



AKADEMIA GÓRNICZO-HUTNICZA
im. Stanisława Staszica w Krakowie

WYDZIAŁ ZARZĄDZANIA



Programowanie Obiektowe

Dokumentacja Techniczna projektu zaliczeniowego
Temat projektu: System rezerwacji biletów firmy Łęcz-Trans
Informatyka i ekonometria

Wykonały:
Natalia Trąba
Andżelika Wójcik
Urszula Zastawna

Kraków, 22.01.2024

Spis treści

1.	Opis projektu:	3
2.	Zaimplementowane klasy.....	3
2.1	Klasa Bilet	3
2.2	Klasa BiletUlgowy	4
2.3	Klasa BiletNormalny	4
1.4	Klasa Kurs.....	5
1.5	Klasa ZarzadzanieBiletami	5
1.6	Klasa ZarzadzanieBiletamiException	6
1.7	Klasa ListaBiletów	6
1.8	Klasa ZakupBiletu	7
1.9	Klasa StrefaAdministratora.....	8
1.10	Klasa Zwrot	8
1.11	Klasa PotwierdzenieU	9
1.12	Klasa PotwierdzenieN.....	9
1.13	Klasa MainWindow	10
2	Okna	11
2.1	Okno MainWindow	11
2.2	Okno ZakupBiletu	11
2.3	Okno PotwierdzenieU	12
2.4	Okno PotwierdzenieN.....	12
2.5	Okno StrefaAdministratora	13
3	Instrukcja działania.....	13

1. Opis projektu:

Nazwa projektu: System Rezerwacji Biletów dla Firmy Łęcz-Trans

Cel projektu:

Celem projektu jest stworzenie systemu rezerwacji biletów dla firmy transportowej Łęcz-Trans. System ma umożliwiać klientom wygodne dokonywanie zakupów biletów online oraz zapewniać administratorowi narzędzia do zarządzania zakupionymi biletami.

Główne funkcje systemu:

1. Zakup Biletów:
 - Klienci mogą wybierać trasę, datę, godzinę oraz rodzaj biletu (normalny, ulgowy).
 - Podczas zakupu klient wprowadza swoje dane osobowe, takie jak imię, nazwisko, adres e-mail, i numer telefonu.
 - Klient może wybrać miejsce siedzące, a także dostosować preferencje dotyczące biletu.
2. Zwroty:
 - Użytkownicy mają możliwość dokonywania zwrotów zakupionych biletów.
 - System sprawdza, czy dany bilet jest zwracalny, a następnie zwalnia miejsce na wybranym kursie.
 - Klient otrzymuje informacje zwrotne o statusie operacji.
3. Strefa Administratora:
 - Administrator ma dostęp do specjalnej strefy, w której może przeglądać zakupione bilety.
 - Może sortować listę biletów według różnych kryteriów (data, imię i nazwisko, itp.).
 - Zarządza również zwrotami biletów i podejmuje inne działania administracyjne.
4. Generowanie Biletów w Formie PDF:
 - System generuje bilety w formie plików PDF, które są zapisywane na urządzeniu użytkownika.
 - Bilety zawierają informacje o podróży, dane klienta, datę zakupu, itp.
5. Regulamin:
 - Użytkownicy mają dostęp do regulaminu firmy Łęcz-Trans, który zawiera ważne informacje dotyczące korzystania z usług.

2. Zaimplementowane klasy

2.1 Klasa Bilet

Klasa Bilet jest klasą abstrakcyjną – jest używana jako klasa bazowa dla klas pochodnych. Implementuje interfejsy `Comparable<Bilet>` oraz `Equatable<Bilet>`.

