



---

# HILLTH

---

Projet SGBD – Adrien MOUSTY



01 JUIN 2016

HEPH CONDORCET

Projet personnel – Mme COPIN

## Table des matières

|  |    |
|--|----|
| Consignes .....  | 4  |
| Énoncé.....  | 5  |
| Pourquoi « Hillth » ? .....  | 5  |
| Présentation .....   | 5  |
| Fonctionnalités .....  | 6  |
| Schéma - diagrammes .....  | 7  |
| Schéma conceptuel.....   | 7  |
| Diagramme de contexte statique .....                                       | 8  |
| Diagramme des use case .....   | 9  |
| Diagramme de classe .....  | 10 |
| TOE (Tâche – objet – événement).....                                       | 11 |
| MainWindow .....   | 11 |
| Login .....  | 11 |
| Register .....   | 12 |
| Accueil.....   | 13 |
| Register Patient .....   | 15 |
| Register Medecin .....   | 16 |
| Patient Recherche.....   | 17 |
| Écrans de l'application .....  | 18 |
| Connexion .....  | 18 |
| S'enregistrer .....  | 19 |
| Accueil.....   | 20 |
| Afficher profil .....  | 21 |
| Rechercher .....   | 22 |
| Statistique .....  | 23 |
| Ajouter patient .....  | 24 |
| Ajout médecin.....   | 26 |
| Afficher patient.....  | 27 |
| Afficher tous les patients enregistrés .....                               | 27 |
| Afficher les patients correspondant uniquement au filtre sélectionné ..... | 28 |
| Nouveautés .....   | 29 |
| Theme (Material design in xaml toolik master) .....                        | 29 |
| Datepicker.....  | 30 |
| Insertion d'une video .....  | 31 |
| Procédure stockée .....  | 31 |

|  |    |
|--|----|
| Travail additionnel .....                  | 32 |
| Diagramme des use case par iteration ..... | 32 |
| Diagramme de classe partiel 1.....         | 33 |
| Diagramme de classe partiel 2.....         | 33 |
| diagramme de classe partiel 3 .....        | 34 |
| Diagramme de classe partiel 4.....         | 35 |
| Conclusion .....                           | 36 |
| Table des illustrations .....              | 37 |



## CONSIGNES

L'examen de juin 2016 portera sur la réalisation d'un projet individuel réalisé en C# intégrant une base de données.

Ce travail respectera les consignes suivantes :

1. Réalisation d'une application graphique WPF connectée à une base de données.
2. La base de données sera implémentée avec SQL Server et Linq<sup>1</sup>.
3. L'analyse comportera les points suivants :
  - a. L'énoncé
  - b. Le schéma conceptuel de la base de données (4 tables minimum)
  - c. Le diagramme des cas d'utilisation et le diagramme des classes
  - d. Le diagramme TOE (Tâches/ Objet/ Evènement)
  - e. Les différents écrans
  - f. Le code annoté.
4. Trois nouveautés à présenter.

Le dossier sera remis une semaine avant l'examen et présenté oralement

---

<sup>1</sup> LINQ to SQL est un composant de la version 3.5 du .NET Framework qui fournit une infrastructure runtime pour gérer les données relationnelles comme des objets. Dans LINQ to SQL, le modèle de données d'une base de données relationnelle est mappé à un modèle objet exprimé dans le langage de programmation du développeur. Lors de l'exécution de l'application, LINQ to SQL traduit des requêtes LINQ dans le modèle objet en SQL et les envoie à la base de données pour exécution. Lorsque la base de données retourne les résultats, LINQ to SQL les traduit en objets que vous pouvez utiliser dans votre propre langage de programmation (Microsoft)

## ÉNONCÉ

### POURQUOI « HILLTH » ?

Hillth est un mélange entre le mot anglais « Health » et « Hill », signifiant respectivement le soin et la montagne.

Quel est le rapport avec la montagne ? Le relief est similaire à celui d'un électrocardiogramme. L'analogie était donc assez simple et intéressante.



### PRÉSENTATION

Bien organiser ses papiers et avoir un suivi de plusieurs patients n'est pas une chose aisée. Surtout si leurs pathologies, leurs allergies, leurs antécédents familiaux sont classés dans un dossier papier.

Le dossier papier pour un patient n'est pas quelque chose de pratique. C'est encombrant, difficilement transportable et tenir une centaine de patients nécessite une très bonne organisation.

Le programme Hillth a été développé afin de faciliter toutes ces choses. Cela permettra au médecin d'optimiser d'une manière sûre son temps.

## FONCTIONNALITÉS

- Se loguer via son numéro INAMI et un mot de passe
- Ajouter plusieurs médecins, ayant des spécialisations variées
- Visionner son profil si le médecin connecté s'est ajouté dans la base de données
- Ajouter des patients
- Supprimer un patient
- Modifier certaines données pour un patient
- Rechercher des patients selon différents critères
  - Le nom
  - Le prénom
  - Le numéro NIS
  - S'il est tiers payant ou non
  - Les mutuelles
  - Les hôpitaux
- Affichage de statistiques diverses
  - Le nombre actuel de patients enregistrés
  - Le nombre de médecins enregistrés
  - Le pourcentage d'homme – femme enregistrés
  - Le nombre de patients affiliés à une certaine mutuelle
- Affichage d'une vidéo explicative sur le métier de médecin généraliste.

SCHÉMA - DIAGRAMMES

SCHÉMA CONCEPTUEL

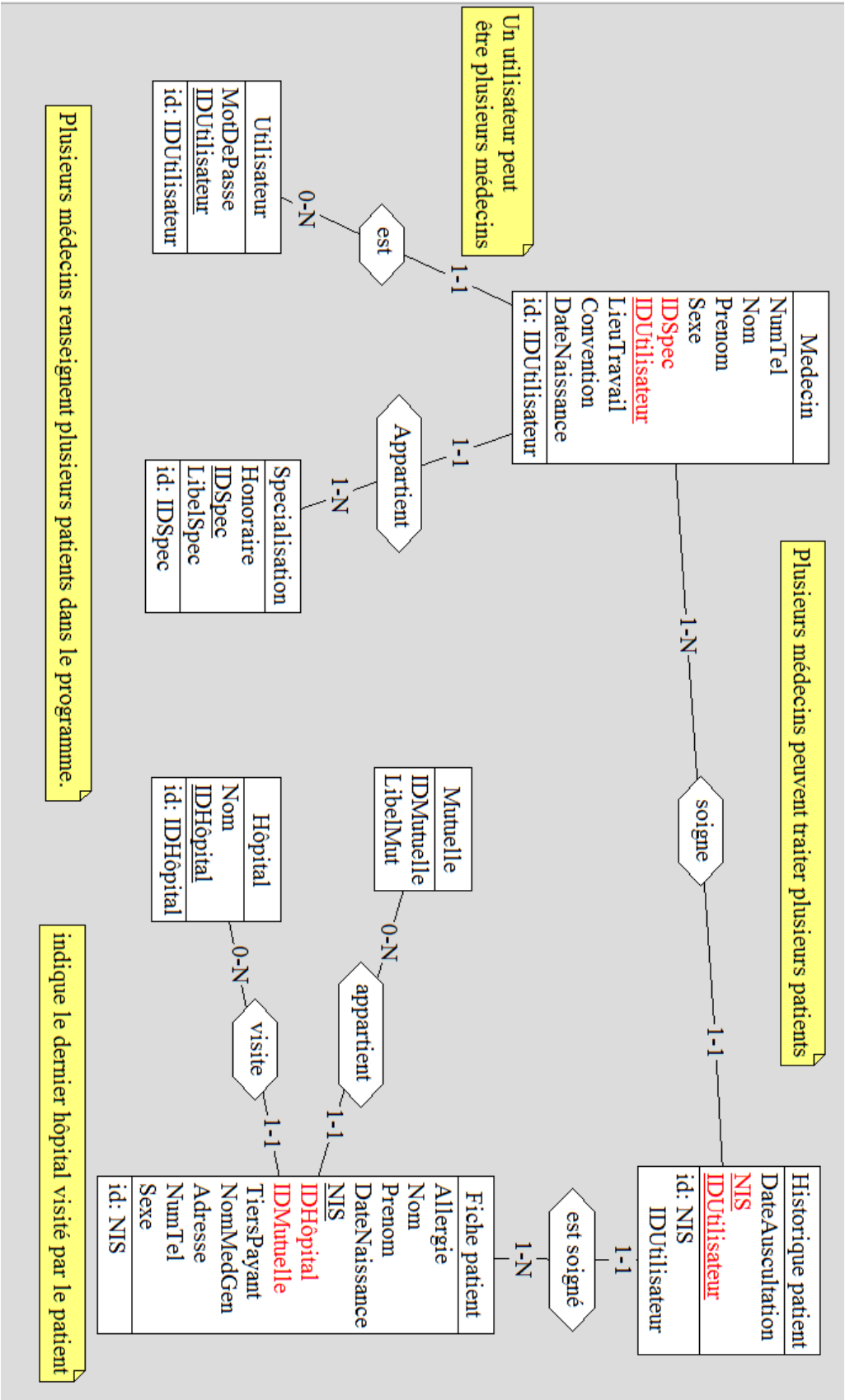


Figure 1 Schéma conceptuel



## DIAGRAMME DE CONTEXTE STATIQUE

Ce diagramme affiche les acteurs liés à l'application « Hillth ».

