Contents

[I. Présentation 2](#_Toc72359546)

[II. Conception de la Base de Données 3](#_Toc72359547)

[A. Modèle Conceptuel de Données (MCD) 3](#_Toc72359550)

[B. Modèle relationnel 3](#_Toc72359551)

[III. Services Web 4](#_Toc72359552)

# Présentation

Le laboratoire Galaxy Swiss Bourdin (GSB) provient de la fusion de deux entreprises qui unis le géant américain Galaxy (spécialisé dans le secteur des maladies virales dont le SIDA et les hépatites) et le conglomérat européen Swiss Bourdin (travaillant sur des médicaments plus conventionnels), dont lui-même comprenais déjà trois petits laboratoires.

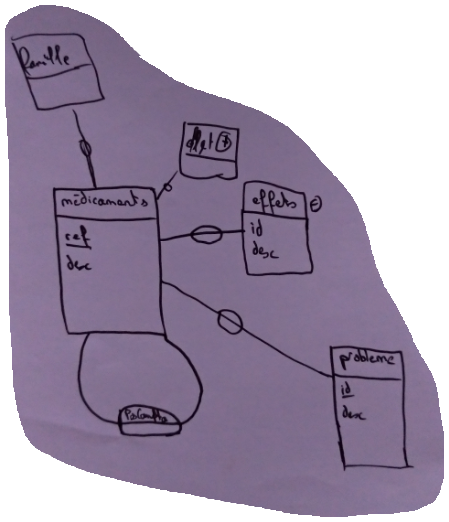
Le projet consiste à réaliser une application en PHP avec une structure MVC, des services web écrits en C# ou des méthodes écrites directement en PHP.

# Conception de la Base de Données



## Modèle Conceptuel de Données (MCD)

Le modèle conceptuel de données est un outil permettant la visualisation globale des relations entre les différentes tables de la base de données.



## Modèle relationnel

Le Modèle relationnel permet de voir, sous la forme détaillée, le contenu exact des tables de la base de données.

Categories (code\_categorie, nom\_categorie, description\_categorie)

Compatibilite (id\_compatibilite, description\_compatibilite, #code\_medicament1, #code\_medicament2, compatibilite)

Concerner (#code\_medicament, code\_effet)

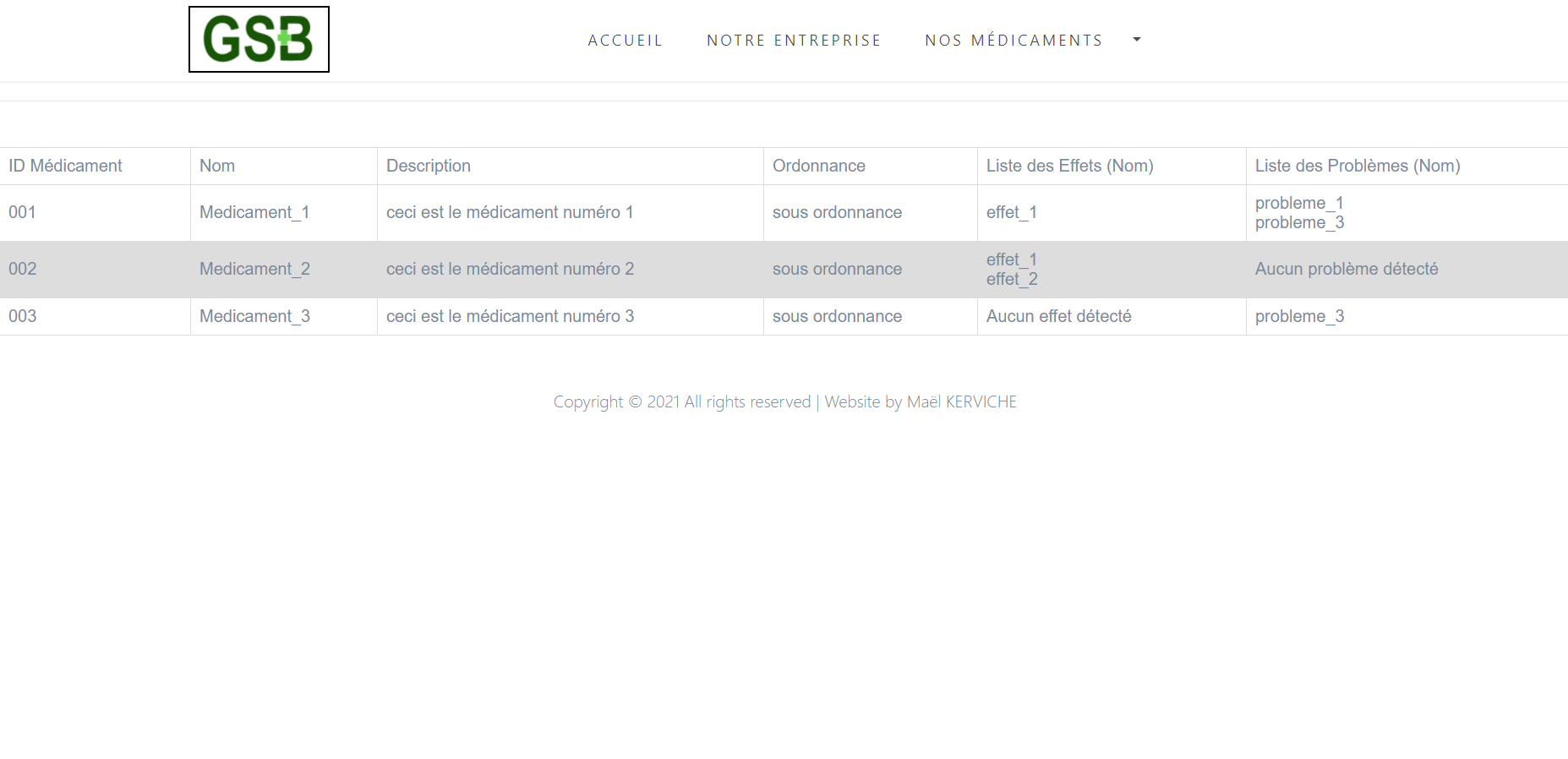
Effets (id\_effet, nom\_effet, description\_effet)

