**Baranya Megyei Szakképzési Centrum**

**Simonyi Károly Technikum és Szakképző Iskola**

**OM azonosító:**  **203049**

**Szoftverfejlesztő és -tesztelő**

A szakma azonosítószáma: 5 0613 12 03

# Vizsgaremek készítés követelményrendszere

**Aliasly**

1. **Fejezet - bevezetés**

A digitális kor egyik legnagyobb kihívása a biztonságos adatkezelés. A felhasználók számos online szolgáltatást használnak, melyekhez egyedi és erős jelszavakra van szükség. A gyakori gyakorlat azonban**,** a könnyebb megjegyzés érdekében a jelszavak újrafelhasználása vagy egyszerű kombinációk alkalmazása, ami kompromittálja a fiókok biztonságát. Statisztikák szerint a biztonsági incidensek 80%-a gyenge jelszavakkal hozható összefüggésbe. A jelszókezelő alkalmazások ennek a problémának a megoldását szolgálják: központilag tárolják a titkos adatokat, titkosítva őrzik őket, és lehetővé teszik a felhasználók számára, hogy egyetlen mester**kulccsal** hozzáférjenek mindenhez.

Az „Aliasly” egy lokálisan futtatható, rendszergazdák számára tervezett jelszókezelő alkalmazás, amelyet **Németh Noel**, **Rapcsák János** és **Lőrincz Noel** alkotókból álló csapat fejlesztett a Visual Studioban, C# (C Sharp) nyelven és WPF (Windows Presentation Foundation) technológiával. Az alkalmazás fő célja, hogy biztonságos, offline környezetben tárolja a felhasználók hitelesítési adatait (pl. jelszavak, felhasználónevek), miközben egyszerű felületet kínál a kezelésükhöz.

A program használatához a felhasználóknak egy **mesterkulcsot** kell létrehozniuk, amelyet minden belépéskor meg kell adniuk. Ez a kulcs nemcsak belépési mechanizmus, hanem része a titkosítási folyamatnak is, biztosítva, hogy az adatok csak a jogosult felhasználó számára legyenek hozzáférhetők. Az adatokat egy helyi adatbázisban tárolja az alkalmazás, így nincs szükség internetkapcsolatra, és minimalizálja a külső támadások kockázatát.

Az Aliasly készítése során a csapat a következő eszközöket és technológiákat alkalmazta:

* **C# programozási nyelv**: A Microsoft által fejlesztett nyelv kiválóan alkalmas Windows-alapú alkalmazásokhoz, és gazdag könyvtárai támogatják a biztonságos kódolást.
* **WPF (Windows Presentation Foundation)**: Modern felhasználói felület kialakítására szolgáló keretrendszer, amely lehetővé teszi dinamikus designok és adatkötések használatát.
* **Visual Studio 2022**: Az integrált fejlesztői környezet (IDE) lehetővé tette a csapat számára a hatékony együttműködést, debuggolást és verziókezelést.
* **AES-256 titkosítás**: Az adatbázisban tárolt információk titkosításához ipari szabványnak számító algoritmust használtunk, amelyet a mesterkulcs erősít meg.
* **C# programozási nyelv**: A Microsoft által fejlesztett nyelv kiválóan alkalmas Windows-alapú alkalmazásokhoz, és gazdag könyvtárai támogatják a biztonságos kódolást.
* **WPF (Windows Presentation Foundation)**: Modern felhasználói felület kialakítására szolgáló keretrendszer, amely lehetővé teszi dinamikus designok és adatkötések használatát.
* **Visual Studio 2022**: Az integrált fejlesztői környezet (IDE) lehetővé tette a csapat számára a hatékony együttműködést, debuggolást és verziókezelést.
* **XAMPP és MAMP környezetek**: A helyi adattárolás és szerveroldali funkciók teszteléséhez a csapat az **XAMPP** (Windows) és **MAMP** (macOS) fejlesztői környezeteket használta.
* **AES-256 titkosítás**: Az adatbázisban tárolt információk titkosításához ipari szabványnak számító algoritmust használtunk, amelyet a mesterkulcs erősít meg.

Az Aliasly elsősorban **rendszergazdáknak** és informatikai szakembereknek készült, akiknek nagy mennyiségű hitelesítési adatot kell kezelniük kritikus rendszerekhez (pl. vállalati hálózatok, szerverek, adatközpontok).

A lokális futtatás előnye:

* Nincs függőség internetkapcsolattól vagy külső szolgáltatóktól.
* Teljes kontroll az adatok felett, nem kerülnek fel felhőbe.
* Ideális olyan környezetekben, ahol szigorú adatvédelmi előírások érvényesek.

Az Aliasly fejlesztésének fő céljai:

* **Biztonság első helyen**: A mesterkulcs és az AES-256 titkosítás kombinációja garantálja, hogy az adatok akkor is védve maradjanak, ha a rendszer fizikailag illetéktelen kezekbe kerül.
* **Felhasználóbarát felület**: A WPF által lehetővé tett modern UI tervezési lehetőségek kihasználása, hogy a komplex funkciók is könnyen kezelhetőek legyenek.

Az Aliasly csapata úgy tervezte, hogy az alkalmazás hosszú távon is megbízható maradjon, és alkalmazkodjon a felhasználók igényeihez. A dokumentáció további fejezeteiben bemutatjuk, hogyan valósítottuk meg ezeket az ambiciózus célokat, és milyen leckéket vontunk le a fejlesztés során.

Ez a dokumentáció részletesen bemutatja az Aliasly fejlesztési folyamatát, architektúráját és biztonsági mechanizmusait.

A következő fejezetekben részletesen taglaljuk: (atirasra szorul)

* A rendszer követelményeit és tervezési elképzeléseit.
* A felhasználói felület működését és felépítését.
* A titkosítási folyamat technikai részleteit.
* Az adatbázis szerkezetét és kezelését.
* A tesztelési fázis eredményeit és a jövőbeli fejlesztési lehetőségeket.

1. Fejezet - asd