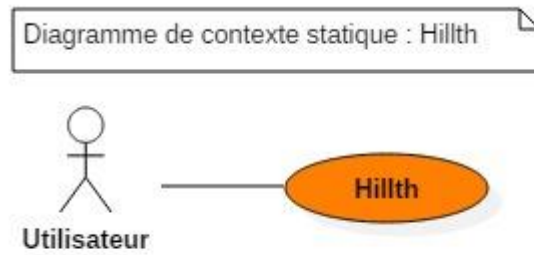


Figure 2 Diagramme de contexte statique

## DIAGRAMME DES USE CASE

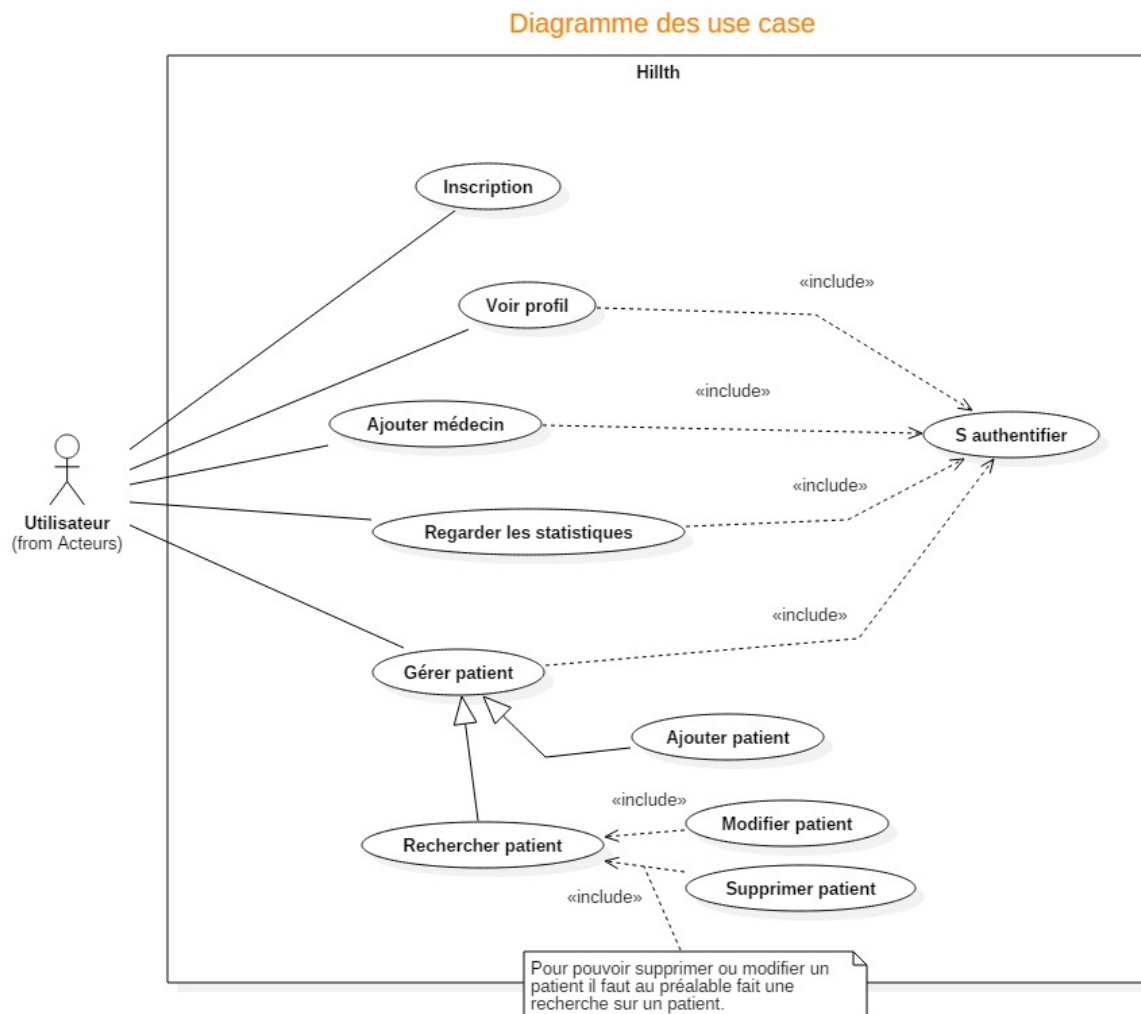


Figure 3 Diagramme des use case complet

Ce diagramme représente les fonctionnalités (ou **cas d'utilisation**) nécessaires aux utilisateurs.

On peut faire un diagramme de cas d'utilisation pour le logiciel entier ou pour chaque package.

Dans ce cas c'est un diagramme partiel des use case.

Dans ce diagramme, des « include » ont été utilisés. Que permettent-ils ?

Le premier cas d'utilisation inclut le second et son issue dépend souvent de la résolution du second. Ce type de description est utile pour extraire un ensemble de sous-comportements communs à plusieurs tâches.

## DIAGRAMME DE CLASSE

## Diagramme de classe : HILLTH

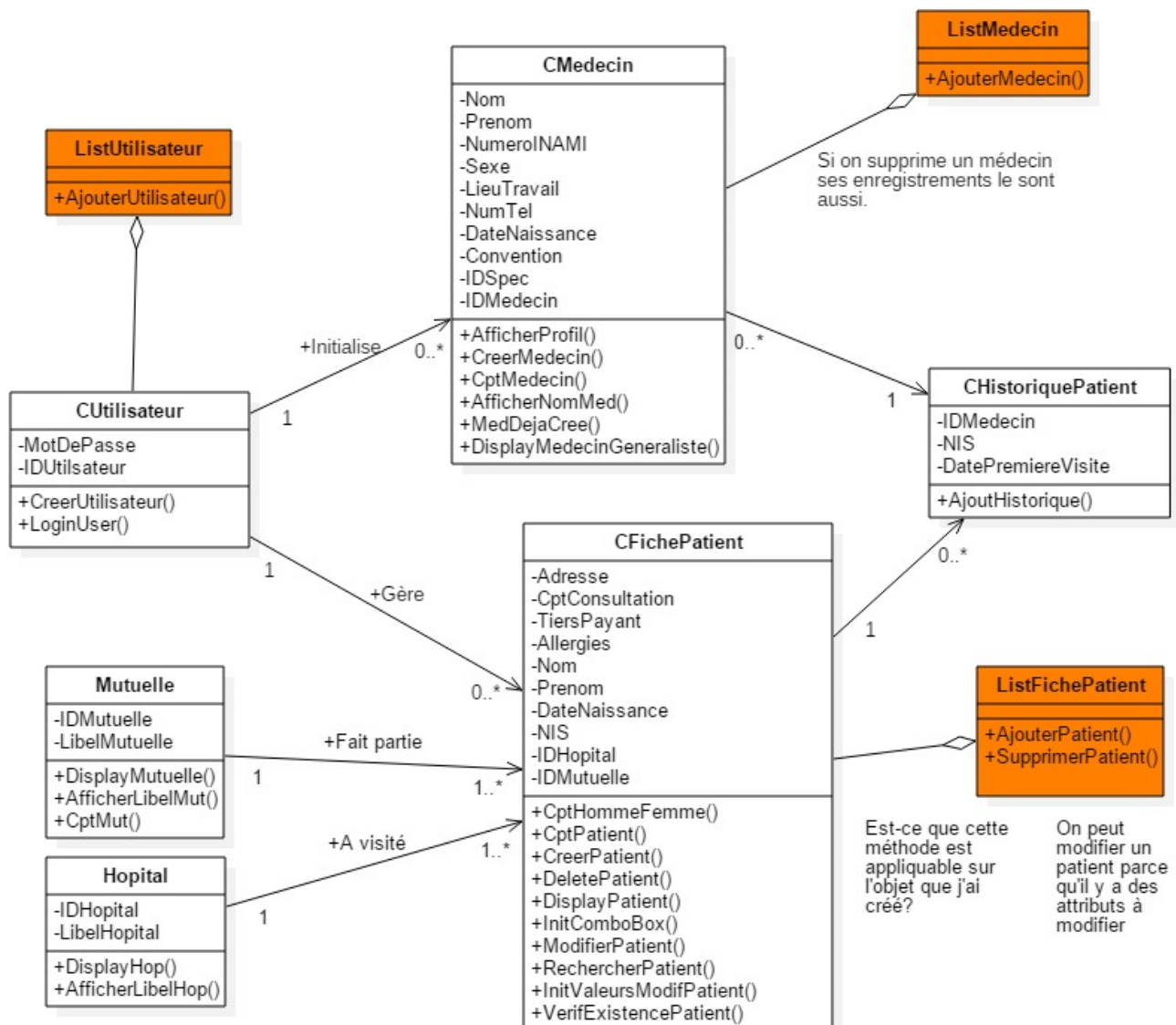


Figure 4 Diagramme de classe

## TOE (TÂCHE – OBJET – ÉVÉNEMENT)

## MAINWINDOW

| TÂCHE | OBJET | ÉVÉNEMENT |
|-------|-------|-----------|
| /     | /     | /         |

## LOGIN

| TÂCHE   | OBJET     | ÉVÉNEMENT |
|---|-----------|-----------|
| - Bouton permettant de se connecter au programme si les valeurs sont correctes  | btn_co    | Onclick   |
| - Si la connexion se fait, ferme la page « login » et ouvre la page « accueil » |           |           |
| - Bouton ouvrant la page « register » et fermant la page « login »              | btn_enreg | Onclick   |
| - Bouton fermant le programme   | btn_quit  | Onclick   |
| - Recueille le n° INAMI de l'utilisateur  | txt_login | /         |
| - Recueille le mot de passe de l'utilisateur                                    | txt_mdp   | /         |

