

Un serveur web couplé à un système de gestion de base de données pour l'enregistrement des données de la surveillance microbiologique d'une zone d'atmosphère contrôlée

Thomas Liautaud
Pharmacien des hôpitaux
Grand Hôpital de l'Est Francilien

26 avril 2018

projet pour le
DU Bionformatique et Datamanagement
Ioannis Nicolis
Université Paris-Descartes

Table des matières

1	Introduction	2
2	Objectif	3
3	Materiel et méthode	3
3.1	Système de gestion de base de données	3
3.1.1	Base de données postgresQL	3
3.1.2	Création des tables	3
3.1.3	Peuplement des tables	6
3.2	Serveur Web	6
4	Résultats	6
5	Discussion	7

1 Introduction

Les mélanges de nutrition parentérale sont utilisés pour les patients dont les apports en nutriments par voie orale sont impossibles ou insuffisants. Ces mélanges sont des préparations parentérales pour perfusion au sens de la Pharmacopée européenne¹ : elles sont stériles et apyrogènes.

Pour la préparation des médicaments stériles les Bonnes pratiques de préparation² (BPP) imposent des exigences particulières en vue de réduire les risques de contamination microbienne, particulaire et pyrogène. Elles définissent notamment les zones d'atmosphère contrôlée (ZAC) qui sont des locaux et équipements dont les qualités microbiologique et particulaire sont contrôlées.

Aux fins de préparation de médicaments stériles, 4 classes de ZAC sont distinguées. Des recommandations pour la surveillance microbiologique des ZAC sont présentées dans la table 1. Les méthodes d'échantillonnages utilisent des boîtes de Petri, des échantillons volumétriques d'air et des contrôles de surface (prélevés au moyen de géloses de contact et/ou écouvillons).

TABLE 1 – Recommandations pour la surveillance microbiologique des zones d'atmosphère contrôlée en activité. Limites de contamination en unité formant colonie (ufc).

Classe	Air	Surfaces	Gants
A	<1	<1	<1
B	10	5	5
C	50	25	-
D	100	50	-

La pharmacie de notre établissement prépare des mélanges de nutrition parentérale pour la néonatalogie. L'unité de préparation des mélanges de nutrition parentérale de notre établissement est constituée d'un isolateur en classe A dans une zone de préparation en classe D. Afin de surveiller la conformité de notre unité aux exigences des BPP, des prélèvements microbiologiques sont effectués quotidiennement. Une quarantaine de points de prélèvements ont été définis :

-
1. Pharmacopée européenne 9^{ème} édition, EDQM
 2. Bonnes Pratiques de Préparations, Afssaps, 2007

- Selon le lieu : isolateur ou salle ;
- Selon le type de prélèvement : air, surface, gant ;
- Selon le dispositif utilisé : boîte de Petri, gélose contact, écouvillon ;
- Selon le point précis de prélèvement : tel gant, tel champ stérile, tel point de la balance...

Moins d’une dizaine de points sont prélevés quotidiennement. Afin de prélever tous les points chaque semaine, un planning de prélèvement répartit les points à prélever sur l’ensemble de la semaine.

Les prélèvements microbiologiques sont traités par le laboratoire d’hygiène du service de Biologie. En cas de contamination microbiologique de l’isolateur, les résultats (en nombre de colonies) sont communiqués immédiatement par téléphone au pharmacien. L’identification des germes est faite ultérieurement. Dès l’identification des germes, un compte rendu signé par un biologiste est rendu à la pharmacie.

Actuellement les prélèvements effectués et leurs résultats sont enregistrés sur un tableur.

2 Objectif

L’objectif de ce projet est de mettre à disposition des utilisateurs un serveur web couplé à système de gestion de base de données qui doit permettre de saisir et consulter les prélèvements microbiologiques et leurs résultats.

3 Matériel et méthode

3.1 Système de gestion de base de données

3.1.1 Base de données postgresSQL

La base de données postgresSQL (version 9.3) est installée sur le système d’exploitation GNU/Linux (Linux Mint 17.3). La base de données est nommée *bacterio_upnp*.

3.1.2 Création des tables

Table disp_prelev (Table 2) référence les dispositifs de prélèvement. Elle a été créée avec la requête :

```
CREATE TABLE disp_prelev(
id SERIAL PRIMARY KEY,
disp VARCHAR(50) NOT NULL
```

TABLE 2 – Table disp_prelev

Nom de variable	Type	Clé primaire	Clé étrangère
id	SERIAL	oui	
disp	VARCHAR(50)		

TABLE 3 – Table limites_classes

Nom de variable	Type	Clé primaire	Clé étrangère
id	SERIAL	oui	
classe	VARCHAR(10)		
type	VARCHAR(50)		
limite	INTEGER		

);

Table limites_classes (Table 3) liste les classes microbiologiques, les types de prélèvements et les limites microbiologiques. Elle a été créée avec la requête :

```
CREATE TABLE limites_classes( id SERIAL PRIMARY KEY,
classe VARCHAR(10) NOT NULL
type VARCHAR(50) NOT NULL
limite INTEGER NOT NULL
);
```

