Thesis submitted to the

UNIVERSITY OF MOHAMED BOUDIAF – MSILA

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE

In partial fulfillment of the requirements for the degree of

Master in **Computer science**

By

Mezrag, Nour El Houda

Merniz, Maroua

Title of the thesis

BloodLink. Smart Blood Donation Management System

Under the supervision of

Hemmak Allaoua

Composition of the jury

Amraoui Noureddine	University of Msila	Président
Hemmak Allaoua	University of Msila	Reporter
Bougherara Seddik	University of Msila	Examiner

June, 2026

Abstract

Blood donation is essential for saving lives and supporting medical procedures. This project focuses on developing a smart digital platform for managing blood donations, connecting donors with those in need through a matching system based on blood type and the type of request (emergency/scheduled). The project's primary objective is to improve the organization of blood donations by streamlining request management, sending instant notifications to suitable donors, and providing customized dashboards for both donors and patients. This will contribute to the digitalization of the healthcare sector and facilitate rapid access to blood donation services in Algeria.

Key words: blood donation; Digital platform; Smart health system; Donor Management

---

الملخص

يعتبر التبرع بالدم عاملاً أساسياً في إنقاذ الأرواح ودعم الإجراءات الطبية. يركز هذا المشروع على تطوير منصة رقمية ذكية لإدارة عمليات التبرع بالدم، تربط بين المتبرعين والمحتاجين للدم، من خلال نظام مطابقة يعتمد على فصيلة الدم ونوع الطلب (طارئ/مخطط).يتمثل الهدف الأساسي للمشروع في تحسين تنظيم عمليات التبرع بالدم من خلال تسهيل إدارة الطلبات، وإرسال إشعارات فورية للمتبرعين المناسبين، وتوفير لوحات تحكم مخصصة للمتبرع والمريض ، مما يسهم في رقمنة قطاع الصحة وتيسير الوصول السريع لخدمات التبرع بالدم في الجزائر.

الكلمات المفتاحية: التبرع بالدم؛ منصة رقمية ؛ نظام صحي ذكي؛ إدارة التبرع بالدم

Dedications

قال تعالى

فاذكروني اذكركم واشكروا لي ولا تكفرون «الحمد لله الذي بنعمته تتم الصالحات وبفضله تتحقق الغايات و الصلاة والسلام على افضل خلق الله ولحسن حظي وباسم اول خريجة بعائلتها بكل فخر اهدي تخرجي وفرحتي التي انتظرتها طويلا الى من كانو مصدر الدعم والعطاء دائما

الى الذي تحمل الصعاب وسخر حياته من اجل سعادتي الى الذي سعى طوال حياته لأكون لأفضل الى يدي اليمنى ابي الغالي

الى من انارت كلماتها طريقي الى من كانت دعواتها تحيطني الى امي الغالية

كانت الدعم لأول لتحقيق طموحاتي

الى مصدر قوتي الى خيرة ايامي وصفوتها الى ضلعي الثابت وامال ايامي اخوتي اللذين كان حبهم يعلو فوق كل حب لكل من كان عونا وسندا في هده الطريق

واخيرا

لنفسي فقط لمقاتلتي كل يوم ، لا أحد يعرف كم عانيت حتى أصل لما أنا عليه الآن، لا أحد

شكراً لنفسي التي صبرت واجتهدت

نورالهدى مزراق , مروة مرنيز

Acknowledgments

"وآخر دعواهم أن الحمد لله رب العالمين ,"الحمد لله الذي ماتيقنت به خيراً وأملاً إلا وأغرقني سرورا

أتقدم بجزيل الشكر والتقدير إلى الأستاذ الدكتورهماك علاوة على جهوده المباركة وتوجهاته القيّمة التي كان لها الأثر الكبير في إخراج هذا البحث إلى النور،وأُعرب عن خالص شكري ووافر تقديري لأعضاء لجنة المناقشة الموقرة على تفضلهم بقبول قراءة هذه الرسالة، وعلى ما خصصوه من وقت وجهد ثمين في تقييمها,كما لا يفوتني أن أشكر كافة أعضاء هيئة التدريس في قسم الاعلام الآلي على ما قدّموه من علم ومعرفة،ولا أنسى تقديم الشكر لعائلتي، التي كانت دائما الداعم الأول، ووقفت بجانبي في كل المراحل.

.

Contents

List of figures v
List of tables vi
General Introduction 7
Chapter 1: General Overview on Blood Donation 9
1.1. Introduction 9
1.2. Significance of Study 9
1.3. Blood Components 9
1.3.1. Red Blood Cells 9
1.3.2. Plasma 9
1.3.3. Platelets 10
1.4. Blood Donation Processin Digital Systems 10
1.4.1. Challenges in Traditional Blood Donor Management 10
1.4.2. Information Systems for Blood Donation Management 10
1.4.3. Advantages of Digital Blood Platforms 11
1.5. Presentation of similar projects 11
1.5.1. DamDz 11
1.5.2. Ahyaha 12
1.5.3. BloodLinks 13
1.5.4. Zomraty 13
1.6. Conclusion 14
Chapter 2: Analysis and Modeling 16
2.1. Introduction 16
2.2. Specification of needs 16
2.2.1. Definition of actors 16
2.2.2. Functional Requirements 17
2.2.3. Non- Functional Requirements 18
2.3. UML language 18
StarUML 19
Use case diagram 20
Class diagram 21
Sequence diagram 22
2.4. Conclusion 32
Chapter 3: Implementation 33
3.1. Introduction 33
3.2. System Requirements 33

3.3. System architecture 33
3.3.1. HTML 34
3.3.2. CSS 34
3.3.3. MySQL 34
3.3.4. JavaScript 34
3.3.5. PHP 34
3.4. Software architecture 34
3.5. Interface overview 36
3.5.1. Home page 37
3.5.2. Admin login 37
3.5.3. User login 39
3.5.4. Donor registration 40
3.5.5. Request Blood 41
3.5.6. Admin home 42
3.5.7. Donation appointment 43
Conclusion 44
General Conclusion 45
References 46

List of figures

Fig. 1:'DamDz' platform Homepage Interface 11
Fig. 2:Registration form 12
Fig. 3:Donor search page 12
Fig. 4:StarUml interface 19
Fig. 5:Use case Diagram 20
Fig. 6:Class Diagram 22
Fig. 7:: Sequence Diagram for Signup 23
Fig. 8:Sequence Diagram for Login 25
Fig. 9:Sequence Diagram for recipient 26
Fig. 10:Sequence Diagram for donor 28
Fig. 11:Sequence Diagram for admin 30
Fig. 12:Visual studio code interface 35
Fig. 13:PhpMyAdmin interface 36
Fig. 14:Bloodlink interface 37
Fig. 15:Admin Login 38
Fig. 16:User Login interface 39
Fig. 17:Donor regestrition 40
Fig. 18:Request blood 41
Fig. 19:Admin panel 42
Fig. 20:Donation appointment interface 43

List of tables

Table 1: comparison between similar projects 14

General Introduction

Blood donation is an essential medical procedure that significantly contributes to saving lives and bolstering healthcare services globally. Each day, hospitals and healthcare facilities require blood transfusions for patients undergoing surgical procedures, victims of accidents, individuals afflicted with chronic illnesses, and urgent medical cases. Given that no artificial alternative can entirely substitute human blood, the presence of blood donors is crucial for ensuring a sufficient blood supply.

With the swift advancement of information and communication technologies, healthcare systems are progressively embracing digital solutions to enhance service quality and operational effectiveness. In the realm of blood donation, digital platforms have emerged as invaluable resources for facilitating communication among donors, recipients, blood banks, and healthcare organizations. These systems aid in improving donor management, expediting blood request processing, and delivering quicker responses during emergencies.

Despite the availability of numerous blood donation applications and websites, various challenges continue to exist. Blood banks frequently encounter obstacles in promptly identifying appropriate donors, while patients and their families may find it difficult to locate compatible donors in critical situations. Furthermore, conventional management practices can result in delays, restricted communication, and ineffective handling of blood donation requests.

In response to these challenges, this project introduces "BloodLink", a Smart Blood Donation Management System aimed at linking blood donors with those in need via a centralized and user-friendly digital interface. The system streamlines donor registration, manages blood requests, schedules appointments, provides notification services, and ensures effective matching between donors and recipients based on blood type and urgency of requests.

The impetus for this initiative stems from the humanitarian significance of blood donation and the necessity to enhance access to blood donation services in Algeria. By leveraging contemporary web technologies, BloodLink aspires to aid in the digital transformation of healthcare services and promote quicker, more effective blood donation procedures.

The primary goals of this project include:

* Creating an intelligent platform for managing blood donations.
* Enhancing communication between donors and recipients.
* Optimizing the matching process for blood requests and available donors.
* Allowing for appointment scheduling and management of notifications.

* Offering a straightforward and effective solution for both emergency and scheduled blood requests.

This dissertation is structured as follows:

Chapter 1: Provides a comprehensive overview of blood donation, its elements, challenges, and current blood donation platforms.

Chapter 2: Outlines the analysis and modeling phase of the proposed system, encompassing requirements specification, UML modeling, and system design.

Chapter 3: Details the implementation phase, including the technologies employed, the software architecture, and the primary interfaces of the BloodLink platform.

Chapter 1: General Overview on Blood Donation

1.1. Introduction

Blood donation is a procedure that can help save lives. People volunteer to donate blood. There are many types of blood donation. Each type helps meet different medical needs; donors must meet certain essential criteria before they can donate blood.

Therefore, it is difficult to maintain a blood donor database without the use of software. In this chapter, we will describe the concepts inherent to the issue of blood donation.

1.2. Significance of Study

Blood donation is a vital medical service that helps save lives in emergencies, surgeries, and for patients suffering from chronic diseases. However, many blood banks still face challenges in ensuring the availability of blood and managing donor information efficiently.

This study is significant because it proposes a smart blood donation management system, BloodLink, which aims to improve the organization of the donation process through digital technologies. The system facilitates communication between donors and recipients, reduces delays in finding compatible blood types, and supports faster responses during urgent situations.

1.3. Blood Components

Blood is often separated into its individual components, including:

1.3.1. Red Blood Cells

Red blood cells are used to treat all kinds of anemia (where people have low hemoglobin levels) including:

- as a result of rheumatoid arthritis or cancer
- when red cells break down in newborn babies
- sickle cell disorder

Red blood cell transfusions replace the heavy blood loss that can occur in an accident, during surgery or in childbirth**.**

1.3.2. Plasma

Plasma is the part of your blood that carries platelets, red blood cells and white blood cells around the body.

Plasma contains antibodies, which fight infection. These antibodies can be made into medicines to help people with rare diseases, immune disorders and genetic conditions. Some plasma can also be given to patients as a transfusion.

