Спецификация на изискванията

за

GoodsLog – Store Management System

Версия 1.0

Изготвена от:

Джан Сали – 2101321036

ПУ „Паисий Хилендарски“

20.03.2024 г.

Съдържание

[1. Въведение 1](#_Toc164420130)

[1.1 Цел на документа 1](#_Toc164420131)

[1.2 Конвенции 1](#_Toc164420132)

[1.3 За кого е предназначен този документ 2](#_Toc164420133)

[1.4 Обхват на продукта 2](#_Toc164420134)

[2. Общо Описание 2](#_Toc164420135)

[2.1 Обща Перспектива 2](#_Toc164420136)

[2.2 Характеристики на продукта 3](#_Toc164420137)

[2.3 Видове потребители и техните правомощия 3](#_Toc164420138)

[2.4 Операционна среда 4](#_Toc164420139)

[2.5 Ограничения за дизайн и имплементация 5](#_Toc164420140)

[3. Външни изисквания към интерфейса 5](#_Toc164420141)

[3.1 Потребителски интерфейси 5](#_Toc164420142)

[3.2 Комуникационни интерфейси 6](#_Toc164420143)

[4. Функционалности на системата 7](#_Toc164420144)

[4.1 Управление на персонал (добавяне/премахване на потребители) 7](#_Toc164420145)

[4.1.1 Описание и приоритет: 7](#_Toc164420146)

[4.1.2 Последователност от действия: 7](#_Toc164420147)

[4.1.3 Предварителни изисквания: 7](#_Toc164420148)

[4.2 Вписване на доставчици 8](#_Toc164420149)

[4.2.1 Описание и приоритет: 8](#_Toc164420150)

[4.2.2 Последователност от действия: 8](#_Toc164420151)

[4.2.3 Предварителни изисквания: 8](#_Toc164420152)

[4.3 Осъществяване на транзакция 9](#_Toc164420153)

[4.3.1 Описание и приоритет: 9](#_Toc164420154)

[4.3.2 Последователност от действия: 9](#_Toc164420155)

[4.3.3 Предварителни изисквания: 9](#_Toc164420156)

[5. Други нефункционални изисквания 10](#_Toc164420157)

[5.1 Изисквания за производителност 10](#_Toc164420158)

[5.2 Изисквания за сигурност 10](#_Toc164420159)

[5.3 Атрибути за качеството на софтуера 11](#_Toc164420160)

[6. Други изисквания 11](#_Toc164420161)

# Въведение

## Цел на документа

Документът цели да запознае читателя със спецификациите и изискванията на софтуерния продукт „GoodsLog – Store Management System“ (версия 1.0), както и да опише основните му функционалности и архитектурни характеристики. Документът ще предостави ясна представа за гореизброените атрибути на софтуерните разработчици, екипа за поддръжка, отговорниците за тестване на софтуера (QA екип), както и на екипа, отговорен за дистрибуцията на софтуера („Sales“ отдел), за да бъдат изцяло подготвени, в случай че възникнат въпроси относно същността на софтуерния продукт от страна на клиента.

## Конвенции

Документът е изготвен, следвайки конвенцията за „Спецификация на софтуерните изисквания“ ( на английски – “Software Requirement Specification” – SRS). Следва се консистентен стил на документа – шрифт и размер на текста. Що се отнася до именуването на класове, функции, методи и променливи, ще се следва следния подход:

* Класове – класовете следват да бъдат именувани, като първата буква винаги е главна (каквато е утвърдената конвенция за именуване на класове), а всяка следваща дума или логически отделна единица, която е част от името на класа също с главна буква - т. нар. “camelCase”. Например при създаването на клас, който би следвало да се казва “worker class”, той трябва да бъде именуван като "WorkerClass". Класовете трябва да бъдат именувани еднозначно, по начин, по който възможно най-добре описват функционалността си. Например, ако бъде създаден друг клас "WorkerClass" при наличието на вече съществуващ такъв е забранено да бъде именуван като "WorkerClass2", "WorkerClassВ" или производни. Класовете интерфейси ще се именуват като винаги започват с “I\_” следвано от семантичното име на класа, а абстрактните класове – “A\_” следвано от семантичното име на класа.

