



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

KOLEGIUM INFORMATYKI STOSOWANEJ

Kierunek: INFORMATYKA

Specjalność: Inżynieria danych

Patrycja Starzyk
Nr albumu studenta w70955

System zarządzania dla korepetytora

Prowadzący: mgr inż. Ewa Żesławska

Praca projektowa programowanie obiektowe C#

Rzeszów 2025

Spis treści

Wstęp	4
1 Opis założeń projektu	5
1.1 Cele projektu	5
1.2 Wymagania funkcjonalne i нефункционалне	5
2 Opis struktury projektu	7
2.1 Narzędzia użyte w projekcie:	7
2.2 Minimalne wymagania sprzętowe:	7
2.3 Diagram klas	8
3 Harmonogram realizacji projektu	10
3.1 Diagram Ganta	10
3.2 Repozytorium	10
4 Prezentacja warstwy użytkowej projektu	11
4.1 Główny interfejs użytkownika	11
4.2 Zarządzanie uczniami	12
5 Podsumowanie	20
Spis rysunków	21
Spis tablic	23

Wstęp

Na co dzień wielu korepetytorów zmaga się z organizowaniem i zarządzaniem różnorodnymi danymi związanymi z uczniami, ocenami, tematami zajęć oraz terminami. Bez odpowiedniego narzędzia, te informacje mogą łatwo stać się chaotyczne i trudne do śledzenia. Niezbędne staje się więc stworzenie systemu, który pozwala na sprawne zarządzanie wszystkimi aspektami pracy korepetytora w jednym miejscu. System ten umożliwi korepetytorom łatwe dodawanie uczniów, przypisywanie im ocen, zapisywanie tematów zajęć oraz ustalanie terminów. Dzięki przechowywaniu danych w formacie JSON, informacje są bezpieczne i łatwe do edytowania. Dodatkowo, system pozwala na generowanie raportów z postępów uczniów, co umożliwi bieżące monitorowanie efektywności nauczania. Dzięki tym funkcjom korepetytorzy mogą skupić się na nauczaniu, nie martwiąc się o organizacyjne detale.

Rozdział 1

Opis założeń projektu

1.1 Cele projektu

Celem projektu jest stworzenie systemu umożliwiającego korepetytorom łatwe zarządzanie uczniami, ocenami, tematami zajęć i terminami. System ma usprawnić organizację pracy, centralizując wszystkie dane w jednym narzędziu. Problemem jest trudność w zarządzaniu informacjami o uczniach, które są często rozproszone w różnych dokumentach, co prowadzi do chaosu i opóźnień. Brak zintegrowanego systemu do przechowywania tych danych utrudnia pracę korepetytorów. Problem jest ważny, ponieważ wpływa na efektywność pracy korepetytora oraz jakość nauczania. Dowodem na jego istnienie są liczne opinie korepetytorów, którzy zgłaszają trudności w organizacji pracy bez odpowiedniego narzędzia. Aby rozwiązać ten problem, zespół musi stworzyć aplikację, która pozwoli na łatwe dodawanie uczniów, ocen, tematów i terminów. System musi być intuicyjny, bezpieczny i łatwy do edytowania.

Realizacja projektu będzie przebiegała krok po kroku. Na początku opracowany zostanie plan systemu oraz jego funkcjonalności. Następnie zaprojektowana i zaimplementowana zostanie baza danych, która umożliwi przechowywanie danych o uczniach. Kolejnym etapem będzie stworzenie interfejsu użytkownika, umożliwiającego korepetytorowi łatwe zarządzanie uczniami, ocenami, tematami i terminami. Na końcu system będzie testowany i optymalizowany. Wynikiem prac będzie aplikacja komputerowa, która rozwiąże problem zarządzania danymi o uczniach i pomoże korepetytorom w organizacji ich pracy.

1.2 Wymagania funkcjonalne i нефункционалне

Wymagania funkcjonalne

1. Zarządzanie uczniami

- System powinien umożliwiać dodawanie, edytowanie i usuwanie uczniów.
- Dane uczniów powinny być przechowywane w pliku JSON i ładowane przy uruchomieniu aplikacji.