Pola:

idStatic: Zmienna statyczna przechowująca identyfikator statyczny, zainicjowana wartością 2137.

imieNazwisko: Zmienna przechowująca imię i nazwisko posiadacza biletu.

email: Zmienna przechowująca adres e-mail posiadacza biletu.

idBiletu: Zmienna przechowująca unikalny identyfikator biletu.

nrTelefonu: Zmienna przechowująca numer telefonu posiadacza biletu.

rodzaj: Zmienna przechowująca rodzaj biletu (normalny lub ulgowy).

dataKupna: Zmienna przechowująca datę zakupu biletu.

idKursu: Zmienna przechowująca identyfikator kursu, na który został zakupiony bilet.

Właściwości:

ImieNazwisko: Właściwość dostępowa do pola imieNazwisko.

Email: Właściwość dostępowa do pola email.

NrTelefonu: Właściwość dostępowa do pola nrTelefonu.

Rodzaj: Właściwość abstrakcyjna reprezentująca rodzaj biletu, implementowana w klasach pochodnych.

DataKupna: Właściwość dostępowa do pola dataKupna.

IdKursu: Właściwość dostępowa do pola idKursu.

IdStatic: Właściwość dostępowa do pola idStatic.

IdBiletu: Właściwość dostępowa do pola idBiletu.

Konstruktor:

Bilet(): Konstruktor bezparametrowy inicjujący pola, zwłaszcza nadając unikalny identyfikator biletu (idBiletu).

Metody:

CompareTo(Bilet? other): Implementacja metody z interfejsu IComparable<Bilet>.

Porównuje bilet z innym biletu na podstawie imienia i nazwiska, a następnie identyfikatora biletu.

Equals(Bilet? other): Implementacja metody z interfejsu IEquatable<Bilet>. Porównuje bilet z innym biletu na podstawie identyfikatora biletu.

2.2 Klasa BiletUlgowy

Klasa BiletUlgowy dziedziczy po klasie abstrakcyjnej Bilet i reprezentuje bilet ulgowy w systemie rezerwacji biletów autobusowych.

Pola:

cena: Cena biletu ulgowego, ustalona na 5.50 zł.

Właściwości:

Rodzaj: Właściwość tylko do odczytu, implementująca abstrakcyjną właściwość z klasy bazowej Bilet. Określa rodzaj biletu jako "ulgowy".

Cena: Właściwość dostępowa do pola cena, umożliwiającą odczyt i ewentualne modyfikacje ceny biletu.

Konstruktor:

Konstruktor domyślny: Inicjalizuje pole cena na wartość 5.50.

2.3 Klasa BiletNormalny

Klasa BiletNormalny dziedziczy po klasie abstrakcyjnej Bilet i reprezentuje bilet normalny w systemie rezerwacji biletów autobusowych.

Pola:

cena: Cena biletu normalnego, ustalona na 7.00 zł.

Właściwości:

Rodzaj: Właściwość tylko do odczytu, implementująca abstrakcyjną właściwość z klasy bazowej Bilet. Określa rodzaj biletu jako "normalny".

Cena: Właściwość dostępowa do pola cena, umożliwiająca odczyt i ewentualne modyfikacje ceny biletu.

Konstruktor:

Konstruktor domyślny: Inicjalizuje pole cena na wartość 7.00.

1.4 Klasa Kurs

Klasa Kurs reprezentuje kurs autobusowy i implementuje interfejs ICloneable w celu umożliwienia klonowania obiektów tej klasy.

Pola:

autobus: Przechowuje informację o autobusie przypisanym do kursu.

dostepneMiejsca: Określa ilość dostępnych miejsc na autobusie dla danego kursu.

trasa: Określa trasę, którą pokonuje autobus w ramach kursu.

dataKursu: Określa datę kursu.

godzina: Określa godzinę rozpoczęcia kursu.

idKursu: Unikalny identyfikator kursu, generowany na podstawie daty, godziny i trasy.

Właściwości:

IdKursu: Właściwość tylko do odczytu, generowana dynamicznie na podstawie daty, godziny i trasy kursu.

Autobus: Właściwość dostępowa do pola autobus.

