**Budowa i implementacja aplikacji zarządzania zbiorami galerii sztuki**

PROJEKT DYPLOMOWY

Poznań 2024

**DANE PARTNERÓW**

**A1. Dane Promotora**

|  |  |
| --- | --- |
| Imię i nazwisko | **Marek Gałązka** |
| Stopień / Tytuł naukowy | dr |
| Data i podpis |  |

**A2. Dane członków Zespołu projektu**

|  |  |
| --- | --- |
| Imię i nazwisko | **Oliwia Głodek** |
| Kierunek studiów | Informatyka |
| Tryb studiów | Niestacjonarne |
| Data i podpis |  |

|  |  |
| --- | --- |
| Imię i nazwisko | **Emil Gielek** |
| Kierunek studiów | Informatyka |
| Tryb studiów | Niestacjonarne |
| Podpis |  |

|  |  |
| --- | --- |
| Imię i nazwisko |  |
| Kierunek studiów |  |
| Tryb studiów |  |
| Data i podpis |  |

|  |  |
| --- | --- |
| Imię i nazwisko |  |
| Kierunek studiów |  |
| Tryb studiów |  |
| Data i podpis |  |

**ZAŁOŻENIA PROJEKTU**

**B1. Opis projektu**

1. Uzasadnienie wyboru tematu
   1. Profesjonalne i rozbudowane oprogramowanie do muzeów, bibliotek czy galerii jest kosztowne choć nie zawsze obligatoryjne w tak rozbudowanej formie. Instytucje prywatne, działające w małym zakresie, zwracają uwagę na chęć cięcia kosztów, i zastosowania prostego oprogramowania nie tylko na wczesnym etapie rozwoju działalności ale także dalszym jako wystarczające i kompletne narzędzie. W trakcie badania rynku Poznańskiego, zauważyliśmy tę potrzebę oraz inne jak chęć łatwiejszego monitorowania oraz prezentowania wydarzeń np. biblioteki, muzea czy galerie sztuki, podjęliśmy decyzję o stworzeniu prostej aplikacji usprawniającej pracę takim instytucjom. Zważywszy na braki takich aplikacji szczególnie dla galerii sztuki, to zdecydowaliśmy że dobrym przykładem będzie właśnie ta działalność. Dodatkową motywacją na podjęcie tego tematu jest nie tylko kwestia rynkowa ale także chęć wsparcia mniejszych instytucji w dalszym rozwoju oraz brak podobnych rozwiązań w tym zakresie. Wartością dodaną dla nas była chęć sprawdzenia się w tworzeniu takiego oprogramowania „od zera” z zastosowaniem nowych nam technologii i narzędzi do tworzenia oprogramowania.
2. Cel główny i cele szczegółowe projektu
   1. Głównym celem pracy jest stworzenie aplikacji umożliwiającej zarządzanie inwentarzem, raportowanie oraz archiwizację danych, skierowanej przede wszystkim do małych instytucji, w tym głownie galerii sztuki
   2. Cele szczegółowe:
      1. Badanie rynku pod kątem rozpoznania co zdecydowało o wyborze tych rozwiązań na terenie Poznania (1 miesiąc).
      2. Dyskusja o formie działania „backend” oraz budowie frontend.
      3. Zapoznanie się z tworzeniem aplikacji WPF w środowisku Visual Studio
      4. Przygotowanie środowiska pracy w tym instalacja potrzebnych bibliotek, narzędzi diagnostycznych, postawienie bazy danych.
      5. Zaprojektowanie diagramu UML, bazy danych z tabelami za pomocą narzędzi draw.io i SQLite oraz utworzenie tabel potrzebnych do aplikacji.
      6. Zapoznanie się z działaniem bibliotek m. in. QuestPDF i testy ich implementacji dla naszych zastosowań.
      7. Projektowanie w Figma oraz oprogramowanie w Visual Studio kolejnych elementów poszczególnych okien aplikacji, w tym menu boczne.
      8. Utworzenie stopni dostępu dla różnego rodzaju poziomów uprawnień.
      9. Wdrożenie systemu powiadomień e-mailingowych związanych z aktualnymi wydarzeniami.
      10. Optymalizacja aplikacji w tym zarządzanie pamięcią aplikacji.
      11. Testowanie aplikacji.
3. Zakres podmiotowy, przedmiotowy, czasowy i przestrzenny
   1. Zakres podmiotowy: Małe prywatne instytucje galerii sztuki szukające prostych rozwiązań do zarządzania swoimi zbiorami w tym ich raportowania przez różnych pracowników.
   2. Zakres przedmiotowy: Aplikacja do zarządzenia przedmiotami określonej instytucji, w której można zmieniać eksponaty, tworzyć datowane wystawy z powiadomieniami na maile użytkowników oraz raportować ich stan.
   3. Zakres czasowy: Od momentu upublicznienia, stabilnej, desktopowej wersji, aż do momentu utraty kompatybilności aplikacji z obecnymi systemami operacyjnymi, lub utraty zainteresowania naszym rozwiązaniem przez wdrożenie lepszego, nowocześniejszego rozwiązania.
   4. Zakres przestrzenny: Zakres oparty o umiejscowienie prywatnych instytucji (m.in. galerii sztuki), gdzie potrzeba jest zastosowania naszego oprogramowania z szczególnym uwzględnieniem obszaru poznańskiego.
4. Metody i techniki badawcze

