
Analyse SGBD

William Wauters

April 6, 2021

Contents

1	Contexte	4
1.1	Cadre	4
1.2	Projet	4
2	Analyse métier	5
2.1	Description de la solution envisagée	5
2.2	Intervenants	5
2.3	Fonctions attendues	6
2.3.1	Exigences fonctionnelles	6
2.3.1.1	Fonctionnalité obligatoire application patient	6
2.3.1.2	Fonctionnalité obligatoire application médecin	6
2.3.2	Exigences non fonctionnelles	6
2.4	Contraintes business	7
2.4.1	Point de vue technique	7
2.4.2	Composition du cahier des charges	7
2.4.3	Architecture du projet, à respecter	7
3	Analyse fonctionnelle	8
3.1	Diagramme USE CASES (Les principaux)	8
3.2	Descriptions textuelles des différents USE CASES	8
3.2.1	Application Patient	8
3.2.1.1	EF-AP-001 – Faire une demande de consultation	8
3.2.1.2	EF-AP-001 – Modifier une demande de consultation	8
3.2.1.3	EF-AP-001 – Supprimer une demande de consultation	8
3.2.2	Application Médecin	8
3.2.2.1	EF-AM-001 – Ajouter ces présences	8
3.2.2.2	EF-AM-002 – Confirmer une consultation	8
4	Contraintes fonctionnelles	9
4.1	Règles d'accès et autorisation	9
4.2	Règles de structure	9
4.3	Règles de validation	9
4.4	Règles de calcul	9
5	Description des entités	10
5.1	Patient	10
5.2	Médecin	10
5.3	Spécialité	10
5.4	Maison médicale	10
5.5	Consultation	10
6	Schéma relationnel de la solution	11
6.1	Schéma entité-association	11
6.2	Schéma relationnel	11
6.3	Implémentation des contraintes	11
7	Analyse technique	12
7.1	Technologies proposées	12
7.2	Architecture applicative	12
7.3	Algorithmes spécifiques	12

7.4 Erreurs	12
8 ANNEXES	13

1. Contexte

1.1 Cadre

Expliquer pourquoi nous faisons ce projet

Ce projet est développé dans le cadre du cours de Projet de développement SGBD de Monsieur Fievez.

A travers ce projet plusieurs points sont demandés à l'étudiant:

- Analyse d'une demande d'un client (pas toujours claire et précise)
- Réalisation d'une entité association
- Réalisation d'une base de données
- Réalisation de triggers et procédures stockées pour appliquer les contraintes
- Réalisation d'un code en 3 couches - DAL/ BL / View
- Utilisations de modèles et d'erreurs séparées de la couche business
- Accès aux données en lecture et écriture uniquement par procédures stockées

1.2 Projet

Le projet en lui-même consiste au développement d'un système d'information permettant de faire fonctionner deux systèmes applicatifs avec des fonctionnalités différentes.

- Application permettant la gestion des maisons médicales et des médecins
- Application permettant la gestion des réservations des patients

Ces deux applications doivent utiliser une même base de données mais des utilisateurs et vues différentes.

2. Analyse métier

2.1 Description de la solution envisagée

Notre solution permet la gestion des réservations de consultations entre patients et médecins dans des maisons médicales. Cette solution sera composé de deux applications distinctes.

Partie médecin

Partie patients

2.2 Intervenants

Les intervenants prévus pour notre solutions

- Médecin: Ajoute ces disponibilités, confirme des consultations
- Patient: Effectue des réservations

2.3 Fonctions attendues

2.3.1 Exigences fonctionnelles

2.3.1.1 Fonctionnalité obligatoire application patient

Code	Description
EF-AP-001	Créer un nouveau patient
EF-AP-002	Sélectionner un patient dans la liste de patients
EF-AP-003	Consulter la liste des prochaines consultations
EF-AP-004	Consulter la liste des consultations confirmées
EF-AP-005	Consulter la liste des consultations non confirmées
EF-AP-006	Faire une demande de consultation
EF-AP-007	Modifier une demande de consultation
EF-AP-008	Supprimer une demande de consultation

2.3.1.2 Fonctionnalité obligatoire application médecin

Code	Description
EF-AM-001	Créer un nouveau médecin
EF-AM-002	Sélectionner un médecin dans la liste des médecins
EF-AM-003	Visualiser son planning
EF-AM-004	Visualiser la liste de ces consultations
EF-AM-005	Visualiser la liste de ces consultations non confirmées
EF-AM-006	Confirmer une demande de consultation
EF-AM-007	Ajouter ces présences dans une maison médicale
EF-AM-008	Modifier ces présences dans une maison médicale
EF-AM-009	Visualiser la liste des maisons médicales dans lesquelles il travail
EF-AM-010	S'inscrire dans une maison médicale
EF-AM-011	Modifier durée minimum d'une consultation dans une maison médicale

2.3.2 Exigences non fonctionnelles

Code	Description
ENF-001	Facilité d'encodage pour l'ajout des présences du médecin
ENF-002	Être ergonomique pour l'utilisateur
ENF-003	Permettre une vue complète et simplifiée des informations

2.4 Contraintes business

Avoir une découpe en 3 couches (DAL, BL, GUI)

Être divisé en deux parties

Accéder à la base de données via Entity Framework DB First

Avoir une seule base de données dédiée pour les deux parties

Autoriser l'accès aux tables de la base de données uniquement par l'utilisateur admin (sa)

Avoir deux schéma différents (patient, médecin)

Accéder aux données et les modifier via des procédures stockées spécifiques par schéma.

2.4.1 Point de vue technique

2.4.2 Composition du cahier des charges

2.4.3 Architecture du projet, à respecter

3. Analyse fonctionnelle

Pas pour le création des locaux, mais bien pour les scénarios de réservations

3.1 Diagramme USE CASES (Les principaux)

3.2 Descriptions textuelles des différents USE CASES

3.2.1 Application Patient

3.2.1.1 EF-AP-001 – Faire une demande de consultation

Titre : Faire une demande de consultation

Résumé : Faire une demande de consultation en fonction d'une spécialité et d'une maison médicale

Acteurs : Patient

Date de création : 06/04/2021

Date de mise à jours : 06/04/2021

Version : 1.0

Responsable : William Wauters

Pré-conditions :

- Le patient à une compte
- Le patient est disponible

Sénario nominale :

Echaimement alternatifs :

Echaimement d'erreurs :

Post-conditions :

3.2.1.2 EF-AP-001 – Modifier une demande de consultation

3.2.1.3 EF-AP-001 – Supprimer une demande de consultation

3.2.2 Application Médecin

3.2.2.1 EF-AM-001 – Ajouter ces présences

3.2.2.2 EF-AM-002 – Confirmer une consultation

4. Contraintes fonctionnelles

4.1 Règles d'accès et autorisation

L'application patient est uniquement accessible aux utilisateurs patients

...

4.2 Règles de structure

4.3 Règles de validation

4.4 Règles de calcul

5. Description des entités

5.1 Patient

5.2 Médecin

5.3 Spécialité

5.4 Maison médicale

5.5 Consultation

- 6. Schéma relationnel de la solution**
 - 6.1 Schéma entité-association**
 - 6.2 Schéma relationnel**
 - 6.3 Implémentation des contraintes**

7. Analyse technique

7.1 Technologies proposées

La base de données a été réalisée en SQL Server

Les applications sont réalisées avec ASP.NET MVC

Utilisation d'une API ASP.NET avec Entity Framework DB first

7.2 Architecture applicative

7.3 Algorithmes spécifiques

7.4 Erreurs

8. ANNEXES