Programowanie Wieloplatformowe
Generator i menadżer haseł

Armand Pajor

Zrealizowane wymagania funkcjonalne:

- możliwość dodawania / edycji / usuwania pary login-hasło do zadanego serwisu
- wyświetlanie wszystkich przechowywanych haseł z możliwością ich tymczasowego ukrycia
- sortowanie haseł (najczęściej kopiowane będą znajdowały się na górze listy)
- czyszczenie listy haseł
- generowanie nowego hasła z możliwością wyboru jego długości i znaków
- zapis haseł do pliku
- odczyt haseł z pliku
- wyświetlenie informacji o autorze
- panel logowania
- kopiowanie hasła do schowka
- szyfrowanie i deszyfrowanie danych

Opis funkcji:

1. **Możliwość dodawania / edycji / usuwania pary login-hasło do zadanego serwisu**
   Program umożliwia zapis nowego zestawu danych: serwis, login i hasło. Hasło może być wymyślone przez użytkownika lub wygenerowane przez program.
   Każdy zestaw danych może być edytowany pod kątem loginu oraz hasła. W celu zmiany nazwy serwisu należy usunąć zestaw danych i stworzyć nowy.
   Usuwanie zestawu danych polega na wybraniu podanego serwisu i kliknięciu odpowiedniego przycisku.

2. **Wyświetlanie wszystkich przechowywanych haseł z możliwością ich tymczasowego ukrycia**
   Hasła pozostają ukryte do momentu świadomego ich odkrycia za pomocą kliknięcia odpowiedniego przycisku.
   Przy użyciu listy rozwijanej wybiera się serwis, do którego przypisany jest zestaw danych. Dane logowania wyświetlają się.

3. **Sortowanie haseł (najczęściej kopiowane będą znajdowały się na górze listy)**
   Po wybraniu zestawu danych i skopiowaniu hasła do schowka, akcja ta zostaje zarejestrowana, a w zestawie danych inkrementowany jest licznik użycia hasła. Na tej podstawie, hasła, które są najczęściej wykorzystywane, będą znajdowały się na początku listy.

4. **Czyszczenie listy haseł**
   Opcja usunięcia wszystkich zapisanych zestawów danych.

5. **Generowanie nowego hasła z możliwością wyboru jego długości i znaków**
   Odpowiednio opisane pola wyboru znaków, oraz lista rozwijana z wyborem długości hasła, pozwalają użytkownikowi zadecydować jak bardzo długie i złożone hasło potrzebuje.
   Zakres długości hasła to <6;32> znaki. Można je skonstruować z małych i wielkich liter alfabetu, cyfr i znaków specjalnych.
   Domyślne ustawienie to hasło składające się z 12 znaków: małych liter, dużych liter i cyfr.

6. **Zapis haseł do pliku**
   Wszystkie hasła są zapisywane do pliku w formacie JSON.
   Każdy użytkownik posiada oddzielny plik z zaszyfrowanymi zestawami danych logowania do serwisów użytkownika. Takie rozwiązanie pozwala na lepsze oddzielenie danych od użytkowników i zwiększenie bezpieczeństwa aplikacji.
   Pliki z zaszyfrowanymi zestawami danych logowania znajdują się pod ścieżką: „C:\ProgramData\MyPasswordManager\data*.json”.

7. **Odczyt haseł z pliku**
   Po zalogowaniu wczytany zostaje plik z zaszyfrowanymi zestawami danych logowania do serwisów użytkownika. W pamięci programu przechowywane są zaszyfrowane dane.

8. **Wyświetlenie informacji o autorze**
   Na panelu logowania widnieje przycisk „info”, który otwiera nowe okno z informacjami o: programie, twórcy, wersji programu i roku implementacji.

9. **Panel logowania**
   Panel logowania pozwala zarejestrować się jako nowy użytkownik lub zalogować po utworzeniu konta. Posiada metody do walidacji danych logowania.
   Dane zarejestrowanych użytkowników są przechowywane w oddzielnym pliku w formacie JSON. Login i hasło są hashowane algorytmem Keccak512, który zapewnia wysoki poziom bezpieczeństwa danych logowania.
   Plik z hashami danych logowania znajduje się pod ścieżką: „C:\ProgramData\MyPasswordManager\users.json”.

10. **Kopiowanie hasła do schowka**
    Przy wyborze zestawu danych istnieje możliwość skopiowana całego hasła do schowa za pomocą jednego przycisku.

11. **Szyfrowanie i deszyfrowanie danych**
    Zestawy danych logowania do serwisów są zaszyfrowane algorytmem AES256.
    Implementacja algorytmu AES pochodzi z: https://github.com/bricke/Qt-AES
    Każdy użytkownik programu posiada swój własny i niepowtarzalny zestaw klucza oraz wektora inicjalizującego. Takie rozwiązanie pozwala zabezpieczyć dane, aby użytkownicy nie uzyskali dostępu do danych innych użytkowników programu.
    Dane zostają deszyfrowane tyko w momencie zapytania o konkretny zestaw danych po wybraniu serwisu, dla którego taki zestaw istnieje.