2. Zarządzanie zajęciami i ocenami

- System powinien umożliwiać przypisywanie uczniom ocen, tematów zajęć oraz terminów lekcji.
- Wprowadzone dane powinny być zapisywane do pliku JSON i dostępne po ponownym uruchomieniu aplikacji.

Wymagania нефункционалне

1. Wydajność

- Aplikacja powinna wczytywać i zapisywać dane z/do pliku JSON w czasie nie dłuższym niż 15 sekund.

2. Użyteczność

- Interfejs graficzny powinien być intuicyjny i umożliwiać łatwe zarządzanie danymi uczniów.
- Aplikacja powinna zapewniać komunikaty informacyjne i ostrzegawcze.

Rozdział 2

Opis struktury projektu

Aplikacja została napisana w języku C# z wykorzystaniem Windows Forms do stworzenia interfejsu graficznego. Do zarządzania danymi wykorzystywany jest format JSON, obsługiwany za pomocą biblioteki Newtonsoft.Json.

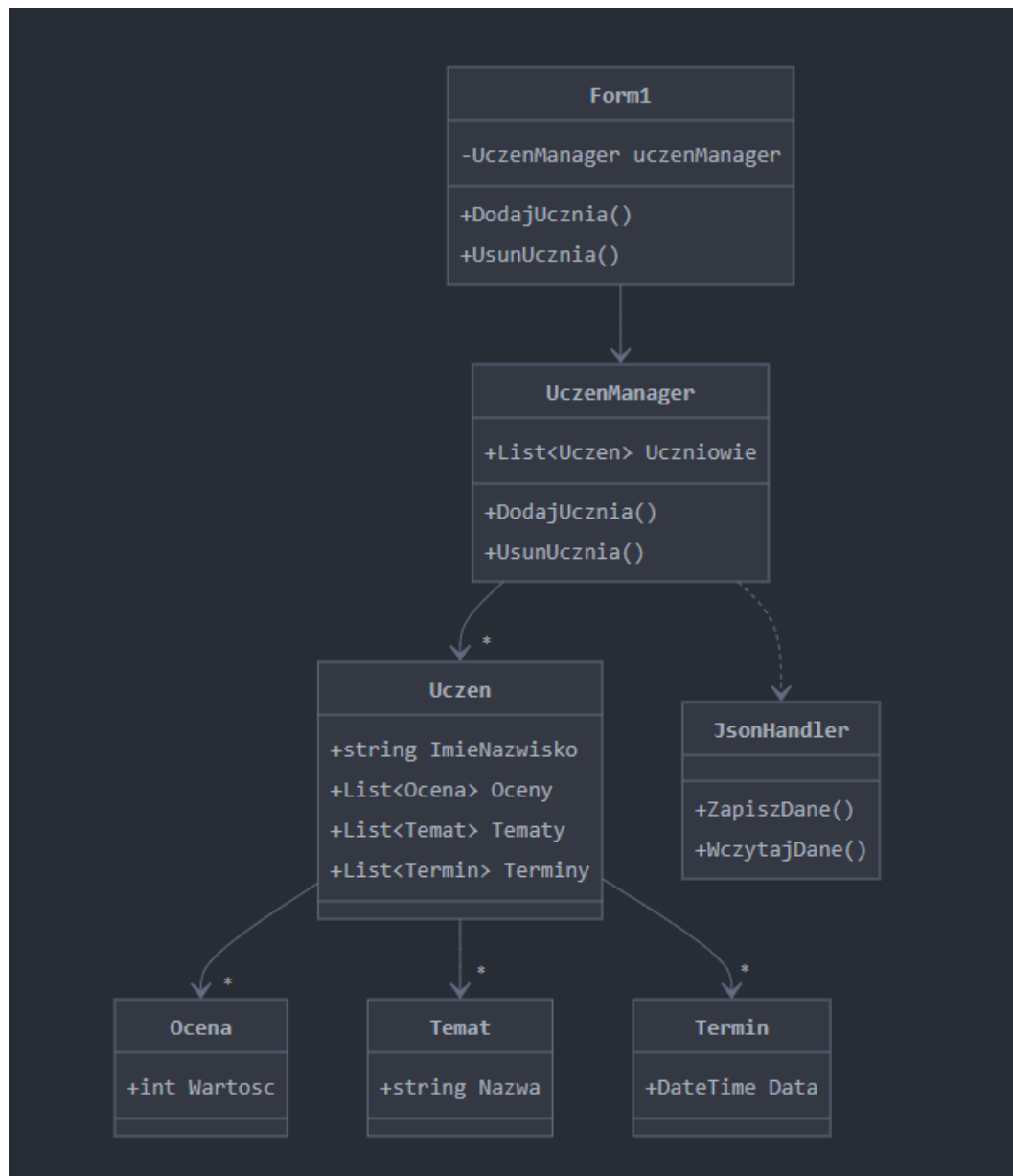
2.1 Narzędzia użyte w projekcie:

