Krzysztof Sobociński, 226187 Data wykonania ćwiczenia:

Paweł Szynal, 226026 16.11.2017 - 30.11.2017

Termin zajęć: WT 730- 900

Sala: L2.6 (C-16)



**SPRAWOZDANIE**

Laboratorium z przedmiotu:

**Inżynieria Oprogramowania INEK 011**

**Temat projektu:** Program obsługujący Dział Ewidencji Ludności

**Etap 5,6,7**

**Prowadzący:** Dr inż. Paweł Głuchowski

### Wrocław 2017

1. **Cel etapu 5**

Celem etapu piątego było stworzenie modelu projektowego programowania opartego na identyfikacji klas, reprezentujących logikę biznesową projektowanego systemu. Należało dokonać definicji atrybutów klas oraz związków między klasami - na podstawie analizy scenariuszy przypadków użycia.

1. **Etapy laboratorium**
   1. Analiza wspólności i zmienności scenariuszy przypadków użycia   
      i identyfikacji klas wraz z powiązaniami pomiędzy klasami należących do warstwy biznesowej oprogramowania, umieszczając je na diagramie klas realizowanego projektu UML w środowisku VP CE.
      1. Klas bazowych

Wyniki analizy identyfikacji klas na podstawie scenariuszy przypadków użycia.

Wykryto cztery główne klasy typu „Entity” ze względu na odpowiedzialność:

• TAplikacja (PU: Dodanie osoby do Ewidencji, Wyświetlanie danych ...mieszkańca, Aktualizacja danych osobowych, Wyszukaj mieszkańca),

• TOsoba (PU: Dodanie osoby do Ewidencji, Aktualizacja danych ...osobowych),

• TLogowanie (PU: Weryfikacja użytkownika),

• TEksportowanie (PU: Eksportowanie danych do pliku .txt)

* + 1. Klas pochodnych, powiązanych relacją Generalization

Brak.

* + 1. Klas powiązanych relacjami typu Association (Asocjacja), lub/i Dependency (Zależność) lub/i Aggregation (Agregacja słaba, Agregacja silna, czyli Kompozycja)

Wykryto następujące związki pomiędzy klasami:

• Silna agregacja między obiektem typu TAplikacja i obiektami typu ...TOsoba (aplikacja posiada kolekcję osób)

• Identyfikacji dokonano na podstawie następujących przypadków użycia: ...PU Wstawianie nowego zakupu, PU Dodanie osoby do Ewidencji, ...Aktualizacja danych osobowych.

• Silna agregacja między obiektem typu TLogowanie i obiektami typu ...TKonto (aplikacja posiada kolekcję pracowników)

• Identyfikacji dokonano na podstawie następujących przypadków użycia: ...PU Weryfikacja użytkownika

* + 1. Określić kierunek i liczność relacji z p. 2.3. pomiędzy klasami

Oznaczenia klas w UML mogą być połączone liniami oznaczającymi asocjacje, czyli powiązania pomiędzy obiektami tych klas.

Zapis liczności w UML jest naturalny i nie wprowadza specjalnych symboli graficznych. Wybrane oznaczenie \* pomiędzy TAplikacja o TOsoba oraz TAplikacja i TLogowanie oznacza dowolną liczbę.

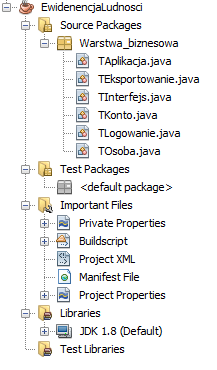
* + 1. Dokonać identyfikacji wzorców projektowych.

Wykryto następujące wzorce projektowe:

• Wzorzec Builder jako klasę TOsoba typu „Control” do tworzenia …złożonych obiektów.