- Visual Studio

- SQLite

- QuestPDF

- Adobe Reader

- draw.io

- Figma

- Trello

- GitHub

**B2. Zadania w projekcie**

|  |  |  |
| --- | --- | --- |
| Cele szczegółowe projektu | Zadania w projekcie  oraz termin rozpoczęcia i zakończenia realizacji zadania | Osoby zaangażowane w realizację zadania |
| Cel 1: Badanie rynku pod kątem rozpoznania co zadecydowało o wyborze tych rozwiązań | Zadanie 1: Rozmowa z pracownikami muzeum, w celu poznania ich potrzeb | 1. Emil Gielek  2. Oliwia Głodek |
| Zadanie 2: Analiza istniejących już rozwiązań | 1. Emil Gielek  2. Oliwia Głodek |
| Cel 2: Zapoznanie z biblioteką QuestPDF | Zadanie 1: Zapoznanie się ze stroną biblioteki www.questpdf.com | 1. Emil Gielek  2. Oliwia Głodek |
| Zadanie 2: Zapoznanie się z nagraniami o tworzeniu PDF na platformie YouTube | 1. Emil Gielek  2. Oliwia Głodek |
| Zadanie 3: Próby implementacji, testowanie na przykładach a w końcu implementacja faktyczna | 1. Emil Gielek |
| Zadanie 4: Pełna implementacja generowania PDF w wariantach ale także plików CSV i podglądu w Adobe Reader |
| Cel 3: Wybór technologii frontend oraz backend oraz ich implementacja | Zadanie 1: Zapoznanie się z materiałami kanałów Youtube np. Mdemy oraz strony codeproject.pl | 1. Emil Gielek  2. Oliwia Głodek |
| Zadanie 2: Stworzenie pierwszych widoków w VisualStudio w języku C# w projekcie opartym o aplikację WPF w tym stworzenie lokalnej bazy danych w SQLite pod użytkowników | 1. Oliwia Głodek |
| Zadanie 3: Stworzenie kolejnych opcji menu w aplikacji w tym: Eksponaty, Raporty PDF, użytkownicy i itp. | 1. Emil Gielek |
| Zadanie 4: Stworzenie systemu logowania i rejestrowania w oparciu o bazę danych i uprawnienia | 1. Oliwia Głodek |
| Zadanie 5: Stworzenie logiki dodawania, usuwania i edycji eksponatów i wystaw | 1. Oliwia Głodek |
| Cel 4: Zaprojektowanie bazy danych aplikacji | Zadanie 1: Stworzenie 5 tabel i struktury: użytkownicy, eksponaty, wystawy, historia i eksponaty w wystawach | 1. Emil Gielek  2. Oliwia Głodek |
| Zadanie 2: Stworzenie “TRIGGERÓW” dla automatycznego rejestrowania zdarzeń w bazie przez użytkowników – stworzenie rekordu “current\_user” | 1. Emil Gielek |
| Zadanie 3: Stworzenie logiki uprawnień użytkowników I implementacja | 1. Oliwia Głodek |
| Cel 5: Pisanie i testowanie aplikacji | Zadanie 1: Kontrolowanie funkcjonalności, stabilności i ogólnego działania aplikacji (nadzór prac) | 1. Emil Gielek  2. Oliwia Głodek |
| Zadanie 2: Testowanie stabilności aplikacji i usuwanie błędów | 1. Oliwia Głodek |
| Zadanie 3: Optymalizacja programu i dodanie elementów wizualnych | 1. Emil Gielek |

**REALIZACJA**

**C1. Opracowanie projektu**

1. **Założenia teoretyczne**

Postęp technologiczny spowodował pojawienie się nowych możliwości zarządzania zbiorami muzealnymi, które ułatwiają dostęp do zasobów oraz wspierają ich ochronę i konserwację. Nasz system zarządzania zbiorami muzeum ma na celu wsparcie instytucji kulturowych w procesie dokumentowania, katalogowania oraz udostępniania eksponatów w formie cyfrowej. Digitalizacja zbiorów muzealnych jest już koniecznością wynikającą z oczekiwań społecznych, dlatego system umożliwia kompleksowe zarzadzanie informacjami o zbiorach poprzez szczegółowe opisy eksponatów i wystaw, informacje o ich lokalizacji oraz uczestnictwa w wystawach. Celem systemu jest intuicyjność i dostępność. Interfejs systemu został zaprojektowany z myślą o użytkownikach, którzy nie są zaawansowani technologicznie.

Aplikacja desktopowa jest programem przeznaczonym do uruchamiana na komputerze osobistym lub laptopie. Działa na systemie operacyjnym, korzystając z jego zasobów takich jak pamięć RAM, procesor czy twardy dysk. Tego typu aplikacje zainstalowane są bezpośrednio na urządzeniu i mogą działać zarówno online jak i offline. Wśród aplikacji desktopowych możemy wyróżnić:

- aplikacje natywne – czyli takie, które są przeznaczone dla konkretnego systemu operacyjnego np. Windows lub Linux. Charakteryzują się wysoką wydajnością oraz pełnym wsparciem dla funkcji offline,

- aplikacje multiplatformowe – czyli takie, które są dostosowane aby działać na różnych systemach operacyjnych przy minimalnych zmianach w kodzie. Ich zaletą jest przede wszystkim jeden kod źródłowy, który może być uruchomiony na różnych systemach oraz niższe koszty produkcji i utrzymania,

- aplikacje hybrydowe – czyli takie, które łączą funkcje aplikacji desktopowych oraz webowych. Często bazują na językach HTML, CSS, JavaScript. Ich największą zaletą jest możliwość aktualizacji i synchronizacji przez Internet.

Tworzenie takiej aplikacji jest dość zawiłym procesem, często składają się na niego kroki takie jak:

- analiza i planowanie, czyli poznanie potrzeb użytkowników końcowych oraz wymagań biznesowych,

- projektowanie, czyli stworzenie koncepcji interfejsu, diagramów UML lub BPMN,

- tworzenie kodu, czyli programowanie i implementacja aplikacji,

- testowanie, czyli sprawdzenie jakości oprogramowania oraz poprawnie błędów.