**Бележка:** "WorkerClass" е само пример – самото име на този клас противоречи на приетите горе конвенции за именуване на класове поради факта, че не описва и не загатва възможно най-подробно за своята функционалност, нито пък е еднозначен.

* Функции и методи – функциите и методите следва да бъдат именувани по начин, по който възможно най-добре описват или загатват за функционалността си. Конвенцията за именуване, която ще следваме тук ще е “snake\_case” - всяка дума или логически отделна единица, която е част от името следват да бъдат разделени с долна черта – “\_”.
* Променливи – променливите следва да бъдат именувани по начин, по който най-добре описват функционалността си, като трябва да се внимава да не надвишават повече от 20 букви. Задължително е променливите да носят смислени имена, като се забраняват имена като “var”, “name1”, “int2”, еднобуквени имена като “a” и съкращения на цели думи, ако съкращението не е интуитивно – например: “smt”, “work\_hlp” (от “worker\_helper”) и др. Конвенцията за именуване, която ще следваме тук също ще е “snake\_case”.

## За кого е предназначен този документ

„Спецификация на изискванията“ цели да изпълнява ролята на “guide” (указател) за **разработчиците на софтуер**, като ще им осигури напътствия при разработката на настоящия софтуер по еднозначен, разбираем и достъпен начин, за да се избегнат недоразумения между различните групи от разработчици, които ще си сътрудничат по време на този проект, като се опишат изчерпателно основните принципи, които те трябва да следват.

Отговорникът за реализация на проекта, а именно – **мениджърът на проекта** – е един от ключовите роли, обвързани с този проект, който трябва да се води по спецификациите описани и разглеждани в настоящия документ и да назначава таргетираните задачи по времеви периоди на съответните длъжности, свързани с изпълнението им.

Документът също така е подходящ за **тестерите** на разработвания продукт (QA екип), **екипа за поддръжка** на софтуера след неговата реализация, както и за **екипа, отговорен за дистрибуцията** на упоменатия софтуер, тъй като настоящият документ ще бъде формулиран по начин, по който е разбираем и достъпен както за разработчиците, така и за останалите екипи. С това се цели пълна запознатост на хората с разработвания софтуер и утвърждаване на фамилиарността с него във всички възможни аспекти, независимо от заеманата длъжност.

## Обхват на продукта

„GoodsLog – Store Management System“ е разработван най-вече за собствениците на малък и среден бизнес, като не е изключено и собствениците на по-големи бизнеси и предприятия да получат полза от настоящия продукт.

Потребителите имат достъп до функционалности на системата, които включват, но не се изчерпват с: управление на персонал, управление на стоки (наличност, цени), извършване на справки (осъществени продажби по клиенти и завеждания по склад), издаване на документи (стокови разписки, фактури).

# Общо Описание

## Обща Перспектива

Предимството на разработвания софтуер е в това, че изисква сравнително по-малко ресурси в сравнение с другите предлагани на пазара системи и услуги като т. нар. POS (Point Of Sale), тъй като единствената нужна на потребителя техника за работа с „GoodsLog“ ще е компютър с периферни устройства (мишка, клавиатура) или лаптоп. Освен това „GoodsLog“ е бърз, удобен за работа, лесен за усвояване, достъпен за всички потребители, независимо от техническите им умения, и със симплистичен UI дизайн за бързо ориентиране в средата.

