**ПРЕДМЕТ: РАЗРАБОТКА НА СОФТУЕР**

**ДЕЙНОСТ: УЧЕНИЧЕСКИ ПРАКТИКИ 2**

***2021-2022***

**Тема: „Система за менажиране на кръвни дарения в болница“**

**Курсов проект**

*Автори:*

*Никола Димитров Пешев, клас XI В*

БургасСъдържание

[1 Въведение 3](#_Toc96509792)

[2 Цели и обхват на софтуерното приложение 3](#_Toc96509793)

[3 Анализ на решението 3](#_Toc96509794)

[3.1 Потребителски изисквания и работен процес 3](#_Toc96509795)

[3.2 Примерен потребителски интерфейс 4](#_Toc96509796)

[3.3 Диаграми на анализа 4](#_Toc96509797)

[3.4 Модел на съдържанието / данните 5](#_Toc96509798)

[4 Дизайн 5](#_Toc96509799)

[4.1 Реализация на структура на приложението (3-layer), Разделение на кода според предназначението му 5](#_Toc96509800)

[4.2 Организация и код на заявките към база от данни 5](#_Toc96509801)

[4.3 Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб) 5](#_Toc96509802)

[5 Тестване 6](#_Toc96509803)

[6 Заключение и възможно бъдещо развитие 6](#_Toc96509804)

[7 Използвани литературни източници и Уеб сайтове 6](#_Toc96509805)

[8 Приложения 6](#_Toc96509806)

[9 Критерии и показатели за оценяване 8](#_Toc96509807)

# Въведение

Темата ми за проект е относно нещо, което всеки (при възможност) трябва да направи поне веднъж, ако не и редовно. Кръвните дарения спасяват животи. Важно е те да могат да се правят, менажират и получават лесно. Моят проект решава проблема с управлението на всичката тази информация, създавайки централизирана система, визуализираща всичката информация пред очите на работниците. Използвайки езика UML и средата за моделиране Lucid Chart успях да създам своите диаграми. Те са няколко вида и всички ми помогнаха да стоя организиран и по план докато работех над кода на проекта.

# Цели и обхват на софтуерното приложение

След като минах курс на Българският Червен Кръст в град Бургас се замислих за всички лесно предотвратими смъртни случай сред нас. Дори и те да не са лесно предотвратими, при загуба на кръв, винаги може да бъде от полза пакет дарена кръв за използване. Така реших да направя тази система за менажиране на кръвни дарения. Тя ще улесни и съкрати работата на медицинските работници и ще предостави лесен за употреба интерфейс за дарителите, за да виждат как са допринесли. Това приложение ще въздейства на собствениците, работниците и посетителите на една болница. Ще намали много съпротивлението от мисълта за извършване на дарение до виждане на резултат в онлайн платформата. Обхваща всички от собственика на болницата до пациентите. С приложението се цели да се направи дарението на кръв по-удовлетворяващо и да се насърчи тази дейност.

# Анализ на решението

## Потребителски изисквания и работен процес

Работният процес представлява въвеждане на малка информация относно пациентите и тяхната обработка.

Вход на данни:

* Създаване на акаунт
* Въвеждане на кръвна група и персонална информация

Обработка на данни:

* Валидация на акаунт и парола
* Валидация на „Access token” и при нужда „Refresh Token”
* Хеширане на паролите на потребителите – включително работници и пациенти
* Криптиране на чувствителните лични данни на пациентите и работниците