DataKursu: Właściwość dostępowa do pola dataKursu.

DostepneMiejsca: Właściwość dostępowa do pola dostepneMiejsca.

Trasa: Właściwość dostępowa do pola trasa.

Godzina: Właściwość dostępowa do pola godzina.

Metoda:

Clone(): Implementacja metody z interfejsu ICloneable. Klonuje obiekt klasy Kurs za pomocą MemberwiseClone().

Typy Wyliczeniowe:

Autobus: Enumeracja reprezentująca różne rodzaje autobusów.

Trasa: Enumeracja reprezentująca różne trasy kursów autobusowych.

1.5 Klasa ZarzadzanieBiletami

Klasa ZarzadzanieBiletami zarządza procesem zakupu, zwrotu biletów oraz generowaniem plików PDF dla biletów w systemie rezerwacji biletów autobusowych.

Konstruktor:

Inicjalizuje listę kursów (listaKursow) poprzez wywołanie metody InicjalizujKursy.

Wywołuje metodę WczytajBilety do wczytania biletów.

Metody:

InicjalizujKursy: Tworzy i dodaje do listy kursów różne kursy autobusowe na przestrzeni 14 dni.

PobierzDostepneKursy: Zwraca listę dostępnych kursów, czyli kursów, na które można jeszcze zakupić bilety.

KupBiletUlgowy i KupBiletNormalny:

- Kupuje bilet ulgowy/normalny i dodaje go do odpowiedniej listy (listaBiletowUlgowych/listaBiletowNormalnych).
- Aktualizuje ilość dostępnych miejsc na kursie.
- Zapisuje bilet do pliku XML poprzez wywołanie metody ZapiszBilety.
- Generuje plik PDF z danymi o zakupionym bilecie.

ZwrocBiletUlgowy i ZwrocBiletNormalny:

- Zwraca bilet ulgowy/normalny, zwiększając ilość dostępnych miejsc na kursie.
- Usuwa bilet z odpowiedniej listy (listaBiletowUlgowych/listaBiletowNormalnych).
- Zapisuje zmiany do pliku XML poprzez wywołanie metody ZapiszBilety.

GenerujPdfUlgowy i GenerujPdfNormalny: Generują plik PDF z danymi o zakupionym bilecie ulgowym/normalnym.

1.6 Klasa ZarzadzanieBiletamiException

Klasa ZarzadzanieBiletamiException dziedziczy po Exception i służy do obsługi błędów związanych z zarządzaniem biletami. Posiada trzy konstruktory, z których jeden przekazuje komunikat o błędzie, a drugi również zawiera informacje o wewnętrznym wyjątku (inner exception).

1.7 Klasa ListaBiletów

Klasa ListaBiletow jest odpowiedzialna za przechowywanie i zarządzanie listami biletów ulgowych i normalnych oraz obsługę zapisu i odczytu danych z pliku XML.

Pola:

listaBiletowUlgowych i listaBiletowNormalnych: Listy przechowujące obiekty biletów ulgowych i normalnych, które są zapisywane do pliku i wczytywane z pliku.

sciezkaBilety: Ścieżka do pliku XML, w którym są zapisywane dane o biletach.

Konstruktor:

Inicjalizuje listy biletów ulgowych i normalnych.

Metody:

ZapiszBilety:

- Tworzy nowy obiekt ListaBiletow, który zawiera listy biletów ulgowych i normalnych.
- Serializuje ten obiekt do formatu XML i zapisuje do pliku o ścieżce sciezkaBilety.

WczytajBilety:

- Deserializuje dane z pliku XML do obiektu ListaBiletow.
- Aktualizuje listy biletów ulgowych i normalnych na podstawie danych wczytanych z pliku.

1.8 Klasa ZakupBiletu

Klasa ZakupBiletu odpowiada za interfejs graficzny użytkownika (GUI) związanego z procesem zakupu biletów.

