**ПРЕДМЕТ: РАЗРАБОТКА НА СОФТУЕР**

**ДЕЙНОСТ: УЧЕНИЧЕСКИ ПРАКТИКИ 2**

***2021-2022***

**Тема: „LOGID – Warehouse Sorting System“**

**Курсов проект**

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)

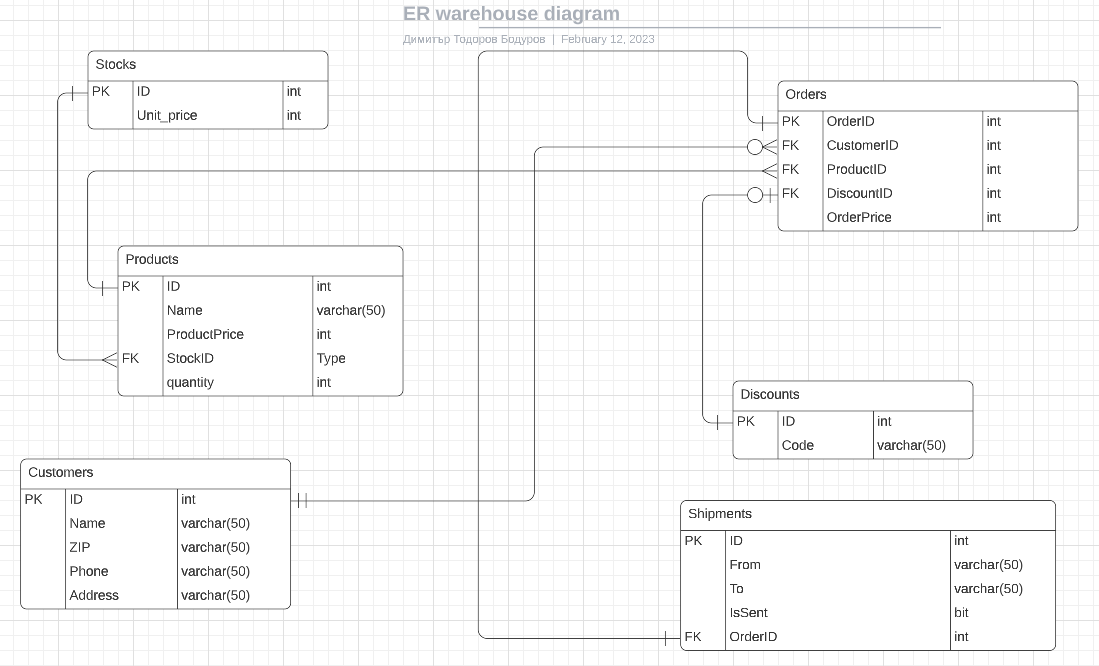
[9 Критерии и показатели за оценяване 8](#_Toc96509807)

