Dossier du projet de janvier 2018

Applications informatiques

Par Bastien Dewez et Wilson Weets

Table des matières

Aucune entrée de table des matières n'a été trouvée.

# Enoncé

## Gestion des différents blocs opératoires d’un hôpital

### Utilisateurs

* Chirurgien
* Secrétaire
* Directeur

### Tâches effectuées par le programme

#### Chirurgien

* Réserver une salle d’opération (en y ajoutant les informations du patient)
* Annuler une réservation
* Modifier une réservation
* Consulter ses réservations
* Gérer ses informations
* Consulter ses notifications

#### Secrétaire

* Réserver une salle d’opération (soit à la place d’un chirurgien, soit lors d’une urgence, …)
* Prévenir un chirurgien d’une modification des réservations
* Modifier une réservation
* Supprimer une réservation
* Gérer ses informations
* Consulter les réservations en fonction des salles d’opération
* Consulter la liste des patients inscrit dans des réservations

#### Directeur

* Consulter les réservations en fonction des salles d’opération
* Consulter la liste des patients inscrit dans des réservations
* Gérer ses informations
* Ajouter du personnel (chirurgiens, secrétaires)
* Supprimer du personnel
* Consulter la liste de son personnel

### Cas spéciaux

#### Urgences

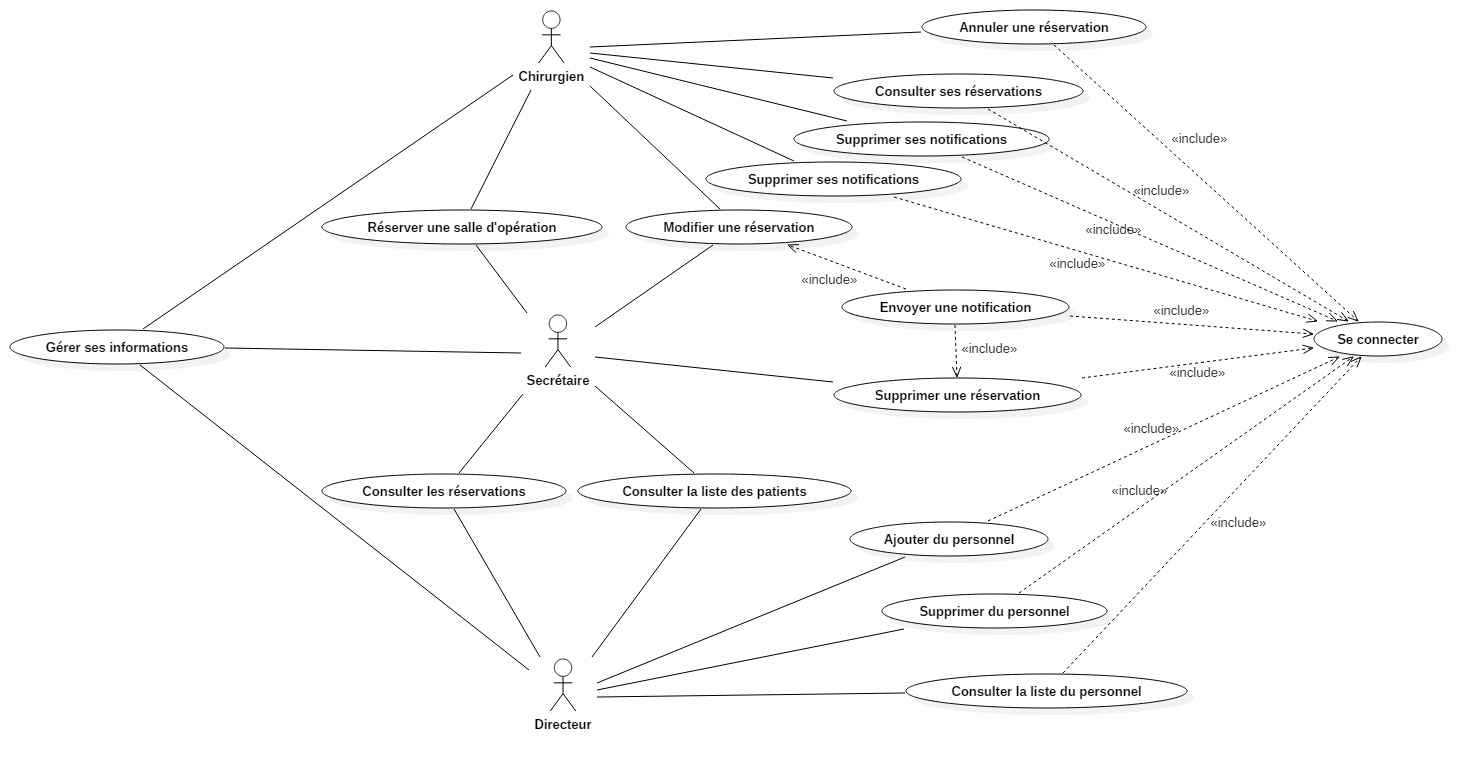
Lorsqu’un patient arriver aux urgences et doit se faire opérer un/une secrétaire doit réserver une salle d’opération en urgence. Si une réservation a dû être enlevée, le chirurgien concerné sera automatiquement prévenu grâce au système de notification.