Medicaments (id\_medicament, nom\_medicament, description\_medicament, #code\_categorie, ordonnance\_medicament)

Problemes (code\_probleme, nom\_probleme, description\_probleme)

Soigner (#code\_medicament, #code\_probleme)

# Affichage medicaments



# Services Web (C#)

|  |
| --- |
| //liste des méthodes concernant la table MEDICAMENTS  /// <summary>  /// Get liste des id de la table Medicaments  /// </summary>  /// <returns>le tableau des ids</returns>  [WebMethod]  public String[] GetListMedicaments()  {    // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  // Creation Commande Select All Medicaments  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "SELECT \* FROM medicaments";  // Initialisation liste de String : id et nom des médicaments  var listeMedicaments = new List<String>();  MySqlDataReader datareader = get\_M.ExecuteReader();  while (datareader.Read())  {  listeMedicaments.Add(Convert.ToString(datareader["id\_medicament"]));  }  // Incrémentation tableau pour return  int size = listeMedicaments.Count;  string[] test = new String[size];  int i = 0;  foreach(String medicament in listeMedicaments)  {  test[i] = medicament ;  i++ ;  }  // return tableau id/nom des médicaments  return test;    } |
| /// <summary>  /// get Nom d'un medicament depuis son id  /// </summary>  /// <param name="id">l'id du medicament</param>  /// <returns>le nom du medicament</returns>  [WebMethod]  public String GetNomMedicament(String id)  {  // Connexion Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT nom\_medicament FROM medicaments WHERE id\_medicament = '" + id + "'";  return Convert.ToString(get\_M.ExecuteScalar());    } |
| /// <summary>  /// get Nom d'un medicament depuis son id  /// </summary>  /// <param name="id">l'id du medicament</param>  /// <returns>le nom du medicament</returns>  [WebMethod]  public String[] GetListProblemesMedicament(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  // Creation Commande Select All Medicaments  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "select nom\_probleme " +  "from problemes " +  "join soigner on problemes.code\_probleme = soigner.code\_probleme " +  "join medicaments on medicaments.id\_medicament = soigner.code\_medicament " +  "where id\_medicament = " + id;  // Initialisation liste de String : id et nom des médicaments  var listeProblemes = new List<String>();  MySqlDataReader datareader = get\_M.ExecuteReader();  while (datareader.Read())  {  listeProblemes.Add(Convert.ToString(datareader["nom\_probleme"]));  }  // Incrémentation tableau pour return  int size = listeProblemes.Count;  string[] test = new String[size];  int i = 0;  foreach (String probleme in listeProblemes)  {  test[i] = probleme;  i++;  }  if (size == 0)  {  string[] test2 = new string[1];  test2[0] = "Aucun problème détecté";  test = test2;  }  // return tableau id/nom des médicaments  return test;  } |
| /// <summary>  /// get Nom d'un medicament depuis son id  /// </summary>  /// <param name="id">l'id du medicament</param>  /// <returns>le nom du medicament</returns>  [WebMethod]  public String[] GetListEffetsMedicament(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  // Creation Commande Select All Medicaments  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "select nom\_effet " +  "from effets " +  "join concerner on effets.id\_effet = concerner.code\_effet " +  "join medicaments on medicaments.id\_medicament = concerner.code\_medicament " +  "where id\_medicament = " + id;  // Initialisation liste de String : id et nom des médicaments  var listeEffets = new List<String>();  MySqlDataReader datareader = get\_M.ExecuteReader();  while (datareader.Read())  {  listeEffets.Add(Convert.ToString(datareader["nom\_effet"]));  }  // Incrémentation tableau pour return  int size = listeEffets.Count;  string[] test = new String[size];  int i = 0;  foreach (String effet in listeEffets)  {  test[i] = effet;  i++;  }  if(size == 0)  {  string[] test2 = new string[1];  test2[0] = "Aucun effet détecté";  test = test2;  }    // return tableau id/nom des médicaments  return test;  } |
