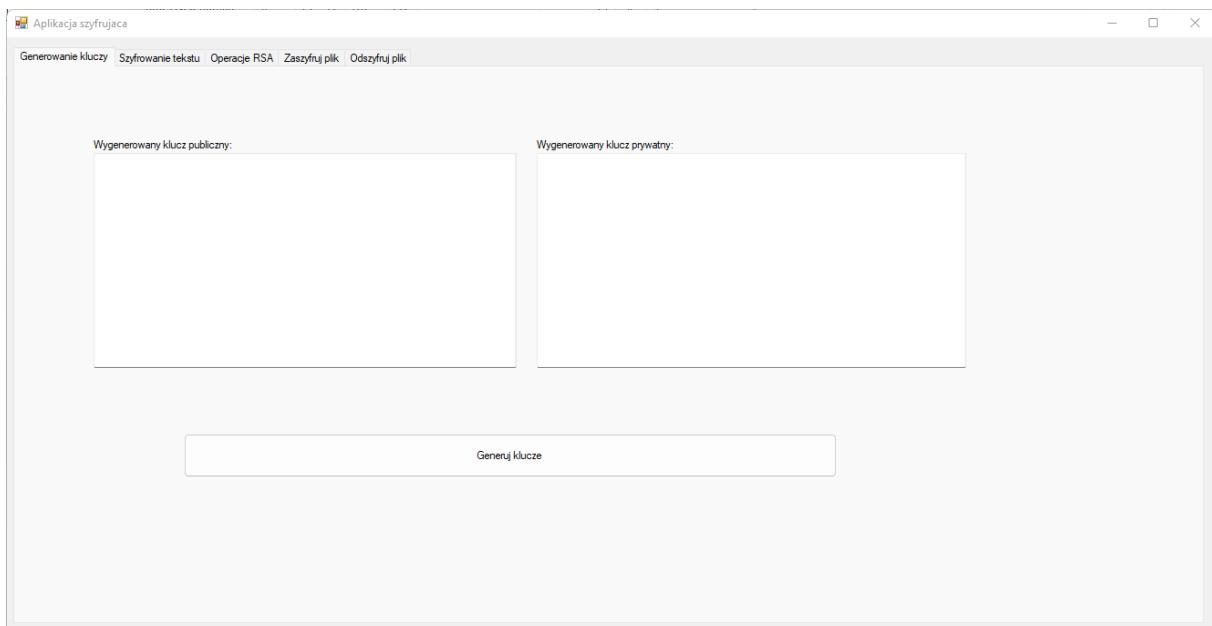


Akademia Nauk Stosowanych w Nowym Sączu  
Wydział Nauk Inżynierijnych  
czesawodność systemów informatycznych –2022/2023

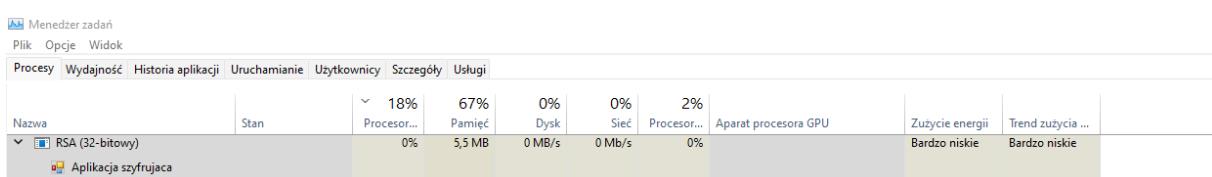
**Imię i nazwisko:** Filip Rzepiela      **Data:** 06.12.2022  
**Grupa:** P2

# **Testy manualne aplikacji do szyfrowania oraz deszyfrowania tekstu oraz plików za pomocą algorytmu RSA (aplikacja opracowana na potrzeby zaliczenia przedmiotu Kryptografia i teoria kodów)**

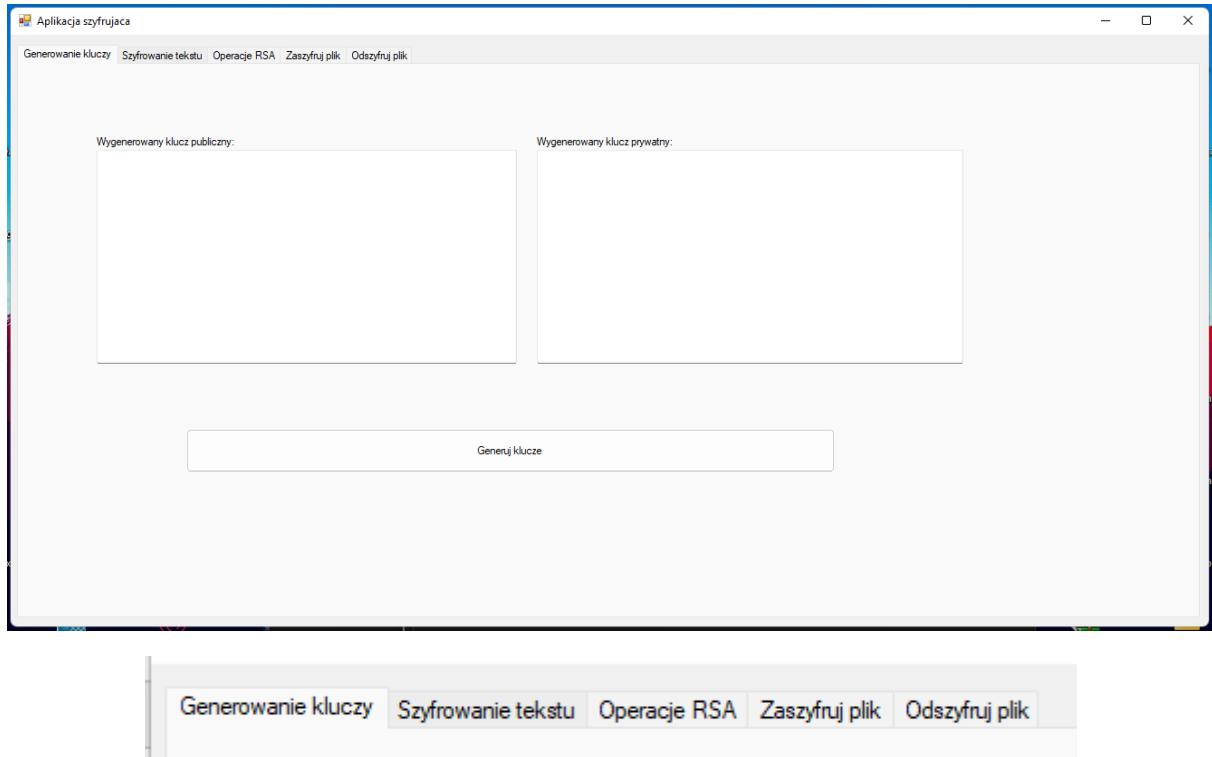
## Zakładka pierwsza – generowanie kluczy publicznych



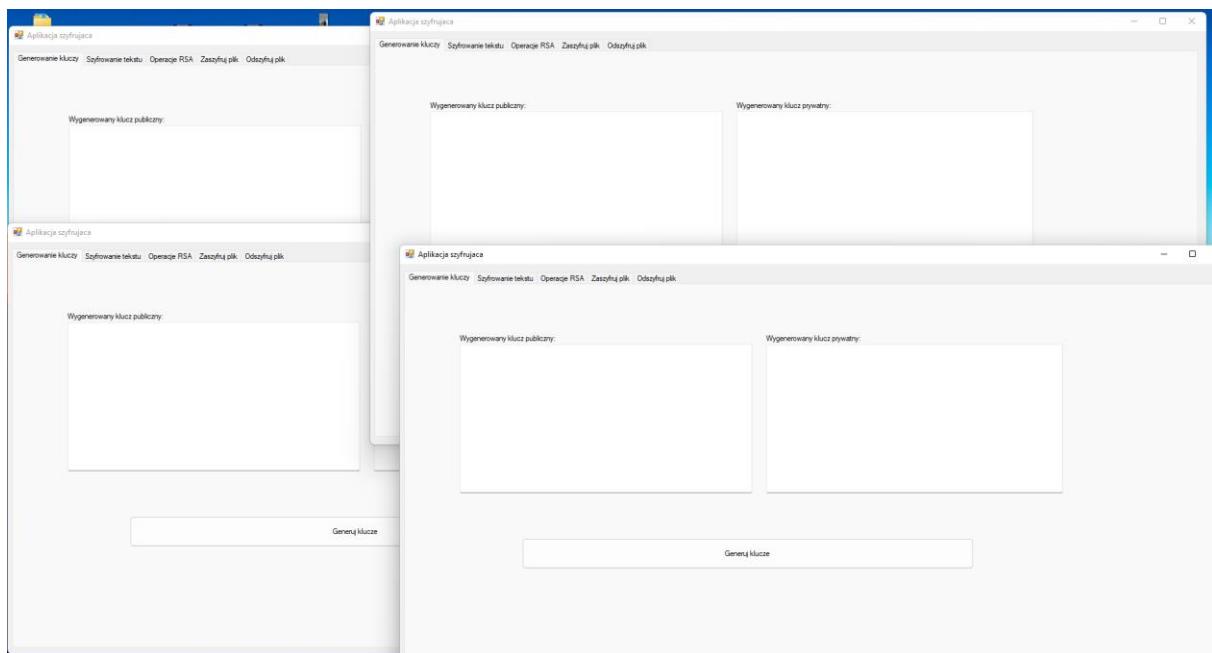
#### Zużycie zasobów w czasie bezczynności



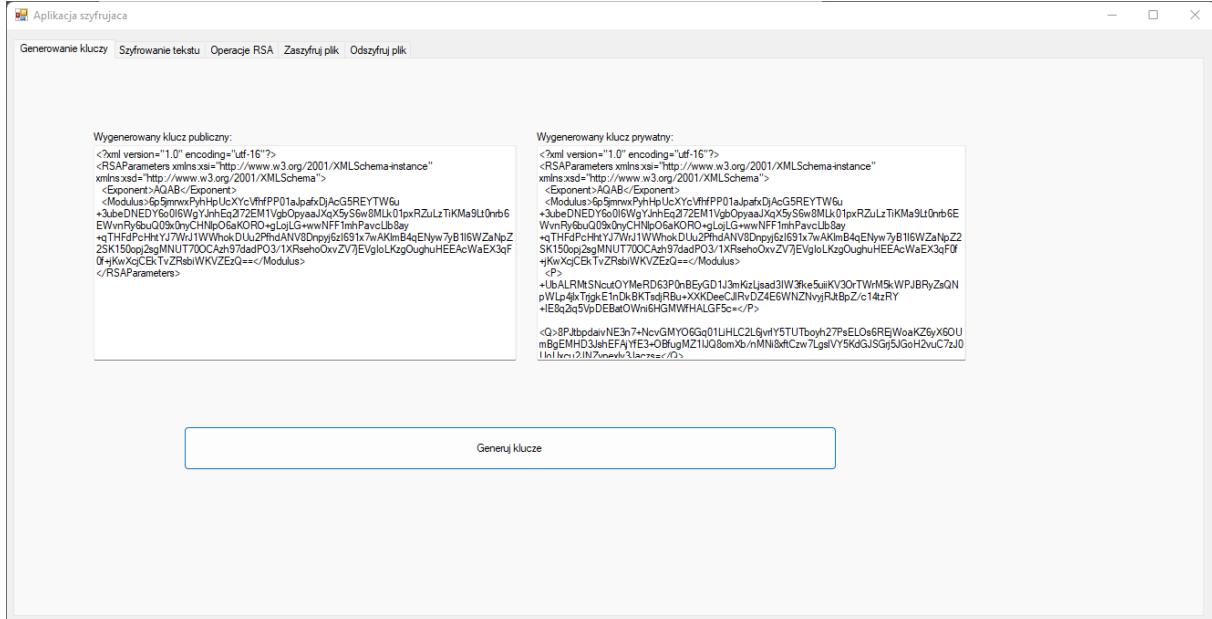
Po najechaniu na dowolną kartę, karta nabiera szarawy kolor



Możliwe jest uruchomienie kilku instancji programu



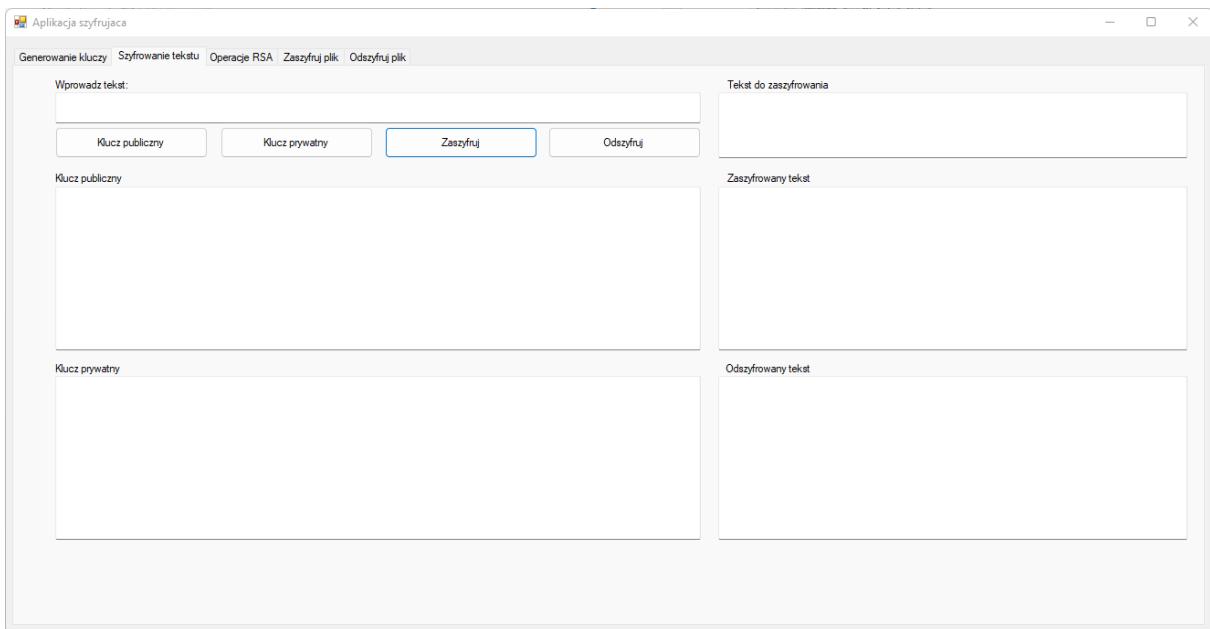
## Test generowania kluczy



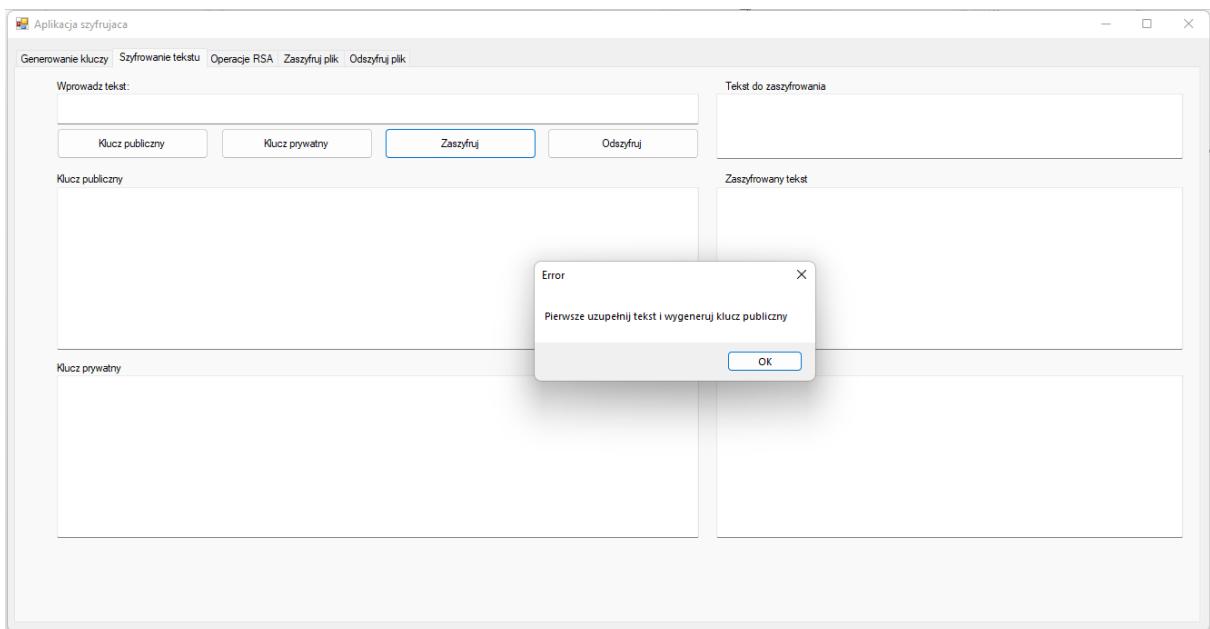
Jak możemy zauważyc, po wciśnięciu generuj klucze otrzymujemy wygenerowany klucz prywatny oraz publiczny.