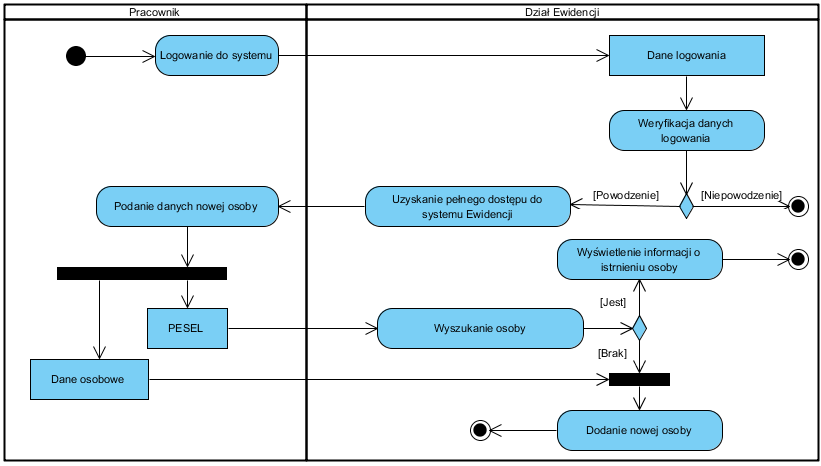
• Wzorzec strukturalny Composite jako klasę TInterfejs typu …„Boundry” który pełni rolę prezencji aplikacji.

* 1. Należy w środowisku NetBeans wykonać projekt typu Java Class Library i w pakiecie o nazwie dopasowanej do dziedziny realizowanego projektu wykonać definicje klas.

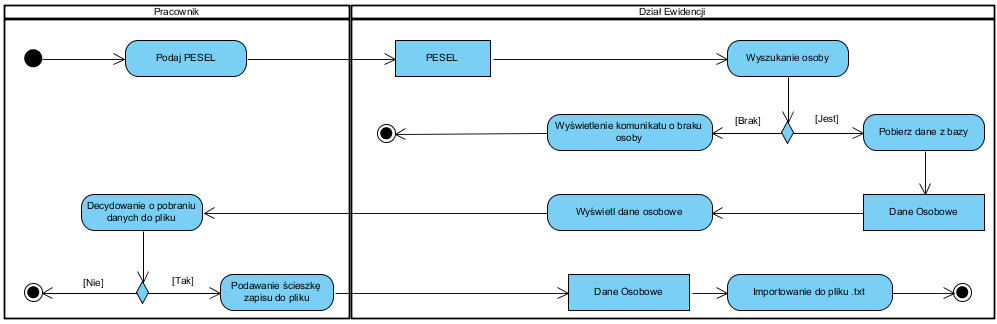


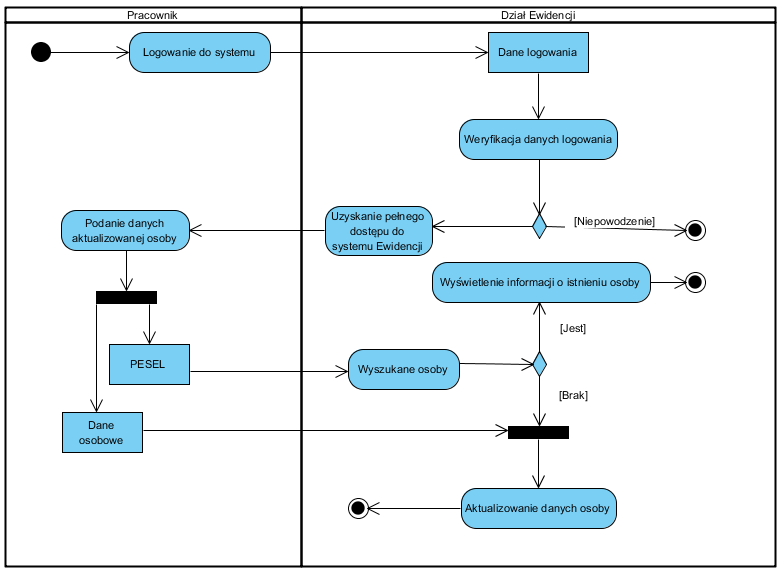
Wykonano, zdjęcia znajdują się w kolejnym zrealizowanym etapie.