1. **Opis sytuacji faktycznej**

Rynek systemów do zarządzania zasobami muzealnymi w Polsce jest obecnie dobrze rozwinięty. Funkcjonuje kilka kluczowych systemów wykorzystywanych w większych organizacjach. Są to miedzy innymi MUZA, Muzeo czy ArtSaas. Systemy te dają duże możliwości funkcjonalne, niestety często są bardzo kosztowne dla mniejszych instytucji. Dodatkowo ich obsługa jest bardziej skomplikowana i wymaga znacznych zasobów do wdrożenia. W efekcie mniejsze instytucje prywatne, galerie sztuki czy instytucje kościelne zostają na marginesie rynku technologicznego.

Aby lepiej zapoznać się z potrzebami tych instytucji przeprowadziliśmy wywiady z ich pracownikami, z których wynika że borykają się z brakiem dedykowanego systemu w przestępnej cenie. Dodatkowo złożoność istniejących już systemów nie jest dopasowana do specyfiki pracy mniejszych instytucji oraz wymaga długotrwałych szkoleń personelu. Z wywiadów wywnioskowaliśmy również, ze w większych systemach często ignorowana jest potrzeba pracy offline.

Na podstawie tych wywiadów stworzyliśmy diagram UML, który obrazuje potrzeby pracowników muzeum oraz administratora systemu (zdjęcie 1.1).

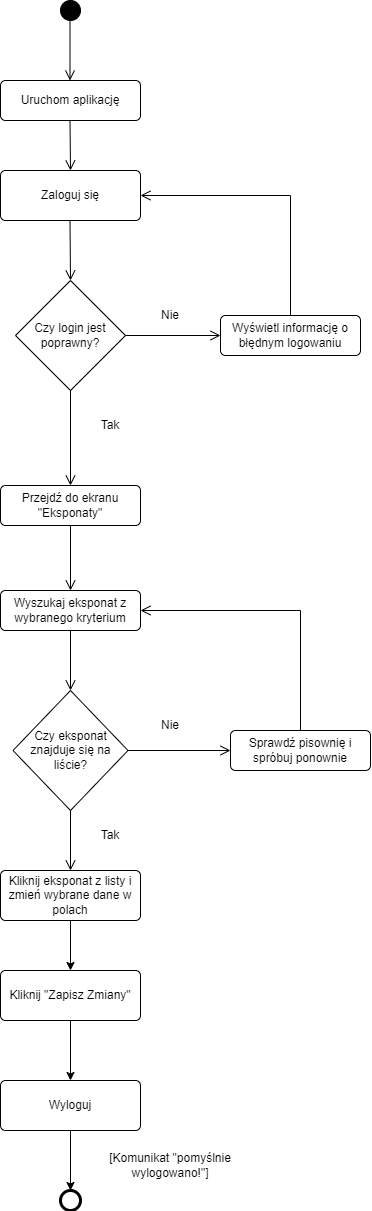
**Obraz zawierający zrzut ekranu, tekst, krąg

Opis wygenerowany automatycznie**

Zdjęcie 1.1 Diagram UML

Powyższy diagram ukazuje funkcje systemu, które wybrzmiewały najczęściej jako potrzeby, których brakuje mniejszym instytucją w pracy na co dzień. Wyróżniliśmy aktorów użytkownika i administratora. Głównymi potrzebami użytkowników okazały się być zarządzanie eksponatami, pracownicy muzeum zaznaczali potrzebę łatwości dodawania, edycji i usuwania eksponatów. Drugą główną potrzeba okazało się być zarządzanie wystawami i przypisanymi do nich eksponatami oraz generowanie raportów z wystaw i eksponatów. Pracownicy działów IT (administratorzy) zwracali uwagę na potrzebę podglądu historii działań użytkowników oraz możliwość tworzenia kopii zapasowej, co jest niezbędne w stałym nadzorowaniu zasobów. Jak również potrzebę zarządzania kontami użytkowników, ich rolami oraz hasłami.

Jedną z ważniejszych cech, które zgłaszali respondenci była łatwość obsługi programu. Z racji, że pracownicy muzeum to z reguły osoby w wieku 40+, interfejs musi być intuicyjny i prosty w obsłudze. Na tej podstawie stworzyliśmy diagram aktywności dla przypadku edycji eksponatu (zdjęcie 1.2).

****

Zdjęcie 1.2 Diagram aktywności edycji eksponatu.

Diagram przedstawia ścieżkę, którą musi przejść użytkownik aby edytować eksponat. Taki sposób przedstawienia funkcjonalności okazał się być bardzo przydatny podczas rozmów, prowadzonych z pracownikami mniejszych instytucji, jak również pomocny dla nas, w dokładniejszej analizie ich potrzeb.

1. **Badania własne / opis metod, technik i narzędzi badawczych / aparatura / oprogramowanie**

Visual Studio

Zdecydowaliśmy się na wykorzystanie Visual Studio z uwagi na framework WPF opracowany przez firmę Microsoft. Służy do tworzenia aplikacji desktopowych dla systemu Windows. Został wprowadzony jako cześć .NET Framework 3.0, pozwala na nowoczesne podejście do budowania interfejsów użytkownika z użyciem języka XAML, co z kolei pozwala na oddzielenie logiki systemu od warstwy wizualnej, dzięki czemu praca jest łatwiejsza i bardziej przejrzysta.