- Środowisko programistyczne: Visual Studio
- Język programowania: C#
- Technologia GUI: Windows Forms
- Biblioteki: Newtonsoft.Json (obsługa plików JSON)

2.2 Minimalne wymagania sprzętowe:

- Procesor: Intel Core i3 lub odpowiednik
- Pamięć RAM: 4 GB
- Dysk: 100 MB wolnej przestrzeni
- System operacyjny: Windows 10 lub nowszy

2.3 Diagram klas



Rysunek 2.1: Diagram klas

Diagram klas pokazujący strukturę aplikacji. Oto główne elementy i relacje:

1. Klasa Form1 - główna klasa formularza Windows Forms:
 - Zawiera instancję UczenManager
 - Implementuje wszystkie metody obsługi zdarzeń (eventy przycisków)
2. Klasa UczenManager - zarządza listą uczniów:
 - Przechowuje listę uczniów
 - Udostępnia metody do zarządzania uczniami
3. Klasa Uczen - reprezentuje pojedynczego ucznia:
 - Przechowuje dane ucznia (imię i nazwisko)

- Zawiera listy ocen, tematów i terminów

4. Klasy pomocnicze:

- Ocena - przechowuje wartość oceny
- Temat - przechowuje nazwę tematu
- Termin - przechowuje datę terminu

5. Klasa statyczna JsonHandler - odpowiada za operacje na pliku JSON:

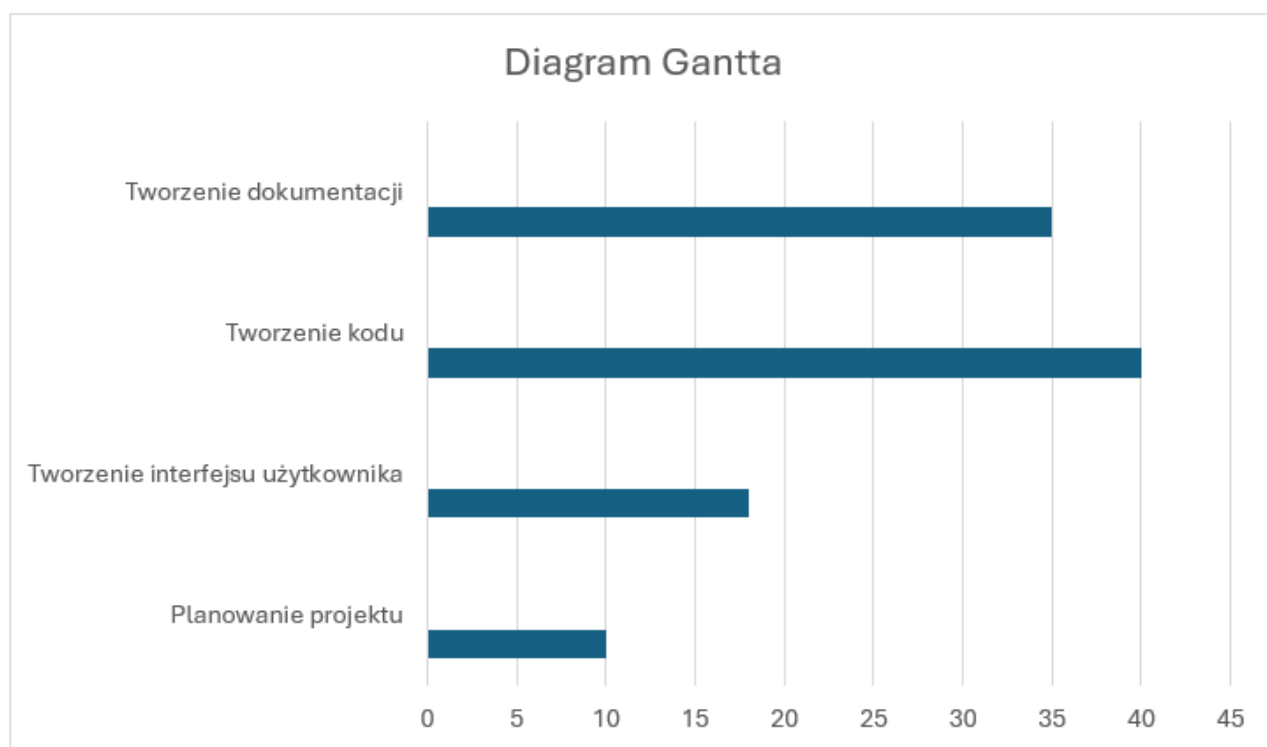
- Zawiera metody statyczne do zapisu i odczytu danych