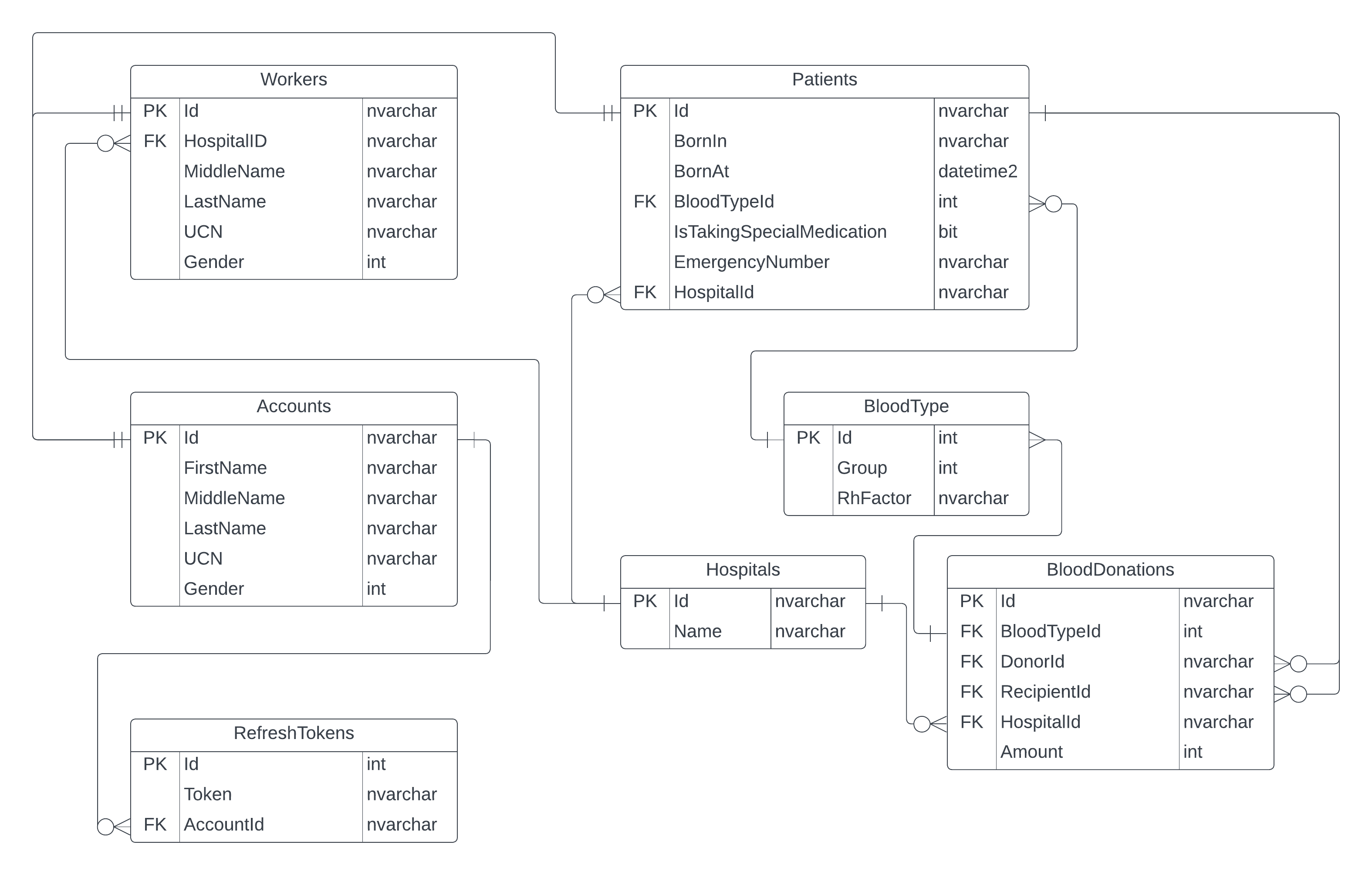
Изход на данни:

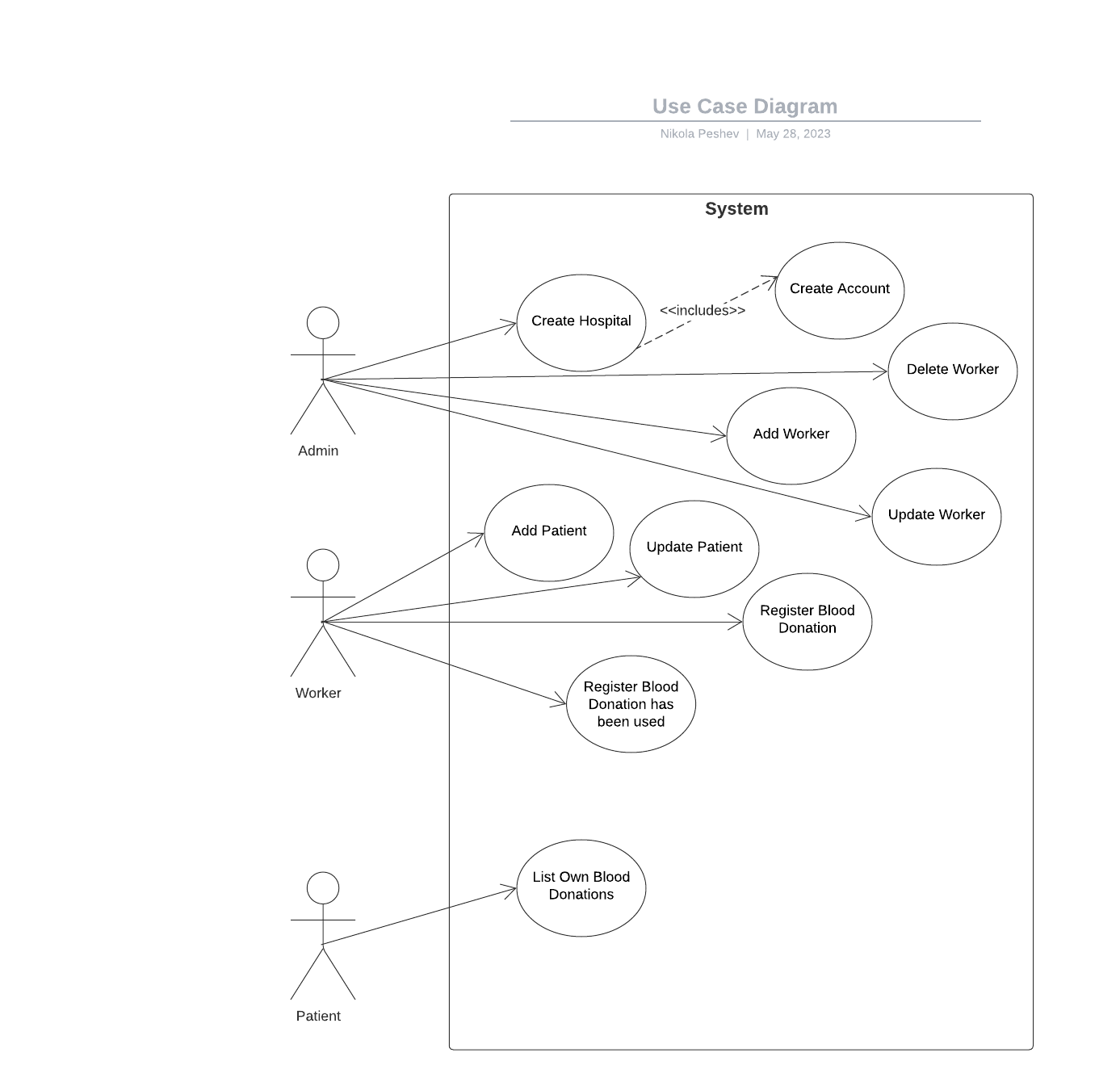
* Визуализация чрез таблици и уеб сайт
* Използват се от всички потребители на системата

