Техническа Документация

Sales Management

Екип:

Деян Митков Куртев

Десислав Андреев Христов

Боян Гошев Кръстев

Вивиан Борисова Данчева

Дата: 2016.05.31

# История на техническата документация

**Автори**

| Фак. номер | Име | Контакт (ел. поща) |
| --- | --- | --- |
| 121213013  121213166  121213195  121213004 | Деян Митков Куртев  Десислав Андреев Христов  Боян Гошев Кръстев Вивиан Борисова Данчева | deiankurtev@gmail.com  desso430@gmail.com  boqngoshev@abv.bg  dancheva14@abv.bg |

**История на версиите**

| Версия | Дата | Автор | Описание |
| --- | --- | --- | --- |
| 0.00.1 | 15.05.16 | Десислав  Боян | Определяне на потребностите и изискванията, намиране на решения и изготвяне на задание за проектиране . |
| 0.00.2 | 18.05.16 | Десислав  Вивиан | Определяне на нужните данни и създаване на нужната база данни за софтуерния продукт. |
| 0.00.3 | 20.05.16 | Боян  Вивиан | Изготвяне на потребителски интерфейс, който да отговаря на функционалните изисквания. |
| 0.00.4 | 23.05.16 |  | Имплементиране на функционалностите. |
| 0.00.5 | 25.05.16 |  | Тестване на софтуера. |

# 

# Съдържание

[1 Въведение 4](#_Toc446758084)

[1.1 Цели на системата 4](#_Toc446758085)

[1.2 Обхват на системата 4](#_Toc446758086)

[2 Спецификация на изискванията 4](#_Toc446758087)

[2.1 Системни изисквания 4](#_Toc446758088)

[2.1.1 Функционални изисквания 4](#_Toc446758089)

[2.1.2 Нефункционални изисквания 4](#_Toc446758090)

[2.2 Изследване на потребителските изисквания 4](#_Toc446758091)

[2.2.1 Потребителски истории 4](#_Toc446758092)

[2.2.2 Критерии за приемане на потребителски истории 5](#_Toc446758093)

[3 Концептуален дизайн на системата 5](#_Toc446758094)

[3.1 Софтуерна архитектура на система 5](#_Toc446758095)

[3.2 Декомпозиция на системата на модули 5](#_Toc446758096)

[3.3 Концептуален модел на БД 5](#_Toc446758097)

[4 Ресурси 5](#_Toc446758098)

[4.1 Акроними 5](#_Toc446758099)

[4.2 Други 5](#_Toc446758100)

# Въведение

## Цели на системата

Системата трябва да дава права на служителите да създават договори, да имат достъп до личната информация на клиентите. За всеки клиент трябва да се пази лична информация. Служителите трябва да имат достъп до регистъра на услуги, методите на ценообразуване и условия на плащане. Системата трябва да синтезира статистики за клиентите. Системата трябва да експортва информацията в PDF файлове. Трябва да може да създава електронни фактури. Системата трябва да има възможност за блокиране на клиенти. Всеки служител трябва да може да прави справки, да добавя и трие информация. Целта на системата е улеснено изготвяне на договори между служители и клиенти.

## Обхват на системата

Системата трябва да обслужва нуждите на клиентите, като пази информацията за всеки клиент в картон, да дава възможност за различни услуги, гъвкаво ценообразуване, плащания. Системата също така трябва да позволява изготвянето на договори по продажби, електронни фактури, извеждане на списък с предлаганите услуги, да дава възможност за плащане и да позволява комуникация с външни системи.