Rozdział 3

Harmonogram realizacji projektu

3.1 Diagram Ganta

Diagram poniżej ilustruje rozkład czasowy poszczególnych zadań, z wyraźnym wskazaniem na etap kodowania, który był najdłuższy w całym projekcie.



Rysunek 3.1: Diagram Ganta

3.2 Repozytorium

Projekt został umieszczony w repozytorium GitHub, gdzie można znaleźć wszystkie pliki źródłowe aplikacji, w tym kod oraz dokumentację.

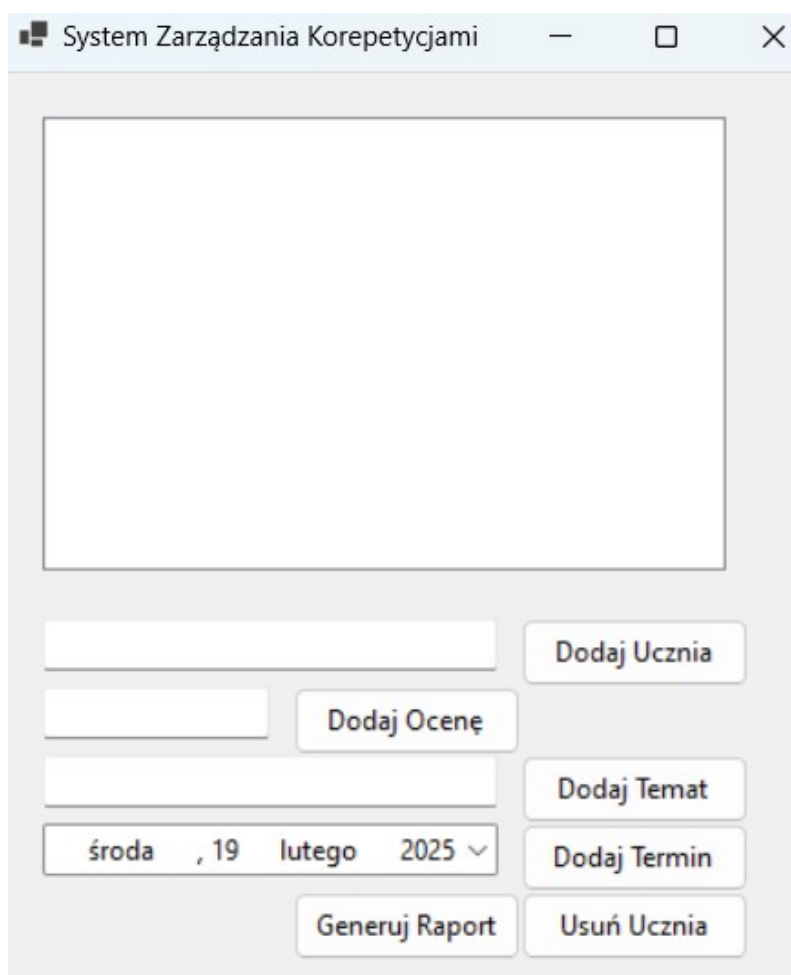
Link do repozytorium: https://github.com/patrycja-s/Programowanie-obiektowe-w70/tree/main/projekt_w70955