## REGISTER

| TÂCHE   | OBJET          | ÉVÉNEMENT |
|---|----------------|-----------|
| - Bouton ouvrant la page « login » et fermant la page « login »   | btn_precUser   | Onclick   |
| - Valide le n° INAMI et le mot de passe. Si les valeurs sont correctes, les ajoute dans la base de données. | btn_submitUser | Onclick   |
| - Si les informations ont bien été enregistrées, ferme la page « register » et ouvre la page « login »      |                |           |
| - Recueille le n° INAMI d'un éventuel utilisateur   | txt_IDUser     | /         |
| - Recueille le mot de passe d'un éventuel utilisateur   | mdpBoxUser     | /         |

## ACCUEIL

| TÂCHE  | OBJET          | ÉVÉNEMENT |
|--|----------------|-----------|
| <ul style="list-style-type: none"> <li>- Si activé, affiche le stackpanel latéral gauche</li> <li>- Si désactivé, cache le stackpanel latéral gauche</li> </ul>  | toggle_Menu    | OnClick   |
| <ul style="list-style-type: none"> <li>- Vérifie qu'il y ait au moins un médecin généraliste enregistré</li> <li>- S'il y a au moins un médecin généraliste enregistré : <ul style="list-style-type: none"> <li>o Ferme la page « accueil » et ouvre la page « register patient »</li> </ul> </li> <li>- S'il n'y a pas de médecin généraliste enregistré : <ul style="list-style-type: none"> <li>o Retourne un message indiquant que l'opération est impossible</li> </ul> </li> </ul>   | btn_ajoutPat   | OnClick   |
| <ul style="list-style-type: none"> <li>- Ferme la page « accueil » et ouvre la page « register medecin »</li> </ul>  | btn_ajoutMed   | OnClick   |
| <ul style="list-style-type: none"> <li>- Vérifie qu'il y ait au moins un patient enregistré</li> <li>- S'il n'y a aucun patient : <ul style="list-style-type: none"> <li>o Indique qu'il n'y a aucun patient enregistré</li> </ul> </li> <li>- S'il y a au moins un patient enregistré : <ul style="list-style-type: none"> <li>o Affiche « Can_Recherche » et s'assure que les autres canvas sont cachés (« Can_Profil » et « Can_Stat »)</li> <li>o Affiche les éléments de « Can_Recherche »</li> <li>o Cache « img_Logo »</li> </ul> </li> </ul> | btn_rechercher | OnClick   |
| <ul style="list-style-type: none"> <li>- Affiche « Can_Profil » et s'assure que les autres canvas sont cachés (« Can_Recherche » et « Can_Stat »)</li> <li>- Affiche les éléments de « Can_Profil »</li> <li>- Cache « img_Logo »</li> </ul>   | btn_profil     | OnClick   |
| <ul style="list-style-type: none"> <li>- Affiche « Can_Stat » et s'assure que les autres canvas sont cachés (« Can_Recherche » et « Can_Profil »)</li> <li>- Affiche les éléments de « Can_Stat »</li> <li>- Cache « img_Logo »</li> </ul>   | btn_stat       | OnClick   |
| <ul style="list-style-type: none"> <li>- Ferme la page « accueil » et ouvre la page « login »</li> </ul>   | btn_deco       | OnClick   |
| <ul style="list-style-type: none"> <li>- Bouton fermant le programme</li> </ul>  | btn_quitter    | OnClick   |
| <ul style="list-style-type: none"> <li>- TextBlock affichant le n° INAMI si la personne est connectée</li> </ul>   | txt_username   | /         |

- TextBlock affichant le nom et le prénom du médecin si celui-ci est enregistré dans la base de données

- |   |              |         |
|---|--------------|---------|
| - Vérifie si les informations entrées vont ressortir au moins un enregistrement                             | btn_rech     | Onclick |
| - Si la recherche aboutit :   |              |         |
| o Ferme la page « accueil » et ouvre la page « patient recherche »  |              |         |
| o Transmet les informations correspondant à la requête à la page « patient recherche »                      |              |         |
| - Si la recherche n'aboutit pas   |              |         |
| o Un message indique la recherche n'aboutit à rien  |              |         |
|   |              |         |
| - Ferme la page « accueil » et ouvre la page « patient recherche »  | btn_affTout  | Onclick |
| - Transmet tous les enregistrements présents dans la table « FichePatient » à la page « patient recherche » |              |         |
| - Efface les données entrées dans les TextBox   | btn_cancel   | Onclick |
| - Si activé, affiche une vidéo et la lance.   | toggle_Video | Onclick |
| - Si désactivé, cache la vidéo et la met sur pause.   |              |         |

## REGISTER PATIENT

| TÂCHE  | OBJET         | ÉVÉNEMENT |
|--|---------------|-----------|
| - Ferme la page « register patient » et ouvre la page « accueil »  | btn_precPat   | Onclick   |
| - Permet de cacher les informations saisies par l'utilisateur et affiche de nouvelles informations à saisir. | btn_suivPat   | Onclick   |
| - Permet de cacher les informations saisies par l'utilisateur et affiche de nouvelles informations à saisir  | btn_precPat2  | Onclick   |
| - Vérifie si toutes les informations saisies sont correctes  | btn_submitPat | Onclick   |
| - Si elles sont correctes, il inscrit le patient dans la base de données                                     |               |           |
| - Si elles sont correctes, ferme la page « register patient » et ouvre la page « accueil »                   |               |           |
| - Si elles sont correctes, affiche un message indiquant la réussite de l'enregistrement                      |               |           |
| - Si elles sont incorrectes, affiche en rouge les informations invalides/vides                               |               |           |
| - Affiche un message indiquant que l'enregistrement n'a pas été effectué                                     |               |           |



## REGISTER MEDECIN

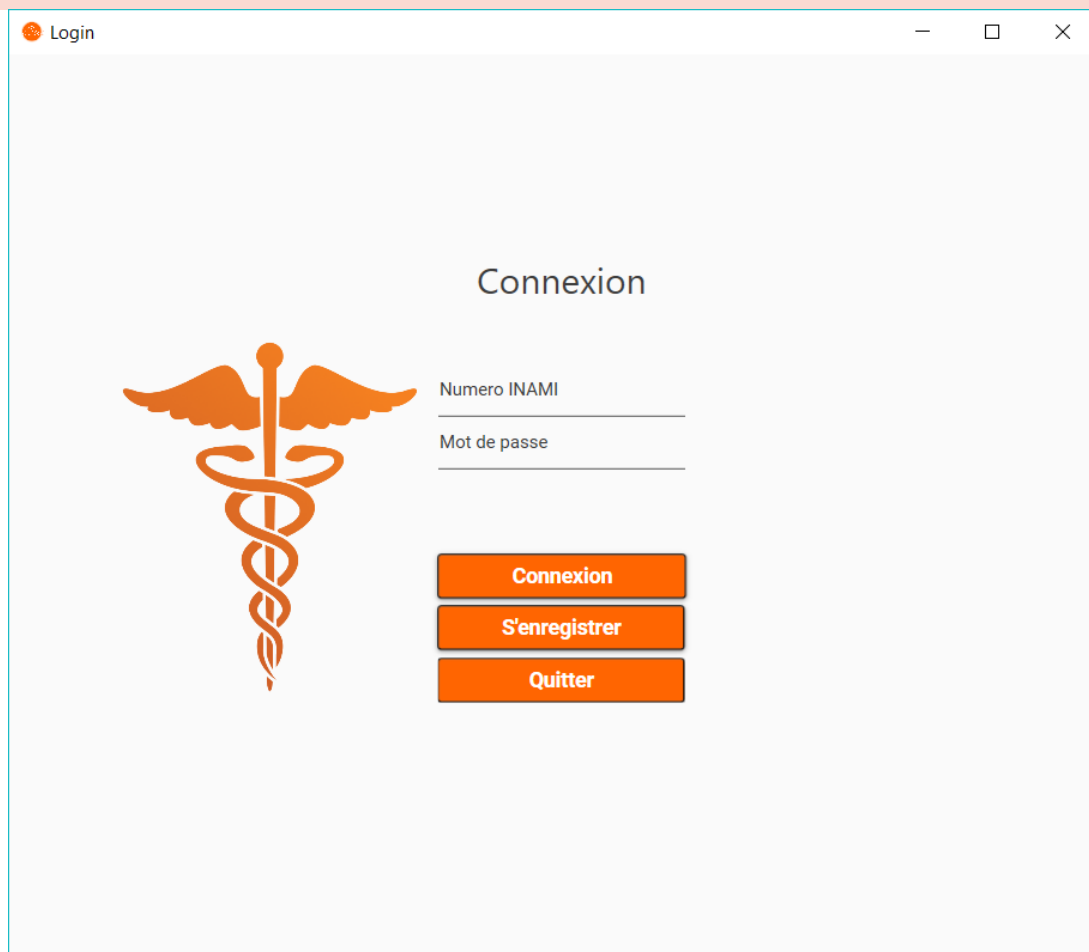
| TÂCHE  | OBJET      | ÉVÉNEMENT |
|--|------------|-----------|
| - Ferme la page « register medecin » et ouvre la page « accueil »                          | btn_prec   | Onclick   |
| - Vérifie si toutes les informations saisies sont correctes                                | btn_submit | Onclick   |
| - Si elles sont correctes, il inscrit le patient dans la base de données                   |            |           |
| - Si elles sont correctes, ferme la page « register medecin » et ouvre la page « accueil » |            |           |
| - Si elles sont correctes, affiche un message indiquant la réussite de l'enregistrement    |            |           |
| - Si elles sont incorrectes, affiche en rouge les informations invalides/vides             |            |           |
| - Affiche un message indiquant que l'enregistrement n'a pas été effectué                   |            |           |