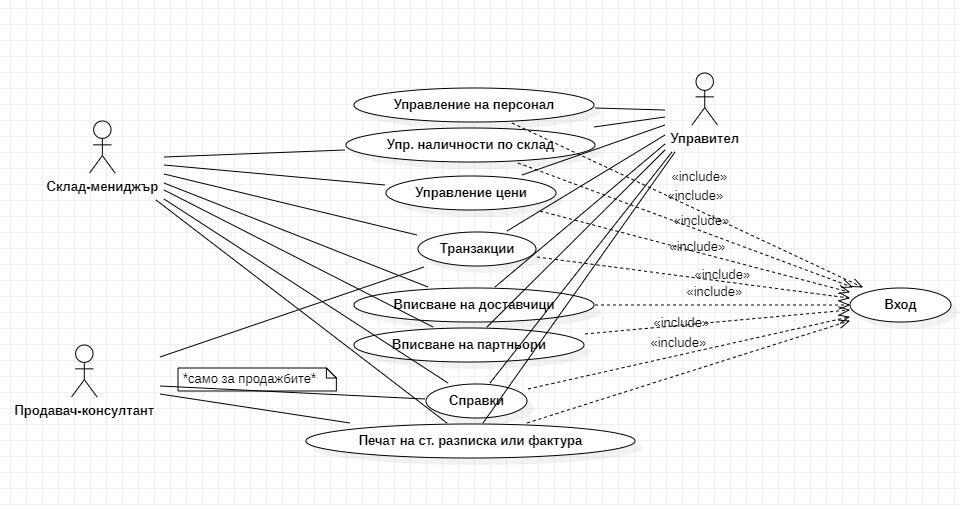
Настоящият софтуер предоставя на собствениците на частен бизнес инструменти за справка на извършените транзакции, следене на тенденциите за повишаване или понижаване на цената на продаваните стоки и бърза и лесна среда на работа на своите служители. Служителите от своя страна получават предимството, освен да осъществяват бързо и удобно обслужване на клиентите, така и да улесняват ежедневната си работа при печатане на стокови разписки и фактури. Високата задоволеност на клиентите е най-висок приоритет на всеки един бизнес, като това ще се осъществи, освен чрез горепосочените предимства на системата, така и чрез прозрачността при покупателна дейност благодарение на издадените от служителите документи.

Настоящият софтуер се разработва за всички Windows операционни системи от Windows 10 до най-новата версия – Windows 11, като в бъдеще той ще бъде надграждан и развиван, за да е винаги up-to-date.

## Характеристики на продукта

Приложението ще позволява:

* Управление на персонал (само за роля „управител“) – добавяне/премахване на потребители на приложението (персонал) или промяна на техните правомощия;
* Управление на наличностите на стоките по склад – вписване/добавяне на стоки при всяка доставка и изписването им при продажба;
* Управление на цени – следене на доставна и продажна цена за всяка стока, всяка доставна цена ще може да бъде променяна при нова доставка, а всяка продажна цена – при процес на изчисляване на обща сума за покупката;
* Създаване на транзакция – вписване на осъществени транзакции, изчисление на обща сума при всяка покупка от страна на клиента и запаметяване в базата данни;
* Вписване на доставчици – добавяне на данни за доставчиците на стоки на фирмата;
* Вписване на партньори – въвеждане на данните на частни и юридически лица в системата (наричани „партньори“ в приложението). Всеки партньор да има специални преференциални цени, валидни само за него;
* Справки – при необходимост потребителят ще може да вижда справки за закупени/продадени стоки по доставчици или партньори;
* Печат на стокова разписка – печатане на стокови разписки за закупените от страна на клиентите стоки;
* Печат на фактури.



## Видове потребители и техните правомощия

Видовете потребители се дефинират от своите правомощия при работата със системата – управител, продавач-консултант, склад-мениджър.

* **Управител** – притежава абсолютни правомощия: да следи постъплението и продажбата на стоки, да управлява тяхната наличност, както и продажна и доставна цена, да извършва всички видове справки, да добавя или изтрива другите потребители в системата или да променя техните правомощия.
* **Склад-мениджър** – да следи постъплението и продажбата на стоки, да управлява тяхната наличност, както и продажна и доставна цена, да извършва всички видове справки.
* **Продавач-консултант** – да извършва продажба на стоки, да издава фактури и стокови разписки, както и да извършва справки само за продажба на стоки.

## Операционна среда

Настоящият софтуерен продукт е проектиран като *standalone application*, и като такъв, той има възможно най-малък на брой зависимости (dependencies) от други услуги и архитектури – идеята е да се постигне възможно най-голяма автономност на системата спрямо другите cloud- и web-базирани конкурентни приложения на пазара. Единственото изискване към клиентския хардуер е той да бъде 64-битово базиран и да има ОС от Windows 10 (или по-нов). За база данни ще се използва Microsoft SQL Server Management Studio, поради надеждността си и ефективната работа в Windows операционна среда.