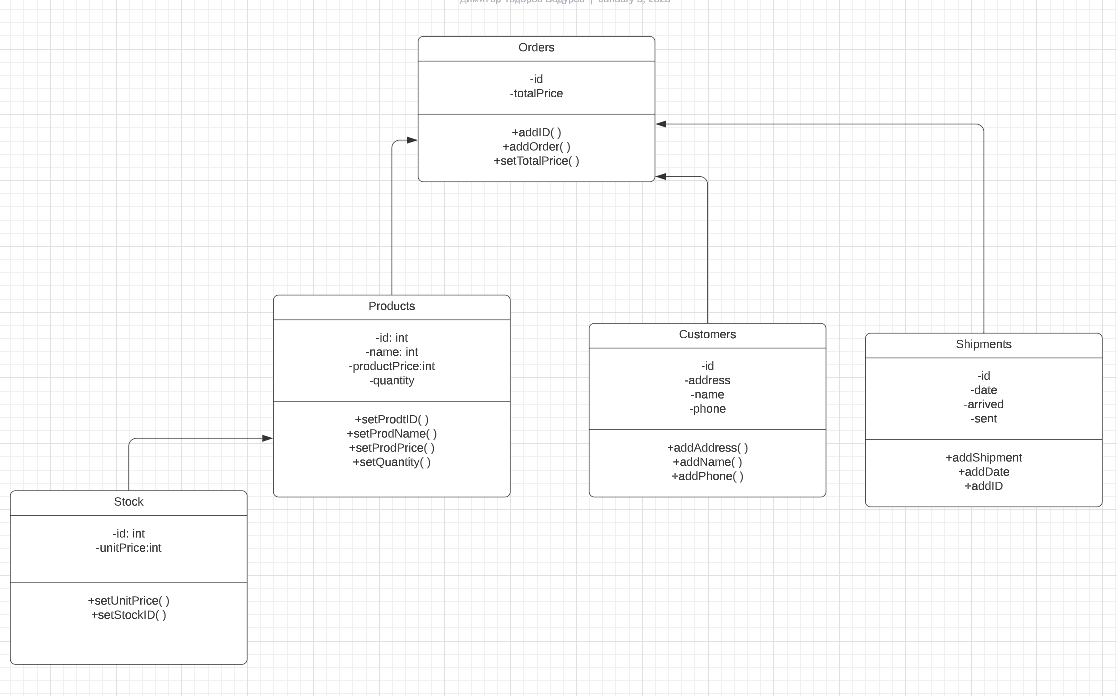
# Въведение

Сортиращите системи са и винаги ще бъдат използвани и актуални. Те се използват навсякъде около нас, в нашето ежедневие и в бизнес средите. Сортирането винаги ще бъде нужно и да имаме приложения ,които могат да сортират всичко, ще улесни работата на всеки.

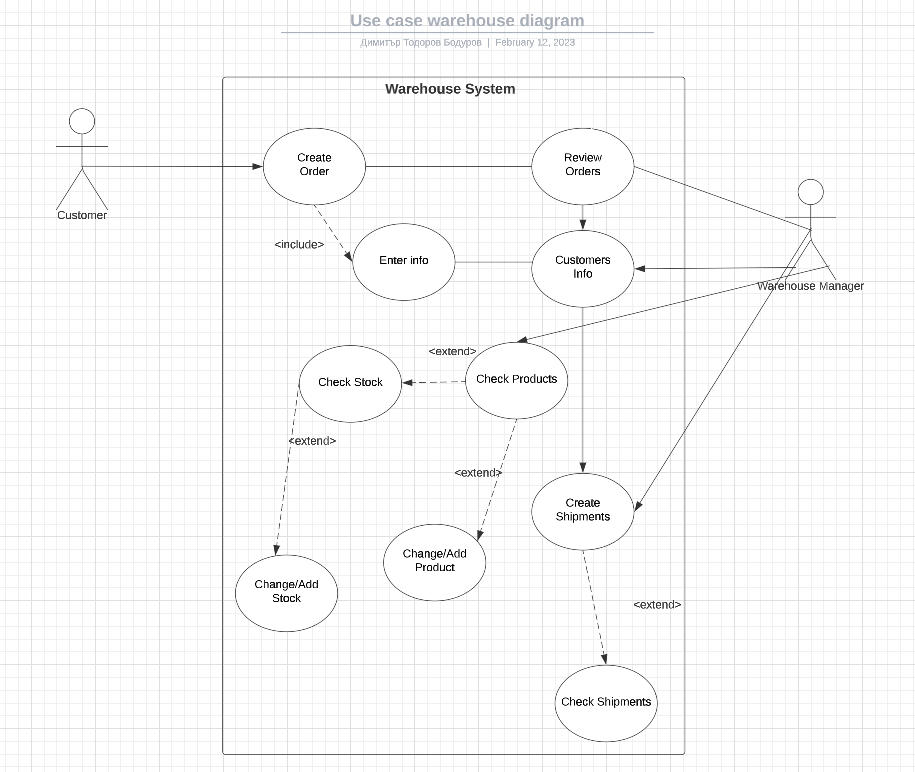
Моят проект ще бъде система ,която ще сортира информацията получена от склад и ще прави информацията по-подредена и по-лесна за разбиране.

***ER diagram***



***Class diagram***

***Use case diagram***



# Цели и обхват на софтуерното приложение

Идеята на Logid е да улесни работата свързана със сортирането в склад. Logid ще направи работата на малки така и на големи бизнеси по лесна чрез сортиране на информация за: получени стоки, стоков запас, купувачи и др. Logid в бъдеще няма да бъде само приложение за складове ,а и за други неща които изискват сортиране.Logid ще е приложението което ще има примерни системи за сортиране на информацията в: складове, ресторанти, болници, книжарници и др. Ще може да се прави и собствена система за сортиране.