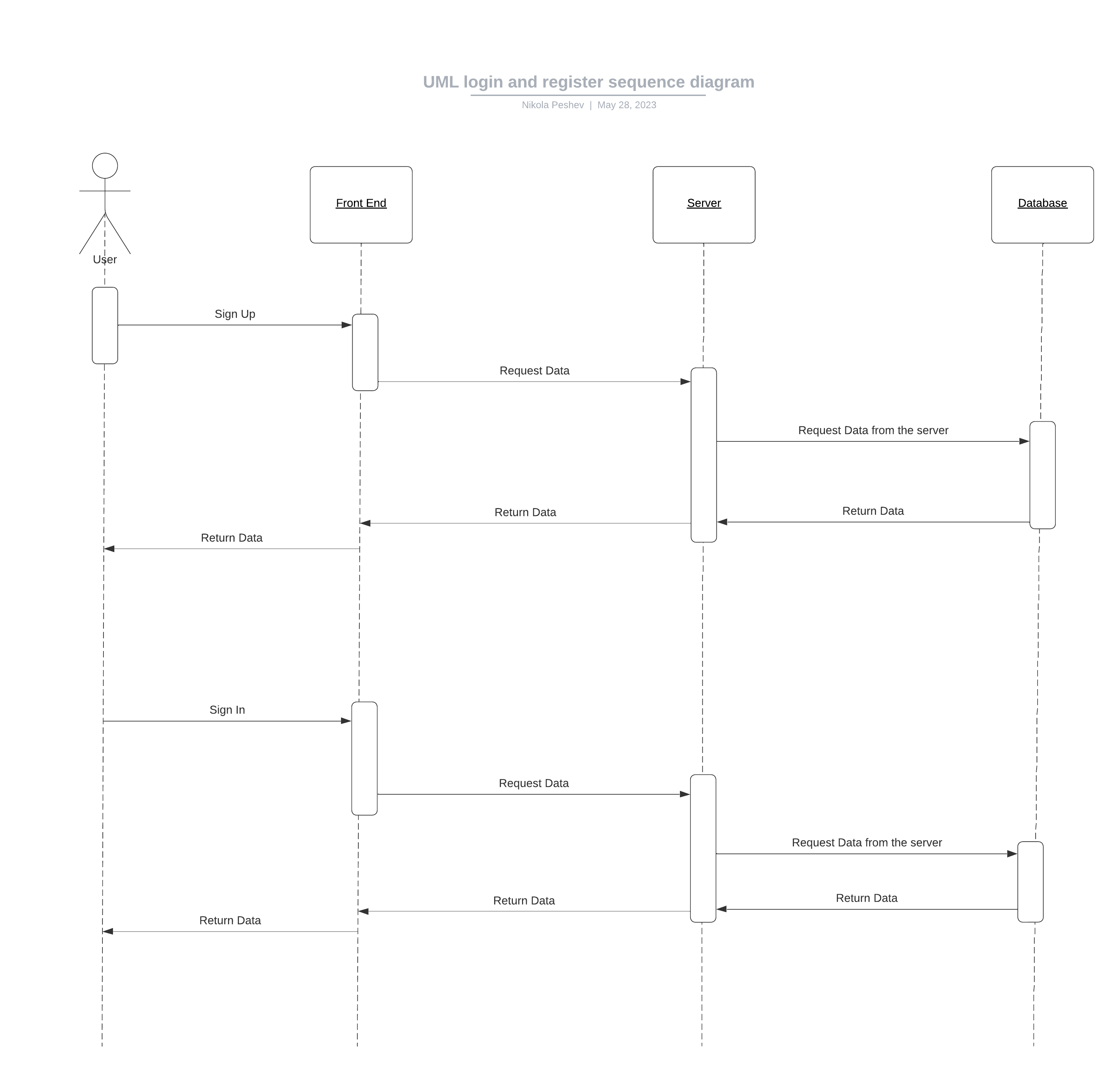
## Примерен потребителски интерфейс

Графичния интерфейс ще бъде изграден чрез употребата на библиотеката „React“ написана на езика за програмиране „Javascript“. Ще представлява доста таблици за визуализиране на кръвните дарения и всичките пациенти. Също така ще има страници и форми за създаване и управление на ресурсите на болницата като самите кръвни дарения и пациентите. Пациентите ще могат да отворят страница, към която получават достъп чрез собствен акаунт, за да видят кръвните си дарения изобразени в табличен вид.