Table points_prelev (Table 4) liste les points de prélèvements utilisés. Elle a été créée avec la requête :

```
CREATE TABLE points_prelev(
id SERIAL PRIMARY KEY,
point VARCHAR(10) NOT NULL,
id_disp INTEGER NOT NULL REFERENCES disp_prelev(id)
id_class INTEGER NOT NULL REFERENCES limites_classes(id),
);
```

Table jours_prelev (Table 5) liste les jours de prélèvements. Elle a été créée avec la requête :

```
CREATE TABLE jours_prelev(
id SERIAL PRIMARY KEY,
jour VARCHAR(10) NOT NULL,
);
```

TABLE 4 – Table points_prelev

Nom de variable	Type	Clé primaire	Clé étrangère
id	SERIAL	oui	
point	VARCHAR(10)		
id_disp	INTEGER		disp_prelev(id)
id_class	INTEGER		limites_classes(id)

TABLE 5 – Table jours_prelev

Nom de variable	Type	Clé primaire	Clé étrangère
id	SERIAL	oui	
jours	VARCHAR(10)		

Table planning_prelev (Table 6) associe des jours de prélèvements et des points de prélèvements. Elle a été créée avec la requête :

```
CREATE TABLE planning_prelev(
  id_jour INTEGER NOT NULL REFERENCES jours_prelev(id),
  id_point INTEGER NOT NULL REFERENCES points_prelev(id),
  PRIMARY KEY (id_jour, id_point) );
```

Table prelevements (Table 7) liste les prélèvements réalisés. Elle a été créée avec la requête :

```
CREATE TABLE prelevements(
  id SERIAL PRIMARY KEY,
  date_prelev DATE NOT NULL,
  id_point INTEGER NOT NULL REFERENCES points_prelev(id) );
```

TABLE 6 – Table planning_prelev

Nom de variable	Type	Clé primaire	Clé étrangère
id_jour	INTEGER	oui	jours_prelev(id)
id_point	INTEGER	oui	points_prelev(id)

TABLE 7 – Table prelevements

Nom de variable	Type	Clé primaire	Clé étrangère
id	SERIAL	oui	
date_prelev	DATE		
id_point	INTEGER		points_prelev(id)

TABLE 8 – Table resultats

Nom de variable	Type	Clé primaire	Clé étrangère
id	SERIAL	oui	
date_prelev	DATE		
id_point	INTEGER		points_prelev(id)

Table resultats (Table 8) pour enregistrer les résultats des prélèvements. Elle a été créée avec la requête :

```
CREATE TABLE resultats(
id_prelev INTEGER REFERENCES prelevements(id),
ufc INTEGER NOT NULL,
germe VARCHAR(100),
type_rendu VARCHAR(10) NOT NULL );
```

3.1.3 Peuplement des tables

Les enregistrements des tables ont été renseignés par des commandes SQL *INSERT*, sauf pour les tables *prelevements* et *resultats*. Ces deux tables ont été renseignées par l'importation de fichiers .csv contenant les données déjà enregistrées sur tableur.

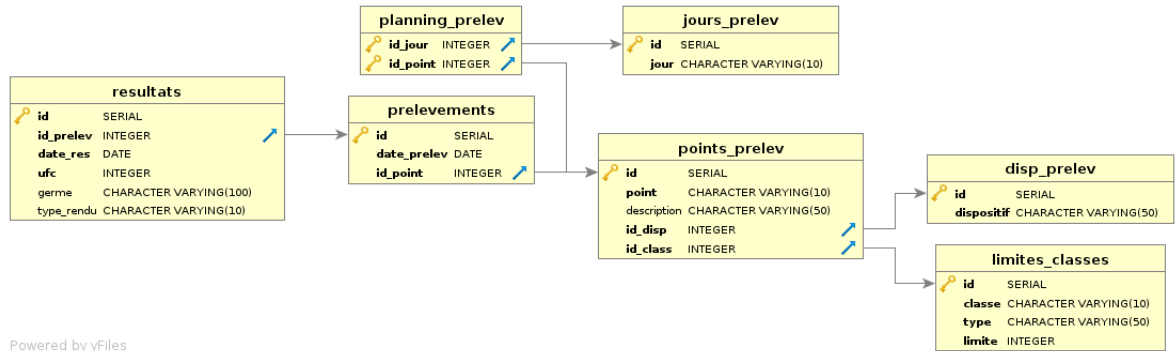
3.2 Serveur Web

Le serveur web Apache (version 2.4.7) est installé sur le système d'exploitation GNU/Linux (Linux Mint 17.3). Les pages Web sont écrites en langage HTML et PHP (version 7.2.4-1)

4 Résultats

Le dump de la base de données est fourni sous le nom de fichier : **bacterio_upnp.sql**. Toutes les requêtes sont faites avec l'utilisateur *pharmacien*

FIGURE 1 – Schéma relationnel de bacterio_upnp



et le mot de passe *zac*.

Le schéma relationnel de la base *bacterio_upnp* (figure 1) a été obtenu par DBVizualizer.

Les fichiers .php et .html du site web sont les suivants :

accueil.html : Présentation de la base de données et accès aux différentes pages par des liens hypertexte.

5 Discussion

Discussion

6 Conclusion

conclusion