## PATIENT RECHERCHE

| Tâche  | Objet           | Événement |
|--|-----------------|-----------|
| <ul style="list-style-type: none"> <li>- Vérifie qu'il y ait au moins un médecin généraliste enregistré</li> <li>- S'il y a au moins un médecin généraliste enregistré : <ul style="list-style-type: none"> <li>o Ferme la page « patient recherche » et ouvre la page « register patient »</li> </ul> </li> <li>- S'il n'y a pas de médecin généraliste enregistré :</li> <li>- Retourne un message indiquant que l'opération est impossible</li> </ul>   | btn_AddPat      | Onclick   |
| <ul style="list-style-type: none"> <li>- Prend en compte les nouvelles données pour l'utilisateur sélectionné et les modifie dans la base de données.</li> </ul>   | btn_ModifierPat | Onclick   |
| <ul style="list-style-type: none"> <li>- Si un utilisateur est sélectionné, un seul clic suffit pour qu'il soit supprimé.</li> </ul>   | btn_SuppPat     | Onclick   |
| <ul style="list-style-type: none"> <li>- Ferme la page « patient recherche » et ouvre la page « accueil »</li> </ul>   | btn_Ret         | Onclick   |
| <ul style="list-style-type: none"> <li>- Bouton fermant le programme</li> </ul>  | btn_Quit        | Onclick   |
| <ul style="list-style-type: none"> <li>- ListView affichant certaines caractéristiques pour chaque patient sélectionné (NIS, nom, prénom, sexe, allergie, tiers payant, dernier hôtel visité, mutuelle, le nom du médecin généraliste)</li> <li>- Quand un enregistrement est sélectionné : <ul style="list-style-type: none"> <li>o Le bouton « btn_ModifierPat » devient cliquable</li> <li>o Le bouton « btn_SuppPat » devient cliquable</li> <li>o Affiches « Can_Modif » et ses éléments</li> </ul> </li> </ul> | LV_Patient      | /         |
| <ul style="list-style-type: none"> <li>- Canevas contenant les éléments pouvant être modifiés par l'utilisateur connecté : <ul style="list-style-type: none"> <li>o « txtB_NISDelete »</li> <li>o « txtB_PreModif »</li> <li>o « comboB_SexeModif »</li> <li>o « comboB_TPModif »</li> <li>o « comboB_DHModif »</li> </ul> </li> <li>- « txtB_NISDelete » ne peut pas être modifié, mais il est affiché</li> </ul>   | Can_Modif       | /         |

## ÉCRANS DE L'APPLICATION

## CONNEXION



The screenshot shows a web browser window with the title 'Login'. The main heading is 'Connexion'. On the left is a large orange caduceus icon. To its right are two input fields: 'Numero INAMI' and 'Mot de passe'. Below these fields are three orange buttons: 'Connexion', 'S'enregistrer', and 'Quitter'.

Figure 5 Connexion

L'utilisateur peut choisir de se connecter à l'application via son numéro d'INAMI.

S'il n'a pas de compte enregistré, il doit en créer un. S'il veut quitter l'application, il n'a qu'à cliquer sur le bouton quitter.

## S'ENREGISTRER



The screenshot shows a web browser window titled 'Register'. The main heading is 'CREATION DE COMPTE : Utilisateur'. On the right side, there is a large orange circular logo with a white mountain range and the word 'Hillth' in a white script font. Below the heading, there are two input fields: 'N° INAMI' and 'Mot de passe'. At the bottom of the form, there are two orange buttons: 'Retour' and 'Enregistrement'.

Figure 6 Enregistrement

Si l'utilisateur veut créer un compte, il se retrouvera sur cette page.

Il peut soit :

- Valider son compte
- Appuyer sur retour. Il se retrouvera sur l'écran de connexion.

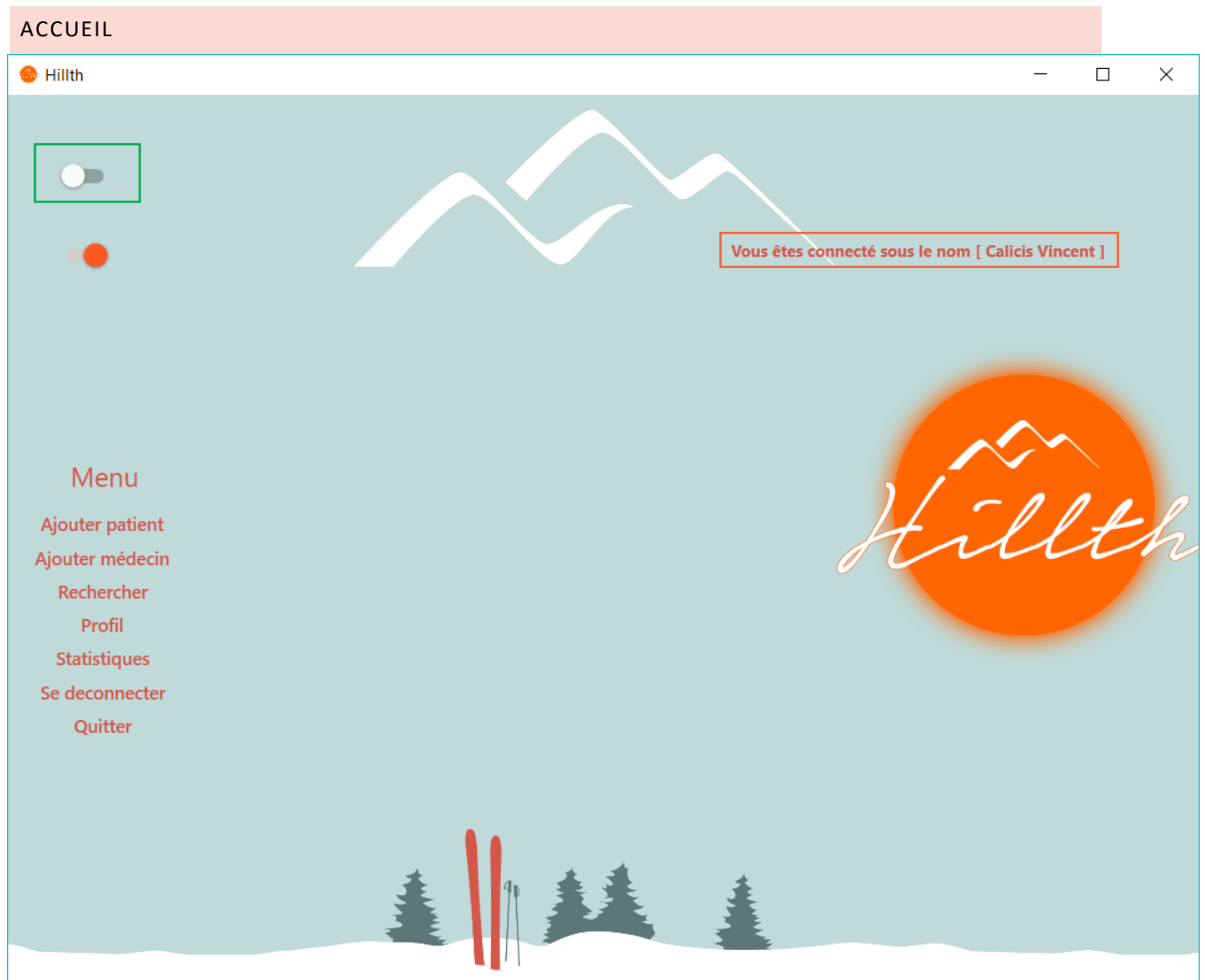


Figure 7 Accueil

L'utilisateur a réussi à se connecter.

On peut voir qu'il a un profil existant en tant que médecin, étant donné que son nom est affiché.

À partir de cet écran, il peut accéder à tous les éléments disponibles dans le programme.

S'il clique sur le « toggle » encadré en vert, une vidéo apparaîtra, le sujet portant sur les médecins généralistes.

