Fiche Technique

bts sio / 2°Année

2020-2021

Table des matières

[1 - Résumé du document 3](#_Toc52289130)

[2 - Rappel sur le fonctionnement de l'application 4](#_Toc52289131)

[2.1 - Description du logiciel 4](#_Toc52289132)

[2.2 - Décomposition du projet 5](#_Toc52289133)

[2.3 - Architecture globale 6](#_Toc52289134)

[2.4 - Technologies utilisées 7](#_Toc52289135)

[4 – Application client 8](#_Toc52289136)

[4.1 – Interactions 8](#_Toc52289137)

[3.4 - Diagramme de classes 9](#_Toc52289138)

[4 - Serveur 10](#_Toc52289139)

[4.1 - Modèle de données 10](#_Toc52289140)

[5 – Bilan et Conclusion 11](#_Toc52289141)

[Annexes 12](#_Toc52289142)

[Cahier des charges 12](#_Toc52289143)

[Document Access 12](#_Toc52289144)

[Vidéo 12](#_Toc52289145)

# 1 - Résumé du document

La DSI a choisi C# comme langage de développement. Grâce à ses «WinForms», ce langage permettra de créer une application à la fois proche de l’existant mais avec des interfaces plus conviviales et modernes.

La nouvelle application doit reprendre l’intégralité des contenus de l’ancienne

Ce document décrit l’intégralité de l’architecture du projet GSB. Notre projet étant une application C#, MySQL . Notre application nécessites des user pour se connecter à son compte, Une fois connecter il est possible de naviguer et d’afficher les données que l’on souhaite consulter.

Nous utilisons essentiellement Github pour gérer le code de notre projet et les tâches en cours de réalisation.

# 2 - Rappel sur le fonctionnement de l'application

## 2.1 - Description du logiciel

Le laboratoire a mis à disposition des visiteurs médicaux sur une application permettant de centraliser les compte-rendu de visite. Cette base d'information est utilisée à des fins d'élaboration de la démarche de communication auprès des praticiens et donne une vision individuelle et synthétique de l'activité de représentation.

Pour permettre une aide au renseignement des rapports, l'application fournit une description des produits du laboratoire, les coordonnées précises des praticiens et des informations détaillées les concernant.

Cette application a été développée avec Access et se nomme Swiss\_Visite.mdb.

L’application développée avec Access, quoique très pratique pour les visiteurs, présente de nombreuses limites. Son ergonomie est très largement perfectible et chaque visiteur a sur son poste une version des données : il convient que la base de données soit centralisée.

La DSI a choisi C# comme langage de développement. Grâce à ses «WinForms», ce langage permettra de créer une application à la fois proche de l’existant mais avec des interfaces plus conviviales et modernes.

Les visiteurs médicaux sont habitués à l’application Access et sont inquiets face à ce changement. La nouvelle application devra reprendre les éléments graphiques de l’ancienne application (logoGSB, texte en bleu, fond d’écran blanc).L’ergonomie de la nouvelle application doit cependant être plus conviviale que l’ancienne.

## 2.2 - Décomposition du projet

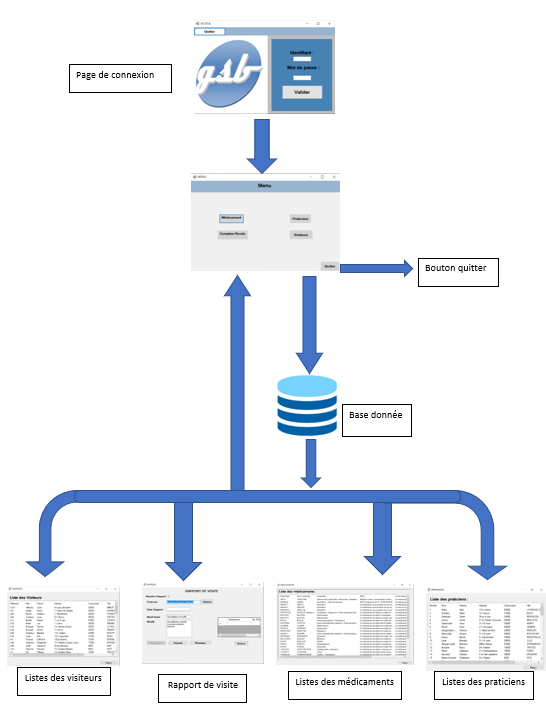
Notre projet se décompose en différentes parties :

• Une application client, développer en c#, des classes métiers et des classes d’accès aux données doivent être développées ;

• Une base de données, les données d’Access ont été exportées vers MySQL. Communiquant en permanence avec le logiciel client, pour assurer de multiples fonctions comme le contrôle de la licence du logiciel et la gestion du planning à distance.