Pola i Właściwości:

Kurs wybranyKurs: Pole przechowujące informacje o wybranym kursie.

List<Kurs> listaKursow: Pole przechowujące listę dostępnych kursów.

Konstruktor:

- Inicjalizuje komponenty interfejsu użytkownika.
- Pobiera dostępne kursy poprzez wywołanie metody PobierzDostepneKursy.
- Ustawia źródło danych dla kontrolki TrasaCb na wartości zdefiniowane w enumie Trasa.
- Dodaje obsługę zdarzenia SelectionChanged dla kontrolki GodzinaCb, przypisując mu metodę GodzinaCb_SelectionChanged.

Metody:

DalejBtn_Click:

- Sprawdza, czy użytkownik wybrał kurs oraz czy pola są poprawnie wypełnione.
- Tworzy nowy bilet (normalny lub ulgowy) w zależności od wyboru użytkownika.
- Wyświetla okno potwierdzenia (PotwierdzenieN lub PotwierdzenieU) i obsługuje rezultat.
- Przy pomyślnym zakupie, wyświetla komunikat i przechodzi do głównego okna MainWindow.

SprawdzImieNazwisko, SprawdzEmail, SprawdzNumer: Wykorzystują wyrażenia regularne do sprawdzania poprawności wprowadzonych danych (imię i nazwisko, adres e-mail, numer telefonu).

SprawdzPola:

- Sprawdza, czy wszystkie pola są wypełnione poprawnie.
- Wykorzystuje wcześniej zdefiniowane metody do walidacji imienia i nazwiska, e-maila oraz numeru telefonu.
- Zwraca wartość logiczną w zależności od poprawności pól.

PobierzDostepneKursy: Tworzy instancję klasy ZarzadzanieBiletami i wywołuje jej metodę PobierzDostepneKursy.

ObliczCeneBiletu: Przyjmuje jako argument rodzaj biletu (RodzajBiletu) i zwraca cenę biletu na podstawie tego rodzaju.

DatePicker_SelectedDateChanged: Wywołuje metodę OdswiezKursy po zmianie wybranej daty w kontrolce DatePicker.

TrasaCb_SelectionChanged: Wywołuje metodę OdswiezKursy po zmianie wybranej trasy w kontrolce TrasaCb.

OdswiezKursy:

- Pobiera datę i trasę z kontrolki DatePicker i TrasaCb.
- Tworzy instancję klasy ZarzadzanieBiletami i wywołuje jej metodę PobierzDostepneKursy.
- Aktualizuje źródło danych dla kontrolki GodzinaCb na podstawie dostępnych kursów.

GodzinaCb_SelectionChanged: Ustawia pole wybranyKurs na kurs wybrany przez użytkownika w kontrolce GodzinaCb.

1.9 Klasa StrefaAdministratora

Klasa StrefaAdministratora odpowiada za interfejs graficzny użytkownika (GUI) w strefie administratora, umożliwiającą przeglądanie i sortowanie zakupionych biletów. P

Pola i Właściwości:

ZarządzanieBiletami **zarządzanieBiletami**: Pole przechowujące instancję klasy ZarządzanieBiletami.

Konstruktor:

- Inicjalizuje komponenty interfejsu użytkownika.
- Tworzy nową instancję ZarządzanieBiletami.
- Ustawia źródło danych dla kontrolki lbZakupioneBilety na połączoną listę biletów ulgowych i normalnych.

Metody:

BtnDataKupnaSort_Click:

- Obsługuje kliknięcie przycisku sortowania po dacie zakupu.
- Czyści aktualne ustawienia sortowania, a następnie dodaje nowe ustawienie dla kolumny DataKupna rosnąco.

BtnImieNazwiskoSort_Click:

- Obsługuje kliknięcie przycisku sortowania po imieniu i nazwisku.
- Czyści aktualne ustawienia sortowania, a następnie dodaje nowe ustawienie dla kolumny ImieNazwisko rosnąco.