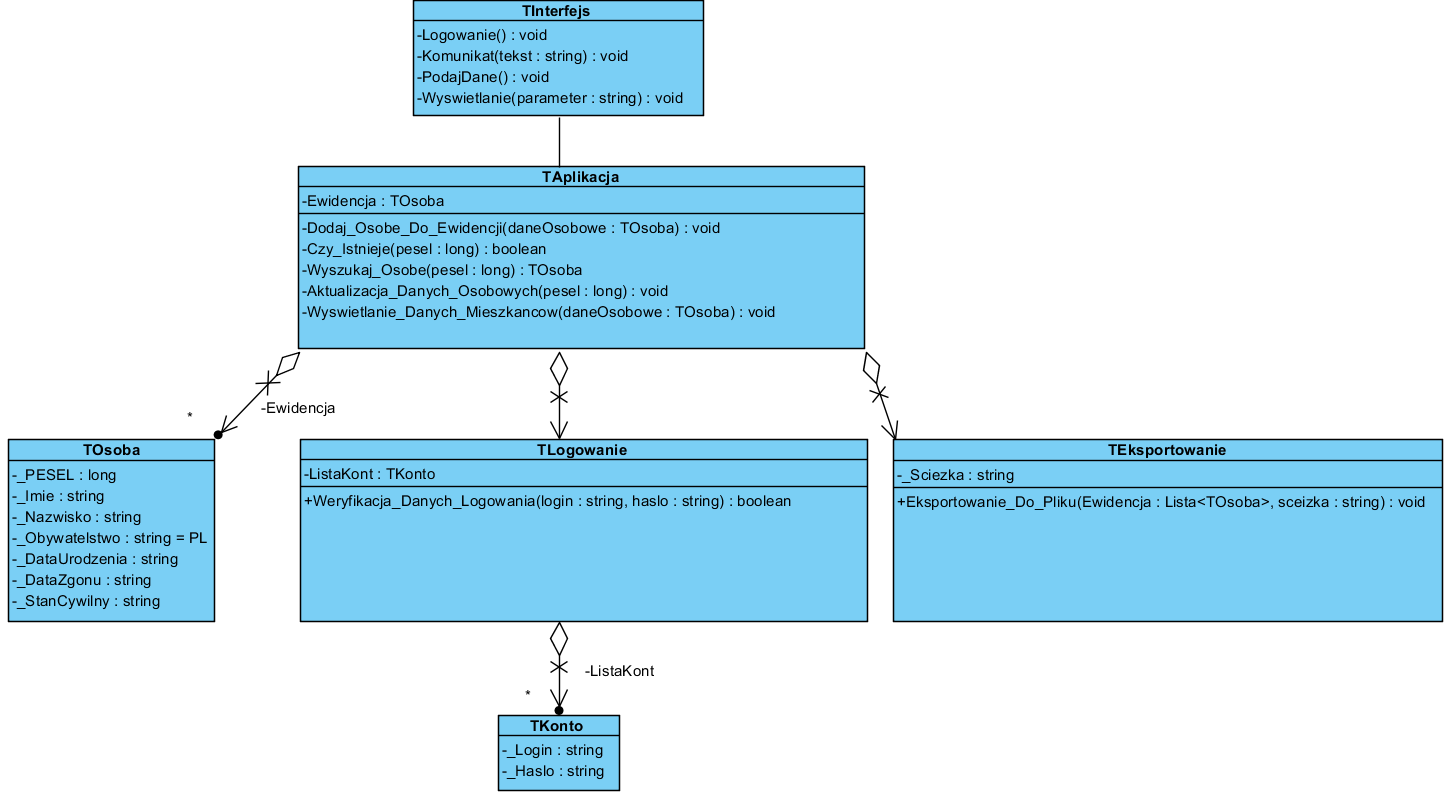
* 1. Grupa dwuosobowa powinna wykonać model projektowy 2-3 złożonych przypadków użycia tzn. opartych na relacjach <<include>> lub/i <<extend>> lub/ i <<use>> – kontynuacja prac wg instrukcji 2-4.

