



**WYŻSZA SZKOŁA  
INFORMATYKI i ZARZĄDZANIA**  
z siedzibą w Rzeszowie

## **KOLEGIUM INFORMATYKI STOSOWANEJ**

**Kierunek: INFORMATYKA**

**Specjalność: Technologie internetowe i mobilne**

Zuzanna Cupryś  
Nr albumu studenta w67387

### ***System obsługi biura podróży***

Prowadzący: mgr inż. Ewa Żesławska

**Praca projektowa programowanie obiektowe C#**

**Rzeszów 2025**



# Spis treści

<b>Wstęp</b>	<b>4</b>
<b>1 Opis założeń projektu</b>	<b>5</b>
1.1 Cele projektu . . . . .	5
1.2 Wymagania funkcjonalne . . . . .	5
1.3 Wymagania niefunkcjonalne . . . . .	6
<b>2 Struktura projektu</b>	<b>7</b>
2.1 Diagram klas . . . . .	7
2.1.1 Klasa Connection_Class . . . . .	7
2.1.2 Klasa formBookTrip . . . . .	8
2.1.3 Klasa formClientVersion . . . . .	8
2.1.4 Klasa formEditTripDetails . . . . .	8
2.1.5 Klasa formEmployeeVersion . . . . .	8
2.1.6 Klasa formLogin . . . . .	9
2.1.7 Klasa formTripDetails . . . . .	9
2.1.8 Klasa Program . . . . .	9
2.2 Opis techniczny . . . . .	9
2.3 Minimalne wymagania sprzętowe . . . . .	9
<b>3 Harmonogram realizacji projektu</b>	<b>10</b>
3.1 Diagram Gantta . . . . .	10
3.2 Repozytorium . . . . .	10
<b>4 Prezentacja warstwy użytkowej projektu</b>	<b>11</b>
4.1 Ekran główny . . . . .	11
4.2 Podgląd wycieczki . . . . .	12
4.3 Rezerwacja . . . . .	13
4.4 Logowanie pracownika . . . . .	22
4.5 Tryb pracownika . . . . .	25
<b>5 Podsumowanie</b>	<b>28</b>
<b>Bibliografia</b>	<b>29</b>
<b>Spis rysunków</b>	<b>30</b>

# Wstęp

Współczesny rynek turystyczny charakteryzuje się szybkim, jak i dynamicznym rozwojem, a także stale rosnącą konkurencją między biurami podróży. Klienci stają się coraz bardziej wymagający, oczekują szybkiego dostępu do informacji, wygodnych, przejrzystych narzędzi do rezerwacji oraz spersonalizowanej obsługi. Biura podróży nie stając na wysokości zadania jakim jest dopasowanie się do oczekiwań konsumenta, ryzykują utratą klientów na rzecz zmodernizowanych konkurentów. Projekt "System obsługi biura podróży" ma na celu stworzenie nowoczesnego narzędzia odpowiadającego na potrzeby zarówno pracowników biura, jak i klientów, zapewniając wygodę obsługi, przejrzystość i wspierając podstawowe funkcje, takie jak zarządzanie ofertą turystyczną, oraz automatyzację procesów rezerwacyjnych.

# Rozdział 1

## Opis założeń projektu

### 1.1 Cele projektu

Głównym celem projektu jest stworzenie i opracowanie systemu wspierającego obsługę biura podróży, umożliwiającego zarządzanie rezerwacjami i ofertą turystyczną. Ma on poprawić jakość obsługi klienta i uprościć procesy operacyjne co zwiększy efektywność działań. Problemem, który projekt ma rozwiązać jest czasochłonne i mało intuicyjne obsługiwanie procesów rezerwacyjnych w biurach podróży. Firmy w dalszym ciągu korzystają ze standardowych rozwiązań, takich jak używanie papierowych formularzy lub aplikacji, które nie są ze sobą zintegrowane, co prowadzi do licznych błędów, utrudnionego dostępu do bazy klientów oraz braku możliwości uzyskania aktualnych informacji o ofertach. Głównym źródłem problemu jest unikanie nowoczesnych, zintegrowanych systemów informatycznych, które znacząco mogłyby zoptymalizować te procesy. Problem wpływa bezpośrednio na zadowolenie i satysfakcję klientów, a także na pozycję biur podróży na rynku turystycznym. Konsumenci zdecydowanie częściej sięgają po internetowe platformy rezerwacyjne, które zapewniają szybką, wygodną i intuicyjną obsługę, co prowadzi do utraty klientów przez tradycyjne biura podróży. Skuteczne rozwiązanie problemu umożliwi stworzenie intuicyjnego systemu w języku C#. Kluczowym elementem projektu będzie integracja z bazą danych oraz zapewnienie bezpieczeństwa przechowywanych informacji, a także zrozumienie potrzeb konsumenta by opracować łatwy w obsłudze interfejs dla pracowników jak i klientów. Realizacja projektu nastąpi etapami, w pierwszej kolejności należy przeanalizować wymagania i zrozumieć specyfikę pracy biura podróży jak i potrzeb konsumentów. Następnie zaprojektowana zostanie architektura systemu, zawierająca moduły zarządzania ofertą i obsługi rezerwacji. Implementacja projektu nastąpi w języku C# oraz systemie Microsoft SQL do zarządzania bazą danych. Kolejny krok, jakim jest testowanie pozwoli wyeliminować ewentualne błędy. Na końcu nastąpi wdrożenie systemu. Wynikiem pracy będzie funkcjonalny system obsługi biura podróży zwiększający efektywność pracy i poprawiający doświadczenie klientów.

### 1.2 Wymagania funkcjonalne

#### 1. Autoryzacja i logowanie:

- Pracownik musi się zalogować z użyciem odpowiednich danych w celu korzystania z aplikacji

#### 2. Zarządzanie ofertami wycieczek:

- Edycja i modyfikacja aktualności ofert wycieczek jest możliwa tylko przez pracowników,
- Przeglądanie dostępnych ofert i rezerwacja jest możliwa przez klientów,
- System przechowuje informacje o nazwie, terminach, cenie i dostępnych wycieczkach.

### 3. Rezerwacje wycieczek:

- Możliwość dokonywania rezerwacji wycieczek jest możliwa przez klientów.

## 1.3 Wymagania niefunkcjonalne

### 1. Wydajność:

- Aplikacja ma obsłużyć odpowiednią ilość połączeń w tym samym czasie bez spadku odczuwalnej wydajności działania systemu,
- Maksymalny czas odpowiedzi interfejsu dla zapytań użytkowników to 2 sekundy.

### 2. Bezpieczeństwo:

- Tylko pracownicy mogą zarządzać ofertami.
- Nieautoryzowany użytkownik nie dokona zmian w systemie.

