Projet GSB FRAIS (Suite)

Présentation

Contexte

Dans l'industrie pharmaceutique, le laboratoire Galaxy Swiss Bourdin(GSB) est issus d'une fusion de deux énormes leaders de ce secteur qui ont pour but d'unirent leur force pour créer un seul leader sur ce marché

Objectif

Comprendre et développer l'Application Appli-Frais.

Documentation utilisée:

- APPLI-FRAIS.doc
- ArchitectureApplicativeMVC.doc Travail_a_faireMVC.docx
- Normes_de_developpement.doc
-

Développeurs de l'Application

- Stéphane Benatout
- Quentin Gimié D'Arnaud

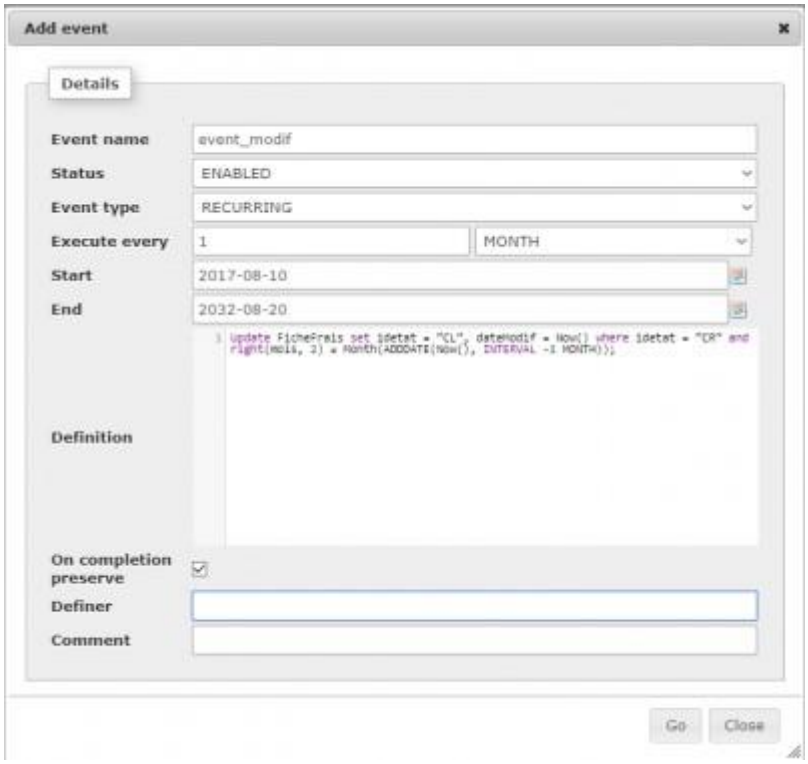
I\ Compte-Rendu du PPE3.2 : Application Frais

Semaine 1 :

Cette semaine, nous avons lu la documentation de la deuxième partie et réfléchi sur la requête SQL. Nous avons réfléchi sur le fait qu'une fiche de frais doive se mettre à jour tous les mois. Il fallait trouver le moyen de faire en sorte que la requête soit exécutée tous les mois, même l'or d'un changement d'année. Après Nous l'avons faite valider auprès des professeurs.
Update FicheFrais set idetat = “CL”, dateModif = Now() where idetat = “CR” and right(mois, 2) = Month(ADDDATE(Now(), INTERVAL -1 MONTH));

Semaine 2 :

Cette semaine, nous avons fait les tests de la requête, et cherché des solutions pour éviter les erreurs lorsque l'on arrive au mois de janvier. Nous en avons trouvé 2 autres, une avec la fonction sql DATE_SUB() et en changeant la date en millisecondes.
Puis on a créé, l'évent dans la partie dans déclencheur de phpMyAdmin.



Cela, nous donne ce résultat :

Dans la table visiteur, on a créé un nouveau champ nommé “Statut” et on a attribué grâce à deux indices différents (C pour comptable et V pour Visiteur) dans le but de différencier les visiteurs par leur fonction.

Semaine 3 :

Nous commencerons la semaine part la réussite de la connexion d'un comptable.

Puis dans la partie visiteur, on a rectifié les soucis d'accolade pour afficher les frais occasionnés durant un mois donné. De plus, nous avons commencé à développer la partie comptable. Nous commençons à créer la listeVisiteur.

Semaine 4 :

Nous avons commencé par corriger les erreurs d'affichage sur la page gererListeFiche. Quand cela fut fait, nous avons commencé essayer de coder la page.

Semaine 5 :

Nous avons finis de corriger les erreurs apparentes sur le projet.
Documentation Technique

II\ Documentation Technique

Dans cette partie, nous allons voir l'aspect technique de l'application GSB.

