

Projekt: Binary file handling tool



**Politechnika
Śląska**

Spis treści

Wstęp	2
Założenia	3
Cel projektu	4
Wymagania funkcjonalne	5
Wymagania niefunkcjonalne	8
Diagram sekwencji	8

Wstęp

Przeglądanie zawartości plików binarnych jest zazwyczaj kłopotliwe i wymaga użycia narzędzi programowych, które obsługują dany format pliku. Pomocne może być opracowanie narzędzia do przeglądania takich plików o zdefiniowanym formacie z opcją eksportowania ich do pliku tekstowego.

Celem tego dokumentu jest przedstawienie założeń funkcjonalnych, нефункциональных oraz diagramu sekwencji omawianego narzędzia. Pozwolą one na bezpośrednie zrozumienie budowy oraz sposobu działania programu przez klienta.

autorzy:

Marcin Przybylski

Marcin Barczyk

Mateusz Major

Hubert Szymski

Maciej Walkowiak

wersja 2

Teleinformatyka semestr IV, grupa TI2

prowadzący: Dr. inż Jakub Nalepa

Wymagania

Wymagania funkcjonalne - zbiór

Poniższa tabela przedstawia zbiór wymagań funkcjonalnych opracowywanego przez nas narzędzia. Znajdziemy w niej numer id wymagania, nazwę oraz krótki opis, którego rozszerzenie znajdziemy w dalszej części dokumentu.

ID	Nazwa	Opis
REQ01	Możliwość wyboru struktury pliku wejściowego	Struktura pliku binarnego powinna być określona przez użytkownika - narzędzie powinno umożliwiać definiowanie zarówno struktur liniowych, jak i iteracyjnych (liczba iteracji zapisana w pliku).
REQ02	Lista dostępnych formatów i plików wraz z historią ich użycia	Lista kilku ostatnio używanych formatów powinna być dostępna w GUI. To samo dotyczy ostatniego folderu plików.
REQ03	Wybór pliku do przetworzenia	Okno programu pozwalające przeglądać i zaznaczać pliki przeznaczone do otwarcia.
REQ04	Otwarcie pliku	Odczytuje dane zapisane w pliku binarnym.

REQ05	Prezentacja danych jako tekst oraz przechowywanie danych	Dane odczytane z pliku powinny być przedstawione jako tekst, a dane mogą być przechowywane w pliku tekstowym (całkowicie lub częściowo).
REQ06	Konwersja pliku binarnego na tekstowy	Otwarty plik binarny może być zapisywany do formatów tekstowych (.txt, .doc, .rft)
REQ07	Obsługa przetwarzania wsadowego	Program ma możliwość konwersji całego zestawu danych.
REQ08	Zapis pliku tekstowego	Tworzy na dysku plik o zadanej formie, zawierający przekonwertowane wcześniej dane.

tabela nr 1 - wymagania funkcjonalne (zbiór).

Wymagania funkcjonalne - charakterystyka

Poniższe tabele mają na celu dokładne scharakteryzowanie kluczowych wymagań funkcjonalnych, są to kolejno:

- wybór pliku do przetworzenia (REQ03),
 - otwarcie pliku (REQ04),
 - konwersja pliku binarnego na tekstowy (REQ06),
 - obsługa przetwarzania wsadowego (REQ07),
 - zapis pliku tekstowego (REQ08).
-

Nazwa funkcji	Wybór pliku do przetworzenia (REQ03)
Opis	Wyświetla w nowym oknie domyślną lokalizację, umożliwiając wskazanie pliku binarnego, posiada również opcję wyświetlenia lokalizacji ostatnio użytych przez użytkownika.
Dane wejściowe	ostatnio odwiedzane lokalizacje
Źródło danych wej.	vector<string>

Wynik	zwrócenie typu string z lokalizacją pliku przeznaczonego do odczytu
Warunek wstępny	plik istnieje i nie jest uszkodzony
Warunek końcowy	zwrócony string z lokalizacją pliku do przetworzenia
Efekty uboczne	dodanie nowej lokalizacji do wektora ostatnio odwiedzanych folderów
Powód	

tabela nr 2.

Nazwa funkcji	Otwarcie pliku (REQ04)
Opis	Odczytuje plik binarny do pamięci programu.
Dane wejściowe	ciąg znaków bitowych
Źródło danych wej.	plik binarny
Wynik	odczytanie ciągu znaków bitowych do wektora typu string
Warunek wstępny	plik istnieje i jest plikiem binarnym.
Warunek końcowy	utworzony vector.
Efekty uboczne	zajęcie dodatkowego miejsca w pamięci komputera.
Powód	

tabela nr 3.

Nazwa funkcji	Konwersja pliku binarnego na tekstowy (REQ06)
Opis	Konwertuje plik binarny na plik tekstowy w formacie wybranym przez użytkownika.
Dane wejściowe	Ciąg znaków bitowych
Źródło danych wej.	Ciąg binarny odczytany z pliku
Wynik	Plik tekstowy z zawartością
Warunek wstępny	Istniejący plik binarny oraz wybrany format do zapisu
Warunek końcowy	Koniec pliku
Efekty uboczne	Zajęcie pamięci oraz czas trwania
Powód	

tabela nr 4.

Nazwa funkcji	Obsługa przetwarzania wsadowego (REQ07)
Opis	Przez polecenie użytkownika wybiera konwersję zestawu plików lub jednego.
Dane wejściowe	Lista wyboru
Źródło danych wej.	Vector <string>
Wynik	Konwersja wybrana przez użytkownika
Warunek wstępny	Wybrano opcję
Warunek końcowy	Zwrócono string
Efekty uboczne	Zajętość pamięci, możliwość złego wyboru
Powód	

tabela nr 5.

