

Projet C#

Gestion d'une infirmerie

GRANIT Dimitri
BRUNET Alexandre
FAVERNAY Marcus

Sommaire

ce sommaire est cliquable ! interactif même!

I - Ressources

I.1) Documentation

I.2) Outils

II - Spécifications fonctionnelles

II.1) Contexte

II.2) Cas d'utilisation

II.3) Règles de gestion

II.4) Dictionnaire de données

III - Spécifications techniques

III.1) Maquettage

III.2) Modélisation

III.3) Schéma relationnel

III.4) Scripts SQL

III.4.1) Structure de la base de données

III.4.2) Jeu d'essai

IV - Cahier de recettes

V - Guide utilisateur

VI - Répartition des tâches

VI.1) Dossier de spécifications fonctionnelles

VI.2) Dossier de spécifications techniques

VI.1) Sprint 1

VI.2) Sprint 2

VI.3) Sprint 3

VI.4) Sprint 4

VI.5) Sprint 5

I - Ressources

1) Documentation

[Sujet](#)

[Sprints](#)

[Organisation](#)

[Exemples de cas d'utilisations](#)

[Exemples de cahier de recette](#)

2) Outils

Modélisation : *JMerise*

Maquettage : *Figma*

Versionning : *Git/Github*

Bugtracker : *Redmine*

Communication : *Discord*

Développement : Visual Studio

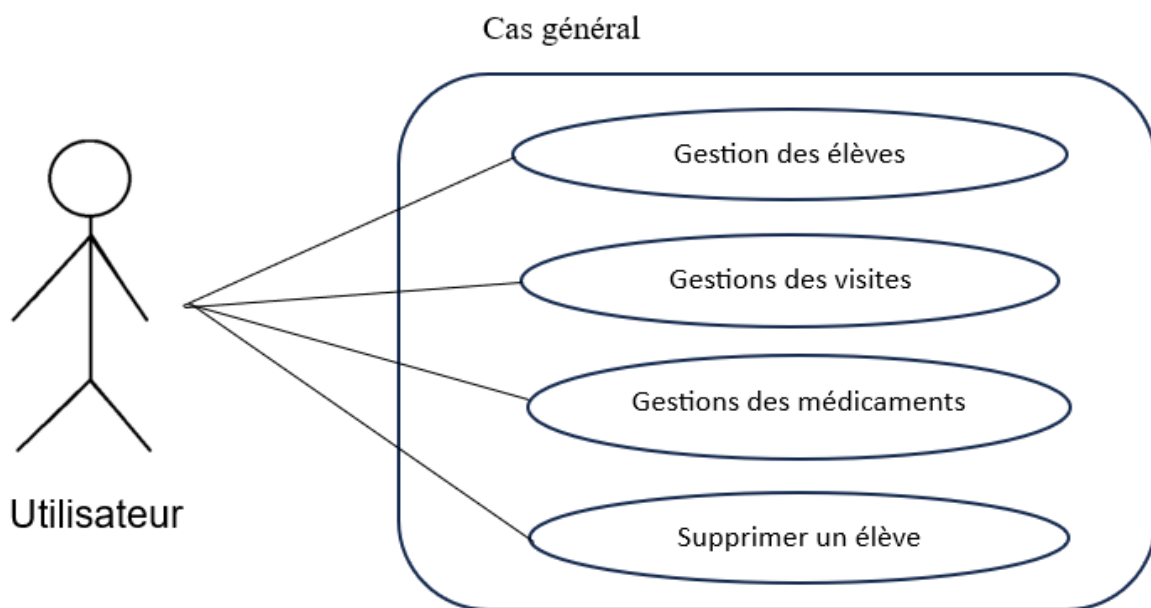
Base de données : Microsoft SQL Server

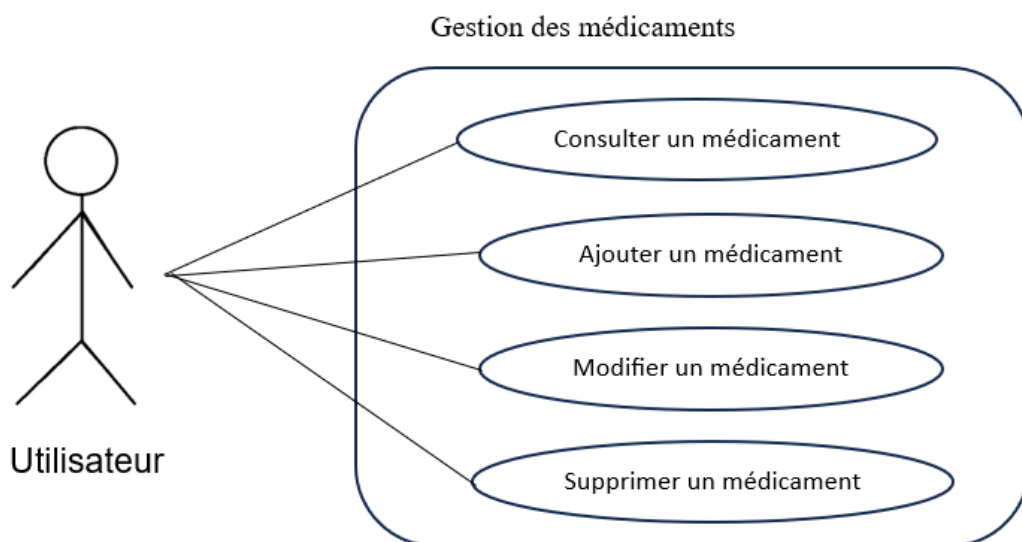
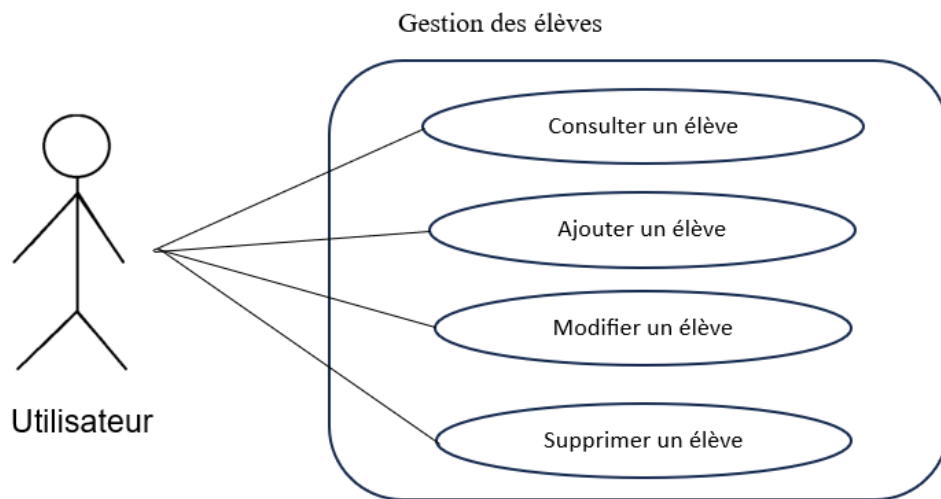
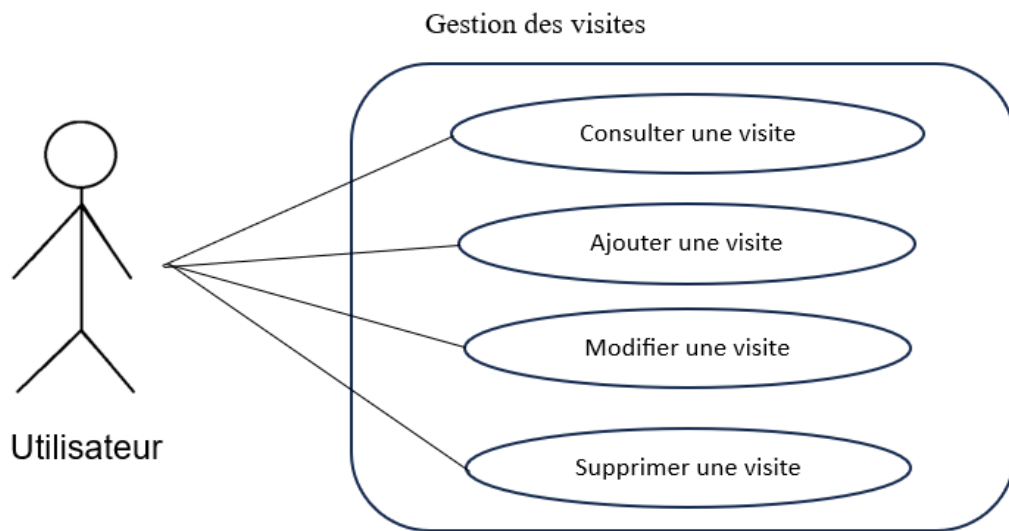
II - Spécifications fonctionnelles

