



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

KOLEGIUM INFORMATYKI STOSOWANEJ

Kierunek: INFORMATYKA

Specjalność:

Filip Hejzyk
w70682

Przeglądarka Zabytków Światowego Dziedzictwa UNESCO

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	6
2.1 Opis Techniczny	6
2.2 Opis struktury	7
2.3 Zarządzanie Bazą Danych	8
3 Harmonogram realizacji projektu	9
4 Prezentacja warstwy użytkowej projektu	10
5 Podsumowanie	13
Bibliografia	14
Spis rysunków	15
Spis tablic	16

Wstęp

Lista Światowego Dziedzictwa UNESCO jest to stale rosnący zbiór ponad 1200 zabytków o wartości kulturowej bądź przyrodniczej. Wraz z jej poszerzaniem się, coraz bardziej odnotowywalna stała się nieporęczność oraz powolne działanie spisu zabytków znajdujących się na liście, który jest dostępny na oficjalnej stronie UNESCO. W celu rozwiązania tego problemu, oraz stworzenia alternatywy o znacznie lepszej wydajności, przedstawiony został tutaj prototyp przeglądarki zabytków światowego dziedzictwa UNESCO, która dzięki działaniu jako lokalny program, cechuje się lepszym komfortem obsługi.

Rozdział 1

Opis założeń projektu

1.1 Cele projektu

Celem projektu jest stworzenie przeglądarki zabytków światowego dziedzictwa UNESCO, działającej jako lokalny program komputerowy. Ma ona na celu zniwelowanie problemu powolnego działania listy znajdującej się na oficjalnej stronie UNESCO, oferując lepszą wydajność i komfort pracy. Problem ten jest dostrzegalny już przy wejściu na stronę UNESCO, co utrudnia zastosowanie edukacyjne, bądź turystyczne zawartych na niej danych. W celu zniwelowania opisanego problemu konieczne jest zaimplementowanie prostej i wydajnej struktury danych, opcje szybkiego i klarownego zarządzania nimi, oraz prosty i lekki interfejs umożliwiający sprawną nawigację. Pierwszym krokiem w celu realizacji zadania będzie zaplanowanie hierarchii klas, oraz wymogów funkcjonalnych. Następnym etapem będzie implementacja ich w środowisku i integracja z bazą danych. Ostatecznie, testowanie i optymalizacja wynikającego z poprzednich kroków programu. W efekcie powstać powinna lekka, prosta w obsłudze i wydajna aplikacja pozwalająca na agregację i manipulowanie danymi o zabytkach Światowego Dziedzictwa UNESCO, tak aby informacje zawarte mogły być na bieżąco uaktualniane.

1.2 Wymagania funkcjonalne i нефunkcjonalne

Wymagania funkcjonalne programu:

- Odczyt i zapis spreparowanych danych
- Wyświetlanie danych wprowadzonych w program
- Manipulacja wprowadzonymi danymi z poziomu aplikacji
 - Dodawanie nowych rekordów
 - Usuwanie rekordów
 - modyfikacja istniejących rekordów

Wymagania нефunkcjonalne:

- Program powinien być wydajny. Przy rzeczywistej liczbie zabytków spowolnienie działania powinno być nieznaczne (czas odczytu i nadpisywania ok. kilku sekund)
- Menu i obsługa programu powinny być klarowne i intuicyjne, a sam program nie powinien ważyć więcej niż kilka MB.
- Natychmiastowy zapis zmian wprowadzanych w danych, powinien gwarantować niezawodność programu i brak utraty danych w wypadku awarii.
- Praca z programem nie powinna wymagać połączenia z internetem, oparta wyłącznie o lokalną maszynę użytkownika.

Rozdział 2

Opis struktury projektu

2.1 Opis Techniczny

Specyfikacja techniczna projektu:

- **Zastosowany język programowania:** C# (.NET 8.0)
- **Zastosowane środowisko:** Microsoft Visual Studio Community 2022 (64-bit) - Version 17.13.2
- **Rodzaj Aplikacji:** Aplikacja Konsolowa