La reconnexion :

Dans le but de différentier un visiteur lambda à un comptable

Le contrôleur *valideConnexion* permet de valider la connexion au visiteur grâce aux formulaires d'authentification .Cependant le test en fin case permet par le statut de différencier les visiteurs lambda et les comptables

Les fonctions :

Les fonctions commentées sont traité par le contrôleur *c_gérerFraitComptable.php*

Les contrôleurs développés et leurs explications :

Au cours de ce PPE, nous avons développé les constructeurs:

1. c_gérerFraitComptable
2. c_validFraibis

Le controleur : c_gererFraisComptable

Ce constructeur va permettre de gérer les différentes actions du comptable.

Explication du code

Dans le case nommé ***"gererListeFiche"***, nous gérons le code qui permet d'afficher les listes déroulantes et le moyens d'afficher les fiches d'un visiteur lambda pour un mois donnée.

Nous affectons à la variable *$lesMois* tous les mois d'une année. Puis, nous faisons une condition ternaire dans l'objectif de d'afficher dans les tous cas la valeur "Janvier" dans le cas où il n'y a pas de mois comportant de création de fiche. Ensuite, dans une variable en parallèle nommée "$lesVisiteur" qui retourne grâce les visiteurs selon le mois donné en paramètre. De plus, nous listons tous les mois et les visiteurs partant du début de la table pour ensuite afficher dans listes déroulantes correspondante. Nous appelons ensuite la vue v_listesdesVisiteurs.php.

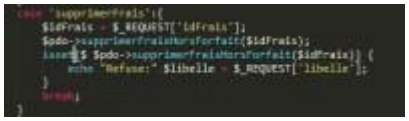
Nous déclarons trois variables: * $lesinfos qui récupère les informations des fiches frais selon le mois et le visiteur.

$lesinfoshf qui récupère les informations des fiches hors-forfait selon le mois et le visiteur. $lesinfosfiche qui récupère les informations des fiches frais selon le mois et le visiteur.

Enfin, nous affichons la vue v_etatFraisComptable.php pour afficher les données du visiteur dans un mois sélectionné de ces fiches de

fraits.

case **//____"supprimerFrais"____//**



Cette case permet de supprimer une fiche d'un visiteur par l'intermédiaire de l'identifiant de celle-ci et grâce à la fonction $supprimerFraisHorsForfait retournée par nos soins et récupérer par la variable super globale $request lors de la création de la fiche. Toutefois, une fois supprimée la fiche, nous affichons au libellé de la fiche "Refusé:…."pour informer le visiteur lambda de la décision du comptable.

Case : gerermodification

Nous récupérons grâce à la variable $requiest les données suivantes (la date, le libellé et le montant) lors de la création d'une fiche ; puis nous modifions ces données avec l'aide de la fonction updatefichefrais($idVisiteur,$mois) d'un visiteur et d'un mois donné. Enfin, nous n'oublions pas de vérifier les erreurs grâce à la méthode ejouterErreur()qui affiche si besoin la vue d'erreur.

Le contrôleur : c_validFraisbis.php :

Ce contrôleur permet de valider les fiches qui n'ont pas été encore validé par le comptable.

```php
<?php
include("vues/v_sommaire_comptable.php");
$idVisiteur = $_SESSION['idVisiteur'];
$mois = getMois(date("d/m/Y"));
$numAnnee = substr( $mois,0,4);
$numMois = substr( $mois,4,2);
$action = $_REQUEST['action'];
$validation = 0;

switch($action){
    case 'IdentifierUtilisateur':{
        break;
    }
    case 'afficheFrais':{
        $validation = 1;
        $_SESSION['idVisiteur'] = $_GET['idVisiteur'];
        $_SESSION['mois'] = $_GET['mois'];
        break;
    }
    case 'saisirFrais':{
        if($pdo->estPremierFraisMois($idVisiteur,$mois)){
            $pdo->creeNouvellesLignesFrais($idVisiteur,$mois);
        }
        break;
    }
    case 'validerMajFraisForfait':{
        $lesFrais = $_REQUEST['lesFrais'];
        if(lesQteFraisValides($lesFrais)){
            $pdo->majFraisForfait($idVisiteur,$mois,$lesFrais);
        }
        else{
            ajouterErreur("Les valeurs des frais doivent être numériques");
            include("vues/v_erreurs.php");
        }
    break;
    }
    case 'validerCreationFrais':{
        $dateFrais = $_REQUEST['dateFrais'];
        $libelle = $_REQUEST['libelle'];
        $montant = $_REQUEST['montant'];
        valideInfosFrais($dateFrais,$libelle,$montant);
        if (nbErreurs() != 0 ){
            include("vues/v_erreurs.php");
        }
        else{
```

```php
    case 'validerCreationFrais':{
        $dateFrais = $_REQUEST['dateFrais'];
        $libelle = $_REQUEST['libelle'];
        $montant = $_REQUEST['montant'];
        valideInfosFrais($dateFrais,$libelle,$montant);
        if (nbErreurs() != 0 ){
            include("vues/v_erreurs.php");
        }
        else{
            $pdo->creeNouveauFraisHorsForfait($idVisiteur,$mois,$libelle,$dateFrais,$montant);
        }
        break;
```

