

Programowanie obiektowe (Java) - projekt

Grupa:
2ID12A

Zespół realizujący:
Wiktor Gozdek

Temat projektu:

Przeglądarka oraz edytor danych EXIF w plikach graficznych
(multimedialnych)

1. Opis projektu:

Celem wybranego tematu projektu było napisanie aplikacji posiadającej graficzny interfejs użytkownika w języku Java, która umożliwi przeglądanie i edytowanie metadanych EXIF plików graficznych. Aplikacja powinna pozwalać użytkownikowi na wybór pliku do odczytu i zapisu.

Do implementacji wyżej opisanej aplikacji wykorzystałem bibliotekę Swing języka Java w celu utworzenia interfejsu graficznego użytkownika oraz do wykonywania operacji na metadanych EXIF plików skorzystałem z biblioteki Apache Commons Imaging. Dodatkowo do utworzenia testów jednostkowych użyłem frameworka JUnit. Cały projekt został utworzony w środowisku programistycznym IntelliJ IDEA jako projekt Maven. Biblioteki Apache Commons Imaging oraz JUnit zostały dodane do projektu w formie zależności poprzez odpowiednio skonfigurowany plik pom.xml.

Biblioteka Apache Commons Imaging to biblioteka w całości napisana w Javie, która pozwala na odczyt i zapis różnych formatów obrazów oraz szybką analizę informacji o zdjęciu w tym metadanych. W przypadku przetwarzania metadanych wspiera jedynie obrazy w formacie JPEG.

2. Funkcje programu:

Program umożliwia:

- wybór pliku do odczytu i zapisu metadanych EXIF,
- odczytywanie, usuwanie i zapisywanie istniejących metadanych EXIF w wybranym pliku, przy czym zapis dotyczy tylko niektórych znaczników będących w formacie ASCII,
- wyświetlanie odczytanych metadanych i ich edycji za pomocą interfejsu graficznego.

3. Sposób uruchomienia oraz obsługa programu:

Program można uruchomić poprzez skompilowanie klasy EXIFEditorApp w której znajduje się funkcja main() stanowiąca główny punkt wejściowy programu. Inną możliwością jest skorzystanie z wygenerowanego pliku JAR o nazwie PJ_projekt.jar znajdującego się w folderze out/artifacts/PJ_projekt_jar/.

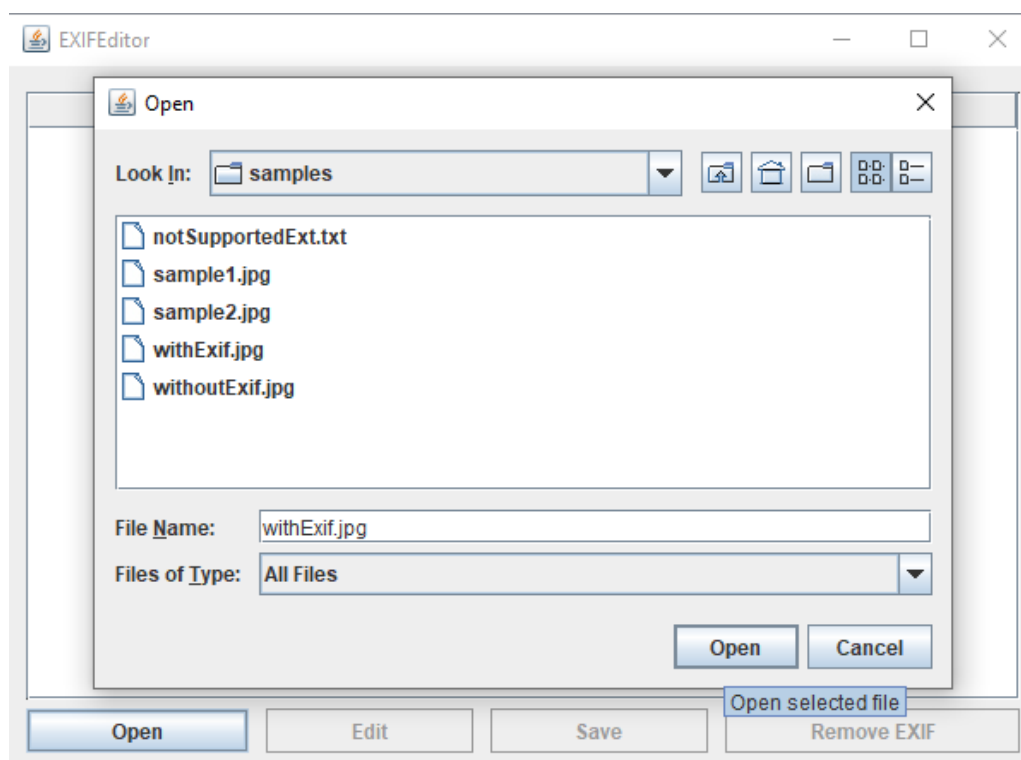
Po uruchomieniu powinno ukazać się niewielkie okno składające się z głównego panelu do wyświetlania metadanych plików w formie tabeli oraz z następujących przycisków:

- Open – uruchamia okienko wyboru pliku z którego mają zostać odczytane metadane,
- Edit – przycisk dostępny po dokonaniu wyboru pliku. Pozwala przejść w tryb edycji w którym można zmieniać wartości w tabeli,
- Save – po dokonaniu modyfikacji możliwe jest zapisanie wprowadzonych zmian za pomocą tego przycisku. Po kliknięciu uruchamia okno wyboru pozwalające określić ścieżkę i nazwę dla pliku wynikowego,
- Remove EXIF – umożliwia usunięcie wszystkich metadanych EXIF z pliku i zapisu tak powstałych danych do pliku wynikowego określonego w oknie wyboru pliku.

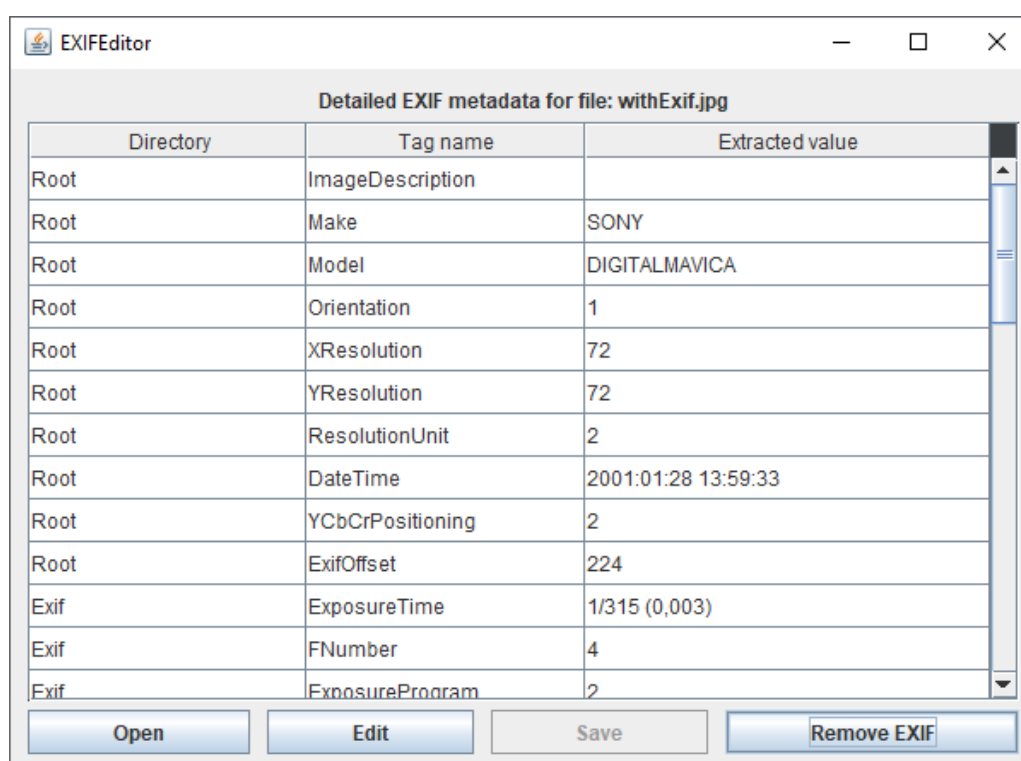
W górnej części okna wyświetlana jest informacja o nazwie aktualnie otwartego pliku.

