MédiLabo Solutions

1. Présentation du projet

Le client Médilabo Solutions à besoin d'une solution pour l'aider à détecter le diabète de type 2. A cause d'un mauvais choix nutritionnels, les patients s'exposent à certaines maladies dont le diabète de type2. L'outil est destinés au médecin, il doit permettre :

-de renseigner les données démographiques (nom, age, etc.) .
-d'ajouter des notes.
-de générer un rapport indiquant la probabilité qu'un patient développe le diabète.

MédiLabo Solutions
We care for you

MédiLabo Solutions

2. Contraintes de développement

-L’application doit être découpé en plusieurs microservices, basés sur des projets Spring Boot.
-Ces microservices seront joignables via un microservice de type gateway développé avec Spring Cloud Gateway.
-Il faut une image Docker de chaque microservices de l’architecture.
-Les bases de données doivent toutes être normalisées (3NF).
-L’accès aux données des patients doit être sécurisé. Il faut mettre en place un système d’authentification avec Spring Security.
-Renseigner dans le fichier readme.md les suggestions d’actions à mener pour appliquer le Green Code au projet.

MédiLabo Solutions
We care for you

MédiLabo Solutions

3. User story

Le projet est découpé en 3 user Story. Chaque user Story correspond à un sprint.

MédiLabo Solutions

3.1. User story 1

En tant qu'organisateur, j'aimerais voir les informations personnelles de mes patients afin de vérifier leur identité lorsqu'ils arrivent en rendez-vous.
Mise à jour des informations personnelles.
Ajouter des informations personnelles des patients.
Les informations personnelles des patients sont :

- prénom
- nom
- date de naissance
- genre
- adresse postale (optionnel)
- numéro de téléphone (optionnel)

MédiLabo Solutions
We care for you

MédiLabo Solutions

3.1. User story 2

En tant que praticien, je veux voir l'historique des informations de mon patient afin d'avoir en tête les problèmes qu'il a eus par le passé.

En tant que praticien, je veux pouvoir ajouter une note d'observation à l'historique du patient afin de vérifier que mes conseils sont suivis d'une séance à l'autre.

MédiLabo Solutions

3.1. User story 3

En tant que praticien, je veux pouvoir consulter le risque de diabète pour un patient afin de le prévenir si sa santé est potentiellement en danger.

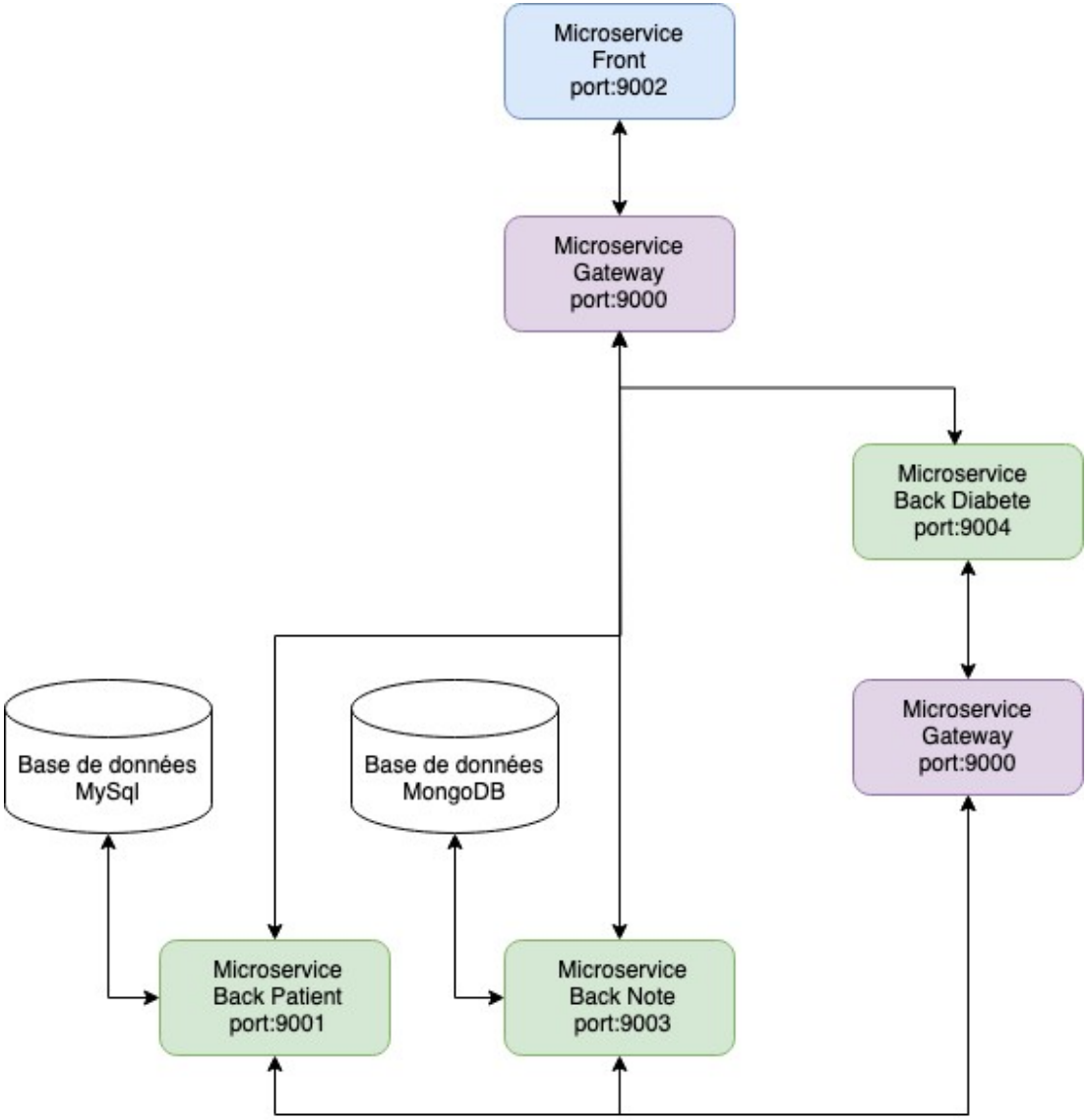
Un patient pourra avoir l'un des 4 niveaux de risque suivants :
- aucun risque (None) ;
- risque limité (Borderline) ;
- danger (In Danger) ;
- apparition précoce (Early onset).