Rozdział 4

Prezentacja warstwy użytkowej projektu

4.1 Główny interfejs użytkownika

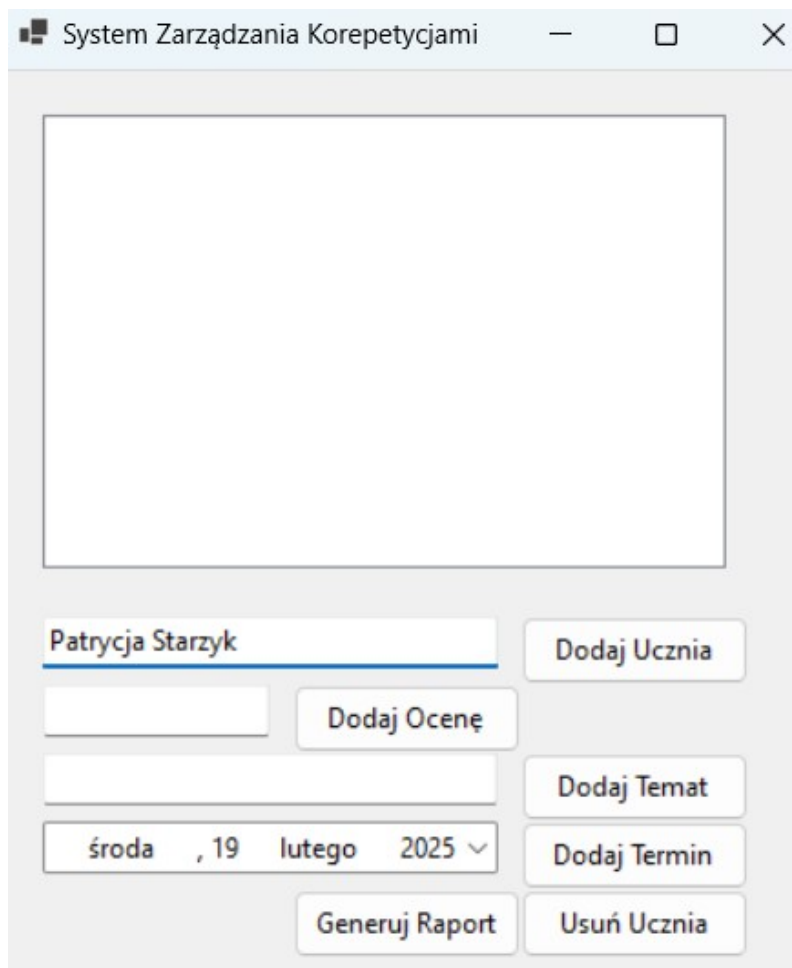
Główny interfejs aplikacji, na którym widoczne są dostępne opcje do wyboru przez użytkownika. Na tym ekranie użytkownik może wybrać odpowiednią opcję, taką jak dodanie ucznia, oceny, tematu lub terminu zajęć.



Rysunek 4.1: Główny interfejs

4.2 Zarządzanie uczniami

Wyświetlana część aplikacji umożliwia użytkownikowi edytowanie danych uczniów, dodawanie nowych uczniów, ocen, tematów oraz terminów zajęć. Użytkownik może również usuwać uczniów z listy, co pozwala na pełną kontrolę nad bazą danych uczniów. Funkcje dostępne w tej sekcji obejmują zarządzanie informacjami o uczniu, takie jak imię i nazwisko, oceny, tematy lekcji oraz terminy zajęć, zapewniając elastyczność w dostosowywaniu bazy danych do potrzeb korepetytora.