# Test szyfrowania tekstu

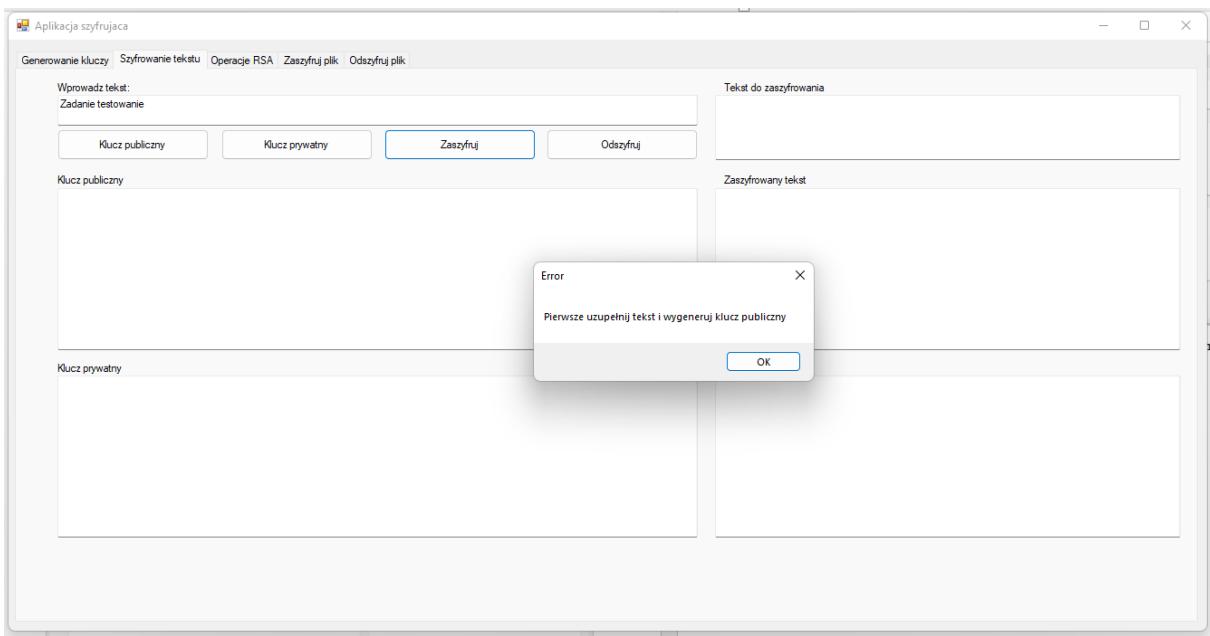
## Okno szyfrowania tekstu



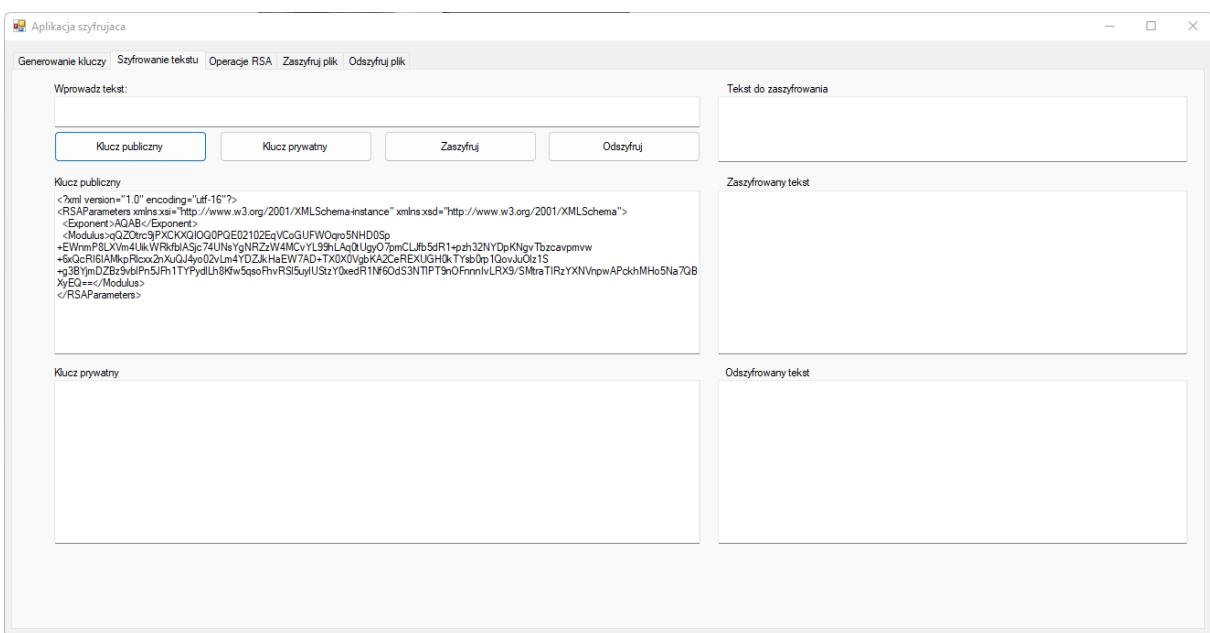
Próba zaszyfrowania przy pustych polach:



Aplikacja wyświetla powiadomienie, że pole tekstowe oraz klucz publiczny nie mogą być puste. Dodatkowo, gdy jedno z pól zostanie niewypełnione również wyświetla się ten sam komunikat

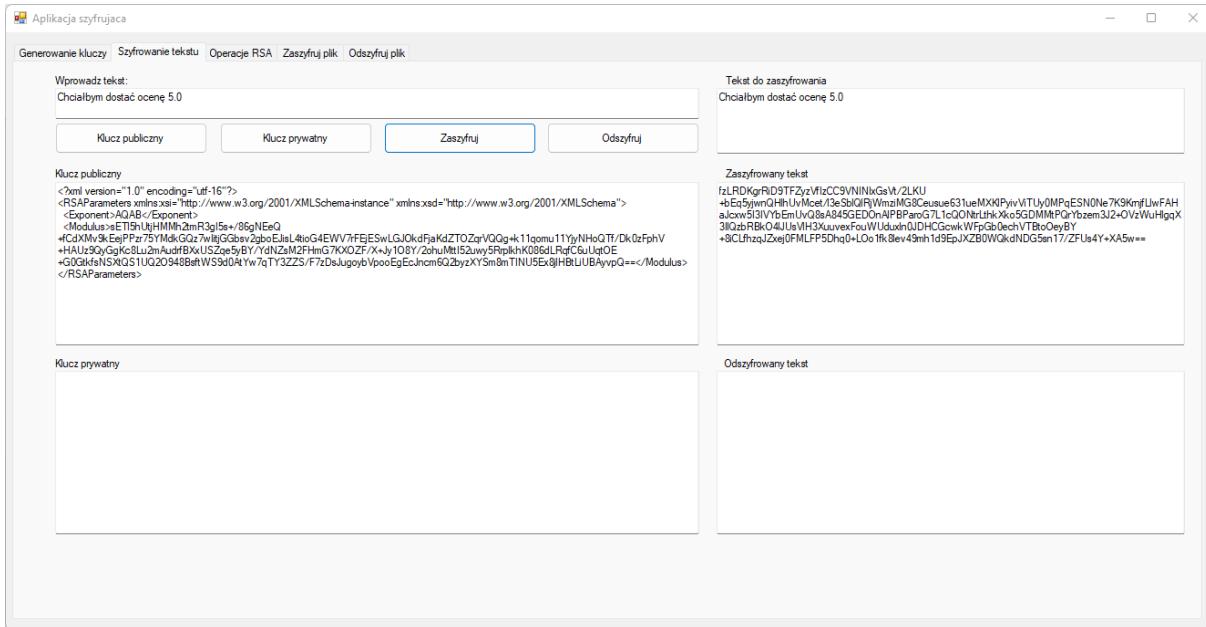


Test generowania klucza publicznego.



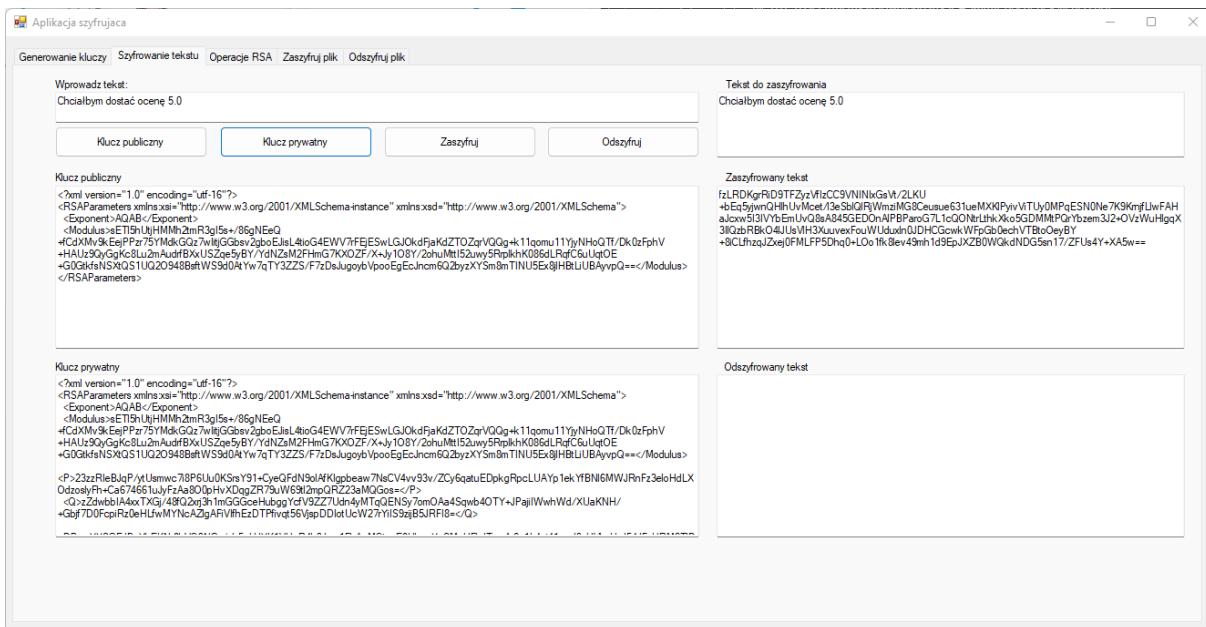
Po wciśnięciu przycisku klucz publiczny wygenerowany zostanie żądzany przez nas klucz

## Test szyfrowania krótkiego tekstu.



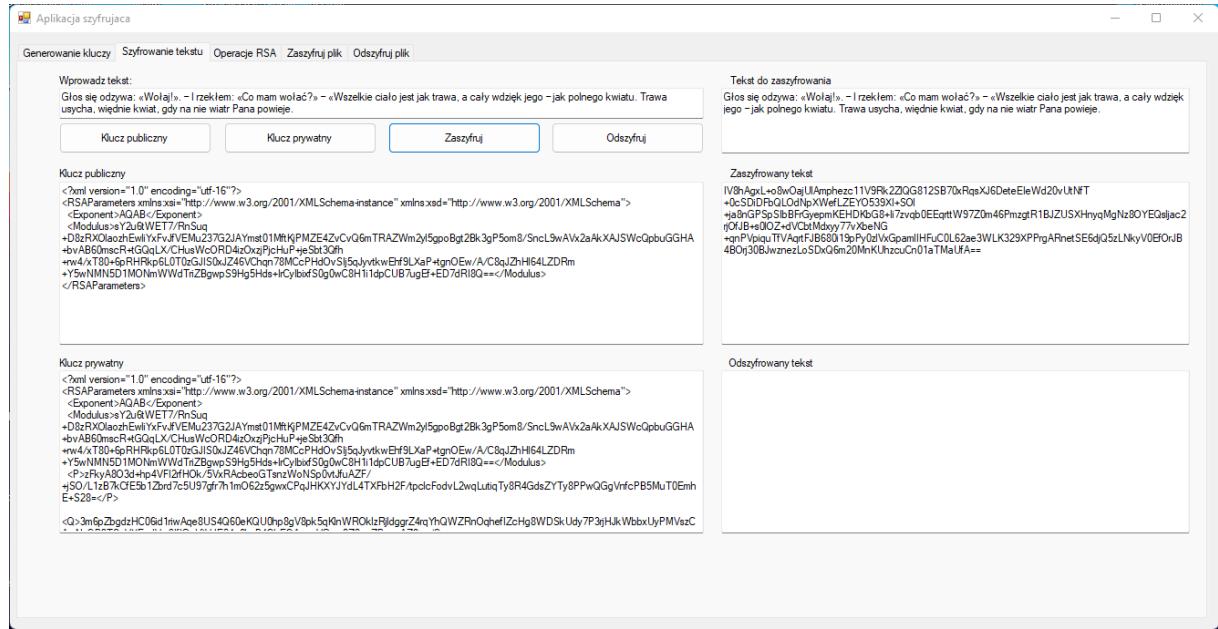
Tekst został zaszyfrowany i pojawił się w polu zaszyfrowany tekst

## Test generowania klucza prywatnego.



Klucz został wygenerowany i pojawił się w polu klucz prywatny

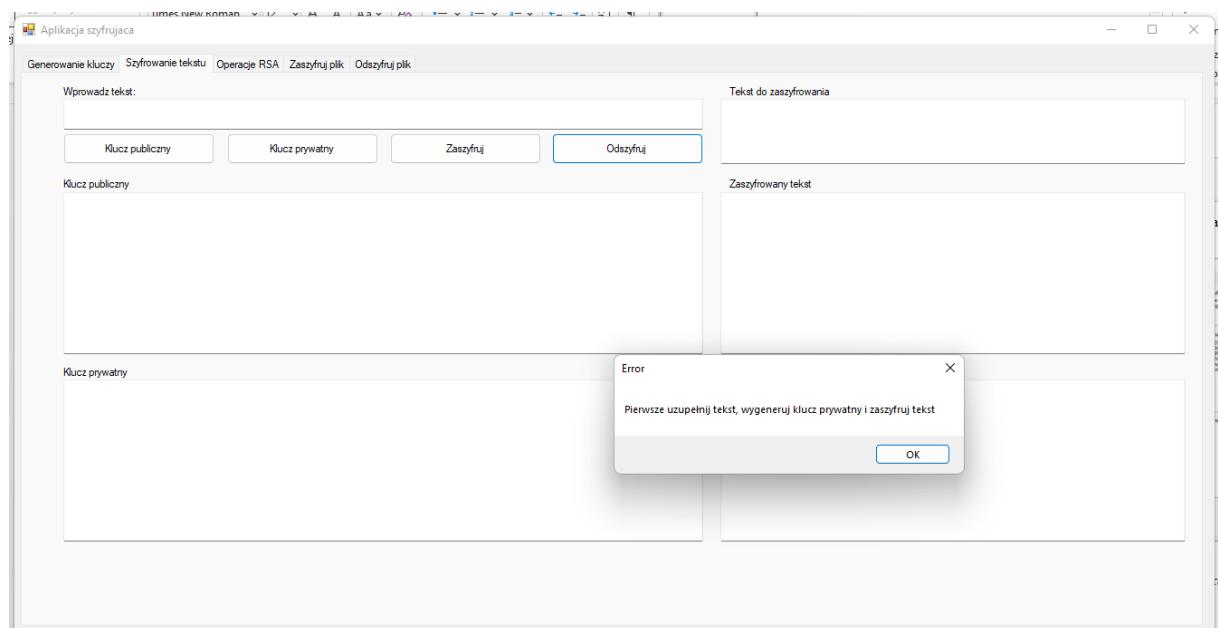
Test szyfrowania dłuższego fragmentu tekstu.



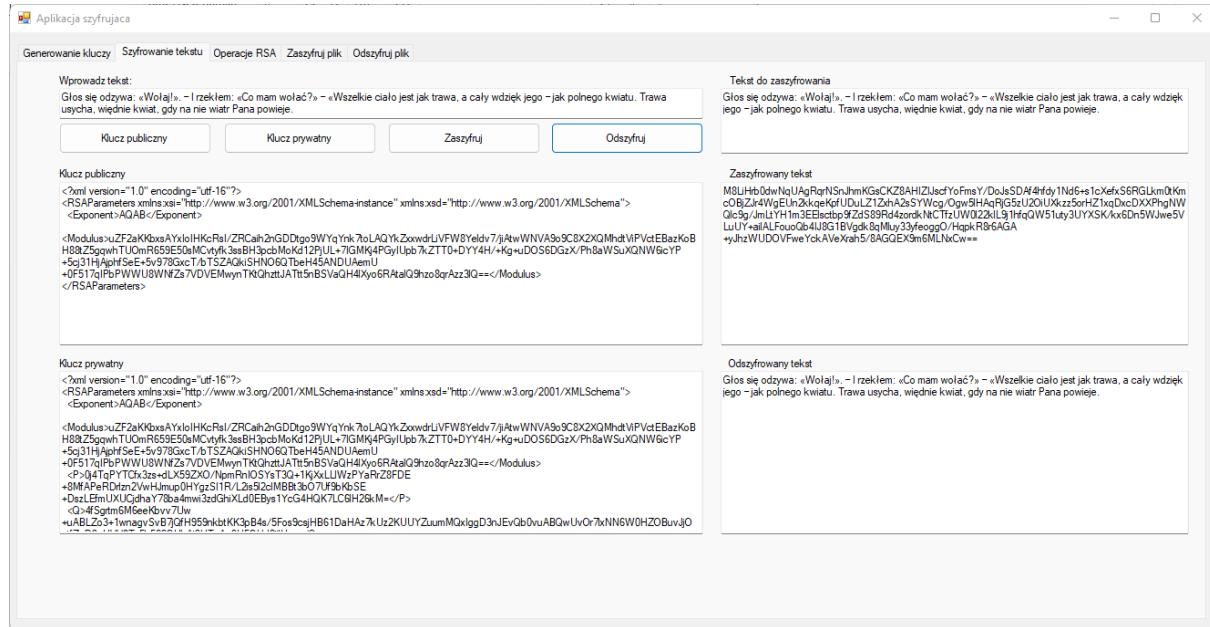
Aplikacja ogranicza maksymalną ilość znaków możliwą do wpisania w polu Tekst do zaszyfrowania do 214 znaków. Szyfrowanie odbywa się poprawnie.

## Test deszyfracji tekstu

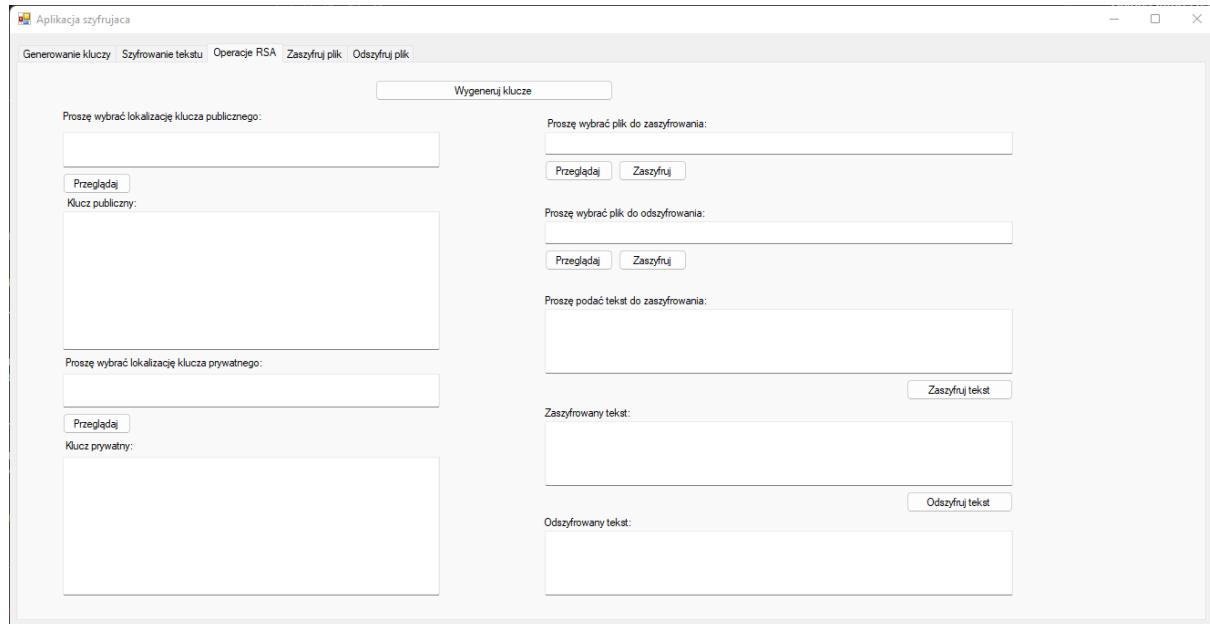
Test deszyfracji przy pustych polach.