Pour ce spring des règles sont fournies pour déterminer les niveaux de risque de diabète type 2.

MédiLabo Solutions
We care for you

MédiLabo Solutions

4. Architecture Microservices



MédiLabo Solutions
We care for you

MédiLabo Solutions

5. Bases de données

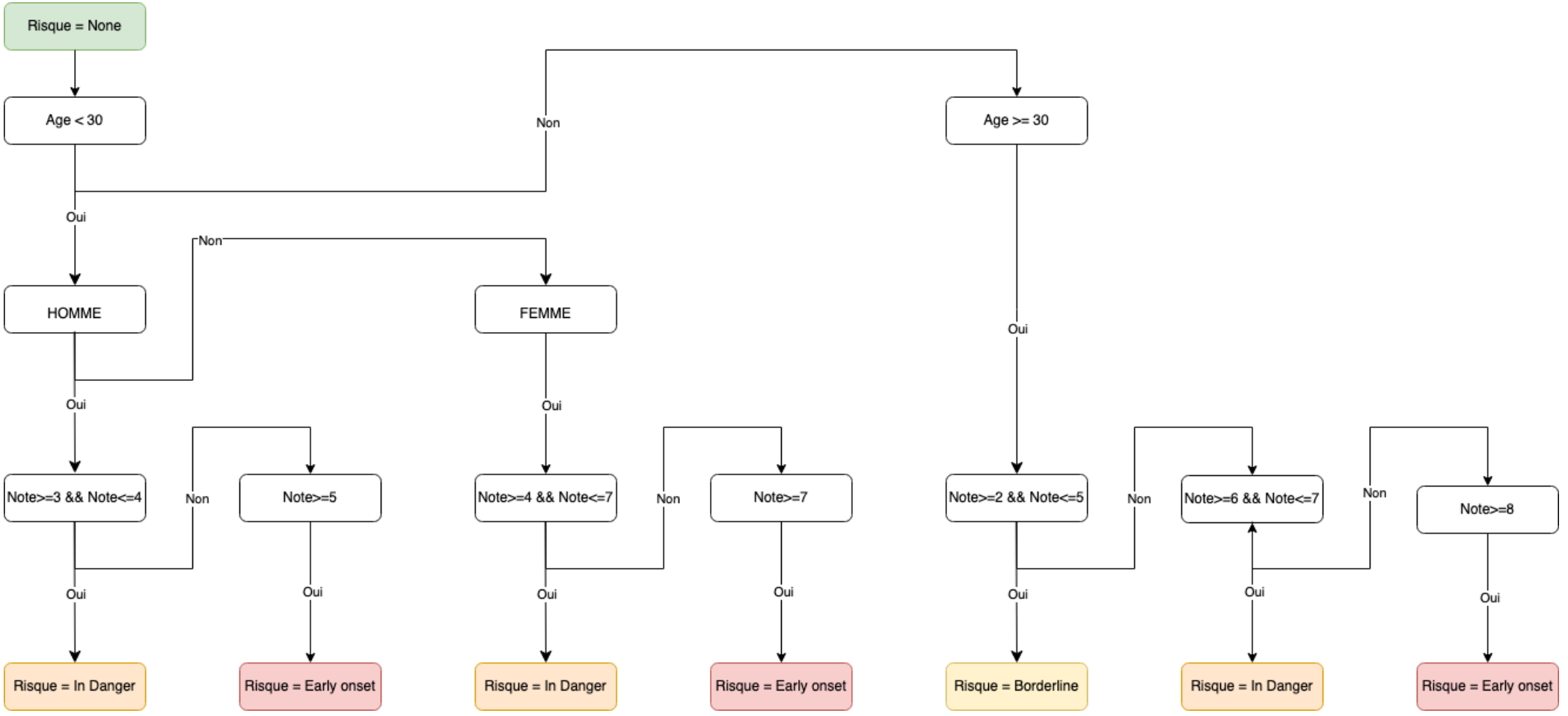
-Le microservice Back Patient est associé à une base de données sql Mysql ‘patient’.
-Le microservice Back Note est associé à une base de données nosql MongoDb ‘note’.