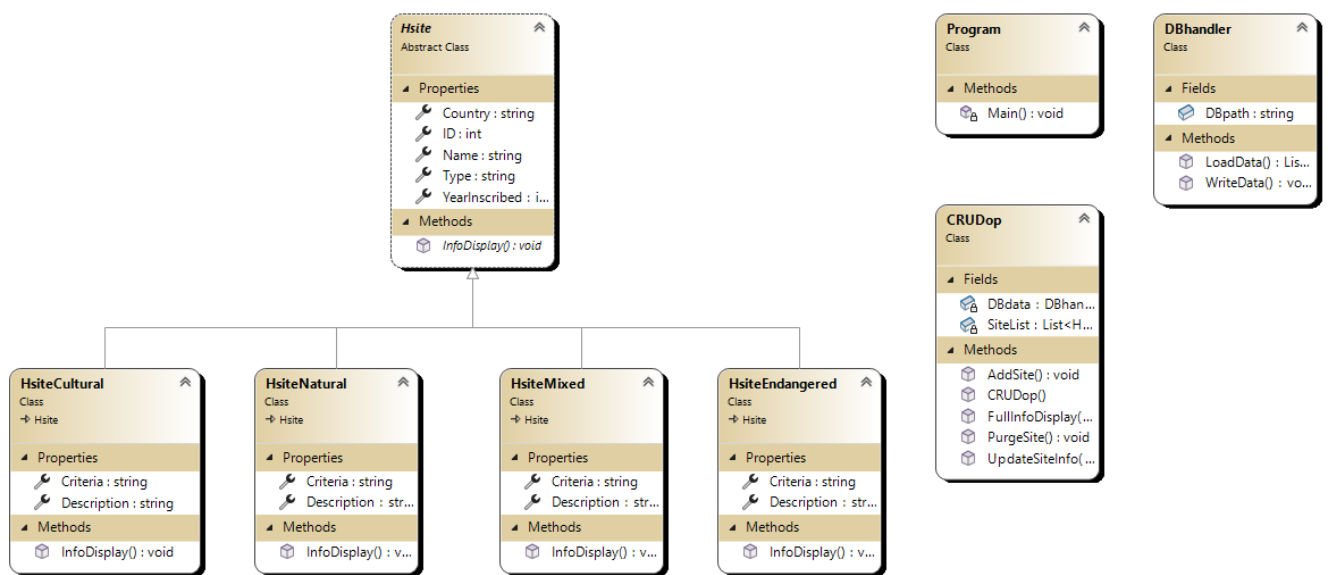
Wymagania sprzętowe rekomendowane na podstawie sprzętu na którym program był testowany:

- **OS:** Windows 10
- **Procesor:** AMD Ryzen 5 5500
- **Pamięć RAM:** 32 GB
- **Przestrzeń na dysku:** 10 MB

Minimalne wymagania sprzętowe:

- **OS:** Windows 10/11
- **Procesor:** 1 GHz lub więcej
- **Pamięć RAM:** 4 GB
- **Przestrzeń na dysku:** 1 MB

2.2 Opis struktury



Rysunek 2.1: Diagram Klas dla programu

Struktura klas w programie - Program składa się łącznie z ośmiu klas, w tym hierarchii pięciu klas opartych o abstrakcyjną klasę **Hsite**:

- **Program** - Główna klasa programu, odpowiada za uruchamianie i sterowanie całym programem poprzez metodę **Main()**.
- **DBhandler** - klasa odpowiedzialna za powiązanie Bazy Danych z programem. Pole **DBpath** zawiera ścieżkę pliku, natomiast metody **LoadData()** i **WriteData()** pozwalają odpowiednio na odczytywanie i nadpisywanie pliku zawierającego bazę danych.
- **CRUDop** - klasa odpowiedzialna za wykonywanie operacji na danych. Pole **DBdata** pobiera dane z **DBpath** poprzez **LoadData()** i następnie wypełnia nimi listę zawartą w polu **SiteList** (Operacja Wykonywana jest w metodzie **CRUDop()** przy uruchamianiu programu. Pozostałe metody **AddSite()**, **PurgeSite()**, **FullInfoDisplay()** i **UpdateSiteInfo()** są przeznaczone do wykonywania operacji na danych przez użytkownika.
- **Hsite** - klasa abstrakcyjna, służąca jako baza dla klas wyróżniających poszczególne typy zabytków. Zawiera właściwości **ID**, **Name**, **Country**, **Type** i **YearInscribed** oraz abstrakcyjną metodę **DisplayInfo()**
- **HsiteCultural**, **HsiteNatural**, **HsiteMixed**, **HsiteEndangered** - Są to klasy pochodne od klasy abstrakcyjnej **Hsite**, podzielone według czterech typów zabytków: Kulturowe, Przyrodnicze, Mieszane i Zagrożone. Posiadają właściwości **Criteria** oraz **Description**, a także pochodne metody **DisplayInfo()**.