### 3. Użyteczność:

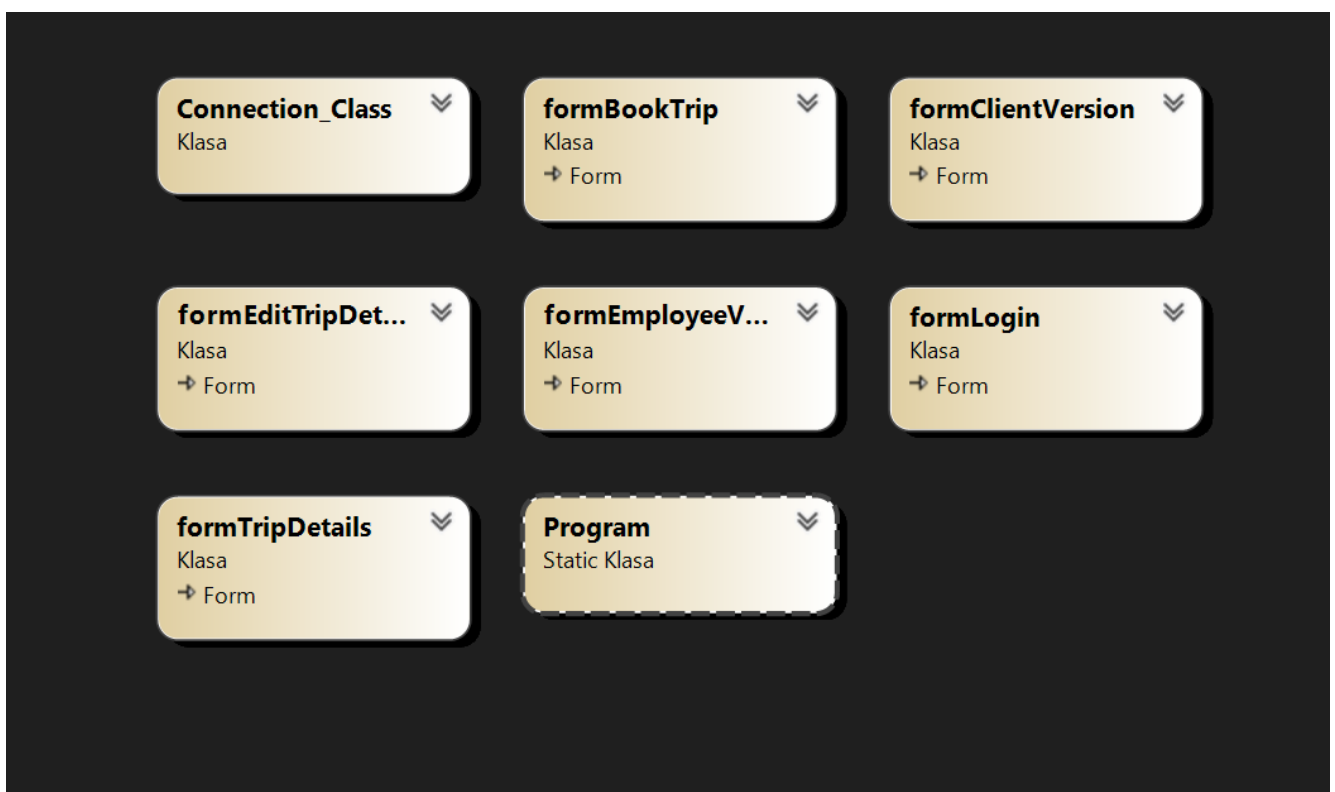
- Interfejs użytkownika jest intuicyjny i przejrzysty.
- Interfejs w aplikacji WinForms jest responsywny (np. dynamiczne skalowanie dla różnych rozdzielczości).

# Rozdział 2

## Struktura projektu

### 2.1 Diagram klas

Na rysunku poniżej został przedstawiony diagram klas wygenerowany w Visual Studio. Klasy zostały opisane w kolejności alfabetycznej.



Rysunek 2.1: Diagram klas

#### 2.1.1 Klasa Connection\_Class

Jest to publiczna klasa umożliwiająca obsługę połączenia z bazą danych. Zawiera metody:

- **GetConnection** - metoda zwracająca połączenie z bazą danych typu `SqlConnection` z wykorzystaniem właściwości `connectionString`,
- **ExecuteQuery** - metoda wykonująca przekazane jej zapytanie SQL,
- **ReturnQueryBoolValue** - metoda zamieniająca typ bit z bazy danych na wartość `true` lub `false`,
- **ReturnQueryValue** - metoda zwracająca pojedynczy wynik zapytania SQL,

- ReturnQueryValues - metoda zwracająca wynik z wielu kolumn tabeli,
- ReturnImage - metoda pobierająca z bazy danych obraz.

### 2.1.2 Klasa formBookTrip

- formBookTrip\_Load - metoda wykonująca się podczas ładowania formularza(przed jego pojawieniem się). Analogicznie dla kolejnych formularzy,
- btnConfirmBook\_Click - metoda przypisuje właściwość text kontrolek do zmiennych a następnie insertuje daną rezerwację do bazy danych,
- formBookTrip\_FormClosed - po zamknięciu formularza pojawia się jego poprzednik - formTripDetails, a obecny jest zamykany,
- txtBoxPhoneNumber\_KeyPress - walidacja wprowadzanych danych, możliwość wpisania jedynie cyfr,
- IsValidEmail - sprawdza poprawność wprowadzonego adresu e-mail.

### 2.1.3 Klasa formClientVersion

- btnDetailsTrip1\_Click - otwiera nowy formularz, w którym pokazane są szczegóły pierwszej wycieczki. Analogicznie dla dwóch kolejnych przycisków,
- formClientVersion\_FormClosed - metoda zamyka całą aplikację,
- btnGoToEmployeeVersion\_Click - metoda otwiera formularz formLogin, mający na celu weryfikację czy użytkownik jest autoryzowany poprzez logowanie.

### 2.1.4 Klasa formEditTripDetails

- formEditTripDetails\_FormClosed - metoda zamyka bieżący formularz i otwiera poprzedni - formEmployeeVersion,
- btnBookTrip\_Click - właściwości value, text, checked z kontrolek w formularzu zostają przypisane do zmiennych a następnie po pozytywnej walidacji aktualizują informacje odpowiedniej wycieczki,
- ConvertImageToByteArray - metoda konwertująca zdjęcie do bajtowej tablicy mająca na celu późniejsze zapisanie w bazie danych obrazu,
- boolToBit - metoda zamieniająca wartość true lub false na odpowiednik w postaci typu bit,
- btnChangePhoto\_Click - pozwala na wybranie z dysku komputera zdjęcia a następnie wyświetla je w kontrolce pictureBox1.
- txtBoxPricePerPerson\_KeyPress - walidacja wprowadzonych danych, przyjmuje jedynie cyfry.

### 2.1.5 Klasa formEmployeeVersion

- formEmployeeVersion\_FormClosed - metoda zamyka całą aplikację.
- btnGoToClientVersion\_Click - metoda otwierająca poprzedni formularz (formClientVersion) zamykająca obecny,
- btnEditTrip1\_Click - metoda uruchamiająca formularz formEditTripDetails, w którym można edytować szczegóły wycieczki, bieżący formularz zostaje zamknięty. Analogicznie dla dwóch kolejnych przycisków.



### 2.1.6 Klasa formLogin

- btnLogin\_Click - metoda weryfikująca dane użytkownika z tabeli pracownik, po pozytywnej autoryzacji uruchamia się formularz formEmployeeVersion,
- formLogin\_FormClosed - metoda zamykająca obecny formularz, a w jego miejsce pokazująca formClientVersion.