SQLite

SQLite to otwarto-źródłowy system zarządzania relacyjną bazą danych, która poprzez implementację języka C implementuje SQL. Nie wymaga oddzielnego serwera żeby działać. Biblioteka SQLite waży kilkaset kilobajtów, dzięki czemu jest szybka i dopasowana do zastosowań lokalnych. Dzięki temu ze działa jako biblioteka to baza jest dostępna bez potrzeby instalowania dodatkowego oprogramowania. Zapewnia też wsparcie dla transakcji ACID (Atomicity, Consistency, Isolation, Durability), przez co jest bezpieczna i spójna.

QuestPDF

QuestPDF to biblioteka otwarto-źródłowa, umożliwiająca dynamiczne generowanie i podgląd dokumentów PDF w nowoczesny sposób bez polegania na konwersji HTML. Pozwala definiować dokumenty w sposób deklaratywny za pomocą kodu C#. Dokumenty są budowane w formie hierarchicznej, oznacza to ze zachowują podział na kontenery, kolumny, wiersze, sekcje, co ułatwia ich tworzenie. QuestPDF jest responsywna, czyli silnik automatycznie dostosowuje treść do rozmiaru stron, obsługuje również różne elementy takie jak tabele, obrazy, wykresy itp.

Adobe Reader

Adobe Reader to oprogramowanie stworzone przez Adobe System. Pozwala na m. in. na przeglądanie i drukowanie dokumentów w formacje PDF. Firma Adobe opracowała również format PDF jako jeden z najpopularniejszych standardów wymiany dokumentów w postaci cyfrowej. Kluczowe funkcje tego programu to czytanie, otwieranie i przeglądanie plików PDF, drukowanie i podstawowe interakcje np. podpisywanie cyfrowe, dodawanie komentarzy, adnotacji i wiele innych.

draw.io

Draw.io to narzędzie do tworzenia diagramów, schematów i wykresów przepływu. Dostępne jako aplikacja webowa oraz desktopowa. Umożliwia wizualizację diagramów takich jak diagramy przepływu, UML, schematy procesu. System ten umożliwia integrację z chmurą Google Drive, GitHub, Dropbox, dzięki czemu praca w grupie staje się łatwiejsza. Oferuje wiele gotowych bibliotek symboli dla notacji takich jak UML czy BPMN. Gotowe diagramy można eksportować do formatów PNG, JPEG, PDF, SVG czy XML.

Figma/Uno platform

Figma to aplikacja pozwalająca zaprojektować dowolny pod kątem UX/UI interfejs graficzny dla każdego rodzaju urządzenia. Projektowanie odbywa się za pomocą strony internetowej gdzie działamy na frame’ach i kolejno dodajemy elementy interfejsu. Możemy skorzystać z asset’ów do wizualizacji efektu końcowego lub stworzyć własne, a na końcu możemy prototypować z wykorzystaniem np. aplikacji mobilnej Figma. Oprócz dużej, wciąż rosnącej ilości design kit’ów, ma możliwość instalowania wtyczek co znacznie ułatwia, i przyspiesza pracę z różnymi systemami. Dobrym przykładem jest wtyczka Uno Platform, która umożliwiła nam przeniesienie do XAML designu aplikacji.

Trello

Trello to narzędzie do zarządzania zadaniami. Pozwala na tworzenie tablic, podzielonych na listy i karty, co w przejrzysty sposób powalają śledzić postęp prac. Zadania mogą mieć przypisane osoby i terminy wykonania. Dużą dogodnością jest integracja z aplikacjami takimi jaki Google Drive, Slack czy Jira. Dzięki tej aplikacji łatwo zarządzaliśmy zadaniami i bieżącymi etapami projektu.

GitHub

GitHub to serwis wykorzystujący system kontroli wersji „Git”, przeznaczony do projektów programistycznych, gdzie wymagane jest ścisłe, przejrzyste zarządzanie plikami przez wiele użytkowników, z uwzględnieniem ich uprawnień dla zabezpieczenia danych przed niepowołanym dostępem. Jest to także „kopalnia wiedzy” dla laików programowania oraz hosting gotowych publicznych rozwiązań dla przeciętnego użytkownika szukającego rozwiązania swojego problemu. Obecnie serwis posiada aplikację mobilną dla podglądu i autoryzacji logowań oraz aplikację desktopową, która dodatkowo usprawnia kontrolę nad pobieraniem, wysyłaniem oraz podglądem plików projektów.

**C2. Efekty realizacji projektu**

Pierwszym oknem, które jest widoczne po uruchomieniu aplikacji jest okno logowania (zdjęcie 2) i rejestracji (zdjęcie 3.1). Pozwala na zalogowanie się lub założenie konta, które automatycznie ma przypisaną rolę „user”, czyli standardowy użytkownik. Role można zmienić logując się na konto z rolą „admin” w zakładce „Użytkownicy” Logowanie zarówno dla roli „admin” jak i „user” obywa się przez ten sam formularz. Podczas rejestracji otrzymujemy również powitalną wiadomość email (zdjęcie 3.2).

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

Zdjęcie 2. Okno logowania

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

Zdjęcie 3.1. Okno rejestracji

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

Zdjęcie 3.2. Email powitalny

Po lewej stronie okna, znajduje się menu nawigacyjnie, na zrzutach ekranu poniżej prezentujemy widok dla administratora. Ma on dostęp do wszystkich możliwości menu.

Zakładka „Eksponaty” (zdjęcie 4) pozwala na zarządzanie zbiorami muzeum. W tablicy prezentuje wszystkie zabytki istniejące w bazie danych, wraz z ich zdjęciem. Pasek wyszukiwania na górze okna, umożliwia łatwe i szybkie filtrowanie listy.