The screenshot shows a window titled "System Zarządzania Korepetycjami". Inside, there is a large empty rectangular box at the top. Below it, there is a form with several input fields and buttons. The first input field contains the text "Patrycja Starzyk" and is underlined. To its right is a button labeled "Dodaj Ucznia". Below the first input field is another empty input field, followed by a button labeled "Dodaj Ocenę". Below that is another empty input field, followed by a button labeled "Dodaj Temat". Below the third input field is a date selector showing "środa , 19 lutego 2025" with a dropdown arrow, followed by a button labeled "Dodaj Termin". At the bottom, there are two buttons: "Generuj Raport" and "Usuń Ucznia".

Rysunek 4.2: Wpisanie ucznia

System Zarządzania Korepetycjami

Rysunek 4.3: Dodanie ucznia

The screenshot shows a web application window titled "System Zarządzania Korepetycjami". Inside the window, there is a large text area at the top containing the name "Patrycja Starzyk". Below this text area, there are several input fields and buttons. On the left, there are three empty text input fields stacked vertically. To the right of the first input field is a button labeled "Dodaj Ucznia". To the right of the second input field is a button labeled "Dodaj Ocene". To the right of the third input field is a button labeled "Dodaj Temat". Below these three input fields is a date selector showing "środa , 19 lutego 2025" with a dropdown arrow. To the right of the date selector is a button labeled "Dodaj Termin". At the bottom of the form area, there are two buttons: "Generuj Raport" on the left and "Usuń Ucznia" on the right.

Rysunek 4.4: Uczeń dodany do listy

Aplikacja pozwala na dodanie nowego ucznia do listy, co umożliwia późniejsze zarządzanie jego danymi, takimi jak oceny, tematy zajęć i terminy spotkań. Aby dodać ucznia, użytkownik musi wpisać jego imię i nazwisko w odpowiednim polu tekstowym, a następnie kliknąć przycisk "Dodaj ucznia".

The screenshot shows a web application window titled "System Zarządzania Korepetycjami". Inside the window, there is a list of students, with "Patrycja Starzyk" selected and highlighted in blue. Below the list, there is a form to add a grade. The form includes a text input field for the grade, which currently contains the number "4". To the right of this field is a blue button labeled "Dodaj Ocenę". Above this button is a text input field for the subject, with a button "Dodaj Ucznia" to its right. Below the subject field is a date selector showing "środa , 19 lutego 2025" with a dropdown arrow, and a button "Dodaj Temat" to its right. At the bottom of the form are two buttons: "Generuj Raport" and "Usuń Ucznia".

Rysunek 4.5: Dodanie oceny

Aplikacja umożliwia przypisanie ocen do konkretnego ucznia, co pozwala na monitorowanie jego postępów. Aby dodać ocenę, użytkownik musi wybrać ucznia z listy oraz wpisać wartość oceny w odpowiednim polu tekstowym. Następnie, po kliknięciu przycisku "Dodaj ocenę", zostanie ona przypisana do ucznia i zapisana w systemie.

The screenshot shows a web application window titled "System Zarządzania Korepetycjami". Inside the window, there is a large text area at the top containing the name "Patrycja Starzyk". Below this, there are several input fields and buttons. The first row has an empty text input field followed by a button labeled "Dodaj Ucznia". The second row has another empty text input field followed by a button labeled "Dodaj Ocenę". The third row has a text input field containing "matematyka dyskretna" followed by a button labeled "Dodaj Temat". The fourth row has a date selection field showing "środa , 19 lutego 2025" with a dropdown arrow, followed by a button labeled "Dodaj Termin". At the bottom, there are two buttons: "Generuj Raport" and "Usuń Ucznia".