# Спецификация на изискванията

## Системни изисквания

### Функционални изисквания

| **Номер** | **Изисквания** | **Приоритет** |
| --- | --- | --- |
| 1 | Създаване на договор. | Висок |
| 2 | Модифициране на лични данни на клиентите в системата. | Среден |
| 3 | Възможност за добавяне на услуги/продукти към съответстващи катеории. | Висок |
| 4 | Връщане на списък с клиенти които са в даден диапазон приходи. | Нисък |
| 5 | Да позволява преглеждане и отчитане на бъдещи вземания от клиенти. | Среден |
| 6 | Промяна на публичния ценоразпис на стоките в системата. | Среден |
| 7 | Прикачане на документи(удостоверения, заповеди, лицензи, разрешителни, други.) към договор. | Нисък |
| 8 | Комуникация с външни системи за факуриране. | Висок |
| 9 | Добавяне на рестрикции към клиента(забрана за пазаруване, ограничение в максимална цена на поръчка) | Висок |
| 10 | Системата трябва да има бази данни с опция за достъп и модифициране | Висок |
| 11 | Системата трябва да има страница с личните данни на клиенти с функционалност за добавяне на информация | Среден |
| 12 | Системата трябва да реализира роли за потребителите си, като за всяка от тях предоставя достъп на различни нива. | Среден |
| 13 | Справки от тип drill-down | Нисък |

### Нефункционални изисквания

| **Номер** | **Изисквания** | **Приоритет** |
| --- | --- | --- |
| 1 | Системата трябва да поддържа TCP/IP. | Висок |
| 2 | Възможност за ъпгрейд на системата. | Среден |
| 3 | Системата трябва да бъде реализирана с обектно-ориентиран език за програмиране (C#) | Висок |
| 4 | Да имплементира архитектурата MVVM. | Среден |

## Изследване на потребителските изисквания

### Потребителски истории

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Номер* на потр. историята** | **Като** | **аз искам да** | **така, че** | **точки на историята** |
| 1 | Служител | създавам договори с клиенти, отнасящи се за търговски сделки | да пазя история на сключените договори | 10 |
| 2 | Служител | имам права върху данните на клиентите | да мога да създавам, променям и изтривам клиенти. | 7 |
| 3 | Служител | имам достъп до регистъра на услуги, методите на ценообразуване и условия на плащане | да мога да ги променям | 6 |
| 4 | Служител | разполагам с подходящи категории за услуги/продукти | да мога да намирам  търсения от мен услуга/продукт | 4 |
| 5 | Служител | групирам клиентите според техните приходи | да мога да извеждам статистики. | 5 |
| 6 | Служител | получавам актуална информация за дължими суми от клиенти | да правя счетоводни сметки. | 5 |
| 7 | Служител | експортвам договорите в Word и PDF формати | да мога да ги разпечатвам. | 8 |
| 8 | Служител | достъпвам външни системи чрез текущата система | да мога да работя с електронни фактури. | 10 |
| 9 | Служител | мога да блокирам клиент за фактуриране | да мога да го разблокирам при настъпването на определени условия на плащане. | 4 |
| 10 | Служител | мога да прикачвам външни за системата документи към клиент, фактура, плащане | използвам прикачените документи при необходимост. | 3 |