По време на разработката на настоящия проект не бива да се забравя, че макар GoodsLog да е подходящ за всички търговски обекти, независимо от техния мащаб, главният фокус и основната целева група за този софтуер е малкият и средният бизнес, като в същото време една от най-важните характеристики на продукта е неговата леснота и ниският learning curve, изискван от новите потребители.

В тази си частност, продуктът ще е фокусиран основно върху функционалностите, описани в точка 2.2. от настоящия документ и ще се цели възможно най-голямо „капсулиране“ на средата и с това ще се постигне:

* **Независимост на системата** - ще се сведат до минимум използваните външни библиотеки и услуги, а връзка към сървър след интегрирането на приложението в устройството на клиента ще се прави само периодично и то единствено за докладване на възникнали по време на експлоатация на продукта грешки. По тази методика, клиентът няма да има непрекъсната нужда от свързаност към интернет, а дори и въобще да не разполага с такъв, това няма да представлява проблем.
* **По-достъпна цена за крайния потребител** – настоящият софтуер ще работи единствено с Microsoft SQL Server като външна услуга, чиято безплатна версия ще е достатъчна в по-голяма част от случаите, поради естеството на таргетираната клиентела (описана по-горе). GoodsLog няма да ползва услуги, обвързани с месечен (или друг) абонамент като например AWS, Azure, Firebase или други релационни бази от данни, което означава, че той ще бъде финансово по-достъпно в сравнение с голяма част от конкурентните приложения.
* **Гаранция, че приложението ще работи утре по същия начин, както е работил днес –** GoodsLog не разчита на това чужда собственост (сървър, услуга) да работи правилно, за да работи правилно и той – архитектурата е *монолитна* – всичко се намира на клиентската машина и единствено негова е отговорността да е добър стопанин. А при възникване на проблем от софтуерно естество, клиентът може да свърже с нас чрез нашия уебсайт, където са изброени всички видове за установяване на контакт с отдела по поддръжка и да поиска съдействие с отстраняване на проблема.

Като обобщаващо изречение за тази точка ще кажем, че GoodsLog комбинира всеизвестното user-friendliness на Windows, финансова атрактивност и сигурността да пазиш данните за себе си и за бизнеса си на собствената машина, без да поверяваш отговорността за тях на някой друг – било то компания или хостинг.

*Операционната среда на приложението е избрана на базата на най-голяма достъпност според потребителя, а архитектурите, технологиите и инструментите – спрямо операционната среда.*

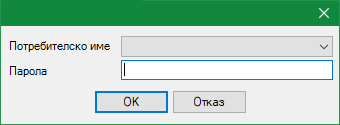
## Ограничения за дизайн и имплементация

Монолитната архитектура на GoodsLog е съсредоточена в единствено приложение, което интегрира всички функционалности в едно цяло. Този подход позволява на приложението да предостави бързо, надеждно и удобно за работа преживяване на потребителите. Също така, спрямо характеристиките за софтуер, монолитните приложения са много по-лесни за разработване, дебъгване, тестване и деплойване.

