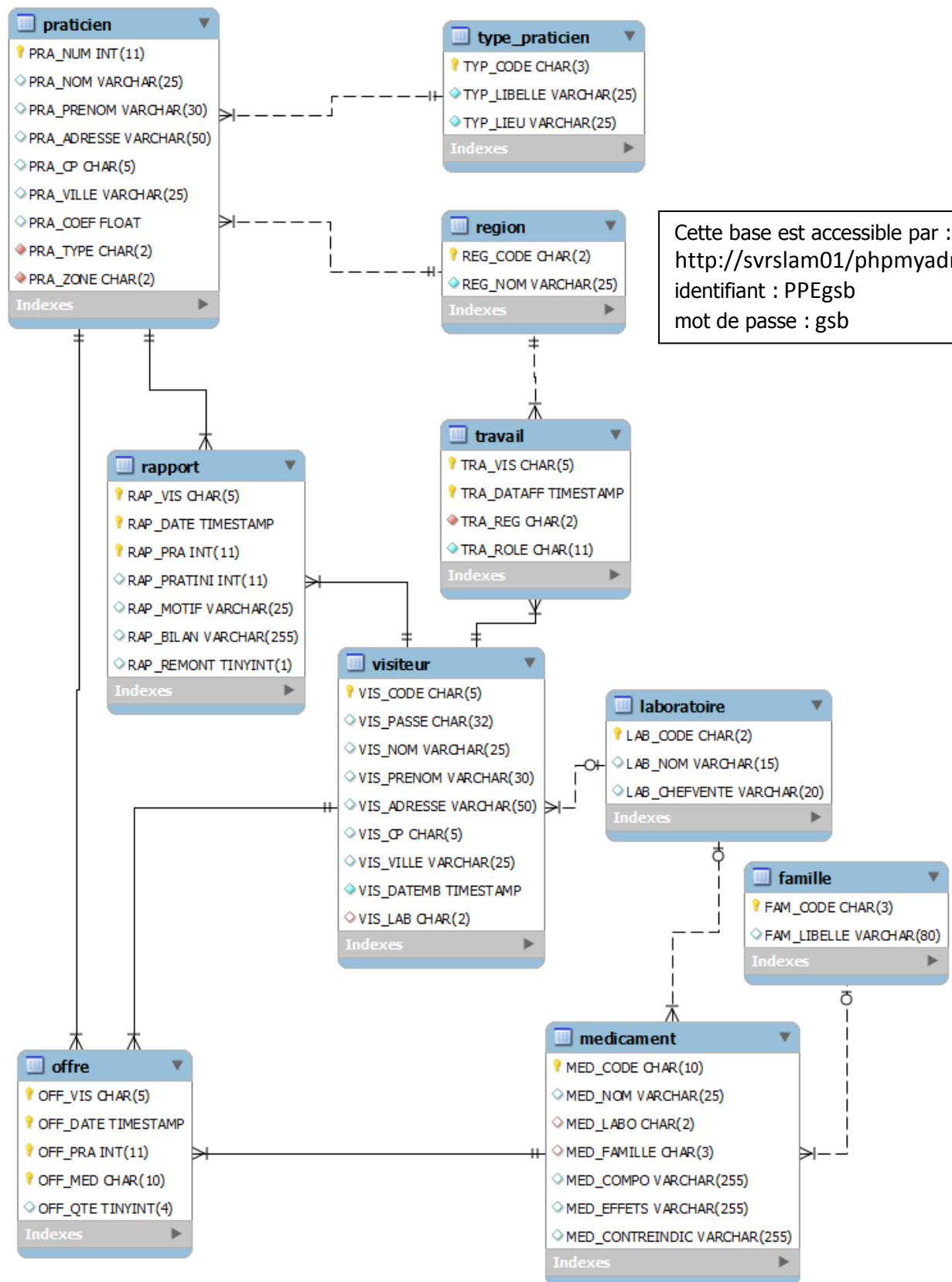


La base de données gsb



Le formulaire des médicaments

A ce stade du développement, l'application qui est rappelons-le, destinées à aider le visiteur à préparer ses visites ou à lui permettre de répondre à une question d'un praticien, permet la consultation des information sur les praticiens.

Vous allez maintenant développer une fonctionnalité de consultation des médicaments. Le fonctionnement souhaité est le suivant :

1. l'utilisateur sélectionne une famille de médicaments, puis clique sur Ok.
2. La liste des médicaments est alors actualisée avec les médicaments de la famille sélectionnée. L'utilisateur choisit un médicament, puis clique sur Ok.
3. La fiche du médicament sélectionné est alors affichée.

