

## **General introduction :**

Throughout their pregnancy journey, women not only face inevitable mental and physical drastic changes but also struggle with a myriad of challenges that go along with it; from navigating the complexities of monitoring their health effectively, to the daunting amount of information available about pregnancy. Managing prenatal appointments amidst all these dire challenges puts an even harder strain on the pregnant, leading to confusion and even anxiety.

My second semester project consists of developing a pregnancy tracker web application as one of the requirements for the successful completion of my second year at the Higher Institute of Information and Communication Technologies .

### **1. Introduction :**

In the first chapter, I will focus on addressing the requirements specification of my project. I will start with specifying the functional and nonfunctional requirements of the latter. Next, I will showcase the global use cases and class diagrams of the application . Finally, I will specify the working environment that I have chosen for my project.

### **2. Functional requirements :**

Functional requirements, known also as the functional specification, are a collection of requirements that outlines the main goal of the system and the purpose of its offerings to all users. It features the system's functionalities that the team must fulfill during the process of development, which helps them to keep track of their progress.

- a. Authenticate**
- b. Record pregnancy reports**
- c. Manage calendar**
- d. Access educational resources**
- e. Access dietary recommendations**
- f. Monitor pregnancy reports**
- g. Manage dietary recommendations**
- h. Manage pregnancy documents**

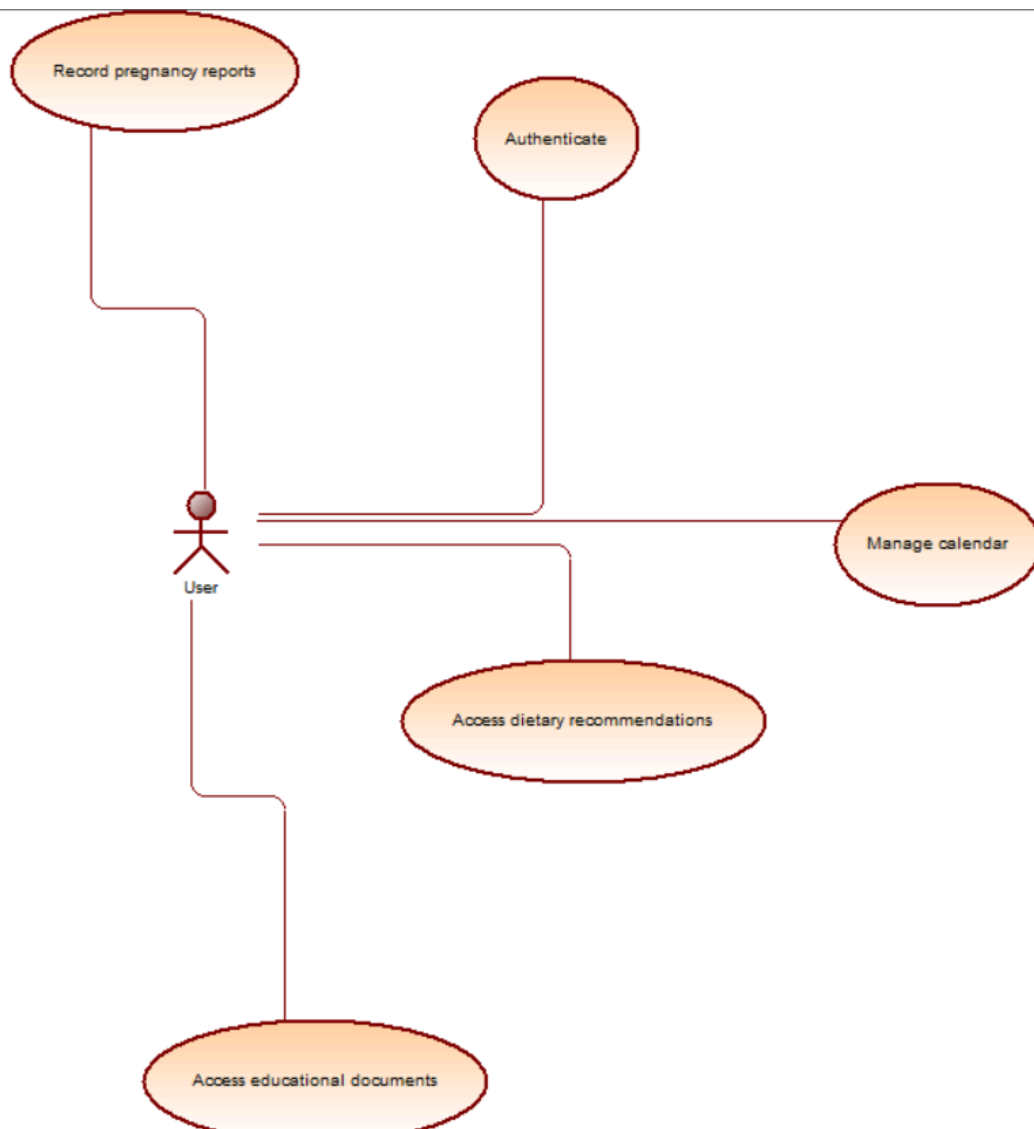
### 3. Non-Functional requirements :