1.3.3. Platelets

Platelets are crucial in helping blood to clot: they do this by clumping together to stop bleeding after an injury. They can be used:

- to treat cases of bone marrow failure;
- following a transplant or chemotherapy treatments;

1.4. Blood Donation Processin Digital Systems

1.4.1. Challenges in Traditional Blood Donor Management

It is Traditionally; blood donor management has faced several challenges. These include donor recruitment and retention, inventory management, matching blood types, and ensuring the safety and quality of the blood supply. Manual processes, often cumbersome and error-prone, have been a significant bottleneck, leading to inefficiencies in blood collection, testing, storage, and distribution.

1.4.2. Information Systems for Blood Donation Management

Blood donation management systems are defined as digital information systems used to .organize and coordinate the donation process between donors and healthcare institutions These systems focus on donor registration, data management, appointment scheduling, and communication facilitation, thereby improving the donor experience and increasing retention .rates

This system also contributes to faster response times to emergencies, enhanced data .accuracy, and reduced administrative complexities associated with traditional procedures

1.4.3. Advantages of Digital Blood Platforms

Information systems help allowing for speedier decision-making and greater coordination among patients, donors and healthcare organization. With the growth to online technologies, digital platform has become reliable alternatives for blood donation and request management. Web-based system provides real-time access to donor information, increase donor-Recipient / Patients matching and reduces response time in emergency situations through instant communication.

1.5. Presentation of similar projects

1.5.1. DamDz

The 'DamDz' platform is a non-profit humanitarian platform aiming to Connect blood donors and those needing blood in Algeria, it facilitates donor searches by blood type and location and Save lives by providing a fast and efficient communication channel.

As illustrated in **Fig.1.1**, the homepage layout of the DamDz platform enables users to effortlessly sign up as donors, request blood, and access information regarding urgent donation requirements, including details about location and timing.

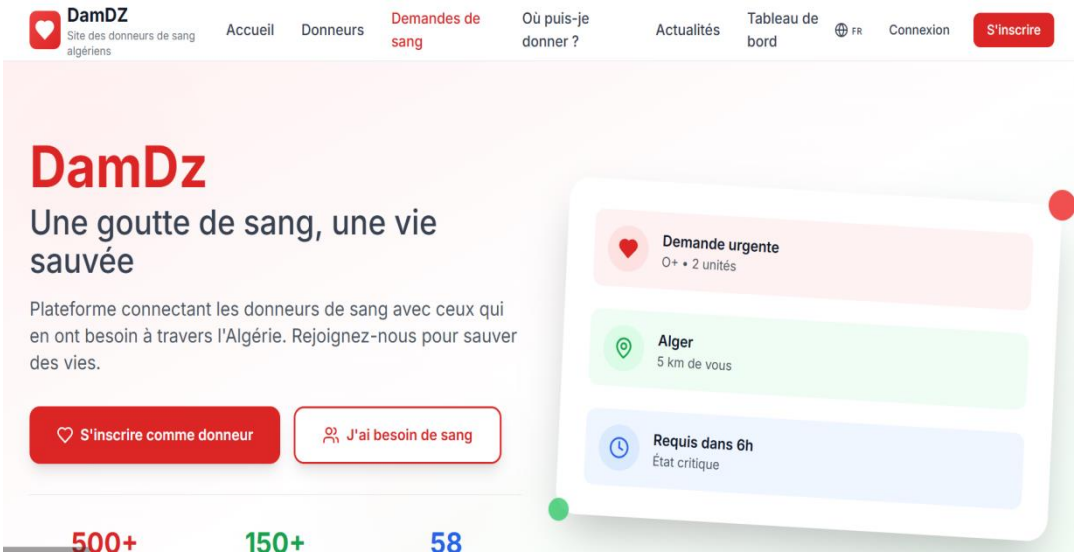

Fig. 1:'DamDz' platform Homepage Interface

1.5.2. Ahyaha

As illustrated in Figures 1.2 and 1.3, the 'Ahyaha' website serves as a platform aimed at linking individuals requiring blood with prospective donors. It enables the search for donors and the registration of users, and is overseen at the state level by local teams that collaborate with hospitals and civil protection units to orchestrate blood donation campaigns and promote community involvement.

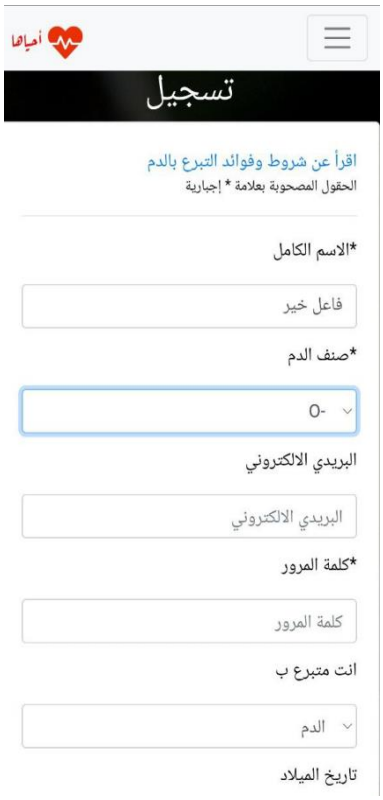

Fig. 2:Registration form

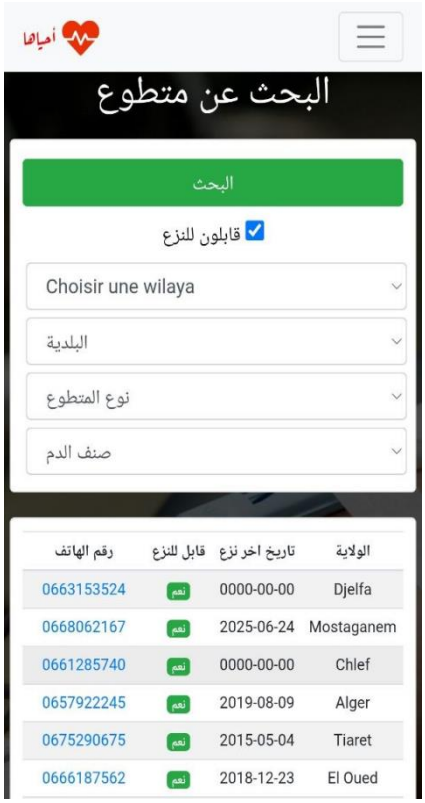

Fig. 3:Donor search page

1.5.3. BloodLinks

Bloodlinks is a system designed to facilitate the real-time identification of blood banks or hospitals within a specific region. Its primary objective is to simplify the process of locating blood banks and hospitals for both blood donation and requests. The Bloodlinks application and web platform enable users to conduct online searches for blood banks and hospitals, ensuring access to reliable sources for blood donation and retrieval. Additionally, it assists blood banks in managing the acceptance of blood, processing donations, handling blood requests, and distributing blood to individuals in need through its comprehensive blood bank management software.

1.5.4. Zomraty

This application is designed to save thousands of lives in Algeria by serving as an essential first step for individuals requiring urgent blood transfusions. It offers comprehensive information about donors, enabling those in need to select a suitable donor. The app's key features encompass the ability to search for local donors, the option to donate blood, and the possibility of becoming a regular donor.

The table below provides a comparative examination of the key digital platforms focused on blood donation, specifically DamDZ, Zomraty, BloodLinks and Ahyaha.

The assessment is grounded in various functional and technical criteria, which encompass the type of platform available, integrated communication features, request posting and management capabilities, and the incorporation of geolocation services through Google Maps. Additionally, the analysis takes into account the management of emergency alerts, the process of sending requests to the blood bank, as well as metrics related to overall system efficiency and user satisfaction levels.

Application / Criteria	Damdz	Ahyaha	BloodLinks	Zomraty
Communication Methods	Notifications, SMS	Call	In-app notifications	Message
Google Maps Search	No	Yes	Yes	No
The person in need sends a request to the blood bank	No	No	No	No
Urgent Request Alerte	No	No	Yes	No
Efficiency and user satisfaction	Medium	Low	Medium	Good

Table 1.1: comparaison between similar projects

1.6. Conclusion

In this chapter, we have delineated the essential concepts pertaining to our project, which focuses on blood donation and its various components. Additionally, we have explored other comparable projects and offered a concise comparison. In the subsequent chapter, we will analyze the needs and design aspects of our system.

Chapter 2: Analysis and Modeling

2.1. Introduction

After presenting the general context of our project in the first chapter, this second chapter, entitled " Analysis and Modeling of BloodLink", we will examine and define the main functions, identify stakeholders, and describe initial use cases. We will also review various diagrams illustrating our project's workflow.

In practice, all software development work should be carried out using a methodology that enables developers to manage their projects more efficiently. Many modeling languages exist; we have chosen the Unified Modeling Language (UML).

2.2. Specification of needs

This phase entails comprehending the context of the system along with its primary actors. Additionally, it outlines the system's goals and the limitations it must follow. The requirements are categorized into functional and non-functional types.

2.2.1. Definition of actors

An actor represents a user role that interacts with the system you are modeling. The user can be a human user, an organization, a machine, or another external system.

In our website, there are three main actors who are:

- **Donor:** A donor is a registered user in the system who can manage their profile (view and update it), view blood requests, and approve or decline donation requests. They can also book appointments at the blood transfusion center and receive notifications about new requests.
- **Recipient:** is the person who needs blood, they can search for available donors, browse the website, send requests (either urgent or scheduled), and receive notifications about the status of their requests.

- ✧ **Admin: The** administrator plays a vital role in the efficient management of the blood donation process. Their responsibilities include managing donor appointments, processing blood requests, and accurately adding new donors to the system, in addition, The admin has an overview of all system activities.

2.2.2. Functional Requirements

Functional requirements describe how the system should respond to user inputs, perform operations, and produce expected results. These requirements are a set of behaviors and functions a system must possess to meet user needs.

Donor :

- ✧ Authentication: Log in or create a new account.
- ✧ Manage Profile : Révise personnelle détails.
- ✧ Manage recipient requests: Review and respond to blood donation requests.
- ✧ Book Appointment at blood center Arrange a time to donate blood.
- ✧ Receive notifications Obtain updates and alerts.
- ✧ Browse the notifications Obtain available information.

Récipient :

- ✧ Authentication: Log in or sign up.
- ✧ Manage Authentiquassions : Log personnelle information.
- ✧ Search for donors: Locate appropriate blood donors.
- ✧ Browse the website: Explore and access information.
- ✧ Send blood request: Submit a request for blood from donors.
- ✧ Receive notifications: Get updates regarding requests.
- ✧ Track Request Status: Monitor whether the request has been accepted or is still pending.
- ✧ Choose the type of blood request (urgent or planned): opt for urgent or scheduled requests.

Administration :

- ✧ Authentication: Securely log into the system.
- ✧ Manage appointments: Create, modify, or remove appointments.
- ✧ Manage blood collection (process and verify donations): Oversee and confirm donations.
- ✧ Add donors: Enroll new donors into the system.
- ✧ Manage requests: Organize and address blood requests.
- ✧ Receive notifications: Get alerts from the system.
- ✧ Manage profile : Révise administration information.