**Dodanie osoby do Ewidencji**

**Wyświetlanie i eksportowanie do plików**

****

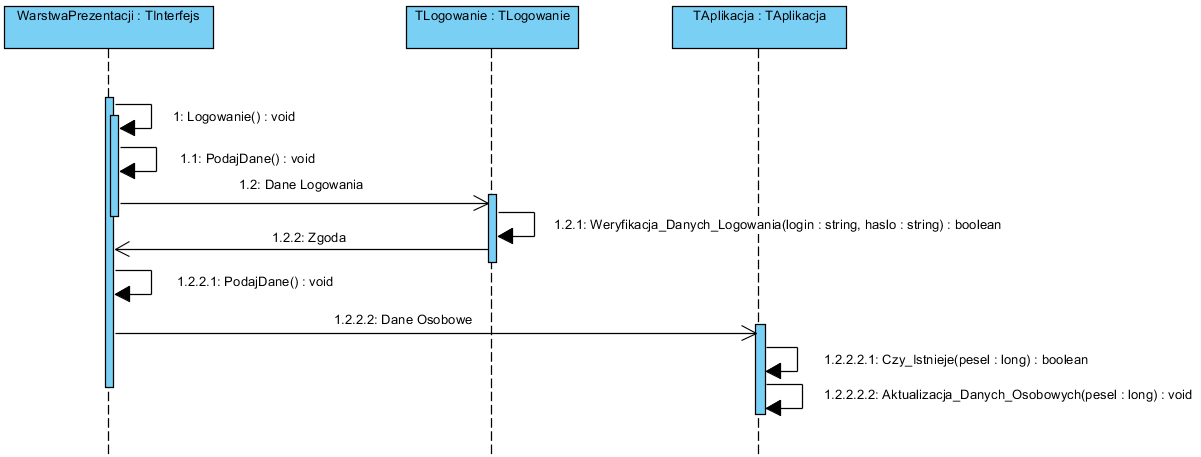
**Aktualizacja danych osobowych**

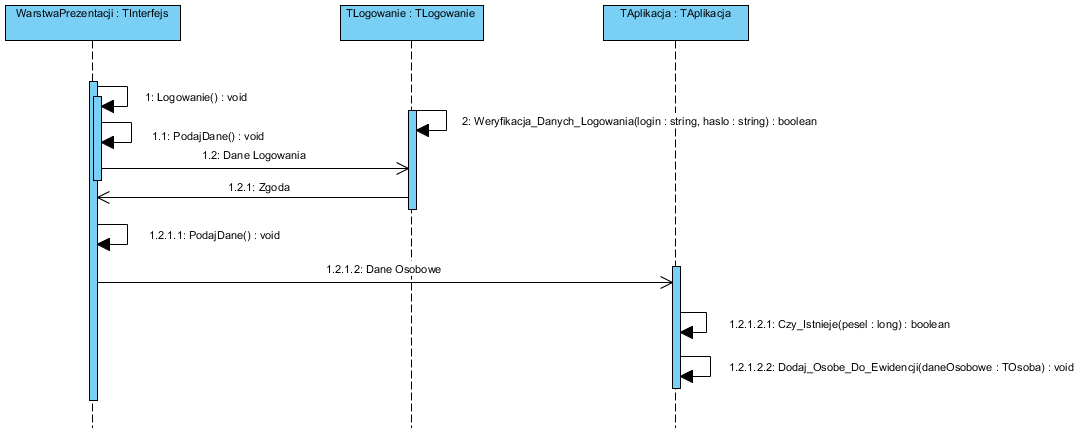
**Diagram Klas**

1. **Cel etapu 6 -7**

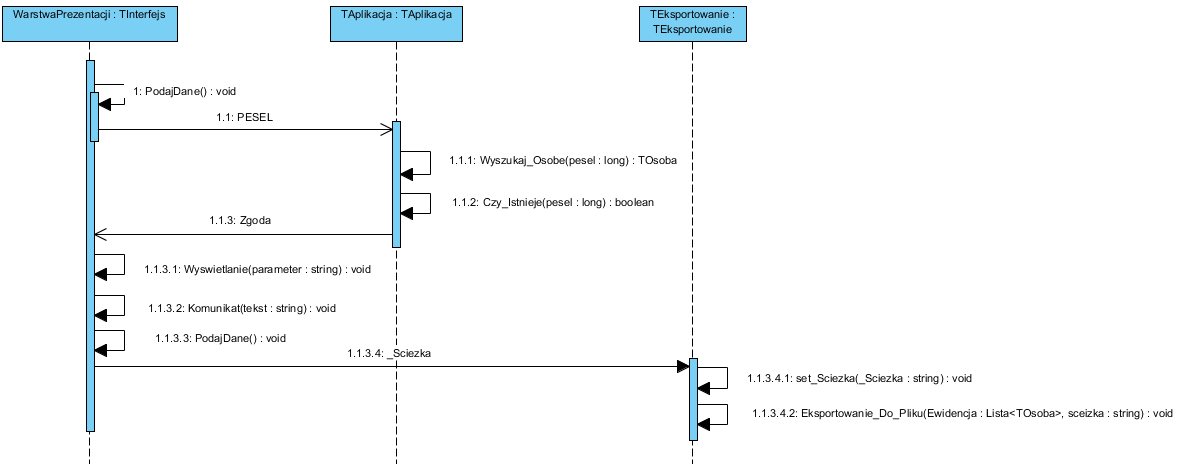
Opracowanie diagramów sekwencji dla wybranych przypadków użycia reprezentujących usługi oprogramowania wynikających również z wykonanych diagramów czynności; definicja operacji klas na podstawie diagramów sekwencji w języku Java. Zastosowanie projektowych wzorców zachowania.