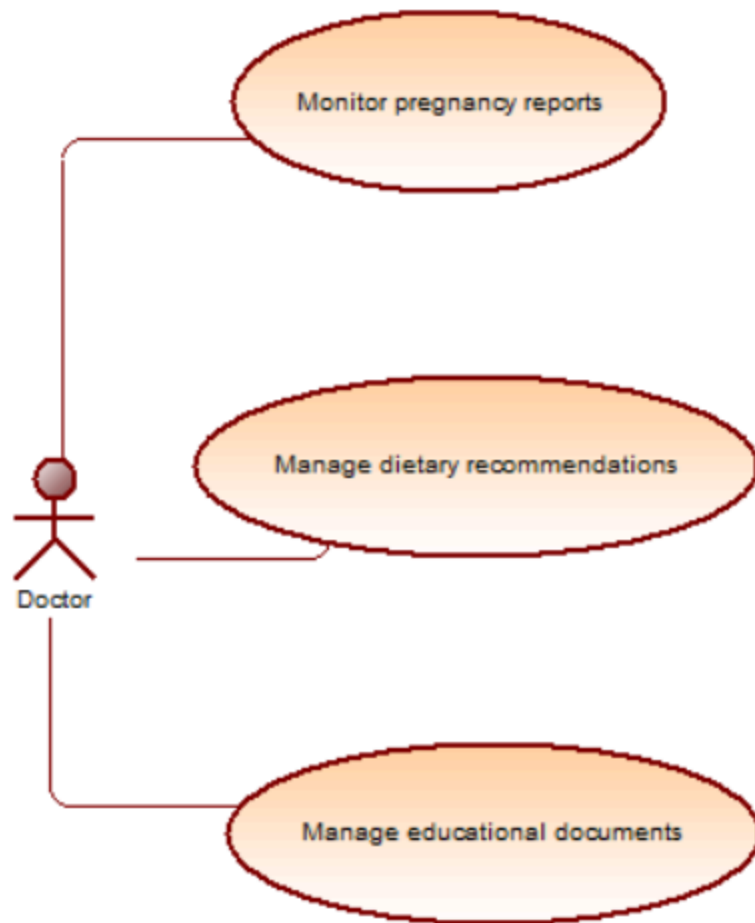
The non-functional requirements describe the developed system qualities when it is performing one of its use cases. In fact, these requirements are more of a set that defines the manners to hold the functional ones on a line that is bound to the software system judgements followed by its security, portability, usability etc. . . . Therefore, the system won't meet the expectations of neither the owner nor the user unless these requirements are fulfilled. Yet, even if the requirements are not respected, the system would still work.

- a. Usability
- b. Security
- c. Scalability
- d. Availability
- e. Documentation