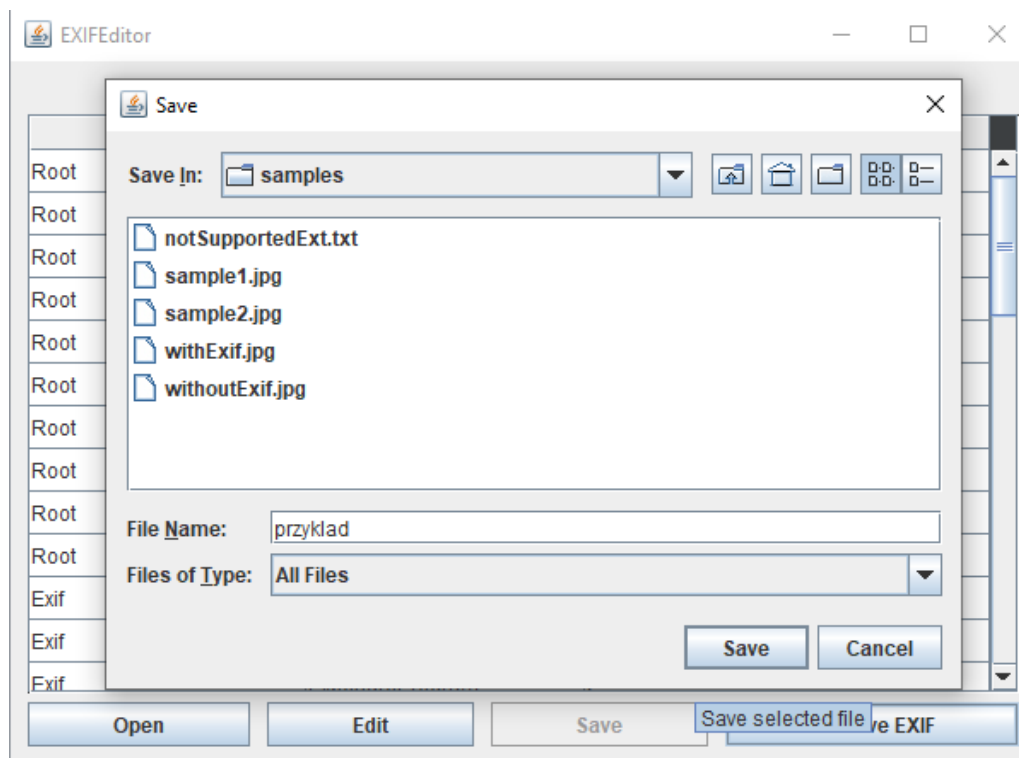
Przykładowe obrazy dla aplikacji znajdują się w katalogu /src/test/samples/.

Przykładowa obsługa programu w celu wyświetlenia i usunięcia metadanych EXIF z wybranego pliku:

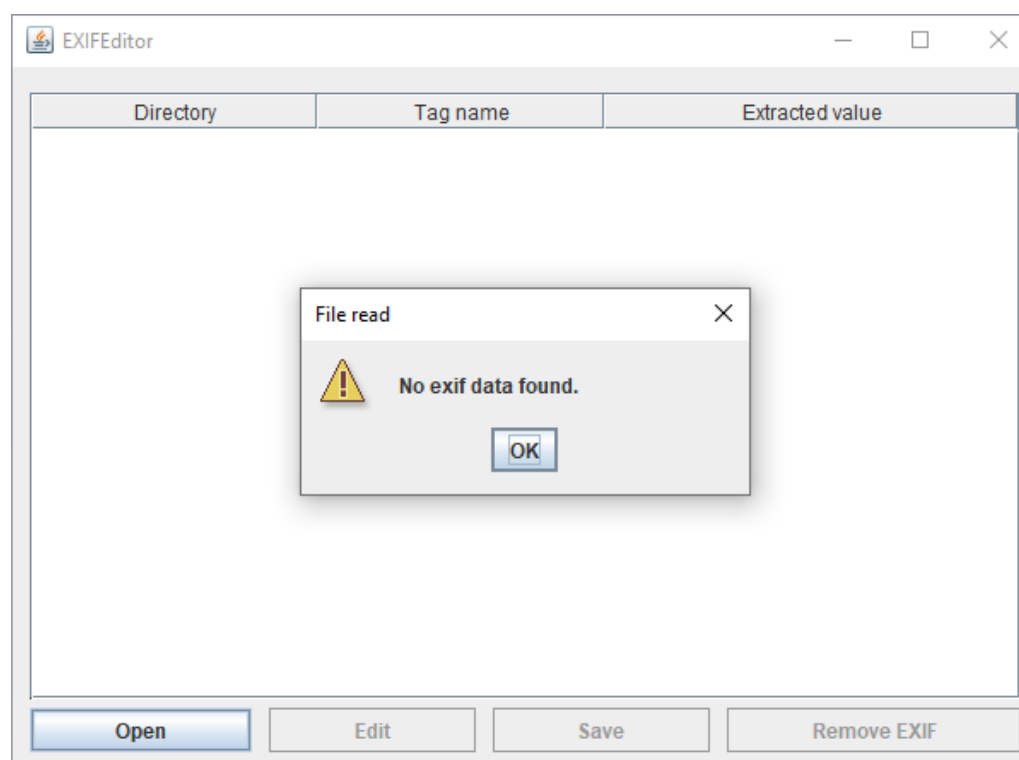


Wybór pliku źródłowego po uruchomieniu programu.





Wybór ścieżki i nazwy dla pliku wynikowego po usunięciu metadanych.



Po próbie ponownego odczytu pliku z usuniętymi metadany wyświetlony zostanie komunikat o braku metadanych exif.

4. Wykorzystane klasy, metody, funkcje:

Główne okno programu zostało zaimplementowane jako klasa EXIFEditorGUI, która dziedziczy pola i metody klasy JFrame z biblioteki Swing. Klasa głównego okna programu posiada implementacje metod pozwalających na obsługę plików takie jak:

- readAndLoadMetadataToTable – służy do wczytania i wyświetlenia wartości metadanych odczytanych z wybranego pliku do tabeli,
- writeMetadataFromTableToFile – służy do pobrania zawartości pól z tabeli i zapisu ich w metadanych pliku wynikowego,
- removeExifMetadataFromSrcFile – służy do usunięcia metadanych z otwartego pliku źródłowego i zapisu tak zmodyfikowanych danych w pliku wynikowym,
- chooseDestFile – służy do określenia ścieżki i nazwy dla plików wynikowych.

Okno aplikacji wykorzystuje również klasy paneli JPanel i JScrollPane do odpowiedniego rozmieszczenia innych komponentów takich jak przyciski (JButton), dla których zostały zdefiniowane odpowiednie metody obsługujące zdarzenia czy etykiety (JLabel). Głównym elementem okna jest tabela wyświetlająca metadane, która jest obiektem klasy JTable. Większość z wymienionych komponentów jest konfigurowana w metodzie initComponents wywoływanej w konstruktorze klasy okna głównego.

Za przetwarzanie metadanych EXIF odpowiada klasa ExifEditor posiadająca w swojej definicji trzy funkcje:

- readEXIFDataFromFile – pozwala na odczytanie zawartości metadanych pliku. Do ich odczytu korzysta z klas ImageMetadata i JpegImageMetadata. Dostęp do metadanych EXIF jest dostępny przy wywołaniu metody getExif() dla obiektu klasy JpegImageMetadata, która zwraca je jako TiffImageMetadata. Następnie możliwy jest już dostęp do poszczególnych katalogów, znaczników i ich wartości z użyciem klas TiffDirectory i TiffField.
- writeEXIFDataToFile – umożliwia zapis podanych jako argument wartości w odpowiednich polach metadanych pliku. W tym celu z pliku źródłowego pobierany jest zbiór wyjściowy metadanych w formie obiektu klasy TiffOutputSet. Z tej klasy możliwy jest już dostęp do poszczególnych katalogów i znaczników. W celu zmodyfikowania wartości znacznika wymagane jest jego usunięcie, a następnie dodanie go w tej samej formie, ale z inną wartością. Zmodyfikowany zbiór metadanych jest zapisywany do pliku z pomocą obiektu klasy ExifRewriter biblioteki Apache Commons Imaging.
- removeEXIFDataFromFile – służy do usunięcia metadanych EXIF z pliku. Podobnie jak w przypadku funkcji zapisu metadanych korzysta ona z obiektu klasy ExifRewriter i metody removeExifMetadata.

Więcej informacji na temat pozostałych elementów aplikacji znajduje się w wygenerowanej dokumentacji JavaDoc w katalogu doc/JavaDoc.

5. Ilość pracy włożonej przy tworzeniu projektu:

Projekt został w całości napisany i utworzony przez Wiktora Gozdka. W celu utworzenia projektu wymogiem było szersze zapoznanie się z bibliotekami Swing, Apache Imaging Commons oraz JUnit tj. klasami, metodami tych klas oraz w przypadku biblioteki Swing także obsługą zdarzeń.