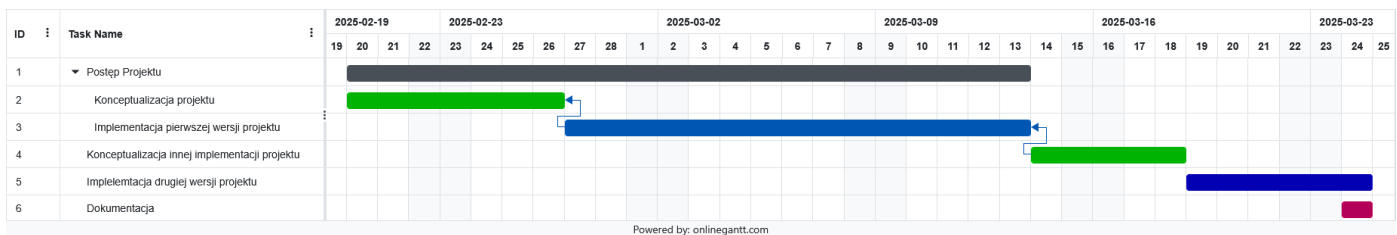
2.3 Zarządzanie Bazą Danych

Dane wykorzystywane przez program są zawarte w pliku UNESCOdata.txt. W rekordy przyjmują następujący format: **Type;ID;Name;Country;YearInscribed;Criteria;Description**, gdzie symbol ";" służy jako oddzielnik dla każdej rubryki. Z poziomu programu użytkownik może działać na danych na 4 następujące sposoby:

- Wyświetlać zawartość bazy z wykorzystaniem metody **FullInfoDisplay()**
- Dodawać nowe rekordy do bazy z wykorzystaniem metody **AddSite()**
- Usuwać rekordy z bazy z wykorzystaniem metody **PurgeSite()**
- Aktualizować dane zawarte w rekordach z wykorzystaniem metody **UpdateSiteInfo()**

Rozdział 3

Harmonogram realizacji projektu



Rysunek 3.1: Diagram Gantta wdrażania projektu

Na zielono zostały oznaczone okresy poświęcone planowaniu koncepcji programu, m.in. struktura, funkcjonalność, hierarchia klas. Na niebiesko zostały zaznaczone okresy implementacji programu w środowisku. Kolor Czerwony odzwierciedla przygotowanie dokumentacji. Jedną z głównych przeszkód w wykonaniu okazała się trudność z implementacją GUI w ASP.NET. Ostatecznie koncepcja została przebudowana i zaimplementowana w formie aplikacji konsolowej.

Repozytorium: https://github.com/GenEuffe/UNESCO_Hlist_Browser

System Kontroli Wersji: Nie było wykorzystane w trakcie tworzenia programu.

Rozdział 4

Prezentacja warstwy użytkowej projektu

Program jest aplikacją konsolową. Po uruchomieniu programu poprzez plik **UNESCO.exe** użytkownik trafia do menu (Rys. 4.1), gdzie zostaje powitany przez szyld, krótki opis celu programu, oraz listę opcji które użytkownik może wybrać.

```
=====
PRZEGLADARKA ZABYTEKÓW UNESCO
=====
Ten program ma na celu umożliwić użytkownikowi wyświetlanie oraz manipulację baza danych zabytków UNESCO.
Dostępne Opcje:
1. Wyświetl wszystkie zabytki
2. Dodaj zabytek
3. Usun zabytek
4. Aktualizuj zabytek
5. Wyjdź
> -
```