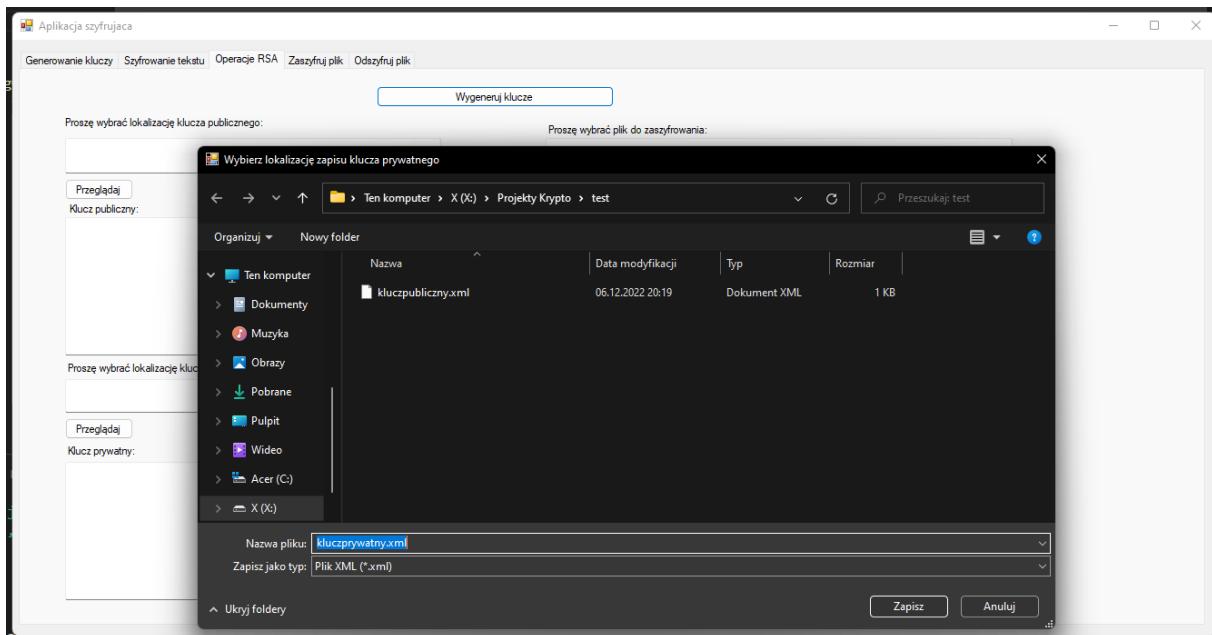
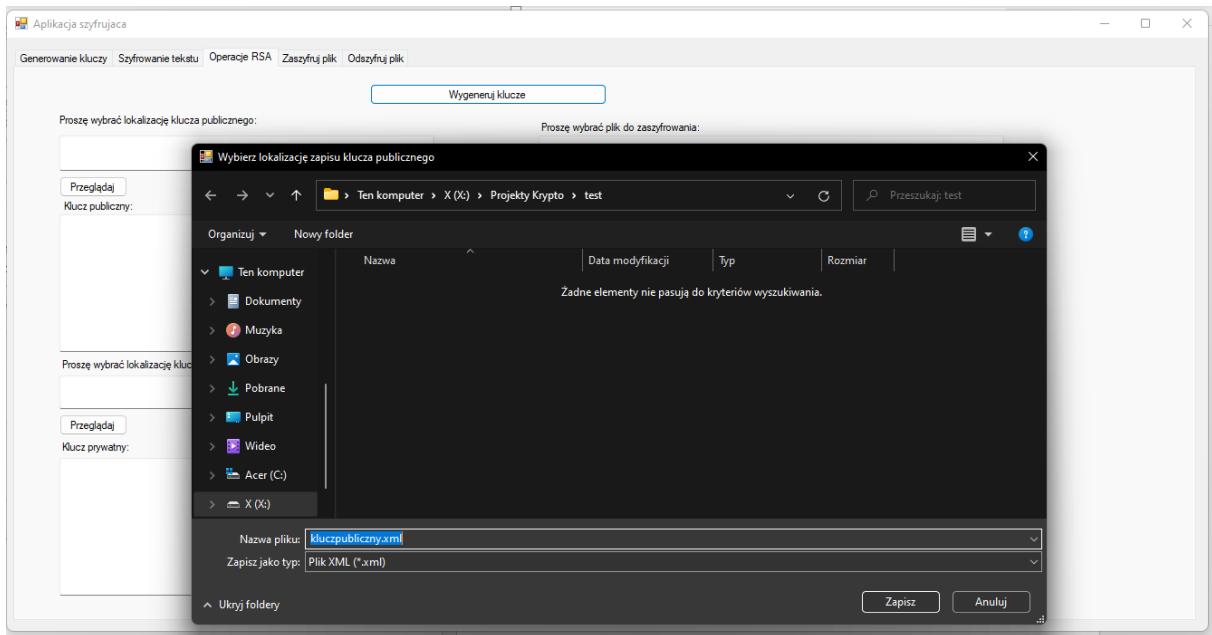
## Test deszyfracji tekstu



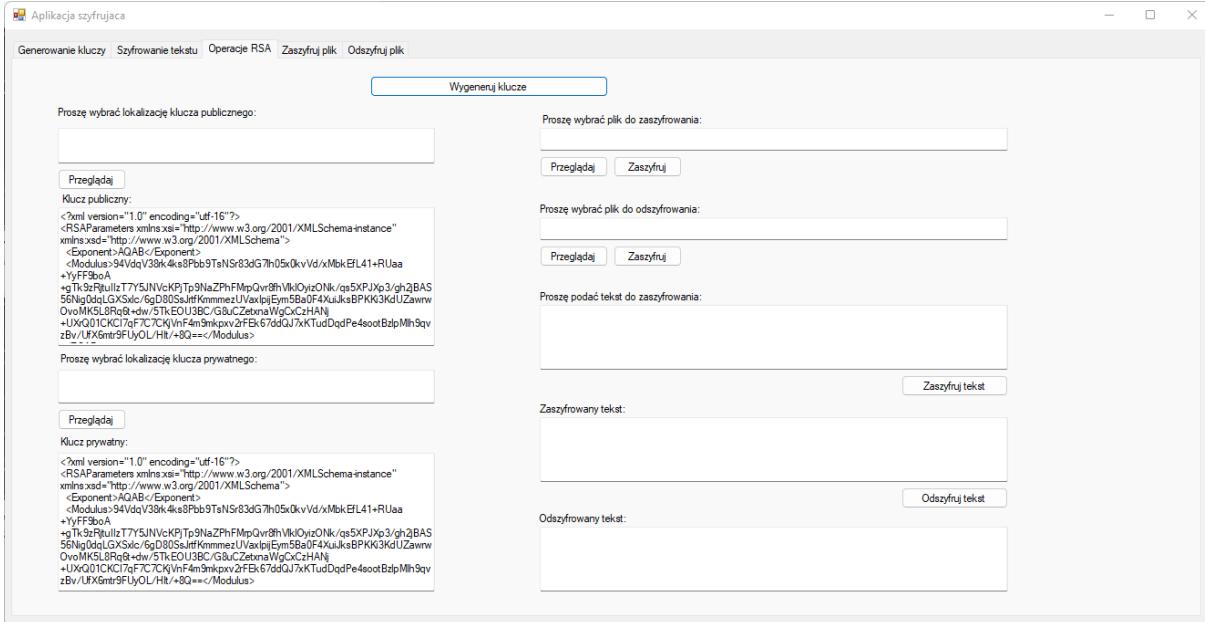
## Zakładka trzecia to OPERACJE RSA.



## Test generacji kluczy

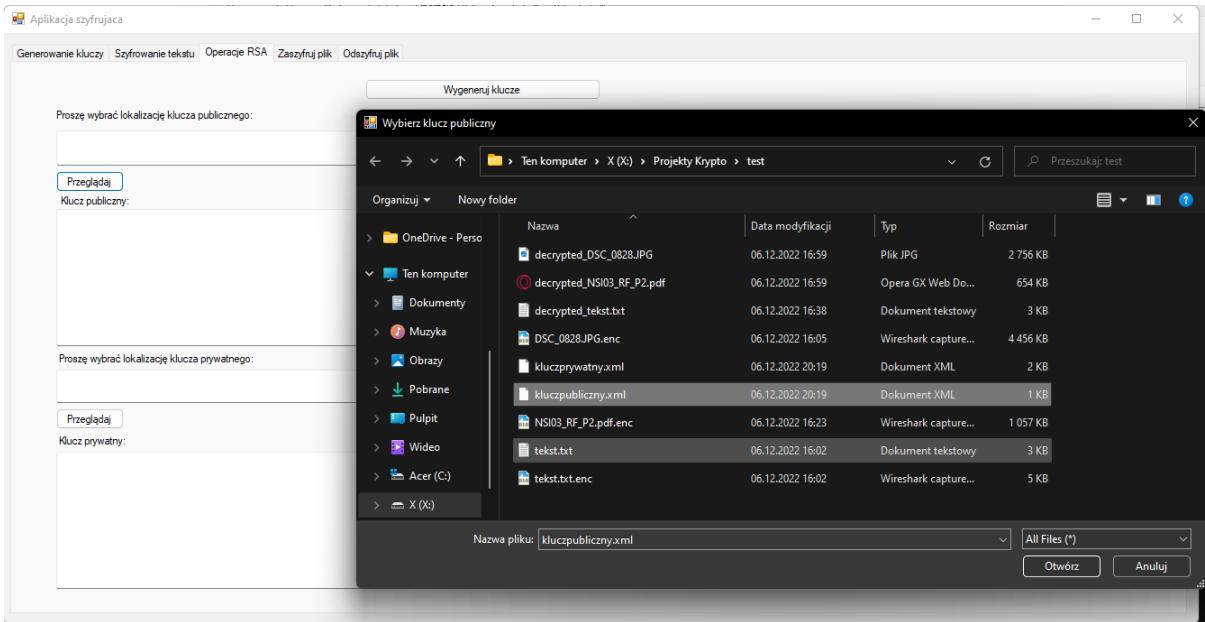


Kliknięcie przycisku wygeneruj klucze powoduje otworzenie okna, w którym wskazujemy lokalizację do zapisu wygenerowanych kluczy w formacie .xml

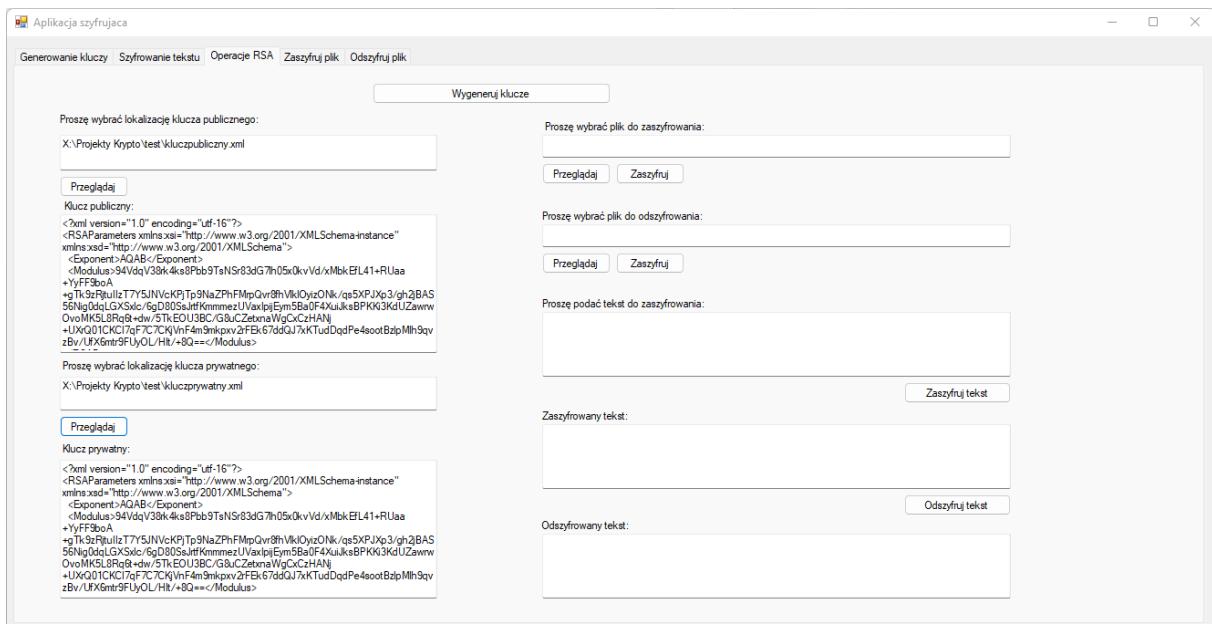


Wygenerowane klucze automatycznie uzupełniają pola klucz publiczny oraz klucz prywatny.

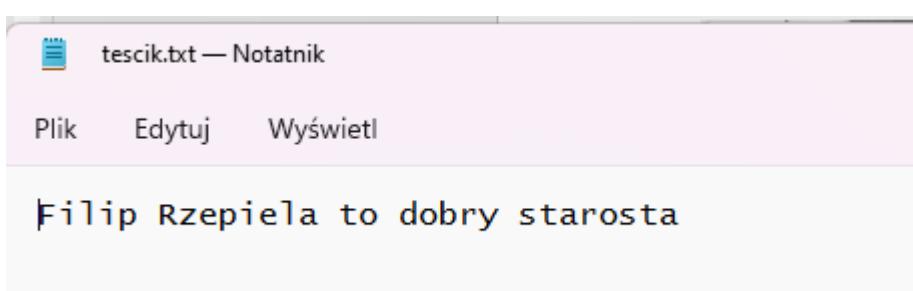
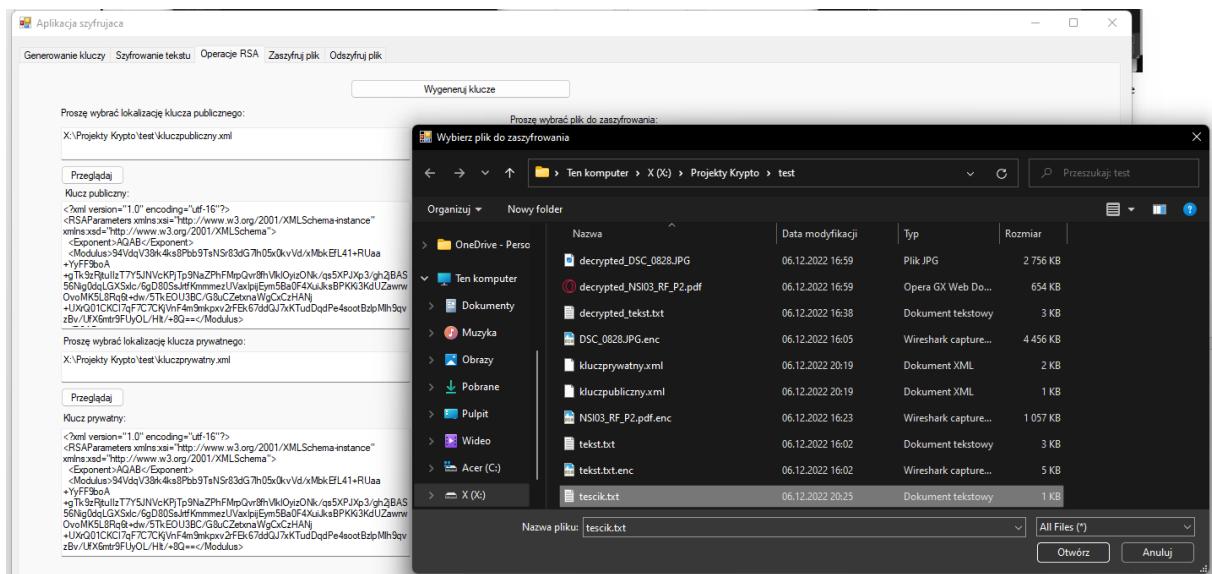
### Test pobrania kluczy z pliku.



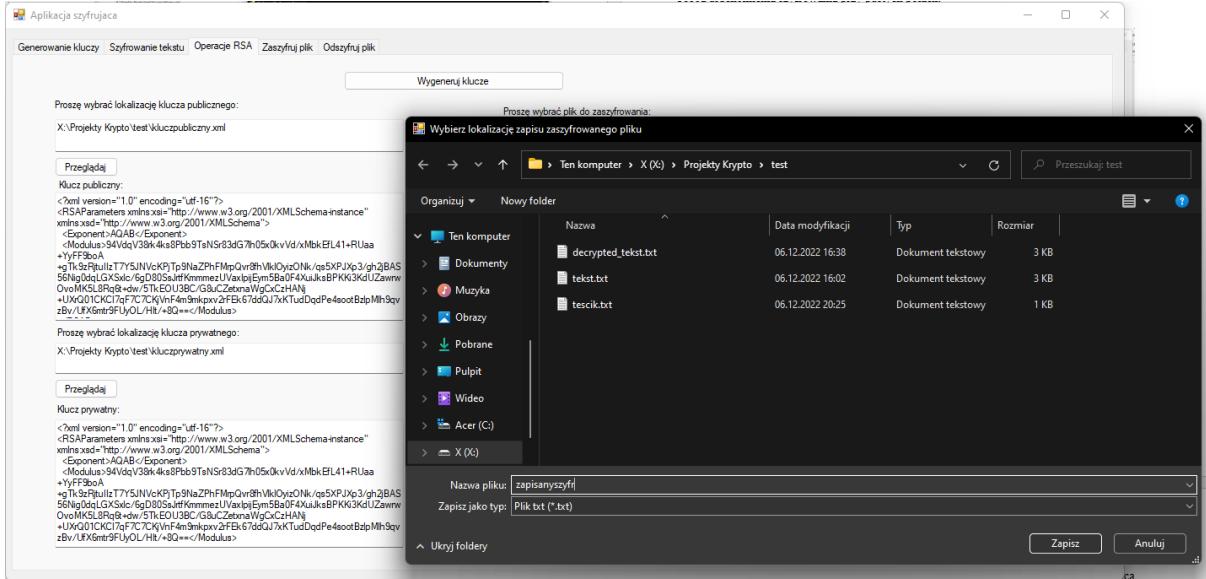
Wybieramy plik klucza publicznego, który pojawia się w polu klucza publicznego. Podobnie w przypadku klucza prywatnego.



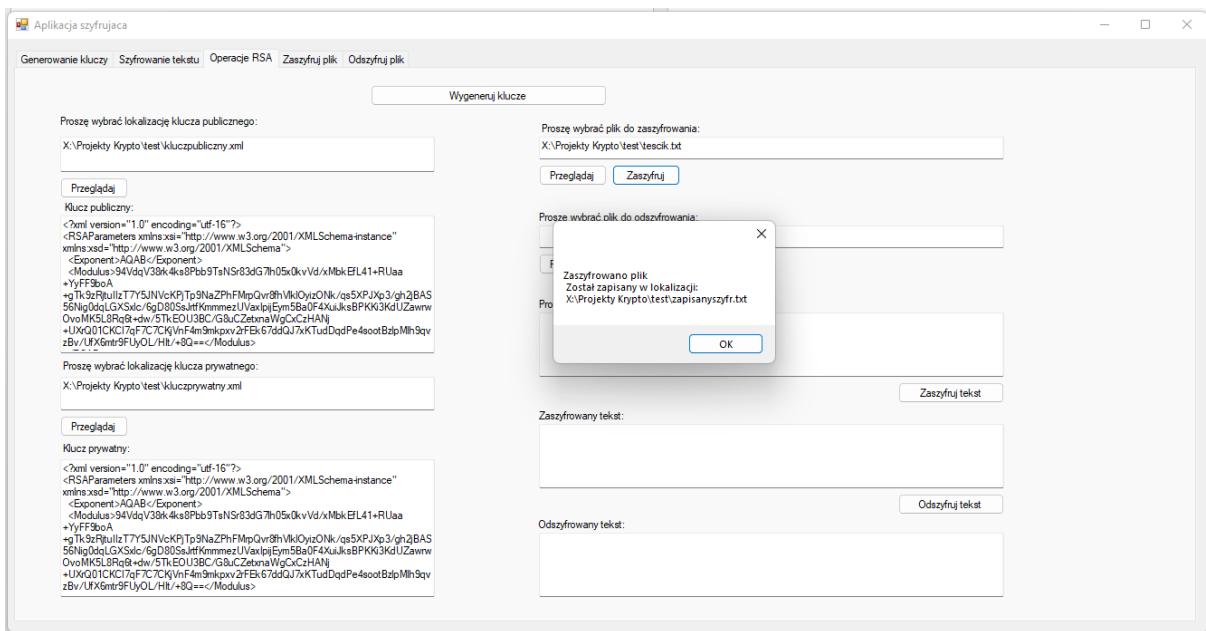
## Test zaszyfrowania tekstu z pliku



Wybieramy plik i klikamy zaszyfruj.