#### Priorités

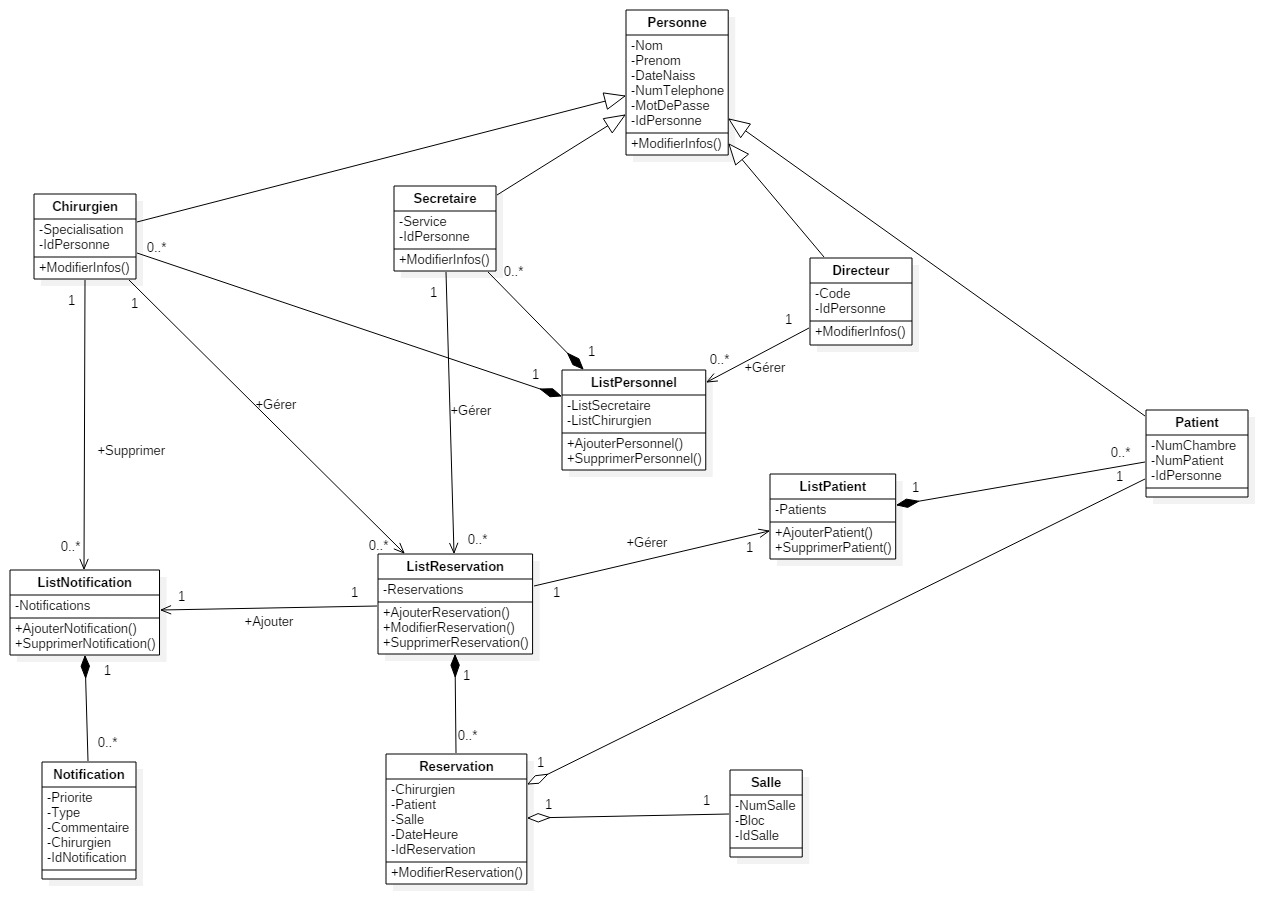
Chaque réservation possèdera une priorité en rapport avec la gravité de la pathologie du patient. Si une réservation a une priorité plus haute qu’une autre, elle sera considérée comme plus urgente et pourra dès lors la remplacer dans l’agenda. Seul les secrétaires peuvent effectuer cette opération. Si un chirurgien veut remplacer une réservation par la sienne, il devra s’adresser au secrétariat.