Технологиите, които ще бъдат ползвани за направата на това приложение:

C#

Visual Studio

Visual Studio Code

SQL – SSMS

За презентация и документация:

Power point

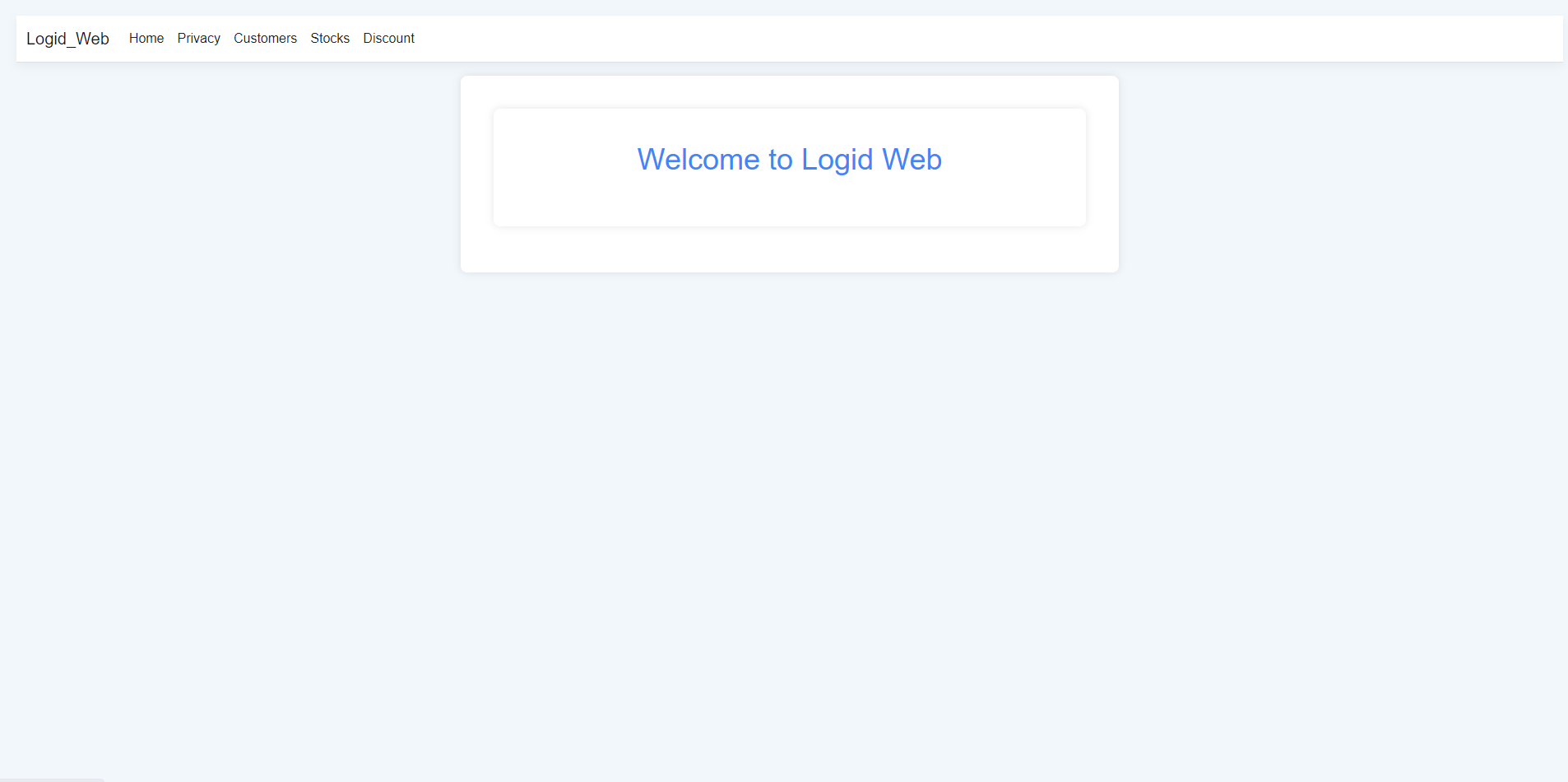
Word

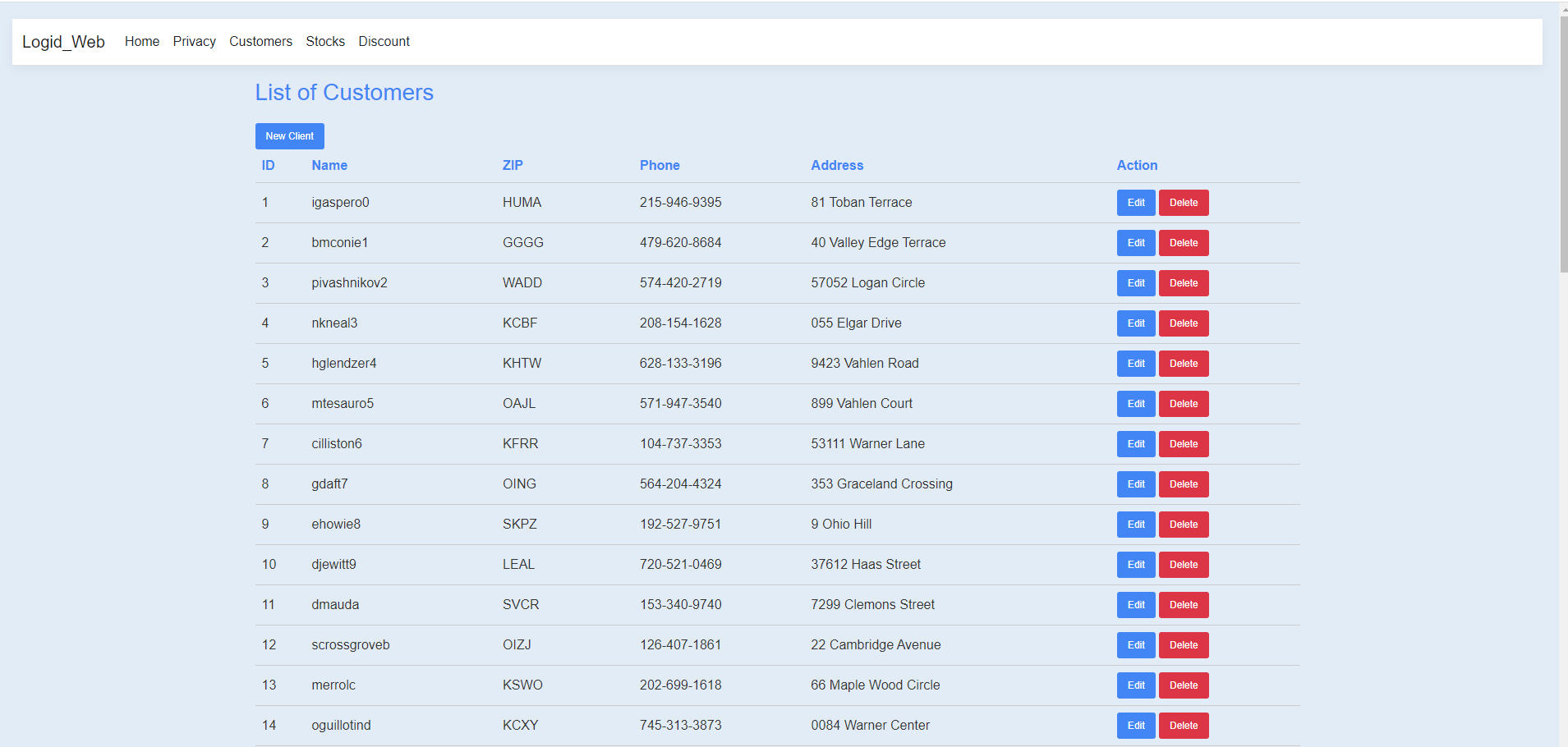