### 2.1.7 Klasa formTripDetails

- formTripDetails\_FormClosed - metoda zamykająca bieżący formularz, uruchamiająca formClientVersion,
- Truncate - metoda operująca na przekazanym ciągu znaków - odrzuca wszystko powyżej 10 znaków,
- btnBookTrip\_Click - metoda zamykająca bieżący formularz, uruchamiająca formBookTrip.

### 2.1.8 Klasa Program

Podczas włączania aplikacji metoda inicjalizuje jej wstępną konfigurację oraz uruchamia formularz formClientVersion.

## 2.2 Opis techniczny

Aplikacja "System zarządzania biurem podróży" została napisana w języku C# v12.0 z wykorzystaniem środowiska .NET 8.0 i technologii WinForms. Do pracy z danymi wykorzystano bazę SQL, a do zarządzania jej treścią SQL Server Management Studio v20.2

## 2.3 Minimalne wymagania sprzętowe

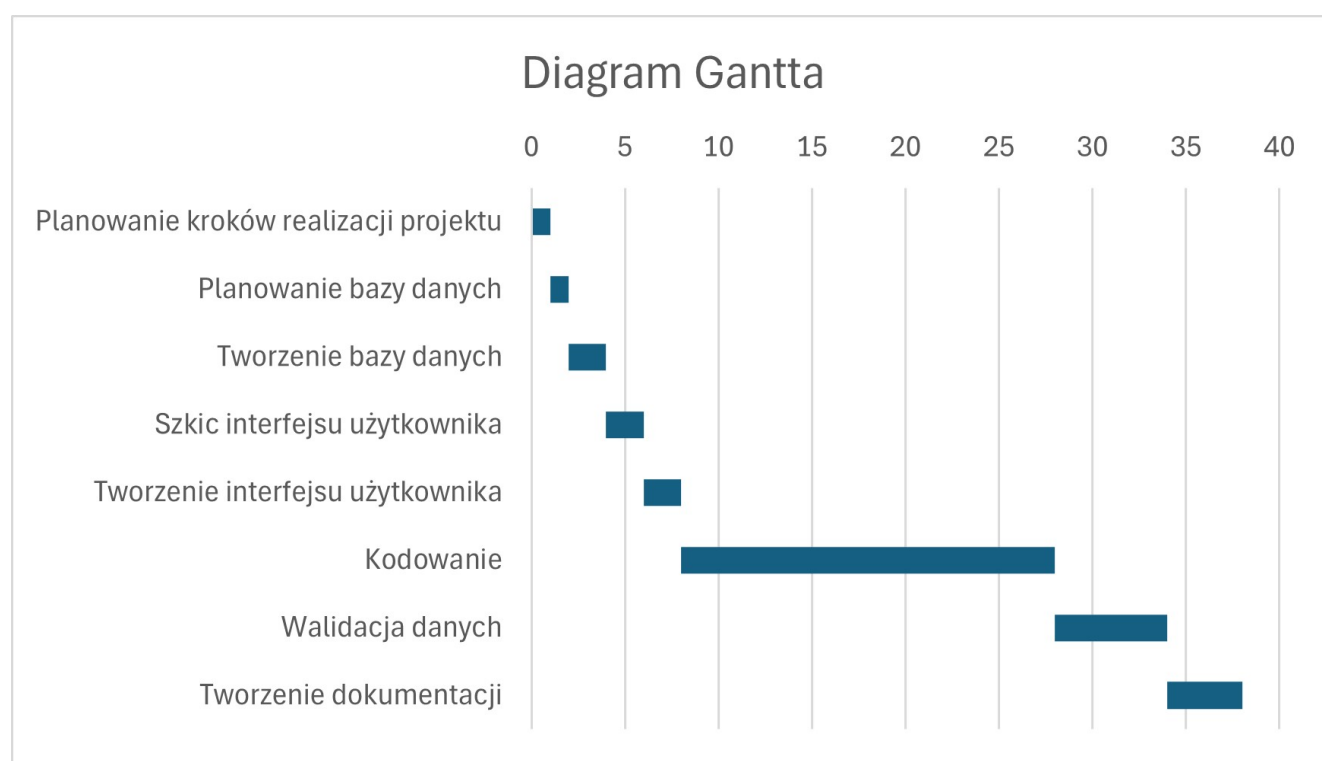
- Procesor: Intel Core i3 lub równoważny
- Pamięć RAM: 4 GB
- Przestrzeń dyskowa: 100 MB
- System operacyjny: Windows 11 64-bit

## Rozdział 3

# Harmonogram realizacji projektu

### 3.1 Diagram Gantta

Poniżej został przedstawiony diagram Gantta, pokazujący postęp tworzenia aplikacji. Zgodnie z przedstawionymi informacjami najdłuższym procesem było kodowanie aplikacji.



Rysunek 3.1: Diagram Gantta

### 3.2 Repozytorium

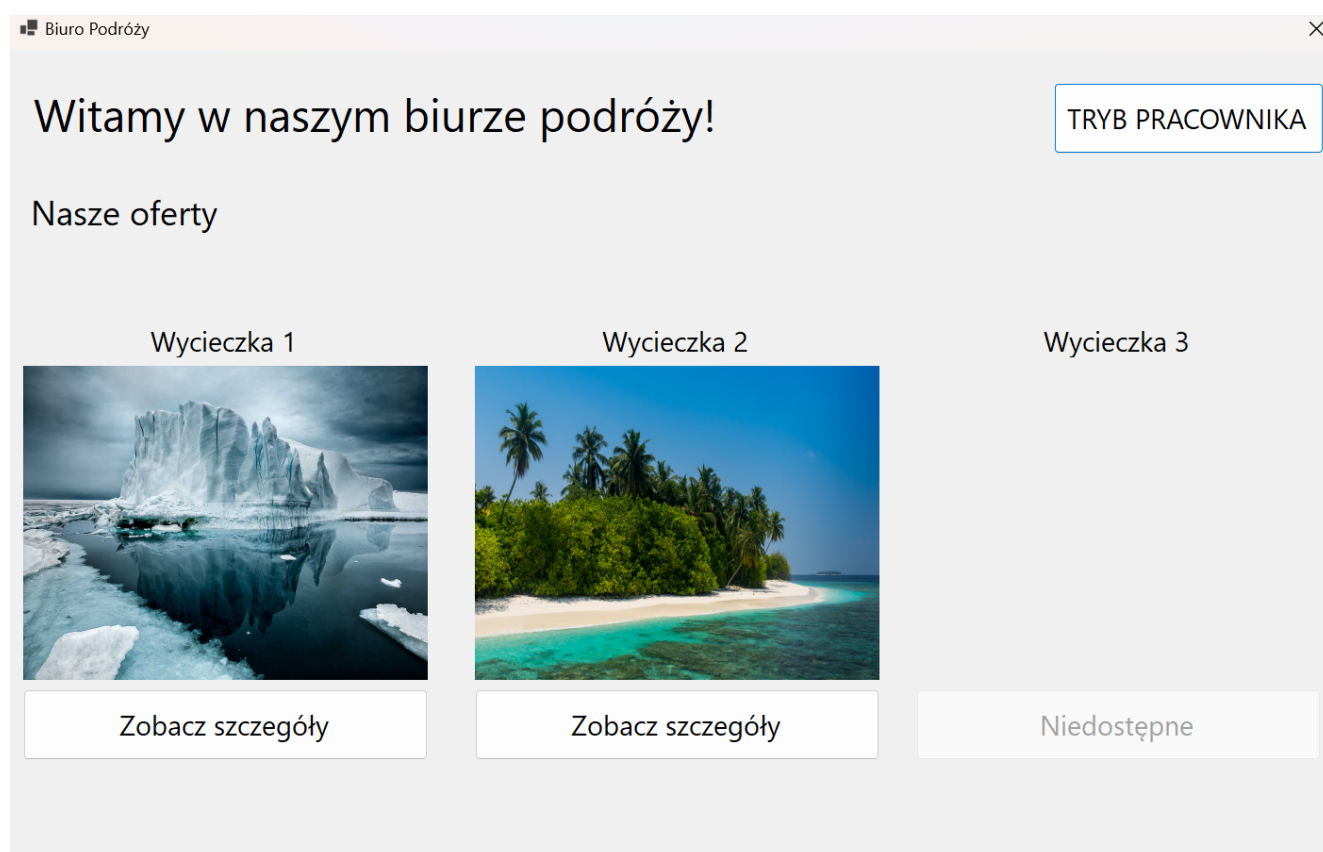
Podczas tworzenia projektu używano systemu kontroli wersji GIT. Wszystkie pliki znajdują się w repozytorium na GitHub pod adresem: [https://github.com/ZCuprys/System\\_obsługi\\_biura\\_podrozy](https://github.com/ZCuprys/System_obsługi_biura_podrozy)