Po wciśnięciu zaszyfruj otwiera nam się okno w którym zapisujemy zaszyfrowany plik.

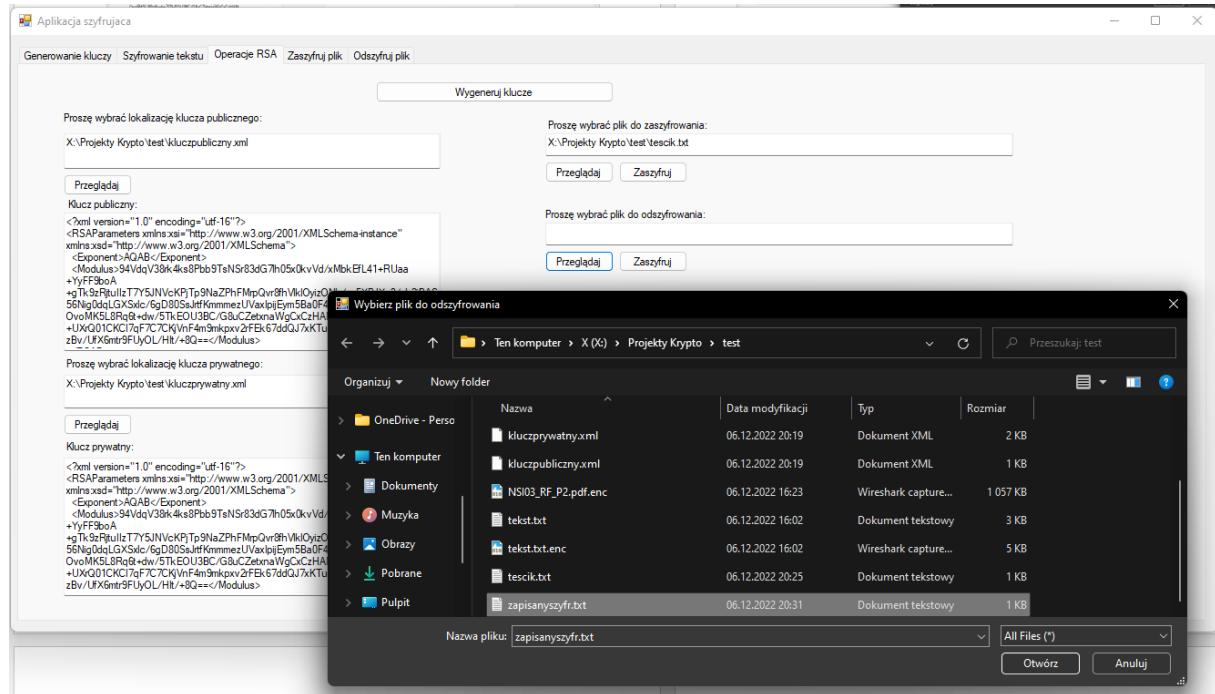


Program zwraca powiadomienie o zapisanym pliku i jego lokalizacji

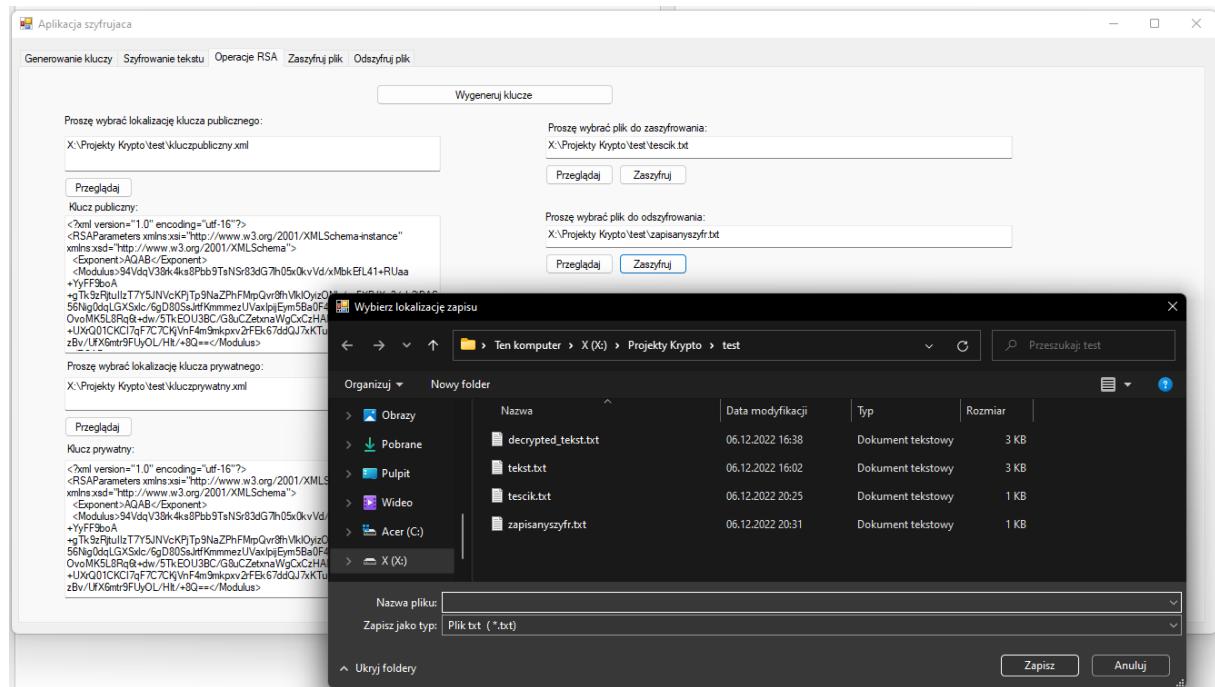


## Test odszyfrowywania pliku

Klikamy przeglądaj w celu wskazania pliku do odszyfrowania

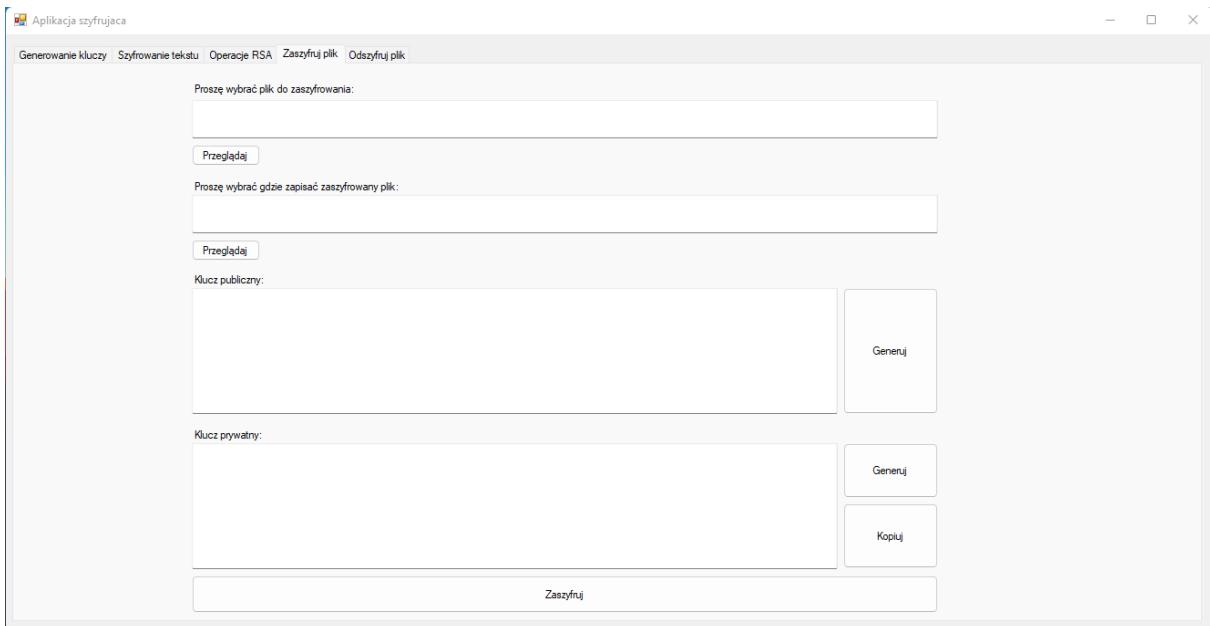


Klikamy odszyfruj (przycisk zaszyfruj zapomniałem poprawić) i wskazujemy lokalizację

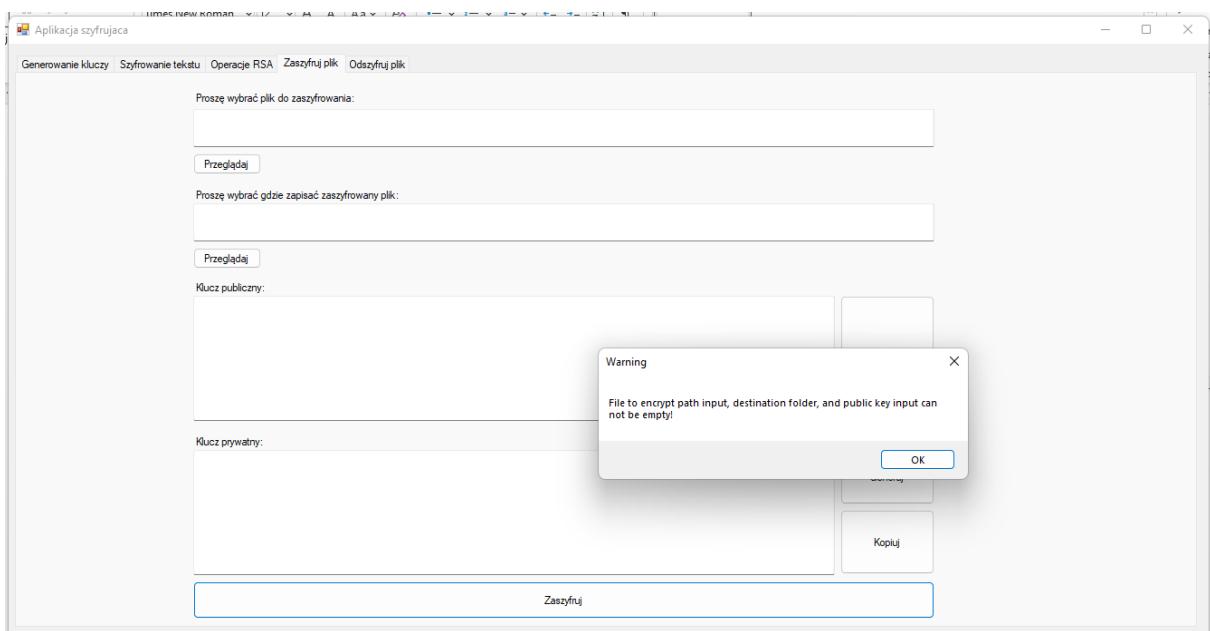


# Test szyfrowania plików

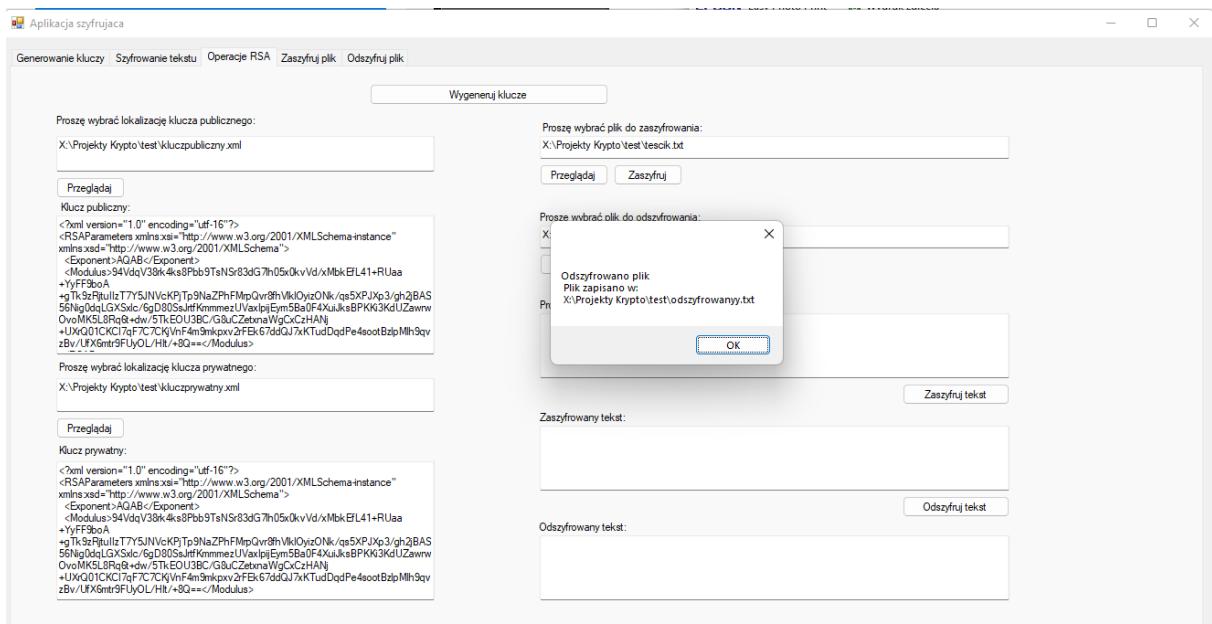
Widok okna szyfrowania plików.



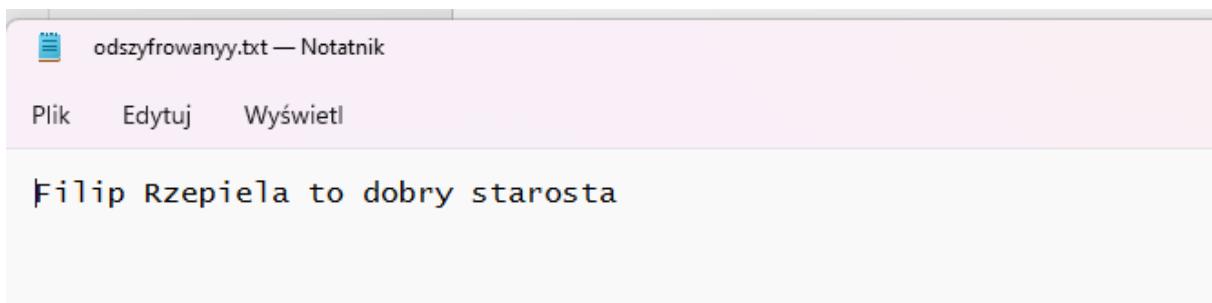
Próba uruchomienia szyfrowania przy pustych polach.



Wyświetla się komunikat, że pola nie mogą być puste. Aplikacja wyświetla wyżej pokazany komunikat zarówno, jeśli wszystkie pola są puste i jeśli jedno z wymaganych pól jest puste.

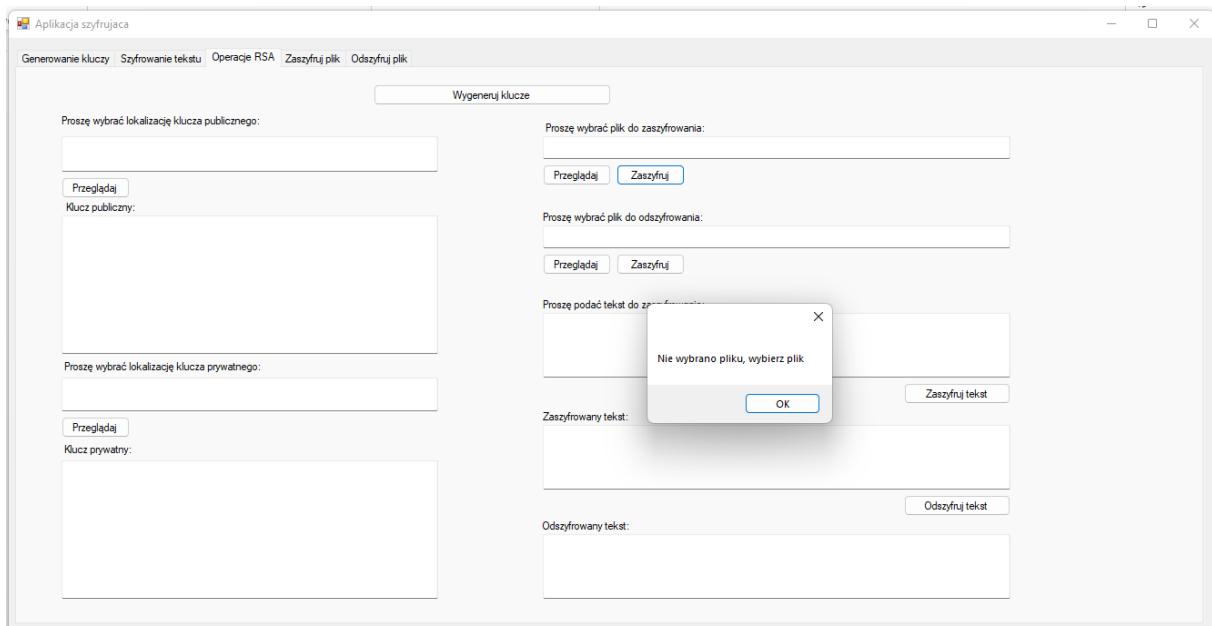


Program informuje nas o odszyfrowaniu oraz lokalizacji

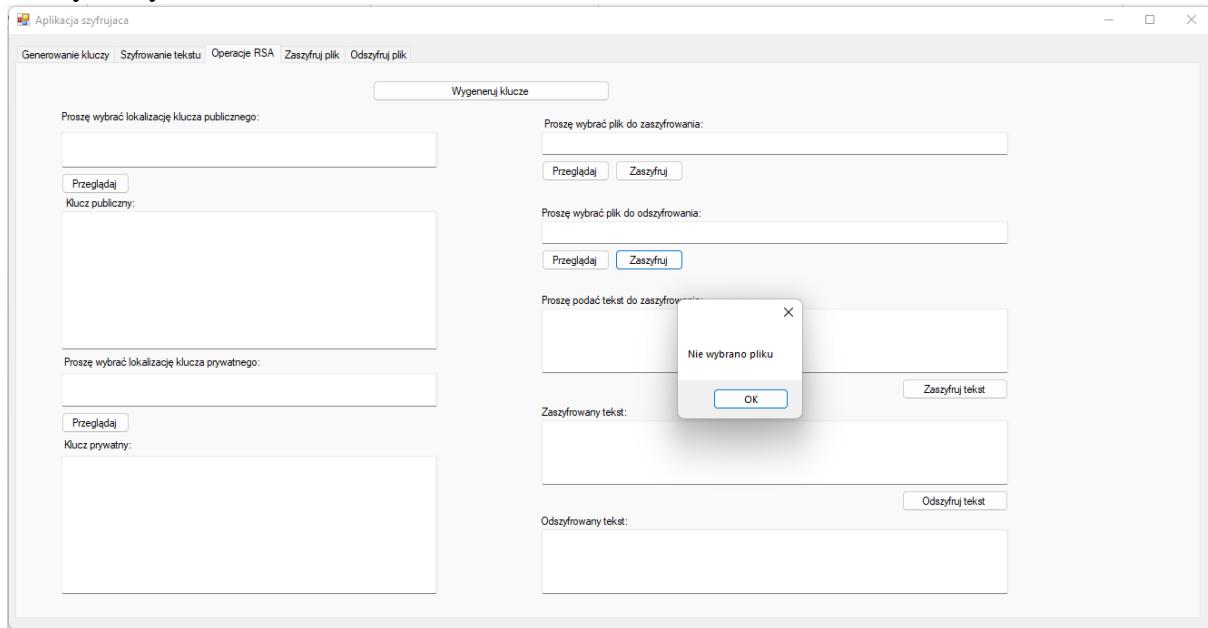


W przypadku niewprowadzenia bądź wygenerowania kluczy zwrócony zostanie komunikat

