Yohann Lemonnier

Vincent Audebert

Romain Azemat

Jean Osscini

Rapport de projet

Application GSB Visiteurs : Application d’enregistrement et de suivi des comptes-rendus de visites

SOMMAIRE

[Introduction 3](#_Toc477953984)

[Définition du besoin et prise de connaissance de l’existant 4](#_Toc477953985)

[Cas d’utilisation 5](#_Toc477953986)

[Rapport de tests fonctionnels 7](#_Toc477953987)

[MCD 9](#_Toc477953988)

[Schéma Entité-Associations 9](#_Toc477953989)

[Diagramme de Gant prévisionnel 10](#_Toc477953990)

[Maquettage prévisionnel 12](#_Toc477953991)

Introduction : La société GSB utilise une application de gestion des comptes-rendus des visites commerciales auprès des différents médecins et autres organisations médicale démarchées. Cependant, ladite application fonctionne pour l’instant via le logiciel Access, un certain nombre d’erreurs étant à corriger. L’idée est donc de développer une application bureau, sous NetBeans utilisant une base de données Oracle. L’application doit remplir les mêmes fonctionnalités que l’original.

# Définition du besoin et prise de connaissance de l’existant

* Définition du besoin :

Un projet d’application Web est en cours pour rendre accessible la gestion à partir de n’importe quel ordinateur. L’accès à cette gestion doit être maintenu à l’aide d’une application bureau.

L’application doit être développée en JAVA et la base de donnée portée sur Oracle.

* Fonctionnalités :

L’application remplira les mêmes fonctionnalités que l’existant

* Contraintes :

L’application sera facilement accessible, respectera le modèle MVC, l’ergonomie sera soignée, les noms de classes respecteront les normes

* Methode de travail

Le projet respectera la méthode de développement agile « scrum ».

Des tests unitaires seront présents pour le développement de chaque classe du projet.

Des tests fonctionnels seront effectués préalablement au développement de l’application.

# Cas d’utilisation

Use case 1 :

Cas d’utilisation : ajouter un rapport de visite

|  |  |
| --- | --- |
| Acteur | Système étudié |
| 1-L’utilisateur se connecte |  |
|  | 2- l’application affiche 5 boutons :  Comptes-rendus, visiteurs, praticiens, médicaments et quitter |
| 3- l’utilisateur sélectionne le bouton Comptes-rendus |  |
| 4- l’utilisateur clique sur nouveau et remplie les champs du formulaire |  |
|  | 5-l’application vérifie les données du formulaire de saisie d’un compte-rendu |
|  | 6-le nouveau compte rendues est créé |
|  |  |

Use case 2 :

Cas d’utilisation : visionner les visiteurs

|  |  |
| --- | --- |
| Acteur | Système étudié |
| 1-L’utilisateur se connecte |  |
|  | 2- l’application affiche 5 boutons :  Comptes-rendus, visiteurs, praticiens, médicaments et quitter |
| 3- l’utilisateur sélectionne le bouton visiteurs |  |
|  |  |

Use case 3 :

Cas d’utilisation : visionner les praticiens

|  |  |
| --- | --- |
| Acteur | Système étudié |
| 1-L’utilisateur se connecte |  |
|  | 2- l’application affiche 5 boutons :  Comptes-rendus, visiteurs, praticiens, médicaments et quitter |
| 3- l’utilisateur sélectionne le bouton praticiens |  |
|  | 4-L’application affiche un formulaire correspondant à la page praticiens |

Use case 4:

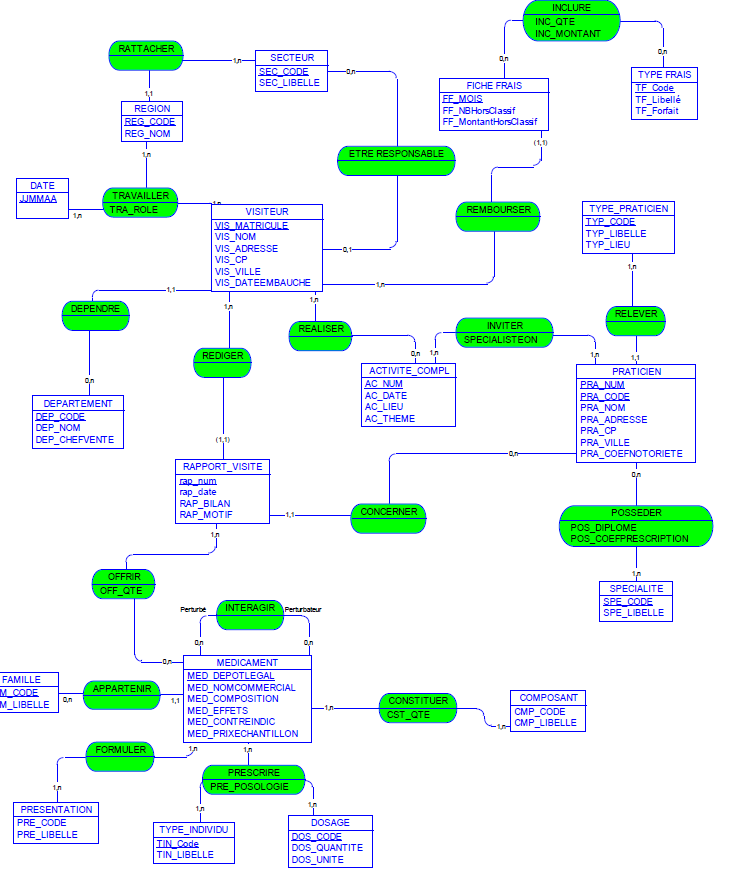
Cas d’utilisation : visionner les médicaments