2.2.3. Non- Functional Requirements

Non-functional requirements specify criteria that evaluate how a system performs a function, rather than the function itself. Where functional requirements describe specific system functions, non-functional requirements define aspects like:

- ✧ Security: The system must protect sensitive data related to donors and patients.
- ✧ Reliability: The system must operate correctly and consistently without failures.
- ✧ Portability: The system should be usable across different devices, platforms, or environments.
- ✧ Scalability: The system must be able to handle an increasing number of users and data efficiently.
- ✧ Performance: The system should provide fast response times and efficient processing of user requests.

2.3. UML language

The Unified Modeling Language (UML) comprises various built-in diagrams that software developers utilize for visually representing objects, states, and processes within a software application or system. This modeling language can act as a blueprint for a project, ensuring a well-structured information architecture. Additionally, it helps developers in

conveying their system descriptions in a manner that is comprehensible to external specialists.

StarUML

StarUML is an open-source UML (Unified Modeling Language) modeling software that can replace expensive commercial software like Rational Rose or Together in many situations. Being easy to use, requiring few system resources, and supporting UML 2, this software is a good option for getting acquainted with modeling.

The figure below presents the starUML software:

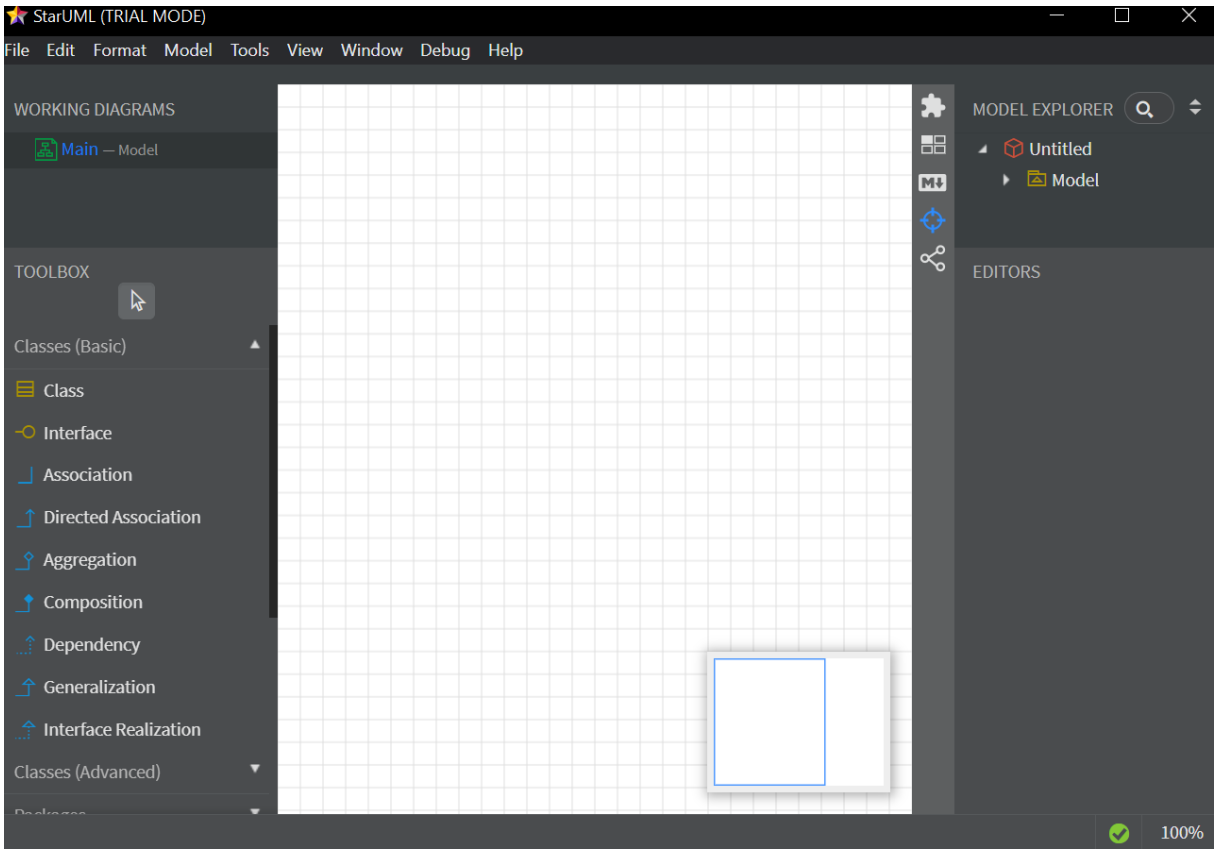

Fig. 4:StarUml interface

Fig. 2.1: StarUml interface

Use case diagram

A Use Case Diagram is a type of Unified Modeling Language (UML) diagram that represents the interaction between actors (users or external systems) and a system under consideration to accomplish specific goals. It provides a high-level view of the system's functionality by illustrating the various ways users can interact with it.

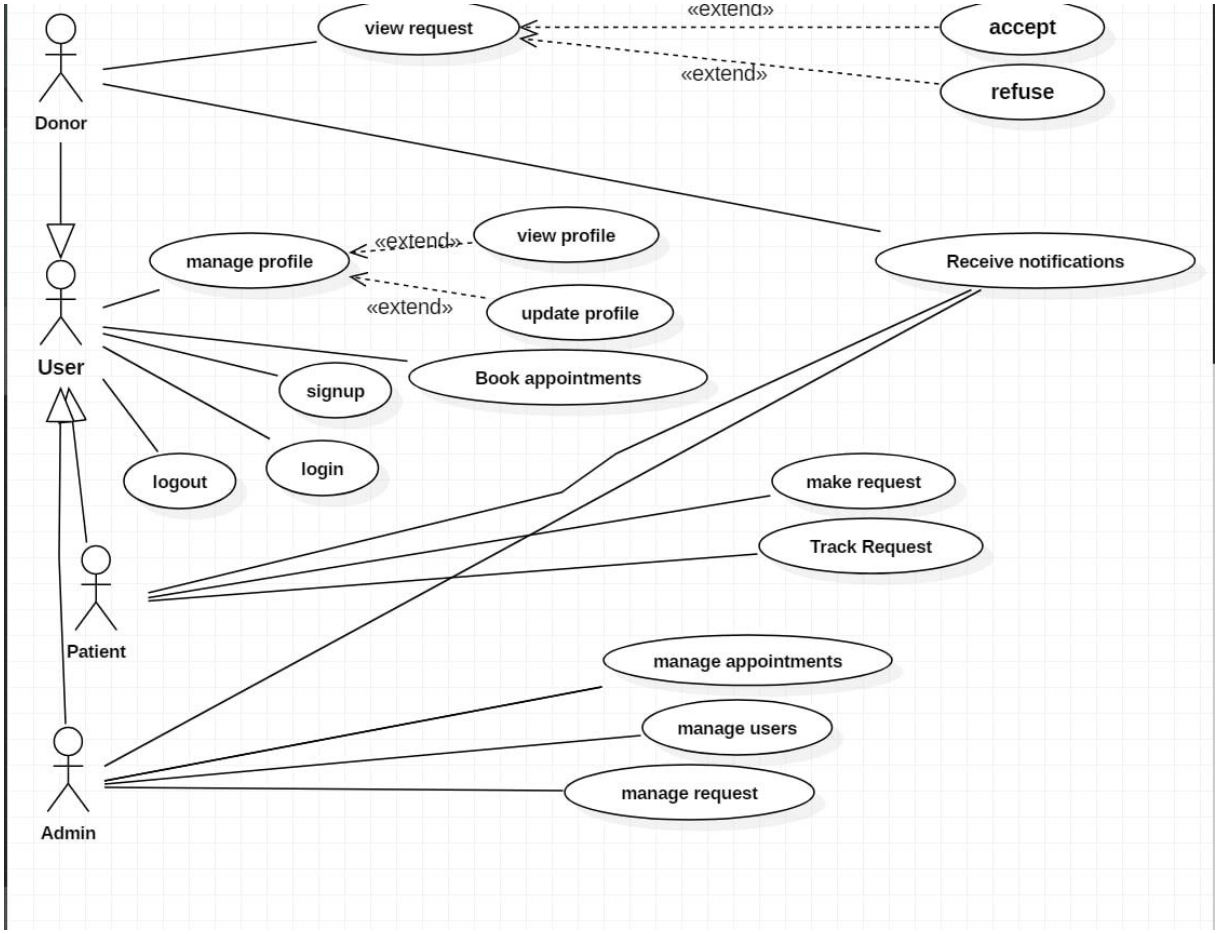

Fig. 5:Use case Diagram

Class diagram

A UML class diagram shows the structure of a system by displaying its classes, their attributes, methods, and the relationships between them. It helps the team understand how the system is organized and how components interact.

The primary class is User, has attributes like Name, Email, and Password. The attribute Password and Email are private (–) to protect data, and the methods are kept as public (+) so that such data can be accessed by other parts of the system. Moreover, the specialized classes like Donor, Recipient, Guest, and Admin inherit from User. The diagram also illustrates relationships: generalization represents the inheritance hierarchy.