1. **Etapy laboratorium**

**PU Aktualizowanie Danych Osobowych**

**PU Dodanie osoby do Ewidencji**

**PU Wyswietlanie danych mieszkanca**

****

**Kod źródłowy**

**TInterfejs**

package Warstwa\_biznesowa;

import java.util.Scanner;

import java.io.IOException;

/\*\*

\*

\* @author Pawel Szynal & Krzysztof Soboscinski

\*/

public class TInterfejs

{

public TInterfejs() {}

public void Logowanie()

{

String login,haslo;

Scanner odczyt = new Scanner(System.in);

System.out.println("Podaj login:");

login = odczyt.nextLine();

System.out.println("\nPodaj haslo:");

haslo = odczyt.nextLine();

TLogowanie weryfikacja = new TLogowanie();

weryfikacja.Weryfikacja\_Danych\_Logowania(login, haslo);

}

public void Komunikat(String tekst)

{

System.out.println(tekst);

}

public void PodajDane()

{

}

public void Wyswietlanie(String parameter)

{

}

}

**TAplikacja**

package Warstwa\_biznesowa;

import java.io.IOException;

import java.util.LinkedList;

import java.util.List;

import java.util.Scanner;

/\*\*

\*

\* @author Pawel Szynal & Krzysztof Soboscinski

\*/

public class TAplikacja

{

private List<TOsoba> \_Ewidencja;

public void Dodaj\_Osobe\_Do\_Ewidencji(TOsoba daneOsobowe)

{ \_Ewidencja.add(daneOsobowe);

}

public boolean Czy\_Istnieje(long pesel)

{

boolean czyJest = false;

for(int i =0 ; i<\_Ewidencja.size(); i++)

{

if(\_Ewidencja.get(i).Get\_PESEL() == pesel)

{

czyJest = true;

}

}

return czyJest;

}

public TOsoba Wyszukaj\_Osobe(long pesel)

{

TOsoba wyszukiwanaOsoba = new TOsoba(pesel);

for(int i=0; i<\_Ewidencja.size();i++)

{

if(\_Ewidencja.get(i).Get\_PESEL() == pesel)

{

wyszukiwanaOsoba = \_Ewidencja.get(i);

break;

}

}

return wyszukiwanaOsoba;

}

public void Aktualizacja\_Danych\_Osobowych(long pesel)

{

for(int i=0; i<\_Ewidencja.size();i++)

{

if(\_Ewidencja.get(i).Get\_PESEL() == pesel)

{

int wybor;

try

{

boolean wyjscie = true;

Scanner odczyt = new Scanner(System.in);

while(wyjscie)

{

System.out.print("Aktualzuj: \n");

System.out.print("1. Imie \n");

System.out.print("2. Nazwisko \n");

System.out.print("3. Obywatelstwo \n");

System.out.print("4. Data urodzenia \n");

System.out.print("5. Data zgonu \n");

System.out.print("6. Stan cywilny \n");

System.out.print("7. Wyjscie \n");

wybor = System.in.read();

System.out.flush(); // czyszczenie konsoli

switch (wybor)

{

case 1:

{

System.out.print("Imie: ");

\_Ewidencja.get(i).Set\_Imie(odczyt.nextLine());

break;

}

case 2:

{

System.out.print("Nazwisko: ");

\_Ewidencja.get(i).Set\_Nazwisko(odczyt.nextLine());

break;

}

case 3:

{

System.out.print("Obywatelstwo: ");

\_Ewidencja.get(i).Set\_Obywatelstwo(odczyt.nextLine());

break;

}

case 4:

{

System.out.print("Data urodzenia: ");

\_Ewidencja.get(i).Set\_DataUrodzenia(odczyt.nextLine());

break;