Rysunek 4.1: Menu główne programu

Wybierając je po kolei, wpierv użytkownik wybierze opcję **"1. Wyświetl wszystkie zabytki"**. Po jej wybraniu, program wyświetli użytkownikowi aktualną listę zabytków, przechowywaną w bazie danych. (Rys. 4.2)

```
ID: 1 | Nazwa: Wielki Mur Chinski | Kraj: Chiny | Rok wpisania: 1987 | Typ: Kulturowy | Kryteria Wpisu: I
Opis: Jeden z największych cudów inżynierii starożytnego świata. Rozciąga się na długości ponad 21 000 km.

ID: 2 | Nazwa: Koloseum | Kraj: Włochy | Rok wpisania: 1980 | Typ: Kulturowy | Kryteria Wpisu: I
Opis: Amfiteatr Flawiuszów, symbol starożytnego Rzymu. Mógł pomieścić nawet 50 000 widzów.

ID: 3 | Nazwa: Park Narodowy Yellowstone | Kraj: Stany Zjednoczone | Rok wpisania: 1978 | Typ: Przyrodniczy | Kryteria Wpisu: VII
Opis: Pierwszy park narodowy na świecie, znany z gejzerów i dzikiej przyrody. Jest domem dla wielu gatunków zwierząt, w tym niedźwiedzi grizzly.

ID: 4 | Nazwa: Wielka Rafa Koralowa | Kraj: Australia | Rok wpisania: 1981 | Typ: Przyrodniczy | Kryteria Wpisu: VII,VIII,IX,X
Opis: Największy na świecie system raf koralowych. Przy powierzchni 346 000 km, jest widoczny z kosmosu.

ID: 5 | Nazwa: Machu Picchu | Kraj: Peru | Rok wpisania: 1983 | Typ: Mieszany | Kryteria Wpisu: I,III,VII,IX
Opis: Starożytne miasto Inków położone w Andach. Położone na wysokości 1999-2400 m. n. p. m.

ID: 6 | Nazwa: Góra Athos | Kraj: Grecja | Rok wpisania: 1988 | Typ: Mieszany | Kryteria Wpisu: I,VII
Opis: Skupisko wielu klasztorów prawosławnych. Jest miejscem pielgrzymek od ponad 1000 lat.

ID: 7 | Nazwa: Stare Miasto w Aleppo | Kraj: Syria | Rok wpisania: 1986 | Typ: Zagrożony | Kryteria Wpisu: I,IV
Opis: Jedno z najstarszych miast na świecie. Znaczna część miasta uległa zniszczeniu na skutek konfliktów.

ID: 8 | Nazwa: Królewskie Kopalnie Soli w Wieliczce i Bochni | Kraj: Polska | Rok wpisania: 1976 | Typ: Kulturowy | Kryteria Wpisu: IV
Opis: Kopalnia Soli Kamiennej w Wieliczce pod Krakowem. Sole wydobywane w kopalni pochodzą z miocenu.

ID: 9 | Nazwa: Akropol | Kraj: Grecja | Rok wpisania: 1987 | Typ: Kulturowy | Kryteria Wpisu: I,IV
Opis: Starożytny kompleks świątynny w Atenach. Jest symbolem starożytnej greckiej cywilizacji i kultury.

ID: 10 | Nazwa: Park Narodowy Serengeti | Kraj: Tanzania | Rok wpisania: 1981 | Typ: Przyrodniczy | Kryteria Wpisu: VII,IX
Opis: Słynny z corocznej migracji zwierząt. Jest jednym z najbardziej zróżnicowanych ekosystemów na świecie.
```

Rysunek 4.2: Lista zabytków wyświetlona przez program.

Użytkownikowi ukaże się pełna lista zabytków. Każdy rekord na liście posiada 7 rubryk zawierających informacje o danym zabytku. Wszystkie rubryki są podpisane i oddzielone symbolem "|".

Po wykonaniu każdej operacji program samoistnie powróci do menu, umożliwiając użytkownikowi dalszą pracę.

Następną opcją którą użytkownik może wybrać jest "**2. Dodaj zabytek**". Po jej wybraniu, użytkownik dostanie szereg pytań odnoście informacji o nowym zabytku. (rys. 4.3)

```
Podaj typ zabytku (Kulturowy/Przyrodniczy/Mieszany/Zagrozony):  
Kulturowy  
Podaj ID:  
11  
Podaj Nazwe Obiektu:  
WSIiZ  
Podaj kraj pochodzenia:  
Polska  
Podaj Rok Wpisania obiektu na liste:  
2025  
Podaj Kryteria:  
IV,II  
Podaj Opis:  
Wyższa Szkoła Informatyki i Zarządzania z siedzibą w Rzeszowie  
Zabytek pomyślnie dodany.
```

Rysunek 4.3: Wprowadzanie danych do nowego rekordu.