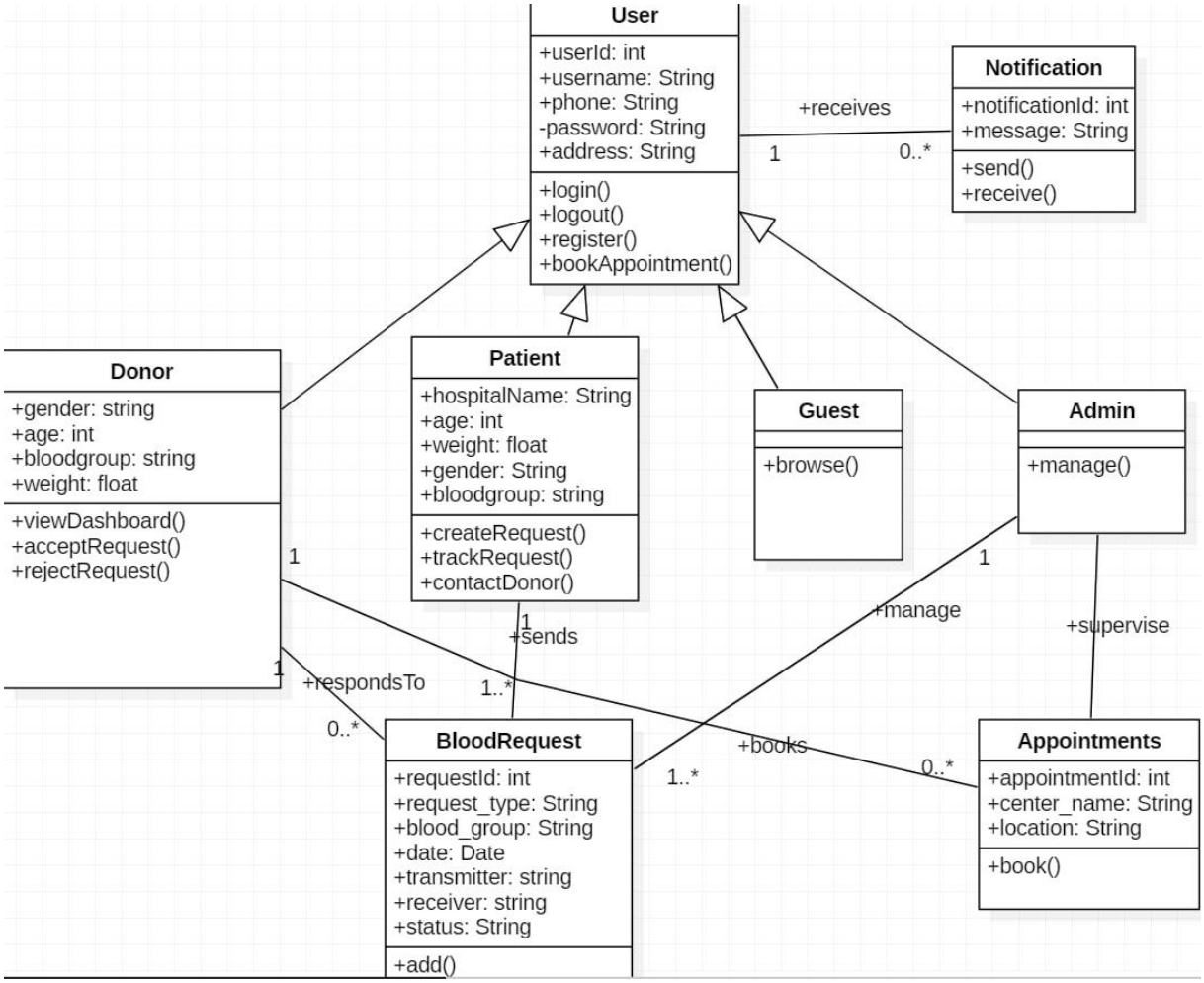

Fig. 6:Class Diagram

Sequence diagram

A sequence diagram represents a specific category of UML diagram that demonstrates the sequence of messages transmitted between objects during an interaction. This diagram includes a set of objects, represented by lifelines, as well as the messages exchanged among these objects throughout the interaction.

The first sequence diagram depicts the procedure for user registration, in which a new user establishes an account within the system.

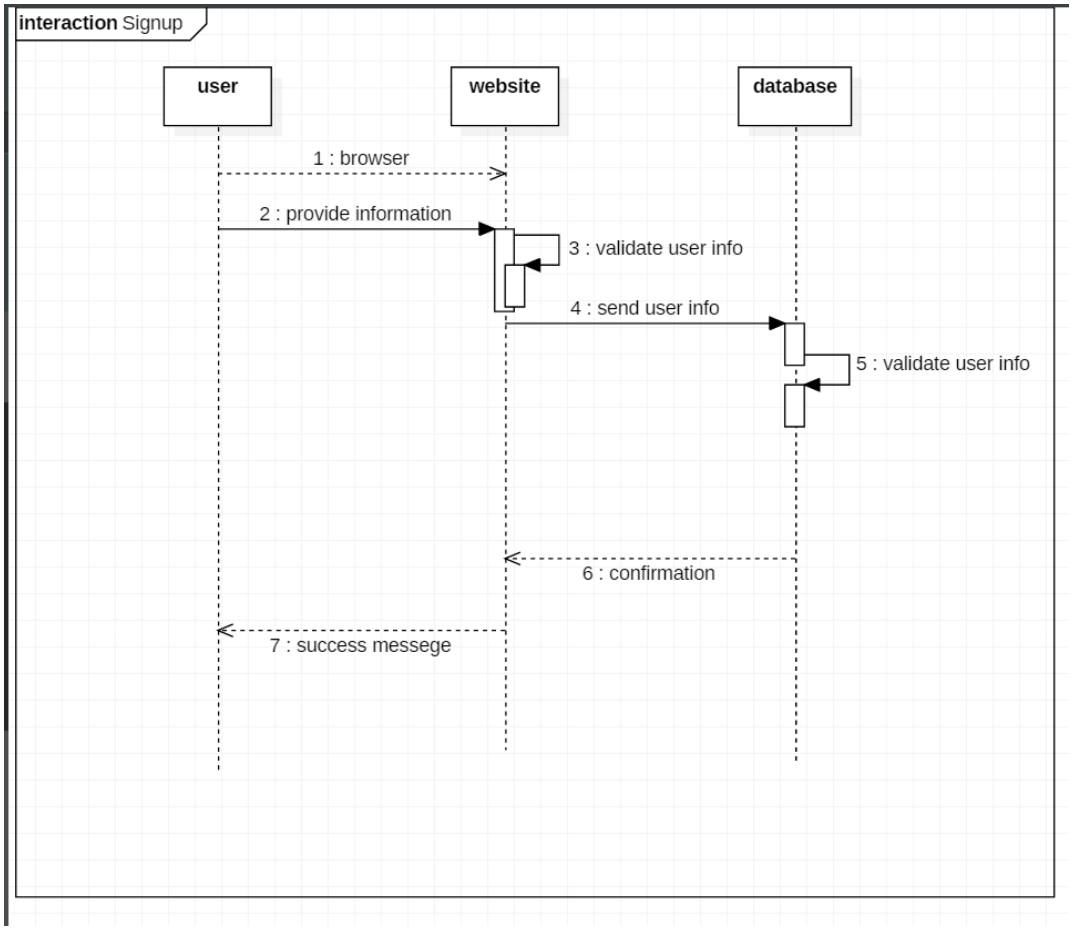

Fig. 7:: Sequence Diagram for Signup

Description: Outline the technical procedures that enable a user to establish a new account (register) within the system.

Primary actor: User.

Start: When the user navigates to the registration page (through the browser).

Precondition: The user has not yet created an account (or the application is operational).

Scenario description:

The user inputs the necessary details into the form (provide information):

- ✧ The system (website) checks the local validity of the information (validate user info);
- ✧ The system transmits the user information to the database (send user information);
- ✧ The database verifies and stores the data (validate user info);
- ✧ The database sends a confirmation back to the system (confirmation);
- ✧ The system presents a "Success Message" to the user (success message);

Sequence Diagram of User Login:

Description: Outline the technical steps that enable a user to authenticate and access their personal space.

Primary Actor: User (this may be a donor, requester, or guest.

Start: When the user navigates to the login page of the website.

Precondition: The user must have an account that has already been created (registered in the database).

Scenario Descriptions :

Nominal Scenario :

- ✧ The user visits the website (visit);
- ✧ The user enters their credentials (valid information);
- ✧ The system (Website) performs a local verification of the fields (verification);
- ✧ The system sends the information to the database for validation (valid info);
- ✧ The database checks the match of the credentials (verification);
- ✧ The database confirms that the information is correct (correct);
- ✧ The system redirects the user to the homepage (enter home).

Alternative Scenario (if incorrect):

If the information is incorrect:

- ✧ The database returns an error indication to the system (incorrect info);
- ✧ The system informs the user that the information is incorrect (incorrect info);
- ✧ The system keeps the user on the login page (rest login page) to retry.