# Rozdział 4

## Prezentacja warstwy użytkowej projektu

### 4.1 Ekran główny

Po uruchomieniu aplikacji pokazuje się widok panelu klienta (ekran główny aplikacji).



Rysunek 4.1: Ekran główny

Głównym elementem są widoczne aktualne oferty biura podróży. Do każdej aktywnej wycieczki przypisane jest zdjęcie. Pod każdą wycieczką dzięki kliknięciu przycisku "Zobacz szczegóły" otwiera nam się formularz pokazujący szczegóły wycieczki. Jeśli dana wycieczka jest nieaktywna, nie mamy możliwości podglądu szczegółów. W prawym górnym rogu znajduje się przycisk, dzięki któremu możemy przełączyć się do panelu pracownika poprzez panel logowania.

## 4.2 Podgląd wycieczki

Po kliknięciu przycisku Zobacz szczegóły otwiera się formularz z informacjami na temat danego wyjazdu tj. data wyjazdu, data przyjazdu, standard pokoju, wliczone posiłki i informacje dodatkowe.

Biuro Podróży

×

### Szczegóły wycieczki

Data wyjazdu

15.06.2025

Data przyjazdu

25.06.2025

Standard pokoju

Deluxe

Wliczone posiłki


tak

Dodatkowe informacje

WiFi: tak; Klimatyzacja: tak;  
Atrakcje dla dzieci: tak;

Rezerwuj

### Wycieczka 1



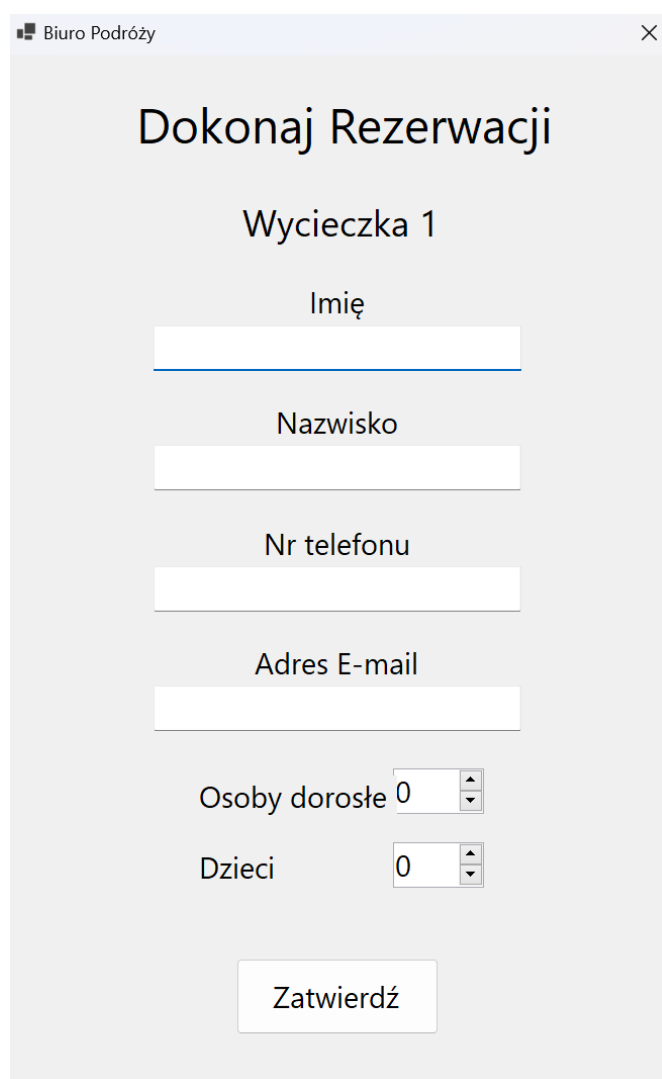
Cena za osobę: 2500,00 PLN

Rysunek 4.2: Szczegóły wycieczki

Po prawej stronie widoczne jest zdjęcie i nazwa wycieczki, a także jej cena za osobę poniżej. Pod informacjami na temat wycieczki znajduje się przycisk Rezerwuj przekierowujący klienta do formularza z podsumowaniem rezerwacji. W prawym górnym rogu znajduje się krzyżyk zamykający formularz ze szczegółami wycieczki. Nie zamyka on całej aplikacji.

## 4.3 Rezerwacja

Po kliknięciu przycisku Rezerwuj otwiera się formularz finalizujący rezerwację.



Biuro Podróży

### Dokonaj Rezerwacji

Wycieczka 1

Imię

Nazwisko

Nr telefonu

Adres E-mail

Osoby dorosłe 0

Dzieci 0

Zatwierdź

Rysunek 4.3: Dokonywanie rezerwacji

Umożliwia on uzupełnienie danych potrzebnych do dokonania rezerwacji przez klienta. Formularz został poddany walidacji, więc aby go zatwierdzić dane muszą zostać uzupełnione poprawnie.

