Dokumentace SmichovPass

2023

Vypracoval: Kryštof Bruthans, 1.C

Obsah

[Požadavky 3](#_Toc133337026)

[Architektura a design programu 4](#_Toc133337027)

[Základní informace 4](#_Toc133337028)

[Ukládání záznamů 4](#_Toc133337029)

[Kryptografie 5](#_Toc133337030)

[Technická dokumentace 6](#_Toc133337031)

[Ovládací prvky programu a jejich metody 6](#_Toc133337032)

[Uživatelská dokumentace 7](#_Toc133337033)

[Budoucí funkce, rozvoj 8](#_Toc133337034)

# Požadavky

Pro správnou funkčnost programu jsou vyžadovány následující specifikace:

**OS:** Windows 10/11

**Vstupní zařízení:** myš, klávesnice

**Výstupní zařízení:** monitor

**Úložiště:** 21 kB volného místa

**Ostatní požadavky na hardware jsou shodné s minimálními požadavky na spuštění operačního systému Windows.**

*Pozn.: Toto jsou minimální prověřené požadavky, tudíž je možné spustit program např. i na starších verzích Windows, avšak není zaručené, že funkce programu budou pracovat správně.*

# Architektura a design programu

## Základní informace

Každý prvek programu je prvkem ze sady rozhraní Windows Forms. Užívané fonty v programu jsou dva: Bahnschrift a MS Sans Serif.

Každý prvek, který provádí funkci jinou než estetickou (tlačítka, textová pole, datapole) mají svůj kód definovaný v *App.Designer.cs*. Na metodu provedenou po kliknutí či jakékoli jiné interakci s ovládacím prvkem je odkázáno v designeru.

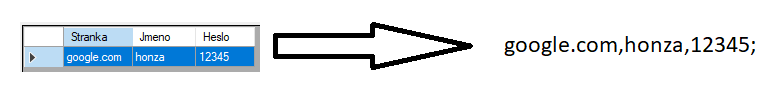
Tyto metody lze nalézt v *App.cs*. Každá taková metoda vždy nevrací hodnotu (její návratový typ – void).

## Ukládání záznamů

Každý záznam o stránce, jménu a heslu je uložený v objektu *record* (z angl. záznam). Ten obsahuje veřejné proměnné *Stranka, Jmeno, Heslo* typu string (řetězec). Tyto záznamy jsou uložené v seznamu *database* typu *List<record>*, ty jsou poté v metode *LoadDataTable* převedeny na formát *DataTable*, který je nahrán do objektu *PasswordViewer* typu *DataGridView*.

*Pozn. toto je z důvodu mnohých problémů při přímém nahrávání database do zobrazovače datapole DataGridView, který lépe pracuje se zdrojem typu DataTable.*

Před/po kryptografii je potřeba převést všechny záznamy do jednoho dlouhého řetězce a naopak. To je prováděno přímo v metodách *LoadButton\_Click* a *CreateButton\_Click*. Každá proměnná záznamu je od sebe oddělena čárkou, záznamy jsou oddělené středníkem.



Generovaný řetězec je dále v dokumentu označován jako „surová data záznamů“.

## Kryptografie

SmichovPass používá pro šifrování standard AES (Advanced Encryption Standard). Ještě před samotným je klíč zvolený uživatelem na hash pomocí algoritmu SHA256. Tento hash je používán jako opravdový klíč používaný při samotném šifrování (tento krok zajiš´tuje vyšší bezpečnost šifrovaného obsahu).

Před šifrováním je ještě vygenerován inicializační vektor (IV), který slouží jako bezpečnostní prvek (v praxi zajišťuje, že dvě data zašifrovaná stejným klíčem nejsou stejná kvůli náhodně vygenerovanému IV; tento IV je potřeba k následnému dešifrování, takže je obsažen v souboru .txt společně se zašifrovanými daty).

Poté je pomocí knihovny *System.Security.Cryptography* použit AES pro zašifrování surových dat záznamů. Výstupní zašifrovaná data jsou jeden řetězec.

V konečném kroku je k řetězci se zašifrovanými daty přidán středník a za ním IV použitý k zašifrování těchto dat. Tento výsledný řetězec je zapsán jako text do souboru .txt.