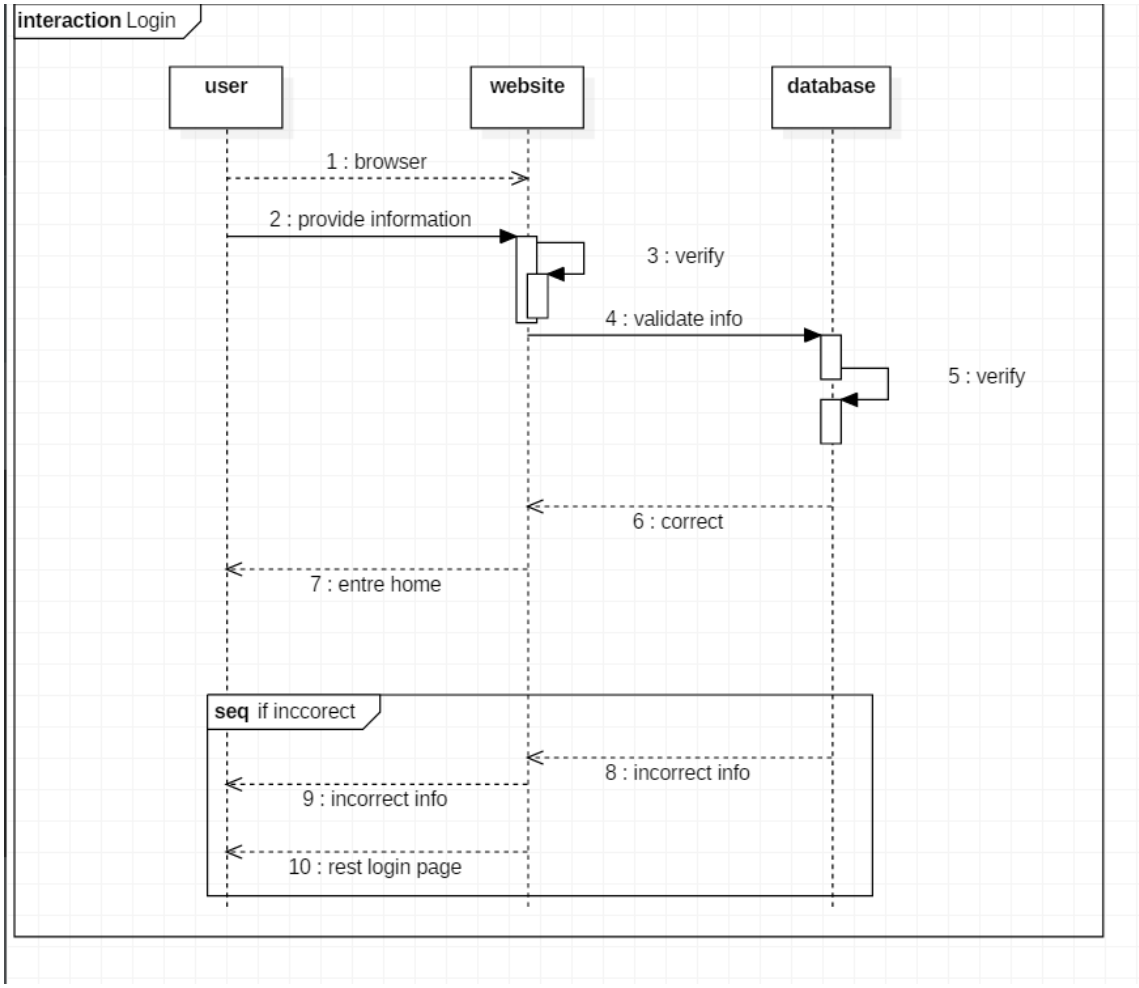

Fig. 8:Sequence Diagram for Login

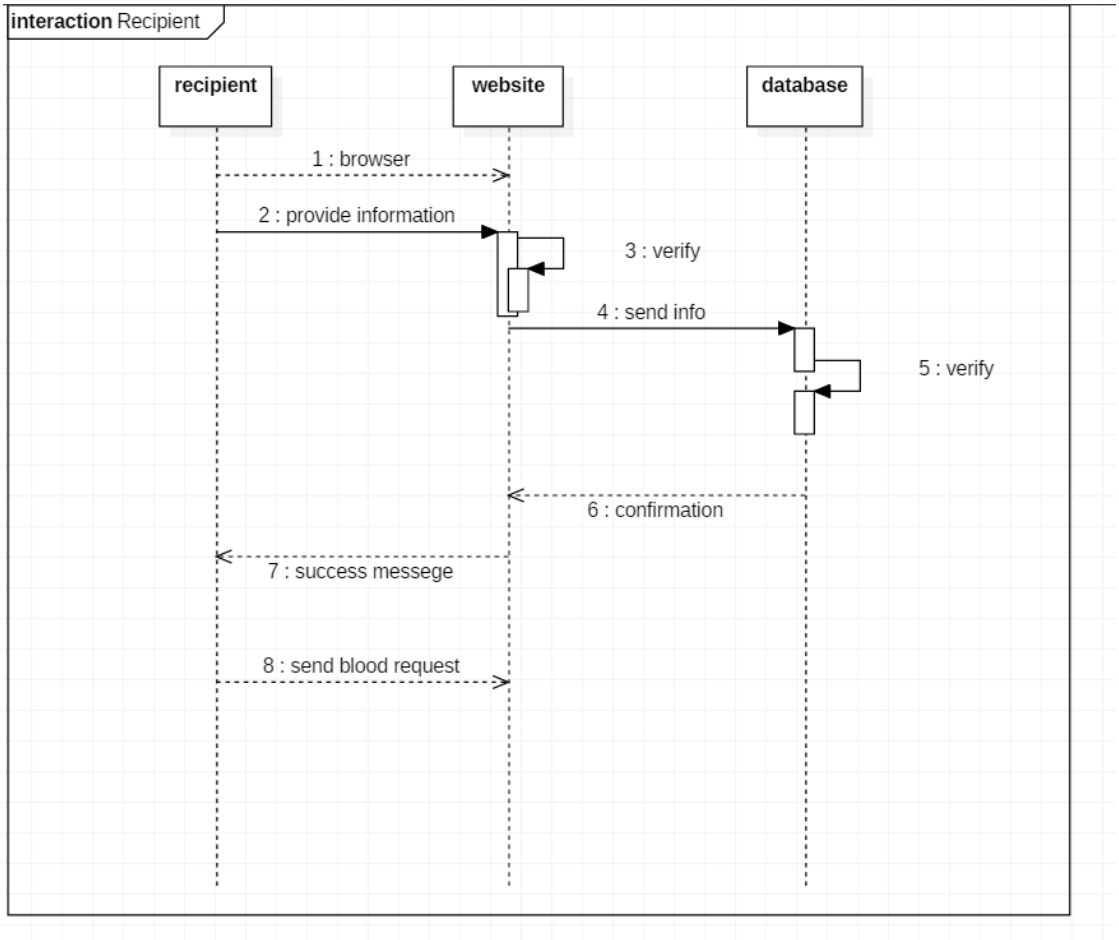

Fig. 9:Sequence Diagram for recipient

Description: Outline the steps that enable a recipient (requester) to update their personal information, and submit a blood request.

Primary Actor: Recipient (Recipient / Requester).

Start: When the recipient accesses the application through the browser.

Pre-condition: The recipient must have an account and be logged in.

Scenario Descriptions :

- ✧ The recipient accesses the site via the browser.
- ✧ The recipient updates their personale information (modifies information).
- ✧ The system (Website) performs a local validation of the fields (validates info).
- ✧ The system sends the updated information to the database (send information).

- ✧ The database validates and records the data (validate info).
- ✧ The database returns a confirmation to the system (confirmation).
- ✧ The system displays a success message to the recipient (success message).
- ✧ The recipient conducts a donor search (search donor).
- ✧ The system executes the search process (searching).
- ✧ The system sends the list of available donors to the recipient (send donor list).
- ✧ The recipient submits a blood request (Send Blood Request).

Sequence Diagram of donor :

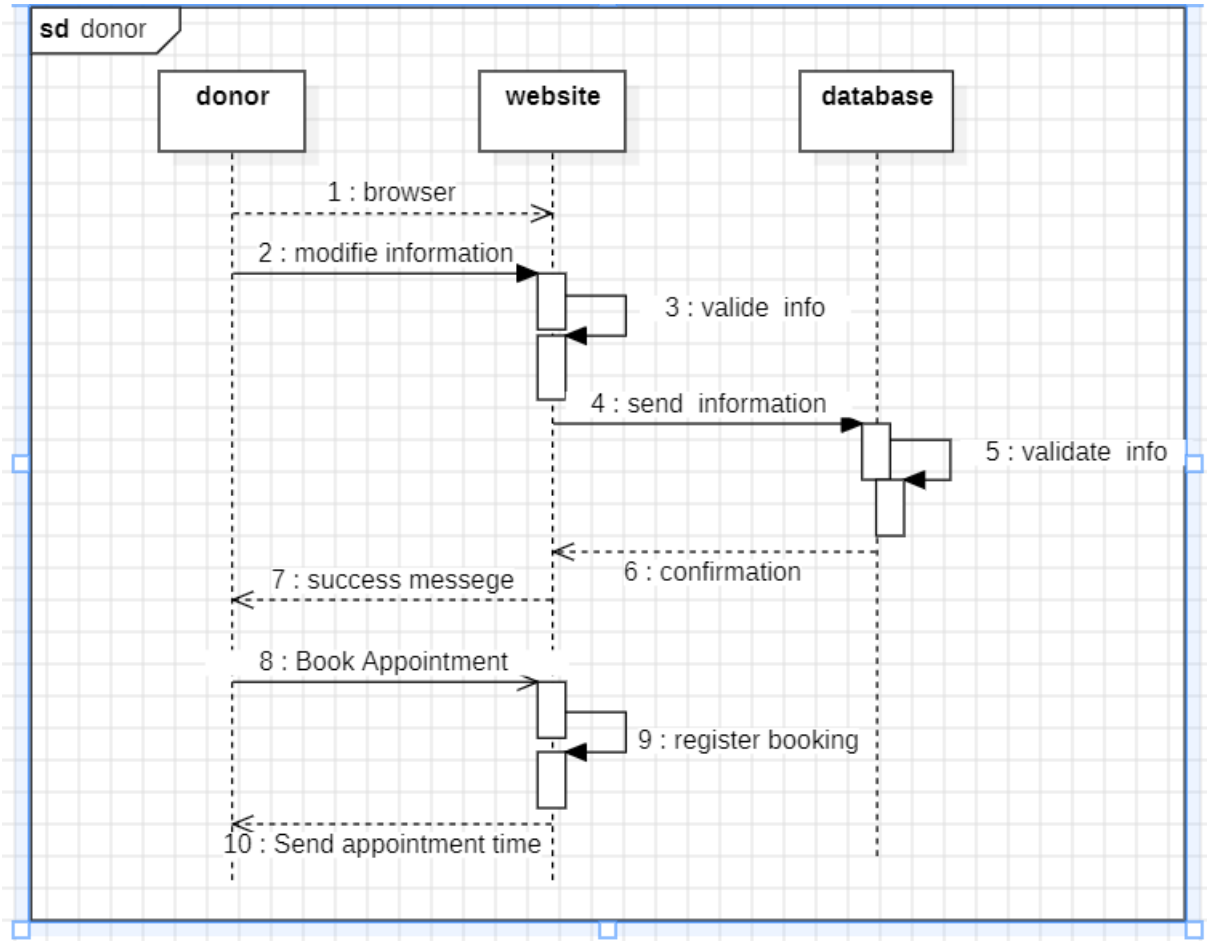

Fig. 10:Sequence Diagram for donor

Description: how the donor work in the system

Primary Actor:

User: (Donor)

Start:

When the donor accesses the website via a browser.

Precondition:

The recipient must have an account and be logged in.

Nominal Scenario (Normal Flow) :

✧ The user navigates to the website using the browser;

- ✧ The user inputs or updates their personal details in the form;
- ✧ The website verifies the validity of the information (local validation);
- ✧ The website transmits the user information to the database;
- ✧ The database authenticates and saves the information;
- ✧ The database sends a confirmation back to the website;
- ✧ The website displays a success message to the user.

Sequence Diagram of admin :