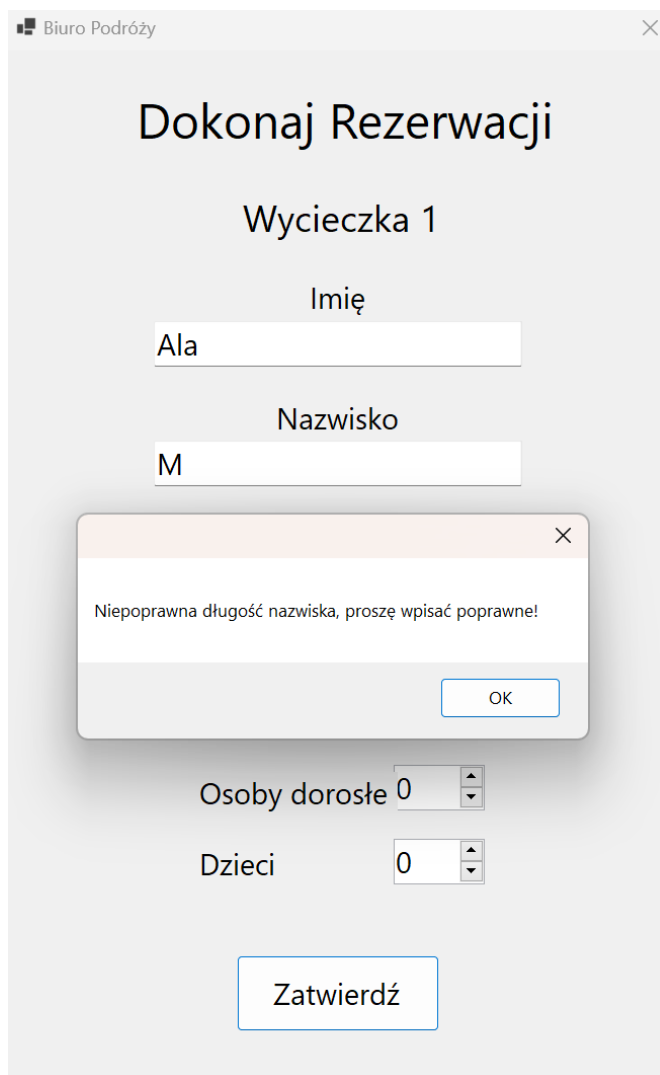
Jeśli nie wprowadzimy poprawnie imienia, pojawi się komunikat z informacją:

The screenshot shows a web application window titled "Biuro Podróży" with a close button. The main heading is "Dokonaj Rezerwacji" (Make a Reservation), followed by "Wycieczka 1" (Excursion 1). There is a label "Imię" (Name) above a text input field containing "AI". A modal error message box is displayed over the form, stating "Niepoprawna długość imienia, proszę wpisać poprawne!" (Incorrect name length, please enter a correct one!) with an "OK" button. Below the name field is an "Adres E-mail" label and an empty text input field. At the bottom, there are two spinners: "Osoby dorosłe" (Adults) and "Dzieci" (Children), both set to 0. A "Zatwierdź" (Confirm) button is at the very bottom.

Rysunek 4.4: Komunikat z błędnym imieniem

Po zatwierdzeniu komunikatu formularz wyczyści wprowadzone dane w polu Imię i umożliwi wpisanie ich ponownie. Aby wprowadzone imię było poprawne, musi mieć powyżej 2 znaków, lecz poniżej 50.

Jeśli klient wprowadzi imię poprawnie, musi następnie uzupełnić nazwisko. Jeśli nazwisko jest niepoprawne pojawi się komunikat z informacją:



The screenshot shows a web application window titled "Biuro Podróży". The main heading is "Dokonaj Rezerwacji" (Make a Reservation). Below it is the sub-heading "Wycieczka 1" (Excursion 1). The form contains two text input fields: "Imię" (First Name) with the value "Ala" and "Nazwisko" (Last Name) with the value "M". A modal dialog box is displayed in the center, indicating an error: "Niepoprawna długość nazwiska, proszę wpisać poprawne!" (Incorrect last name length, please enter a correct one!). The dialog has an "OK" button. Below the error message, there are two spinners: "Osoby dorosłe" (Adults) set to 0 and "Dzieci" (Children) set to 0. At the bottom of the form is a "Zatwierdź" (Confirm) button.

Rysunek 4.5: Komunikat z błędnym nazwiskiem

Po zatwierdzeniu komunikatu formularz wyczyści wprowadzone dane w polu Nazwisko i umożliwi wpisanie ich ponownie. Aby wprowadzone nazwisko było poprawne, musi mieć powyżej 2 znaków, lecz poniżej 50.

Jeśli klient wprowadzi nazwisko poprawnie, musi następnie uzupełnić numer telefonu. Jeśli numer telefonu jest niepoprawny pojawi się komunikat z informacją:

The screenshot shows a web application window titled "Biuro Podróży". The main heading is "Dokonaj Rezerwacji". Below it is the sub-heading "Wycieczka 1". The form contains three input fields: "Imię" with the value "Ala", "Nazwisko" with the value "Kot", and "Nr telefonu" with the value "123". Below the phone number field is a "Dzieci" label followed by a dropdown menu showing "0". At the bottom of the form is a "Zatwierdź" button. A modal dialog box is open in the center, displaying the error message: "Numer telefonu nie jest poprawny, proszę wprowadzić ponownie!". The dialog has an "OK" button.

Rysunek 4.6: Komunikat z błędnym numerem telefonu

Po zatwierdzeniu komunikatu formularz wyczyści wprowadzone dane w polu Nr telefonu i umożliwi wpisanie ich ponownie. Aby wprowadzony numer telefonu był poprawny, musi mieć przynajmniej 9 cyfr lecz poniżej 15. Pole zostało zwalidowane tak, aby klient mógł wpisać tylko cyfry.



Jeśli klient wprowadzi numer telefonu poprawnie, musi następnie uzupełnić adres e-mail. Jeśli adres e-mail jest niepoprawny pojawi się komunikat z informacją:

The screenshot shows a window titled "Biuro Podróży" with a close button. The main heading is "Dokonaj Rezerwacji" (Make a Reservation). Below it is "Wycieczka 1" (Excursion 1). There is a label "Imię" (Name) above a text input field containing "Ala". A modal dialog box is centered on the screen with a close button. The dialog has a light orange header and contains the text "Niepoprawna długość adresu email, spróbuj jeszcze raz!" (Incorrect email address length, try again!). There is an "OK" button at the bottom right of the dialog. Below the dialog, there is a label "Adres E-mail" above a text input field containing "ala". Below that are two spinners: "Osoby dorosłe" (Adults) with a value of 0, and "Dzieci" (Children) with a value of 0. At the bottom of the form is a "Zatwierdź" (Confirm) button.

Rysunek 4.7: Komunikat z błędną długością adresu e-mail

Po zatwierdzeniu komunikatu formularz wyczyści wprowadzone dane w polu Adres E-mail i umożliwi wpisanie ich ponownie. Aby wprowadzony adres e-mail był poprawny, musi mieć powyżej 3 znaków lecz nie więcej niż 100.