Po prawej stronie klikniecie przycisku „Dodaj” lub „Edytuj” wyświetla, grida poniżej z polami do dodania nowego eksponatu lub edycji już istniejącego, aktualnie zaznaczonego w dataGridzie. Edycja automatycznie wypełnia pola tj. nazwa, rok, kategoria, autor, pochodzenie, lokalizacja oraz zdjęcie, aktualnymi danymi, które można łatwo zmienić. Przycisk „Usuń” usuwa z bazy danych aktualnie zaznaczony rekord w tabeli eksponatów.

Elementy takie jak Kategoria, zostały o słownikowane w celu zachowania spójności kategoryzacji zabytków.

**Obraz zawierający tekst, elektronika, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie**

Zdjęcie 4. Okno zarządzania eksponatami

Zakładka „Wystawy” (zdjęcie 5.1) pozwala na zarządzanie wystawami i przypisanymi do nich eksponatami. Analogiczny pasek wyszukiwania wystaw, tak jak w zakładce „Eksponaty”, pozwala na łatwe filtrowanie rekordów w pierwszej tabeli, tabeli wystaw.

Przyciski „Dodaj wystawę” oraz „Edytuj wystawę”, powodują pojawienie się grida (zdjęcie 5.2) z polami umożliwiającymi dodanie lub edycje aktualnych danych. Podobnie jak w eksponatach, pola status wystawy i typ wystawy zostały o słownikowane w celu zachowania spójności danych. Dodanie nowej wystawy powoduje również wysyłkę email informującego o zaplanowaniu nowej wystawy (zdjęcie 5.3). Przycisk „Usuń wystawę” usuwa aktualnie zaznaczony rekord w tabeli wystaw.

Zakładka „Wystawy” umożliwia, również przypisanie eksponatu z bazy danych, występującego na danej wystawie w danym terminie. Wybór wystaw i eksponatów możliwy jest z listy rozwijalnej, która prezentuje dane z bazy danych. Po prawej stronie mamy również możliwość wyszukiwania po wystawie lub eksponacie.

**Obraz zawierający tekst, zrzut ekranu, Czcionka, oprogramowanie

Opis wygenerowany automatycznie**

Zdjęcie 5.1. Okno zarządzania wystawami widok podstawowy

**Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie**

Zdjęcie 5.2. Okno zarzadzania wystawami widok do dodawania lub edycji wystawy

Obraz zawierający tekst, Czcionka, zrzut ekranu, algebra

Opis wygenerowany automatycznie

Zdjęcie 5.3. Email informujący o zaplanowaniu nowej wystawy

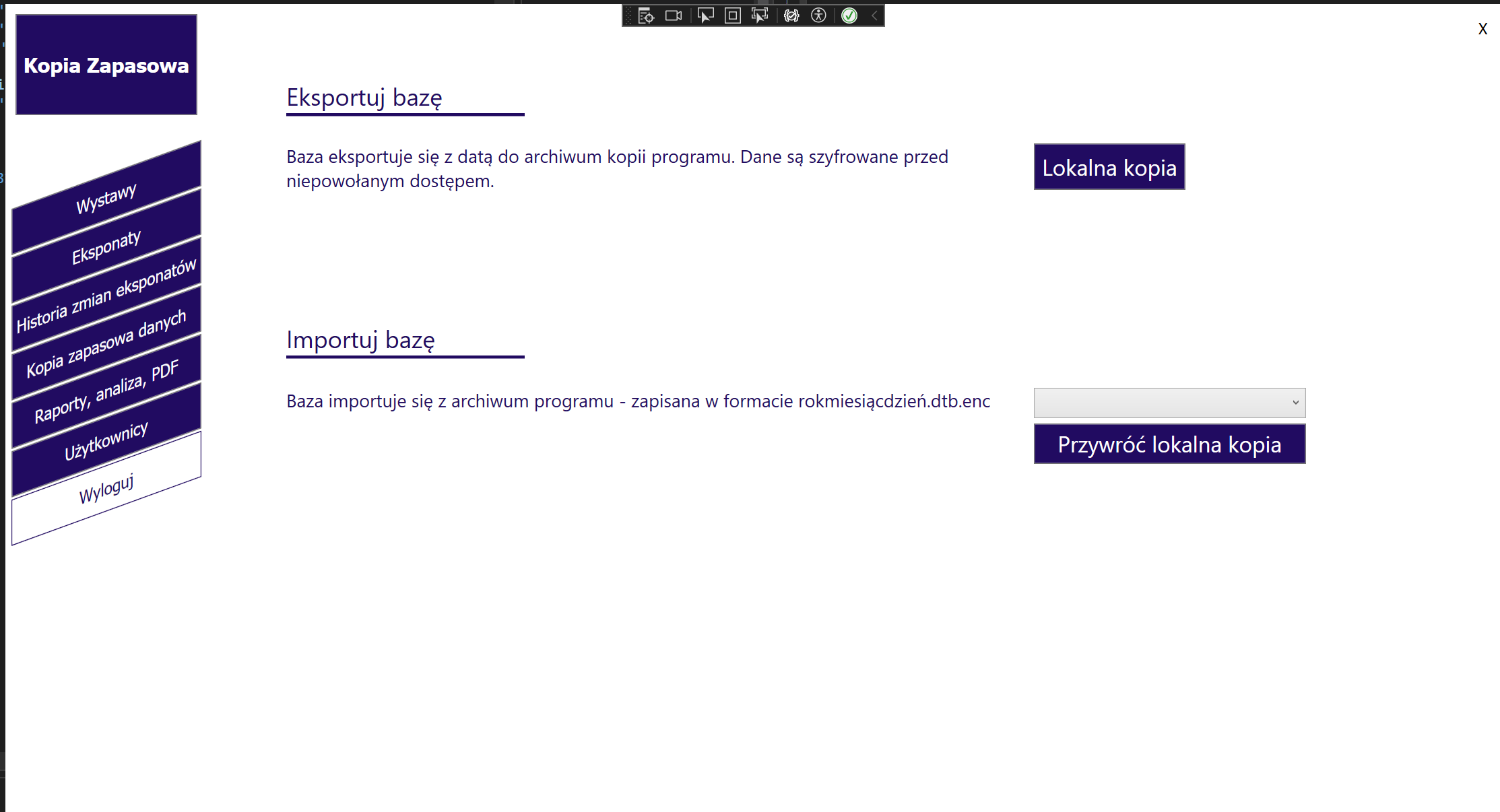
Zakładka „Historia zmian eksponatów” prezentuje tabele z aktualnymi zmianami oraz historią logowań do systemu.