1) Contexte:

Un lycée privé souhaite réaliser, pour son infirmerie, une application de gestion permettant de gérer les dossiers santé des élèves. Les élèves, les visites à l'infirmerie et les médicaments devront être gérés. Le lycée fait appel à votre société, qui confie le développement de l'application à l'équipe de développeurs dont vous faites partie. Aucune application n'existe, vous devez donc créer l'application dans sa globalité.

2) Cas d'utilisations





Infirmière :

Nom	Utilisateur
Type	Profil utilisateur, administrateur
Rôle	Utiliser l'application
Description	L'utilisateur utilise l'application pour consulter, ajouter, modifier et supprimer un élève et/ou une visite et/ou un médicament.

Détail des sous fonctionnalités :

Gestion des élèves :

Cas	Description
Consulter un élève	L'utilisateur peut consulter la liste des élèves
Ajouter un élève	L'utilisateur peut enregistrer un nouvel élève
Modifier un élève	L'utilisateur peut modifier les informations de l'élèves
Supprimer un élève	L'utilisateur peut supprimer un élève

Gestion des visites :

Cas	Description
Consulter une visite	L'utilisateur peut consulter la liste des visites
Ajouter une visite	L'utilisateur peut enregistrer une nouvelle visite
Modifier une visite	L'utilisateur peut modifier les informations de la visite
Supprimer une visite	L'utilisateur peut supprimer une visite

Gestion des médicaments :

Cas	Description
Consulter un médicament	L'utilisateur peut consulter la liste des médicaments
Ajouter un médicament	L'utilisateur peut enregistrer un nouveau médicament
Modifier un médicament	L'utilisateur peut modifier les informations d'un médicament
Supprimer un médicament	L'utilisateur peut supprimer un médicament

3) Règles de gestion:

ÉLÈVES:

- Un élève est caractérisé par identifiant, un nom, un prénom, une date de naissance, son numéro de téléphone portable, une classe, un numéro de téléphone d'un parent, s'il dispose ou non d'un tiers temps aux examens et d'un commentaire santé libre.
- Un élève fait partie d'**1** classe
- Un élève peut se rendre à **plusieurs** reprises à une visite.

VISITES:

- Une visite est caractérisée par un identifiant, un lieu, une date, une heure de début, une heure de fin, d'une quantité de médicament, un motif de visite et un commentaire. Des informations comme le fait que l'élève soit rentré chez soi, qu'il soit à l'hôpital et/ou que les parents ont été prévenu doivent être entrées.
- Une visite concerne **1** élève.
- Une visite peut être amenée à ne délivrer **aucun ou plusieurs** médicaments.

MÉDICAMENTS:

- Un médicament est caractérisé par un identifiant et un nom
- Un médicament peut être délivré lors d'**1 ou plusieurs** visites

CLASSE :

- Une classe est caractérisée par un identifiant et libellé
- Une classe est attribuée à **1 ou plusieurs** élèves

4) Dictionnaire de données

Entité UTILISATEUR

utilisateur_login	Identifiant d'un utilisateur	#str
utilisateur_mdp	Mot de passe de l'utilisateur	str

Entité ELEVE

eleve_id	Identifiant unique d'un élève	#int
eleve_nom	Nom de l'élève	char
eleve_prenom	Prenom de l'eleve	char
eleve_naiss	Date de naissance de l'élève	date
eleve_port	N° de portable de l'élève	int
eleve_parent_port	N° de portable d'un parent de l'élève	int
eleve_classe	Identifiant de la classe de l'élève	int
eleve_tiers_temps	L'élève possède un tiers temps Oui/Non	bool
eleve_comm_sante	Commentaire santé de l'élève (facultatif)	char -null