The application interface consists of a sidebar menu and a main content area. The sidebar menu includes 'Outils' with sub-items: 'Comptes-Rendus', 'Nouveaux', 'Consulter', 'Médicaments', 'Praticiens', 'Autres', and 'visiteurs'. The main content area is titled 'Pharmacopée' and contains a form for selecting a medication family and then a specific medication.

Screenshot 1 (localhost/GCR/index.php?action=50): The 'Famille' dropdown is set to 'Antibiotique de la famille des bêta-lactamines (pénicilline A)'. The 'Ok' button is highlighted with a red circle and the number 1.

Screenshot 2 (localhost/GCR/index.php?action=60): The 'Famille' dropdown remains the same, but the 'Médicament' dropdown is now set to 'ADIMOL'. The 'Ok' button is highlighted with a red circle and the number 2.

Screenshot 3 (localhost/GCR/index.php?action=70): The medication card for ADIMOL is displayed. It includes the following information:

- NOM COMMERCIAL : ADIMOL
- COMPOSITION : Amoxicilline + Acide clavulanique
- EFFETS : Ce médicament, plus puissant que les pénicillines simples, est utilisé pour traiter des infections bactériennes spécifiques.
- CONTRE INDIC. : Ce médicament est contre-indiqué en cas d'allergie aux pénicillines ou aux céphalosporines.
- LABORATOIRE : Swiss Kane

 The 'Ok' button for the medication selection is highlighted with a red circle and the number 3.

L'utilisateur peut ensuite sélectionner soit un autre médicament, soit une autre famille de médicaments. Dans ce dernier cas la liste des médicaments est actualisée.

I – Préparation

Commencez par changer l'extension du fichier `formMEDICAMENT.html` en `php`.

Modifiez tout le début du fichier de la même manière que pour `formPRATICIEN.php` (n'hésitez pas à utiliser ce fichier comme modèle). Au final :

- votre code ne doit plus comporter aucun attribut de présentation.
- Le `div "bas"` a un attribut `id` et non `name`.
- Pharmacopée s'écrit avec un "é".

Cette page (`formMEDICAMENT.php`) doit être appelée à partir du menu de l'application. En respectant l'architecture mise en place dans la phase de développement précédente (le site est doté d'un point d'entrée unique), modifiez en conséquence le menu. Testez ensuite le bon fonctionnement du menu.

II – Recherche d'un médicament

2 formulaires vont être utilisés. Le premier permettra la sélection d'une famille de médicaments, le second celle d'un médicament de la famille sélectionnée.

Chaque fois que sera sélectionnée une nouvelle famille de médicament, le contenu de la liste des médicaments devra être rafraîchi pour proposer uniquement les médicaments de la famille.

Donc, dans le cadre de la recherche d'un médicament, 3 demandes peuvent être adressées au point d'entrée unique de l'application (`index.php`) :

1. l'affichage du formulaire de sélection d'une famille de médicaments,
2. l'affichage du formulaire de sélection d'un médicament,
3. l'affichage de la fiche d'un médicament.

- La première demande doit entraîner l'affichage du seul formulaire de sélection d'une famille.
- La deuxième demande doit entraîner l'affichage du formulaire précédent avec la famille choisie sélectionnée par défaut, et du formulaire de sélection d'un médicament.
- La troisième demande doit entraîner l'affichage des 2 formulaires précédents (avec pour les médicaments celui qui a été choisi sélectionné par défaut), et de la fiche médicament correspondante.

Attention ! le deuxième formulaire devra garder la trace de la famille de médicament sélectionnée, afin que lorsqu'il sera soumis, il soit possible de réafficher le premier formulaire en sélectionnant par défaut la famille choisie, et de réafficher le second avec la liste des médicaments de la famille choisie.

Comment faire ? Il va falloir stocker dans le second formulaire le code de la famille de médicaments sélectionné dans le premier. Pour cela on utilisera un contrôle de saisie `<input>` de type `hidden` (caché). Rappelons que ce contrôle de saisie est invisible à l'utilisateur, mais que lorsque le formulaire est soumis sa valeur est transmise avec celle des autres contrôles de saisie du formulaire.

Commencez par créer 2 nouvelles fonctions dans votre bibliothèque :

- `formBoutonSubmit()` qui retourne le code HTML d'un bouton de type `submit` et qui prend en paramètre :
 - le nom du bouton, | - son id, | - sa valeur, | - son index de tabulation.
- `formInputHidden()` qui retourne le code HTML d'un contrôle de saisie de type `hidden` et qui prend en paramètre :
 - le nom du contrôle, | - son id, | - sa valeur.