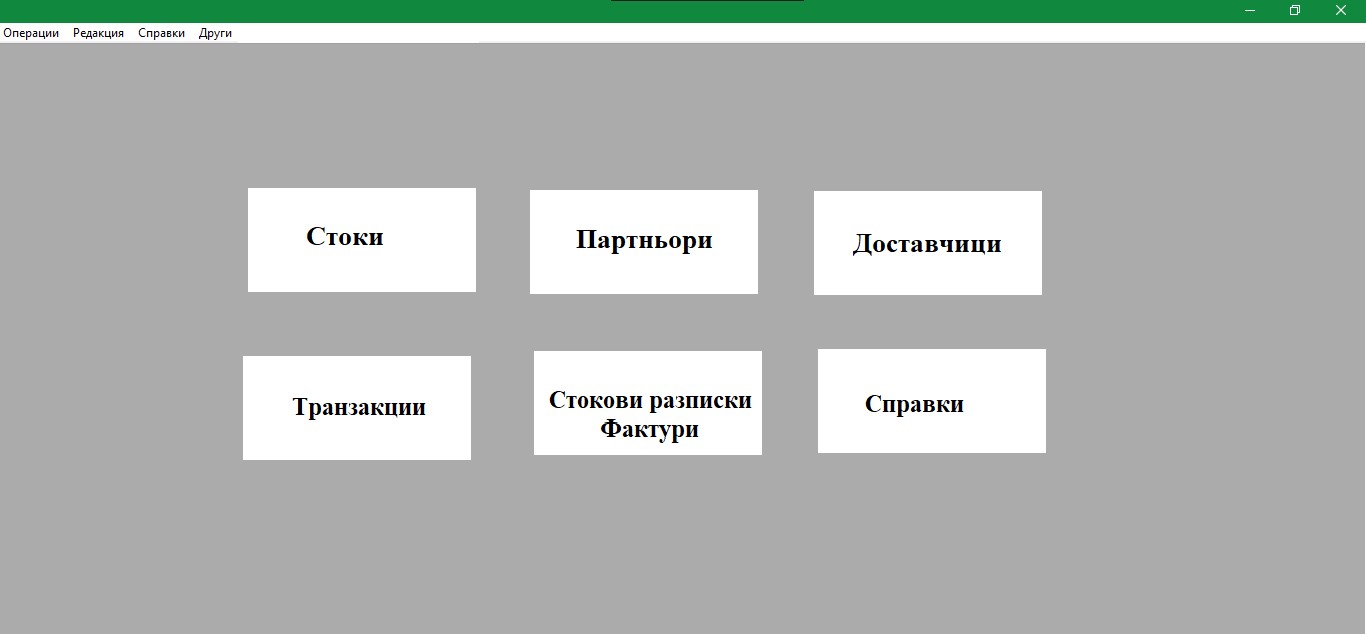
* **GUI** – за потребителския интерфейс ще се използва разработения и предоставения от Microsoft фреймуорк Windows Forms. Този фреймуърк предоставя мощни инструменти и функционалности за създаване на интерактивни и интуитивни потребителски интерфейси. Благодарение на интеграцията с широката екосистема на Microsoft, разработчиците на приложението имат достъп до широк набор от ресурси и инструменти за по-бързото разработване и по-лесната поддръжка на приложението.
* **Бази от данни** – тъй като избраната операционна среда за приложението е Windows и изборът на технологии е тясно обвързан с нея, е само логично да изберем релационна база от данни, създадена за работа в екосистемата на Windows – MS SQL Server.

# Външни изисквания към интерфейса

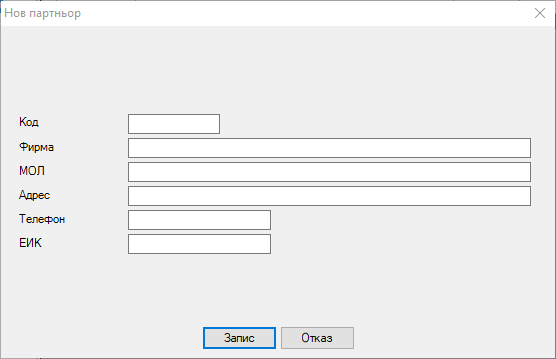
## Потребителски интерфейси

Входен екран:

Начален екран след вход:



Добавяне на нов партньор:



Редактиране на съществуващи партньори:



## Комуникационни интерфейси

Поради архитектурните особености на приложението, наблюдаваме два вида комуникация:

* Между ГПИ и бекенд – взаимодействието на потребителя с приложението трябва да протича максимално бързо, като не бива да има лагване, замръзване на екрана, а в случай на грешка – да се представи добре описано и разбираемо описание на проблема и евентуалното му решение. ГПИ трябва да откликва моментално на всяко действие на потребителя, незабавно да прати съответните данни към бекенда, той от стоя страна да ги обработи и да върне резултата визуализиран за потребителя.
* Между бекенд и база от данни – трябва да се подсигури консистентност на данните от страна на бекенда преди да бъдат изпратени за съхранени от базата от данни. Също така в базата от данни ще бъдат имплементирани механизми за предотвратяване на deadlocks в случай на забавяне на обработването на данните и постъпването на нови заявки.

