions.

On aurait aimé réussir à inclure des vues affichant dans la liste du bloc « rdv » sur l'attribut « nomservice » uniquement les services hébergeant les actes prescrits, mais nous n'avons pas eu le temps de bien finir.

Module Facture :



FACTURE DE MALADE

N° securite sociale 98

Nom BERNARD

Prenom Sofia

Adresse luminy

Taux de remboursement 98 %

N° ordonnance 65 *Date de rendez vous* 16/10/1989

Retour

Rechercher

Imprimer

N° Acte	Nom Acte	Cout Acte
1	RASAGE	1
2	RASAGE	1
3	AMPUTAGE	300
4	BANDAGE	500

Cout Total 802 €

COUT A PAYER 16 €

Ce bloc de donnée sert à organiser la facture d'un malade. On saisit le numéro de sécurité sociale, puis on appuie sur le bouton « Rechercher »,

On a ensuite les détails sur :

Les coordonnées des malades comme le nom, le prénom, l'adresse, le taux de remboursement,

Ainsi que le numéro d'ordonnance avec sa date de rendez-vous, les numéros d'acte, leurs noms et leurs coûts correspondants (grâce à la vue facture)

L'affichage du coût total et de la part restant à payer au malade se réalise grâce au déclencheur « Remplissagecout » et « Remplissagepartsrestante » sur la table estcomposé.

Module Planning :

Planning des services

Nomservice:

Date Courante:

Nom	Nomservice	Acte	Creneau	Daterdv
NAUDIN	CHIRURGIE	POINTS	1	16/10/1989
Naudin	CHIRURGIE	POINTS	1	16/10/1989
Naudin	CHIRURGIE	POINTS	3	16/10/1989
NAUDIN	CHIRURGIE	POINTS	3	16/10/1989
BERNARD	CHIRURGIE	SCANNER	3	16/10/1989

Lorsqu'une valeur est choisie dans le bloc PLANNING, la relation maitre/détail s'occupe d'aller rechercher dans la vue «PLANNING » les rendez-vous pris à la date dans le bloc DATE_SYS:
 Nom du patient, nom du service, nom de l'acte, nom du créneau, date de l'intervention.

Nous avons rajouté un bouton "date courante" : il nous permet de changer la date pour pouvoir obtenir un planning futur et non seulement celui du jour j.

Conclusion

Ce projet a été intéressant sur plusieurs points:

Il nous a tout d'abord fait découvrir un nouvel outil informatique : Oracle Forms.
 De plus, ce projet était concret et reprenait toutes les notions acquises en cours de bases de données, Comme la conception de base de données ; dictionnaire de données, règle de gestion, dictionnaire des propriétés, des relations, la méthode de merise, le schéma conceptuel au schéma relationnel. Nous avons pu distinguer la complexité de la mise en place d'une base de données, mais en contrepartie prit du plaisir à la voir se concrétiser. Nous nous en sommes sortis avec 11 tables, 2 vues, et 13 déclencheurs.

Malheureusement, par faute de temps nous n'avons pas pu terminer notre projet comme prévu.

Bibliographies

Claude SABATIER, le cours de base de donnée L3 Info Semestre 5, 2010/2011

Claude SABATIER, le cours de base de donnée L3 Info Semestre 6, 2010/2011

<http://sheikyerbouti.developpez.com/forms10g/>, guide oracle 9/10g