Nazwa funkcji	Zapis pliku tekstowego (REQ08)
Opis	Zapisuje tekst do wcześniej wybranego formatu pliku.
Dane wejściowe	Format pliku, lokalizacja zapisu
Źródło danych wej.	Zmienna typu string
Wynik	Zapis pliku
Warunek wstępny	Lokalizacja jest dostępna, plik o podanej nazwie nie istnieje przed zapisem
Warunek końcowy	Zapisany plik
Efekty uboczne	Dodanie formatu do ostatnio używanych
Powód	

tabela nr 6.

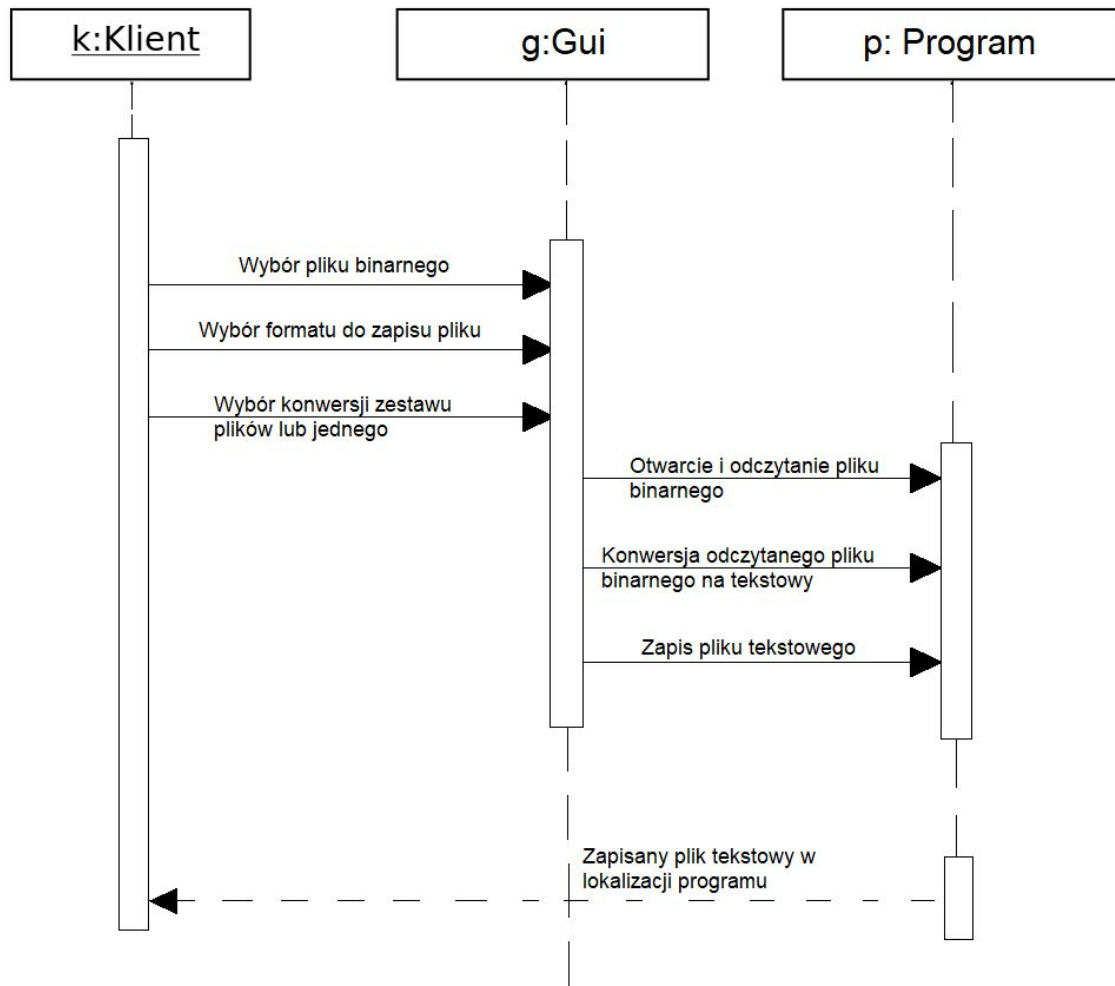
Wymagania niefunkcjonalne

Cecha	Metryka
Szybkość	Jeden plik binarny na 10 sekund. Czas reakcji na sygnał użytkownika: 0.1 sekundy.
Zasoby pamięciowe	Narzędzie podczas odczytu dużego pliku tekstowego (10mb) zużywa 500mb pamięci RAM, jest to więc wartość minimalna, która zapewnia płynną pracę z programem.
Zasoby dyskowe	Po instalacji narzędzie zajmuje nie więcej niż 50mb pamięci dyskowej.
Łatwość użycia	Czas przeszkolenia ok. 30 minut. Liczba stron systemu pomocy: 5.
Niezawodność	Narzędzie zostało przekazane do testów stu pracownikom, spośród których 4% dokonało błędnego wyboru opcji, która skończyła się awarią oprogramowania.
Odporność na błędy	Czas restartu po awarii ok. 10 sekund.
Przenośność	Liczba systemów docelowych: jeden.

tabela nr 7.

Diagram sekwencji działania programu

Diagram przedstawia w uproszczeniu zarys działania programu. Pierwsza część pokazuje jakie opcje do wyboru ma Klient (użytkownik programu). Natomiast druga prezentuje jakie funkcje musi wykonać program po dokonaniu wyboru przez Klienta. Po wykonaniu wszystkich operacji i zapisaniu pliku tekstowego użytkownik dostaje gotowy plik zapisany w lokalizacji programu.



Rys. 1 Diagram sekwencji działania programu