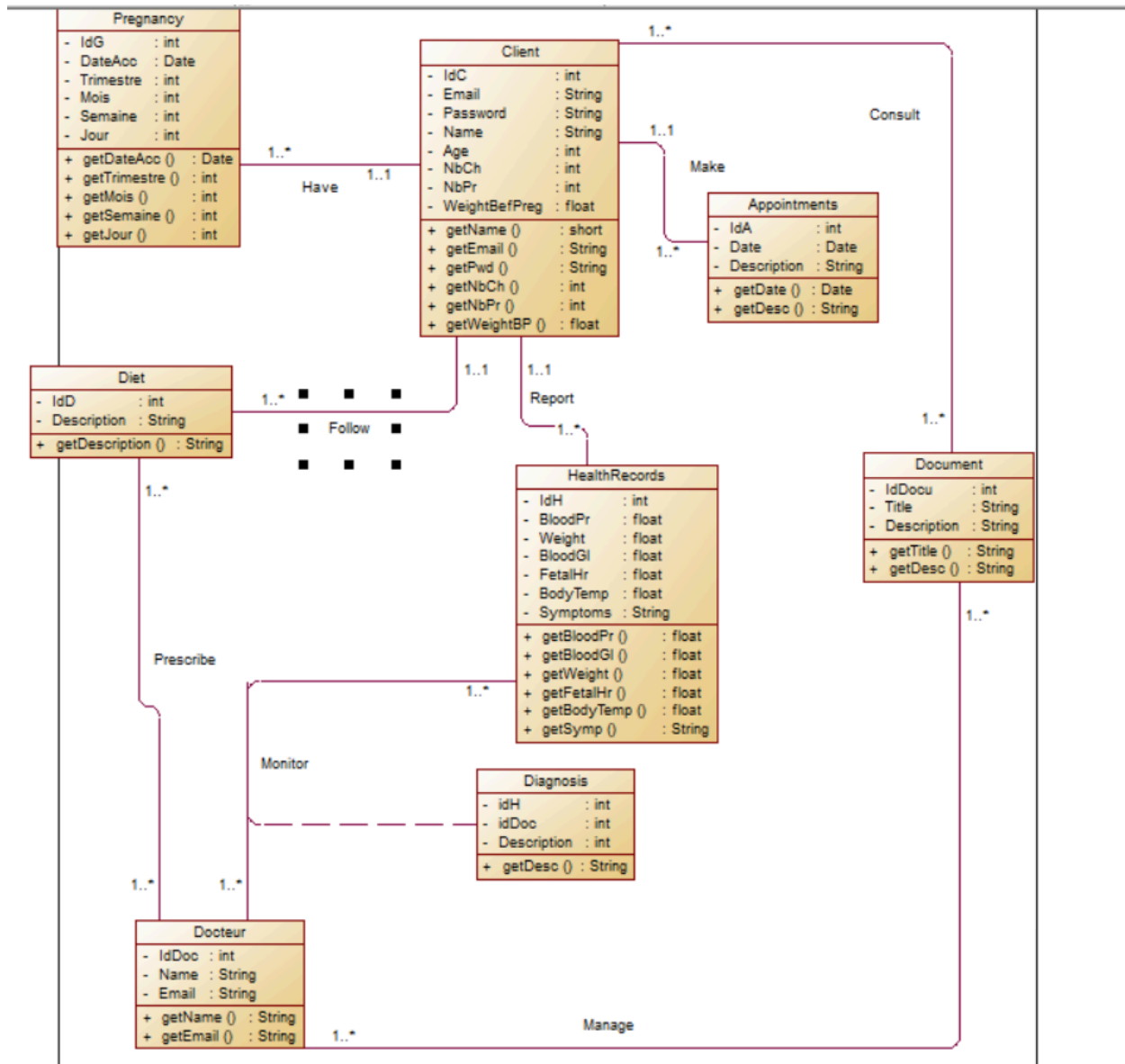
### 4. Global use cases diagram :

---





## 5. Global classes diagram



## 6. Working environment :

For this part, I will be presenting our technological choices that I thought about and followed during the process of developing the system. These technological choices can be classed into front-end technologies and back-end technologies.

- **Frontend :**

The front-end technologies are responsible for anything displayed on a website, desktop or a mobile application, which the user can interact with when looking at the things represented on those platforms.



**Angular :** Angular is an open-source JavaScript framework primarily maintained by Google, aimed at crafting dynamic web applications. It equips developers with a structured framework for creating single-page client-side applications, facilitating the development process thanks to features such as data binding, dependency injection, and modularization.

- **Backend :**

Back-end technologies are responsible for anything that is not displayed on a website, desktop or a mobile application and which the user can't interact with directly.



**PHP :** PHP is an open-source server-side scripting language famously used for web development. It provides developers with a versatile platform for creating dynamic and interactive web applications, through powerful features like easy integration with databases, session management, and extensive libraries for various functionalities. PHP's simplicity and flexibility make it a popular choice for developing websites and web-based applications.



**MySQL :** MySQL is an open-source relational database management system (RDBMS) that enables users to interact with databases either directly by using SQL, or more often with other programs to implement applications that require relational database capabilities.

- **Software :**



**Visual Studio Code :** VS Code is Microsoft's Integrated Development Environment (IDE) for every existing operating

system. With features such as smart code completion, snippets, debugging, and an extensive library of extensions, it stands as the premier choice for developers worldwide.



**Power AMC :** Power AMC is a powerful and versatile modeling tool developed by SAP. It offers an extensive suite of features for designing and visualizing databases, business processes, and software architectures. With its user-friendly interface and comprehensive range of modeling options, Power AMC allows users to easily create complex data models, entity-relationship diagrams, and UML diagrams.

- **Teamwork tools :**



**Git and GitHub :** Git is a version control system distributed by GitHub; a code hosting platform. It is a software developed to keep track of the changes made in files over time, which helps users always have the latest version of each file in a repository, compare those changes and revert them to a specific state if needed later .