## Диаграми на анализа







## Модел на съдържанието / данните

Всички чувствителни данни въведени от потребителите на системата се криптират за да бъдат защитени от пробив в базите данни. Информацията се пази на Microsoft SQL Server базирана база от данни. Използва се Microsoft Entity Framework за мапиране и създаване на всичките релационни данни.

# Дизайн

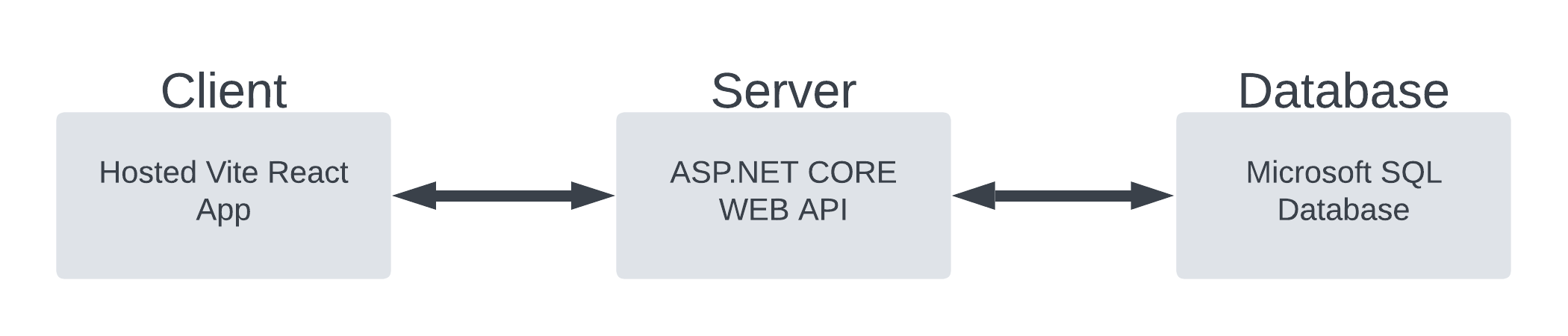
За визуалната част използвам „Javascript” библиотеката „React” за менажиране на състоянието на програмата и всичките променливи. Относно дизайна:

За дизайна на приложението е важно да бъде възможно най-изчистен и семпъл. Достъпност е приоритет номер едно при създаване на апликация от подобно естество. Нужно е платформата да бъде достъпна за потребители от всякакъв тип използвайки възможно най-голямо разнообразие от входни устройства. Използването на семантични html тагове и слагането на alt html атрибут на снимките например помага за достъпността на приложението.

Сървърната структура е трислойна и е използван „ASP.NET CORE WEB API” за направата на „API“.

Разделена е на Data, Services и WebHost части. Те се отнасят съответно за данните, бизнес логиката и откриването на API-я.

В github ще бъдат създадени специални процеси за постоянно деплойване на платформата. За целта ще се използва платформата на Microsoft – Azure.



## Реализация на структура на приложението (3-layer), Разделение на кода според предназначението му

Сървърната част на проекта представлява трислойна архитектура съставена от WebHost, Services и Data слой. Те отговарят съответно за представянето на API-то, бизнес логиката и обработката и запазването на данните.

## Организация и код на заявките към база от данни

За заявките към базата данни отговаря Microsoft Entity Framework. Конфигурирана е по специален начин за разделяне на заявките, което подобрява производителността. Всичко това е организирано в класова йерархична структура.

Описание на инструментариума за достъп до базата данни от гледна точка на програмния код. Описание на методите за извличане, добавяне и изтриване на обекти в базата данни.

## Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб)

Приложението е проектирано да бъде изключително лесно-достъпно и интуитивно. Предназначено е за широка аудитория, което изисква висока достъпност и адаптивност на потребителския интерфейс. Приложението ще поддържа добавяне, триене и редактиране на информация.

# Тестване

За юнит тестване на .Net частта на проекта ще се използва популярната библиотека „XUnit.Net”. Там ще бъдат структурирани и написани всички тестове по кода. За „React” частта ще се правят юнит тестове чрез jest и end-to-end тестове чрез Playwright. Две библиотеки създадени специално за правене на тестове. Чрез Playwright ще се направят тестове за мобилни устройства, таблети, даже и „ultrawide” монитори.

# Заключение и възможно бъдещо развитие

В заключение, приложението ще е от голяма полза на медицинските работници и ще представлява като мотивация на хората да правят кръвни дарения, което ще подпомогне медицината. Доволен съм от избора си на технологии поради високото им ниво на интеграция с други полезни инструменти относно документация, тестове, бази данни и други. Използването на трислойна архитектура ми помогна да стоя организиран и фокусиран върху това, което разработвам на момента. Като алтернатива за следващия път бих пропуснал създаването на API и бих работил директно с Next js и React за създаване на логиката и визуалната част обединено. Това ще ми помогне да се движа много по-бързо и да пропусна доста от стъпките, които трябваше да правя сега. Смятам, че разширението на моето приложение би било много реалистично, постижимо и лесно, поради трислойната архитектура и добрите стандарти дадени от Microsoft за уеб разработка. При спазване на техните конвенции кода остава чист и ясен дори за хора нови към проекта. Развитие във функционалността на приложението би се отразило добре. Например записване на часове или нещо друго по-интересно за пациентите и като цяло потребителите на уебсайта.

# Използвани литературни източници и Уеб сайтове

1. Уеб сайт на Microsoft, [линк](https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-7.0)
2. Уеб сайт на React, [линк](https://react.dev/reference/react)
3. Уеб сайт на Swagger, [линк](https://swagger.io/docs/specification/about/)
4. Литературен източник 4
5. Литературен източник 5

# Приложения

# Критерии и показатели за оценяване

|  |  |  |
| --- | --- | --- |
| **Показател** | **точки** | **срок** |
| 2. [Цели и обхват на софтуерното приложение](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Цели_и_обхват)  3.1 [Потребителски изисквания и работен процес](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Потребителски_изисквания_и) | 5  5 | 04.03.2022 |
| 3.2 [Примерен потребителски интерфейс](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Примерен_потребителски_интерфейс)  3.3 [Диаграми на анализа](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Диаграми_на_анализа)  3.4 [Модел на съдържанието/данните](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Модел_на_съдържанието) | 5  5  5 | 18.03.2022 |
| 4.1 [Реализация на структура на приложението (3-layer),  Разделение на кода според предназначението му.  Допълване на Class диаграми/3.3/](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 10  10 | 16.04.2022 |
| 4.2 [Организация и код на заявките към база от данни](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 15 | 30.04.2022 |
| 4.3 [Наличие и интуитивност на потребителски интерфейс (конзолен, графичен, уеб)](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Дизайн). | 10 | 31.05.2022 |
| 5. [Наличие и организация на автоматизирани тестове](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Тестване). | 15 | 15.06.2022 |
| 6. Организация на проекта в система за контрол на изходния код и употреба на добри практики (merge requests, code reviews, branching strategy) | 10 | 25.06.2022 |
| 1. [Въведение. Ниво на завършеност на проекта](file:///O:/Admin204.1/Проекти/2021-2022/Ученически%20практики%202/Ученическа%20документация/user_name_Software%20Development.docx#_Въведение) |  | 30.06.2022 |
| Документация на проекта (XML comments, wiki, etc.) | 5 | текущо |
| Презентация на проекта |  | 30.06.2022 |
| Общо | 100 | Финал на първа фаза |