}

case 5:

{

System.out.print("Data zgonu: ");

\_Ewidencja.get(i).Set\_DataZgonu(odczyt.nextLine());

break;

}

case 6:

{

System.out.print("Stan cywilny ");

\_Ewidencja.get(i).Set\_StanCywilny(odczyt.nextLine());

break;

}

case 7:

{

System.out.print("Czy napewno? T/N ");

\_Ewidencja.get(i).Set\_StanCywilny(odczyt.nextLine());

if(odczyt.nextLine() == "t" || odczyt.nextLine() == "T")

{

wyjscie = false;

}

break;

}

default:

{

break;

}

}

System.out.flush(); // czyszczenie konsoli

}

}

catch (IOException e)

{

System.err.println("Nieprawidlowy format wejsciowy");

}

break;

}

}

}

public List<TOsoba> Wyswietlanie\_Danych\_Mieszkancow(TOsoba daneOsobowe)

{

List <TOsoba> wyszukiwaneOsoby = new LinkedList<TOsoba>();

for(int i = 0 ; i <\_Ewidencja.size(); i++)

{

if(\_Ewidencja.get(i).Get\_PESEL() == daneOsobowe.Get\_PESEL() ||

\_Ewidencja.get(i).Get\_Imie() == daneOsobowe.Get\_Imie() ||

\_Ewidencja.get(i).Get\_Nazwisko() == daneOsobowe.Get\_Nazwisko() ||

\_Ewidencja.get(i).Get\_Obywatelstwo() == daneOsobowe.Get\_Obywatelstwo() ||

\_Ewidencja.get(i).Get\_DataUrodzenia() == daneOsobowe.Get\_DataUrodzenia() ||

\_Ewidencja.get(i).Get\_DataZgonu() == daneOsobowe.Get\_DataZgonu() ||

\_Ewidencja.get(i).Get\_StanCywilny() == daneOsobowe.Get\_StanCywilny())

{

wyszukiwaneOsoby.add(\_Ewidencja.get(i));

}

}

for (int i = 0; i<wyszukiwaneOsoby.size();i++)

{

System.out.println("PESEL: " + wyszukiwaneOsoby.get(i).Get\_PESEL() +

" Imie: " + wyszukiwaneOsoby.get(i).Get\_Imie() +

" Nazwisko: " + wyszukiwaneOsoby.get(i).Get\_Nazwisko() +

" Obywatelstwo: " + wyszukiwaneOsoby.get(i).Get\_Obywatelstwo() +

" Data Urodzenia: " + wyszukiwaneOsoby.get(i).Get\_DataUrodzenia() +

" Data Zgonu: " + wyszukiwaneOsoby.get(i).Get\_DataZgonu() +

" Stan Cywilny: " + wyszukiwaneOsoby.get(i).Get\_StanCywilny()

);

System.out.println("\n\n");

}

System.out.println("Czy eksportowac do pliku? [T/N] ");

Scanner odczyt = new Scanner(System.in);

if(odczyt.nextLine() == "t" || odczyt.nextLine() == "T")

{

TEksportowanie eksportowanie = new TEksportowanie();

eksportowanie.Eksportowanie\_Do\_Pliku(wyszukiwaneOsoby, eksportowanie.Get\_Sciezka());

}

return wyszukiwaneOsoby;

}

}

**TLogowanie**

package Warstwa\_biznesowa;

import java.util.List;

/\*\*

\*

\* @author Pawel Szynal & Krzysztof Soboscinski

\*/

public class TLogowanie

{

private List<TKonto> \_ListaOsob;

public TLogowanie(){}

public boolean Weryfikacja\_Danych\_Logowania(String login, String haslo)

{

TKonto pracownik = new TKonto(login,haslo);

boolean weryfikacja = false;

for(int i =0 ;i< \_ListaOsob.size(); i++)

{

if(pracownik == \_ListaOsob.get(i))

{

weryfikacja = true;

}

}

return weryfikacja;

}

}

**TKonto**

package Warstwa\_biznesowa;

/\*\*

\*

\* @author Pawel Szynal & Krzysztof Soboscinski

\*/

public class TKonto

{

private String \_Login;

private String \_Haslo;

public TKonto()

{

}

public TKonto(String login , String haslo)

{

this.\_Login = login;

this.\_Haslo = haslo;