# Analyse

## Diagramme des Use Cases



## Diagramme de classe

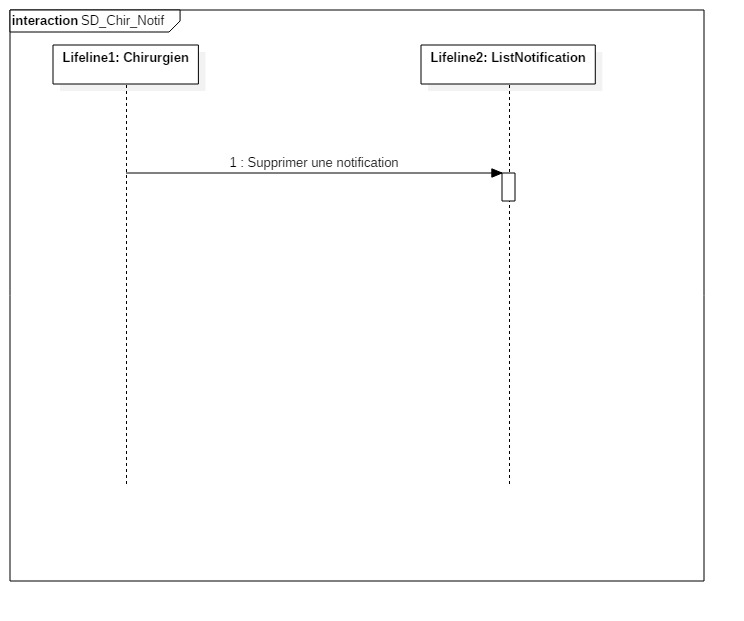


## Diagrammes de séquence

### Lorsque le chirurgien ajoute, modifie ou supprime une réservation

### C:\Users\shado\AppData\Local\Microsoft\Windows\INetCache\Content.Word\SD_Chir_Res.jpg

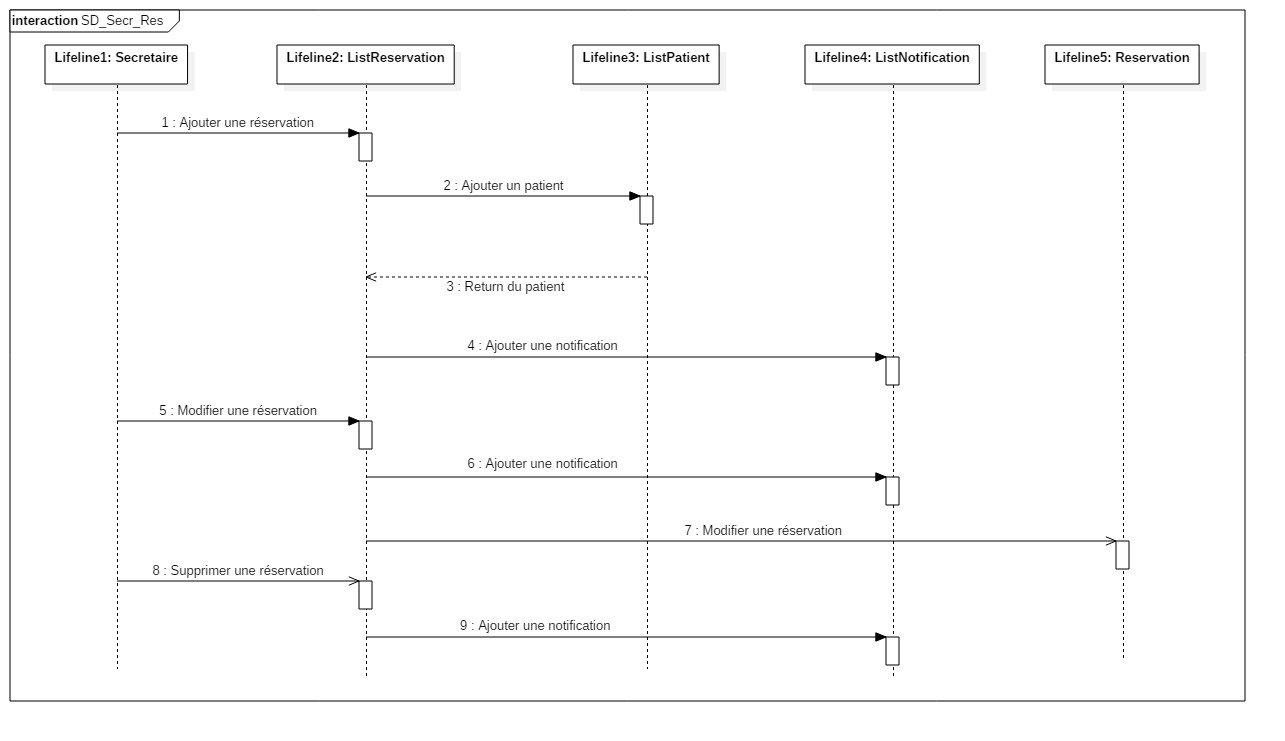
### Lorsque le chirurgien supprime une notification