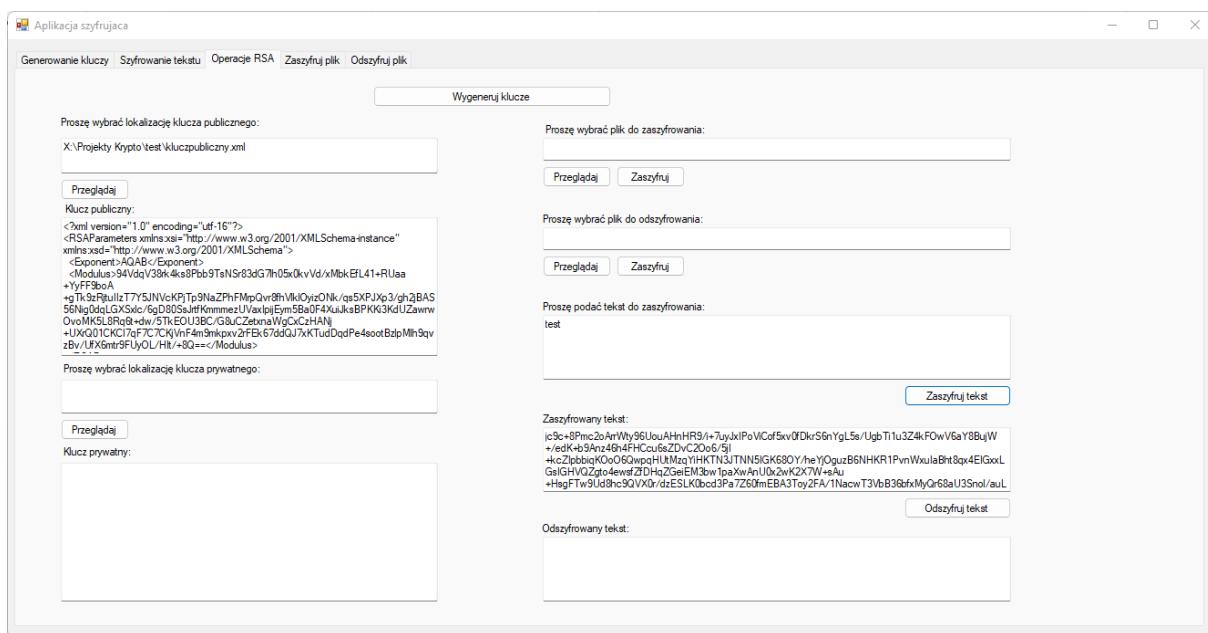
Szyfrowanie:



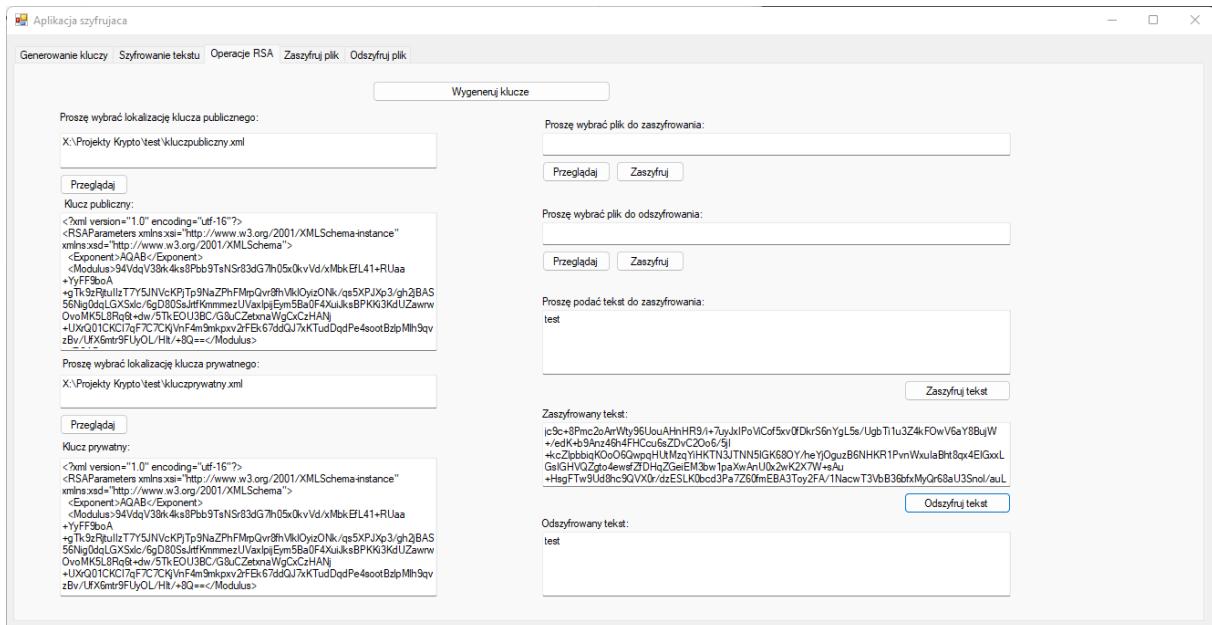
## Odszyfrowywanie:



## Test szyfrowania pliku

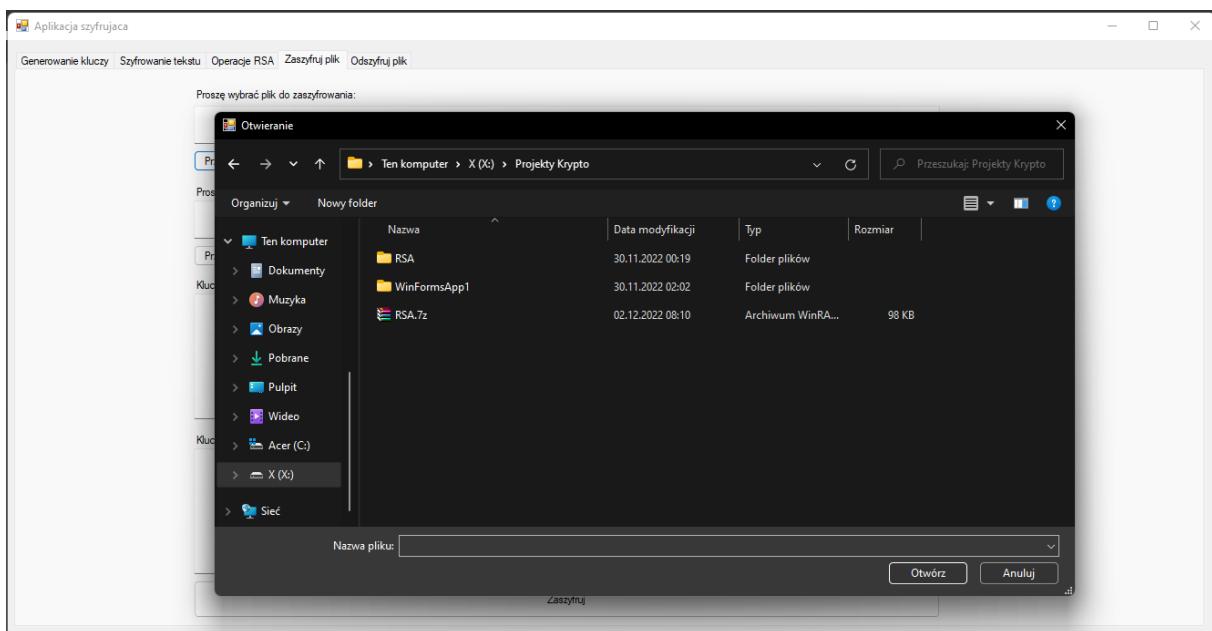


Aby zaszyfrować plik musimy mieć klucz publiczny.

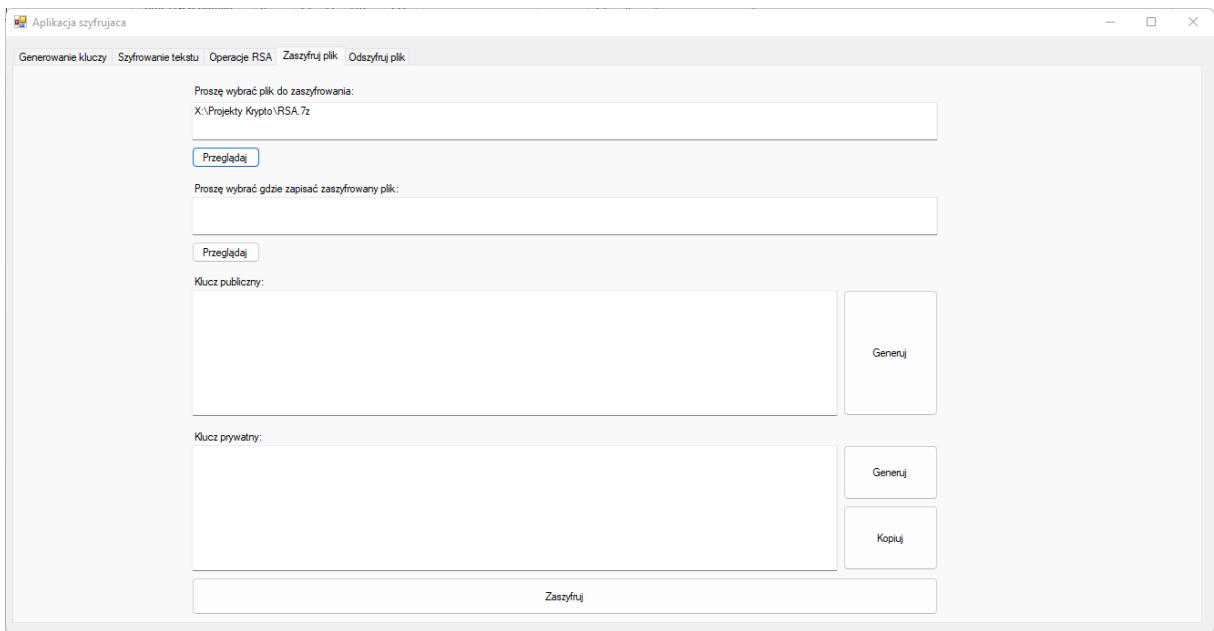


W przypadku odszyfrowania potrzebujemy również klucza prywatnego

Test pola wskazania pliku do zaszyfrowania

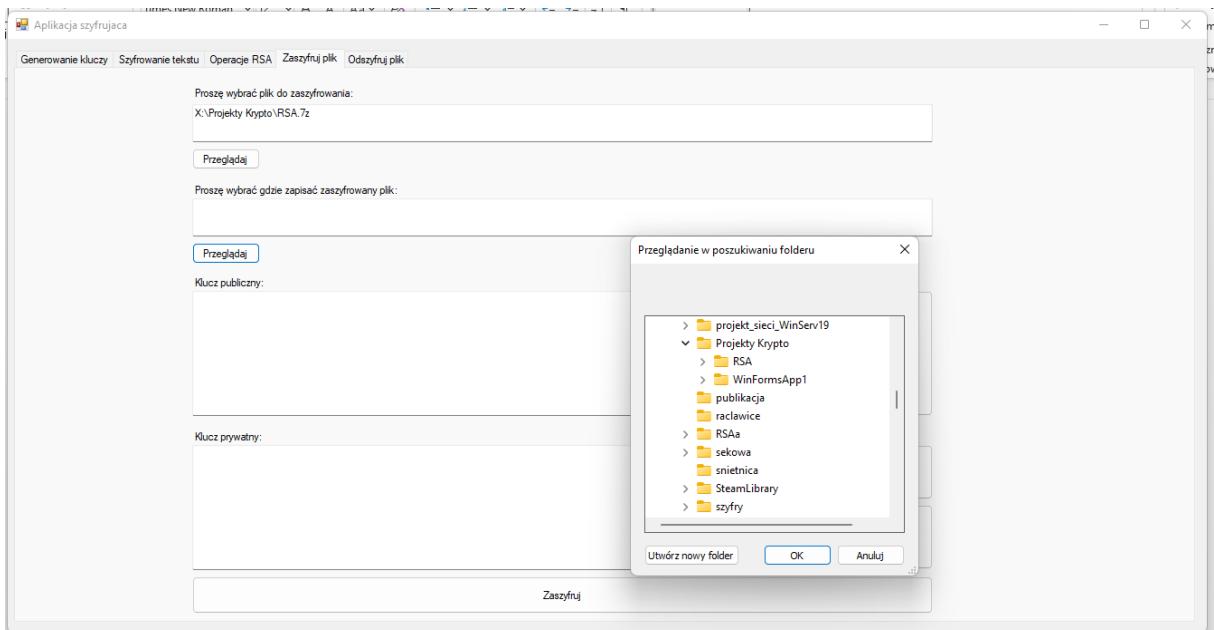


Po wybraniu pliku do zaszyfrowania jego ścieżka pojawia się w polu

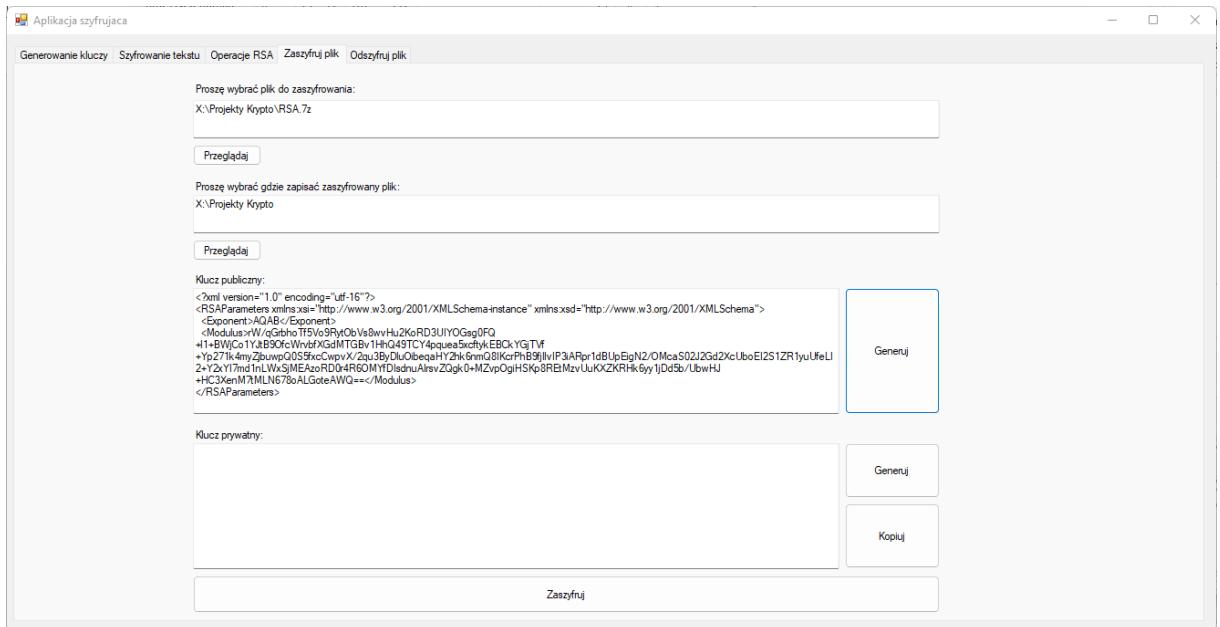


## Test wybrania lokalizacji dla zaszyfrowanego pliku

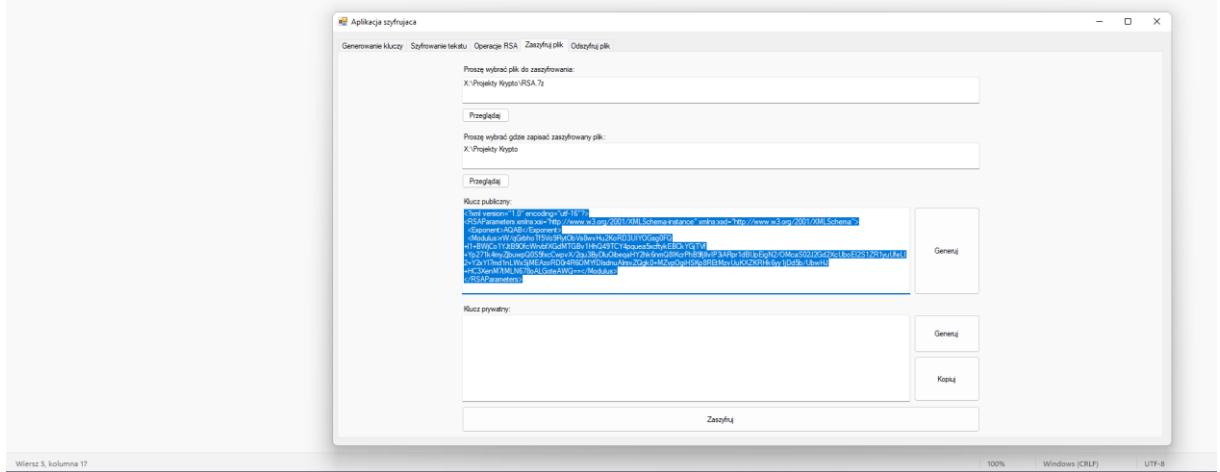
Sytuacja jest podobna jak w przypadku pola wskazującego plik do zaszyfrowania. Za pomocą przycisku znajdującego się obok pola uruchamiane jest okno wyboru folderu.



