**ПРЕДМЕТ: РАЗРАБОТКА НА СОФТУЕР**

**ДЕЙНОСТ: УЧЕНИЧЕСКИ ПРАКТИКИ 2**

***2021-2022***

**Тема: Банкова система с онлайн банкиране и ПОС терминал**

**Курсов проект**

*Автори:*

*Мартин Стефанов Севов 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

Настоящият документ представлява документация на курсов проект за предмета „Разработка на софтуер“.

* Светът работи на дебит и кредит, затова банките са от голямо значение
* Много банки нямат съвременна система и се целя да покажа примерна система

# Цели и обхват на софтуерното приложение

Целта на проекта е да се реализира банкова система, със онлайн банкиране, софтуер за работници в банка, уеб страница за онлайн банкиране, централен сървър за обработка на заявки. Допълнителна цел на проекта е да се осигури достъп и на лица със леки до средни увреждания.

Системата ще позволи на потребители да си създадат сметка в клон на банка (или онлайн; бъдеща идея;). Със тази сметка, потребителите ще могат да правят онлайн разплащания, да плащат на ПОС терминал[[1]](#footnote-1), с карта.

Банкерите ще имат достъп до софтуер, който ги свързва с банковият сървър и им позволява много на брой действия с потребители, с цел улесняване на работата чрез намаляване на бюрокрацията.

Целевата демографска група са юридически и физически лица на възраст по-голяма от 14 години.

# Анализ на решението

## Потребителски изисквания и работен процес

Потребителите на системата са:

* Банкерите, които използват приложение за работа със системата;
* Банковите системни администратори (накратко „*администратори*“), които наблюдават банковият сървър за проблеми и профилактика;
* Стандартни потребители (накратко „*потребители*“), които имат издадена сметка(акаунт) в системата;
* Касиер и собственик на пос устройство.

Всеки потребител със сметка може да:

* Прави онлайн транзакции;
* Следи своето салдо[[2]](#footnote-2);
* Да издава и използва карта;
* Да закрие сметката.

Уеб страницата, банкерското приложение и ПОС терминалът ще изпращат заявки до банковият сървър, който ще ги запазва в списък, и паралелно ще ги изпълнява (едновременно заявки за получаване на информация, и заявки за промяна на информация).

## Примерен потребителски интерфейс

### Уеб страница

Graphical user interface, text

Description automatically generatedGraphical user interface, text, email

Description automatically generatedA picture containing graphical user interface

Description automatically generated

Страница на сметката

Страница за онлайн банкиране

Титулярна страница

### Банкерско приложение

Chart, treemap chart

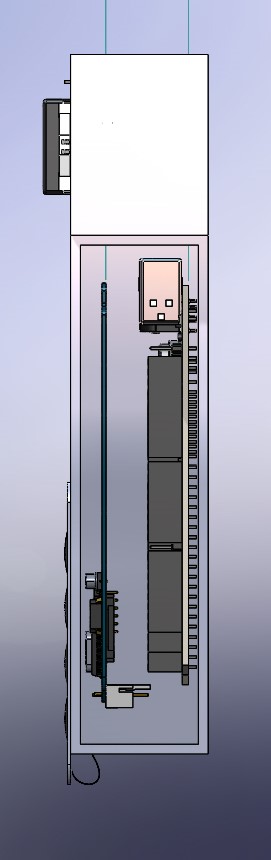
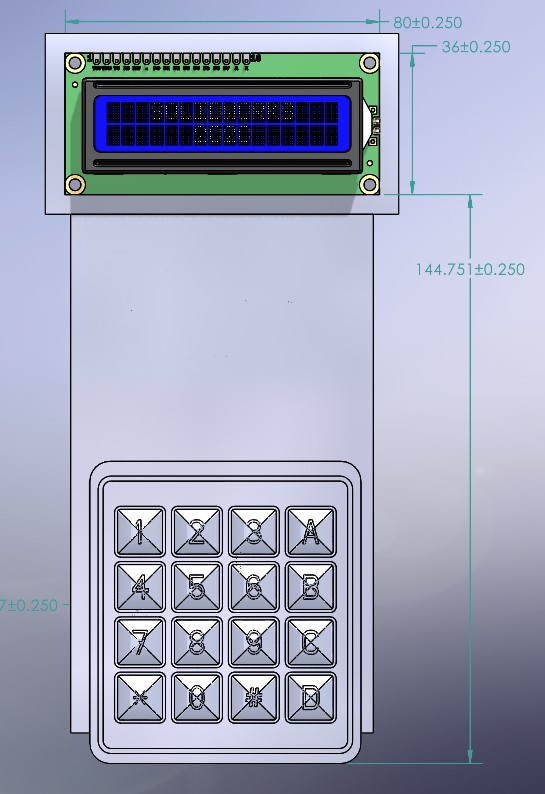
Description automatically generated