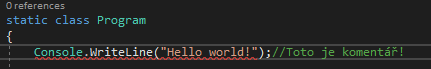
Dešifrování funguje obdobným způsobem, nejdříve je zahashován pomocí SHA256 klíč uživatele, poté od sebe oddělen IV a zašifrovaná data a pomocí algoritmu v knihovně *System.Security.Cryptography*  jsou data dešifrována na surová data, která jsou nahrána do *database* a metoda *LoadDataTable* data načte do datapole.

Obsah obrázku diagram

Popis byl vytvořen automaticky

# Technická dokumentace

Technická dokumentace se nachází přímo v kódu v podobě komentářů. Komentáře jsou vždy na jedné lince a začínají znaky „//“.

Příklad:

## Ovládací prvky programu a jejich metody

**Tlačítko *LoadButton*** po stisknutí zjistí jakékoli úpravy uživatelem do *dataTable* a zobrazí okno pro potvrzení klíče, poté inicializuje okno pro výběr souboru, ze kterého má číst. Po vybrání provede dešifrování pomocí AES a klíče zadaného uživatelem (*více v kapitole* ***Kryptografie***), násedně převede formát surových dat do záznamů typu *record* a ty nahraje do *database*.

**Tlačitíko *CreateButton*** po stisknutí zjistí jakékoli úpravy uživatelem do *dataTable* a zobrazí okno pro potvrzení klíče, poté inicializuje okno pro určení, kam má výsledný soubor zapsat. Nejdříve převede všechny záznamy *database* do surových dat, která zašifruje pomocí AES a klíče zadaného uživatelem (*více v kapitole* ***Kryptografie***). Výsledný řetězec zapíše do zvoleného souboru.

**Tlačítko *AddRecord*** po stisnutí přečte ze *StrankaBox, JmenoBox* a *HesloBox* text ve formě řetězců, incializuje nový záznam, který zaplní těmito řetězci a ty přidá do *database* a zavolá metodu *LoadDataTable*.

**Tlačítko *CopyButton*** po stisknutí zkopíruje heslo zvoleného záznamu v datapoli do schránky.

**Tlačítko *RemoveRecord*** po stisknutí smaže zvolený záznam v datapoli z *database* a zavolá metodu *LoadDataTable*.

**Metoda *LoadDataTable*** inicializuje dataTable nahraje každý záznam z *database* do něj a následně aktualizuje zobrazovač datapole(*PasswordViewer, typ DataGridView*).

**Objekt *PasswordViewer*** je typu *DataGridView* a zobrazuje data z *dataTable* uživateli.

# Uživatelská dokumentace

Děkujeme za používání programu SmichovPass!

## Vytváření hesel

1. Vyplňte pole „Název stránky“ – to slouží pro Vaší orientaci, abyste věděli, které přihlašovací údaje náleží jaké aplikaci/službě.
2. Vyplňte pole „Uživatelské pole“ – to slouží k identifikaci vašeho přihlašovacího jména k aplikaci/službě.
3. Vyplňte pole „Heslo“ – to je samotné heslo, které se společně s předchozími daty uloží do zabezpečeného, zašifrovaného formátu.
4. Klikněte na tlačítko „Přidat heslo“.

## Úprava hesel

Pro úpravu jakéhokoli hesla v datapoli stačí na požadovaný prvek kliknout dvakrát. To vám umožní upravovat jeho hodnotu.

# Budoucí funkce, rozvoj

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc tincidunt ante vitae massa. Etiam sapien elit, consequat eget, tristique non, venenatis quis, ante. Sed elit dui, pellentesque a, faucibus vel, interdum nec, diam. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Curabitur vitae diam non enim vestibulum interdum. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Integer malesuada. In laoreet, magna id viverra tincidunt, sem odio bibendum justo, vel imperdiet sapien wisi sed libero. Praesent id justo in neque elementum ultrices. Duis sapien nunc, commodo et, interdum suscipit, sollicitudin et, dolor. Duis viverra diam non justo. Pellentesque ipsum. Nulla est.