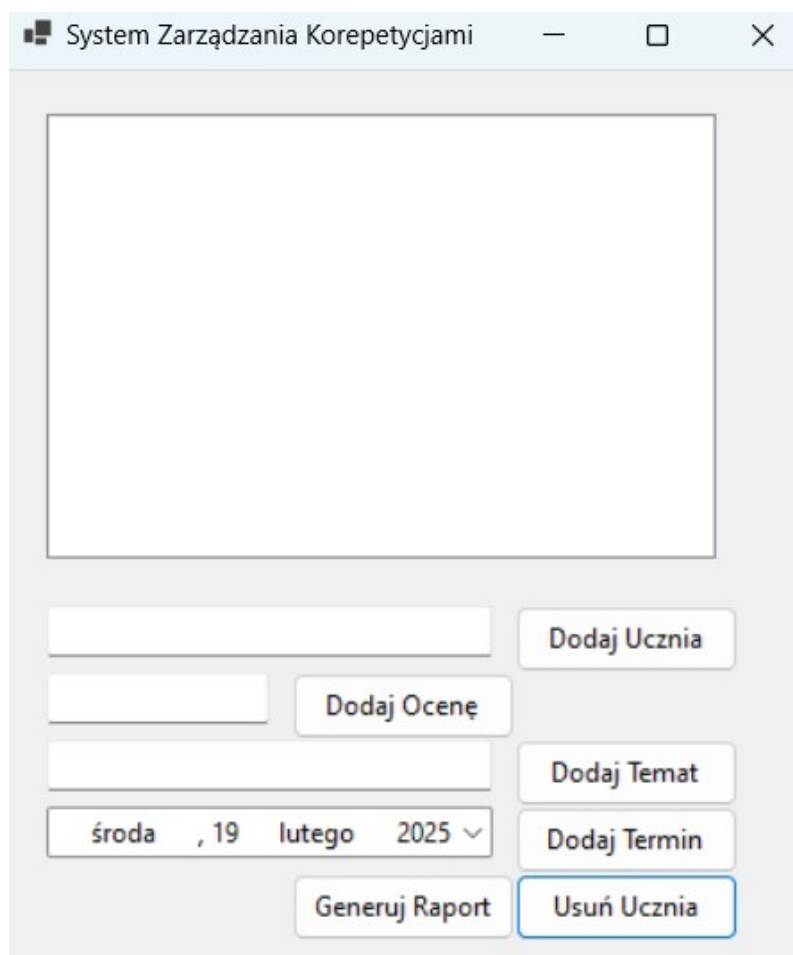
Rysunek 4.6: Dodanie tematu

Aplikacja pozwala na przypisanie tematów zajęć do konkretnego ucznia. Aby dodać temat, użytkownik musi wybrać ucznia z listy oraz wpisać nazwę tematu w odpowiednim polu tekstowym. Po kliknięciu przycisku "Dodaj temat", temat zostanie zapisany i przypisany do wybranego ucznia.

The screenshot shows a window titled "System Zarządzania Korepetycjami". Inside the window, there is a list box at the top containing the name "Patrycja Starzyk". Below the list box, there are several input fields and buttons. The buttons are: "Dodaj Ucznia", "Dodaj Ocenę", "Dodaj Temat", "Dodaj Termin" (highlighted in blue), "Generuj Raport", and "Usuń Ucznia". The date field shows "środa , 19 lutego 2025".