Po wprowadzeniu informacji, program poinformuje użytkownika, że do bazy danych trafił nowy zabytek. Na powyższym rysunku (rys. 4.3) jest przedstawiony proces dodawania nowego zabytku, z wykorzystaniem WSIiZ jako przykładu. Wybierając ponownie opcję 1, użytkownik będzie mógł zobaczyć, że zabytek rzeczywiście znajduje się na liście. (rys. 4.4)

```
ID: 11 | Nazwa: WSIiZ | Kraj: Polska | Rok wpisania: 2025 | Typ: Kulturowy | Kryteria Wpisu: IV,II  
Opis: Wyższa Szkoła Informatyki i Zarządzania z siedziba w Rzeszowie
```

Rysunek 4.4: Rekord z nowo dodanym zabytkiem.

Po wybraniu opcji **3. Usuń zabytek**, program prosi użytkownika o podanie numeru ID zabytku który pragnie on usunąć z listy. (rys. 4.5). Jako przykład, wprowadzony został nr. ID wcześniej dodanego WSIiZu. Po wprowadzeniu go, program poinformuje, że zabytek został pomyślnie usunięty z listy. Jeżeli w celu weryfikacji, użytkownik ponownie wyświetli listę wszystkich zabytków, zobaczy, że nie znajduje się on już na liście (rys. 4.6).

```
Podaj ID zabytku do usuniecia:  
11  
Zabytek pomyślnie usuniety.
```

Rysunek 4.5: Zabytek został usunięty po wprowadzeniu przypisanego mu ID.

```

ID: 1 | Nazwa: Wielki Mur Chinski | Kraj: Chiny | Rok wpisania: 1987 | Typ: Kulturowy | Kryteria Wpisu: I
Opis: Jeden z największych cudów inżynierii starożytnego świata. Rozciąga się na długości ponad 21 000 km.

ID: 2 | Nazwa: Koloseum | Kraj: Włochy | Rok wpisania: 1980 | Typ: Kulturowy | Kryteria Wpisu: I
Opis: Amfiteatr Flawiuszów, symbol starożytnego Rzymu. Mógł pomieścić nawet 50 000 widzów.

ID: 3 | Nazwa: Park Narodowy Yellowstone | Kraj: Stany Zjednoczone | Rok wpisania: 1978 | Typ: Przyrodniczy | Kryteria Wpisu: VII
Opis: Pierwszy park narodowy na świecie, znany z gejzerów i dzikiej przyrody. Jest domem dla wielu gatunków zwierząt, w tym niedźwiedzi grizzly.

ID: 4 | Nazwa: Wielka Rafa Koralowa | Kraj: Australia | Rok wpisania: 1981 | Typ: Przyrodniczy | Kryteria Wpisu: VII,VIII,IX,X
Opis: Największy na świecie system raf koralowych. Przy powierzchni 346 000 km, jest widoczny z kosmosu.

ID: 5 | Nazwa: Machu Picchu | Kraj: Peru | Rok wpisania: 1983 | Typ: Mieszany | Kryteria Wpisu: I,III,VII,IX
Opis: Starożytne miasto Inków położone w Andach. Położone na wysokości 1999-2400 m. n. p. m.

ID: 6 | Nazwa: Góra Athos | Kraj: Grecja | Rok wpisania: 1988 | Typ: Mieszany | Kryteria Wpisu: I,VII
Opis: Skupisko wielu klasztorów prawosławnych. Jest miejscem pielgrzymek od ponad 1000 lat.

ID: 7 | Nazwa: Stare Miasto w Aleppo | Kraj: Syria | Rok wpisania: 1986 | Typ: Zagrożony | Kryteria Wpisu: I,IV
Opis: Jedno z najstarszych miast na świecie. Znaczna część miasta uległa zniszczeniu na skutek konfliktów.

ID: 8 | Nazwa: Królewskie Kopalnie Soli w Wieliczce i Bochni | Kraj: Polska | Rok wpisania: 1976 | Typ: Kulturowy | Kryteria Wpisu: IV
Opis: Kopalnia Soli Kamiennej w Wieliczce pod Krakowem. Sole wydobywane w kopalni pochodzą z miocenu.

ID: 9 | Nazwa: Akropol | Kraj: Grecja | Rok wpisania: 1987 | Typ: Kulturowy | Kryteria Wpisu: I,IV
Opis: Starożytny kompleks świątynny w Atenach. Jest symbolem starożytnej greckiej cywilizacji i kultury.

ID: 10 | Nazwa: Park Narodowy Serengeti | Kraj: Tanzania | Rok wpisania: 1981 | Typ: Przyrodniczy | Kryteria Wpisu: VII,IX
Opis: Słynny z corocznej migracji zwierząt. Jest jednym z najbardziej zróżnicowanych ekosystemów na świecie.