Les boutons « se déconnecter » et « quitter » mènent respectivement à la page de connexion et à la fermeture du programme.

---

AFFICHER PROFIL

Si nous cliquons sur profil deux cas son possible :

- Un médecin correspond à l'utilisateur, les informations sont affichées.
- Personne ne correspond à l'utilisateur, un message l'indique et rien n'est affiché.

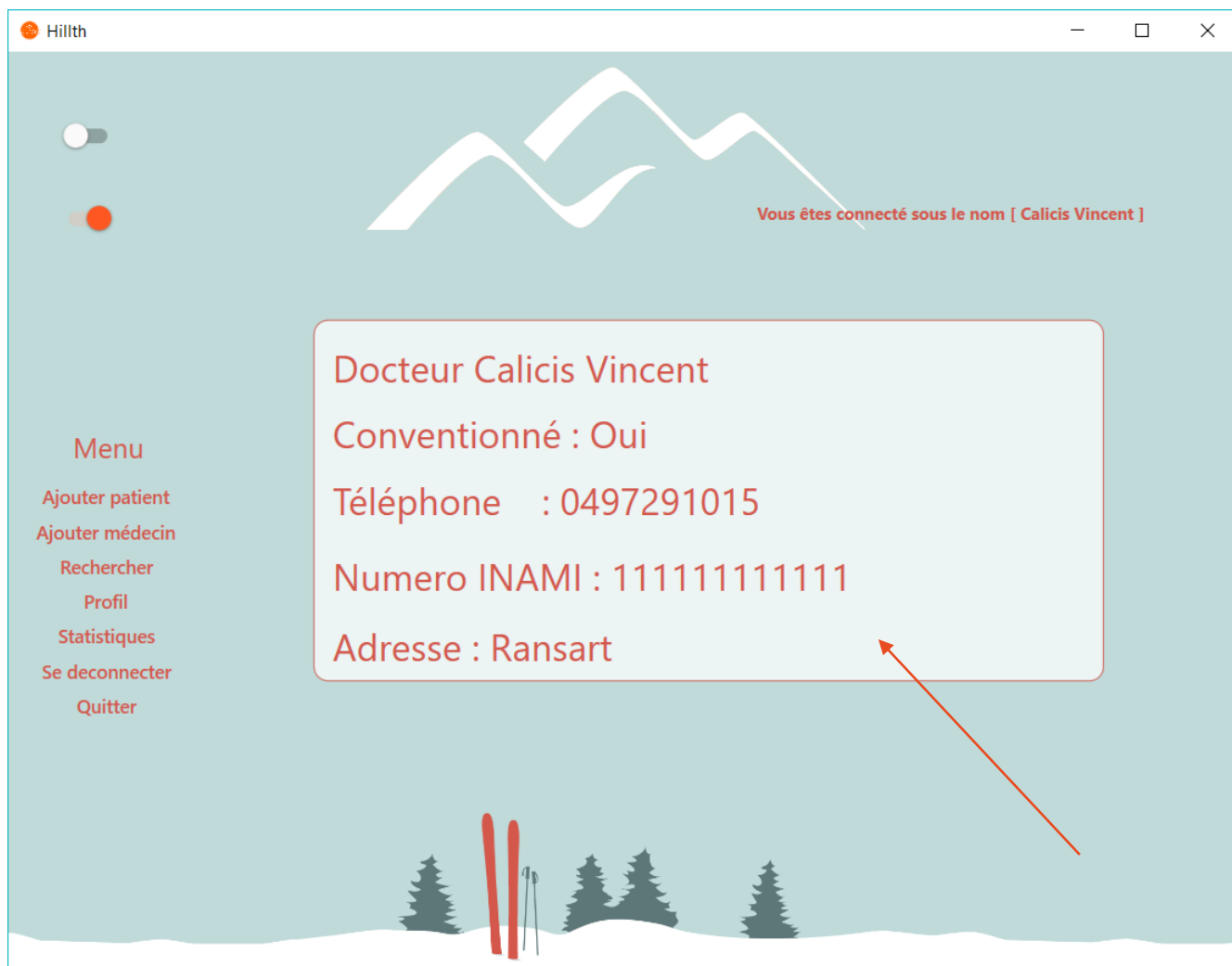


Figure 8 Profil

## RECHERCHER

Hillth

Vous êtes connecté sous le nom [ Calicis Vincent ]

**Rechercher les patients par :**

Nom

Prenom

Numero patient

Mutuelle

Dernier hôpital

Tout afficher

Rechercher

Reset

Menu

- Ajouter patient
- Ajouter médecin
- Rechercher
- Profil
- Statistiques
- Se deconnecter
- Quitter

Figure 9 Rechercher

Si nous cliquons sur rechercher, deux cas sont possibles :

- Au moins un patient est présent et cet écran apparaît.
- Il n'y a pas de patients enregistrés et un message apparaît.

## STATISTIQUE

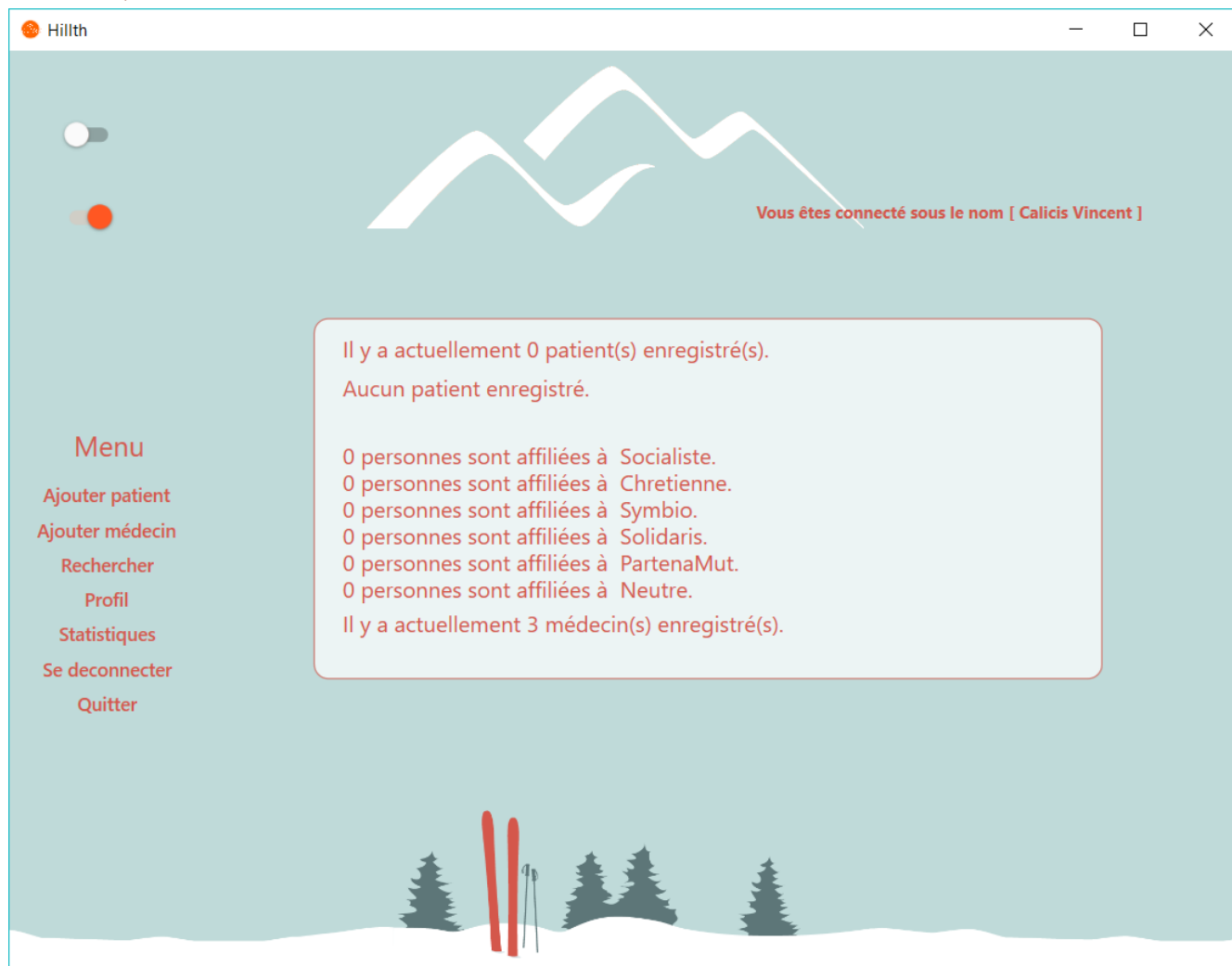


Figure 10 Statistique

Si l'utilisateur clique sur « statistiques » il verra cet affichage.

Celui-ci n'est pas figé, il évolue selon les informations entrées dans la base de données.



## AJOUTER PATIENT



The screenshot shows a web application window titled "RegisterPatient" with standard window controls (minimize, maximize, close) in the top right corner. The main heading is "AJOUT D'UN PATIENT" in a large, bold, red font. Below it, the subtitle "Informations personnelles" is in a smaller, regular red font. The form contains several input fields, each with a red label: "Nom", "Prénom", "Date naissance" (with a calendar icon), "Sexe" (with a dropdown arrow and the letter "F" selected), "Médecin généraliste" (with a dropdown arrow), "N° téléphone patient", and "Adresse". At the bottom of the form, there are two red buttons: "Retour" and "Suivant".