|  |  |
| --- | --- |
| Acteur | Système étudié |
| 1-L’utilisateur se connecte |  |
|  | 2- l’application affiche 5 boutons :  Comptes-rendus, visiteurs, praticiens, médicaments et quitter |
| 3- l’utilisateur sélectionne le bouton médicaments |  |
|  | 4-L’application affiche un formulaire correspondant à la page médicaments |

# Rapport de tests fonctionnels

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ordre | Scénario | Cas de test : valeurs | Cas de test : résultats attendus | Test : résultat obtenus | Test : diagnostic |
| 1 | Use case1- ajouter un rapport de visite | Praticien : Cauchy Denis  Date rapport : 09/03/2017  Motif visite : ddd  Bilan : ddd | Messages : « le rapport de visite a bien été ajouté »  Le rapport de visite a été ajouté la base de données | Pas de message de validation | Résultat non conforme  Pas de message de validation |
| 2  2 | Use case2- visionner les visiteurs | Nom cacheux  Prenom: bernard | Dans la page visiteurs il est possible de faire une recherche pour un visiteur  La recherche s’effectue sans devoir cliquer sur le bouton « ok » | Affichage de la recherche sans cliquer sur le bouton « ok » | Résultat non conforme  On doit cliquer sur le bouton « ok » pour effectuer la recherche |
| 3 | Use case1-ajouter un rapport de visite | Praticien : Leveneur Hugues  Date Rapport : aaaa | Dans la page rapports de visite il est possible de renseigner une date.  Si le champ n’ai pas bien renseigné, afficher précisément les erreurs et replacer le curseur dans le champs de saisie | Un message d’erreur apparait qui nous indique une erreur dans pour le champ « date rapport » | Conforme |
| 4 | Connexion a l’application | Identifiant : aaa  Mot de passe : aaa | La connexion a l’application se fait avec un identifiant et un mot de passe  Si erreur de l’un deux, affiche un message d’erreur | Pas de message d’erreur lorsque l’identifiant et le mot de passe sont incorrect | Non conforme  Pas de message d’erreur en cas de mauvais identifiant et mot de passe |
| 5 | Connexion a l’application | Identifiant : aaa  Mot de passe : 18-jun-2003 | La connexion a l’application se fait avec un identifiant et un mot de passe  Si erreur de l’un deux, affiche un message d’erreur | Message d’erreur lorsque le mot de passe est incorrect | Conforme |
| 6 | Connexion a l’application | Identifiant : swiss  Mot de passe : aaa | La connexion a l’application se fait avec un identifiant et un mot de passe  Si erreur de l’un deux, affiche un message d’erreur | Pas de message d’erreur si identifiant ok  Et mot de passe incorrect | Non conforme |
| 7 | Connexion a l’application | Identifiant : swiss  Mot de passe : 18-jun-2003 | La connexion a l’application se fait avec un identifiant et un mot de passe  Si les deux sont correcte -> connexion a l’application | Connexion a l’application | Conforme |
| 8 | Bouton précédent | Bouton précédent page « rapport de visite » | Afficher le rapport de visite précédent | Affichage du rapport de visite précédent | Conforme |
| 9 | Bouton suivant | Bouton suivant page « rapport de visite » | Afficher le rapport de visite suivant | Affichage du rapport de visite suivant | Conforme |
| 10 | Bouton précédent | Bouton précédent page « visiteurs » | Afficher le visiteur précédent | Affichage du visiteur précédent | Conforme |
| 11 | Bouton suivant | Bouton suivant page « visiteurs » | Afficher le visiteur suivant | Affichage du visiteur suivant | Conforme |
| 12 | Bouton précédent | Bouton précédent page « praticiens » | Afficher le praticien précédent | Affichage du praticien précédent | Conforme |
| 13 | Bouton suivant | Bouton suivant page « praticiens » | Afficher le praticien suivant | Affichage le praticien suivant | Conforme |
| 14 | Bouton précédent | Bouton précédent page « médicaments » | Afficher le médicament précédent | Affichage du médicament précédent | Conforme |
| 15 | Bouton suivant | Bouton suivant page « médicaments » | Afficher le médicament suivant | Affichage du médicament suivant | Conforme |
| 16 | Liste déroulante lieux d’exercice praticiens | Valeur fixe | Affiche la liste des métiers | Menu déroulant inutilisable | Non conforme  Liste déroulante inutile, suppression nécessaire afin d’afficher simplement le métier correspondant |