### Сървърно приложение

Graphical user interface, text, application

Description automatically generated

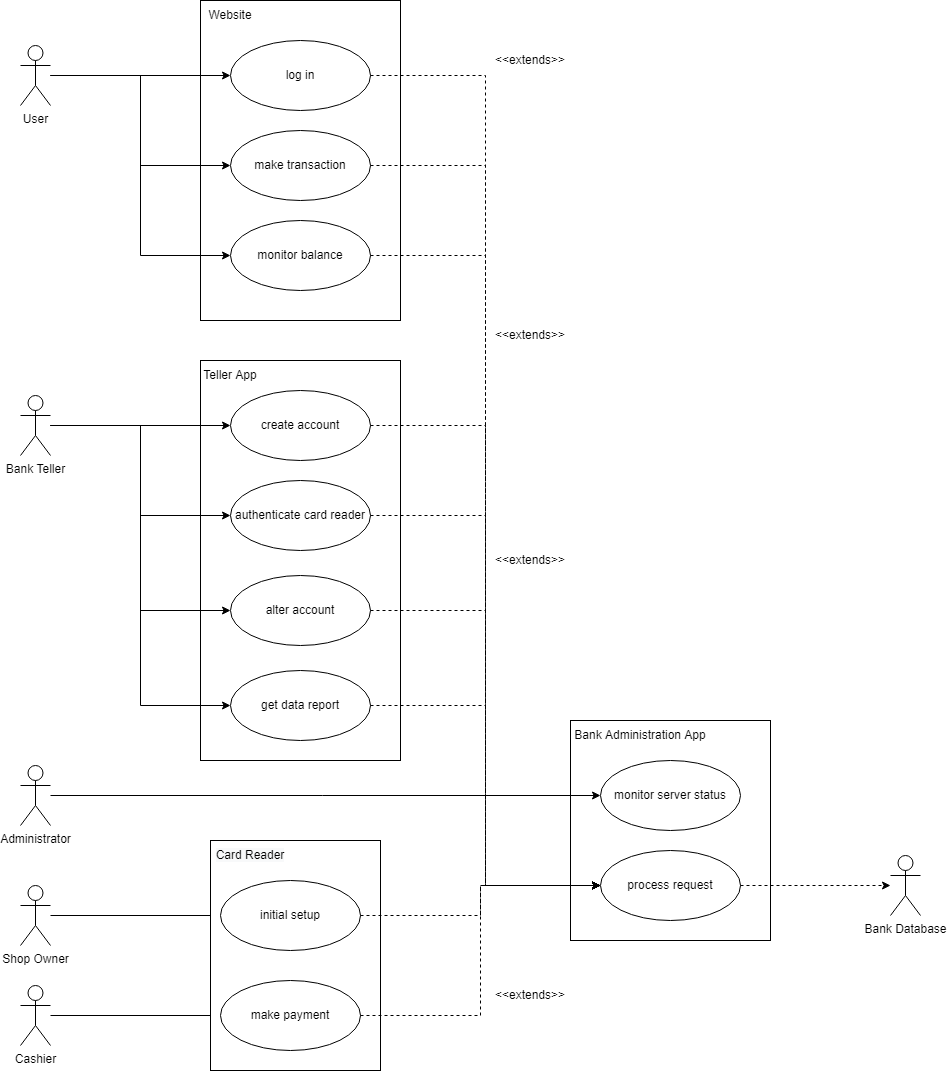
### ПОС терминал



Изработен в Solidworks и принтиран на 3d принтер. Захранен от Arduino.1

## Диаграми на анализа

### Диаграма на случаите (Use case diagram)



*Потребител*(**User**) има директен достъп до Уеб страницата[Да влезе в акаунта му, Да направи транзакция, Да проследи сметката му]