La nouvelle application devra reprendre les éléments graphiques suivant (logo GSB, texte en bleu, fond d’écran blanc). L’ergonomie de la nouvelle application doit cependant être plus conviviale que l’ancienne

## 2.3 - Architecture globale



# 2.4 - Technologies utilisées

Le logiciel client est codé en **C#** et utilise **Mysql**. Visual Studio fournit tous les éléments nécessaires à la réalisation de notre application.

**Mysql** est un système de gestion de base de données relationnelles (SGBDR) basé sur SQL (Structured Query Language).  
  
**MySQL** fonctionne sur pratiquement toutes les plates-formes, y compris Linux, Unix et Windows. Il est entièrement multi-thread avec un noyau de threads, et fournit des API (Application Programming Interface) pour de nombreux langages de programmation, notamment C#.

MySQL pour Visual Studio permet d'accéder aux objets et données MySQL sans forcer les développeurs à quitter Visual Studio.

Conçu et développé en tant que package Visual Studio, MySQL pour Visual Studio s'intègre directement dans l'Explorateur de serveurs, offrant une expérience transparente pour la configuration de nouvelles connexions et l'utilisation d'objets de base de données.

# 4 – Application client

## 4.1 – Interactions

Pour accéder à l’application, le client doit se connecter via son compte préalablement créer.

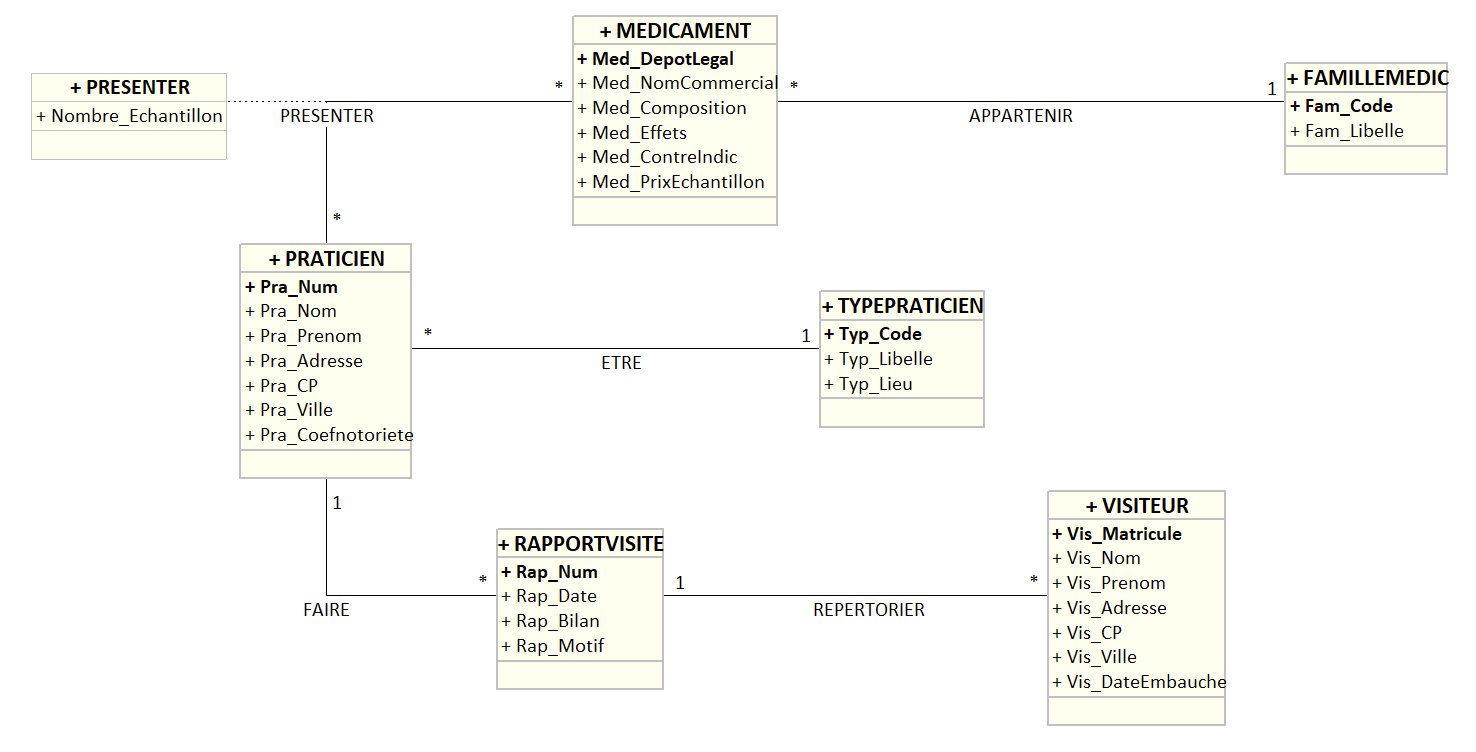
Une fois connecter à son compte. L’utilisateur tombe sur le menu.