# Функционалности на системата

## Управление на персонал (добавяне/премахване на потребители)

### Описание и приоритет:

Потребителят с върховни права в приложението, управителят, може да добавя нови потребители или да изтрива вече съществуващи такива, както и да променя правомощията им – да надгради правата на потребител на продавач-консултант към по-горен клас потребители – склад-мениджъри или съответно да ги понижи (по аналогичен начин).

Приоритет: Висок

### Последователност от действия:

### Предварителни изисквания:

#### Потребител с правомощия „управител“ да е влязъл в системата.

* Потребител с правомощия „управител“ да има персонал за назначаване/уволняване или за промяна на правомощия.

## Вписване на доставчици

### Описание и приоритет:

Потребителите с роли „Управител“ и „Склад-мениджър“ имат правомощията да създават/изтриват/променят фирми-доставчици, които снабдяват търговския обект със стоки. Те се назовават „доставчици“ за по-кратко в разработвания софтуер. Данните, които са нужни при създаване на нов доставчик са: ЕИК/БУЛСТАТ, търговско име, адрес по регистрация на фирма, МОЛ, телефон, имейл (последните две са опционални).

Приоритет: Висок

### Последователност от действия:

### Предварителни изисквания:

* Вход в системата като „Управител“ или „Склад-мениджър“
* Избиране на подменю „Доставчици“
* Набавяне на нужните данни за създаване на нов запис за доставчик

## Осъществяване на транзакция

### Описание и приоритет:

След избиране на елементи от списъка със стоки се пристъпва към финализиране на покупката с натискане на бутон „Плащане“. Следва инициализиране на прозорец за потвърждение, че покупката ще бъде финализирана. С натискане на бутон „ОК“ се инициализира нов прозорец с опция да се въведе паричната сума, с която възнамерява да плати посетителят на търговски обект, извършващ покупката. Този прозорец може да спомогне на продавач-консултанта да изчисли разликата между паричната стойност, дадена от лицето, извършващо покупката и общата стойност на покупката с цел по-бързо осъществяване на транзакцията и елиминиране на възможността за връщане на грешно ресто по човешка вина. Прозорецът може да не бъде използван, като за това може просто да се натисне бутон „Пропусни“. В противен случай се въвежда паричната сума, предоставена от лицето, извършващо покупката и се натиска бутон „ОК“ и системата автоматично извежда разликата между двете стойности. Независимо от избора на потребителя на приложението, то се връща в екрана със списъка със стоки, за да е готов за нова транзакция.

Приоритет: Висок

### Последователност от действия:

### Предварителни изисквания:

* Вход в системата
* Избиране на опция за започване на нова транзакция
* Съществуващата база данни със стоки

# Други нефункционални изисквания

## Изисквания за производителност

Действия като създаване на транзакция, приключване на транзакция, избор на партньор/доставчик/стока, CRUD операции върху партньор/доставчик/стока, зареждане на таблица със стоки, справка (показване на детайли) за осъществена вече транзакция и подобни действия, изискващи малки изчислителни ресурси, трябва да дават ответна реакция на потребителя мигновено – за по-малко от 100 милисекунди. Забавяне при тези операции по вина на разработвания софтуер, а не по вина на други приложения в операционната система (или по нейна собствена вина), е недопустимо.

Една от най-важните и приоритетни черти на GoodsLog, с която се цели той да бъде отличаващ се от конкурентните софтуерни продукти е именно производителността. Всички решения, взети при проектирането на този софтуер са именно в тази насока.

Сложни справки от вида на:

* месечна справка
* справка по **n** на брой месеци, като **n** < 5

трябва да се осъществяват за по-малко от 300 милисекунди, а сложни справки от вида на:

* справка по **n** на брой месеци, като **n** >= 5
* годишни справки

трябва да се осъществяват за по-малко от 500 милисекунди.

## Изисквания за сигурност

Потребителските роли в приложението трябва да бъдат ясно дефинирани и да се предотврати попадането на каквито и да е несъответстващи и нерегламентирани на логнатия потребител права в неговите ръце.