BtnIdKursuSort_Click:

- Obsługuje kliknięcie przycisku sortowania po identyfikatorze kursu.
- Czyści aktualne ustawienia sortowania, a następnie dodaje nowe ustawienie dla kolumny IdKursu rosnąco.

BtnIdBiletuSort_Click:

- Obsługuje kliknięcie przycisku sortowania po identyfikatorze biletu.
- Czyści aktualne ustawienia sortowania, a następnie dodaje nowe ustawienie dla kolumny IdBiletu rosnąco.

BtnPowrot_Click:

- Obsługuje kliknięcie przycisku powrotu.
- Tworzy nową instancję MainWindow, wyświetla ją i zamyka aktualne okno.

1.10 Klasa Zwrot

Klasa Zwrot odpowiada za obsługę okna zwrotu biletu w systemie rezerwacji. Umożliwia użytkownikowi wybór biletu do zwrotu, potwierdzenie operacji oraz nawigację do głównego okna.

Składowe klasy:

1. Pola

zarządzanieBiletami (typ: ZarządzanieBiletami): Pole przechowujące instancję klasy ZarządzanieBiletami do zarządzania biletami.

2. Konstruktor

Zwrot(): Konstruktor inicjalizujący okno zwrotu biletu, ustawiający źródło danych dla ComboBoxa oraz przypisujący obsługę zdarzeń.

3. Metody

PowrotBtn_Click()- Obsługuje kliknięcie przycisku "Powrót". Otwiera główne okno programu i zamyka okno zwrotu.

ZwrotBtn_Click() - Obsługuje kliknięcie przycisku "Zwrot". Sprawdza, czy wybrano bilet do zwrotu, a następnie próbuje zrealizować zwrot biletu za pomocą klasy ZarzadzanieBiletami. Wyświetla komunikaty o sukcesie lub błędzie i aktualizuje widok.

ZwrotCb_SelectionChanged() - Obsługuje zmianę wybranego biletu w ComboBoxie.

1.11 Klasa PotwierdzenieU

Klasa PotwierdzenieU odpowiada za interfejs graficzny użytkownika (GUI) potwierdzenia zakupu biletu ulgowego.

Pola i Właściwości:

BiletUlgowy nowyBilet: Pole przechowujące obiekt reprezentujący nowy bilet ulgowy.

bool CzyPotwierdzono: Właściwość publiczna przechowująca informację, czy potwierdzono zakup biletu.

Konstruktor:

- Inicjalizuje komponenty interfejsu użytkownika.
- Przekazuje obiekt reprezentujący nowy bilet ulgowy do klasy.
- Wywołuje metodę WyświetlCene() do wyświetlenia ceny biletu.

Metody:

WyświetlCene():

- Aktualizuje zawartość etykiety CenaLbl wyświetlając informacje o zakupionym bilecie, takie jak numer biletu, rodzaj biletu i cena.

AnulujBtn_Click:

- Obsługuje kliknięcie przycisku "Anuluj".
- Ustawia flagę CzyPotwierdzono na false.
- Zamyka okno potwierdzenia zakupu.

PotwierdzamBtn_Click:

- Obsługuje kliknięcie przycisku "Potwierdzam".
- Ustawia flagę CzyPotwierdzono na true.
- Ustawia DialogResult na true, co oznacza potwierdzenie dialogu.
- Zamyka okno potwierdzenia zakupu.

1.12 Klasa PotwierdzenieN

Klasa PotwierdzenieN odpowiada za interfejs graficzny użytkownika (GUI) potwierdzenia zakupu biletu normalnego.

Pola i Właściwości:

BiletNormalny nowyBilet: Pole przechowujące obiekt reprezentujący nowy bilet normalny.

CzyPotwierdzono: Właściwość publiczna przechowująca informację, czy potwierdzono zakup biletu.