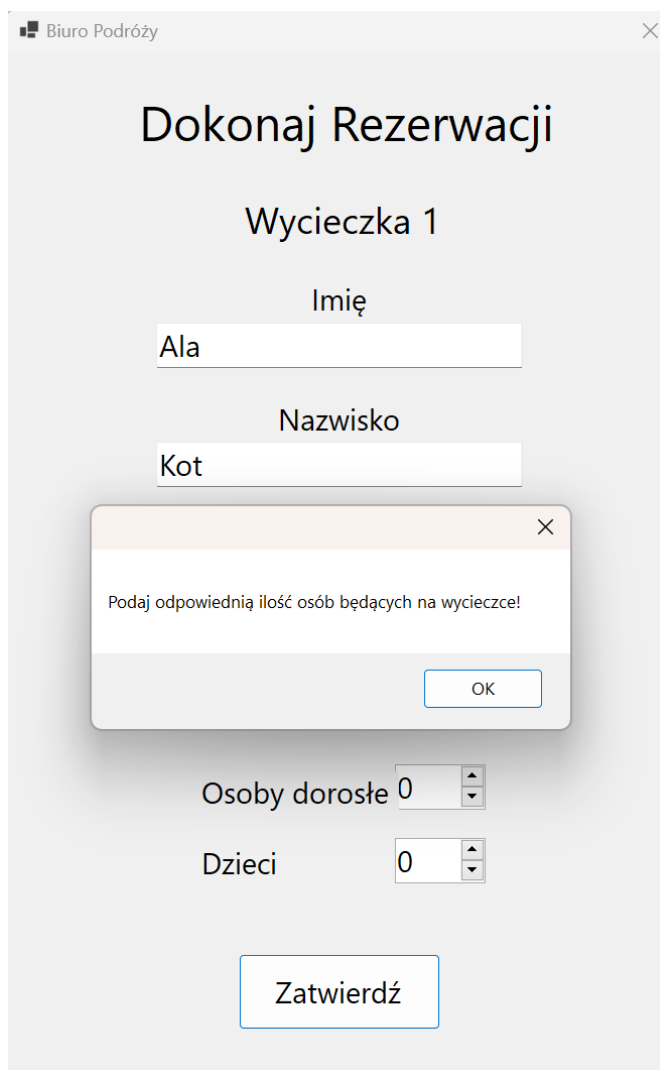
Dodatkowo adres e-mail powinien zostać wpisany według ogólnie przyjętego wzoru walidacji. W przeciwnym wypadku wyświetli się komunikat z błędem:

The screenshot shows a window titled "Biuro Podróży" with a close button. The main heading is "Dokonaj Rezerwacji". Below it is the sub-heading "Wycieczka 1". The form contains four input fields with labels above them: "Imię" (containing "Ala"), "Nazwisko" (containing "Kot"), "Nr telefonu" (containing "123456789"), and "Adres E-mail" (containing "alamakota"). At the bottom of the form is a "Zatwierdź" button. An error dialog box is open over the email field, with a title bar containing a close button and the text "Adres email nie jest poprawny, spróbuj ponownie!". The dialog has an "OK" button.

Rysunek 4.8: Komunikat z błędnym adresem e-mail

Po zatwierdzeniu komunikatu formularz wyczyści wprowadzone dane w polu Adres E-mail i umożliwi wpisanie ich ponownie.

Jeśli klient wprowadzi adres e-mail poprawnie, musi następnie uzupełnić informacje na temat liczby osób biorących udział w wycieczce. Jeśli liczba osób nie zostanie zaznaczona i wyniesie 0, pojawi się komunikat z informacją:



Biuro Podróży

## Dokonaj Rezerwacji

### Wycieczka 1

Imię

Ala

Nazwisko

Kot

Podaj odpowiednią ilość osób będących na wycieczce!

OK

Osoby dorosłe 0

Dzieci 0

Zatwierdź

Rysunek 4.9: Komunikat z błędną liczbą osób

Jeśli klient zaznaczy, że w wycieczce biorą udział tylko dzieci, również wyświetli się komunikat z informacją:

Biuro Podróży

## Dokonaj Rezerwacji

### Wycieczka 1

Imię

Ala

Dzieci nie mogą podróżować same!

OK

Adres E-mail

alama@kota

Osoby dorosłe 0

Dzieci 1

Zatwierdź

Rysunek 4.10: Komunikat z informacją dotyczącą dzieci

Po zatwierdzeniu komunikatu formularz nie wyczyści wprowadzonej liczby dzieci, umożliwi dodanie osób dorosłych.

Jeśli klient wprowadzi wszystkie dane poprawnie, pojawi się komunikat potwierdzający rezerwację wraz z jej kwotą:

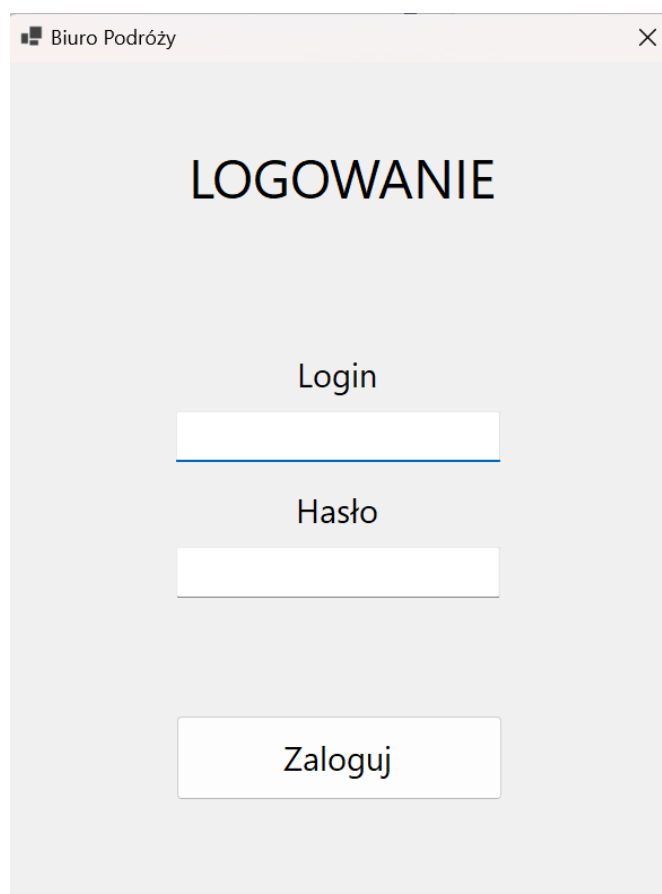
The screenshot shows a web application window titled "Biuro Podróży". The main heading is "Dokonaj Rezerwacji" (Make a Reservation), followed by "Wycieczka 1" (Excursion 1). A modal dialog box is open in the center, titled "Imie" (Name). The dialog contains the text "Zapisano rezerwację! Finalna kwota wycieczki: 2500,00 PLN" (Reservation saved! Final excursion amount: 2500.00 PLN) and an "OK" button. In the background, the booking form is visible with the following fields: "Nr telefonu" (Phone number) with the value "123456789", "Adres E-mail" (Email address) with the value "alama@kota", "Osoby dorosłe" (Adults) with a value of "1", and "Dzieci" (Children) with a value of "0". A "Zatwierdź" (Confirm) button is at the bottom of the form.

Rysunek 4.11: Pomyślne dokonanie rezerwacji

Następnie klient zostanie przekierowany z powrotem do formularza Szczegóły wycieczki.