Patients	
patient_id	INT NOT NULL [PK]
adresse	VARCHAR(255)
date_de_naissance	DATETIME(6)
genre	VARCHAR(255)
nom	VARCHAR(255)
prenom	VARCHAR(255)
telephone	VARCHAR(255)

Notes	
_id	ObjectId
patid	String
note	String

MédiLabo Solutions
We care for you

Algorithme risque de diabète



MédiLabo Solutions

6. Green Code

Concernant le numérique, La fabrication des terminaux est la principale source d'impacts écologiques. Pour agir sur cette source, le dévelloppeur doit avoir une démarche écoconception. Cette démarche vise notamment à diminuer l'obsolescence des terminaux. Il faut la prendre en compte dès la conception. Il faut des sites plus légers pour prolonger la durée de vie des terminaux.Encourager les utilisateurs à garder leur matériel (via des applications légères et efficaces).

-3 leviers permettent d'alléger un site web:

-Frugalité fonctionnelle: supprimer les fonctionnalités peu utilisées, performances des algorithmes.

-Optimisation du contenant : Optimiser les codes sources serveur et client : inspecter le code pour identifier les éléments de code inutiles

-Optimisation du contenu: définition des images, contenu statique.

MédiLabo Solutions
We care for you

MédiLabo Solutions

6. Green Code

-Coté front:

-Concevoir des applications légères et efficaces.
-Eliminer les fonctionnalités non essentielles.
-Support des anciens terminaux (mobiles, avec un réseau peu performant)
-Limiter le poids de la page, le nombre de requètes serveur.
-70 % des fonctionnalités demandées par les utilisateurs ne sont pas essentielles , 45 % ne sont jamais utilisées.

-Coté back:

-Données : durée de stockage des données, expiration des données.
-Test de performances.

MédiLabo Solutions
We care for you

MédiLabo Solutions

-Pour notre application les points qui pourraient être intéressant d'analyser:

-Dans le microservice Back Diabète, l'optimisation de l'algorithme de recherche des termes diabètes (test de performance).

-Dans le microservice Front, il est possible de diminuer le nombre de requètes. Il y a une requête pour récupérer la liste des patients, et une requête pour récupérer un patient précis pour le mettre à jour, alors que l'on a déjà la liste avec tous les patients disponibles. Pour afficher la liste est ce qu'il est utile de récupérer l'intégralité des objets. Idem pour les notes.

-Dans le microservice Front, est ce que les touches Supprimer un patient et une note, sont bien utiles.

-Dans le microservice Front, les touches visu fiche et note sont redondantes.

-Concernant la durée de vie des notes dans la base de données MongoDb, peut être qu'il serait pertinent de rajouter une donnée pour la durée de validité de la note.

MédiLabo Solutions
We care for you