```

Rysunek 4.6: Zabytek o nr. ID 11 nie jest już widoczny na liście.

Następną opcją która jest dostępna dla użytkownika jest **"4. Aktualizuj zabytek."** Po jej wybraniu program poprosi o ID zabytku który należy zaktualizować. Po wprowadzeniu, program poprosi użytkownika o wprowadzenie nowych danych dla rekordu o podanym numerze ID. (Rys. 4.7) Po ich wprowadzeniu, program poinformuje użytkownika, że aktualizacja się powiodła. Sprawdzając ponownie listę w celu weryfikacji, użytkownik zobaczy zaktualizowane dane w odpowiednim rekordzie. (rys. 4.8 i 4.9)

```

Podaj ID zabytku do aktualizacji:
2
Podaj Nazwę Obiektu:
Koloseum
Podaj kraj pochodzenia:
Włochy
Podaj Rok Wpisania obiektu na listę:
1980
Podaj Kryteria:
I,II,III
Podaj Opis:
Amfiteatr Flawiuszów, symbol starożytnego Rzymu. Mógł pomieścić nawet 75 000 widzów.
Zabytek pomyślnie zaktualizowany.

```

Rysunek 4.7: Wprowadzanie korekty do danych zawartych w rekordzie o koloseum.

```

ID: 2 | Nazwa: Koloseum | Kraj: Włochy | Rok wpisania: 1980 | Typ: Kulturowy | Kryteria Wpisu: I
Opis: Amfiteatr Flawiuszów, symbol starożytnego Rzymu. Mógł pomieścić nawet 50 000 widzów.

```

Rysunek 4.8: Rekord dla Koloseum w Rzymie przed aktualizacją.

```

ID: 2 | Nazwa: Koloseum | Kraj: Włochy | Rok wpisania: 1980 | Typ: Kulturowy | Kryteria Wpisu: I,II,III
Opis: Amfiteatr Flawiuszów, symbol starożytnego Rzymu. Mógł pomieścić nawet 75 000 widzów.

```

Rysunek 4.9: Rekord dla Koloseum w Rzymie po aktualizacji.

Ostatnia opcja **"5. Wyjdź"** spowoduje zamknięcie programu i zakończenie pracy.

Rozdział 5

Podsumowanie

Projekt w większości wykonuje zamierzone cele. Spełnia wszystkie wymagania funkcjonalne, oraz większość wymagań нефункциональных: Z przykładową bazą danych załączoną do programu, cały program waży poniżej 1 MB. Wykonuje zaimplementowane operacje w minimalnym czasie, a wszystkie zmiany są zapisywane natychmiastową, co zabezpiecza przed utratą danych w przypadku awarii. Program funkcjonuje całkowicie lokalnie, w oparciu o zasoby komputera, co przy jego prostocie sprawia, że jest znacznie bardziej wydajny.

Pomimo, że trudności z implementacją GUI w ASP.NET doprowadziły do opóźnienia w produkcji, przy dalszym rozwoju programu głównymi celami byłoby zaimplementowanie graficznego interfejsu, dalsze uproszczenie wybranych operacji, poszerzenie funkcjonalności programu o nowe operacje oraz implementacja kompatybilności z bazami danych różnych formatów, takich jak sql czy też csv.

Bibliografia

- [1] <https://www.unesco.org/en/world-heritage/grid?hub=68246> z dnia 24.03.2025
- [2] <https://learn.microsoft.com/en-us/dotnet/csharp/>

Spis rysunków

2.1	Diagram Klas dla programu	7
3.1	Diagram Gantta wdrażania projektu	9
4.1	Menu główne programu	10
4.2	Lista zabytków wyświetlona przez program.	10
4.3	Wprowadzanie danych do nowego rekordu.	11
4.4	Rekord z nowo dodanym zabytkiem.	11
4.5	Zabytek został usunięty po wprowadzeniu przypisanego mu ID.	11
4.6	Zabytek o nr. ID 11 nie jest już widoczny na liście.	12
4.7	Wprowadzanie korekty do danych zawartych w rekordzie o koloseum.	12
4.8	Rekord dla Koloseum w Rzymie przed aktualizacją.	12
4.9	Rekord dla Koloseum w Rzymie po aktualizacji.	12

Spis tabel