# MCD



# Schéma Entité-Associations

Type Frais (TF\_Code, TF\_Libellé, TF\_Forfait)

INCLURE (TF\_Code#, FF\_MOIS#, INC\_QTE, INC\_MONTANT)

Fiche Frais (FF\_MOIS, VIS\_Matricule#, FF\_NBHorsClassif, FF\_MontantHorsClassif)

Visiteur (VIS\_Matricule, DEP\_Code#, SEC\_Code#, VIS\_Nom, VIS\_Adresse, VIS\_Cp, VIS\_Ville, VIS\_DateEmbauche)

Secteur(SEC\_Code, SEC\_Libelle)

Region(REG\_Code, SEC\_Code#, REG\_Nom)

TRAVAILLER(REG\_Code#, JJMMAA#, VIS\_Matricule#, TRA\_Role)

Date(JJMMAA)

Departement(DEP\_Code, DEP\_Nom, DEP\_ChefVente)

REALISER(VIS\_Matricule#, AC\_Num#)

Activite\_COMPL(AC\_Num, AC\_Date, AC\_Lieu, AC\_Theme)

INVITER(AC\_Num#, PRA\_Num#,PRA\_Code#, Specialisteon)

Praticien(PRA\_Num, PRA\_Code, TYP\_Code#, PRA\_Nom, PRA\_Adresse, PRA\_Cp, PRA\_Ville, PRA\_CoefNotoriete)

Type\_Praticien(TYP\_Code, TYP\_Libelle, TYP\_Lieu)

POSSEDER(PRA\_Num#, SPE\_Code#, POS\_Diplome, POS\_CoefPrescription)

Specialite(SPE\_Code, SPE\_Libelle)

Rapport\_Visite(RAP\_Num, VIS\_Matricule#, RAP\_Date, RAP\_Bilan, RAP\_Motif)

OFFRIR(RAP\_Num#, MED\_DepotLegal#, OFF\_Qte)

Medicament(MED\_DepotLegal, FAM\_Code#, MED\_NomCommercial, MED\_Composition, MED\_Effets, MED\_Contreindic, MED\_PrixEchantillon)

Famille(FAM\_Code, FAM\_Libelle)

FORMULER(MED\_DepotLegal#, PRE\_Code#)

Presentation(PRE\_Code, PRE\_Libelle)

PRESCRIRE(MED\_DepotLegal#, TIN\_Code#, DOS\_Code#, PRE\_Posologie)

Type\_Individu(TIN\_Code, TIN\_Libelle)

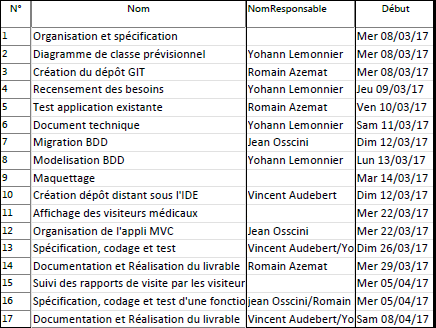
Dosage(DOS\_Code, DOS\_Quantite, DOS\_Unite)

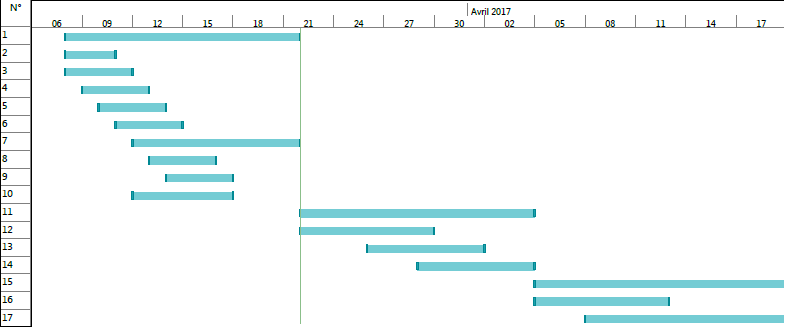
CONSTITUER(MED\_DepotLegal#, CMP\_Code#, CST\_QTE)

Composant(CMP\_Code, CMP\_Libelle)

INTERAGIR(MED\_DepotLegal\_Perturbe#, MED\_DepotLegal\_Perturbateur#)

# Diagramme de Gant prévisionnel





# Maquettage prévisionnel