}

public void Set\_Login(String login)

{

this.\_Login = login;

}

public String Get\_Login()

{

return \_Login;

}

public void Set\_Haslo(String haslo)

{

this.\_Haslo = haslo;

}

public String Get\_Haslo()

{

return \_Haslo;

}

}

**TEksportowanie**

package Warstwa\_biznesowa;

import java.io.FileNotFoundException;

import java.io.PrintWriter;

import java.util.List;

/\*\*

\*

\* @author Pawel Szynal & Krzysztof Soboscinski

\*/

public class TEksportowanie

{

private String \_Sciezka = "plik.txt";

private PrintWriter \_ZapisDoPliku;

public TEksportowanie() {}

public String Get\_Sciezka()

{

return this.\_Sciezka;

}

public void Eksportowanie\_Do\_Pliku( List <TOsoba> ewidencja , String sciezka)

{

try

{

\_ZapisDoPliku = new PrintWriter(sciezka);

for(int i =0 ; i < ewidencja.size() ; i++)

{

\_ZapisDoPliku.println("PESEL: " + ewidencja.get(i).Get\_PESEL() +

" Imie: " + ewidencja.get(i).Get\_Imie() +

" Nazwisko: " + ewidencja.get(i).Get\_Nazwisko() +

" Obywatelstwo: " + ewidencja.get(i).Get\_Obywatelstwo() +

" Data Urodzenia: " + ewidencja.get(i).Get\_DataUrodzenia() +

" Data Zgonu: " + ewidencja.get(i).Get\_DataZgonu() +

" Stan Cywilny: " + ewidencja.get(i).Get\_StanCywilny()

);

\_ZapisDoPliku.println("\n");

}

\_ZapisDoPliku.close();

}

catch (FileNotFoundException e )

{

System.err.println("Nieprawidlowa sciezka");

}

}

}

**TOsoba**

package Warstwa\_biznesowa;

/\*\*

\*

\* @author Pawel Szynal & Krzysztof Soboscinski

\*/

public class TOsoba

{

private long \_PESEL;

private String \_Imie;

private String \_Nazwisko;

private String \_Obywatelstwo = "PL";

private String \_DataUrodzenia;

private String \_DataZgonu;

private String \_StanCywilny;

public TOsoba(long pesel )

{

this.\_PESEL = pesel;

}

public TOsoba(long pesel , String imie, String nazwisko, String obywatelstwo, String dataUrodzenia,String dataZgonu, String stanCywilny)

{

this.\_PESEL = pesel;

this.\_Imie = imie;

this.\_Nazwisko = nazwisko;

this.\_Obywatelstwo = obywatelstwo;

this.\_DataUrodzenia = dataUrodzenia;

this.\_DataZgonu = dataZgonu;

this.\_StanCywilny = stanCywilny;

}

public long Get\_PESEL()

{

return \_PESEL;

}

public void Set\_Imie(String imie)

{

this.\_Imie = imie;

}

public String Get\_Imie()

{

return \_Imie;

}

public void Set\_Nazwisko(String nazwisko)

{

this.\_Nazwisko = nazwisko;

}

public String Get\_Nazwisko()

{

return \_Nazwisko;

}

public void Set\_Obywatelstwo(String obywatelstwo)

{

this.\_Obywatelstwo = obywatelstwo;

}

public String Get\_Obywatelstwo()

{

return \_Obywatelstwo;

}

public void Set\_StanCywilny(String stanCywilny)

{

this.\_StanCywilny = stanCywilny;

}

public String Get\_StanCywilny()

{

return \_StanCywilny;

}

public void Set\_DataUrodzenia(String dataUrodzenia)

{

this.\_DataUrodzenia = dataUrodzenia;

}

public String Get\_DataUrodzenia()

{

return \_DataUrodzenia;

}

public void Set\_DataZgonu(String dataZgonu)

{

this.\_DataZgonu = dataZgonu;

}

public String Get\_DataZgonu()

{

return \_DataZgonu;

}

}

1. **Wnioski**

Wszystkie założone cele do zrealizowania, dla tego laboratorium zostały wykonane. Stworzony diagramy klas oraz diagramy sekwencji ułatwiły nam zrozumienie całego przebiegu użycia programu w konkretnych przypadkach użycia oraz ułatwiły implementację metod.