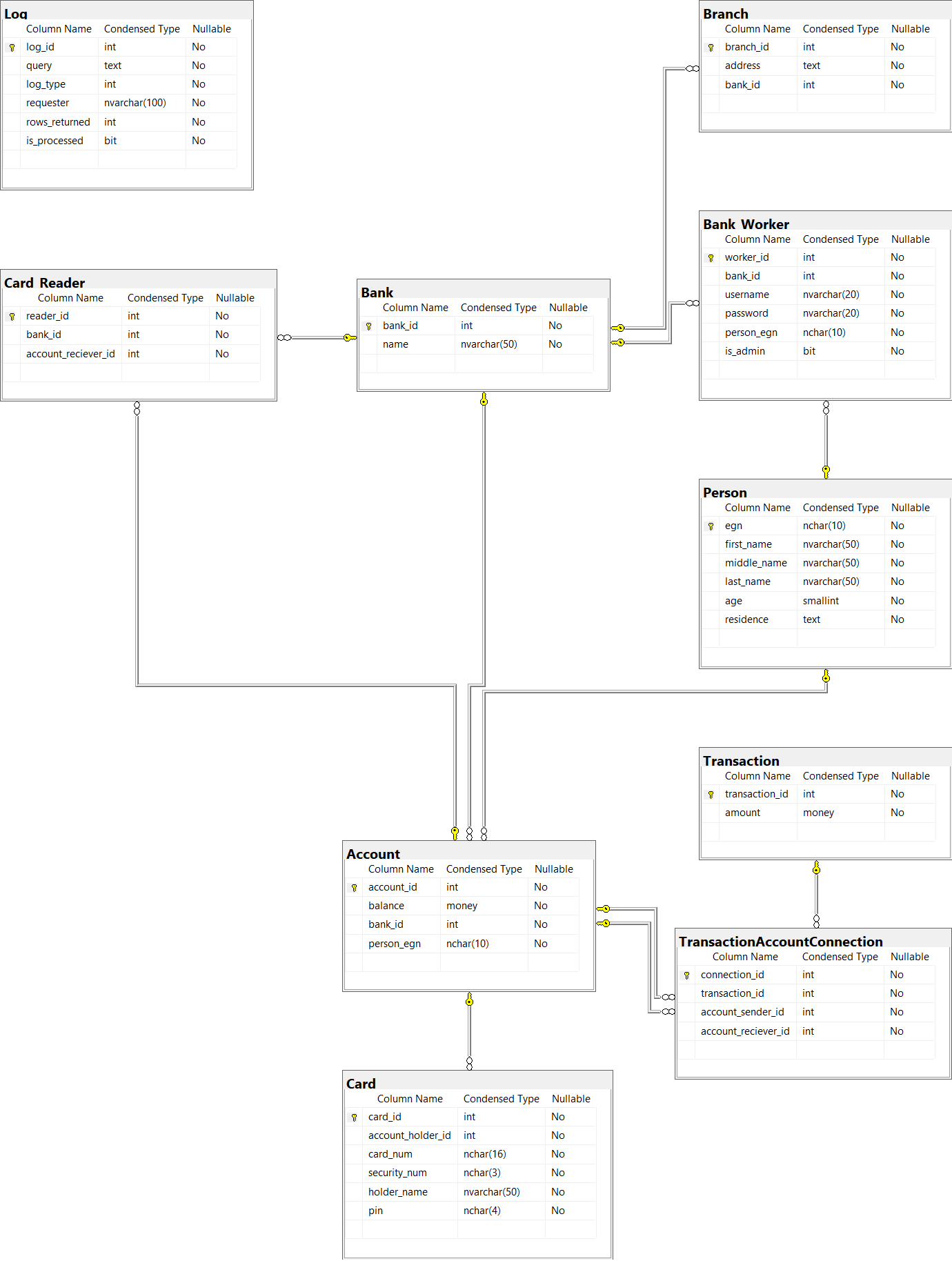
*Банков работник*(**Bank Teller**) има директен достъп до Банкерското приложение[Да създава потребителски акаунт и сметка, Да потвърди четец на карти, Да промени потребителски данни, Да получи доклад за информация]