# Анализ на решението

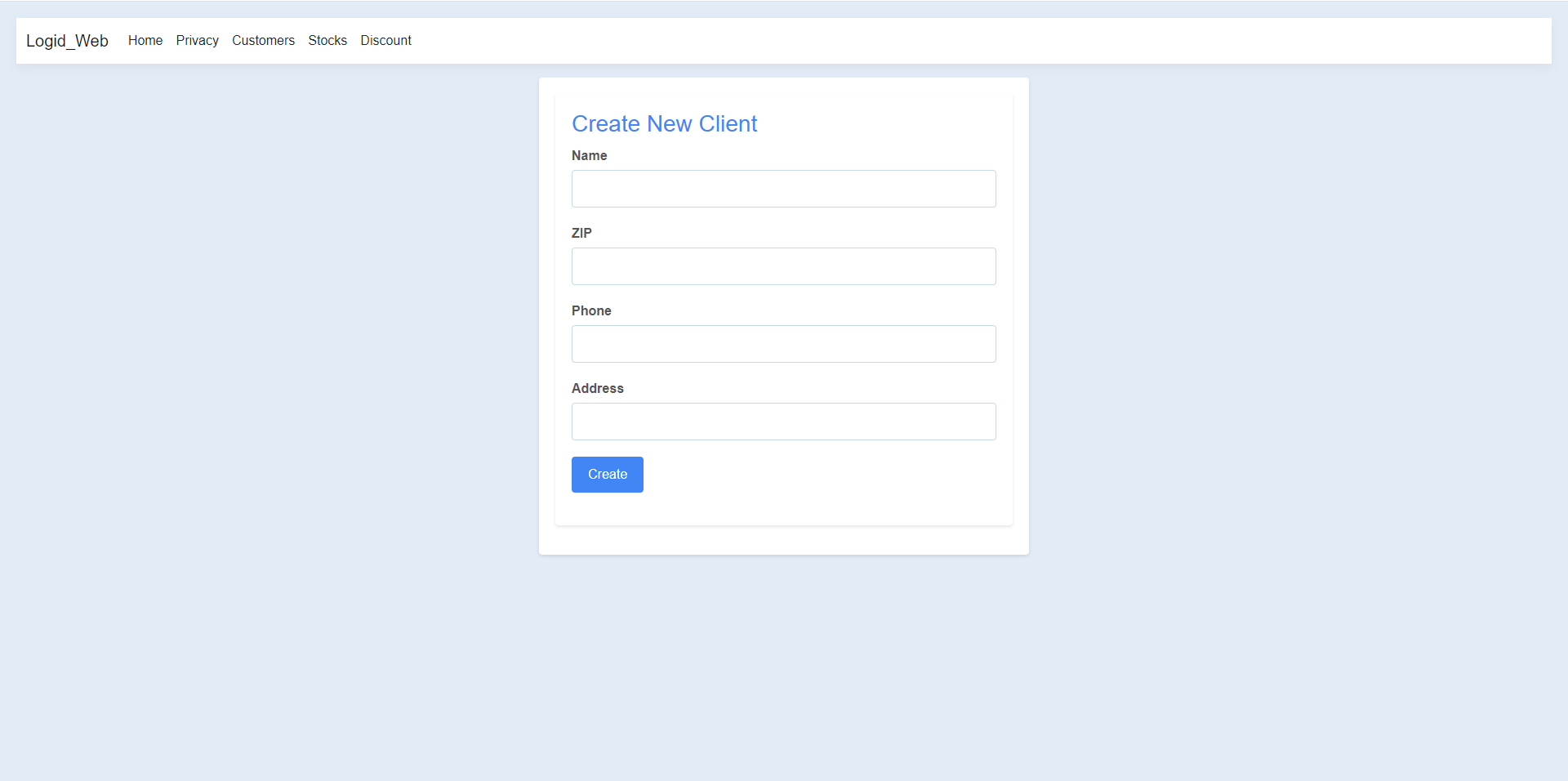
## Потребителски изисквания и работен процес

Потребителят ще трябва да влезе с акаунта, с който потребителя ще може да прави промени или поръчки зависимост дали е Customer или Manager. Customer ще трябва да си направи регистрация и ще може да прави само поръчки и да разглежда каталога. Manager ще може да променя поръчки, да проверява колко стока от определен продукт има в склада, колко свободни превозни средства има (за транспортиране на стока) и др.