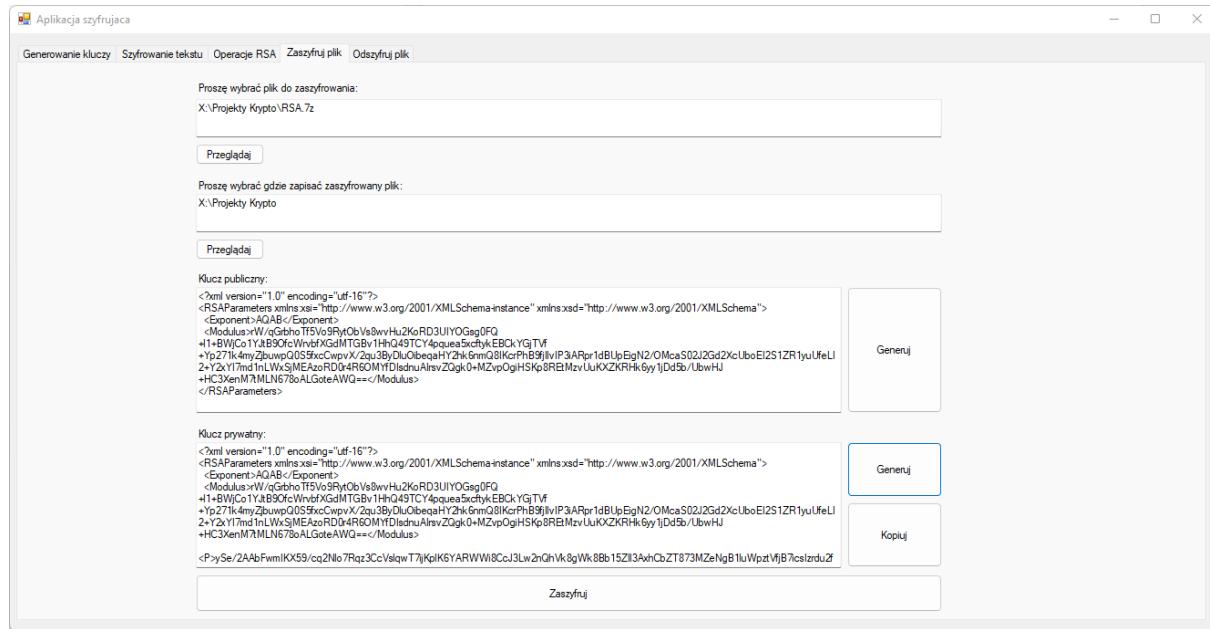
## Test generowania klucza publicznego



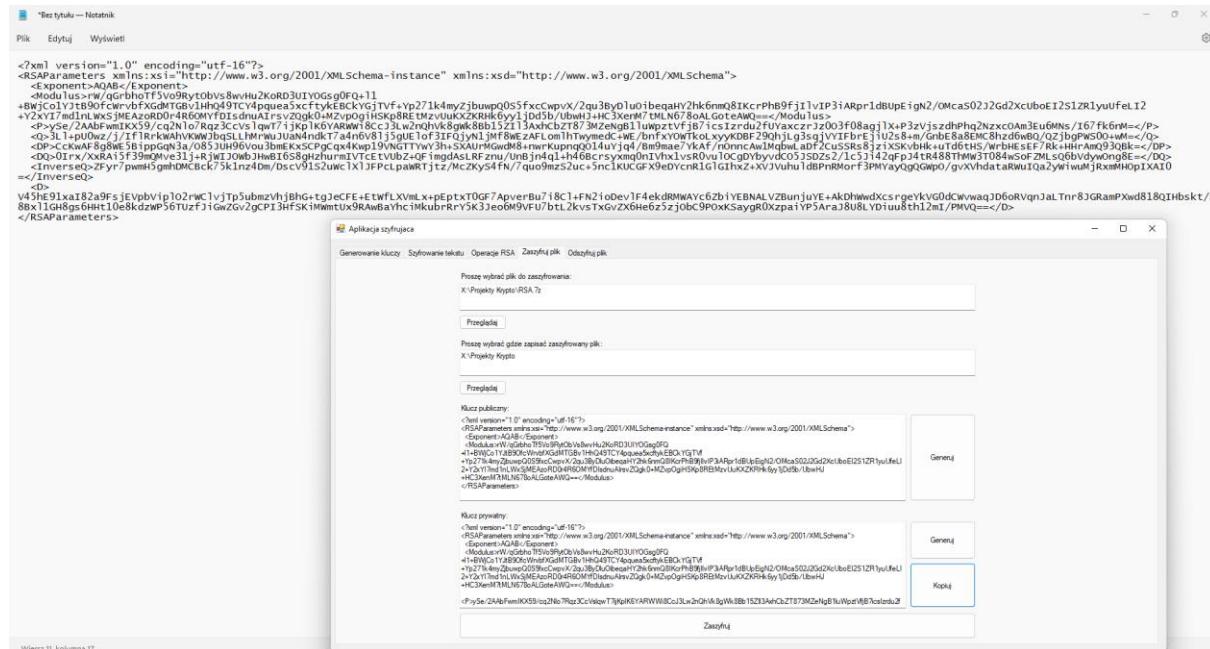
Oczywiście wygenerowany klucz, możemy zaznaczyć w całości za pomocą ctrl+a, skopiować i wkleić do np. notatnika



## Test generowania klucza prywatnego.

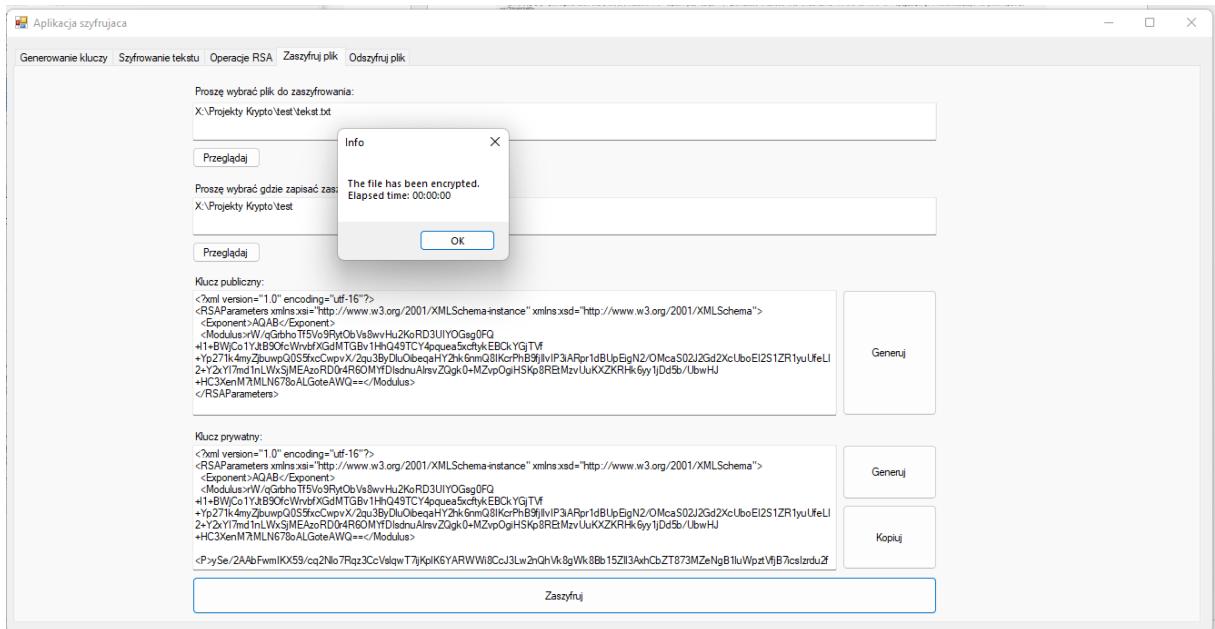


Możemy go skopiować za pomocą przycisku kopij bądź jak wyżej 😊

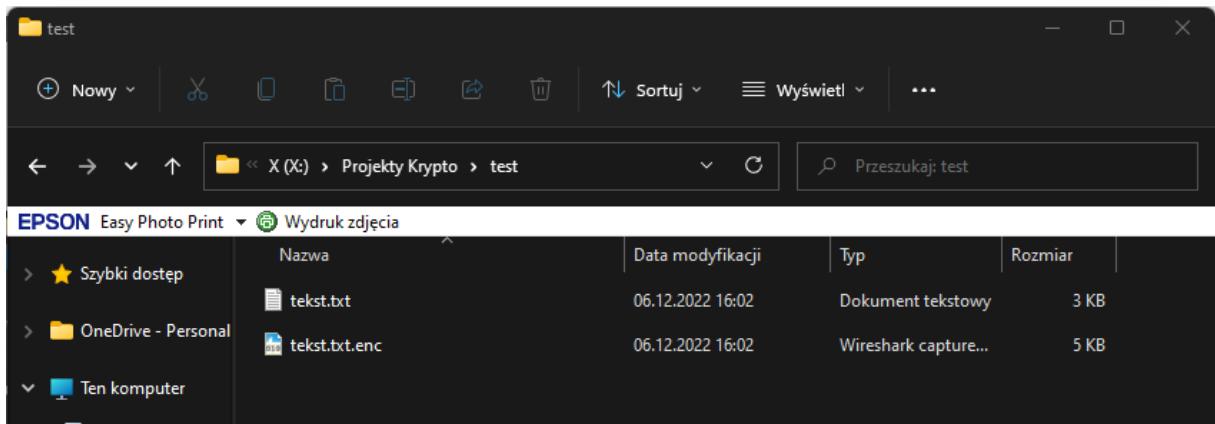


Wiersz 11, kolumna 17

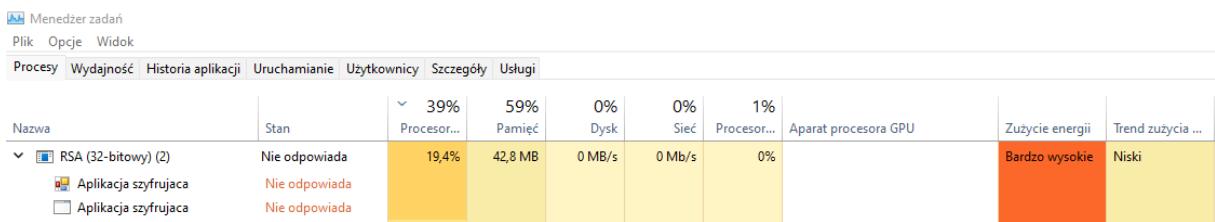
## Test szyfrowania pliku tekstowego

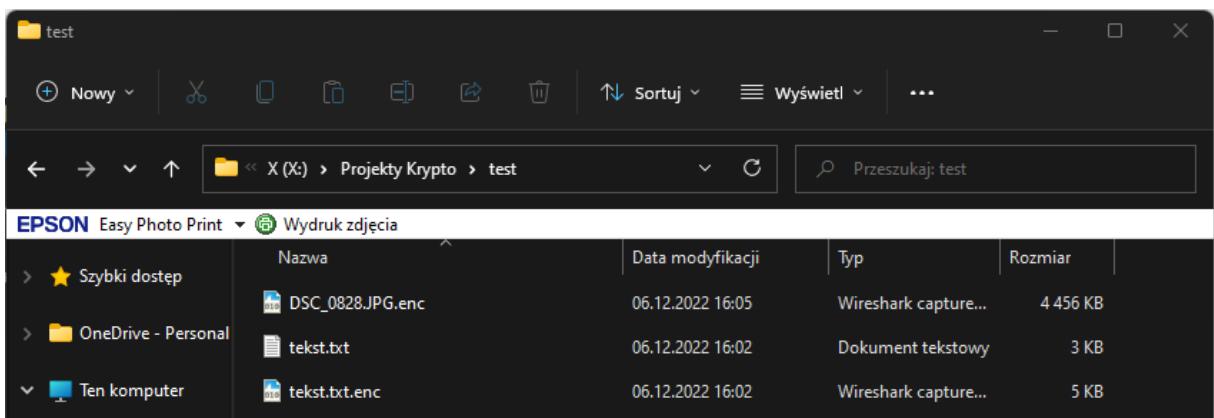
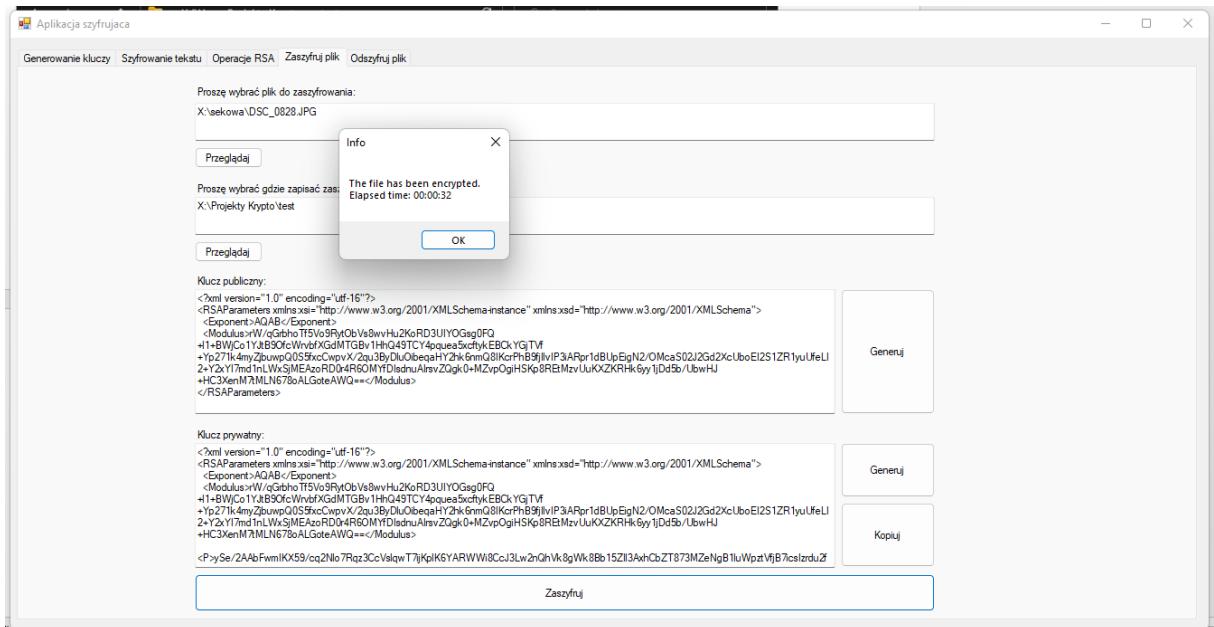


W folderze pojawił się zaszyfrowany plik

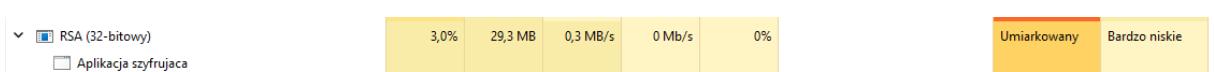
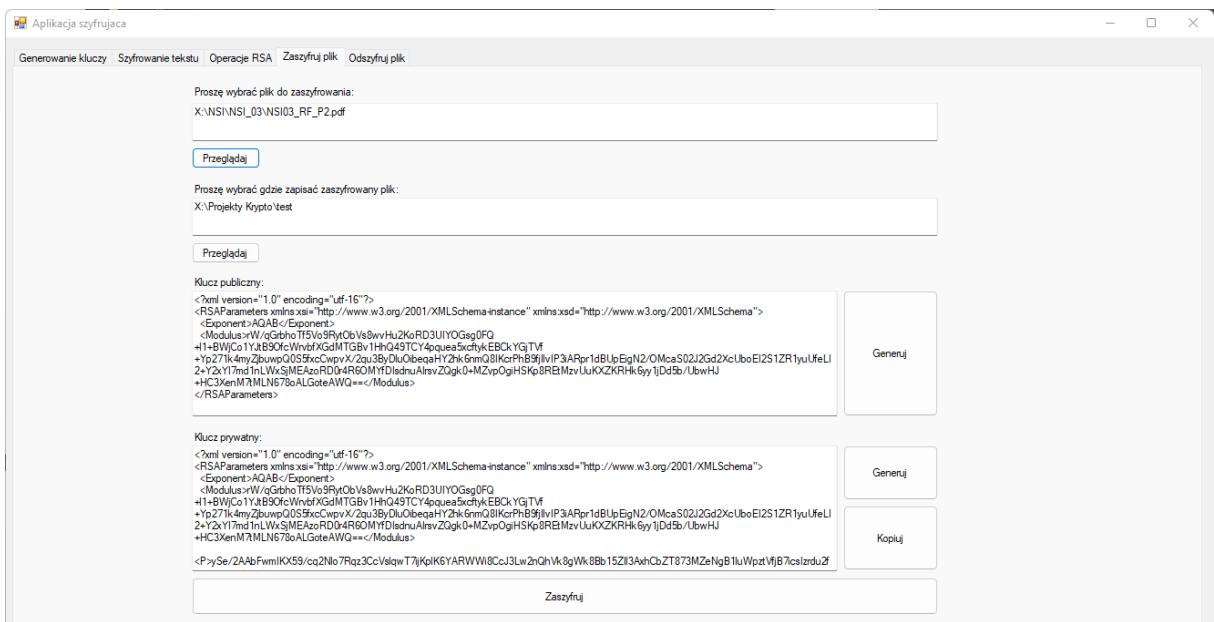


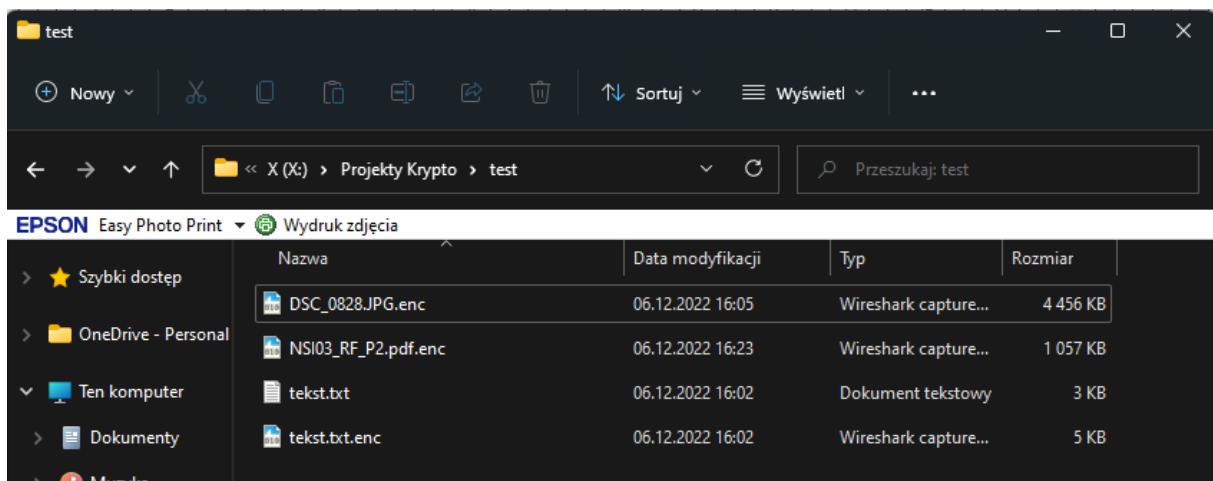
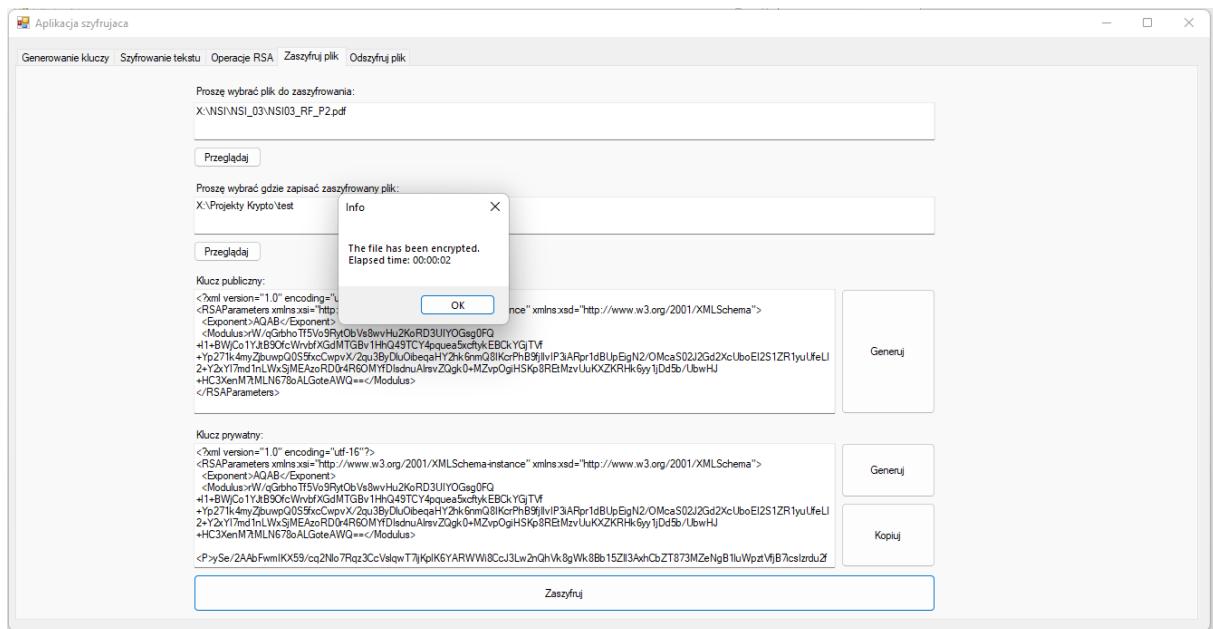
Test szyfrowania pliku .jpg