**Obraz zawierający tekst, zrzut ekranu, numer, dokument

Opis wygenerowany automatycznie**

Zdjęcie 6. Okno historii zmian

Zakładka „Kopia zapasowa” pozwala na utworzenie kopii zapasowej lokalnie w specjalnie przeznaczonym do tego folderze oraz przywrócenie wybranej wersji kopii zapasowej.

****

Zdjęcie 7. Okno tworzenia i przywracania kopii zapasowej

Zakładka „Raporty, analiza, PDF” umożliwia pobranie raportu z wystaw lub eksponatów w formie PDF, co jest dostępne zarówno dla roli „admin” oraz „user”. Raport z historii zmian oraz z historii logować zarówno w PDF, jaki i CSV dostępny ejst tylko dla administratora systemu. Dwa ostatnie raporty posiadają również możliwość podglądu i wydruku w Adobe Reader.

**Obraz zawierający tekst, zrzut ekranu, Czcionka, design

Opis wygenerowany automatycznie**

Zdjęcie 8. Okno raportów, analiz i PDF

Poniżej wygenerowany przykładowy raport PDF z eksponatów (zdjęcie 9).

**Obraz zawierający tekst, zrzut ekranu, Ludzka twarz, Strona internetowa

Opis wygenerowany automatycznie**

Zdjęcie 9. Przykładowy raport z eksponatów

Zakładka „Użytkownicy” przeznaczona tylko dla roli „admin” pozwala na zarzadzanie użytkownikami. Pasek na górze umożliwia wyszukiwanie danego użytkownika, tabela poniżej wyświetla wszystkich użytkowników z bazy danych, ich role oraz zahaszowane hasła.

Przycisk „Dodaj” umożliwia administratorowi dodanie nowego użytkownika wraz z hasłem startowym, które jest od razu haszowane oraz odpowiednią rolą. Przycisk „Usuń” usuwa rekord aktualnie zaznaczony w tabeli. Przycisk „Wyczyść” czyści pola email, hasło oraz ustawia role na „user”.

W tym miejscu możliwe jest też zmiana hasła dla danego użytkownika lub zmiana roli. Aby tego dokonać należy wybrać użytkownika z tabeli. Wtedy pola email, hasło oraz rola wypełnią się jego danymi, a przycisk „Dodaj” zmieni się na „Zmień” co pozwoli na edycję danych i zapisanie ich.

**Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie**

Zdjęcie 10. Okno zarządzania użytkownikami

Przycisk „Wyloguj” służy do wylogowania się z konta i przenosi do ekranu logowania.

**C3. Użyteczność projektu**

System do zarządzania zbiorami muzeum jest przeznaczony dla pracowników instytucji kulturowych. Został stworzony w celu usprawnienia pracy pracowników muzeum, poprawy nadzorowania zabiorów oraz ich konserwacje, poprawienia dostępu do informacji o eksponatach (ich lokalizacji, historii, uczestnictwa w wystawach). Usprawnienia raportowania i zbierania danych dotyczących wystaw i eksponatów w celu planowania przyszłych działań.

**C4. Autoewaluacja zespołu projektowego**

**Autor: Oliwia Głodek**

Tworząc system zarządzania zbiorami muzeum mogę stwierdzić, że moje umiejętności programowania w języku C# oraz wykorzystania WPF znacznie się rozwinęły. Przed rozpoczęciem projektu posiadałam podstawowe umiejętności pracy w języku C#, które wraz z rozwojem oprogramowania się poszerzały. Wzrosły również moje umiejętności pracy w grupie, dzielenia się zadaniami i wymiany informacji.

Moim zadaniem było napisanie funkcjonalności dodawania, edytowania, usuwania eksponatów i wystaw, stworzenie funkcji logowania i rejestracji oraz zarządzania użytkownikami. Pracując nad tymi elementami musiałam wykorzystać wiedze z zakresu baz danych oraz komunikacji aplikacji z bazą. Doceniłam również testowanie aplikacji i zrozumiałam jak wielki ma wpływ na prawidłowe funkcjonowanie programu. Ciągłe testowanie i pojawianie się nowych błędów, nauczyło mnie jak udoskonalać kod oraz wypracowywać metody, które pomogą w optymalizacji i komunikacji z bazą danych.

Podsumowując praca nad systemem dla muzeum była cennym doświadczeniem i znacznie poprawiła moje kompetencje programistyczne oraz kompetencje miękkie pracy w grupie. Zdobyłam nowe umiejętności z zakresu C#, WPF oraz baz danych, jak i komunikacji z zespołem.

**Autor: Emil Gielek**

W procesie tworzenia naszej aplikacji do zarządzania zbiorami muzeum kluczową dla mnie kwestią było zrozumienie jak funkcjonuje Software Development Lifecycle, w praktycznym kontekście naszego projektu. Przed startem projektu znałem podstawowe zagadnienia z języka programowania C# oraz Figmy, jednak dopiero obecny projekt pozwolił mi to wszystko połączyć i przetworzyć w logiczną całość. W końcu zrozumiałem istotność synergii nie tylko w zespole, ale w naszym oprogramowaniu i jak je wykorzystać.