## Примерен потребителски интерфейс







## Диаграми на анализа

За да видите как работи проекта чрез UML диаграми, може да погледнете отделните папки, създадени специално за тях в папката ми във файлове в Teams.

## Модел на съдържанието / данните

Като се отвори приложението, първото нещо което потребителя ще види е Log in прозореца. След въвеждане на име и парола, потребителя зависи дали е Manager или Customer, ще имат различни опции. Customer ще може само да разглежда каталога с стоки и ще може да прави поръчки. Ако е Manager, то тогава потребителя ще има възможност да разглежда поръчки, да вкарва данни от Excel таблици, да създава доставки и да въвежда друг тип данни. (На точка 3.2 може да видите набързо направен интерфейс, който ще има много промени в бъдеще)

# Дизайн

В тази секция ще представим дизайна на решението за проекта, използвайки ASP.NET, HTML, CSS и C#. Ще представим софтуерната архитектура на решението, разделението на кода по слоеве, диаграма на разгръщането и диаграми на класовете.

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

Решението е базирано на 3-layer архитектура, която се състои от следните слоеве:

1. Presentation Layer (Слой за представяне) - Този слой се грижи за визуализацията и взаимодействието с потребителите. Включва HTML, CSS код за създаване на потребителския интерфейс.
2. Business Layer (Слой за бизнес логика) - Този слой съдържа логиката на приложението и правилата за обработка на данните. Тук се използва C# код за изпълнение на операции като създаване, редактиране и изтриване на задачи.
3. Data Access Layer (Слой за достъп до данни) - Този слой осигурява достъп до базата данни и извършва операции като извличане, добавяне и изтриване на данни. Използва се C# код за комуникация с базата данни.

## Организация и код на заявките към база от данни

## За достъп до базата данни използваме технологията на Microsoft SQL Server в ASP.NET приложението. В Data Access Layer имаме следните методи за извличане, добавяне и изтриване на обекти в базата данни:

## Метод за извличане на всички клиенти от базата данни: ListCustomers()

## Метод за добавяне на нов клиент в базата данни: CreateCustomer()

## Метод за редактиране на съществуващ клиент: EditCustomer()

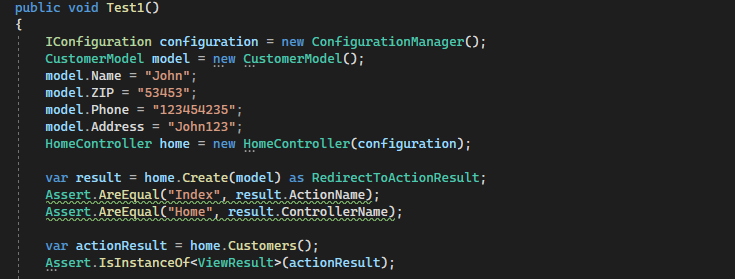
1. Метод за изтриване на клиент от базата данни: DeleteCustomer()

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

Потребителският интерфейс на приложението е уеб базиран и използва HTML, CSS и JavaScript за визуализацията. Основните функционалности на интерфейса включват:

1. Регистрация на потребител
2. Вход в системата
3. Създаване на нов клиент
4. Преглед на списък със съществуващи клиенти
5. Редактиране на клиент
6. Изтриване на клиенти

# Тестване



# Заключение и възможно бъдещо развитие

1. Разширение на функционалността на приложението - Добавяне на нови модули и функции, които да отговарят на конкретните нужди и изисквания на потребителите.
2. Усъвършенстване на потребителския интерфейс - Оптимизиране на визуалния дизайн, подобряване на потребителското изживяване и въвеждане на интерактивни елементи.
3. Разширение на базата данни - Добавяне на нови таблици, релации и полета, за да се поддържа по-широка гама от информация и функционалности.
4. Подобряване на сигурността - Въвеждане на механизми за аутентикация, авторизация и защита на данните, за да се гарантира, че приложението е сигурно и предпазва личната информация на потребителите.
5. Тестване и отстраняване на грешки - Извършване на систематични тестове и корекции на евентуални грешки и пропуски в приложението, за да се гарантира неговата стабилност и надеждност.

# 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 |