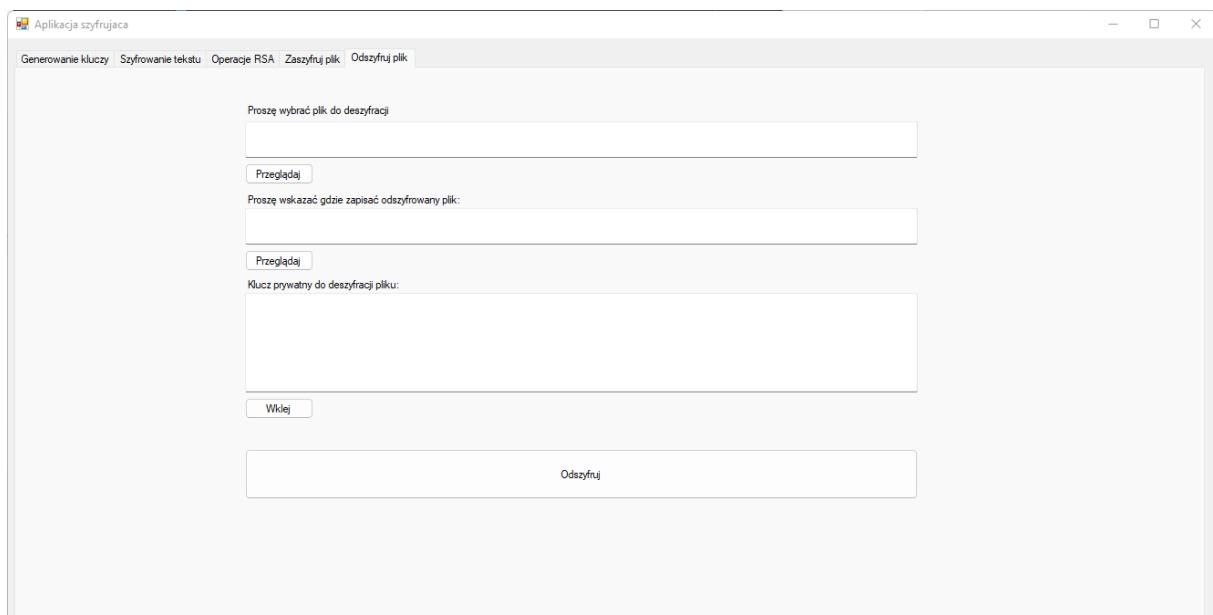
## Test szyfrowania pliku .pdf



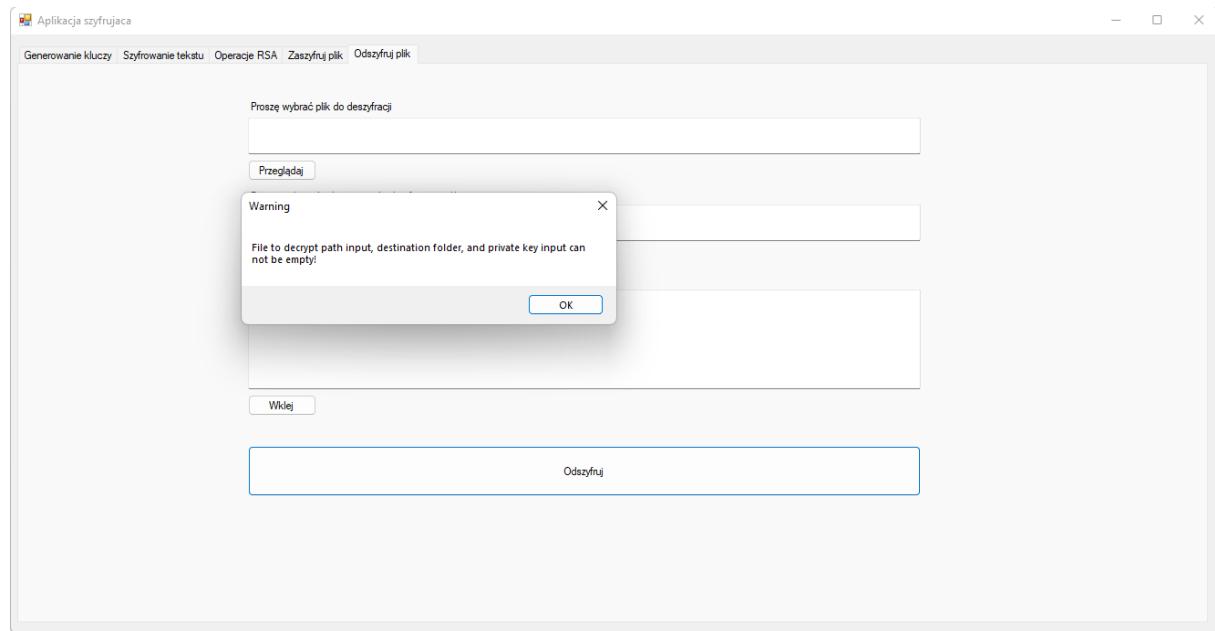


## Test deszyfrowania pliku

Widok okna deszyfrowania pliku

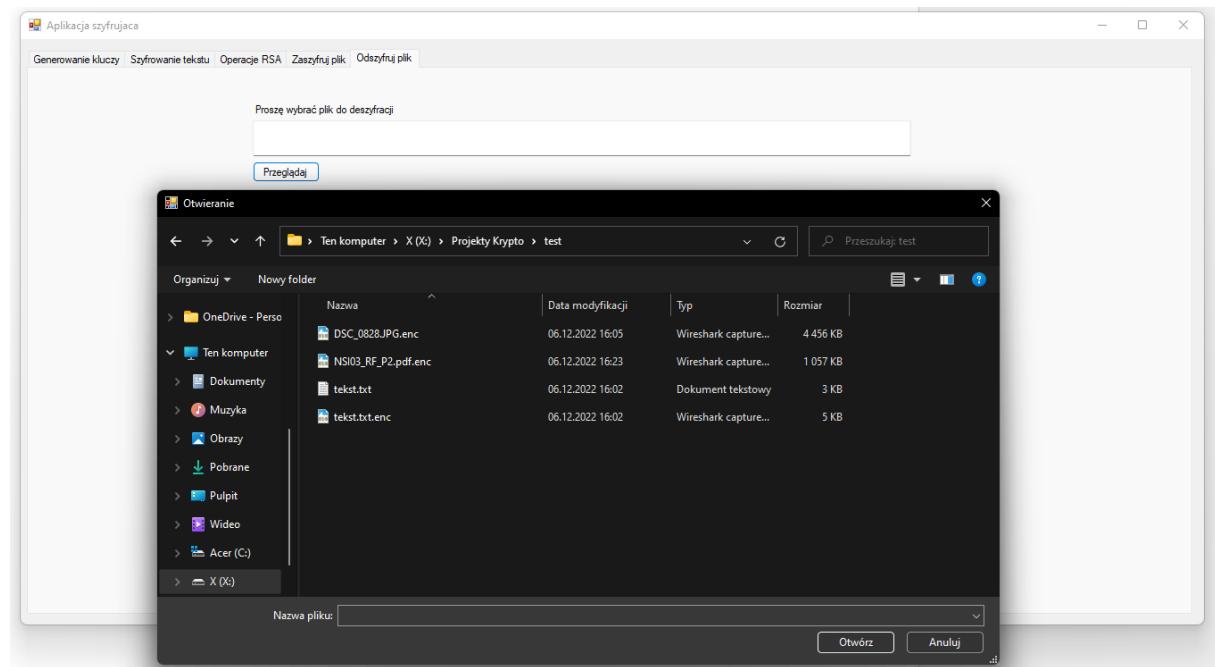


## Test uruchomienia deszyfrowania przy pustych polach



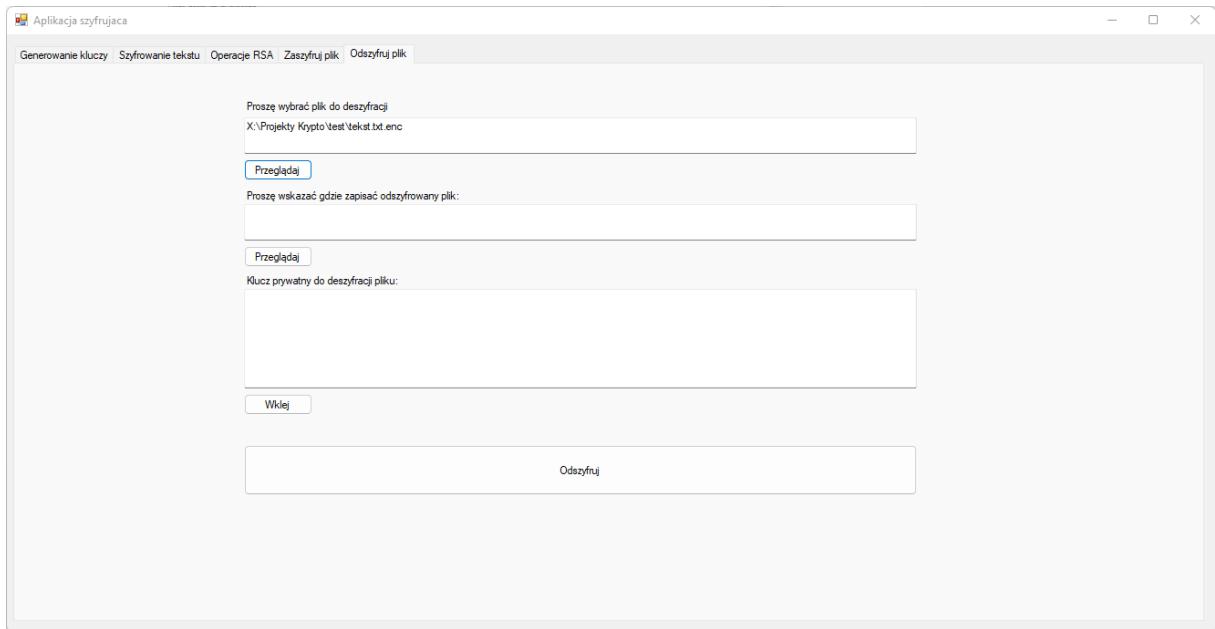
Aplikacja wyświetla komunikat, że pola nie mogą być puste. Komunikat wyświetlany jest zarówno, gdy jedno lub więcej z pól jest puste.

## Test pola wskazania pliku do deszyfracji

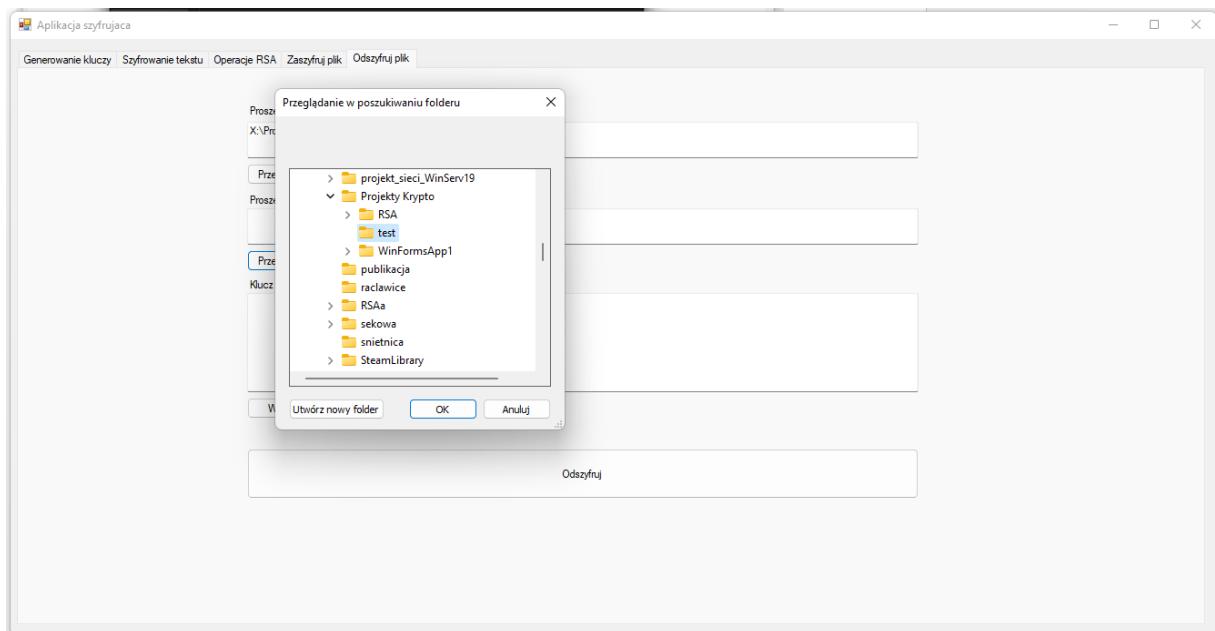


	Nazwa	Data modyfikacji	Typ	Rozmiar
	DSC_0828JPG.enc	06.12.2022 16:05	Wireshark capture...	4 456 KB
	NSI03_RF_P2.pdf.enc	06.12.2022 16:23	Wireshark capture...	1 057 KB
	tekst.txt	06.12.2022 16:02	Dokument tekstowy	3 KB
	tekst.txt.enc	06.12.2022 16:02	Wireshark capture...	5 KB

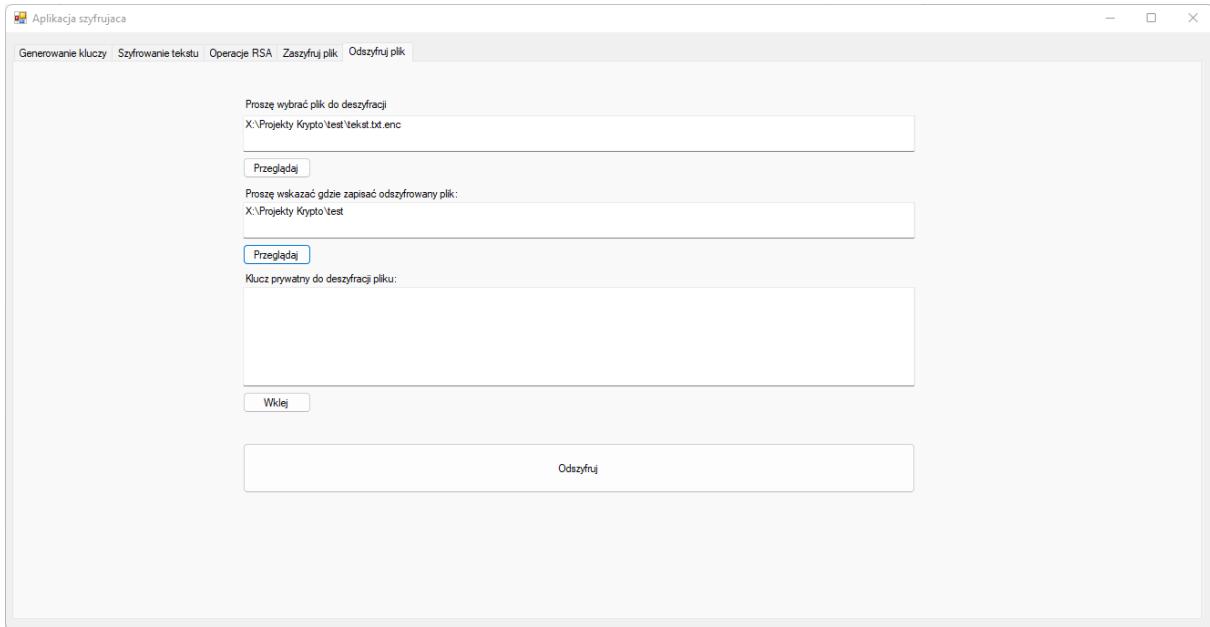
Po wybraniu pliku ścieżka pojawia nam się w polu.



Test pola wskazania, gdzie zapisać odszyfrowany plik

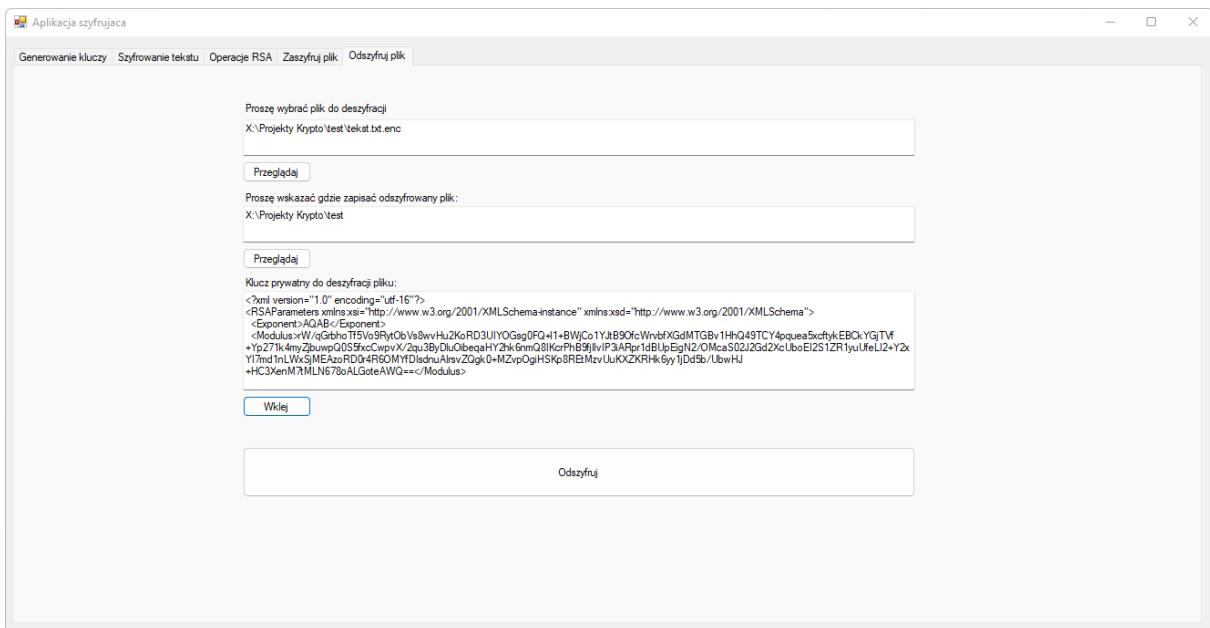


Po wskazaniu ścieżka pokazuje nam się w polu

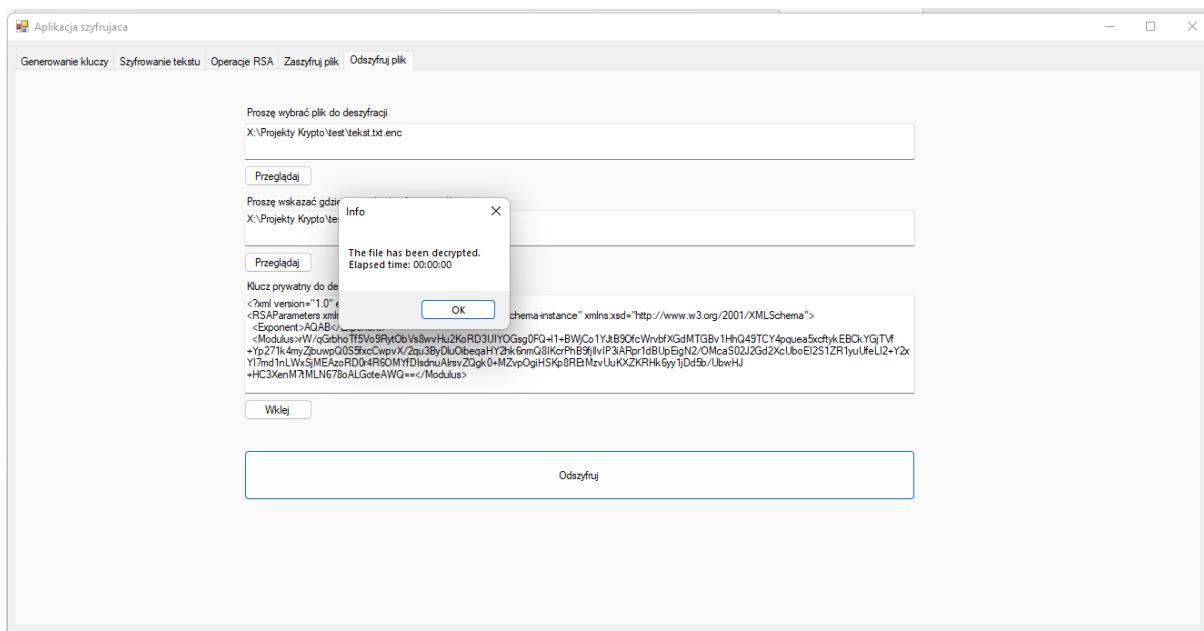


Test generowania klucza prywatnego (skopiowanego z zakładki szyfrowanie)