Depuis le menu il lui est possible de naviguer sur les diffèrent formulaire (Visiteur, Praticiens, Médicaments et Rapport\_Visite.

Les formulaires Visiteur, Praticiens, et Médicament afficherons tous les visiteur, praticiens et médicaments référencer dans la base de données.

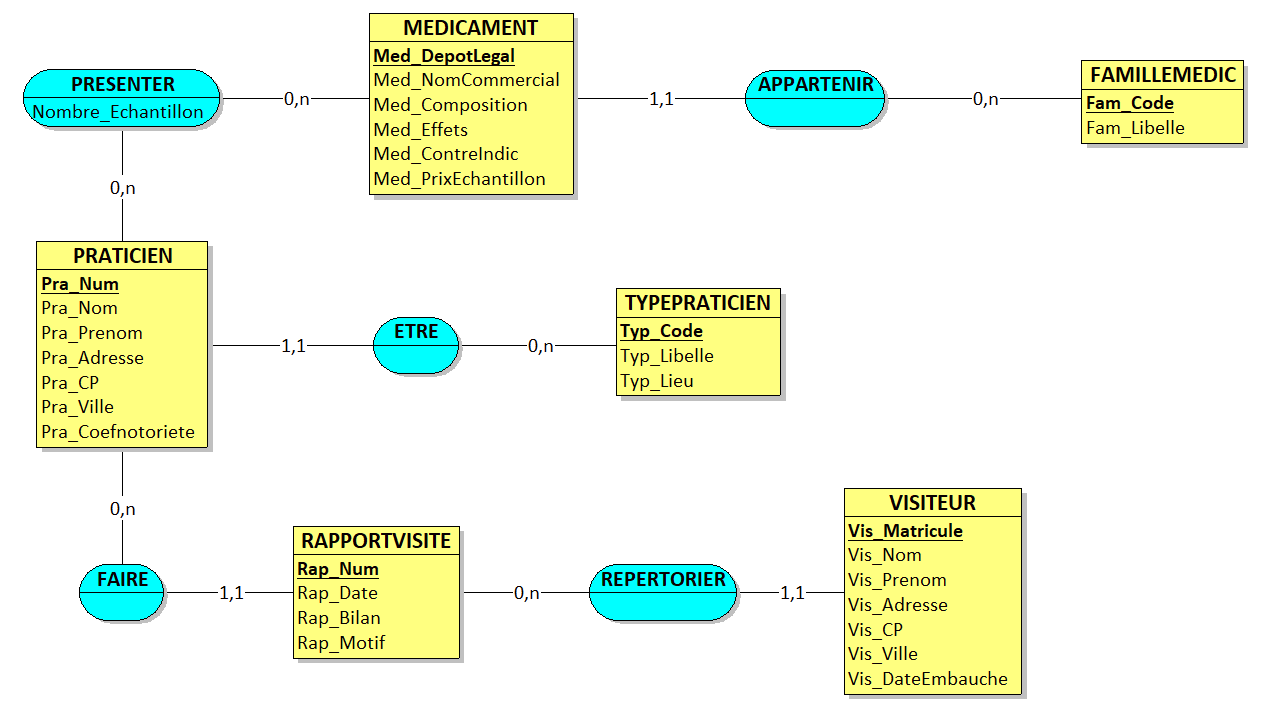
Le formulaire rapport quant à lui permet de rechercher des rapports de visites. Affichant le médicament prescrit par le praticiens.

## 3.4 - Diagramme de classes



# 4 - Serveur

## 4.1 - Modèle de données



La base de données MySQL permet de stocker les données de l’application (telles les données d’un praticien, les données des médicaments…A quelle famille appartient-il…etc.), afin de pouvoir les afficher sur le programme et permettre à l'utilisateur de consulter ces données.

Cette base de données contient également les comptes utilisateurs, permettant de se connecter à l’application Client via un compte préalablement crée

# Annexes

## Cahier des charges

## Document Access

## Vidéo