Figure 11 Ajout patient 1

L'utilisateur ajoute des patients via ces deux écrans.

Le bouton retour mène à l'écran d'accueil.

Le bouton suivant mène à la seconde fenêtre d'ajout patient.

Le bouton précédent mène à la première fenêtre d'ajout patient.

RegisterPatient

# AJOUT D'UN PATIENT

Information médicale

Mutuelle

Dernier hôpital visité

Tiers payant

☐ Oui ☐ Non

NIS

Allergies

Précédent

S'enregistrer

Figure 12 Ajout patient 2

## AJOUT MÉDECIN

## CREATION DE COMPTE : Médecin

|                         |                      |            |                          |
|-------------------------|----------------------|------------|--------------------------|
| Nom                     | <input type="text"/> | Prénom     | <input type="text"/>     |
| Date naissance          | <input type="text"/> | Sexe       | F <input type="text"/>   |
| N° INAMI                | <input type="text"/> | Convention | Non <input type="text"/> |
| Spécialisation          | <input type="text"/> |            |                          |
| N° de téléphone         | <input type="text"/> |            |                          |
| Adresse professionnelle | <input type="text"/> |            |                          |



Retour

S'enregistrer

Figure 13 Ajout d'un médecin

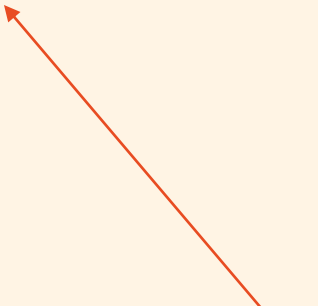
L'utilisateur ajoute des médecins via cette fenêtre.

Le bouton retour mène à l'écran d'accueil.

## AFFICHER PATIENT

## AFFICHER TOUS LES PATIENTS ENREGISTRÉS

| NIS         | Nom       | Prénom  | Sexe | Allergie | Tiers payant | Dernier hôpital | Mutuelle   | Medecin généraliste |
|-------------|-----------|---------|------|----------|--------------|-----------------|------------|---------------------|
| 12454879875 | Mousty    | Adrien  | H    | Neurofen | Oui          | St-Luc-UCL      | Socialiste | Nonon               |
| 14548798756 | Sebastien | Patrick | H    | Musique  | Oui          | CHU Charleroi   | Chretienne | Calicis             |
| 14549876565 | DiStasio  | Anthony | H    | Chats    | Oui          | St-Luc-UCL      | Chretienne | Calicis             |
| 14578793232 | Mousty    | Emry    | H    | Pollen   | Oui          | St-Luc-UCL      | Chretienne | Calicis             |
| 14598786545 | Stefano   | Naomi   | F    | Aucun    | Oui          | CHU Charleroi   | Solidaris  | Calicis             |



Ajouter patient

Retour

Quitter

Nous affichons tous les patients actuellement enregistrés dans la base de données.

## AFFICHER LES PATIENTS CORRESPONDANT UNIQUEMENT AU FILTRE SÉLECTIONNÉ

| NIS         | Nom       | Prénom  | Sexe | Allergie | Tiers payant | Dernier hôpital | Mutuelle   | Medecin généraliste |
|-------------|-----------|---------|------|----------|--------------|-----------------|------------|---------------------|
| 14548798756 | Sebastien | Patrick | H    | Musique  | Oui          | CHU Charleroi   | Chretienne | Calicis             |
| 14598786545 | Stefano   | Naomi   | F    | Aucun    | Oui          | CHU Charleroi   | Solidaris  | Calicis             |

14598786545    Stefano    Naomi    \_\_\_\_\_    \_\_\_\_\_    \_\_\_\_\_

Ajouter patient

Modifier un patient

Supprimer un patient

Retour

Quitter

Ici, nous décidons de n'afficher que les personnes ayant visité le CHU de Charleroi.

De plus, l'utilisateur a cliqué sur « Naomi Stefano » afin de modifier un élément, ou de la supprimer de la base de données.

## NOUVEAUTÉS

## THEME (MATERIAL DESIGN IN XAML TOOLIK MASTER)

Ce thème permet de rendre l'application plus agréable visuellement.

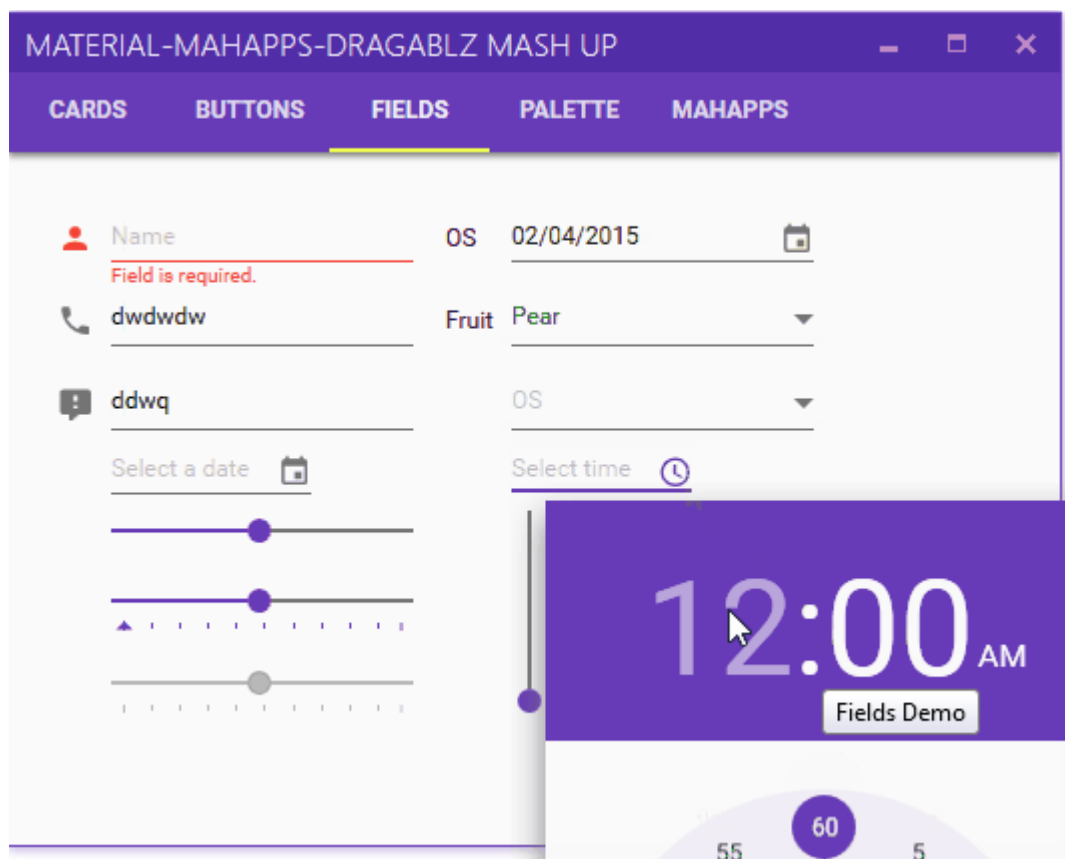


Figure 14 Thème

Ce thème est hautement personnalisable, nous pouvons, par exemple, modifier les couleurs dominantes.

Il change l'aspect d'énormément d'éléments dans l'application. Notamment l'apparence du textblock, textbox, datepicker, calendar, etc.

## DATEPICKER

Le « datepicker » est un moyen très intéressant pour saisir des dates. Il ne nécessite pas de vérifications vis-à-vis des dates sélectionnées, étant donné que celles-ci sont toujours valides.

```
<DatePicker x:Name="date_naissancePat"
    Canvas.Left="210.245"
    Canvas.Top="131"
    Height="51"
    Width="157.8"
    FontFamily="Muli"
    FontSize="26.667"
    DisplayDate="1992-12-16"
    Foreground="#FFD4574B"/>
```

Figure 16 Date Picker code

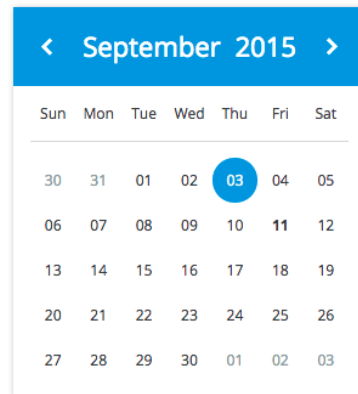


Figure 15 Date Picker

## INSERTION D'UNE VIDEO

Quand la personne clique sur le bouton lié au code ci-dessous, une vidéo sur le métier de médecin généraliste apparaît. S'il reclique sur ce même bouton la vidéo est cachée et est arrêtée.

```
<!-- Affiche la vidéo -->
<MediaElement x:Name="video_medecin"
    HorizontalAlignment="Left"
    Height="268"
    Margin="219,192,0,0"
    VerticalAlignment="Top"
    Width="576"
    Source="C:\Users\Adrien\Downloads\medecin.mp4"
    Visibility="Hidden"
    LoadedBehavior="Manual"
    UnloadedBehavior="Stop"/>
```

Figure 17 Code vidéo

## PROCÉDURE STOCKÉE

Ils intérêts des procédures stockées sont multiples :

1. **simplification** : code plus simple à comprendre
2. **rapidité** : moins d'informations sont échangées entre le serveur et le client
3. **performance** : économise au serveur l'interprétation de la requête car elle est précompilée
4. **sécurité** : les applications et les utilisateurs n'ont aucun accès direct aux tables, mais passent par des procédures stockées prédéfinies

Elles sont utilisées à la places de requête Linq classique.

```
CREATE PROCEDURE AfficherLibelHop
AS
SELECT H.LibelHopital
FROM Hopital AS H
```

Figure 19 Procédure stockée pour afficher le nom des hôpitaux

```
var querHop = from h
               in HDB.Hopital
               select h;
```

Figure 18 Requête Linq pour afficher les informations sur les hôpitaux



## TRAVAIL ADDITIONNEL

Cette partie n'était pas nécessaire, mais je tenais tout de même à la faire afin de m'entraîner. Cela me permet de faire un peu plus d'exercices en analyse étant donné que je ne suis pas très bon dans ce domaine.