### Lorsque le directeur ajoute ou supprime du personnel

### C:\Users\shado\AppData\Local\Microsoft\Windows\INetCache\Content.Word\SD_Dir_Pers.jpg

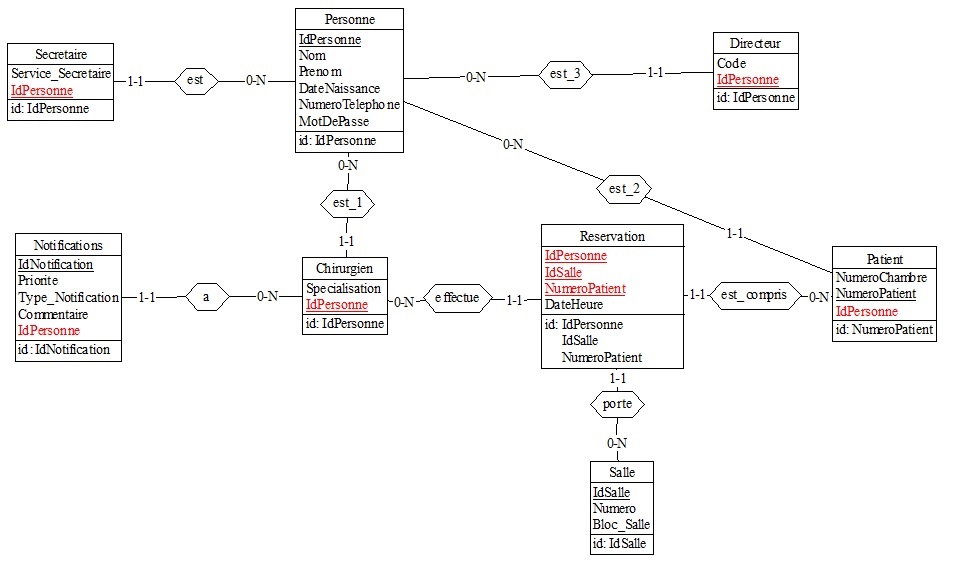
### Lorsque le secrétaire ajoute, modifie ou supprime une réservation



# Ecrans

# Base de données

## Schéma conceptuel de la base de données



## Scripts

### Création de la base de données et des tables (CreationTable.sql)

Ce script crée les tables de notre base de données.

drop table Personne;

drop table Secretaire;

drop table Patient;

drop table Directeur;

drop table Chirurgien;

drop table Notifications;

drop table Salle;

drop table Reservation;

create table Personne (

IdPersonne number not null,

Nom varchar2(50),

Prenom varchar2(50),

DateNaissance date,

NumeroTelephone varchar2(20),

MotDePasse varchar2(30),

constraint PK\_Personne primary key (IdPersonne)

);

create table Secretaire (

IdPersonne number not null,

Service\_Secr varchar2(30),

constraint PK\_Secretaire primary key (IdPersonne),

constraint FK\_Secretaire\_Personne foreign key (IdPersonne)

references Personne (IdPersonne)

);

create table Chirurgien (

IdPersonne number not null,

Specialisation varchar2(30),

constraint PK\_Chirurgien primary key (IdPersonne),

constraint FK\_Chirurgien\_Personne foreign key (IdPersonne)

references Personne (IdPersonne)

