



GESAMM

Validation de compétences



A1.3.4 Déploiement d'un service

A1.4.1 Participation à un projet

A4.1.2 Conception ou adaptation de l'interface utilisateur d'une solution applicative

A4.1.3 Conception ou adaptation d'une base de données

A4.1.6 Gestion d'environnements de développement et de test

A4.1.7 Développement, utilisation ou adaptation de composants logiciels

A4.1.9 Rédaction d'une documentation technique

A5.2.1 Exploitation des référentiels, normes et standards adoptés par le prestataire informatique

A5.2.4 Étude d'une technologie, d'un composant, d'un outil ou d'une méthode



A4.1.2 Conception ou adaptation de l'interface utilisateur d'une solution applicative

C4.1.2.1 Définir les spécifications de l'interface utilisateur de la solution applicative

J'ai eu à réaliser des IHM pour permettre à mon utilisateur de discuter avec la machine. J'ai donc eu à réaliser une interface pour la connexion et une pour la modification des étapes normées. J'ai donc dû utiliser pour les étapes normées, ce que l'utilisateur devait modifier comme données afin qu'elle soit modifiée (dans ce cas la norme en elle-même et la date de celle-ci). Puis j'ai dû penser à faire que la machine communique avec le serveur SQL pour modifier les informations dans la base de données.

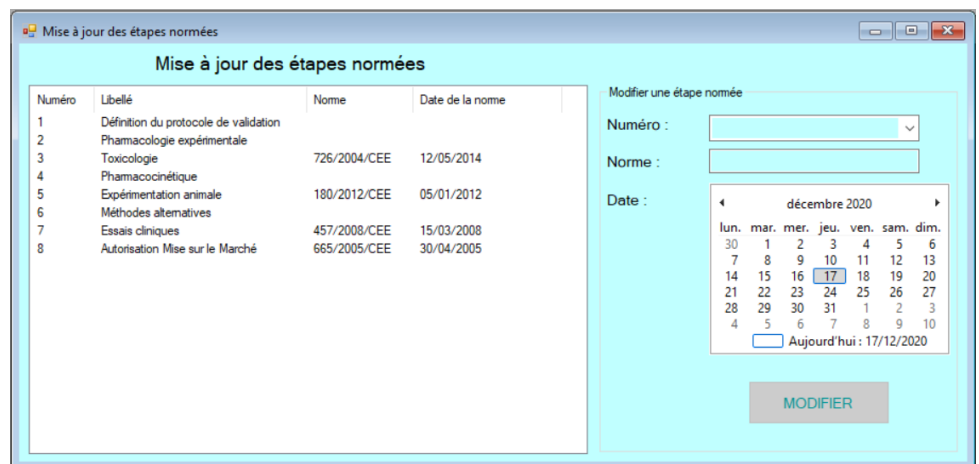
C4.1.2.2 Maquetter un élément de la solution applicative



g**sb**

Login :

Mot de passe :



Mise à jour des étapes normées

Numéro	Libellé	Norme	Date de la norme
1	Définition du protocole de validation		
2	Pharmacologie expérimentale		
3	Toxicologie	726/2004/CEE	12/05/2014
4	Pharmacocinétique		
5	Expérimentation animale	180/2012/CEE	05/01/2012
6	Méthodes alternatives		
7	Essais cliniques	457/2008/CEE	15/03/2008
8	Autorisation Mise sur le Marché	665/2005/CEE	30/04/2005

Modifier une étape normée

Numéro :