| /// <summary>  /// set Nom d'un medicament depuis son id  /// </summary>  /// <param name="id">l'id du medicament</param>  /// <returns>le nom du medicament</returns>  [WebMethod]  public void SetNomMedicament(String id, String newNom)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand set\_Id\_Com = conn.CreateCommand();  set\_Id\_Com.Parameters.AddWithValue("@new\_Nom", newNom);  set\_Id\_Com.Parameters.AddWithValue("@id", id);  set\_Id\_Com.CommandText = "UPDATE medicaments SET nom\_medicament = @new\_Nom WHERE id\_medicament = @id";  set\_Id\_Com.ExecuteNonQuery();  } |
| /// <summary>  /// get Description d'un medicament depuis son id  /// </summary>  /// <param name="id">l'id du medicament</param>  /// <returns>la description du medicament</returns>  [WebMethod]  public String GetDescMedicament(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT description\_medicament FROM medicaments WHERE id\_medicament = '" + id + "'";  return (Convert.ToString(get\_M.ExecuteScalar()));    } |
| /// <summary>  /// get boolean ordonnance d'un medicament depuis son id  /// </summary>  /// <param name="id">l'id du medicament</param>  /// <returns>la valeur de l'ordonnance du medicament</returns>  [WebMethod]  public Boolean GetOrdMedicament(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "SELECT ordonnance\_medicament FROM medicaments WHERE id\_medicament = '" + id + "'";  return Convert.ToBoolean(get\_M.ExecuteScalar());  } |
| /// <summary>  /// Get liste des id des médicaments sous ordonnance ou non  /// </summary>  /// <returns>le tableau des ids</returns>  [WebMethod]  public String[] GetListMedicamentsOrdonnance(Boolean ord)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  // Creation Commande Select All Categories  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "SELECT \* FROM medicaments WHERE ordonnance\_medicament = " + ord;  // Initialisation liste de String : id et nom des categories  var listeCategories = new List<String>();  MySqlDataReader datareader = get\_M.ExecuteReader();  while (datareader.Read())  {  listeCategories.Add(Convert.ToString(datareader["code\_categorie"]));  }  // Incrémentation tableau pour return  int size = listeCategories.Count;  string[] test = new String[size];  int i = 0;  foreach (String categorie in listeCategories)  {  test[i] = categorie;  i++;  }  // return tableau id/nom des médicaments  return test;  } |
| //liste des méthodes concernant la table CATEGORIES  /// <summary>  /// Get liste des id de la table categories  /// </summary>  /// <returns>le tableau des ids</returns>  [WebMethod]  public String[] GetListCategories()  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  // Creation Commande Select All Categories  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "SELECT \* FROM categories";  // Initialisation liste de String : id et nom des categories  var listeCategories = new List<String>();  MySqlDataReader datareader = get\_M.ExecuteReader();  while (datareader.Read())  {  listeCategories.Add(Convert.ToString(datareader["code\_categorie"]));  }  // Incrémentation tableau pour return  int size = listeCategories.Count;  string[] test = new String[size];  int i = 0;  foreach (String categorie in listeCategories)  {  test[i] = categorie;  i++;  }  // return tableau id/nom des médicaments  return test;    } |
| /// <summary>  /// get Nom d'une categorie depuis son id  /// </summary>  /// <param name="id">l'id de la categorie</param>  /// <returns>le nom de la categorie</returns>  [WebMethod]  public String GetNomCategorie(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT nom\_categorie FROM categories WHERE code\_categorie = '" + id + "'";  return (Convert.ToString(get\_M.ExecuteScalar()));    } |
| /// <summary>  /// get description d'une categorie depuis son id  /// </summary>  /// <param name="id">l'id de la categorie</param>  /// <returns>la description de la categorie</returns>  [WebMethod]  public String GetDescCategorie(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT description\_categorie FROM categories WHERE code\_categorie = '" + id + "'";  return (Convert.ToString(get\_M.ExecuteScalar()));    } |