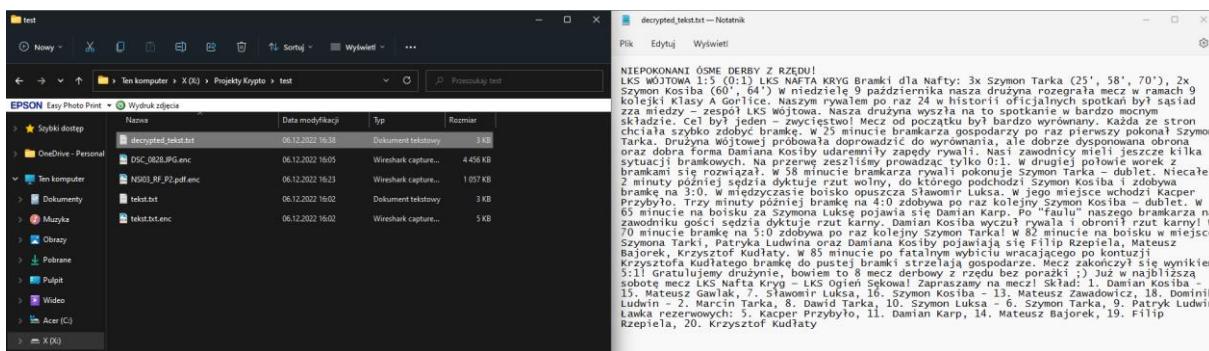
Za pomocą przycisku wklej dodajemy wcześniej skopiowany przez nas klucz prywatny



W celu rozszyfrowania pliku wciskamy przycisk Odszyfruj. Plik zostaje odszyfrowany



W folderze pojawia się plik decrypted\_NAZWA.txt

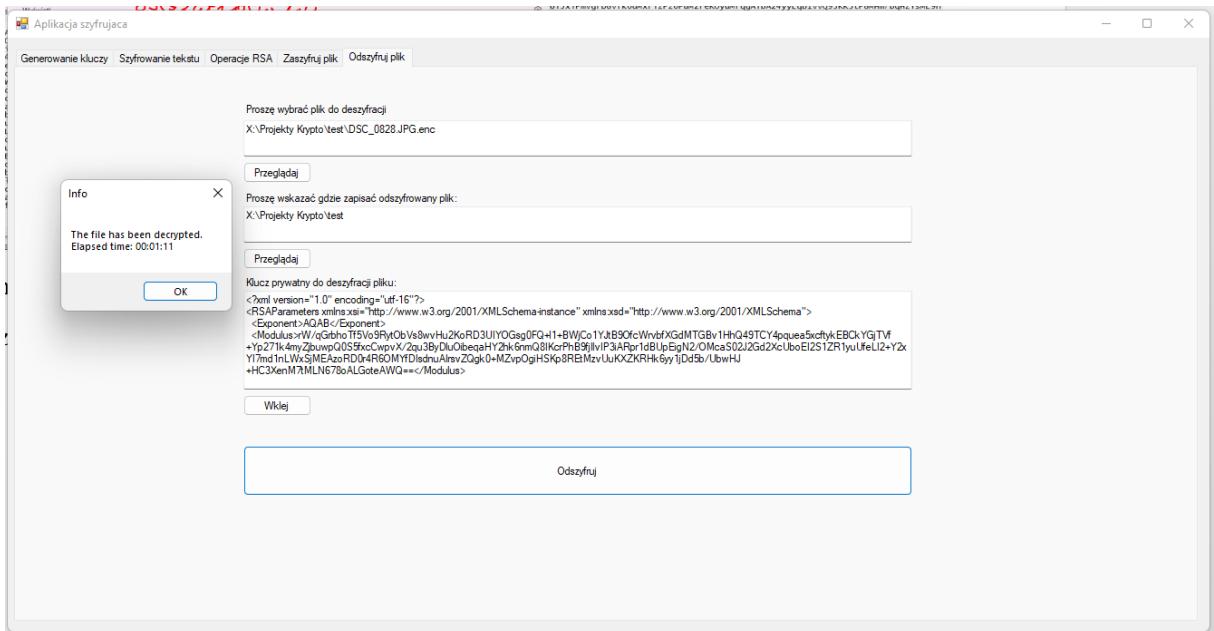


Porównanie zawartości plików. Przed szyfrowaniem. Po zaszyfrowaniu. Po rozszyfrowaniu

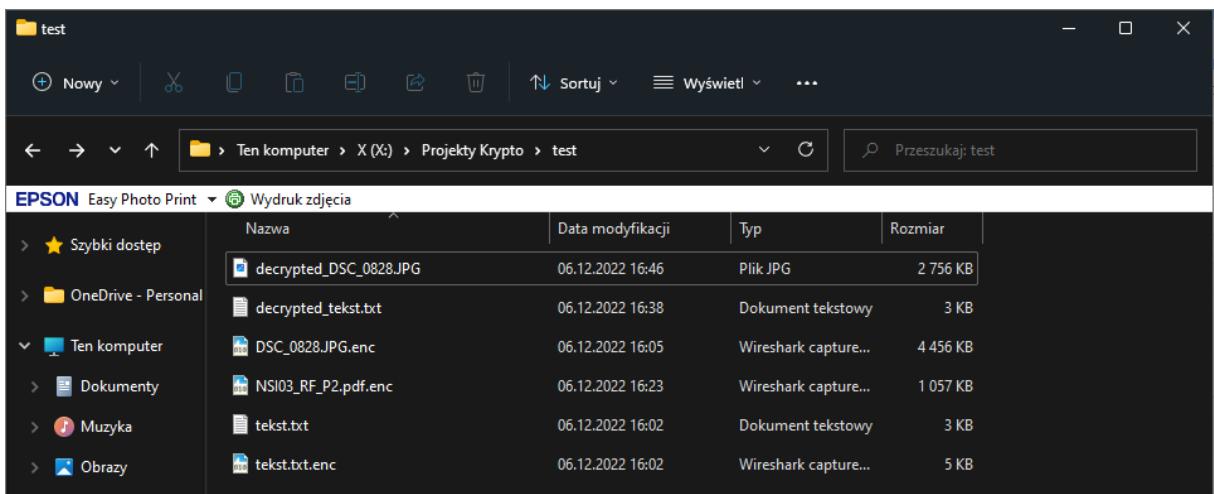


Jak można zauważać plik został poprawnie odszyfrowany.

Odszyfrowanie pliku .jpg



W folderze pojawia się plik decrypted\_NAZWA.jpg



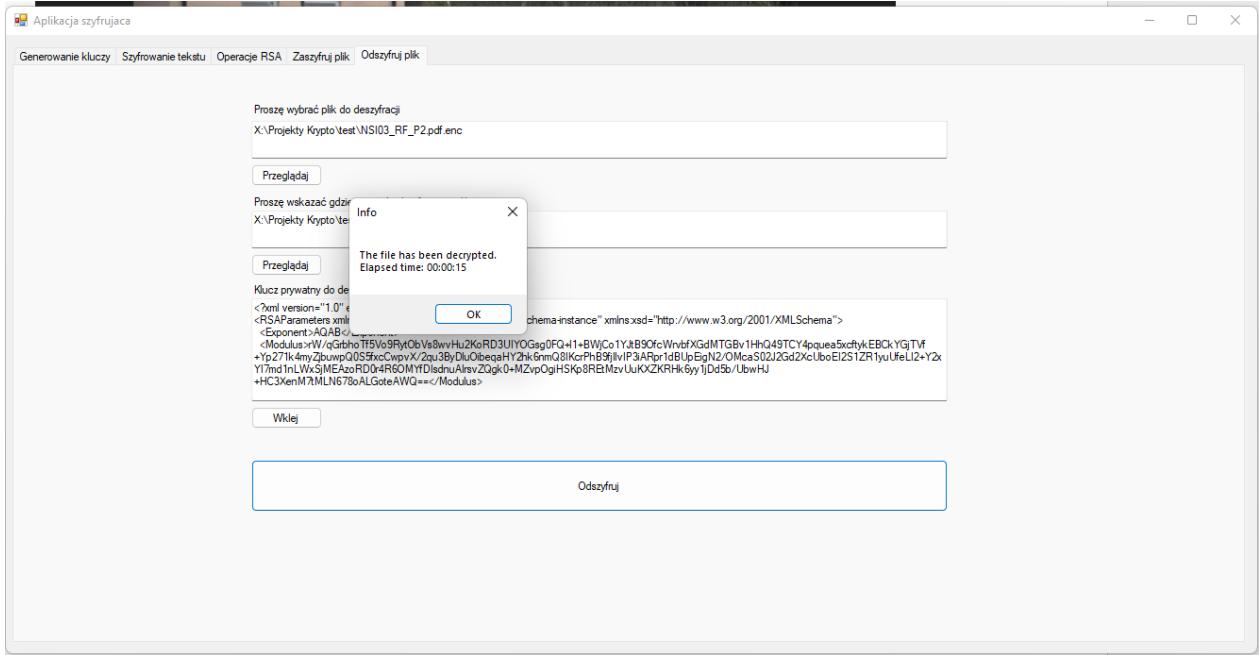
Przed odszyfrowaniem:



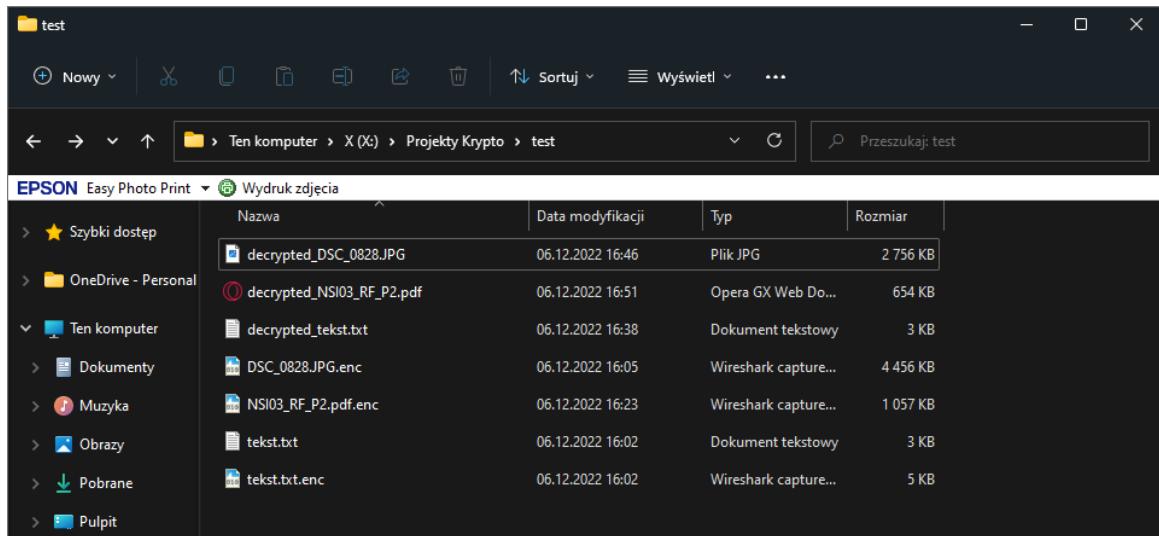
Po zaszyfrowaniu:

## Po odszyfrowaniu:

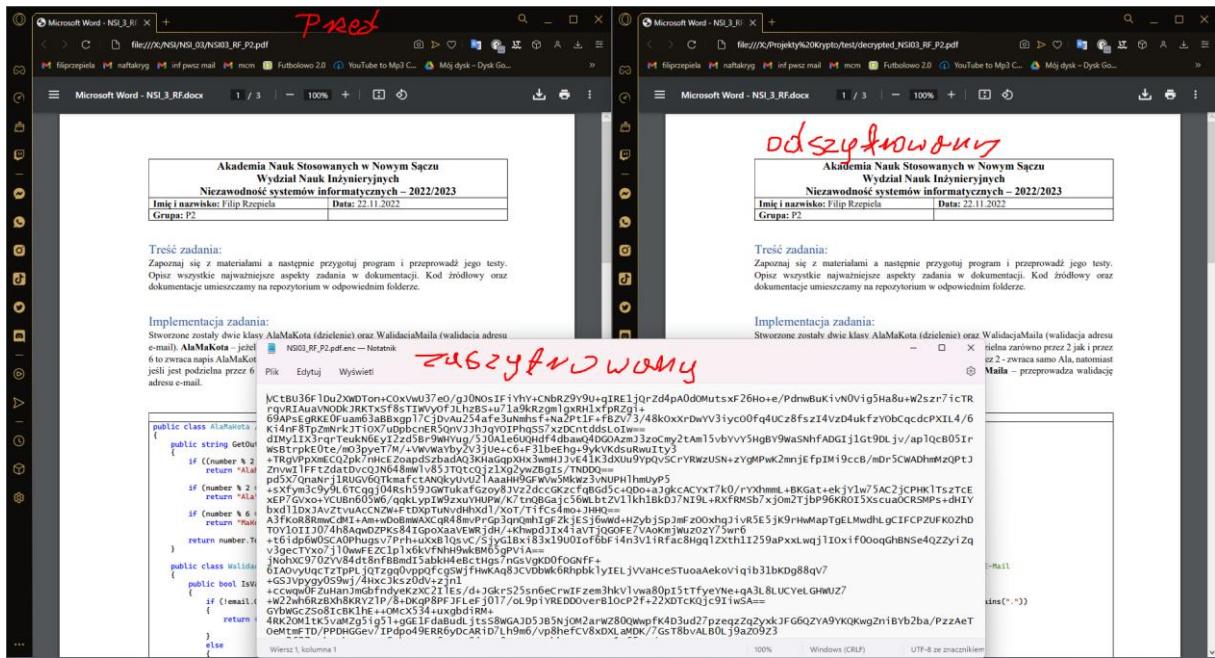




W folderze pojawia się plik decrypted\_NAZWA.pdf



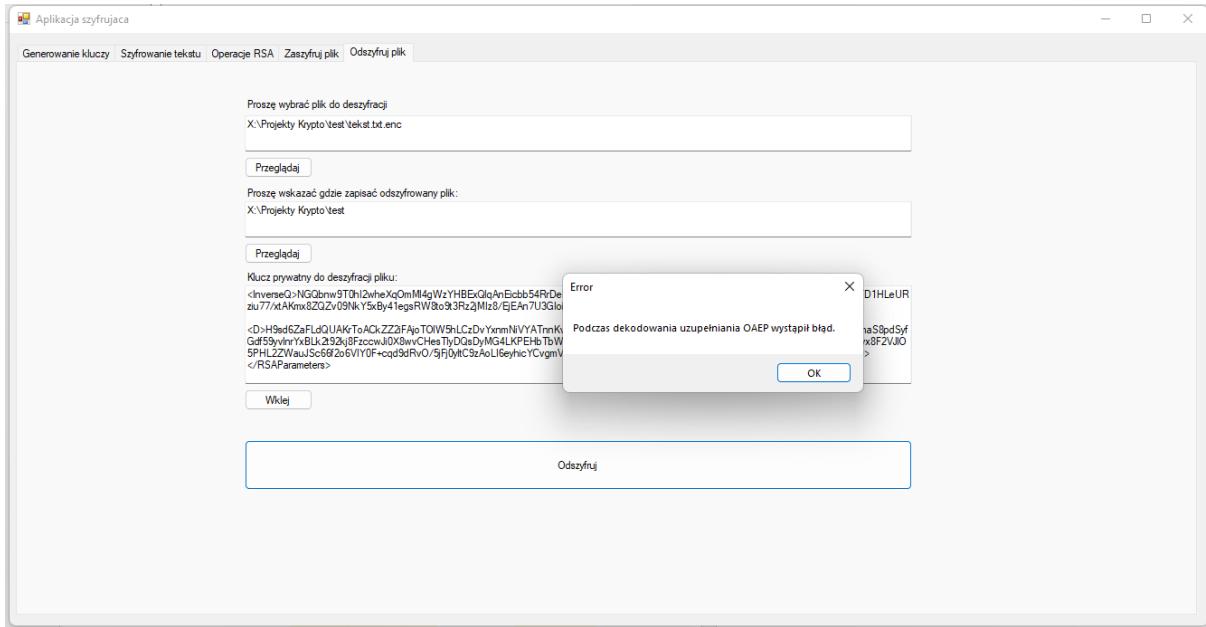
Porównanie przed zaszyfrowaniem, zaszyfrowany i odszyfrowany



### Zużycie zasobów podczas odszyfrowywania

Menedżer zadań									
Plik		Opcje		Widok					
				Procesy		Wydajność		Historia aplikacji	
Nazwa	Stan	53%	67%	0%	0%	3%	Aparat procesora GPU	Zużycie energii	Trend zużycia ...
RSA (32-bitowy)	14,9%	21,6 MB	0 MB/s	0 Mb/s	0%			Bardzo wysokie	Niski
Aplikacja szyfrująca									

## Test bezpieczeństwa – niepoprawny klucz prywatny



W przypadku wprowadzenia niepoprawnego klucza prywatnego program zwróci błąd

## Wnioski

- Interfejs graficzny jest łatwy oraz przyjemny w obsłudze.
- Aplikacja została zabezpieczona przed uruchomieniem, jeżeli pola są puste.
- Program działa relatywnie szybko.
- Aplikacja RSA w sposób poprawny dokonuje zaszyfrowania oraz odszyfrowania zarówno tekstu jak i plików. Atutem jest fakt, że nie powoduje uszkodzenia plików.
- Podczas przeprowadzanych testów Aplikacja RSA nie przestawała działać, nie wyrządzała błędów w systemie oraz nie wyłączała się.
- Mamy możliwość uruchomienia kilku okien aplikacji.