Entité VISITE

visite_id	Identifiant unique d'une visite	#int
visite_eleve	Identifiant unique de l'élève	int
visite_medic	Identifiant unique du médicament prescrit (facultatif)	int -null
visite_medic_qte	Quantité de médicament prescrite	int -null
visite_date	Date de la visite	date
visite_heure_arrivee	Heure d'arrivée	time
visite_heure_depart	Heure de départ	time
visite_motif	Motif de la visite	char
visite_comm	Commentaire de l'infirmier(ère) (facultatif)	char -null
visite_parents_prev	Parents de l'élève prévenus Oui/Non	bool
visite_suite	Retour au domicile/Hôpital/Aucune	char -null
visite_utilisateur	infirmier(ère) ayant effectué la visite	str -null

Entité MEDICAMENT

medic_id	Identifiant unique du médicament	#int
medic_lib	Libellé du médicament	char

Entité CLASSE

classe_id	Identifiant unique de la classe	#int
classe_lib	Libellé de la classe	str

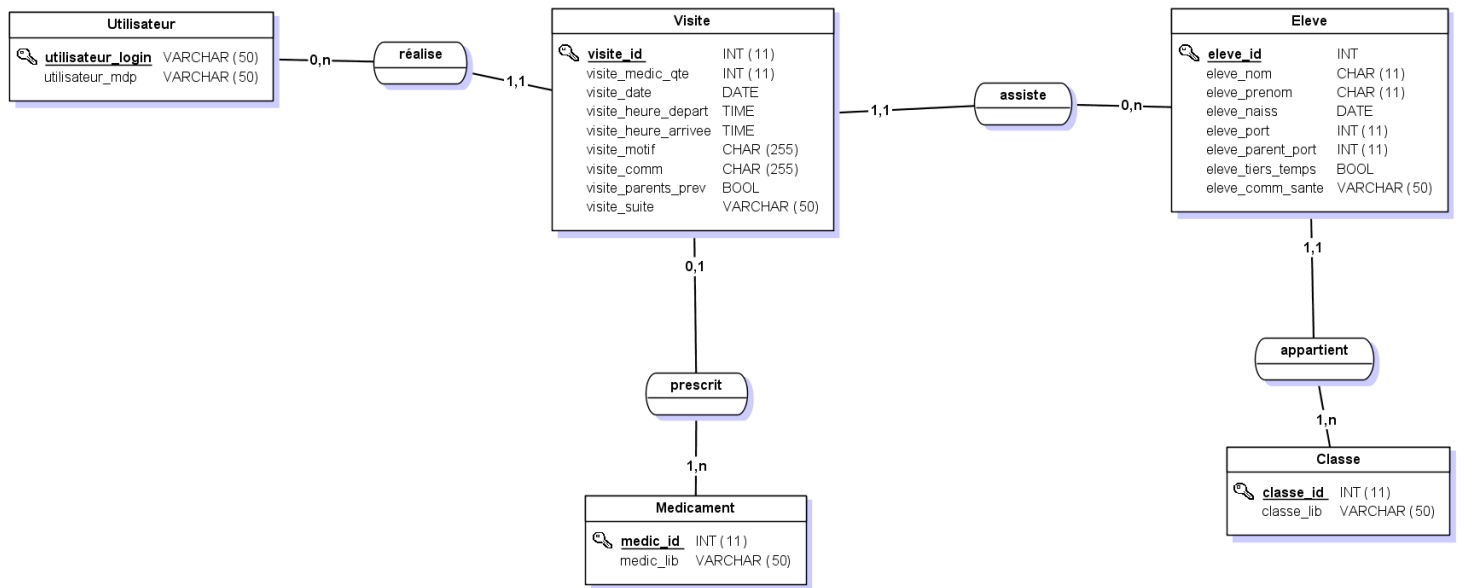
III - Spécifications techniques

1) Maquettage:

Wireframe effectué sur [Figma](#)
[Téléchargement \(.fig\)](#)



2) Modélisation



[Téléchargement du fichier mcd](#)

3) Schéma relationnel

UTILISATEUR (utilisateur_login, utilisateur_mdp)
PK : utilisateur_login

CLASSE (classe_id, classe_lib)
PK : classe_id

MEDICAMENT (medic_id, medic_lib)
PK : medic_id

ELEVE (eleve_id, eleve_nom, eleve_prenom, eleve_naiss, eleve_port, eleve_parent_port, eleve_classe, eleve_tier_temps, eleve_comm_sante)
PK : eleve_id
FK : eleve_classe référence classe_id dans **classe**