Konstruktor:

- Inicjalizuje komponenty interfejsu użytkownika.
- Przekazuje obiekt reprezentujący nowy bilet normalny do klasy.
- Wywołuje metodę WyświetlCene() do wyświetlenia ceny biletu.

Metody:

WyświetlCene():Aktualizuje zawartość etykiety CenaLbl wyświetlając informacje o zakupionym bilecie, takie jak numer biletu, rodzaj biletu i cena.

AnulujBtn_Click:

- Obsługuje kliknięcie przycisku "Anuluj".
- Ustawia flagę CzyPotwierdzono na false.
- Zamyka okno potwierdzenia zakupu.

PotwierdzamBtn_Click:

- Obsługuje kliknięcie przycisku "Potwierdzam".
- Ustawia flagę CzyPotwierdzono na true.
- Ustawia DialogResult na true, co oznacza potwierdzenie dialogu.
- Zamyka okno potwierdzenia zakupu.

1.13 Klasa MainWindow

Klasa MainWindow odpowiada za interfejs graficzny głównego okna aplikacji.

Konstruktor: Inicjalizuje komponenty interfejsu użytkownika.

Metody:

BtnRegulamin_Click:

- Obsługuje kliknięcie przycisku "Regulamin".
- Wyświetla okno dialogowe (MessageBox) z informacjami dotyczącymi regulaminu firmy Łęcz-Trans.

BtnKup_Click:

- Obsługuje kliknięcie przycisku "Kup".
- Tworzy nowe okno klasy ZakupBiletu i je wyświetla.
- Zamyka bieżące okno.

BtnZwroc_Click:

- Obsługuje kliknięcie przycisku "Zwrot".
- Tworzy nowe okno klasy Zwrot i je wyświetla.
- Zamyka bieżące okno.

BtnStrefa_Click:

- Obsługuje kliknięcie przycisku "Strefa".
- Tworzy nowe okno klasy StrefaAdministratora i je wyświetla.
- Zamyka bieżące okno.

2 Okna

2.1 Okno MainWindow



2.2 Okno ZakupBiletu

The screenshot shows the 'ZakupBiletu' window of the 'ŁęczTrans' application. The title bar reads 'ZakupBiletu'. The main area has a yellow background with a diagonal line pattern. At the top, it says 'System rezerwacji biletów firmy Łęcz-Trans'. The form contains the following fields and controls:

- Imię i nazwisko:** A text input field.
- E-mail:** A text input field.
- Nr telefonu:** A text input field.
- Trasa:** A dropdown menu.
- Data podróży:** A date picker with the text 'Wybierz datę' and a calendar icon showing the number 15.
- Godzina odjazdu:** A dropdown menu.
- Rodzaj biletu:** Two radio buttons labeled 'Normalny' and 'Ulgowy'.
- Dalej:** A button at the bottom right.