Moją rolą było utworzenie prostego designu aplikacji, optymalizacji zużycia zasobów aplikacji, wdrożenia oraz napisania szablonów z wykorzystaniem QuestPDF i na koniec stworzenie prostego systemu tworzenia kopii zapasowych dla naszej aplikacji. Aby tworzyć prototypy ekranów, musiałem najpierw poznać podstawy WPF i związanego z nim XAML. Dopiero później zastosować nowo poznaną wiedzę z wtyczki w Figmie – Uno Platform. Trudnym doświadczeniem było wdrażanie biblioteki QuestPDF. Dla naszej aplikacji musiałem zagłębić się w jaki sposób przetwarzane są dane aby wygenerować plik PDF. Kwestią wartą zapamiętania dla mnie był brak kompatybilności części biblioteki z nowszą wersją .NET co uniemożliwiało mi testowanie wyglądu formularzy na żywo. Takie doświadczenie zwróciło szczególna moją uwagę na istotność testowania i kontroli w procesie budowania aplikacji w zespole.

**C5. Wykorzystane materiały i bibliografia związana z realizacją projektu**

1. Dokumentacja jezyka C# <https://learn.microsoft.com/pl-pl/dotnet/csharp/>
2. Przewodnik dotyczący aplikacji klasycznych (WPF .NET) <https://learn.microsoft.com/pl-pl/dotnet/desktop/wpf/overview/?view=netdesktop-9.0>
3. „C# 2005 Wprowadzenie” Jesse Liberty, Brian MacDonald, Helion
4. Krótkie nagranie YouTube: Czym jest szyfrowanie AES i jak działa?

<https://www.youtube.com/watch?v=DiXkP4_E348>

1. Film YouTube kanału Mdemy z tworzenia systemu magazynowego

<https://www.youtube.com/watch?v=x4sw_22gsSc&list=PLcDvtJ2MXvhzBo1gbNJvh9IORAJ6gzn4X&index=42>

1. Filmy na YouTube związane z prostym zastosowaniem QuestPDF :

<https://www.youtube.com/watch?v=T89A_7dz1P8> – kanał: Claudio Bernasconi

<https://www.youtube.com/watch?v=_M0IgtGWnvE> – kanał: Nick Chapsas

1. Dokumentacja na stronie QuestPDF

<https://www.questpdf.com/companion/usage.html> - wprowadzenie

<https://www.questpdf.com/getting-started.html> - użycie

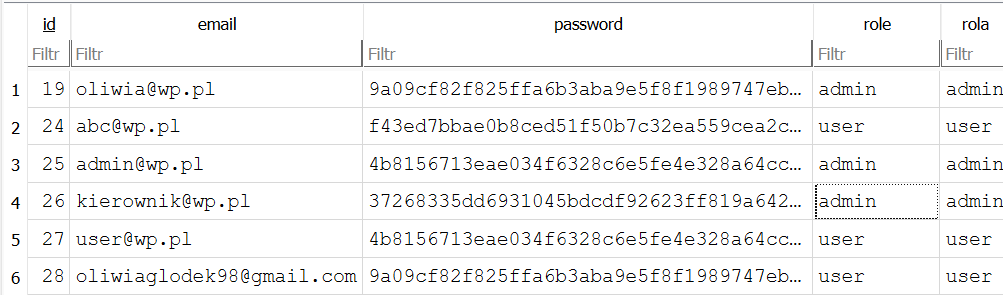
1. Dokumentacja o szyfrowaniu AES:

<https://learn.microsoft.com/pl-pl/dotnet/api/system.security.cryptography.aes?view=net-8.0>

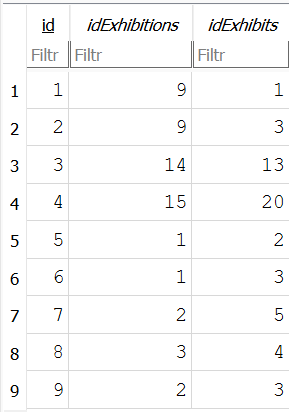
1. Artykuł o tworzeniu diagramów UML:

<https://it-consulting.pl/2022/04/03/diagram-przypadkow-uzycia/>

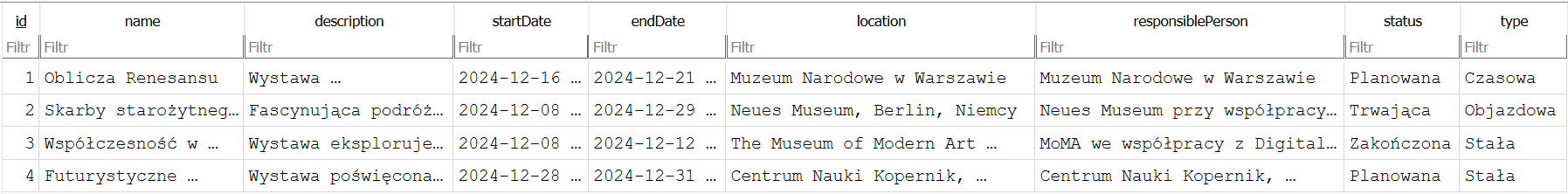
**C6. Spis załączników**

****

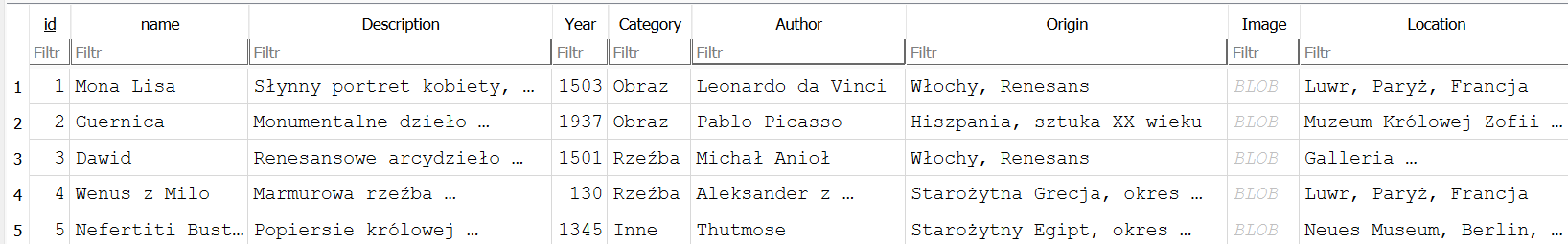
Zdjęcie 11.1 Dane w tabeli users



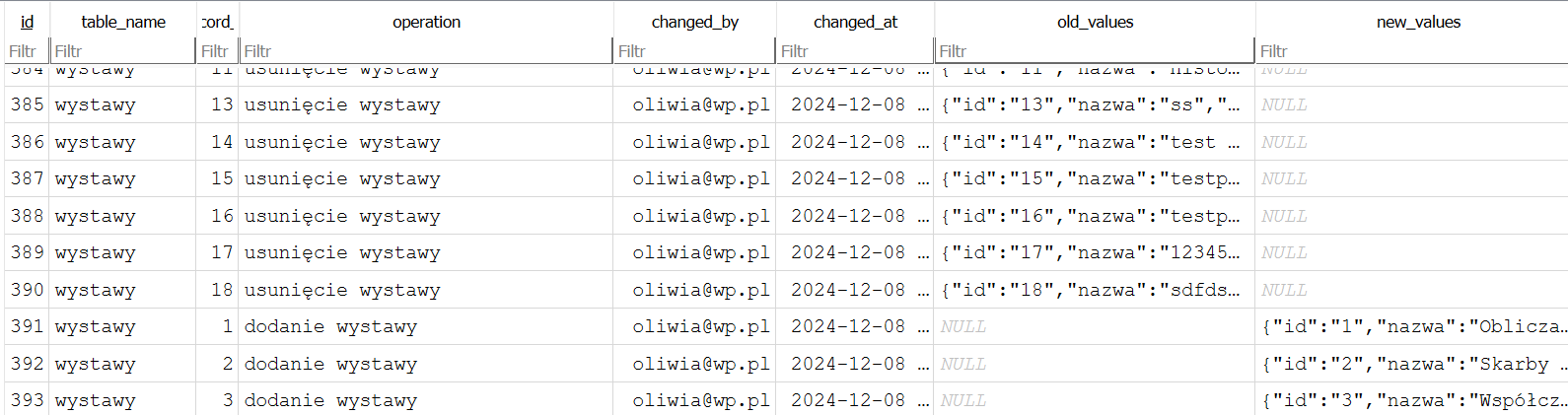
Zdjęcie 11.2. Dane w tabeli łączącej eksponaty z wystawami



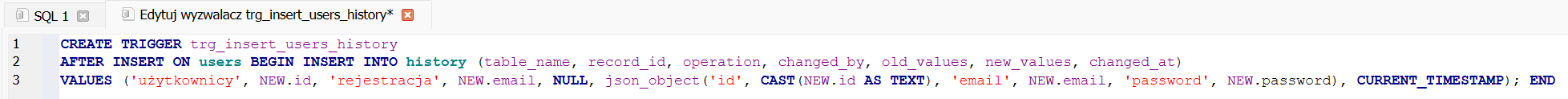
Zdjęcie 11.3. Dane w tabeli wystawy



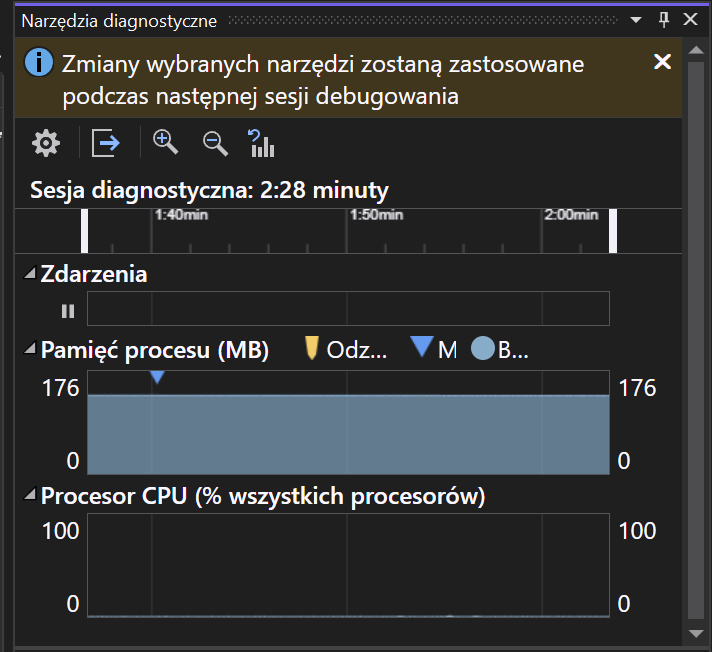
Zdjęcie 11.4. Dane w tabeli eksponaty



Zdjęcie 11.5. Dane w tabeli historia



Zdjęcie 12. Reguła wykorzystywana na potrzeby tworzenia historii

****

Zdjęcie 13. Diagnostyka zużycia pamięci