### Критерии за приемане на потребителски истории

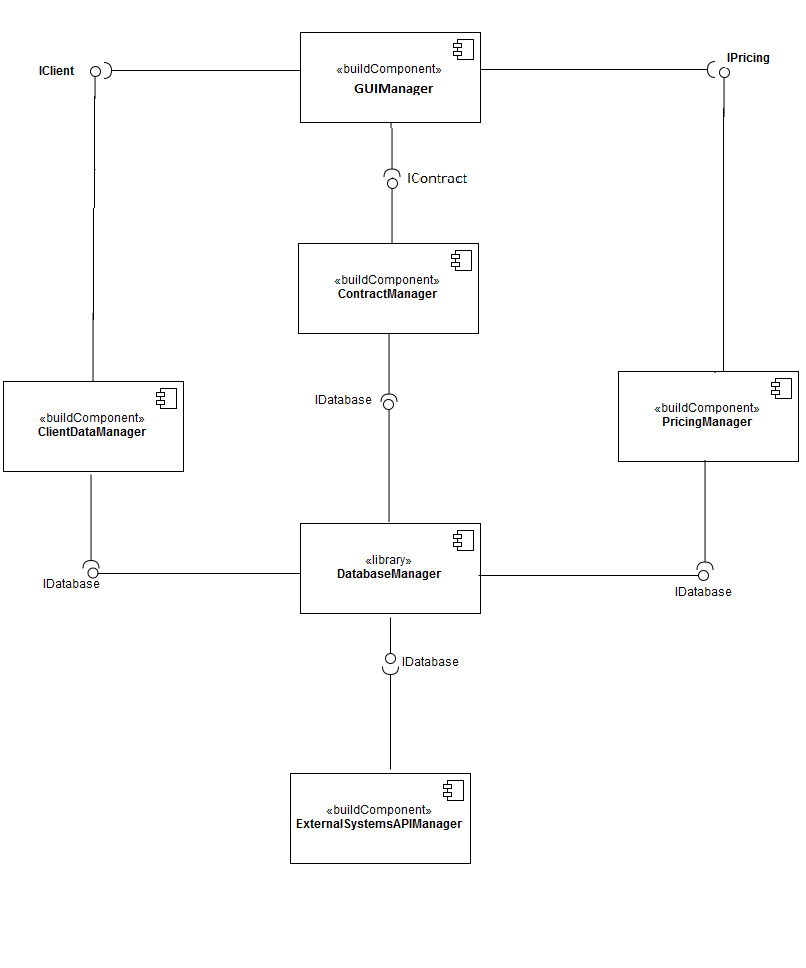
|  |  |
| --- | --- |
| **Номер на потр. историята** | Критерии за приемане |
| 1 | Попълнената форма с информация за сключения договор(участващи клиенти, закупени услиги/продукти) се записва в базата. |
| 2 | Успешно добавяне на клиенти от базата данни.  Успешно редакция на клиенти от базата данни.  Успешно изтриване на клиенти от базата данни. |
| 3 | След въвеждане в системата, услугата е налична за добавяне към договорите.  След редакция на услуга, при търсене са налични обновените и данни.  След изтриване на услуга, тя не се намира в регистъра с услуги.  След промяна в метод за ценообразуване, промените са отразени в базата на системата.  След промяна в условие на плащане, промените са отразени в базата на системата. |
| 4 | Всеки продукт или услуга принадлежи към категория, съответстваща на характера му.  След търсене в дадена категория там се намира очаквания продукт/услуга. |
| 5 | Всеки клиент принадлежи към категория, съответстваща на групата му приходи.  При избор на категория приходи може да се направи статистика за броя на клиентите в нея.  При избое на клиент, може да се изведе в статистика към кои групи приходи е принадлежал. |
| 6 | Поддържане на информация за дълговете на всеки клиент.  Своевременно обновяване на информацията след плащане. |
| 7 | Успешен и коректен експорт на договор в Word и PDF формат. |
| 8 | Успешно и коректно отпечатване на електронна фактура чрез външна система. |
| 9 | Статуса на клиент в базата преминава от „блокиран“ на „разблокиран“ и обратното, в зависимост от предварително дефинирани чрез системата условия за плащане. |
| 10 | Успешно прикачване на външни за системата документи към:   * информацията за даден клиент. * фактура. * плащане. |