Les vues :

Nous avons développées deux vues:

1. v_listesdesVisiteurs.php
2. v_etatFrais Comptable.php

Vue : v_listedesVisiteurs.php :

Cette vue est composée de deux listes déroulantes. Une liste déroulante pour les visiteurs et l'autre liste déroulante pour les mois de l'année.

```php
<div id="contenu">
    <h2>Mes fiches de frais</h2>
    <h3>Mois à sélectionner : </h3>
    <form action="index.php?uc=gererFraisCompta&action=gererListeFiche" method="post">
        <div class="corpsform">
            <p>
                <label for="lstMois" accesskey="n">Mois : </label>
                <select id="lstMois" name="lstMois" type="submit" />
                    <?php
                    foreach ($lesMois as $unMois){
                        $mois = $unMois['mois'];
                        $moisSuivant = substr($unMois['mois'], 4, 2) ;
                        $anneeSuivante = substr($unMois['mois'], 0, 4) ;
                        echo "<option value=".($mois == $moisSelectionne)?'selected':'') ."value='". $mois ."'". $mois ."/". $anneeSuivante ."</o
                            option>";
                    }
                    ?>
                </p>
            </div>
        </div>
        <div class="piedform">
            <p>
                <input id="ok" type="submit" name="ok" value="Valider" title="OK" />
                <input id="annuler" type="reset" value="Effacer" size="OK" />
            </p>
        </div>
    </form>
    <form action="index.php?uc=gererFraisCompta&action=gererListeFiche" method="post">
        <div class="corpsform">
            <p>
                <label for="lstVisiteur" accesskey="v">Liste des Visiteurs:</label>
                <select id="lstVisiteur" name="lstVisiteur">
                    <?php
                    foreach ($lesVisiteurs as $unVisiteur) {
                        echo "<option ".($idSelectVisiteur == $unVisiteur['idVisiteur'])?'selected':'') "value=" $unVisiteur['
                        idVisiteur'] ">". $unVisiteur['nom'] " " $unVisiteur['prenom'] "</option>";
                    }
                    ?>
                </p>
            </div>
        </div>
```

Première liste déroulante : liste des mois :

Dans cette liste déroulante, nous parcourons tous la table des mois et à chaque itération est transmis à la variable $unMois. Nous attribuions les valeurs de la variable $unMois à la variable $val.
La variable $mois reçoit le fragment de chaine grâce à la fonction substr(),et correspond au positionnement du mois que l'on veut afficher par rapport à date donnée (mm/yy).
$numAnnée est la variable qui récupère sous forme de fragment ($substr() l'année de la date passée dans cette liste déroulante.
Dans la balise html option, nous affichons grâce la date sous forme de concaténation des deux variables précédente. Dans un premier, nous faisons l'égalité entre la variable $moisSelectionner déclarée dans le contrôleur avec la variable $mois dans le but d'affiché la même valeur trouver dans le contrôleur.
De plus, après avoir sélectionné une date et valider avec un bouton de type submit, on passe avec la méthode POST l'identifiant du mois pour le ré exploiter au niveau du contrôleur.

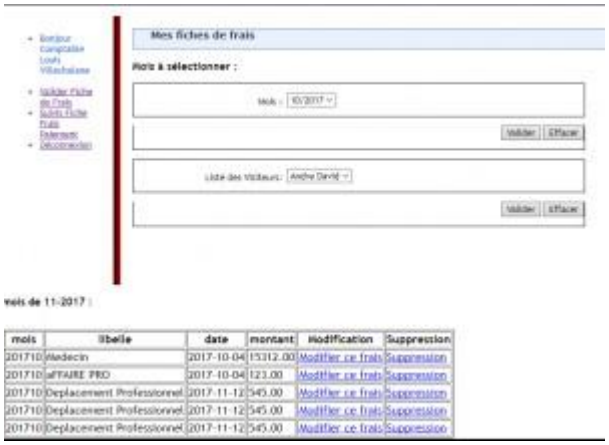
Deuxième partie déroulante : liste des Visiteurs :

Dans cette déroulante, nous parcourons tous la table des visiteurs grâce à un autre foreach et ainsi nous donnons à $unVisiteur toutes les données transmises par itération de la variable $lesVisiteurs.
Dans plus, nous faisons l'égalité entre la variable $SelectVisiteur déclarée dans le contrôleur avec la variable $unVisiteur dans le but d'affiché la même valeur trouver dans le contrôleur. Ensuite, le echo de la balise html option va afficher le nom et le prénom du visiteur concerné grâce l'identifiant.