);

create table Directeur (

IdPersonne number not null,

Code varchar2(4),

constraint PK\_Directeur primary key (IdPersonne),

constraint FK\_Directeur\_Personne foreign key (IdPersonne)

references Personne (IdPersonne)

);

create table Patient (

IdPersonne number not null,

NumeroChambre varchar2(3),

NumeroPatient varchar2(5) not null,

constraint PK\_Patient primary key (NumeroPatient),

constraint FK\_Patient\_Personne foreign key (IdPersonne)

references Personne (IdPersonne)

);

create table Notifications (

IdNotification number not null,

Priorite integer,

Type\_Notif integer,

Commentaire varchar2(300),

IdPersonne number not null,

constraint PK\_Notification primary key (IdNotification),

constraint FK\_Notification\_Chirurgien foreign key (IdPersonne)

references Chirurgien (IdPersonne)

);

create table Salle (

IdSalle number not null,

Numero varchar2(2),

Bloc\_Salle char,

constraint PK\_Salle primary key (IdSalle)

);

create table Reservation (

IdPersonne number not null,

IdSalle number not null,

NumeroPatient varchar2(5) not null,

DateHeure date,

constraint PK\_Reservation primary key (IdPersonne,IdSalle,NumeroPatient),

constraint FK\_Reservation\_Chirurgien foreign key (IdPersonne)

references Chirurgien (IdPersonne),

constraint FK\_Reservation\_Salle foreign key (IdSalle)

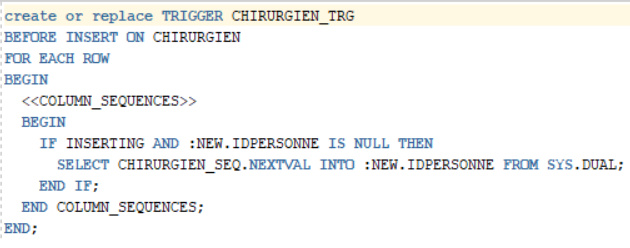
references Salle (IdSalle),

constraint FK\_Reservation\_Patient foreign key (NumeroPatient)

references Patient (NumeroPatient)

);

Les clés primaires de nos tables subissent une incrémentation automatique grâce à des déclencheurs :



### Insertion des premières données (InsertionDonnees.sql)

Ce script insert des données qui vont nous permettre de nous connecter à notre application et d’y effectuer des tests.

insert into Personne (Nom,Prenom,DateNaissance,NumeroTelephone,MotDePasse) values ('Ventura','Manfredi','5/2/1965','0495/15.54.89','Manfrediv65');

insert into Directeur values (1,1234);

insert into Personne (Nom,Prenom,DateNaissance,NumeroTelephone,MotDePasse) values ('Hervieux','Thierry','6/3/1973','0489/89.77.31','Thierryh73');

insert into Chirurgien values (2,'Digestive');

insert into Personne (Nom,Prenom,DateNaissance,NumeroTelephone,MotDePasse) values ('Marquis','Carine','20/8/1972','0487/31.76.98','Carinem72');

insert into Chirurgien values (3,'Obstétrique');

insert into Personne (Nom,Prenom,DateNaissance,NumeroTelephone,MotDePasse) values ('Brasseur','Maurice','11/2/1967','0483/64.62.14','Mauriceb67');

insert into Secretaire values (4,'Ophtalmologie');

insert into Personne (Nom,Prenom,DateNaissance,NumeroTelephone,MotDePasse) values ('Renaud','Francoise','13/7/1987','0472/83.42.68','Francoiser87');

insert into Secretaire values (5,'Génétique');

# Problèmes rencontrés

## Crash du serveur GlassFish

Lors des tests de requêtes vers le service Web, notre serveur GlassFish subissait des crashs. Une exception était alors lancée. Celle-ci nous indiquait que la classe de driver Oracle permettant la connexion à la base de données était introuvable

Pour régler ce problème, nous avons dû copier le fichier contenant ces classes (classes12.jar) dans le répertoire /lib de GlassFish.

# Accès au programme

Pour vous connecter au programme vous pourrez utiliser soit le compte du Directeur, soit un compte de Secrétaire, soit un compte de Chirurgien.

## Comptes

### Directeur

Nom : Manfredi Ventura

Mot de passe : Manfrediv65

### Secrétaires

* Maurice Brasseur => Mauriceb67
* Françoise Renaud => Françoiser87

### Chirurgiens

* Thierry Hervieux => Thierryh73
* Carine Marquis => Carinem72