J'ai tenté de découper le diagramme en 4 parties.

1. L'inscription.
2. Ce qui est lié au(x) médecin(s).
3. Ce qui est lié au(x) patient(s).
4. Les statistiques qui, elles, sont issues des médecins et des patients.

## DIAGRAMME DES USE CASE PAR ITERATION

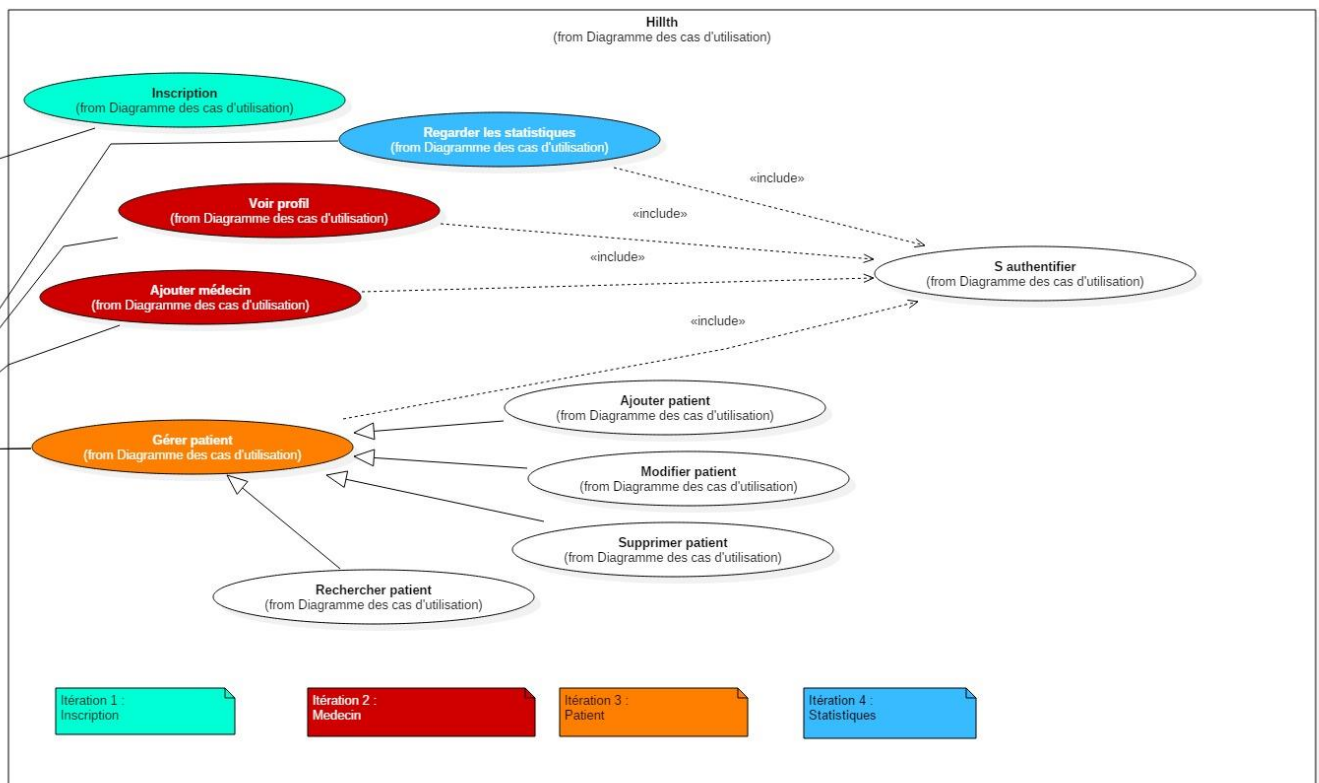


Figure 20 Diagramme Use Case par itération

## DIAGRAMME DE CLASSE PARTIEL 1

## Diagramme de classe partiel 1

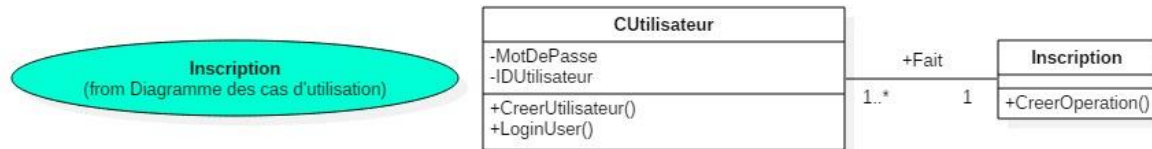


Figure 21 Diagramme de classe partiel 1

## DIAGRAMME DE CLASSE PARTIEL 2

## Diagramme de classe partiel 2

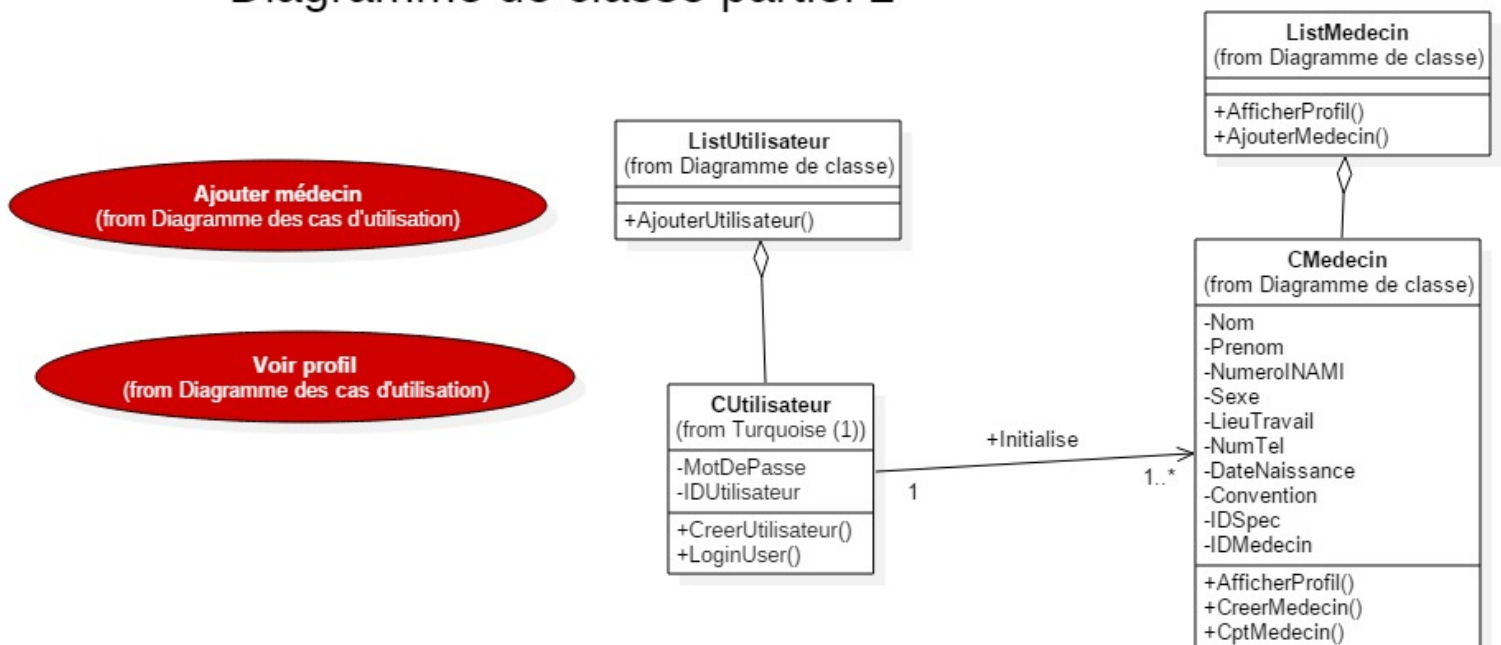


Figure 22 Diagramme de classe partiel 2

## DIAGRAMME DE CLASSE PARTIEL 3

## Diagramme de classe partiel 3

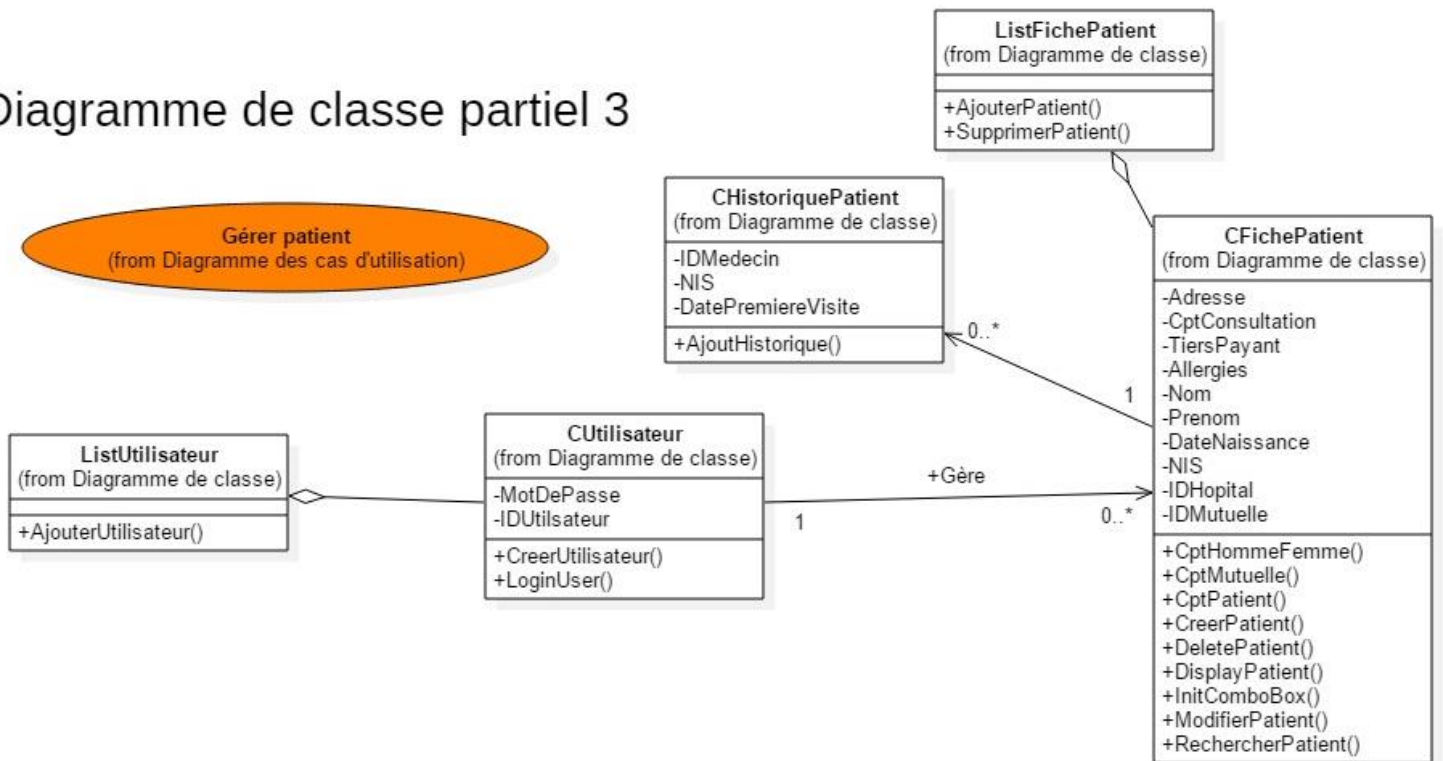


Figure 23 Diagramme de classe partiel 3

## DIAGRAMME DE CLASSE PARTIEL 4

## Diagramme de classe partiel 4

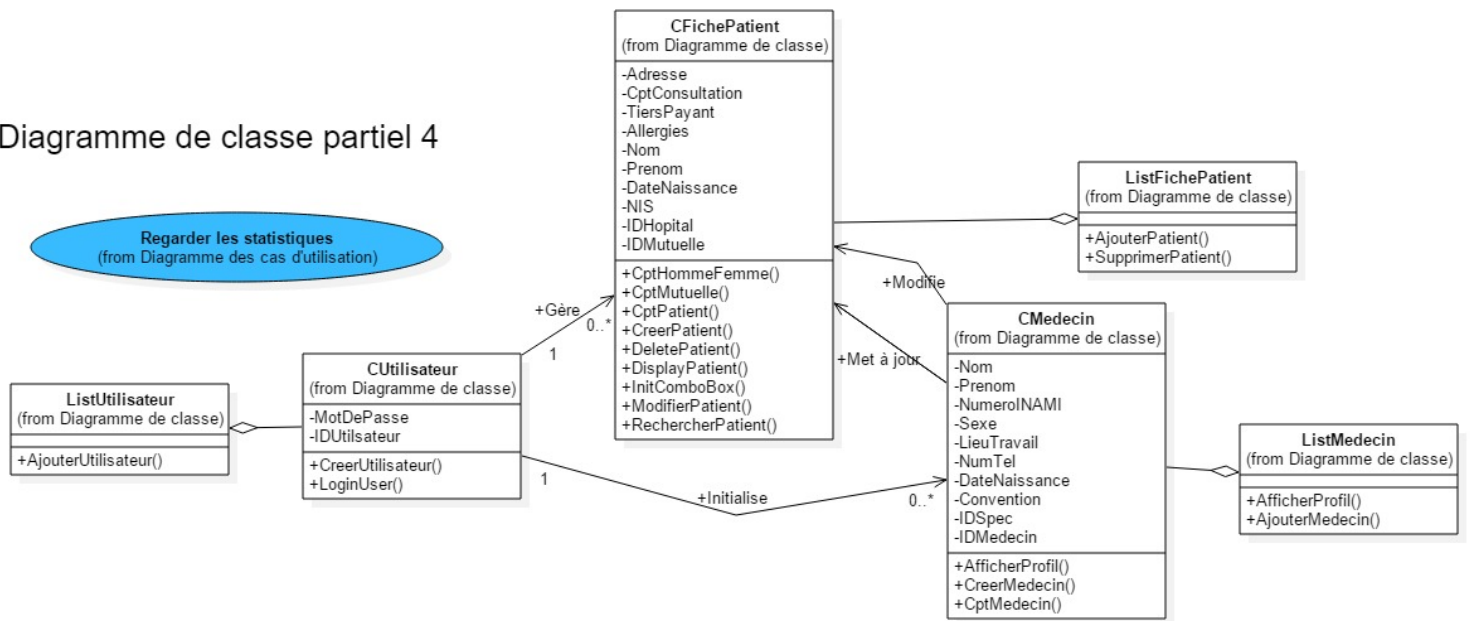


Figure 24 Diagramme de classe partiel 4

Ici les méthodes pour afficher les statistiques se trouvent dans les classes médecin et patient.

## CONCLUSION

Ce travail personnel fut intéressant à développer. Bien qu'il fasse un peu doublon avec le projet de groupe (analyse – C# - Anglais), celui-ci m'a forcé à faire des tâches qui ne correspondent pas forcément, tel que les schémas de pré-développement.

Malheureusement, je déplore le fait de n'avoir pas eu assez de temps pour implémenter tout ce qui me faisait envie. Le fait de devoir gérer les cours ainsi que 3 gros projets ne fut pas quelque chose d'aisé. Mais globalement, je suis relativement satisfait du résultat.