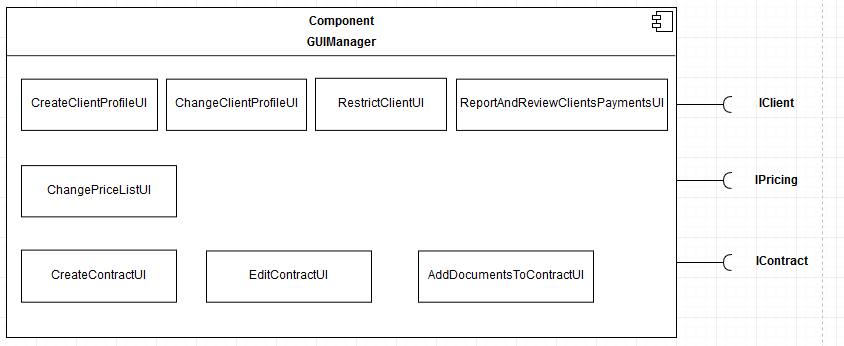
# Концептуален дизайн на системата

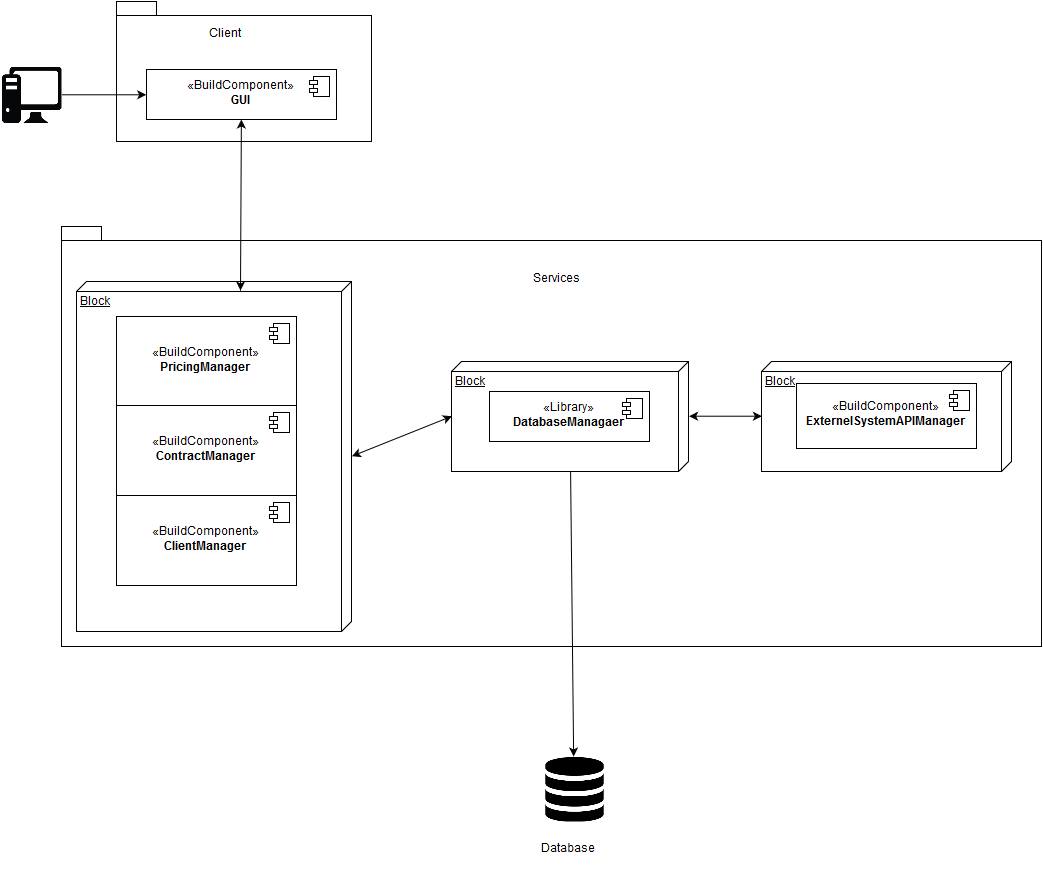
## Софтуерна архитектура на система



## Декомпозиция на системата на модули







## Концептуален модел на БД



# Ресурси

## Акроними

| **Акроним** | **Описание** |
| --- | --- |
| експорт | Изнасяне на докомент от системата в определен формат. |
| MVVM | Софтуерна архитектура за разделяне на потребителски интерфейс от бизнес логиката. |
| Обектно ориентиран език | парадигма в компютърното програмиране, при която една програмна система се моделира като набор от обекти, които взаимодействат помежду си. |

## Други

http://www.tutorialspoint.com/sql/sql-create-database.htm

https://msdn.microsoft.com/en-us/library/hh848246.aspx

http://www.wikihow.com/Develop-Software

http://www.yacoset.com/Home/how-to-design-a-computer-program

http://www.codingthearchitecture.com/2012/05/10/how\_do\_you\_design\_software.html

http://www.wpf-tutorial.com/wpf-application/introduction/

http://www.goldsim.com/Web/Solutions/BusinessSystems/