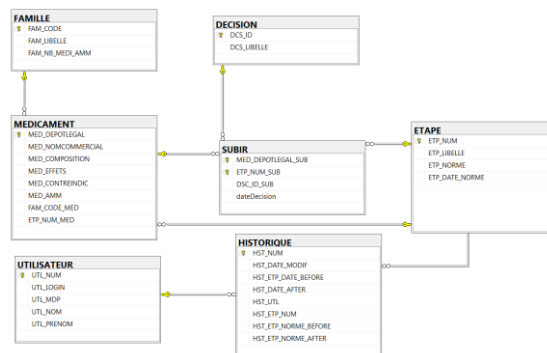
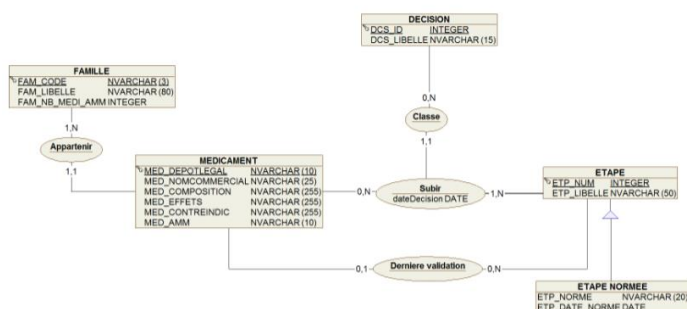
Norme :

Date :

A4.1.3 Conception ou adaptation d'une base de données

C4.1.3.2 Implémenter le schéma de données dans un SGBD





C4.1.3.3 Programmer des éléments de la solution applicative dans le langage d'un SGBD

C4.1.3.4 Manipuler les données liées à la solution applicative à travers un langage de requête

```

1 USE [dtb_gesAMM]
2 GO
3 /***** Object: Trigger [dbo].[trg_update_historique]    Script Date: 17/12/2020 14:54:12 *****/
4 SET ANSI_NULLS ON
5 GO
6 SET QUOTED_IDENTIFIER ON
7 GO
8 ALTER TRIGGER [dbo].[trg_update_historique]
9 ON [dbo].[ETAPE]
10 AFTER UPDATE
11 AS
12 BEGIN
13     SET NOCOUNT ON;
14
15     DECLARE @dateMAJ as date
16     DECLARE @util as int
17     DECLARE @numEtape as int
18     DECLARE @normeAvant as varchar(20)
19     DECLARE @normeApres as varchar(20)
20     DECLARE @dateAvant as date
21     DECLARE @dateApres as date
22
23     --récupérer les nouvelles informations
24     SELECT @dateApres = ETP_DATE_NORME, @normeApres = ETP_NORME, @numEtape = ETP_NUM from inserted
25
26     --récupérer les informations supprimées
27     SELECT @normeAvant = ETP_NORME, @dateAvant = ETP_DATE_NORME from deleted
28
29     --récupérer les anciennes informations
30     SELECT @util = UTL_NUM from UTILISATEUR
31
32     --récupérer la date du jour
33     SELECT @dateMAJ = convert (date, SYSDATETIME());
34
35     --Insérer la ligne dans la table HISTORIQUE
36     INSERT INTO HISTORIQUE(HST_DATE_MODIF, HST_ETP_DATE_BEFORE, HST_ETP_DATE_AFTER, HST_UTL, HST_ETP_NUM, HST_ETP_NORME_BEFORE, HST_ETP_NORME_AFTER)
37     VALUES (@dateMAJ, @dateAvant, @dateApres, @util, @numEtape, @normeAvant, @normeApres)
38
39 END
40

```

A4.1.6 Gestion d'environnements de développement et de test

C4.1.6.1 Mettre en place et exploiter un environnement de développement

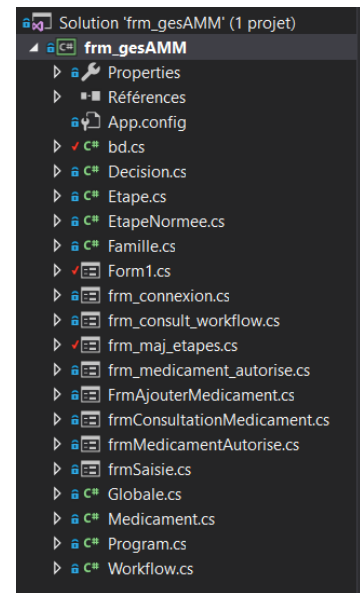
Pour ce projet, nous avons travaillé avec Visual Studio 2017 en windows form. Nous avons également travaillé avec SQL Server et GitHub.