Je dois aussi avouer que le travail de groupe est, selon moi, plus abouti que celui-ci, car nous lui avons octroyé plus de temps.

*« Chaque bonne réalisation, grande ou petite, connaît ses périodes de corvées et de triomphes; un début, un combat et une victoire. »*

## TABLE DES ILLUSTRATIONS

|  |    |
|--|----|
| Figure 1 Schéma conceptuel   | 7  |
| Figure 2 Diagramme de contexte statique                                | 8  |
| Figure 3 Diagramme des use case complet                                | 9  |
| Figure 4 Diagramme de classe   | 10 |
| Figure 5 Connexion   | 18 |
| Figure 6 Enregistrement  | 19 |
| Figure 7 Accueil   | 20 |
| Figure 8 Profil  | 21 |
| Figure 9 Rechercher  | 22 |
| Figure 10 Statistique  | 23 |
| Figure 11 Ajout patient 1  | 24 |
| Figure 12 Ajout patient 2  | 25 |
| Figure 13 Ajout d'un médecin   | 26 |
| Figure 14 Thème  | 29 |
| Figure 15 Date Picker  | 30 |
| Figure 16 Date Picker code   | 30 |
| Figure 17 Code vidéo   | 31 |
| Figure 18 Requête Linq pour afficher les informations sur les hôpitaux | 31 |
| Figure 19 Procédure stockée pour afficher le nom des hôpitaux          | 31 |
| Figure 20 Diagramme Use Case par itération                             | 32 |
| Figure 21 Diagramme de classe partiel 1                                | 33 |
| Figure 22 Diagramme de classe partiel 2                                | 33 |
| Figure 23 Diagramme de classe partiel 3                                | 34 |
| Figure 24 Diagramme de classe partiel 4                                | 35 |