| //liste des méthodes concernant la table EFFETS  /// <summary>  /// Get liste des id de la table Effets  /// </summary>  /// <returns>le tableau des ids</returns>  [WebMethod]  public String[] GetListEffets()  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  // Creation Commande Select All Categories  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "SELECT \* FROM effets";  // Initialisation liste de String : id et nom des categories  var listeEffets = new List<String>();  MySqlDataReader datareader = get\_M.ExecuteReader();  while (datareader.Read())  {  listeEffets.Add(Convert.ToString(datareader["id\_effet"]));  }  // Incrémentation tableau pour return  int size = listeEffets.Count;  string[] test = new String[size];  int i = 0;  foreach (String effet in listeEffets)  {  test[i] = effet;  i++;  }  // return tableau id/nom des médicaments  return test;    } |
| /// <summary>  /// get Nom d'un effet depuis son id  /// </summary>  /// <param name="id">l'id de l'effet</param>  /// <returns>le nom de l'effet</returns>  [WebMethod]  public String GetNomEffet(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT nom\_effet FROM effets WHERE id\_effet = '" + id + "'";  return (Convert.ToString(get\_M.ExecuteScalar()));    } |
| /// <summary>  /// get Description de l'effet depuis son id  /// </summary>  /// <param name="id">l'id de l'effet</param>  /// <returns>la description de l'effet</returns>  [WebMethod]  public String GetDescEffet(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT description\_effet FROM effets WHERE id\_effet = '" + id + "'";  return (Convert.ToString(get\_M.ExecuteScalar()));    } |
| //liste des méthodes concernant la table PROBLEMES  /// <summary>  /// Get liste des id de la table Probleme  /// </summary>  /// <returns>le tableau des ids</returns>  [WebMethod]  public String[] GetListProblemes()  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  // Creation Commande Select All Categories  MySqlCommand get\_M = conn.CreateCommand();  get\_M.CommandText = "SELECT \* FROM problemes";  // Initialisation liste de String : id et nom des categories  var listeProblemes = new List<String>();  MySqlDataReader datareader = get\_M.ExecuteReader();  while (datareader.Read())  {  listeProblemes.Add(Convert.ToString(datareader["code\_probleme"]));  }  // Incrémentation tableau pour return  int size = listeProblemes.Count;  string[] test = new String[size];  int i = 0;  foreach (String probleme in listeProblemes)  {  test[i] = probleme;  i++;  }  // return tableau id/nom des médicaments  return test;    } |
| /// <summary>  /// get Nom d'un probleme depuis son id  /// </summary>  /// <param name="id">l'id du probleme</param>  /// <returns>le Nom du probleme</returns>  [WebMethod]  public String GetNomProbleme(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT nom\_probleme FROM problemes WHERE code\_probleme = '" + id + "'";  return (Convert.ToString(get\_M.ExecuteScalar()));    } |
| /// <summary>  /// get Description d'un probleme depuis on id  /// </summary>  /// <param name="id">l'id du probleme</param>  /// <returns>la description du probleme</returns>  [WebMethod]  public String GetDescProbleme(String id)  {  // Connection Base de Données  String connString = "Server=127.0.0.1;database=gsb;Uid=root;Password=;";  MySqlConnection conn = new MySqlConnection(connString);  conn.Open();  MySqlCommand get\_M = conn.CreateCommand();    get\_M.CommandText = "SELECT description\_probleme FROM problemes WHERE code\_probleme = '" + id + "'";  return (Convert.ToString(get\_M.ExecuteScalar()));    } |

# PHP

|  |
| --- |
| function conn()  {  $utilisateur = "root";  $mdp = "";  $server = "mysql:host=localhost;dbname=gsb";  // connexion à la base de données  try  {  $bd = new PDO($server, $utilisateur,  $mdp);  }  catch(Exception $e)  {  die("Erreur : ".$e->getMessage());  }  return $bd;  } |
| unction getMedicaments()  {  $bd = conn ();  $requete = "SELECT \* FROM medicaments";  $bd->query("SET NAMES utf8");  $resultat = $bd->query($requete);  $medicaments = array();  PHP $medicaments  while($ligne = $resultat->fetch() )  {  $medicaments[] = $ligne;  }  $resultat->closeCursor();  return $medicaments;  } |