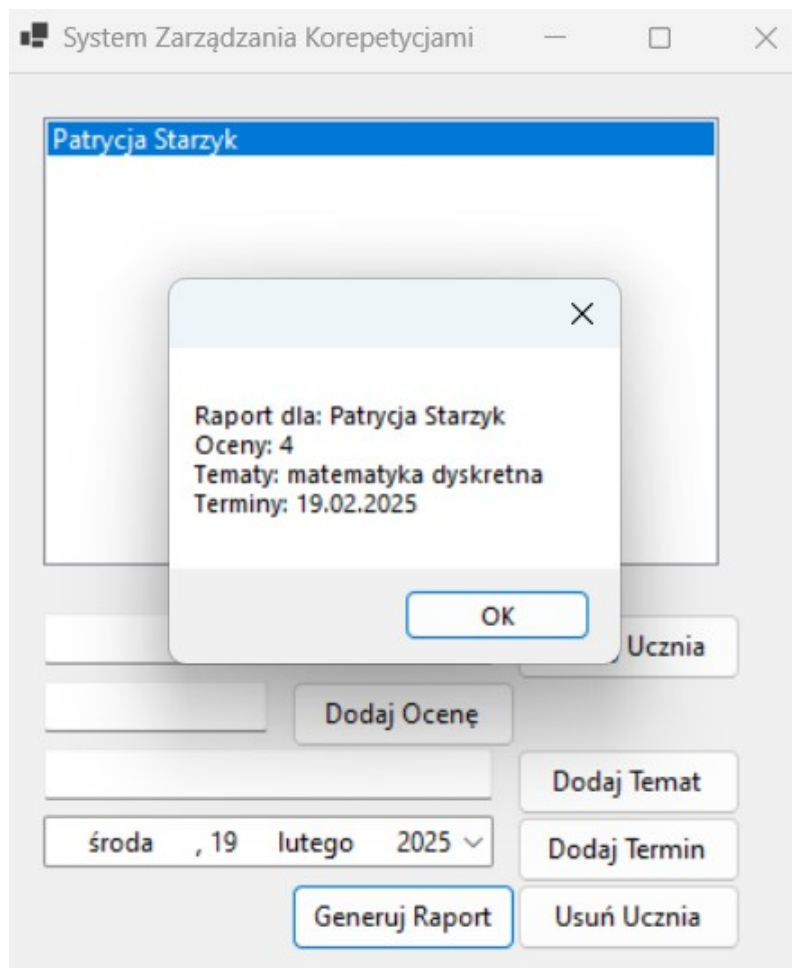
Rysunek 4.7: Dodanie terminu

Aplikacja umożliwia przypisanie terminu zajęć do wybranego ucznia. Aby dodać termin, użytkownik musi zaznaczyć ucznia z listy oraz wybrać odpowiednią datę i godzinę w polu wyboru daty. Następnie po kliknięciu przycisku "Dodaj termin", nowy termin zostanie zapisany w systemie i przypisany do konkretnego ucznia.



Rysunek 4.8: Usunięcie ucznia z bazy

Aplikacja umożliwia usunięcie ucznia z listy zapisanych danych. Aby to zrobić, użytkownik musi wybrać odpowiedniego ucznia z listy, a następnie kliknąć przycisk "Usuń ucznia". Po zatwierdzeniu operacji, wszystkie powiązane z nim dane, takie jak oceny, tematy zajęć oraz terminy spotkań, zostaną usunięte.



Rysunek 4.9: Wyświetlenie raportu ucznia

Po kliknięciu przycisku "Generuj raport" zostanie wyświetlone okienko zawierające szczegółowe informacje na temat wcześniej wybranego ucznia. Raport obejmuje jego oceny, tematy zajęć oraz zaplanowane terminy spotkań. Jeśli użytkownik nie wybierze ucznia przed wygenerowaniem raportu, pojawi się komunikat informujący o konieczności dokonania wyboru.

Rozdział 5

Podsumowanie

Projekt aplikacji dla korepetytora został zaprezentowany jako kompletne narzędzie do zarządzania uczniami, ocenami, tematami, terminami zajęć oraz generowania raportów. W ramach realizacji projektu zaprojektowano i zaimplementowano funkcjonalności pozwalające na łatwe dodawanie, edytowanie i usuwanie uczniów, a także przypisywanie im ocen, tematów zajęć oraz terminów. Całość zarządzania odbywa się za pomocą intuicyjnego interfejsu graficznego (GUI), zapewniając wygodną obsługę w codziennej pracy korepetytora.

Bibliografia

- [1] Harvey M. Deitel, Paul J. Deitel, *C# 7.0 i .NET Core: Kompletny przewodnik*, Helion, Gliwice, Polska, 2017.

Spis rysunków

2.1	Diagram klas	8
3.1	Diagram Ganta	10
4.1	Główny interfejs	11
4.2	Wpisanie ucznia	12
4.3	Dodanie ucznia	13
4.4	Uczeń dodany do listy	14
4.5	Dodanie oceny	15
4.6	Dodanie tematu	16
4.7	Dodanie terminu	17
4.8	Usunięcie ucznia z bazy	18
4.9	Wyświetlenie raportu ucznia	19

Spis tabel