Приложението съдържа предефинирана в системата си роля на „superuser“, от която ще бъдат производни останалите гореупоменати в документа роли. При деплойване на приложението в реална среда (при клиента), тази роля ще бъде използвана, за да се дефинира частен случай на ролята „Управител“, която по-късно ще се използва съгласно нуждите на клиента. С акцент към факта, че приложението не се нуждае от интернет връзка, като с това се цели възможно най-високо ниво на неговата самостоятелност и независимост от външни нему фактори, се обръща внимание, че използването на имейл адрес и парола би противоречило на тези поставени цели. Не се очаква клиентът, закупил настоящия софтуер, да го интегрира сам в своята физическа среда, а ще получи необходимото сътрудничество от страна на екипа за поддръжка. Осланяйки се на този факт, се гарантира правилното интегриране на приложението в клиентската среда и утвърждаването на очакваната и детерминирана работа на функционалността, касаеща ролите в приложението.

За да се гарантира сигурността на данните, приложението ще има вградена система за редовно резервно копиране на данни и процедури за възстановяване, които ще осигурят непрекъснатост на бизнеса и защита на информацията в случай на инциденти или загуба на данни.

## Атрибути за качеството на софтуера

* Лесен за научаване и усвояване – приложението трябва да предоставя интуитивен и лесен за използване потребителски интерфейс, който позволява на потребителите бързо и лесно да усвоят функционалността и да започнат да я използват след минимално време на обучение.
* Отговаря на очакванията за малък, но компактен складов и POS софтуер - осигуряват се всички необходими функционалности за ефективно управление на складовите операции и обслужване на клиентите, без да се добавят излишни функционалности
* Лесна поддръжка – поради специфичността на вече споменатата избрана за приложението архитектура, се очакват минимална интервенция на екипа за поддръжка в дългосрочен план при работата с клиенти, а кодовата база е лесна за следене и бъдещо подобряване
* Бързодействие на софтуера и интерактивност – приложението трябва да осигури бързо реагиране на потребителските заявки и интерактивност на потребителския интерфейс, за да осигури ефективно и безпроблемно използване от страна на потребителите

# Други изисквания

Приложение A: Терминологичен речник

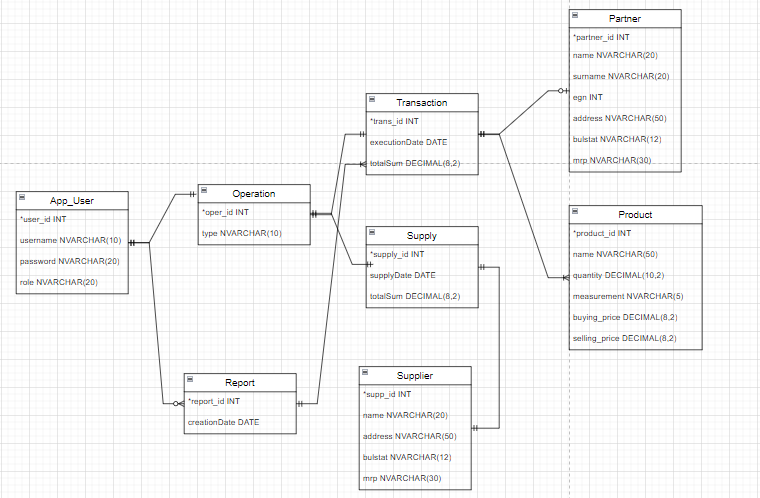
Standalone application – софтуер, проектиран по начин, по който потребителят няма нужда от интернет, връзка към сървър или други външни източници на данни

AWS – Amazon Web Services

User-friendliness – става дума за софтуер, който е лесен и интуитивен за използване продукт от страна на потребителите, които са запознати с Windows операционната система и характерните за нея елементи и действия като например менюта, бутони, панели за инструменти и диалогови прозорци, както и използването на стандартни жестове за управление като кликване, дърпане и пускане и др.

ГПИ – Графичен потребителски интерфейс ( еквивалентен на английското GUI)

CRUD – операции за манипулиране на данни като добавяне, изтриване, промяна и четене на записи

Приложение B: Аналитични Модели