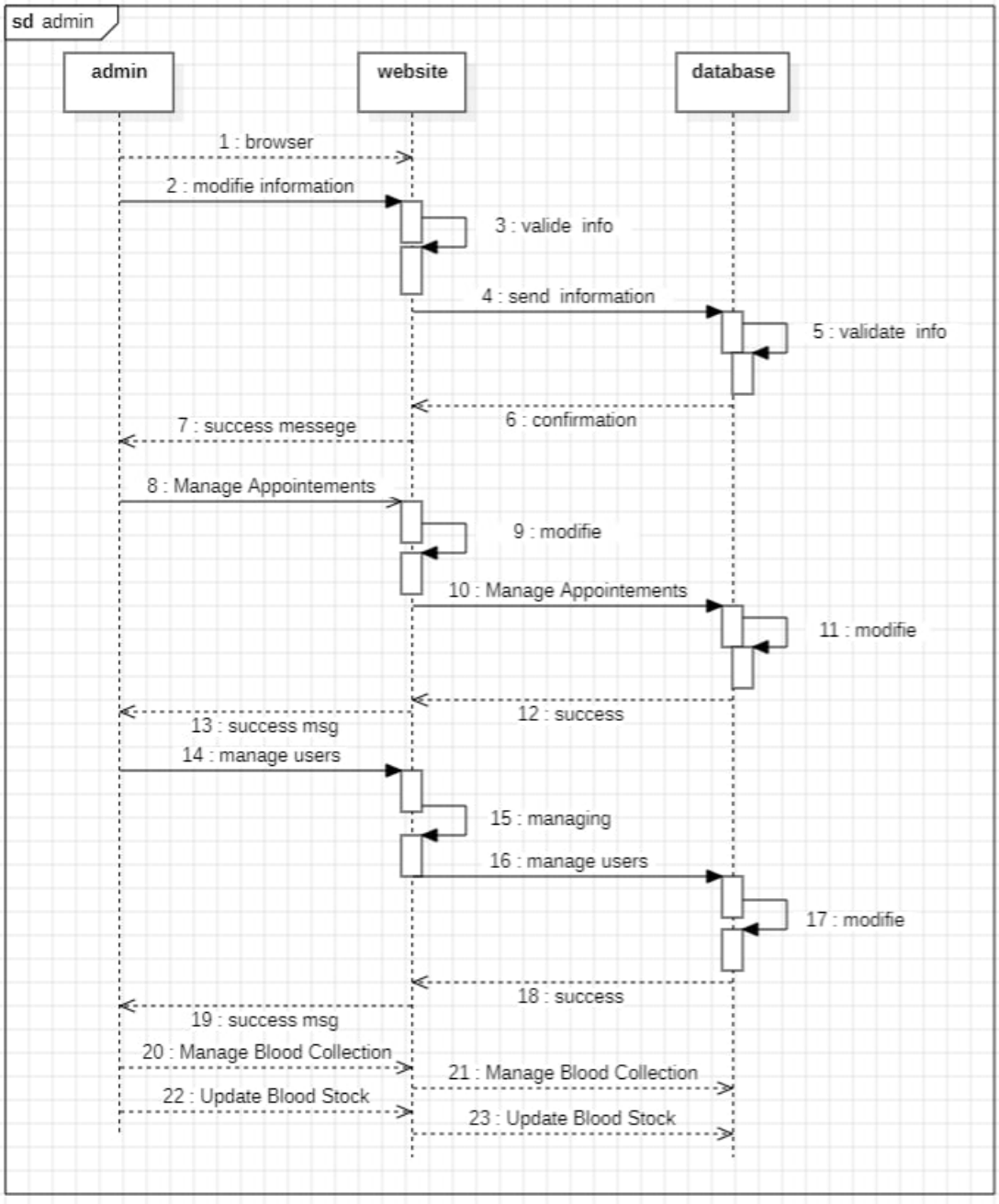

Fig. 11:Sequence Diagram for admin

Description: Outline the technical procedures that allow the administrator to oversee system data, including information, appointments, users, and blood stock.

Primary actor: Administrator.

Start: When the administrator accesses the website via the browser.

Precondition: The administrator account is already established, and the system is functioning.

Scenario description :

Nominal scenario :

- ✧ The administrator accesses the website (browser).
- ✧ The administrator modifies information (update data).
- ✧ The system validates the information (validate info).
- ✧ The system transmits the updated information to the database (send information).
- ✧ The database verifies and updates the data (validate info).
- ✧ The database sends a confirmation back to the system (confirmation).
- ✧ The system presents a success message to the administrator (success message).
- ✧ The administrator manages appointments (manage appointments).
- ✧ The system processes and transmits updates to the database.
- ✧ The database updates the appointment data and returns a success response.
- ✧ The system displays a success message.
- ✧ The administrator manages users (manage users).
- ✧ The system sends user management operations to the database.
- ✧ The database updates user data and provides confirmation.
- ✧ The system shows a success message.
- ✧ The administrator oversees blood collection and updates blood stock.
- ✧ The system transmits these updates to the database.
- ✧ The database successfully stores the updates.

Alternative scenario:

If the information is invalid:

The system displays an error message and prompts the administrator to rectify it.

If a database error occurs:

The system does not receive confirmation and presents an error message.

2.4. Conclusion

In conclusion, this chapter has allowed us to lay the necessary groundwork for the creation of our Bloodlink website. We have outlined the functional and non-functional requirements of our system. We then presented the UML language, the modeling tool (StarUML), and the various diagrams that will enable the design and implementation of the software.

Chapter 3: Implementation

3.1. Introduction

In this chapter, we will focus on the implementation phase, our goal is to provide an overview of the development environments used to build the application, and describe the interface navigation scheme.

3.2. System Requirements

The system requirements are divided into two categories: user-side configuration and server-side configuration:

- Server Requirements:

  The server must be a PHP-compatible SQL server.

  The server must host a relational database containing the tables used.

  The database must be able to communicate with the user-side software.

- User Requirements:

  A computer capable of running all the necessary software.

3.3. System architecture

To guarantee the success of a high-quality IT project, it is essential to meticulously choose the right software tools. Although it is not correct to assert that one programming language is fundamentally better than another, some languages may be more appropriate for particular tasks. For our web application, we have employed the following software environment:

Front-end:

- HTML

- CSS

- JavaScript

Backend:

- PHP

3.3.1. HTML

HTML, or Hyper Text Markup Language, serves as the foundational language for creating and structuring the content of web pages. It constitutes the essential framework of any website, facilitating the organization of text, images, and other elements that are visible in your browser.

3.3.2. CSS

CSS, which stands for Cascading Style Sheets, is a programming language utilized to define the visual presentation of HTML and XML documents. It facilitates the management of the appearance and layout of web page elements, thereby distinguishing content (HTML) from its presentation (CSS).

3.3.3. MySQL

MySQL is an open-source relational database management system (RDBMS) utilized for data storage and management. Its reliability, performance, scalability, and user-friendliness make MySQL a favored choice among developers. In fact, it is found at the core of demanding, high-traffic applications such as Facebook, Netflix, Uber, Airbnb, Shopify, and Booking.com.

3.3.4. JavaScript

JavaScript is a versatile and dynamic scripting language. It plays a crucial role in the web. Established in 1995 by Brendan Eich, it enhances interactivity on websites. It also provides complex functionalities. It is an essential tool for today's web developers.

3.3.5. PHP

PHP est un langage de programmation compilé à la volée très utilisé pour produire des pages Web dynamique. C'est un langage très abordable qui va vous permettre de créer vos premiers site web dynamiques.

3.4. Software architecture

Software architecture pertains to the comprehensive framework of a software system ,and the arrangement of its various elements. It establishes a foundation for the design development, and maintenance of computer applications by outlining how the different components of a system interact with one another.

The objective of software architecture is to delineate the allocation of responsibilities facilitate communication among the various modules, and manage dependencies. It offers ,a holistic view of the system and informs design choices to guarantee the quality maintainability, and performance of the website.

Presentation Layer (First Level):This layer corresponds to the user interface of the BloodLink website, which is accessible to donors, requesters, and blood bank administrators through a web browser. It was created using HTML, CSS, and JavaScript technologies to ensure the design of the web pages, interactivity, and an improved user experience. Furthermore, the development environment utilized for implementing this layer is Visual Studio Code, which aids in coding, debugging, and project management.

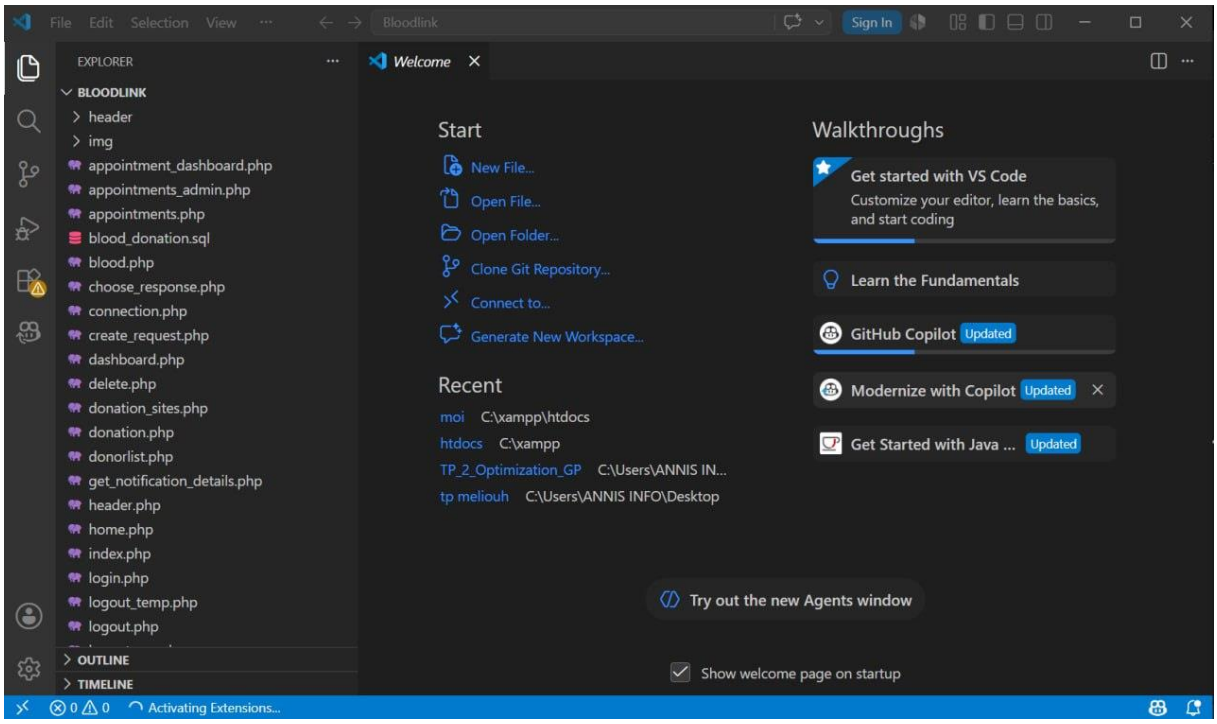

Fig. 12:Visual studio code interface

Business or Functional Layer (Second Level):

This layer represents the business components of the BloodLink website that manage the essential functionalities of the system, such as donor registration, donor matching, blood request management, appointment scheduling, and notification dispatch. These business components were developed using PHP and executed locally through XAMPP, which provides the Apache server and the necessary environment for running the application.

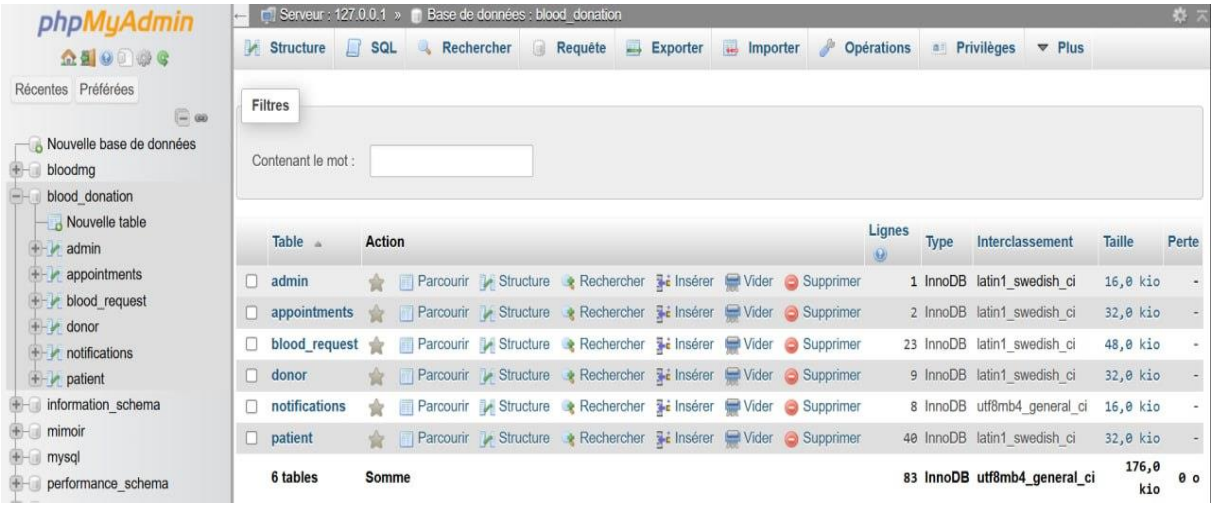

Fig. 13:PhpMyAdmin interface

Data Access Layer (Third Level):

This layer is tasked with managing access to the data of the BloodLink website, which includes the database used to store information on donors, requesters, blood centers, and available blood supplies. The database management was conducted using MySQL through XAMPP, ensuring dependable data storage, retrieval, and management for the system.