A4.1.7 Développement, utilisation ou adaptation de composants logiciels

C4.1.7.1 Développer les éléments d'une solution

Nous avons développé directement via Visual Studio et partagé toute notre avancé via GitHub (notre outil de gestion de projet). Nous avons donc pu travailler en simultané sans que rien ne soit supprimé.



C4.1.7.4 Utiliser des composants d'accès aux données

Pour accéder à nos données, notre base de données était hébergée en local via SQL Server puis lorsque le projet fût terminé nous l'avons exporté au format SQL pour que le client ai juste à lancé le script pour avoir la base de données et le contenu des tables.



A4.1.9 Rédaction d'une documentation technique

C4.1.9.1 Produire ou mettre à jour la documentation technique d'une solution applicative et de ses composants logiciels

Trigger de la table ETAPE : trg_update_historique

Auteur : Thibaut MESLIN

Objectif du trigger :

Garder une trace des précédentes informations concernant les normes dans un table dédiée (HISTORIQUE).

Cas de déclenchement :

Se déclenche quand la colonne ETP_NORME et/ou ETP_DATE_NORME de la table ETAPE sont modifiés.

Paramètre(s) d'entrée :

- ☐ @dateMAJ → **date** : date de la mise à jour (date système)
- ☐ @util → **int** : id de l'utilisateur ayant fait la modification
- ☐ @numEtape → **int** : numéro de l'étape qui a été modifié
- ☐ @normeAvant → **varchar** de longueur 20 : norme précédente (avant insertion)
- ☐ @normeAprès → **varchar** de longueur 20 : norme actuelle (après insertion)
- ☐ @dateAvant → **date** : date de la norme précédente (avant insertion)
- ☐ @dateAprès → **date** : date de la norme actuelle (après insertion)

Requête(s) utile(s) au(x) variable(s) :

- ☐ SELECT @dateAprès = ETP_DATE_NORME, @normeAprès = ETP_NORME, @numEtape = ETP_NUM from inserted
 - ☐ Récupérer les nouvelles informations (nouvelle norme et date) et le numéro de l'étape modifiée
- ☐ SELECT @normeAvant = ETP_NORME, @dateAvant = ETP_DATE_NORME from deleted
 - ☐ Récupérer les informations supprimées (ancienne norme et date)
- ☐ SELECT @util = UTL_NUM from UTILISATEUR
 - ☐ Récupérer le numéro de l'utilisateur
- ☐ SELECT @dateMAJ = convert (date, SYSDATETIME())
 - ☐ Récupérer la date du système

Requête(s) utile(s) à l'objectif :

- ☐ INSERT INTO HISTORIQUE(HST_DATE_MODIF, HST_ETP_DATE_BEFORE, HST_DATE_AFTER, HST_UTL, HST_ETP_NUM, HST_ETP_NORME_BEFORE, HST_ETP_NORME_AFTER) VALUES (@dateMAJ, @dateAvant, @dateAprès, @util, @numEtape, @normeAvant, @normeAprès)
 - ☐ Insertion d'une nouvelle ligne dans la table HISTORIQUE contenant toutes les informations souhaitées



A5.2.1 Exploitation des référentiels, normes et standards adoptés par le prestataire informatique

C5.2.1.2 Identifier et partager les bonnes pratiques à intégrer

Nous avons pour la base de données mis tous les noms en majuscule et les colonnes ont toutes un trigramme sans voyelle de la table (Ex : table HISTORIQUE – colonne HST_XXX). Nous avons ensuite utilisé les nommages de base de la programmation orienté objet en C#. Tous les accesseurs sont précédés de "get" et tous les mutateurs sont précédés de "set".

A5.2.4 Étude d'une technologie, d'un composant, d'un outil ou d'une méthode

C5.2.4.1 Se documenter à propos d'une technologie, d'un composant, d'un outil ou d'une méthode

Avec le projet était fourni une ébauche d'application utilisant les méthodes dont nous avons besoin pour utiliser des procédures stockées directement en langage C#. Ce projet nous a aidés à nous auto-former pour utiliser toutes les méthodes dont nous allions avoir besoin pour mener à bien ce projet et donc développer plus facilement en emmagasinant du savoir.