2.3 Okno PotwierdzenieU



2.4 Okno PotwierdzenieN



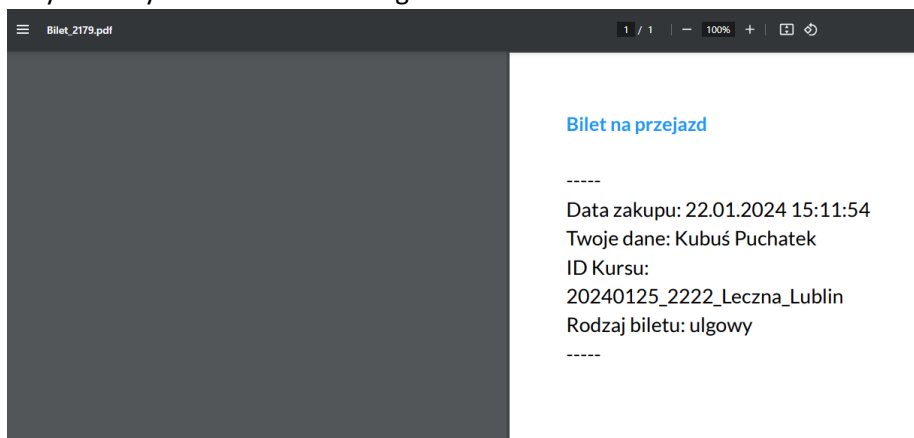
2.5 Okno StrefaAdministratora



3 Instrukcja działania

1. Główne Okno Aplikacji (MainWindow):
Po uruchomieniu aplikacji użytkownikowi ukazuje się główne okno (MainWindow), w którym może podjąć różne działania.
2. Przycisk "Regulamin":
Po kliknięciu przycisku "Regulamin", użytkownik otrzymuje informacje dotyczące regulaminu firmy Łęcz-Trans za pomocą okna dialogowego.
3. Przycisk "Kup":
Po kliknięciu przycisku "Kup", otwiera się nowe okno (ZakupBiletu), gdzie użytkownik może dokonać zakupu biletu. Użytkownik wybiera trasę, datę oraz godzinę kursu, a następnie wprowadza swoje dane osobowe. Może również wybrać rodzaj biletu (normalny lub ulgowy) oraz podać swoje preferencje, np. miejsce siedzące. Po potwierdzeniu zakupu, bilet zostaje zapisany, a użytkownik dostaje informacje o pomyślnym zakupie.

Przykładowy szablon drukowanego biletu:



4. Przycisk "Zwrot":

Po kliknięciu przycisku "Zwrot", otwiera się nowe okno (Zwrot), gdzie użytkownik może zwrócić wcześniej zakupiony bilet. Użytkownik podaje identyfikator biletu, a system sprawdza, czy taki bilet istnieje i czy można go zwrócić. Po potwierdzeniu zwrotu, miejsce na danym kursie zostaje zwolnione, a użytkownik otrzymuje informacje o pomyślnym zwrocie biletu.

5. Przycisk "Strefa":

Po kliknięciu przycisku "Strefa", otwiera się nowe okno (StrefaAdministratora), dostępne tylko dla administratora. Administrator może przeglądać i zarządzać zakupionymi biletami, sortować je według różnych kryteriów oraz przejść do innych funkcji administracyjnych.

6. Zakończenie Aplikacji:

Po zakończeniu każdej operacji użytkownik może zamknąć bieżące okno i wrócić do głównego okna (MainWindow), gdzie może wybrać kolejną akcję lub zakończyć działanie aplikacji.




PotwierdzenieU

Bilet nr: 2157
Rodzaj biletu: ulgowy
Cena: 5,50 zł

AnulujPotwierdzam i płacę

Sukces

 Bilet został pomyślnie kupiony.
Bilet PDF został zapisany na urządzeniu.

OK

Strefa Administratora

Zakupione bilety:

BILET: 2179
Imie i Nazwisko: Kubuś Puchatek
Email: kubus@interia.pl
Nr telefonu: 333-444-555
Rodzaj biletu: ulgowy
Kurs: 20240125_2222_Leczna_Lublin
Data zakupu: 1/22/2024 3:11:54 PM

BILET: 2164
Imie i Nazwisko: Papa Smerf
Email: papa@smerfy.com
Nr telefonu: 343-323-343
Rodzaj biletu: ulgowy
Kurs: 20240125_0622_Leczna_Lublin
Data zakupu: 1/22/2024 4:00:07 PM

BILET: 2157
Imie i Nazwisko: Natalia Kowalska
Email: nataliakowal@studentagh.edu.pl
Nr telefonu: 763-782-768
Rodzaj biletu: ulgowy

Sortuj po Id biletu
Sortuj po imieniu i nazwisku
Sortuj po dacie zakupu
Sortuj po Id kursu
Powrót na stronę główną


Zwrot

Wybierz ID biletu do zwrotu:

2164

Potwierdź zwrot Anuluj

Sukces

 Bilet ulgowy został zwrócony.

OK