3.5. Interface overview

In this section, we will present the main interfaces of our site, designed according to ergonomic principles in order to make operations simple and intuitive for every user.

3.5.1. Home page

Fig. 14:Bloodlink interface

Upon the launch of the site, the homepage will serve as the initial interface visible to all stakeholders involved in our project.

3.5.2. Admin login

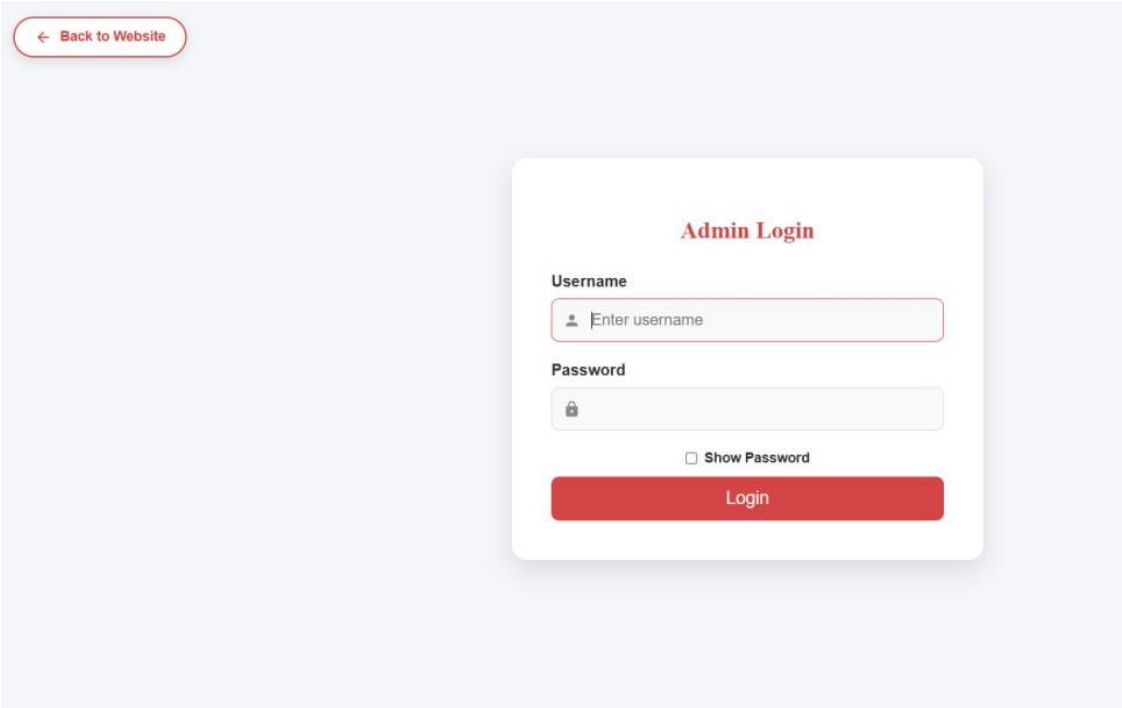

Fig. 15:Admin Login

This section is specifically designated for administrators to log in and access the administrator page.

3.5.3. User login

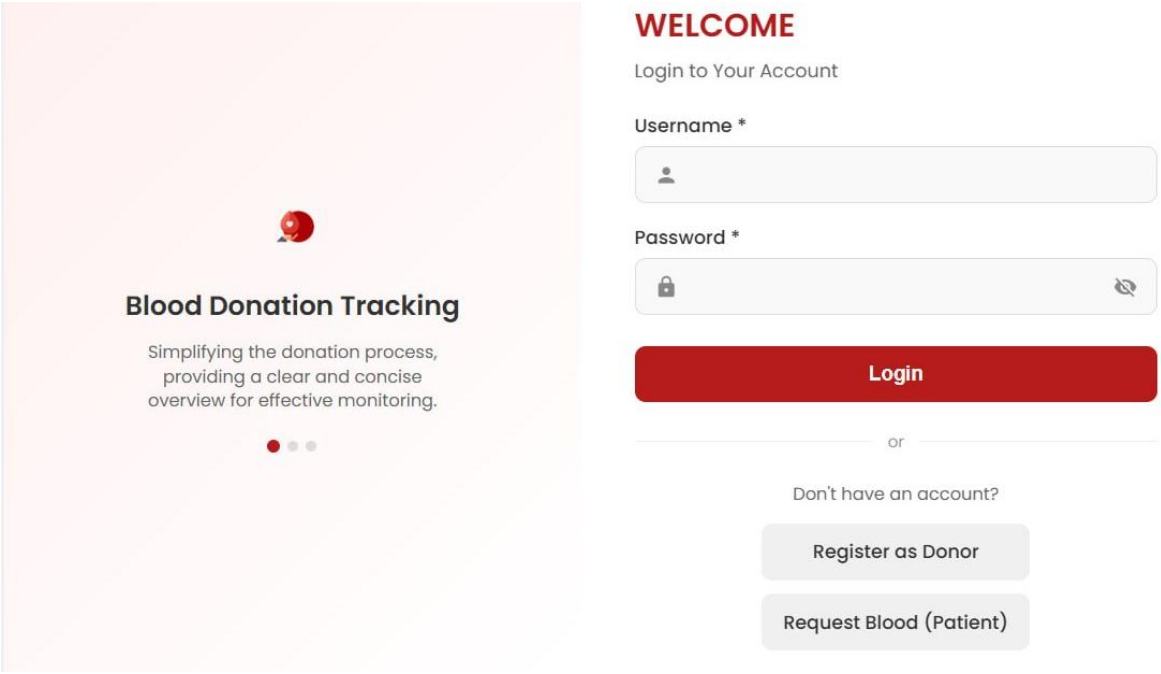

Fig. 16:User Login interface

Users who possess an existing account will be automatically directed to the login page, where they must enter their email address and password to gain access to their personal account. Those without an account will have the opportunity to register by selecting the "Register as donor" or "patient" button.