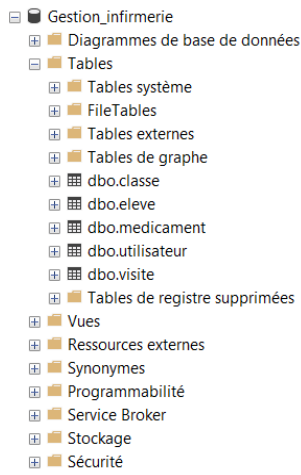
VISITE (visite_id, visite_eleve, visite_medic, visite_medic_qte, visite_date, visite_heure_depart, visite_heure_arrivee, visite_motif, visite_comm, visite_parents_prev, visite_suite, **visite_utilisateur**)
PK : visite_id
FK : visite_eleve référence **eleve_id** dans **eleve**
visite_medic référence **medic_id** dans **medicament**
visite_utilisateur référence **utilisateur_login** dans **utilisateur**

4) Scripts SQL

1) Structure de la Base de données

[Téléchargement \(.bak\)](#)

[Téléchargement \(.sql\)](#)



2) Jeu d'essai

[Téléchargement \(.bak\)](#)

[Téléchargement \(.sql\)](#)

	classe_id	classe_lib
1	1	2e 1
2	2	2e 2
3	3	2e 3
4	4	2e 4
5	5	2e 5
6	6	2e 6
7	7	1e 1
8	8	1e 2
9	9	1e 3
10	10	1e 4
11	11	1e 5
12	12	1e 6
13	13	t 1
14	14	t 2
15	15	t 3
16	16	t 4
17	17	t 5
18	18	t 6

	visite_id	visite_eleve	visite_medic	visite_medic_qte	visite_date	visite_heure_arrivee	visite_heure_depart	visite_motif	visite_comm	visite_parents_prev	visite_suite	visite_utilisateur
1	4	1	NULL	0	2023-10-18	00:13:00.0000000	00:13:30.0000000	sourdit� agrue	est devenue encore plus sourd	0	FALSE	admin
2	7	1	NULL	0	2023-10-18	00:14:00.0000000	00:14:30.0000000	dissidence	NULL	0	FALSE	admin
3	8	2	NULL	0	2023-09-18	00:12:00.0000000	00:12:30.0000000	mal de genou	aucune utilisation de cr�me	0	FALSE	admin
4	11	3	1	1	2023-09-11	00:10:30.0000000	00:10:45.0000000	mal de t�te	NULL	0	FALSE	admin
5	13	4	NULL	6	2023-09-10	00:14:30.0000000	00:15:00.0000000	diarr�e	NULL	0	FALSE	admin

	utilisateur_login	utilisateur_mdp
1	admin	admin

	eleve_id	eleve_nom	eleve_prenom	eleve_naiss	eleve_port	eleve_parent_port	eleve_classe	eleve_tier_temps	eleve_comm_sante
1	1	Zarb	Tom	2004-03-11	1122334455	1122334455	19	0	atteint de sourdit� au niveau de l'oreille gauche
2	2	Toursel	Rapha�l	2002-11-07	1122334455	1122334455	19	1	NULL
3	3	Favernay	Marcus	2004-08-26	1122334455	1122334455	19	0	NULL
4	4	Brunet	Alexandre	2001-07-05	1122334455	1122334455	19	0	NULL
5	7	Granit	Dimitri	2005-09-22	1122334455	1122334455	19	0	NULL
6	1003	Dupas	Lucas	2004-02-09	1122334455	1122334455	19	0	probl�me au niveau des intestins

IV - Cahier de recettes

V - Guide utilisateur

VI - Répartition des tâches

1) Dossier de spécifications fonctionnelles

Dimitri : Contexte, Dictionnaire de données

Marcus : Contexte, Règles de gestion

Alexandre : Cas d'utilisation, Sous Fonctionnalités

2) Dossier de spécifications techniques

Dimitri : Schéma relationnel, Structure BDD, Maquettage

Marcus : Maquettage, Jeu d'essai

Alexandre : MEA (+ vérification Schéma relationnel)

3) Sprint 1

4) Sprint 2

5) Sprint 3

6) Sprint 4

7) Sprint 5