*Администратор*(**Administrator**) има директен достъп до Сървърното приложение[Да следи статуса на сървъра]

*Собственик на магазин*(**Shop Owner**) има директен достъп до Четец на карти[Първоначална подготовка]

*Касиер*(**Cashier**) има директен достъп до Четец на карти[Извършване на транзакция]

### Модел на съдържанието / данните (Entity relation diagram)



Стара диаграма!!! Branch няма връзка към Admin и Bank\_Teller/Bank\_Worker, Admin е част от Bank\_Teller/Bank\_Worker.

Предстои добавянето на таблица за логове, която трябва да улесни свързването и работата с множество от устройства (Централната машина проверява всички логове от последните X минути на всеки X минути).

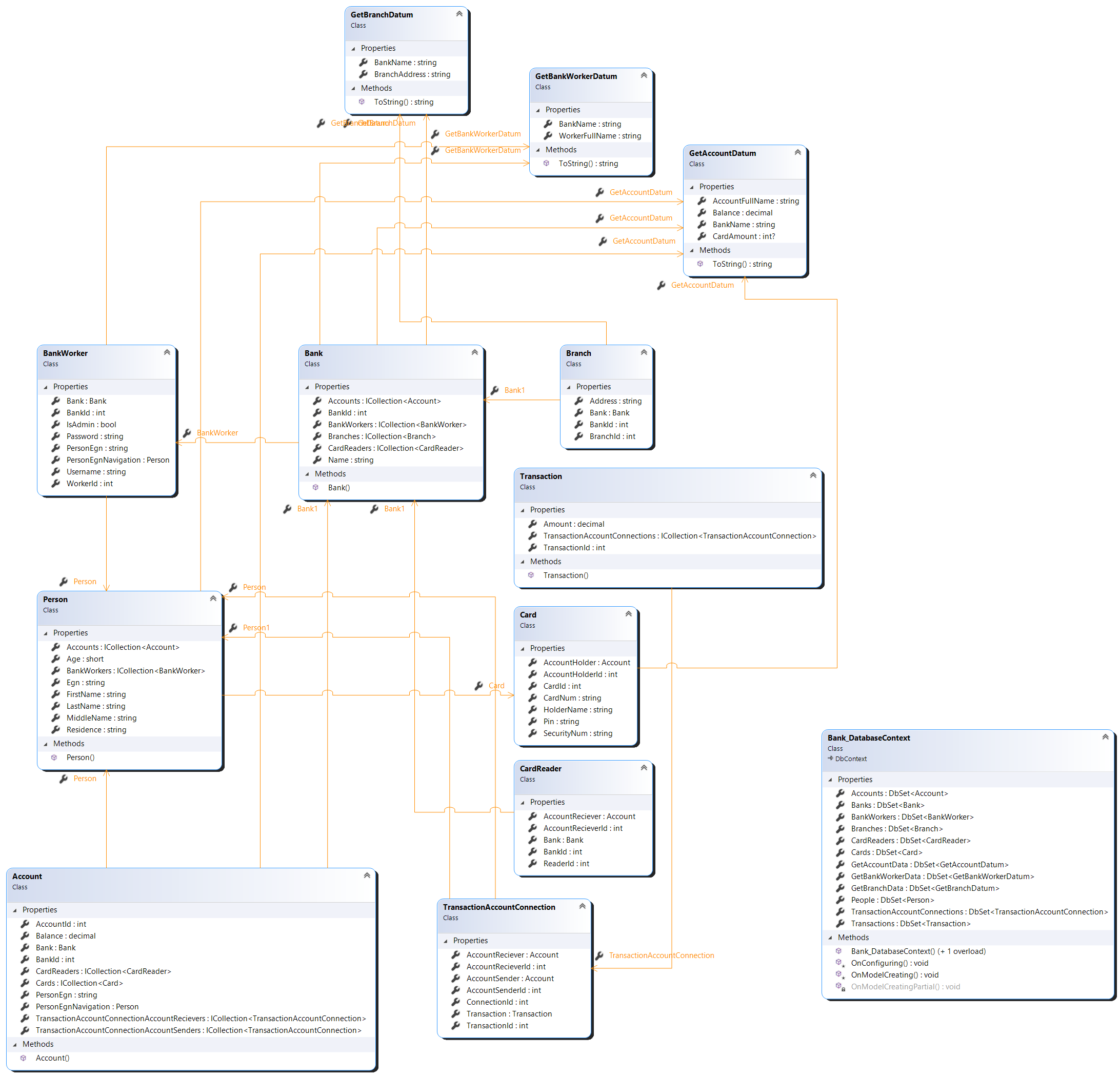
“Bank” е основната таблица, към която всички други са свързани. Една банка може да има много клонове “Branch”, които са физически офиси на тази банка.