3.5.4. Donor registration

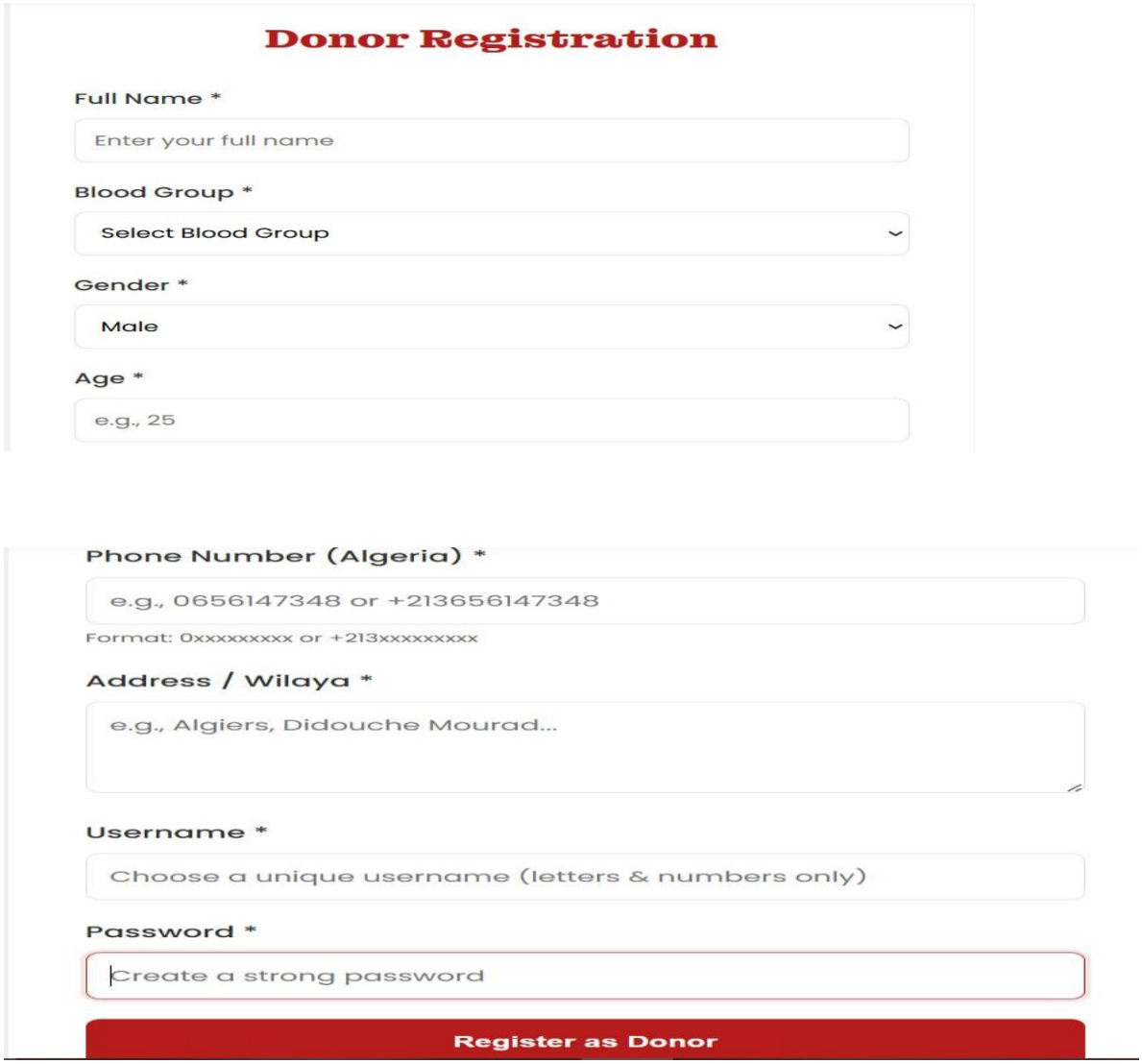

Fig. 17:Donor registration

3.5.5. Request Blood

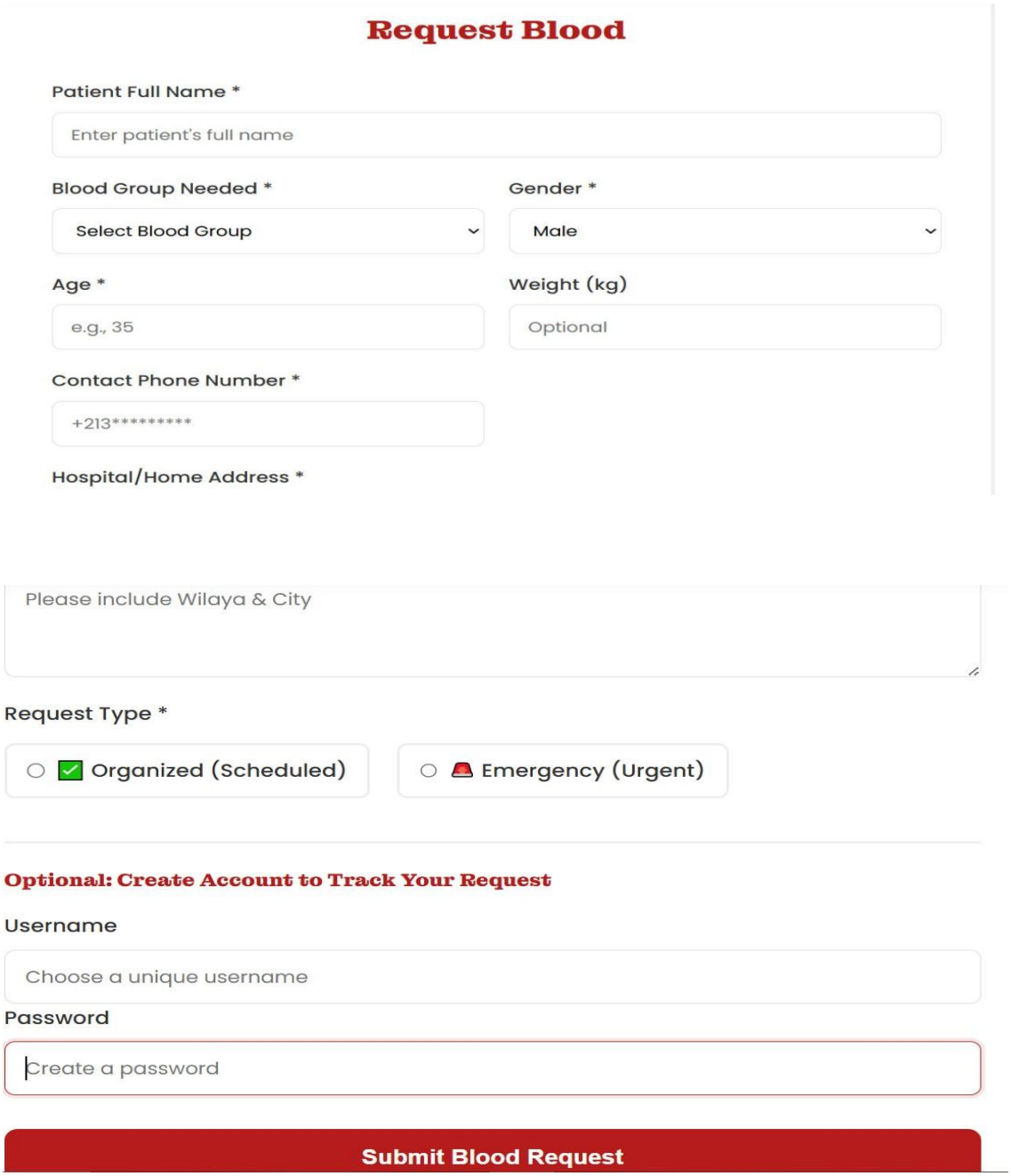

Fig. 18:Request Blood

3.5.6. Admin home

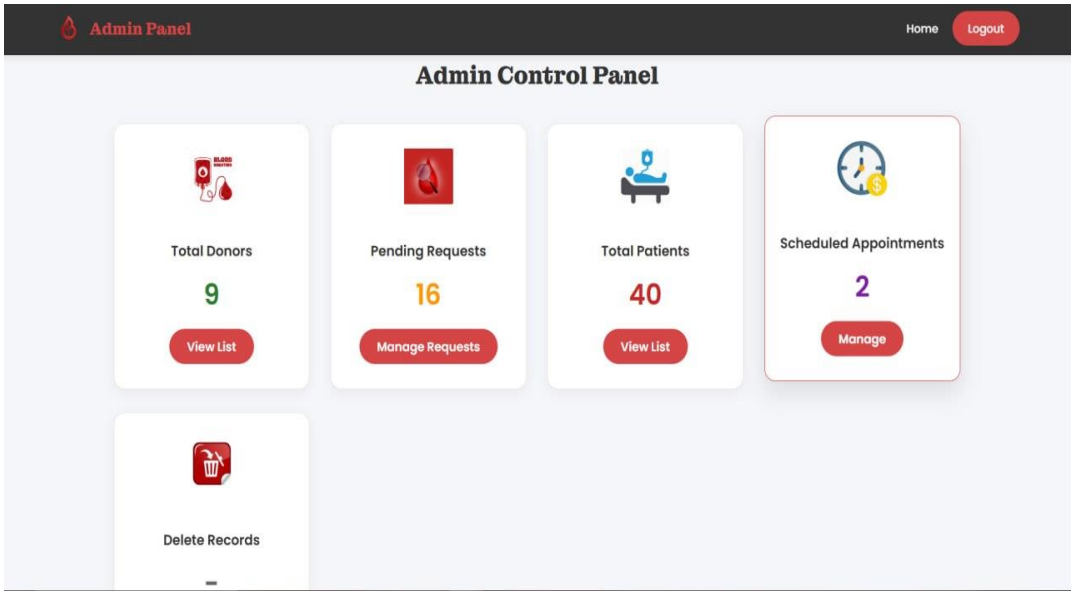

Fig. 19:Admin panel

This page provides the administrator with the capability to oversee blood inventory and observe critical statistics, including the overall count of donors, donations completed today, blood requests submitted, and those requests that have been approved, thereby facilitating effective management of blood supply.

3.5.7. Donation appointment

Fig. 20:Donation appointement interface

This page enables the donor to schedule an appointment at the nearest blood donation center. The user selects a center from the available list, then fills in their information before confirming the appointment. This feature streamlines the organization of blood donations and enhances access to transfusion centers.

Conclusion

In this chapter, we have discussed the practical aspects of creating our website, emphasizing the essential development tools required to ensure the proper functioning of the blood bank application. Furthermore, we have illustrated several key graphical interfaces of our site.

General Conclusion

The objective of this project was to design and create BloodLink, a web-based system for managing blood donations that enhances communication among blood donors, recipients, and administrators. The proposed platform offers an effective solution for overseeing blood requests, donor details, appointments, and notifications, all while ensuring a user-friendly and accessible interface. In the course of this project, we assessed the requirements of various stakeholders, modeled the system through UML diagrams, and developed a functional web application utilizing contemporary web technologies. The resulting solution aids in optimizing the organization of blood donation activities and minimizes the time needed to connect patients with suitable donors, particularly in urgent circumstances. This project also provided an opportunity to apply the knowledge gained during our academic studies in software engineering, database management, system analysis, and web development. Additionally, it underscored the significance of digital technologies in enhancing healthcare services and tackling real-world issues.

Future Perspectives While BloodLink fulfills its primary goals, there are several enhancements that could be explored in future endeavors. The system could be improved by incorporating real-time geolocation services to pinpoint the nearest available donors. Development of mobile application support could also be pursued to boost accessibility and user engagement. Moreover, implementing SMS and email notification services could facilitate quicker communication during emergencies. Future iterations may also integrate artificial intelligence techniques to automatically suggest suitable donors and forecast blood demand trends. Lastly, fostering stronger partnerships with hospitals, blood banks, and healthcare organizations could further enhance the platform's effectiveness and impact. In summary, BloodLink signifies a substantial contribution to the digital transformation of blood donation services and illustrates how technology can aid humanitarian efforts..

References

[1] Diagramme d'activités UML. Available at: https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-activites. Accessed on: 12 /01/ 2026.

[2] IBM Documentation – UML Sequence Diagrams. Available at: https://www.ibm.com/docs/fr/rsas/7.5.0?topic=uml-sequence-diagrams. Accessed on: 15 /01/ 2026.

[3] Blood Components. Available at: https://www.blood.co.uk/why-give-blood/how-blood-is-used/blood-components/. Accessed on: 18 /01/ 2026.

[4] Unified Modeling Language (UML) Class Diagrams. Available at: https://www.geeksforgeeks.org/system-design/unified-modeling-language-uml-class-diagrams/. Accessed on: 22 /01/2026.

[5] Guide to Functional Requirements. Available at: https://www.modernrequirements.com/blogs/guide-to-functional-requirements/#elementor-toc__heading-anchor-1. Accessed on: 27 /01/ 2026.

[6] The Role of Technology in Blood Donor Management. Available at: https://medium.com/@degree37/the-role-of-technology-in-blood-donor-management-d7db0d2a3605. Accessed on: 30 /01/ 2026.

[7] What are Non-Functional Requirements Examples. Available at: https://www.perforce.com/blog/alm/what-are-non-functional-requirements-examples. Accessed on: 03 /02/ 2026.

[8] Blood Biochemistry. Available at: https://www.britannica.com/science/blood-biochem. Accessed on: 07 /02/ 2026.

[9] Can I Donate Blood? Available at: https://www.hsa.gov.sg/blood-donation/can-i-donate. Accessed on: 10 /02/ 2026.

[10] BloodLinks – About. Available at: https://bloodlinks.in/about. Accessed on: 14 /02/2026.

[11] Projet Banque de Sang Électronique. Available at: https://fr.scribd.com/document/847924329/. Accessed on: 18 /02/2026.

[12] Zomraty: E-Blood Bank Android Application. Available at: https://www.semanticscholar.org/paper/Zomraty%3A-E-Blood-Bank-Android-Application-for-and-Oukebdane-Ghouali/1fd75440d71959380ffc5a52df1f523695eee008. Accessed on: 22 /02/ 2026.

[13] Blood Donation Research Papers. Available at: https://www.semanticscholar.org/search?q=blood%20donation&sort=relevance. Accessed on: 27 /02/ 2026.

[14] Conception and Realization of a Blood Donation Platform. Available at: https://theses-algerie.com/1099260831133055/memoire-de-master/universite-mohamed-boudiaf---m-sila/conception-and-realization-of-a-blood-donation-platform. Accessed on: 02 /03/ 2026.

[15] Ahyaaha Platform. Available at: https://www.ahyaaha.com/client/. Accessed on: 08 /03/ 2026.

[16] Blood Donation Management Website. Available at: https://blood2018.epizy.com/index2.php?i=1. Accessed on: 15 /03/2026.