## 4.4 Logowanie pracownika

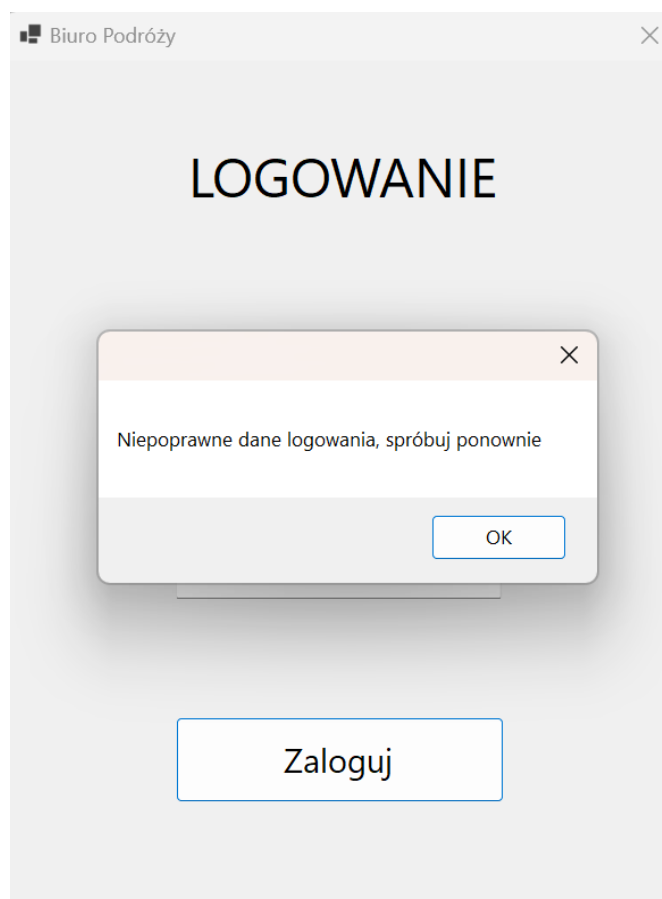
Po kliknięciu przycisku Tryb pracownika znajdującego się na stronie głównej aplikacji, nastąpi przeniesienie do formularza umożliwiającego logowanie do konta pracownika biura podróży:



The image shows a web application window titled "Biuro Podróży" with a close button (X) in the top right corner. The main content area has a light gray background and is titled "LOGOWANIE" in large, bold, black capital letters. Below the title, there are two input fields. The first field is labeled "Login" and has a blue underline. The second field is labeled "Hasło" (Password). Below these fields is a button labeled "Zaloguj" (Log in).

Rysunek 4.12: Formularz logowania

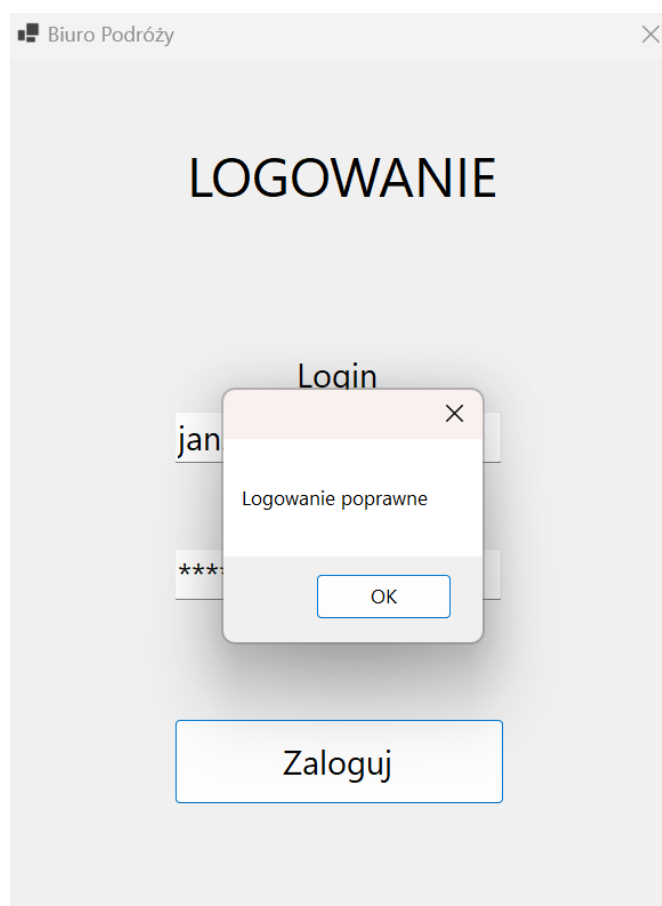
Pracownik musi wprowadzić dane logowania poprawnie. W przypadku podania błędnych danych logowania zostanie wyświetlony komunikat:



Rysunek 4.13: Komunikat z informacją o błędnych danych logowania

Wprowadzając hasło w polu, nie jest ono jawne, aby nie można było go odczytać.

Jeśli dane zostaną wprowadzone poprawnie, wyświetli się komunikat z informacją o zalogowaniu.

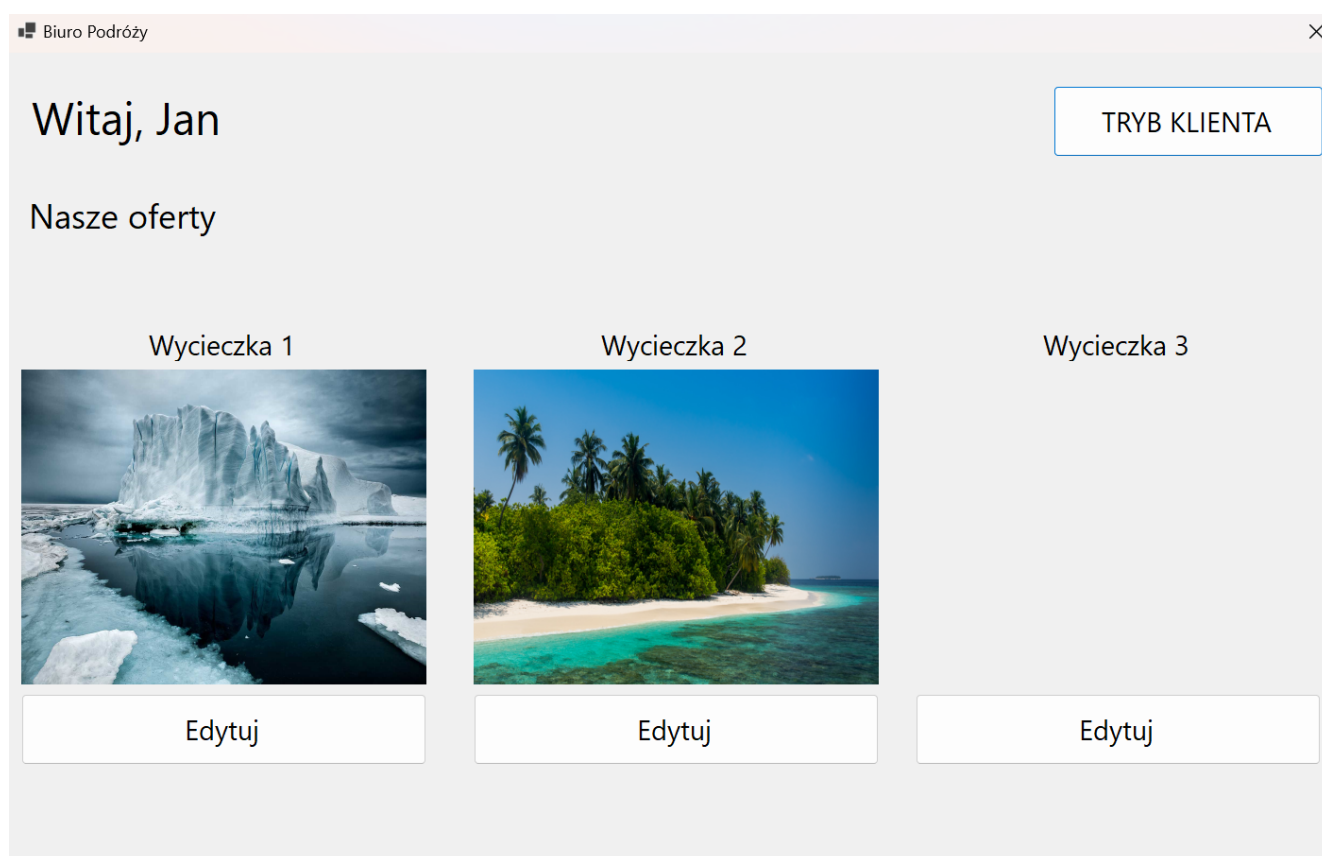


Rysunek 4.14: Informacja o poprawnym logowaniu



## 4.5 Tryb pracownika

Po poprawnym logowaniu pokazuje się formularz z trybem pracownika:




Biuro Podróży

Witaj, Jan

TRYB KLIENTA


Nasze oferty

Wycieczka 1



Edytuj

Wycieczka 2



Edytuj

Wycieczka 3

Edytuj

Rysunek 4.15: Tryb pracownika

W lewym górnym rogu formularza wyświetla się przywitanie zalogowanego pracownika zawierające jego imię. W prawym górnym rogu znajduje się przycisk Tryb klienta umożliwiający powrót do trybu klienta (Ekranu głównego). Poniżej znajdują się wycieczki oferowane przez biuro podróży z tą różnicą, że zamiast przycisku Zobacz szczegóły widnieje opcja edycji.

Po kliknięciu przycisku Edytuj pracownik zostanie przekierowany do formularza gdzie może dokonać edycji informacji o wybranej wycieczce:

## Edytuj szczegóły wycieczki

Data wyjazdu

niedziela , 15 czerwca 2 ▾

Data przyjazdu

środa , 25 czerwca 2 ▾

Standard pokoju

Deluxe ▾

Dodatkowe informacje


☒ Wyżywienie w cenie

☒ WiFi

☒ Klimatyzacja

☒ Atrakcje dla dzieci

### Wycieczka 1



Zmień zdjęcie

Cena za osobę: 2500,00

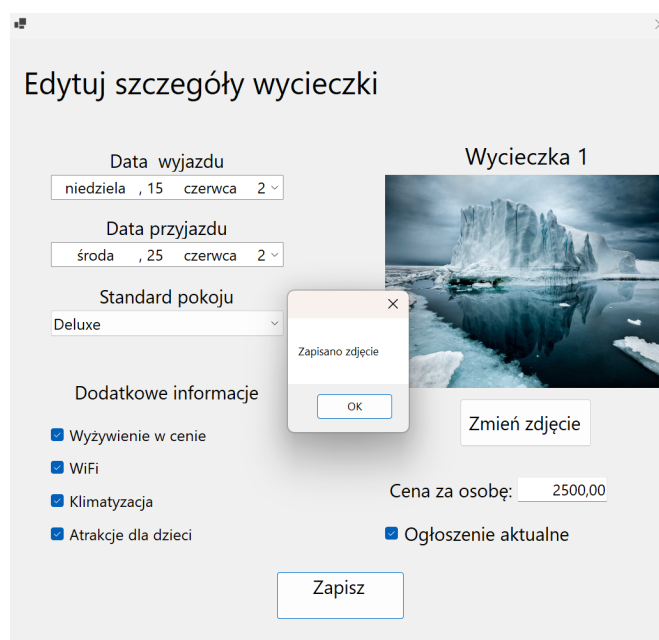
☒ Ogłoszenie aktualne

Zapisz

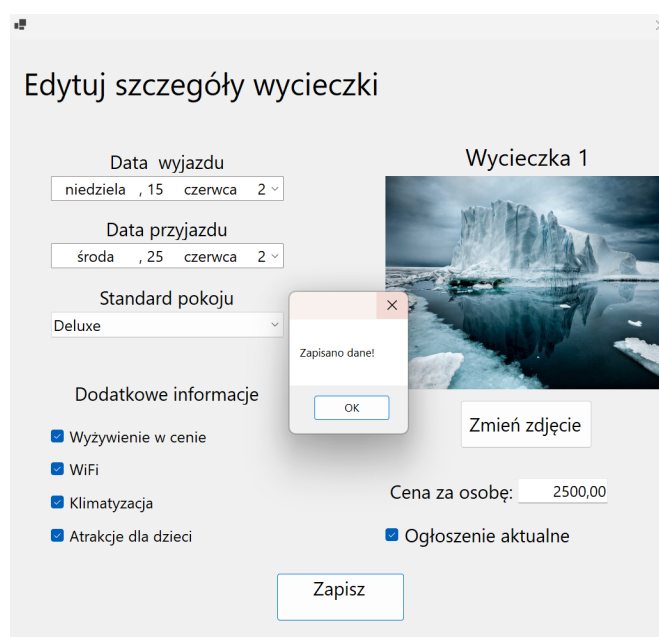
Rysunek 4.16: Edytowanie wycieczki przez pracownika

Pracownik może dokonać edycji dat, standardu pokoju, dodatkowych informacji, ceny, zmienić zdjęcie oraz zdecydować czy wycieczka jest aktualna.

Po dokonaniu zmian i kliknięciu przycisku Zapisz wyświetlą się następujące komunikaty:



Rysunek 4.17: Komunikat o zapisie zdjęcia



Rysunek 4.18: Komunikat o zapisie danych

## **Rozdział 5**

### **Podsumowanie**

W ramach realizacji projektu "System obsługi biura podróży" utworzono aplikację desktopową, która umożliwia zarządzanie ofertami wycieczek z poziomu panelu pracownika, przeglądanie i rezerwacje wycieczek przez klientów. System spełnia podstawowe wymagania funkcjonalne z możliwością jego dalszego rozwoju i dostosowywania go do potrzeb użytkowników oraz branży turystycznej.

# Bibliografia

- [1] Mark J. Price *C# 11 i .NET 7 dla programistów aplikacji wieloplatformowych. Twórz aplikacje, witryny WWW oraz serwisy sieciowe za pomocą ASP.NET Core 7, Blazor i EF Core 7. Wydanie VII*, Helion S.A., Gliwice 2022.

# Spis rysunków

2.1	Diagram klas . . . . .	7
3.1	Diagram Gantta . . . . .	10
4.1	Ekran główny . . . . .	11
4.2	Szczegóły wycieczki . . . . .	12
4.3	Dokonywanie rezerwacji . . . . .	13
4.4	Komunikat z błędnym imieniem . . . . .	14
4.5	Komunikat z błędnym nazwiskiem . . . . .	15
4.6	Komunikat z błędnym numerem telefonu . . . . .	16
4.7	Komunikat z błędną długością adresu e-mail . . . . .	17
4.8	Komunikat z błędnym adresem e-mail . . . . .	18
4.9	Komunikat z błędną liczbą osób . . . . .	19
4.10	Komunikat z informacją dotyczącą dzieci . . . . .	20
4.11	Pomyślne dokonanie rezerwacji . . . . .	21
4.12	Formularz logowania . . . . .	22
4.13	Komunikat z informacją o błędnych danych logowania . . . . .	23
4.14	Informacja o poprawnym logowaniu . . . . .	24
4.15	Tryb pracownika . . . . .	25
4.16	Edytowanie wycieczki przez pracownika . . . . .	26
4.17	Komunikat o zapisie zdjęcia . . . . .	27
4.18	Komunikat o zapisie danych . . . . .	27