В “Person” е основната информация за всеки човек в системата. Един човек може да има профил/сметка “Account” във всяка банка. Един човек може също да бъде и работник в банка “Bank Worker”/”Bank Teller”.

“Account” е връзката между потребителя и системата. Една сметка има връзка към карти за плащане “Card”, терминал за разплащане “Card Reader” и транзакции “Transaction”.

### Диаграма на последователността и на комуникацията (Sequence diagram)

### Диаграма на класовете (Class diagram)



Класовете които ще използвам са разделени на три части:

* Клас със данни (Table, View, Stored Procedure)
* Клас за свързване с базата данни (Database Context, ?)
* Клас за обработка на данни

# Дизайн

Тази секция представя дизайна на решението на проблема за проекта ви. Опишете каква софтуерна платформа сте избрали за вашето решение /напр. .NET, java/. Представете схема на софтуерната архитектура на решението /по модули и/или слоеве/ с диаграма на разгръщането, както и диаграми на класовете на дизайна /с ограничения, описани на OCL/, диаграми на времето /за задаване на времена за синхронизация и комуникация в решението/ и компонентни диаграми. Илюстрирайте решението с извадки от генериран сорс код.

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

Описание на слоевете, предназначението им, библиотеки и методи включени в съответния слой.

## Организация и код на заявките към база от данни

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)

Описание на основните функционалности на интерфейса на приложението.

*Забележка: Няма формално изискване на определен брой диаграми от даден вид, за даден брой проектанти.*

# Тестване

Тук се *включват тестовите случаи* и какви видове тестване предвиждате в реалното изпълнение на проекта, напр. с колко и какви документи, в какви браузъри, с какви приставки, и т.н.

# Заключение и възможно бъдещо развитие

В заключение, обобщете резултатите от работата ви по проекта, както и предимствата и ограничеността на използваните технологии / езици / методи. Укажете какви алтернативи могат да се използват и техните предимства и недостатъци. Опишете каква е използваемостта на подобни решения в практиката и какво бихте предложили като насоки за бъдещо развитие на вашето решение.

# Използвани литературни източници и Уеб сайтове

Използвайте вградената функционалност на Word: References > Citations & Bibliography

1. Уеб сайт на ….., адрес ….
2. Уеб сайт на ….., адрес ….
3. Уеб сайт на ….., адрес ….
4. Уеб сайт на ….., адрес ….
5. Литературен източник 2
6. Литературен източник 3
7. Литературен източник 4
8. Литературен източник 5

# Приложения

При необходимост можете да добавите и допълнителни секции под формата на апендикси. Таблица с диаграми, таблици и графики

*Забележка:*

1. *Документацията на проекта се предава само в електронен вид в MS Word, чрез качването на архив с документа и останалите файлове по проекта, в задание за предаване на проект, в канала на екипа в Teams.*
2. *Кода на проекта, базата данни и документацията трябва да са налични в репозитори в GitHub, което е копие на заданието генерирано в организацията.*

# 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 |

1. ПОС (от Английски POS – Point of Sale) [↑](#footnote-ref-1)
2. Салдо – разликата между приходите и разходите [↑](#footnote-ref-2)