Déterminez 3 valeurs numériques que peut prendre la variable `action` pour la recherche d'un médicament.

Ajoutez une structure `switch` à votre programme `formMEDICAMENT.php` (au besoin consultez la documentation de `php` sur `php.net`).

1^{ère} demande : afficher le formulaire de sélection d'une famille de médicaments

Le premier cas à traiter est la demande d'affichage du formulaire de sélection d'une famille de médicaments. Ajoutez à votre code le `case` correspondant.

Créez un premier formulaire nommé `formChoixFamilleMedicaments` comportant la liste déroulante permettant de sélectionner une famille de médicaments, et le bouton Ok.

Veillez à bien respecter l'architecture de l'application : créez une fonction pour récupérer les données nécessaires, utilisez la bibliothèque que vous avez créée ; utilisez `formPRATICIEN.php` pour modèle.

La soumission de ce formulaire doit entraîner la 2^{ème} demande ...

2^{ème} demande : afficher le formulaire de sélection d'un médicament

Le deuxième cas à traiter est la demande d'affichage du formulaire de sélection d'un médicament.

Il faut d'abord réafficher le premier formulaire avec la famille choisie sélectionnée.

Créez ensuite un deuxième formulaire nommé `formChoixMedicament` comportant la liste déroulante permettant de sélectionner un médicament (cette liste ne doit contenir que les médicaments de la famille sélectionnées dans la liste précédente), le contrôle de saisie caché contenant le code de la famille de médicament, et le bouton Ok.

La soumission de ce formulaire doit entraîner la 3^{ème} demande ...

3^{ème} demande : afficher la fiche d'un médicament

Le troisième cas à traiter est la demande d'affichage de la fiche du médicament sélectionné.

Il faut d'abord réafficher le premier formulaire avec la famille choisie sélectionnée. Puis le second avec le médicament choisi sélectionné.

Redéveloppez ensuite le formulaire nommé `formMEDICAMENT` que vous renommerez `formMedicament`.

Comme les zones COMPOSITION, EFFETS et CONTRE INDIC. affiche des informations textuelles de 255 caractères de long (`varchar(255)` dans la base de données), il est préférable de les afficher dans un contrôle de saisie de type `<textarea>`. Ajoutez à votre bibliothèque une fonction nommée `formTextArea()` qui retourne le code HTML d'un contrôle de saisie de type `<textarea>` et qui prend en paramètre :

- son label,	- sa valeur,	- sa longueur maximale,
- son nom,	- sa largeur, exprimée en nombre de	- son index de tabulation,
- son id,	colonnes.	- un booléen indiquant si la zone
	- sa hauteur exprimée en nombre de lignes.	est en lecture seule.

Si besoin voir sur Internet la documentation de la balise `<textarea>`.

Testez votre code.

III – Finaliser la feuille de style

Le formulaire médicament doit être un peu éloignée du formulaire qui la précède. On joue sur le *padding* :

```
#formMédicament {
    padding-top: 10px;
}
```

Le *padding* gauche du formulaire médicament est le même que celui du formulaire praticien :

```
#formPraticien, #formMédicament {
    padding-left: 20px;
}
```

Le style des `<label>` du formulaire médicament est le même que celui des `<label>` du formulaire praticien :

```
#formPraticien label, #formMédicament label {
    display: inline-block;
    width: 20%;
}
```

La marge haute des `<input>` et des `<textarea>` du formulaire médicament est la même que celle des `<input>` du formulaire praticien :

```
#formPraticien input, #formMédicament input, #formMédicament textarea {
    margin-top: 10px;
}
```

Les `<label>` des `<textarea>` sont alignés sur le bas des `<textarea>` ce qui est laid. On les aligne sur le milieu :

```
label, textarea { /* textarea comme input est par défaut en inline-block. */
    display: inline-block;
    vertical-align: middle;
}
```

Remarque

Les zones de saisie de type `text` et les `textarea` ont une largeur différente bien que l'on ait indiqué `<input type="text" size="50" />` et `<textarea cols="50">`. Si on regarde de plus près, la police de caractère utilisée dans les 2 contrôles de saisie diffère (proportionnelle pour le `input`, non proportionnelle pour le `textarea`).

IV - Fignolez votre code

Corrigez `formPRATICIEN.php` pour faire appel à votre fonction `formInputSubmit()`.