Vue : v_etatFraisComptable.php

Cette vue permet d'afficher les fiches horsforfaits sous forme de tableau selon le mois et le visiteur.

Donc nous créons une variable $fraisfhf qui comme valeur les fiches horsforfait par l'appel de la fonction getfichehorsforfait() avec pour paramètre la variable $SelectVisitor égale l'identifiant d'un visiteur et la variable $moisASelectionner égale au mois sélectionné.



Vue : v_listeUtilisateurCL.php

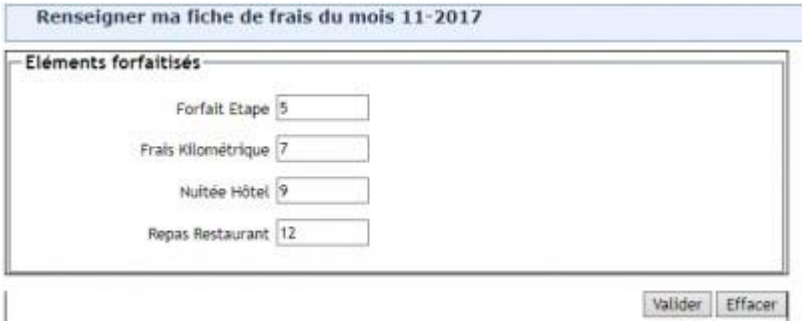
Cette vue permet d'afficher tous les visiteurs lambda qui ont des fiches non validées par le comptable.

III\ Documentation Utilisateur

Dans ce conte rendu, nous allons vous montrez comment l'application GSB fonctionne

Partie 1 : Visiteur

Le visiteur lambda se connecte sur son compte pour pouvoir gérer sa fiche de fraits. Dans un premier temps, le visiteur créer une fiche frais.



Partie 2 : Comptable

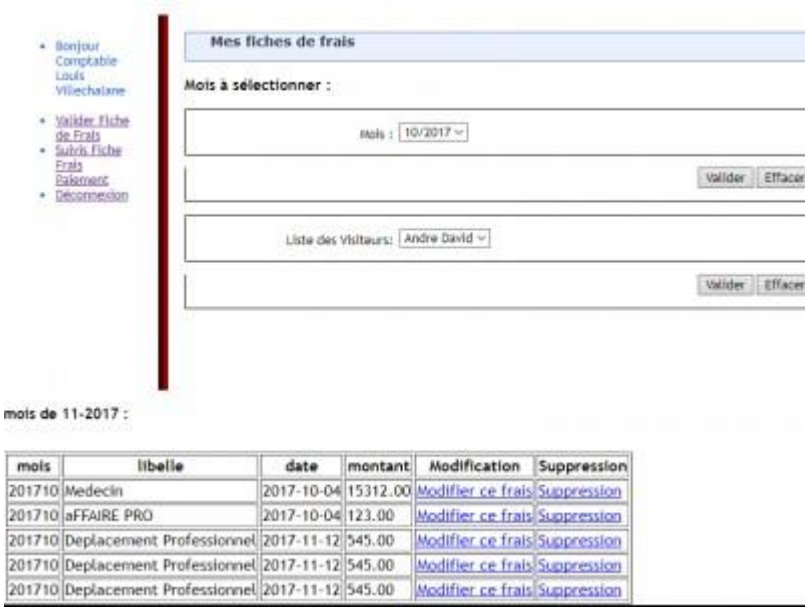
Dans cette partie, nous allons voir la partie de gestion des fiches des visiteurs lambda.

Après que le comptable soit connecté,

La Validation des fiches fraits

Il va commencer à se mettre au travail.

Donc il va aller directement valider les fiches en attente des visiteurs lambda selon le mois; par le moyen de liste déroulante.



mois	libelle	date	montant	Modification	Suppression
201710	Medecin	2017-10-04	15312.00	Modifier ce frais	Suppression
201710	aFFAIRE PRO	2017-10-04	123.00	Modifier ce frais	Suppression
201710	Deplacement Professionnel	2017-11-12	545.00	Modifier ce frais	Suppression
201710	Deplacement Professionnel	2017-11-12	545.00	Modifier ce frais	Suppression
201710	Deplacement Professionnel	2017-11-12	545.00	Modifier ce frais	Suppression

Grâce à cette vue et la rubrique modification, le comptable peux modifier les données des fiches :

La dernière partie correspond à la suppression :

après avoir refuser les fraits et laisser comme libelé “Refusée:…”